

ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΠΟΛΙΤΙΣΜΟΥ

ΠΛΗΡΟΦΟΡΙΚΗ

Γ' ΓΥΜΝΑΣΙΟΥ

Σημειώσεις

Πληροφορική και Επιστήμη Ηλεκτρονικών Υπολογιστών

Γ΄ Γυμνασίου

Σημειώσεις

Συγγραφή:

Οι καθηγητές Πληροφορικής και Επιστήμης Ηλεκτρονικών Υπολογιστών που αναφέρονται στην αρχή της κάθε ενότητας με την καθοδήγηση και συμβολή των ακόλουθων Συμβούλων Καθηγητών Πληροφορικής και Επιστήμης Ηλεκτρονικών Υπολογιστών:

Α΄ Έκδοση

Σωκράτης Μυλωνάς (ΒΔ)
Νικόλαος Ζάγγουλος
Αλέξανδρος Παπαλυσάνδρου (ΒΔ)
Μιχάλης Διονυσίου (ΒΔ)
Πάυλος Παυλικκάς
Ξένιος Ξενοφώντος
Ιωάννης Ιωάννου
Μιλτιάδης Χαριλάου
Ηλίας Θεοδώρου

Β΄ και Γ΄ Έκδοση

Κωνσταντίνος Σωφρονίου (ΒΔ)
Πάυλος Παυλικκάς
Πόλα Μάκκουλα

Δ΄ Έκδοση

Πόλα Μάκκουλα
Δημήτρης Χατζηπαντελής
Νικόλας Στρατής

Ηλεκτρονική σελίδωση:

Πόλα Μάκκουλα
Καθηγήτρια Πληροφορικής και Επιστήμης Ηλεκτρονικών Υπολογιστών

Εποπτεία:

Μάριος Μιλτιάδου
Μιχάλης Τορτούρης
Σωκράτης Μυλωνάς
Μιχάλης Φαραζής
Επιθεωρητές Πληροφορικής και Επιστήμης Ηλεκτρονικών Υπολογιστών

Γλωσσική επιμέλεια:

Αρετή Στυλιανού
Λειτουργός Υπηρεσίας Ανάπτυξης Προγραμμάτων

Σχεδιασμός εξωφύλλου:

Μιχάλης Θεοχαρίδης
Λειτουργός Υπηρεσίας Ανάπτυξης Προγραμμάτων

Συντονισμός έκδοσης:

Χρίστος Παρπούνας
Συντονιστής Υπηρεσίας Ανάπτυξης Προγραμμάτων

Α΄ Έκδοση 2013
Ανατύπωση 2016 (Αναθεωρημένη)
Β΄ Έκδοση 2018
Γ΄ Έκδοση 2019
Δ΄ Έκδοση 2022

Εκτύπωση: GREENPACK GRAPHIC ARTS LTD

© ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΠΟΛΙΤΙΣΜΟΥ
ΠΑΙΔΑΓΩΓΙΚΟ ΙΝΣΤΙΤΟΥΤΟ ΚΥΠΡΟΥ
ΥΠΗΡΕΣΙΑ ΑΝΑΠΤΥΞΗΣ ΠΡΟΓΡΑΜΜΑΤΩΝ

ISBN: 978-9963-54-310-6



Στο εξώφυλλο χρησιμοποιήθηκε ανακυκλωμένο χαρτί σε ποσοστό τουλάχιστον 50%, προερχόμενο από διαχείριση απορριμμάτων χαρτιού. Το υπόλοιπο ποσοστό προέρχεται από υπεύθυνη διαχείριση δασών.

Πρόλογος

Με ιδιαίτερη χαρά και ικανοποίηση προλογίζω το έντυπο υλικό για την υποστήριξη της διδασκαλίας του μαθήματος της Πληροφορικής και της Επιστήμης των Ηλεκτρονικών Υπολογιστών στο Γυμνάσιο.

Το υλικό αυτό είναι διαχωρισμένο σε δύο τόμους, τις *Σημειώσεις* και το *Τετράδιο*. Ο κάθε τόμος είναι οργανωμένος σε επτά ενότητες. Κάθε ενότητα καλύπτει το φάσμα των γνώσεων, των ικανοτήτων και των δεξιοτήτων που προκύπτουν από το γενικό σκοπό του μαθήματος, να προετοιμάσει, δηλαδή, τους μαθητές και τις μαθήτριες για την ένταξή τους στην Κοινωνία της Πληροφορίας. Τους παρέχει ικανοποιητικές γνώσεις και τους καλλιεργεί τις απαραίτητες ικανότητες, δεξιότητες και στάσεις που θα τους επιτρέψουν την υπεύθυνη, την ενσυνείδητη, την ασφαλή, την αποδοτική και τη δημιουργική χρήση σύγχρονων τεχνολογιών της Πληροφορικής και της Επιστήμης των Ηλεκτρονικών Υπολογιστών.

Οι *Σημειώσεις* δεν αποτελούν διδακτικό εγχειρίδιο αλλά υλικό αναφοράς, το οποίο περιέχει παραδείγματα, επεξηγήσεις και επιπρόσθετες πληροφορίες και χρησιμοποιείται ως πηγή άντλησης πληροφοριών. Στην αρχή του κάθε κεφαλαίου διατυπώνονται οι δείκτες επιτυχίας και στο τέλος υπάρχει σύνοψη των βασικών εννοιών για εύκολη αναφορά.

Το *Τετράδιο* αποτελείται από ενδεικτικά φύλλα εργασίας, διαχωρισμένα σε δραστηριότητες εμπέδωσης για την ανάπτυξη νέων γνώσεων, ικανοτήτων και δεξιοτήτων και αξιολόγησης του βαθμού απόκτησης των γνώσεων αυτών, σε επιπρόσθετες δραστηριότητες για εμπάθυνση και διεύρυνση των γνώσεων, των ικανοτήτων και των δεξιοτήτων καθώς και σε δραστηριότητες για το σπίτι. Περιλαμβάνονται, περαιτέρω, ποικίλες δραστηριότητες με διαφοροποιημένο δείκτη δυσκολίας, ώστε να είναι δυνατή η προσαρμογή του μαθήματος τις ικανότητες, τις απαιτήσεις και στις ανάγκες όλων των μαθητών και των μαθητριών με γνώμονα τους δείκτες επιτυχίας του Αναλυτικού Προγράμματος. Υιοθετούνται σύγχρονα διδακτικά μοντέλα, που στηρίζονται στην προώθηση εξελιγμένων μεθόδων οικοδόμησης της γνώσης. Βασικό στοιχείο της διδακτικής προσέγγισης είναι η σύνδεση του μαθήματος με την καθημερινότητα των μαθητών και των μαθητριών με απώτερο στόχο την ανάπτυξη της δημιουργικής και κριτικής σκέψης και την καλλιέργεια δεξιοτήτων επικοινωνίας, συνεργασίας και συστηματικών τρόπων προσέγγισης για την επίλυση προβλημάτων, ώστε να είναι έτοιμοι/ες για να λειτουργήσουν στο συνεχώς μεταβαλλόμενο περιβάλλον των τεχνολογιών της πληροφορίας και των επικοινωνιών.

Για τη δημιουργία του υλικού αυτού αλλά και του ηλεκτρονικού υλικού που το συνοδεύει, η ομάδα υποστήριξης της εφαρμογής του Νέου Αναλυτικού Προγράμματος για το μάθημα της Πληροφορικής και της Επιστήμης των Ηλεκτρονικών Υπολογιστών έχει υιοθετήσει ένα καινοτόμο μοντέλο: το υλικό δημιουργήθηκε από τους υπό επιμόρφωση μάχιμους εκπαιδευτικούς, υπό την καθοδήγηση των υποστηρικτών και μέσα από τις εισηγήσεις τους αυτό διαμορφώνεται και αναθεωρείται συνεχώς.

Επιθυμώ να εκφράσω τις ευχαριστίες μου στην ομάδα επιμόρφωσης και υποστήριξης, στην Υπηρεσία Ανάπτυξης Προγραμμάτων αλλά και στους/στις μάχιμους/ες εκπαιδευτικούς για τη συμβολή τους στην ανάπτυξη του υλικού αυτού.

Δρ Κυπριανός Δ. Λούης

Διευθυντής Μέσης Γενικής Εκπαίδευσης

ΠΕΡΙΕΧΟΜΕΝΑ

| | |
|-----------------------------------------------------------------------------------------------------|-----------|
| Εισαγωγή | 11 |
| ΕΝΟΤΗΤΑ Γ1 Βασικές Έννοιες της Πληροφορικής και της Επιστήμης Ηλεκτρονικών Υπολογιστών | 13 |
| Γ1.1 Αναπαράσταση Δεδομένων και Χωρητικότητα Μονάδων Αποθήκευσης | 15 |
| 1. Αναπαράσταση Δεδομένων – Δυαδικό Σύστημα | 15 |
| 2. Αναπαράσταση Χαρακτήρων με Δυαδικά Ψηφία | 15 |
| 3. Μονάδα Μέτρησης της Χωρητικότητας των Μονάδων Αποθήκευσης..... | 15 |
| 4. Μετατροπή Από Ένα Πολλαπλάσιο του Byte σε Άλλο..... | 16 |
| ΕΝΟΤΗΤΑ Γ2 Το Υλικό/ Αρχιτεκτονική Ηλεκτρονικών Υπολογιστών | 19 |
| Γ2.1 Βασικές έννοιες της Αρχιτεκτονικής του Η/Υ | 21 |
| 1. Το εσωτερικό του Η/Υ..... | 21 |
| 2. Το Τροφοδοτικό (Power Supply)..... | 22 |
| 3. Η Μητρική Κάρτα ή Μητρική Πλακέτα (Motherboard) | 22 |
| 4. Η Κεντρική Μονάδα Επεξεργασίας (ΚΜΕ, Central Processing Unit, CPU)..... | 23 |
| 5. Η Μνήμη Τυχαίας Προσπέλασης –Random Access Memory (RAM)..... | 23 |
| 6. Η Μνήμη Μόνο για Ανάγνωση–Read Only Memory (ROM) | 23 |
| 7. Οι Υποδοχές Επέκτασης (expansion slots) και οι Κάρτες Επέκτασης (expansion cards).... | 24 |
| 8. Θύρες σύνδεσης (Ports) | 25 |
| Γ2.2 Η Κεντρική Μονάδα Επεξεργασίας (ΚΜΕ) | 29 |
| 1. Η Κεντρική Μονάδα Επεξεργασίας, ΚΜΕ (Central Processing Unit, CPU)..... | 29 |
| 2. Τα Μέρη της ΚΜΕ | 29 |
| 3. Ο Χρονοστάτης (Clock)..... | 31 |
| 4. Κύρια χαρακτηριστικά μιας ΚΜΕ..... | 31 |
| Γ2.3 Κύρια και Βοηθητική Μνήμη | 35 |
| 1. Μνήμη | 35 |
| 2. Η Κύρια Μνήμη του υπολογιστή..... | 35 |
| 3. Η Βοηθητική Μνήμη..... | 37 |
| 4. Ο σκληρός δίσκος (Hard Disk Drive) και ο δίσκος SSD (Solid State Drive)..... | 38 |
| 5. Οι Οπτικοί δίσκοι (CD, DVD, CD-R/RW, DVD-R/RW) | 39 |
| 6. Αποθηκευτικά μέσα τύπου Μνήμης Φλας (USB, SD) | 39 |
| 7. Σύνοψη..... | 40 |
| Γ2.4 Εκτυπωτές | 43 |
| 1. Εισαγωγή | 43 |
| 2. Είδη εκτυπωτών | 43 |
| 3. Χαρακτηριστικά εκτυπωτών..... | 44 |

ΠΕΡΙΕΧΟΜΕΝΑ

| | | |
|-------------------|-----------------------------------------------------------------------------------------|-----------|
| Γ2.5 | Οθόνες..... | 51 |
| 1. | Εισαγωγή..... | 51 |
| 2. | Πώς λειτουργεί μια οθόνη | 51 |
| 3. | Είδη Οθόνης | 51 |
| 4. | Χαρακτηριστικά οθονών..... | 53 |
| Γ2.6 | Πώς Συγκρίνουμε και Πώς Επιλέγουμε Υπολογιστικά Συστήματα | 57 |
| 1. | Εισαγωγή..... | 57 |
| 2. | Ο χρήστης και οι υπολογιστικές ανάγκες του | 57 |
| 3. | Επιλογή λογισμικού εφαρμογών και λειτουργικού συστήματος | 58 |
| 4. | Επιλογή υλικού..... | 59 |
| 5. | Πώς επιλέγουμε ένα υπολογιστικό σύστημα για κάποιον χρήστη | 59 |
| ΕΝΟΤΗΤΑ Γ3 | Λειτουργικά Συστήματα | 67 |
| Γ3.1 | Λειτουργικά Συστήματα – Λειτουργίες και Παραδείγματα..... | 69 |
| 1. | Λειτουργικό Σύστημα..... | 69 |
| 2. | Βασικές Λειτουργίες Λειτουργικών Συστημάτων | 69 |
| 3. | Σχέση Λειτουργικού Συστήματος και Λογισμικού Εφαρμογών | 70 |
| 4. | Τα Βασικότερα Λειτουργικά Συστήματα..... | 70 |
| ΕΝΟΤΗΤΑ Γ6 | Βάσεις Δεδομένων και Ανάπτυξη Πληροφοριακών Συστημάτων | 75 |
| Γ6.1 | Εισαγωγή στις Βάσεις Δεδομένων | 77 |
| 1. | Βάση Δεδομένων (Database) | 77 |
| 2. | Ηλεκτρονικές (μηχανογραφημένες) βάσεις δεδομένων | 77 |
| 3. | Πλεονεκτήματα των μηχανογραφημένων βάσεων δεδομένων | 78 |
| 4. | Μειονέκτημα των μηχανογραφημένων βάσεων δεδομένων | 79 |
| 5. | Βάσεις Δεδομένων από την καθημερινή μας ζωή | 79 |
| Γ6.2 | Βασικά Δομικά Συστατικά Στοιχεία και Κύρια Αντικείμενα μιας Βάσης Δεδομένων..... | 81 |
| 1. | Βασικά Δομικά Συστατικά Στοιχεία μιας Βάσης Δεδομένων | 81 |
| 2. | Ειδικές Σχέσεις μεταξύ πινάκων (Relationships) | 82 |
| 3. | Κύρια Αντικείμενα Βάσης Δεδομένων (Objects) | 84 |
| 4. | Η Δομή μιας Βάσης Δεδομένων | 84 |
| 5. | Εφαρμογή Βάσεων Δεδομένων (Microsoft Access) | 84 |
| 6. | Ξεκίνημα της Εφαρμογής Βάσεων Δεδομένων | 85 |
| 7. | Δημιουργία νέας κενής βάσης δεδομένων (Blank database)..... | 85 |
| 8. | Αποθήκευση βάσης δεδομένων (Save)..... | 85 |
| 9. | Άνοιγμα βάσης δεδομένων (Open)..... | 86 |
| 10. | Κλείσιμο βάσης δεδομένων (Close Database) | 86 |
| 11. | Κλείσιμο Εφαρμογής Βάσεων Δεδομένων (Exit) | 86 |
| Γ6.3 | Πίνακες..... | 89 |
| 1. | Πίνακας (Table)..... | 89 |

| | | |
|-------------|-------------------------------------------------------------------------------------------------|------------|
| 2. | Δημιουργία Πίνακα με τη χρήση της Προβολής Σχεδίασης (Table Design)..... | 89 |
| 3. | Δημιουργία Πίνακα με τη χρήση της Προβολής Φύλλου Δεδομένων (Table)..... | 90 |
| 4. | Κανόνες Ονομασίας Πεδίων | 91 |
| 5. | Τύποι Δεδομένων (Data Type)..... | 91 |
| 6. | Ιδιότητες Πεδίων (Field Properties)..... | 92 |
| 7. | Πρωτεύον Κλειδί (Primary Key)..... | 93 |
| 8. | Πώς καθορίζουμε το Πρωτεύον Κλειδί | 95 |
| 9. | Διαγραφή Πίνακα (Delete Table)..... | 95 |
| 10. | Προβολές για Πίνακα (Views)..... | 95 |
| Γ6.4 | Εγγραφές..... | 99 |
| 1. | Εισαγωγή, Επεξεργασία, Διαγραφή και Μετακίνηση μεταξύ των εγγραφών σε ένα πίνακα..... | 99 |
| 2. | Μορφοποίηση Πίνακα..... | 100 |
| 3. | Ταξινόμηση Δεδομένων (Sort) | 101 |
| 4. | Χρήση της Εντολής για την Αναζήτηση Πληροφοριών (Find)..... | 102 |
| 5. | Χρήση Φίλτρων για την Αναζήτηση Πληροφοριών (Filter)..... | 103 |
| Γ6.5 | Σχέσεις Πινάκων | 109 |
| 1. | Εισαγωγή | 109 |
| 2. | Πλεονεκτήματα που προκύπτουν από τις Σχέσεις μεταξύ των Πινάκων..... | 110 |
| 3. | Ο ρόλος του Πρωτεύοντος Κλειδιού στον συσχετισμό των πινάκων..... | 110 |
| 4. | Είδη Ειδικών Σχέσεων (Relationships) | 111 |
| Γ6.6 | Δημιουργία και Διαγραφή Σχέσεων Μεταξύ δύο Πινάκων | 115 |
| 1. | Εισαγωγή | 115 |
| 2. | Δημιουργία σχέσης Ένα-Προς-Πολλά μεταξύ δύο πινάκων (One-To-Many, 1-∞) | 115 |
| 3. | Διαγραφή Σχέσης μεταξύ δύο πινάκων (Delete Relationship) | 117 |
| 4. | Καταχώριση Εγγραφών στον δεύτερο συσχετιζόμενο πίνακα | 117 |
| Γ6.7 | Φόρμες Εισαγωγής Δεδομένων..... | 119 |
| 1. | Φόρμα Εισαγωγής Δεδομένων (Form) | 119 |
| 2. | Δημιουργία Φόρμας με το Εργαλείο Αυτόματης Φόρμας (Form) | 119 |
| 3. | Δημιουργία Φόρμας με τη χρήση του Οδηγού Φορμών (Form Wizard) | 119 |
| 4. | Εισαγωγή, Επεξεργασία και Μετακίνηση μεταξύ των εγγραφών σε μια φόρμα εισαγωγής δεδομένων | 120 |
| 5. | Διαγραφή Εγγραφής σε μια φόρμα (Delete Record) | 120 |
| 6. | Προβολές για Φόρμα (Views)..... | 120 |
| 7. | Μορφοποίηση Φόρμας | 122 |
| Γ6.8 | Ερωτήματα | 129 |
| 1. | Ερωτήματα (Queries)..... | 129 |
| 2. | Δημιουργία Ερωτήματος με τη χρήση του Οδηγού Ερωτημάτων (Query Wizard) | 130 |
| 3. | Δημιουργία Ερωτήματος με τη χρήση της Προβολή Σχεδίασης (Query Design) | 130 |
| 4. | Τροποποίηση Ερωτήματος..... | 131 |

ΠΕΡΙΕΧΟΜΕΝΑ

| | | |
|-------------------|---------------------------------------------------------------------------------------------------------------------|------------|
| 5. | Διαγραφή Ερωτήματος (Delete Query) | 132 |
| 6. | Το Παράθυρο της Προβολής Σχεδίασης ενός Ερωτήματος | 132 |
| 7. | Κριτήρια Ερωτημάτων (Criteria) | 133 |
| Γ6.9 | Ερωτήματα από Δύο Συσχετιζόμενους Πίνακες..... | 137 |
| 1. | Ερωτήματα από δύο Συσχετιζόμενους Πίνακες (Queries in Related Tables) | 137 |
| 2. | Δημιουργία ερωτήματος με τη χρήση του Οδηγού Ερωτημάτων (Query Wizard) | 137 |
| 3. | Δημιουργία ερωτήματος με τη χρήση της Προβολής Σχεδίασης (Query Design) | 138 |
| Γ6.10 | Εκθέσεις..... | 141 |
| 1. | Εκθέσεις (Reports) | 141 |
| 2. | Δημιουργία Έκθεσης με το Εργαλείο Αυτόματης Έκθεσης (Report) | 141 |
| 3. | Δημιουργία Έκθεσης με τη χρήση του Οδηγού Εκθέσεων (Report Wizard) | 142 |
| 4. | Προβολές για Έκθεση (Views) | 142 |
| 5. | Τροποποίηση Έκθεσης..... | 144 |
| 6. | Διαγραφή Έκθεσης (Delete Report) | 144 |
| 7. | Εκτύπωση Έκθεσης (Print Report) | 144 |
| Γ6.11 | Επιπρόσθετες Λειτουργίες | 147 |
| 1. | Ελαχιστοποίηση κορδέλας (ribbon) | 147 |
| 2. | Επαναφορά κορδέλας (ribbon) | 147 |
| 3. | Εμφάνιση και απόκρυψη των γραμμών εργαλείων | 147 |
| 4. | Χρήση της Βοήθειας (Help) | 147 |
| 5. | Εξαγωγή στοιχείων πίνακα ή ερωτήματος | 148 |
| 6. | Αλλαγή του προσανατολισμού της σελίδας, των περιθωρίων και του μεγέθους του χαρτιού | 148 |
| 7. | Εκτύπωση συγκεκριμένων εγγραφών..... | 148 |
| ΕΝΟΤΗΤΑ Γ7 | Αλγοριθμική Σκέψη, Προγραμματισμός και Σύγχρονες Εφαρμογές Πληροφορικής..... | 151 |
| Γ7.1 | Εισαγωγή στους Αλγορίθμους..... | 153 |
| 1. | Ο Κύκλος Ανάπτυξης Εφαρμογής – Ποια βήματα/φάσεις πρέπει να ακολουθήσουμε για να δημιουργήσουμε ένα πρόγραμμα | 153 |
| 2. | Τι είναι αλγόριθμος..... | 156 |
| 3. | Τα χαρακτηριστικά ενός αλγορίθμου | 157 |
| 4. | Τρόποι διατύπωσης αναπαράστασης αλγορίθμων | 159 |
| 5. | Αποδοτικός Αλγόριθμος | 159 |
| Γ7.2 | Αναπαράσταση Αλγορίθμων με Λογικό Διάγραμμα | 163 |
| 1. | Λογικά Διαγράμματα | 163 |
| 2. | Δεδομένα, Επεξεργασία, Πληροφορίες (Ζητούμενα), | 164 |
| 3. | Μεταβλητές | 164 |
| 4. | Αναπαράσταση αλγορίθμου με τη χρήση λογικών διαγραμμάτων | 164 |
| Γ7.3 | Προβλήματα Ακολουθιακής Δομής | 169 |
| 1. | Ακολουθιακή Δομή | 169 |

| | | |
|-------------|-------------------------------------------------------------------------|------------|
| 2. | Προκαταρκτική Εκτέλεση | 169 |
| 3. | Μαθηματικές εκφράσεις | 170 |
| 4. | Προβλήματα με μία (1) είσοδο και έξοδο | 170 |
| 5. | Προβλήματα με μία (1) είσοδο και περισσότερες εξόδους | 171 |
| 6. | Προβλήματα με περισσότερες εισόδους και 1 έξοδο | 172 |
| 7. | Προβλήματα με περισσότερες εισόδους και εξόδους..... | 173 |
| Γ7.4 | Προβλήματα Δομής Διακλάδωσης | 177 |
| 1. | Δομή διακλάδωσης ή Δομή επιλογής | 177 |
| Γ7.5 | Γλώσσες Προγραμματισμού..... | 187 |
| 1. | Εισαγωγή | 187 |
| 2. | Γλώσσες Προγραμματισμού | 187 |
| Γ7.6 | Η Γλώσσα Προγραμματισμού Python | 189 |
| 1. | Εισαγωγή | 189 |
| 2. | Ενεργοποίηση του ολοκληρωμένου περιβάλλοντος ανάπτυξης της Python | 189 |
| 3. | Δημιουργία νέου αρχείου..... | 190 |
| 4. | Αποθήκευση αρχείου..... | 190 |
| 5. | Άνοιγμα αρχείου | 190 |
| 6. | Εκτέλεση προγράμματος | 190 |
| 7. | Δημιουργία Προγράμματος | 190 |
| 8. | Σφάλματα | 193 |
| 9. | Ακολουθιακή Δομή..... | 193 |
| 10. | Η εντολή if...else | 194 |

Εισαγωγή

Όταν τον Μάρτιο του 2009 ξεκίνησε η προσπάθεια για εκπόνηση του Νέου Αναλυτικού Προγράμματος του μαθήματος της Πληροφορικής και Επιστήμης των Ηλεκτρονικών Υπολογιστών, τέθηκε ως προτεραιότητα η θεώρηση του μαθήματος μέσα από το πρίσμα των τριών πυλώνων της εκπαιδευτικής μεταρρύθμισης. Μέσα από το πρίσμα, δηλαδή, της απόκτησης ενός επαρκούς σώματος γνώσεων από τους μαθητές και τις μαθήτριες, την καλλιέργεια αξιών και την εκδήλωση συμπεριφορών που να συνάδουν με τη σύγχρονη έννοια της δημοκρατικής πολιτότητας και την καλλιέργεια κομβικών ικανοτήτων και δεξιοτήτων από τους μαθητές και τις μαθήτριες, ώστε να μπορούν να λειτουργούν στη διαμορφούμενη κοινωνία του 21^{ου} αιώνα.

Στην επιτροπή για την εκπόνηση του Αναλυτικού Προγράμματος συμμετείχαν ο ακαδημαϊκός Ανδρέας Ανδρέου (Αναπληρωτής Καθηγητής στο Τεχνολογικό Πανεπιστήμιο Κύπρου), οι επιθεωρητές Πληροφορικής Μάριος Μιλτιάδου και Μιχάλης Τορτούρης και οι καθηγητές Πληροφορικής Νικόλαος Ζάγγουλος, Ιωάννης Ιωάννου, Μηνάς Καραολής, Θεόδουλος Κωνσταντίνου, Μαρία Μαρδά, Σωκράτης Μυλωνάς και Παναγιώτης Παπέττας. Αποτέλεσμα της προσπάθειας που έγινε είναι το Αναλυτικό Πρόγραμμα που ήδη έχετε στα χέρια σας.

Το νέο Αναλυτικό Πρόγραμμα δεν περιέχει μεγάλο αριθμό νέων θεμάτων. Έγινε αναδιάρθρωση της θεματολογίας, ώστε να ενταχθεί κάτω από επτά ενότητες, οι οποίες αναπτύσσονται παράλληλα από τη μια τάξη στην επόμενη και αντικατοπτρίζουν βασικούς θεματικούς άξονες της Πληροφορικής και Επιστήμης των Ηλεκτρονικών Υπολογιστών:

- Βασικές Έννοιες
- Το Υλικό/Αρχιτεκτονική Υπολογιστών
- Λειτουργικά Συστήματα
- Λογισμικό Εφαρμογών
- Δίκτυα και Διαδίκτυο
- Βάσεις Δεδομένων και Ανάλυση Συστημάτων
- Αλγόριθμοι, Προγραμματισμός και Σύγχρονες Εφαρμογές Πληροφορικής.

Οι ενότητες αυτές αναπτύσσονται σταδιακά μέσα από διαβαθμισμένους στόχους/δείκτες επιτυχίας από τη μία τάξη στην επόμενη, παρέχοντας επιπρόσθετες ευκαιρίες στους μαθητές και στις μαθήτριες που δεν μπόρεσαν να τους κατακτήσουν σε μία τάξη να το πράξουν σε επόμενη, με παράλληλη εμβάθυνση. Θα πρέπει εδώ να τονιστεί ότι κάθε ενότητα αναπτύσσεται σε διαφορετικό βαθμό στην κάθε τάξη, με γνώμονα το γενικό επίπεδο ετοιμότητας των μαθητών/τριών στη συγκεκριμένη ηλικιακή ομάδα. Σημαντικό στοιχείο της νέας προσέγγισης είναι και η αλληλεξάρτηση των ενοτήτων, αλλά και η συγκεκριμενοποίηση και ο περιορισμός των εννοιών που θα πρέπει να γνωρίζουν οι μαθητές/τριες, με έμφαση στην εφαρμογή τους στην πράξη.

Τα ενδεικτικά φύλλα εργασίας που ακολουθούν βασίστηκαν σε διδακτικές προτάσεις που ετοιμάστηκαν από μάχιμους εκπαιδευτικούς και ορισμένα από αυτά δοκιμάστηκαν σε περιβάλλον πραγματικής τάξης. Παρέχονται ως παραδείγματα διάφορων μεθοδολογικών προσεγγίσεων, αλλά οι καθηγητές/τριες ενθαρρύνονται να τα προσαρμόσουν ή και να δημιουργήσουν άλλα, τα οποία να ανταποκρίνονται καλύτερα, τόσο στα ιδιαίτερα χαρακτηριστικά των μαθητών και μαθητριών τους, όσο και στη δική τους διδακτική προσέγγιση στο κάθε μάθημα.

ΕΝΟΤΗΤΑ Γ1 Βασικές Έννοιες της Πληροφορικής και της Επιστήμης Ηλεκτρονικών Υπολογιστών

Για την προετοιμασία και συγγραφή του υλικού αυτής της ενότητας εργάστηκαν οι ακόλουθοι/θες Καθηγητές/τριες Πληροφορικής και Επιστήμης Ηλεκτρονικών Υπολογιστών που παρακολούθησαν το πρόγραμμα επιμόρφωσης μάχιμων εκπαιδευτικών για το Νέο Αναλυτικό Πρόγραμμα:

Αυξεντίου Σοφούλης
Ζάγγουλος Νικόλαος (Σύμβουλος)
Ιάσονος Έλενα
Μάκκουλα Πόλα
Μυλωνάς Σωκράτης (Σύμβουλος)
Ξενοφώντος Ξένιος
Παναγιώτου Χριστίνα

Γ1.1 Αναπαράσταση Δεδομένων και Χωρητικότητα Μονάδων Αποθήκευσης

Τι θα μάθουμε σήμερα:

- ◊ Να αναφέρουμε τον τρόπο αναπαράστασης των δεδομένων (δυναμικό σύστημα)
- ◊ Να αναγνωρίζουμε πώς γράμματα και σύμβολα από το πληκτρολόγιο αντιστοιχούν σε δυαδικούς αριθμούς (με τη βοήθεια πίνακα ASCII)
- ◊ Να αναφέρουμε τις μονάδες μέτρησης της χωρητικότητας των μονάδων αποθήκευσης (KB, MB, GB, TB, PB) και πώς αυτές σχετίζονται μεταξύ τους
- ◊ Να μετατρέπουμε από ένα πολλαπλάσιο σε άλλο
- ◊ Να συγκρίνουμε τη χωρητικότητα δυο ή περισσότερων μονάδων αποθήκευσης.

1. Αναπαράσταση Δεδομένων – Δυαδικό Σύστημα

Ο Ηλεκτρονικός Υπολογιστής αποτελείται από ένα σύνολο συσκευών και κυκλωμάτων, η επικοινωνία των οποίων επιτυγχάνεται με την ανταλλαγή ηλεκτρονικών σημάτων. Κατά συνέπεια, όλα τα είδη πληροφοριών (κείμενο, εικόνα, ήχος, κ.ά.) για να μπορούν να γίνουν αντιληπτά από τον ηλεκτρονικό υπολογιστή, θα πρέπει να μετατραπούν σε ηλεκτρικά σήματα, τα οποία αποτελούν και τη γλώσσα του ηλεκτρονικού υπολογιστή. Η γλώσσα του ηλεκτρονικού υπολογιστή αποτελείται από μόνο δύο σύμβολα, το 0 και το 1, τα οποία ονομάζονται δυαδικά ψηφία - **bits** (binary digits). Το δυαδικό ψηφίο 1 αντιπροσωπεύει την ύπαρξη μιας ψηλότερης τάσης ρεύματος, ενώ το δυαδικό ψηφίο 0 αντιπροσωπεύει μια χαμηλότερη τάση ρεύματος.

Το bit είναι η μικρότερη ποσότητα πληροφορίας που μπορεί να διαχειριστεί ο ηλεκτρονικός υπολογιστής. Για την παράσταση ενός χαρακτήρα στον υπολογιστή απαιτούνται, συνήθως, 8 bits τα οποία αποτελούν και τη στοιχειώδη μονάδα αποθήκευσης, τη ψηφιολέξη (**byte**).

2. Αναπαράσταση Χαρακτήρων με Δυαδικά Ψηφία

Οι περισσότερες από τις πληροφορίες που χρειάζεται να αποθηκεύσουμε είναι, κυρίως, **χαρακτήρες** δηλαδή, γράμματα και σύμβολα από το πληκτρολόγιο. Αυτοί, όπως μάθαμε προηγουμένως, αναπαριστώνται στο εσωτερικό του υπολογιστή με δυαδικά ψηφία, 0 και 1, ως εξής:

- Κάθε χαρακτήρας αποτελείται από έναν μοναδικό συνδυασμό 8 δυαδικών ψηφίων, για να ξεχωρίζει από τους υπόλοιπους χαρακτήρες.
- Υπάρχουν διαφορετικοί συνδυασμοί (κωδικοί) για καθένα από τα γράμματα του αγγλικού αλφαβήτου (κεφαλαία και μικρά), τα 10 αριθμητικά ψηφία (0 - 9) και για κάποιους άλλους ειδικούς χαρακτήρες (+, -, ? κ.α.).
- Οι πληροφορίες αναπαριστώνται στον υπολογιστή σαν μια σειρά από τέτοιους οκταψήφιους δυαδικούς κωδικούς.
- Οι περισσότερες σύγχρονες κωδικοποιήσεις χαρακτήρων βασίζονται στον Πίνακα Κωδικοποίησης ASCII (American Standard Code for Information Interchange).

3. Μονάδα Μέτρησης της Χωρητικότητας των Μονάδων Αποθήκευσης

Το **byte** (ψηφιολέξη) είναι η βασική μονάδα μέτρησης της ποσότητας της πληροφορίας στα υπολογιστικά συστήματα. Το κάθε byte αποτελείται από 8 bits. Λόγω του αυξανόμενου όγκου των ψηφιακών δεδομένων (κειμένων, εικόνων, βίντεο, ήχων) που αποθηκεύονται στη μνήμη του υπολογιστή, δημιουργήθηκαν οι μονάδες πολλαπλάσιων του Byte. Κατά συνέπεια,

όταν αναφερόμαστε στη χωρητικότητα μιας συγκεκριμένης μονάδας αποθήκευσης, μπορούμε να χρησιμοποιούμε ένα ή και περισσότερα από τα εξής πολλαπλάσια του byte:

- (α) KB 1 **KiloByte** = $2^{10} = 1024$ Byte ≈ 1.000 Byte
- (β) MB 1 **MegaByte** = $2^{20} = 1024$ KB ≈ 1000 KB = 1.000.000 Byte
- (γ) GB 1 **GigaByte** = $2^{30} = 1024$ MB ≈ 1000 MB = 1.000.000.000 Byte
- (δ) TB 1 **TeraByte** = $2^{40} = 1024$ GB ≈ 1000 GB = 1.000.000.000.000 Byte
- (ε) PB 1 **PetaByte** = $2^{50} = 1024$ TB ≈ 1000 TB = 1.000.000.000.000.000 Byte

Το καθένα από τα πολλαπλάσια του Byte είναι περίπου χίλιες φορές μεγαλύτερο από το προηγούμενο πολλαπλάσιο. Για παράδειγμα, στο σκληρό δίσκο της *Εικόνας 1*, ο οποίος έχει χωρητικότητα 2 TB, μπορούμε να αποθηκεύσουμε περίπου 2.000.000.000.000 (2X1000X1000X1000X1000) bytes ή χαρακτήρες. Επίσης, στη μνήμη φλας (flash memory) της *Εικόνας 2*, η οποία έχει χωρητικότητα 16 GB, μπορούμε να αποθηκεύσουμε περίπου 16.000.000.000 (16X1000X1000X1000) bytes ή χαρακτήρες.



Εικόνα 1 Σκληρός Δίσκος



Εικόνα 2 Μνήμη Φλας

4. Μετατροπή Από Ένα Πολλαπλάσιο του Byte σε Άλλο

Στην πραγματικότητα, το κάθε πολλαπλάσιο του byte είναι 1024 φορές μεγαλύτερο από το προηγούμενο πολλαπλάσιο. Επειδή όμως η μετατροπή από ένα πολλαπλάσιο σε άλλο, με βάση το 1024, πιθανόν να περιπλέκει του αναγκαίους υπολογισμούς, συνηθίζεται να χρησιμοποιείται ως βάση ο στρογγυλοποιημένος αριθμός 1000. Κατά συνέπεια, όταν μετατρέπουμε από ένα πολλαπλάσιο σε άλλο μεγαλύτερό του διαιρούμε με το 1000, ενώ, όταν μετατρέπουμε από ένα πολλαπλάσιο σε άλλο μικρότερό του πολλαπλασιάζουμε με το 1000. Οι αριθμοί που συμπεριλαμβάνονται στον παρακάτω πίνακα είναι κατά προσέγγιση, αφού χρησιμοποιήθηκε ως βάση για τους υπολογισμούς το 1000 και όχι το 1024. Για παράδειγμα, μπορούμε να πούμε ότι 1 GB ισούται με περίπου 1.000.000 KBs ή 0,001 TBs.

| | 1 Byte | 1 KB | 1 MB | 1 GB | 1 TB |
|---------|----------------|-------------|-----------|---------------|-------------------|
| Byte(s) | 1 | 1.000 | 1.000.000 | 1.000.000.000 | 1.000.000.000.000 |
| KB(s) | 0,001 | 1 | 1.000 | 1.000.000 | 1.000.000.000 |
| MB(s) | 0,000001 | 0,001 | 1 | 1.000 | 1.000.000 |
| GB(s) | 0,000000001 | 0,000001 | 0,001 | 1 | 1.000 |
| TB(s) | 0,000000000001 | 0,000000001 | 0,000001 | 0,001 | 1 |

Πίνακας Κωδικοποίησης ASCII

| | |
|------------|----|
| 00000000 | 0 |
| 00000001 | 1 |
| 00000010 | 2 |
| 00000011 | 3 |
| 00000100 | 4 |
| 00000101 | 5 |
| 00000110 | 6 |
| 00000111 | 7 |
| 00001000 | 8 |
| 00001001 | 9 |
| 00001010 | 10 |
| 00001011 | 11 |
| 00001100 | 12 |
| 00001101 | 13 |
| 00001110 | 14 |
| 00001111 | 15 |
| 00010000 | 16 |
| 00010001 | 17 |
| 00010010 | 18 |
| 00010011 | 19 |
| 00010100 | 20 |
| 00010101 | 21 |
| 00010110 | 22 |
| 00010111 | 23 |
| 00011000 | 24 |
| 00011001 | 25 |
| 00011010 | 26 |
| 00011011 | 27 |
| 00011100 | 28 |
| 00011101 | 29 |
| - 00011110 | 30 |
| 00011111 | 31 |

| | |
|-------------|----|
| 00100000 | 32 |
| ! 00100001 | 33 |
| " 00100010 | 34 |
| # 00100011 | 35 |
| \$ 00100100 | 36 |
| % 00100101 | 37 |
| & 00100110 | 38 |
| ' 00100111 | 39 |
| (00101000 | 40 |
|) 00101001 | 41 |
| * 00101010 | 42 |
| + 00101011 | 43 |
| , 00101100 | 44 |
| - 00101101 | 45 |
| . 00101110 | 46 |
| / 00101111 | 47 |
| 0 00110000 | 48 |
| 1 00110001 | 49 |
| 2 00110010 | 50 |
| 3 00110011 | 51 |
| 4 00110100 | 52 |
| 5 00110101 | 53 |
| 6 00110110 | 54 |
| 7 00110111 | 55 |
| 8 00111000 | 56 |
| 9 00111001 | 57 |
| : 00111010 | 58 |
| ; 00111011 | 59 |
| < 00111100 | 60 |
| = 00111101 | 61 |
| > 00111110 | 62 |
| ? 00111111 | 63 |

| | |
|------------|----|
| @ 01000000 | 64 |
| A 01000001 | 65 |
| B 01000010 | 66 |
| C 01000011 | 67 |
| D 01000100 | 68 |
| E 01000101 | 69 |
| F 01000110 | 70 |
| G 01000111 | 71 |
| H 01001000 | 72 |
| I 01001001 | 73 |
| J 01001010 | 74 |
| K 01001011 | 75 |
| L 01001100 | 76 |
| M 01001101 | 77 |
| N 01001110 | 78 |
| O 01001111 | 79 |
| P 01010000 | 80 |
| Q 01010001 | 81 |
| R 01010010 | 82 |
| S 01010011 | 83 |
| T 01010100 | 84 |
| U 01010101 | 85 |
| V 01010110 | 86 |
| W 01010111 | 87 |
| X 01011000 | 88 |
| Y 01011001 | 89 |
| Z 01011010 | 90 |
| [01011011 | 91 |
| \ 01011100 | 92 |
|] 01011101 | 93 |
| ^ 01011110 | 94 |
| _ 01011111 | 95 |

| | |
|------------|-----|
| ` 01100000 | 96 |
| a 01100001 | 97 |
| b 01100010 | 98 |
| c 01100011 | 99 |
| d 01100100 | 100 |
| e 01100101 | 101 |
| f 01100110 | 102 |
| g 01100111 | 103 |
| h 01101000 | 104 |
| i 01101001 | 105 |
| j 01101010 | 106 |
| k 01101011 | 107 |
| l 01101100 | 108 |
| m 01101101 | 109 |
| n 01101110 | 110 |
| o 01101111 | 111 |
| p 01110000 | 112 |
| q 01110001 | 113 |
| r 01110010 | 114 |
| s 01110011 | 115 |
| t 01110100 | 116 |
| u 01110101 | 117 |
| v 01110110 | 118 |
| w 01110111 | 119 |
| x 01111000 | 120 |
| y 01111001 | 121 |
| z 01111010 | 122 |
| { 01111011 | 123 |
| 01111100 | 124 |
| } 01111101 | 125 |
| ~ 01111110 | 126 |
| □ 01111111 | 127 |

Βασικές Έννοιες

| | |
|--------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Δυαδικό Σύστημα: | Ο τρόπος αναπαράστασης των δεδομένων στους ηλεκτρονικούς υπολογιστές. |
| Bit (Binary Digit/Δυαδικό Ψηφίο): | Η μικρότερη ποσότητα πληροφορίας που μπορεί να διαχειριστεί ο ηλεκτρονικός υπολογιστής. Το διάδικο ψηφίο 1 αντιπροσωπεύει την ύπαρξη μιας ψηλότερης τάσης ρεύματος ενώ το δυαδικό ψηφίο 0 αντιπροσωπεύει μια χαμηλότερη τάση ρεύματος. |
| Byte (Ψηφιολέξη): | Σύνολο από 8 bits τα οποία αντιπροσωπεύουν κάποιο χαρακτήρα και αποτελούν και τη στοιχειώδη μονάδα αποθήκευσης στον ηλεκτρονικό υπολογιστή. |
| ASCII: | Πίνακας Κωδικοποίησης πάνω στον οποίο βασίζονται σχεδόν όλες οι σύγχρονες κωδικοποιήσεις χαρακτήρων. |
| KiloByte KB | = 1024 Byte ≈ 1.000 Byte |
| MegaByte MB | = 1024 KB ≈ 1000 KB = 1.000.000 Byte |
| GigaByte GB | = 1024 MB ≈ 1000 MB = 1.000.000.000 Byte |
| TeraByte TB | = 1024 GB ≈ 1000 GB = 1.000.000.000.000 Byte |
| PetaByte PB | = 1024 TB ≈ 1000 TB = 1.000.000.000.000.000 Byte |
| Χωρητικότητα Βασικών Μονάδων Αποθήκευσης: | <p>Floppy Disk ≈ 1,44 MB</p> <p>CD-Rom/CD-R/CD-RW ≈ 700 MB.</p> <p>DVD-ROM/DVD-R/DVD-RW ≈ 4,7 GB</p> <p>Flash Memory ≈ μεταξύ 512 MB και 64 GB</p> <p>BD-ROM/BD-R/BD-RW ≈ 25 GB</p> <p>Hard Disk ≈ μεταξύ 350 GB και 1 TB</p> |

Πηγές

1. Αράπογλου Α., Μαβόγλου Χ., Οικονομάκος Η., Φύτρος Κ., (2006) *Πληροφορική Α', Β', Γ' Γυμνασίου*, σελ. 106—107, ΟΕΔΒ.
2. www.csi.ucd.ie/staff/jcarthy/home/CourseNotes/Web%20Info%20Representation.htm
3. Wikipedia: en.wikipedia.org/wiki/Binary_code
4. Wikipedia: en.wikipedia.org/wiki/Binary_numeral_system
5. Wikipedia: <http://el.wikipedia.org/wiki/Byte>

ΕΝΟΤΗΤΑ Γ2 Το Υλικό/ Αρχιτεκτονική Ηλεκτρονικών Υπολογιστών

Για την προετοιμασία και συγγραφή του υλικού αυτής της ενότητας εργάστηκαν οι ακόλουθοι/θες Καθηγητές/τριες Πληροφορικής και Επιστήμης Ηλεκτρονικών Υπολογιστών που παρακολούθησαν το πρόγραμμα επιμόρφωσης μάχιμων εκπαιδευτικών για το Νέο Αναλυτικό Πρόγραμμα:

Αναστασίου Νίκος
Ανδρονίκου Καλομοίρα
Αυγουστή Αυγουστίνος
Αχιλλέως Αντώνης
Γιουτανή Μαρία
Ζάγγουλος Νικόλαος (Σύμβουλος)
Καζέλη Σοφία
Λιασή Τατιάνα
Μάκκουλα Πόλα
Μυλωνάς Σωκράτης (Σύμβουλος)
Νεοφύτου Σπύρος
Ξενοφώντος Ξένιος
Στυλιανού Σοφοκλής
Σωτηριάδης Γιάννης
Χριστοδούλου Δημήτρης

Γ2.1 Βασικές έννοιες της Αρχιτεκτονικής του Η/Υ

Τι θα μάθουμε σήμερα:

- ◊ Να αναφέρουμε και να εξηγήσουμε τι είναι Αρχιτεκτονική Η/Υ
- ◊ Να γνωρίσουμε τα βασικά εξαρτήματα που βρίσκονται στο εσωτερικό του υπολογιστή και να εξηγήσουμε τον ρόλο τους:
 - Τροφοδοτικό (Power Supply)
 - Μητρική πλακέτα ή κάρτα (motherboard)
 - Κύρια Μνήμη ή Μνήμη Τυχαίας Προσπέλασης (RAM)
 - Μνήμη Μόνο για Διάβασμα (ROM)
 - Κεντρική Μονάδα Επεξεργασίας (ΚΜΕ, CPU)
 - Κάρτες Επέκτασης και Υποδοχές Επέκτασης
 - Θύρες σύνδεσης(Ports)
- ◊ Να γνωρίσουμε βασικές κάρτες επέκτασης (κάρτα δικτύου, κάρτα ήχου, κάρτα γραφικών)
- ◊ Να γ
- ◊ ναρίσουμε τις κύριες θύρες σύνδεσης (ports) που πιθανόν να υπάρχουν σε έναν υπολογιστή και τι (συσσκευή) μπορεί να συνδέεται στην καθεμιά από αυτές.

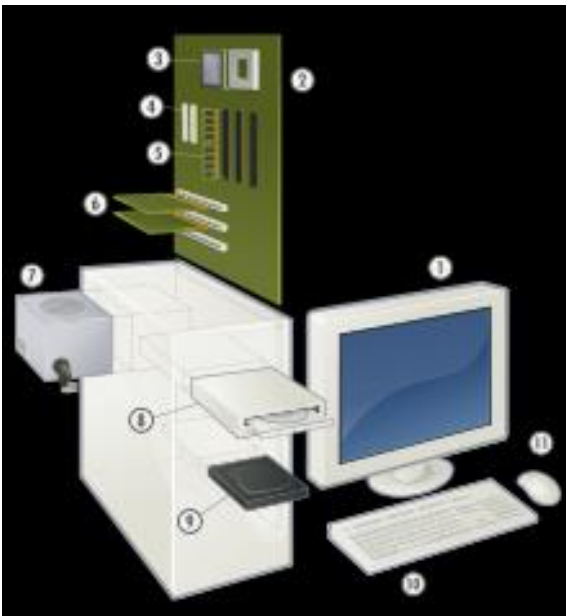
1. Το εσωτερικό του Η/Υ

Όπως μάθαμε σε προηγούμενα μαθήματα, ο ηλεκτρονικός υπολογιστής είναι μία ηλεκτρονική μηχανή με βασικές του λειτουργίες την είσοδο, την επεξεργασία δεδομένων, την αποθήκευση και την έξοδο πληροφοριών. Οι υπολογιστές κάθε είδους, είτε αυτοί είναι φορητοί είτε είναι επιτραπέζιοι, αποτελούνται από πολλά εξαρτήματα. Κάθε εξάρτημα έχει συγκεκριμένο ρόλο στη λειτουργία του υπολογιστή και όλα έχουν σχεδιαστεί για να συνεργάζονται, έτσι ώστε ο υπολογιστής να λειτουργεί ως ένα ενιαίο σύνολο και να επιτυγχάνει τις βασικές του λειτουργίες. Αυτό τον σχεδιασμό είναι που ονομάζουμε **αρχιτεκτονική** ενός Η/Υ, δηλαδή τα μέρη από τα οποία αποτελείται και πώς αυτά συνδέονται/επικοινωνούν.

Όπως όλα τα σπίτια αποτελούνται από βασικά δωμάτια (κουζίνα, υπνοδωμάτια, κ.λπ.), έτσι και οι υπολογιστές αποτελούνται από βασικά μέρη που θα δούμε πιο κάτω. Υπάρχουν όμως πολλά είδη σπιτιών που αν και αποτελούνται από τα ίδια βασικά δωμάτια, εντούτοις διαφέρουν. Για παράδειγμα, σε ένα παραδοσιακό σπίτι (π.χ. στο χωριό) τα δωμάτια συνδέονται με την αυλή, ενώ σε ένα μοντέρνο σπίτι αυτά συνδέονται με διάδρομο. Τα δωμάτια που αποτελούν ένα σπίτι και ο τρόπος που συνδέονται/επικοινωνούν μεταξύ τους καθορίζει την αρχιτεκτονική του (παραδοσιακή, μοντέρνα, κ.λπ.). Με τον ίδιο συλλογισμό, τα μέρη που αποτελούν έναν υπολογιστή και ο τρόπος που αυτά συνδέονται/επικοινωνούν μεταξύ τους καθορίζουν την αρχιτεκτονική του. Όπως υπάρχουν περιπτώσεις όπου παρά τις διαφορές τους, δύο σπίτια μπορεί να ακολουθούν την ίδια αρχιτεκτονική (π.χ. ένα σπίτι και ένα διαμέρισμα, όπου τα δωμάτια συνδέονται με διάδρομο), έτσι και στην περίπτωση των υπολογιστών, μπορεί ένας επιτραπέζιος και ένας φορητός υπολογιστής να ακολουθούν παρόμοια αρχιτεκτονική. Αντίστοιχα, δύο υπολογιστές με παρόμοιο μέγεθος μπορεί να ακολουθούν διαφορετική αρχιτεκτονική, όπως συμβαίνει με τα παραδοσιακά και τα μοντέρνα σπίτια. Επίσης, όπως κάποια σπίτια κτίζονται με δυνατότητα επέκτασης στο μέλλον, έτσι και ένας Η/Υ.

Η παρακάτω εικόνα παρουσιάζει τα πιο συνηθισμένα μέρη του υλικού ενός επιτραπέζιου ηλεκτρονικού υπολογιστή (ο υπολογιστής σας πιθανόν να διαφέρει λίγο στην εμφάνιση, αλλά περιλαμβάνει τα μέρη αυτά). Ένας φορητός υπολογιστής έχει αντίστοιχα μέρη, αλλά τα συνδυάζει σε ένα μόνο πακέτο μικρού μεγέθους. Αν ανοίξουμε το κουτί (πύργο) θα

παρατηρήσουμε τα εσωτερικά του μέρη. Ας δούμε μια σύντομη περιγραφή για τα πιο σημαντικά από αυτά.



Εικόνα 3 Τα μέρη ενός Ηλεκτρονικού Υπολογιστή

1. Οθόνη
2. Μητρική κάρτα
3. Κεντρική Μονάδα Επεξεργασίας
4. Υποδοχές επέκτασης
5. Μνήμη RAM
6. Κάρτες επέκτασης
7. Τροφοδοτικό
8. Οπτικά μέσα αποθήκευσης (CD-DVD)
9. Σκληρός δίσκος – Solid State Drive (SSD)
10. Πληκτρολόγιο

2. Το Τροφοδοτικό (Power Supply)

Όπως ήδη γνωρίζουμε, ο Η/Υ είναι μία μηχανή που χρειάζεται απαραίτητα ηλεκτρικό ρεύμα για να λειτουργήσει. Το *τροφοδοτικό* είναι η συσκευή που συνδέει τον υπολογιστή με την παροχή ηλεκτρικού ρεύματος και προσαρμόζει (μετατρέπει) την ηλεκτρική τάση και ρεύμα σε μορφή κατάλληλη για να τροφοδοτηθούν τα εξαρτήματα στο εσωτερικό του Η/Υ.



Εικόνα 4 Τροφοδοτικό

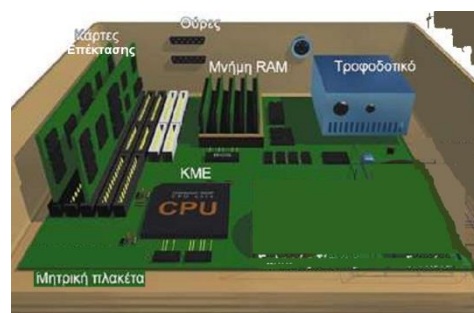
3. Η Μητρική Κάρτα ή Μητρική Πλακέτα (Motherboard)

Η *μητρική πλακέτα* είναι το βασικότερο τυπωμένο ηλεκτρονικό κύκλωμα που θα συναντήσουμε σε έναν Η/Υ. Πάνω σε αυτήν βρίσκονται εγκατεστημένα τα βασικά εξαρτήματα του Η/Υ. Κάποια από αυτά είναι μόνιμα τοποθετημένα, ενώ για άλλα υπάρχουν ειδικές υποδοχές (θέσεις) όπου εφαρμόζονται. Όλες οι συσκευές και τα εξαρτήματα που αποτελούν τον υπολογιστή συνδέονται με τη μητρική κάρτα είτε απευθείας, είτε με κατάλληλα καλώδια, όπως θα δούμε στη συνέχεια.



Εικόνα 5 Μητρική Πλακέτα

Τα κυριότερα εξαρτήματα που θα συναντήσουμε σε μια μητρική κάρτα είναι η Κεντρική Μονάδα Επεξεργασίας (ΚΜΕ, Central Processing Unit, CPU), η οποία εφαρμόζει πάνω σε ειδική βάση (socket), η μνήμη Τυχαίας Προσπέλασης (Random Access memory, RAM), η οποία τοποθετείται σε ειδικές υποδοχές, οι κάρτες επέκτασης που εφαρμόζουν σε ειδικές υποδοχές επέκτασης (expansion slots) και οι θύρες σύνδεσης (ports), με τις οποίες συνδέουμε περιφερειακές συσκευές στον Η/Υ. Θα συναντήσουμε, επίσης, ειδικές υποδοχές στις οποίες συνδέουμε τον



Εικόνα 6 Τα κυριότερα εξαρτήματα της Μητρικής Πλακέτας

σκληρό δίσκο (hard disk drive), τον SSD (Solid State Drive) και τους μηχανισμούς οπτικού δίσκου (CD/DVD).

Πάνω στη μητρική κάρτα υπάρχουν εξειδικευμένα ολοκληρωμένα κυκλώματα (integrated circuits), τα οποία είναι υπεύθυνα για τη συγκέντρωση και μεταφορά δεδομένων και πληροφοριών μεταξύ των διάφορων μερών και συσκευών που είναι συνδεδεμένα στις υποδοχές και τις θύρες της μητρικής κάρτας και άλλες απαραίτητες λειτουργίες. Σε αυτά συμπεριλαμβάνονται η Μνήμη Μόνο για Ανάγνωση (ROM) που περιέχει το Βασικό Σύστημα Εισόδου/Εξόδου (Basic Input/Output System, BIOS) και οι ελεγκτές (controllers), οι οποίοι συντονίζουν τη μεταφορά δεδομένων από τις μονάδες αποθήκευσης (π.χ. τον σκληρό δίσκο).

Συνοπτικά, η μητρική κάρτα είναι ένα μεγάλο τυπωμένο κύκλωμα στο οποίο συνδέονται και μέσω του οποίου επικοινωνούν όλες οι μονάδες του Η/Υ.

Ας δούμε παρακάτω σε συντομία τη λειτουργία των βασικότερων εξαρτημάτων της μητρικής κάρτας.

4. Η Κεντρική Μονάδα Επεξεργασίας (ΚΜΕ, Central Processing Unit, CPU)

Λέγεται διαφορετικά και μικροεπεξεργαστής (microprocessor).

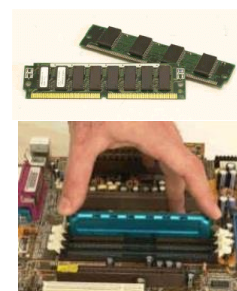
Η ΚΜΕ έχει ως βασική λειτουργία την επεξεργασία των δεδομένων. Εδώ εκτελούνται οι σειρές από εντολές που αποτελούν τα προγράμματα. Είναι αυτή που καθορίζει σε μεγάλο βαθμό την ταχύτητα του υπολογιστή. Εφαρμόζει πάνω στην ειδική βάση (socket) που βρίσκεται στη μητρική πλακέτα. Ο μικροεπεξεργαστής όταν λειτουργεί παράγει αρκετή θερμότητα και συνήθως χρειάζεται να ψύχεται με ειδικό ανεμιστήρα και ψήκτρα για να μην υπερθερμαίνεται.



Εικόνα 7
Μικροεπεξεργαστής

5. Η Μνήμη Τυχαίας Προσπέλασης –Random Access Memory (RAM)

Είναι ο αποθηκευτικός χώρος με τον οποίο η ΚΜΕ συνεργάζεται συνεχώς. Εδώ αποθηκεύονται τα δεδομένα και οι εντολές που θα χρησιμοποιήσει η ΚΜΕ για επεξεργασία, καθώς και τα αποτελέσματα (πληροφορίες) που προκύπτουν από την επεξεργασία. Η αποθήκευση είναι προσωρινή και η διακοπή παροχής ηλεκτρικού ρεύματος οδηγεί στη διαγραφή των δεδομένων που περιέχει. Αποτελείται από ολοκληρωμένα κυκλώματα σε μορφή μικρών καρτών που τοποθετούνται σε ειδικές υποδοχές στη μητρική πλακέτα.



Εικόνα 8 Μνήμη RAM

6. Η Μνήμη Μόνο για Ανάγνωση–Read Only Memory (ROM)

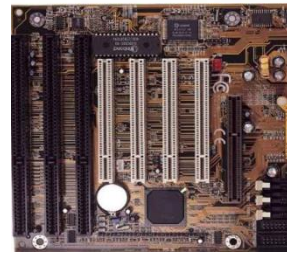
Είναι αποθηκευτικός χώρος με μορφή ολοκληρωμένου κυκλώματος, ενσωματωμένου στη μητρική κάρτα. Περιέχει το **Βασικό Σύστημα Εισόδου/Εξόδου (Basic Input/Output System, BIOS)**, που είναι πρόγραμμα (σειρά από εντολές) που εκτελείται κατά την εκκίνηση του υπολογιστή από την ΚΜΕ για να ελέγξει και να ρυθμίσει τις άλλες μονάδες, ώστε να είναι έτοιμες για λειτουργία.

Τα περιεχόμενά της είναι μόνιμα αποθηκευμένα από τον κατασκευαστή της και δεν χάνονται όταν διακοπεί το ηλεκτρικό ρεύμα.

Η Μνήμη Τυχαίας Προσπέλασης μαζί με τη Μνήμη Μόνο για Ανάγνωση αποτελούν την **Κύρια Μνήμη** του Η/Υ, διότι η ΚΜΕ μπορεί να επικοινωνεί άμεσα και γρήγορα με αυτές.

7. Οι Υποδοχές Επέκτασης (expansion slots) και οι Κάρτες Επέκτασης (expansion cards)

Οι Υποδοχές Επέκτασης είναι θέσεις στη μητρική κάρτα, όπου μπορούμε να τοποθετήσουμε ειδικά διαμορφωμένα τυπωμένα κυκλώματα, που ονομάζονται Κάρτες Επέκτασης (expansion cards), για να επεκτείνουμε τις δυνατότητες του υπολογιστή μας. Επειδή δεν είναι δυνατό ο κατασκευαστής της μητρικής κάρτας να προβλέψει και να παρέχει όλες τις λειτουργίες που επιθυμεί ο κάθε χρήστης, αλλά και επειδή εμφανίζονται νέες τεχνολογίες, παρέχονται οι υποδοχές επέκτασης, όπου ο κάθε χρήστης μπορεί να εφαρμόσει κατάλληλες κάρτες επέκτασης ανάλογα με τις ανάγκες του. Σε άλλες περιπτώσεις, η μητρική κάρτα παρέχει κάποιες λειτουργίες (π.χ. σύνδεση με οθόνη, εισαγωγή/εξαγωγή ήχων, σύνδεση με το δίκτυο), αλλά ο χρήστης αποφασίζει να τοποθετήσει μια αντίστοιχη κάρτα επέκτασης επειδή χρειάζεται κάποιες πιο εξειδικευμένες λειτουργίες.



Εικόνα 9 Υποδοχές Επέκτασης

Τα ακόλουθα είναι παραδείγματα καρτών επέκτασης:

Κάρτα γραφικών (graphics/video card): Επιτρέπει τη σύνδεση της οθόνης με τη μητρική κάρτα. Η απλή «κάρτα» γραφικών που υπάρχει συνήθως ενσωματωμένη στη μητρική κάρτα, δεν μπορεί πολλές φορές να αντεπεξέλθει σε απαιτητικές γραφικές εφαρμογές (π.χ. παιχνίδια στον Η/Υ) και συχνά τοποθετείται άλλη καλύτερη ως κάρτα επέκτασης.



Εικόνα 10 Κάρτα γραφικών

Κάρτα ήχου (sound card): Επιτρέπει στον Η/Υ να εισαγάγει, να επεξεργάζεται και να αναπαράγει ήχους. Η εισαγωγή του ήχου μπορεί να γίνει μέσω μικροφώνου που τοποθετούμε στην κατάλληλη υποδοχή της κάρτας. Η έξοδος του ήχου μπορεί να γίνει μέσω ηχείων ή ακουστικών.



Εικόνα 11 Κάρτα ήχου

Κάρτα δικτύου (network card): Επιτρέπει στον υπολογιστή να συνδεθεί με το τοπικό δίκτυο μέσω ειδικών καλωδίων ή και ασύρματα, ώστε ο υπολογιστής να μπορεί να ανταλλάξει πληροφορίες με άλλες συσκευές.



Εικόνα 12 Κάρτα δικτύου

Κάρτα τηλεόρασης/ραδιοφώνου (TV/Radio tuner card): Μέσω της κάρτας αυτής μετατρέπουμε τον Η/Υ σε τηλεόραση/ραδιόφωνο. Η κάρτα τηλεόρασης περιέχει ένα δέκτη τηλεόρασης και σε συνδυασμό με μια κεραία μπορεί να μετατρέψει την οθόνη του υπολογιστή σε τηλεόραση.



Εικόνα 13 Κάρτα τηλεόρασης

8. Θύρες σύνδεσης (Ports)

Εκτός από τις υποδοχές, που επιτρέπουν τη σύνδεση συσκευών και εξαρτημάτων στο εσωτερικό του υπολογιστή, η μητρική κάρτα και οι κάρτες επέκτασης επιτρέπουν τη σύνδεσή του με περιφερειακές συσκευές (π.χ. πληκτρολόγιο, ποντίκι, οθόνη, μνήμη φλας) και με το δίκτυο, που βρίσκονται στο εξωτερικό του υπολογιστή. Έτσι, τα αντίστοιχα κυκλώματα καταλήγουν στις **θύρες σύνδεσης (ports)**, δηλαδή σε υποδοχές στο κουτί του υπολογιστή για τη σύνδεσή τους, είτε απευθείας είτε με κατάλληλα καλώδια.

Ακολουθεί ένας επεξηγηματικός πίνακας:

| Όνομασία Θύρας | Περιγραφή/Λειτουργία |
|-------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Θύρα USB</p>  | <p>Η θύρα USB (Universal Serial Bus) υποστηρίζει τη μεταφορά δεδομένων σε σειρά με μεγάλη ταχύτητα. Πολλά είδη περιφερειακών συσκευών (εκτυπωτές, modems, σαρωτές, ποντίκια, πληκτρολόγια, ψηφιακές φωτογραφικές μηχανές, μνήμη φλας, βιντεοκάμερες, κ.ά.) είναι φτιαγμένα για να συνδέονται σε θύρα USB.</p> |
| <p>Θύρα VGA</p>  | <p>Σύνδεση οθόνης ή προβολέα</p> |
| <p>Θύρα LAN/Ethernet</p>  | <p>Σύνδεση υπολογιστή με το τοπικό δίκτυο.</p> |
| <p>Θύρα HDMI</p>  | <p>Η θύρα HDMI επιτρέπει τη μεταφορά ψηφιακού σήματος βίντεο Ψηλής Ευκρίνειας High Definition. Συνδέει τον υπολογιστή με συσκευές που εκπέμπουν High Definition video όπως με HDMI τηλεόραση, οθόνη ή προβολέα.</p> |
| <p>Συνδέσεις κάρτας ήχου</p>  | <ul style="list-style-type: none"> ○ Πράσινο – Σύνδεση ηχείων (Audio Out) ○ Μπλε – Σύνδεση με άλλες συσκευές ήχου (line in) ○ Ροζ – Σύνδεση μικροφώνου (Mic) |

Βασικές Έννοιες

| | |
|--------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Αρχιτεκτονική Η/Υ: | Ο τρόπος που σχεδιάστηκε ο Η/Υ, δηλαδή τα μέρη από τα οποία αποτελείται και πώς αυτά συνδέονται/επικοινωνούν. |
| Τροφοδοτικό (Power Supply): | Η συσκευή που συνδέει τον υπολογιστή με την παροχή ηλεκτρικού ρεύματος και προσαρμόζει (μετατρέπει) την ηλεκτρική τάση και το ρεύμα σε μορφή κατάλληλη για τα εξαρτήματα του Η/Υ. |
| Μητρική Πλακέτα (Motherboard) : | Συνοπτικά, η μητρική κάρτα είναι ένα μεγάλο τυπωμένο κύκλωμα, στο οποίο συνδέονται και μέσω του οποίου επικοινωνούν όλες οι μονάδες του Η/Υ. Περιλαμβάνει ειδικές υποδοχές/βάσεις για την Κεντρική Μονάδα Επεξεργασίας, τη Μνήμη RAM και τις μονάδες αποθήκευσης του Η/Υ. |
| Κεντρική Μονάδα Επεξεργασίας, ΚΜΕ (Central Processing Unit, CPU): | Είναι το εξάρτημα που έχει ως βασική λειτουργία την επεξεργασία των δεδομένων. Εδώ εκτελούνται οι σειρές από εντολές που αποτελούν τα προγράμματα. Εφαρμόζει πάνω σε ειδική βάση (socket) που βρίσκεται στη μητρική πλακέτα. Λέγεται διαφορετικά και μικροεπεξεργαστής (microprocessor). |
| Μνήμη Τυχαίας Προσπέλασης (Random Access Memory, RAM): | Αποθηκευτικός χώρος σε μορφή μικρών καρτών (από ολοκληρωμένα κυκλώματα) που τοποθετούνται σε ειδικές υποδοχές στη μητρική πλακέτα. Κρατεί τα δεδομένα και τις εντολές για επεξεργασία από την ΚΜΕ, καθώς και τα αποτελέσματα (πληροφορίες) που προκύπτουν από αυτήν. Η αποθήκευση είναι προσωρινή και τα δεδομένα διαγράφονται μετά από τη διακοπή παροχής ηλεκτρικού ρεύματος. |
| Μνήμη Μόνο για Ανάγνωση (Read Only Memory, ROM): | Αποθηκευτικός χώρος (ολοκληρωμένο κύκλωμα ενσωματωμένο στη μητρική κάρτα) με μόνιμα αποθηκευμένο περιεχόμενο που δεν χάνεται όταν διακοπεί το ηλεκτρικό ρεύμα. Περιέχει το Βασικό Σύστημα Εισόδου/Εξόδου (Basic Input/Output System, BIOS) , που είναι σειρά από εντολές που εκτελούνται κατά την εκκίνηση του υπολογιστή. |
| Κύρια Μνήμη: | Αποθηκευτικοί χώροι στους οποίους η ΚΜΕ έχει άμεσα πρόσβαση, δηλαδή η Μνήμη RAM και η Μνήμη ROM μαζί. |
| Κάρτες Επέκτασης (expansion cards): | Ειδικά διαμορφωμένα τυπωμένα κυκλώματα τα οποία τοποθετούμε για να επεκτείνουμε τις δυνατότητες του υπολογιστή παρέχοντας νέες εξειδικευμένες λειτουργίες που επιθυμεί ο κάθε χρήστης. Παραδείγματα είναι η κάρτα γραφικών (graphics/video card), η κάρτα ήχου (sound card) και η κάρτα δικτύου (Ethernet card). |

Υποδοχές Επέκτασης: Θέσεις στη μητρική κάρτα, όπου μπορούμε να τοποθετήσουμε τις Κάρτες Επέκτασης.

Θύρες σύνδεσης (Ports): Υποδοχές στο κουτί του υπολογιστή που συνδέονται με τη μητρική κάρτα και τις κάρτες επέκτασης και επιτρέπουν τη σύνδεση τους με περιφερειακές συσκευές και δίκτυο. Παραδείγματα:

- Θύρα USB (σύνδεση διάφορων συσκευών, αντικαθιστά σταδιακά τις προηγούμενες)
- Θύρα VGA (σύνδεση οθόνης)
- Θύρα LAN/Ethernet (σύνδεση με δίκτυο)
- Θύρα HDMI (σύνδεση οθόνης, προβολέα, τηλεόρασης)

Πηγές

1. Αράπογλου Α., Μαβόγλου Χ., Οικονομάκος Η., Φύτρος Κ., (2006) *Πληροφορική Α', Β', Γ' Γυμνασίου*, σελ. 109—112, ΟΕΔΒ.
2. *Τα μέρη ενός υπολογιστή*, www.windows.microsoft.com/el-GR/windows-vista/Parts-of-a-computer
3. *Τα βασικά μέρη του υπολογιστή*, www.musesnet.gr/ekp2000/VASIKA%20PC.htm

Γ2.2 Η Κεντρική Μονάδα Επεξεργασίας (ΚΜΕ)

Τι θα μάθουμε σήμερα:

- ◆ Να αναφέρουμε τα μέρη της Κεντρικής Μονάδας Επεξεργασίας (ΚΜΕ) και να εξηγήσουμε σε γενικές γραμμές τον ρόλο τους:
 - Αριθμητική και Λογική Μονάδα (Arithmetic and Logic Unit)
 - Καταχωρητές (Registers)
 - Μονάδα Ελέγχου (Control unit)
- ◆ Να αναφέρουμε τον ρόλο του Χρονιστή (Clock)
- ◆ Να αναγνωρίζουμε να και αναφέρουμε τα κύρια χαρακτηριστικά μιας ΚΜΕ (ταχύτητα, αρχιτεκτονική, αριθμός πυρήνων, κατανάλωση ρεύματος και τιμή)
- ◆ Να συγκρίνουμε δύο ΚΜΕ με βάση την ταχύτητα και την κατανάλωση ρεύματος
- ◆ Να επιλέγουμε σε παραδείγματα την καταλληλότερη από έναν κατάλογο ΚΜΕ με βάση τα χαρακτηριστικά της.

1. Η Κεντρική Μονάδα Επεξεργασίας, ΚΜΕ (Central Processing Unit, CPU)

Είναι το εξάρτημα που είναι υπεύθυνο για την επεξεργασία των δεδομένων, εκτελώντας τις εντολές από ένα πρόγραμμα. Η ΚΜΕ αποτελείται από

ηλεκτρονικό ολοκληρωμένο κύκλωμα και συχνά αναφέρεται και ως **μικροεπεξεργαστής (microprocessor)**.

Η ΚΜΕ βρίσκεται σε συνεχή επικοινωνία με την Κύρια Μνήμη (RAM και ROM) του υπολογιστή, από την οποία παίρνει τις εντολές και τα δεδομένα για επεξεργασία και στην οποία αποθηκεύει τις πληροφορίες που προκύπτουν από την επεξεργασία. Επικοινωνεί, επίσης, με τις άλλες μονάδες του υπολογιστή μέσα από τις διάφορες υποδοχές της Μητρικής Πλακέτας για είσοδο δεδομένων και έξοδο πληροφοριών.

Ο ρόλος της ΚΜΕ είναι τόσο σημαντικός, που καθορίζει πώς πρέπει να είναι οργανωμένο όλο το άλλο υλικό που συνδέεται μαζί της και επομένως καθορίζει την **αρχιτεκτονική** του υπολογιστή.

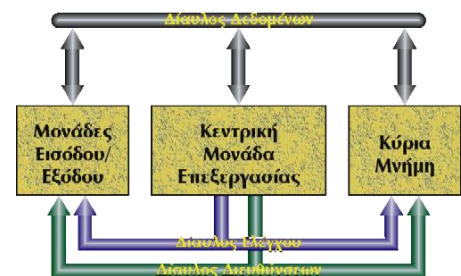
2. Τα Μέρη της ΚΜΕ

Παρόλο που τα ηλεκτρονικά κυκλώματα που αποτελούν την κάθε ΚΜΕ διαφέρουν μεταξύ τους, γενικά όλες οι ΚΜΕ περιέχουν τρία βασικά μέρη:

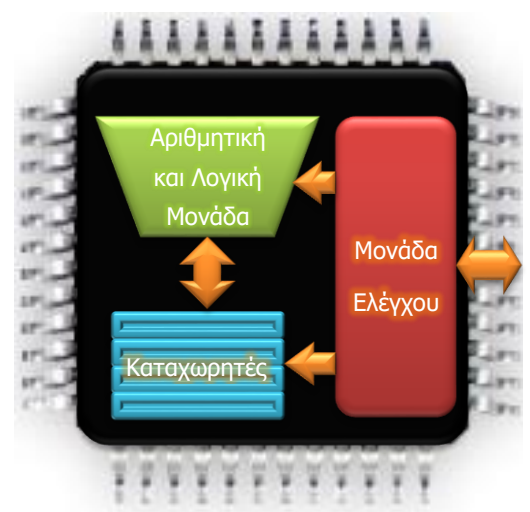
- Την Αριθμητική και Λογική Μονάδα (Arithmetic and Logic Unit, ALU)
- Τη Μονάδα Ελέγχου (Control Unit)
- Τους Καταχωρητές (Registers).



Εικόνα 14 Η Κεντρική Μονάδα Επεξεργασίας



Εικόνα 15 Η Κ.Μ.Ε. επικοινωνεί με την Κύρια Μνήμη και τις Μονάδες Εισόδου και Εξόδου



Εικόνα 16 Τα μέρη της Κεντρικής Μονάδας Επεξεργασίας

2.1 Η Αριθμητική και Λογική Μονάδα (Arithmetic and Logic Unit -ALU)

Είναι μια συλλογή από ηλεκτρονικά κυκλώματα φτιαγμένα για να εκτελούν αριθμητικές πράξεις (πρόσθεση, αφαίρεση, πολλαπλασιασμό, διαίρεση, κ.λπ.), συγκρίσεις (π.χ. να πάρει δύο δεδομένα και να αποφασίσει εάν το πρώτο είναι ίσο, μεγαλύτερο ή μικρότερο από το δεύτερο) και λογικούς συλλογισμούς, δηλαδή συλλογισμούς που απαντούνται με Ορθό ή Λάθος (π.χ. «Περιττός αριθμός **και** μικρότερος του 10», «πολλαπλάσιο του 5 **ή** μικρότερος του 10»).

Η Αριθμητική και Λογική Μονάδα από μόνη της δεν μπορεί να κάνει κάτι χρήσιμο, διότι χρειάζεται να της δοθούν τα δεδομένα που θα επεξεργαστεί και να της καθοριστεί ποια από τις αριθμητικές πράξεις ή λογικούς συλλογισμούς θα εκτελέσει. Από τη στιγμή που καθορίζεται ποια πράξη ή συλλογισμό θα εκτελέσει, η επεξεργασία των δεδομένων γίνεται με φοβερή μεγάλη ταχύτητα (σε ένα δευτερόλεπτο μπορούν να εκτελεστούν περίπου ένα δισεκατομμύριο πράξεις και συλλογισμοί).

2.2 Η Μονάδα Ελέγχου (Control Unit)

Είναι μια συλλογή από ηλεκτρονικά κυκλώματα που σκοπό έχουν να ελέγχουν και να συντονίζουν τη λειτουργία της ΚΜΕ και των άλλων μερών του υπολογιστή, ώστε να λειτουργούν αρμονικά και αξιόπιστα. Η μονάδα ελέγχου φροντίζει να μεταφέρονται από την κύρια μνήμη του υπολογιστή μία μία με τη σειρά οι εντολές του προγράμματος που εκτελείται. Μελετά την κάθε εντολή και ανάλογα με το τι χρειάζεται να γίνει συντονίζει τις μονάδες εισόδου και αποθήκευσης για να αποστείλουν τα δεδομένα που χρειάζονται, τα οποία διοχετεύει στην Αριθμητική και Λογική Μονάδα, στην οποία καθορίζει και το είδος της επεξεργασίας που χρειάζεται να γίνει. Όταν τελειώσει η επεξεργασία, συντονίζει τις απαραίτητες μονάδες εξόδου ή αποθήκευσης για να πάρουν τις πληροφορίες που παράγονται.

2.3 Οι Καταχωρητές (Registers)

Πρόκειται για ειδικούς αποθηκευτικούς χώρους που βρίσκονται μέσα στην ΚΜΕ και συνεργάζονται με την Αριθμητική και Λογική Μονάδα, η οποία μπορεί να παίρνει από αυτούς δεδομένα ή να αποθηκεύει σε αυτούς πληροφορίες πολύ πιο εύκολα και γρήγορα από οπουδήποτε αλλού. Το μέγεθός τους (δηλαδή από πόσα bit αποτελούνται) καθορίζεται από την αρχιτεκτονική της ΚΜΕ (π.χ. 16 bit, 32 bit, 64 bit, κ.λπ.).

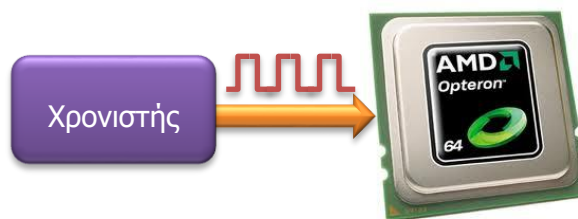
Δεν είναι εύκολο να περιγράψουμε ακριβώς πώς χρησιμοποιούνται οι καταχωρητές, διότι στους περισσότερους τύπους ΚΜΕ ο κάθε καταχωρητής έχει εξειδικευμένο ρόλο και χρήση. Ακολουθεί ένα παράδειγμα για να πάρουμε μια γενική ιδέα για τη χρήση τους.

Παράδειγμα:

Έστω ότι θέλουμε να υπολογίσουμε το αποτέλεσμα $(a+b) * (a-b)$. Το a και το b είναι δεδομένα που θα πάρουμε από το χρήστη (π.χ. $a=5$ και $b=2$). Για να γίνει αυτός ο υπολογισμός σε ένα πρόγραμμα, οι εντολές που θα πρέπει να εκτελεστούν είναι η πρόσθεση του a και του b (π.χ. $5+2=7$), η αφαίρεση του b από το a (π.χ. $5-2=3$) και μετά ο πολλαπλασιασμός των δύο ενδιάμεσων αποτελεσμάτων (π.χ. $7*3=21$). Επειδή η Αριθμητική και Λογική Μονάδα μπορεί να εκτελεί μία πράξη κάθε φορά, *αποθηκεύονται σε δύο καταχωρητές τα δύο ενδιάμεσα αποτελέσματα* (π.χ. το $a+b=7$ και το $a-b=3$), ώστε αμέσως μετά να προχωρήσει με τον πολλαπλασιασμό τους. Το τελικό αποτέλεσμα συνήθως αποθηκεύεται στη μνήμη RAM ή αποστέλλεται σε κάποια μονάδα εξόδου.

3. Ο Χρονοιστής (Clock)

Για να μπορούν τα μέρη της ΚΜΕ να συντονίζονται τη λειτουργία τους, χρειάζονται κάτι που να τους καθορίζει τον ρυθμό με τον οποίο η Μονάδα Ελέγχου μπορεί να στέλνει δεδομένα για επεξεργασία στην Αριθμητική και Λογική Μονάδα. Ο ρυθμός λειτουργίας της ΚΜΕ καθορίζεται από ένα εξωτερικό κύκλωμα που ονομάζεται **χρονοιστής** (ή ρολόι) και το οποίο στέλνει αδιάκοπα στα μέρη της μια σειρά από ηλεκτρικούς παλμούς για να συντονίζονται και καθορίζει την ταχύτητα με την οποία η ΚΜΕ εκτελεί εντολές από ένα πρόγραμμα. Όσο πιο συχνοί είναι οι παλμοί, τόσο πιο πολλές εντολές εκτελούνται ανά δευτερόλεπτο. Η συχνότητα των παλμών του χρονοιστή μετριέται σε Hz (Hertz). Οι σημερινές ΚΜΕ έχουν συχνότητα χρονοιστή αρκετών εκατοντάδων MHz (μεγαχέρτζ) ή μερικών GHz (γιγαχέρτζ). Ένα MHz αντιστοιχεί σε 1 εκατομμύριο παλμούς ανά δευτερόλεπτο ενώ 1 GHz αντιστοιχεί σε 1 δισεκατομμύριο παλμούς ανά δευτερόλεπτο.



Εικόνα 17 Ο χρονοιστής (Clock) παρέχει συνεχώς ηλεκτρικούς παλμούς με συγκεκριμένο ρυθμό για να συντονίζονται τα μέρη της Κ.Μ.Ε.

4. Κύρια χαρακτηριστικά μιας ΚΜΕ

Αδιαμφισβήτητα, το κυριότερο χαρακτηριστικό μιας ΚΜΕ είναι η **αρχιτεκτονική** της, δηλαδή πώς σχεδιάστηκε από τον κατασκευαστή της. Επειδή η αρχιτεκτονική καθορίζει τον τρόπο που λειτουργεί εσωτερικά μια ΚΜΕ, πόσους καταχωρητές έχει και πώς επικοινωνεί με τα υπόλοιπα εξαρτήματα του υπολογιστή, στην ουσία προσδιορίζει πώς η ΚΜΕ «αντιλαμβάνεται» και εκτελεί τις διάφορες εντολές. Συνήθως δημιουργούνται σειρές (οικογένειες) από ΚΜΕ με παρόμοια αρχιτεκτονική, που όλες «αντιλαμβάνονται» και εκτελούν εντολές με τον ίδιο τρόπο. Γενικά, δεν μπορούμε εύκολα να συγκρίνουμε δύο ΚΜΕ που έχουν διαφορετική αρχιτεκτονική, όπως δεν μπορούμε να συγκρίνουμε τη μηχανή ενός αυτοκινήτου και μιας μοτοσυκλέτας.

Η μέγιστη **ταχύτητα του χρονοιστή** που μπορεί να συνδυαστεί με κάποια ΚΜΕ αναγράφεται από τον κατασκευαστή της. Όσο πιο μεγάλη είναι, τόσο πιο πολλές εντολές μπορεί να εκτελεί η ΚΜΕ ανά δευτερόλεπτο. Επειδή όμως για την εκτέλεση μιας εντολής καταναλώνεται από τα κυκλώματά της συγκεκριμένη **ηλεκτρική ενέργεια**, αυτό σημαίνει ότι μια γρήγορη ΚΜΕ καταναλώνει περισσότερη ενέργεια για κάθε δευτερόλεπτο λειτουργίας της από μίαν αργή της ίδιας αρχιτεκτονικής. Η κατανάλωση ηλεκτρικής ενέργειας ανά δευτερόλεπτο ονομάζεται ισχύς και μετριέται σε **Watt (Watt, W)**. Η ΚΜΕ σε έναν σύγχρονο προσωπικό υπολογιστή καταναλώνει περίπου 25–150 W.

Τα τελευταία χρόνια έχουν χρησιμοποιηθεί από τους κατασκευαστές Κ.Μ.Ε διάφοροι τρόποι για εξοικονόμηση ενέργειας, που είναι ένας σημαντικός παράγοντας, ιδιαίτερα για φορητά συστήματα, όπου η ενέργεια παρέχεται από μπαταρία. Ένας τρόπος είναι να απενεργοποιούνται αυτόματα όσα κυκλώματα δεν χρησιμοποιούνται, ενώ άλλος τρόπος είναι να μειώνει εσωτερικά η ΚΜΕ τη συχνότητα του χρονοιστή όταν δεν χρειάζεται να επεξεργάζεται δεδομένα με τη μέγιστη δυνατή ταχύτητα.

Κάποιοι κατασκευαστές έχουν στραφεί προς μίαν εντελώς διαφορετική κατεύθυνση. Αντί να σχεδιάζουν ΚΜΕ που να περιέχουν ολοένα και πιο πολύπλοκα ηλεκτρονικά κυκλώματα, προχώρησαν στην υιοθέτηση αρχιτεκτονικής, που να βασίζεται σε απλούστερα κυκλώματα,

ώστε να καταναλώνουν λιγότερη ενέργεια, έστω κι εάν έχουν λιγότερες δυνατότητες από άλλες ΚΜΕ. Το αποτέλεσμα είναι να δημιουργηθεί μια μεγάλη ποικιλία από ΚΜΕ που καταναλώνουν ελάχιστη ενέργεια, αλλά είναι αρκετά αποδοτικές για να αποτελέσουν τη βάση διάφορων συσκευών όπως κινητών τηλεφώνων και υπολογιστών τύπου tablet.

Δύο παραδείγματα, ΚΜΕ για να αντιληφθούμε αυτές τις έννοιες δίνονται στον πιο κάτω πίνακα:

| Ονομασία Αρχιτεκτονικής | x86 | ARM |
|------------------------------|------------------------------------------------------|----------------------------------------------------------------------|
| Δημιουργός/ Κατασκευαστής | Εταιρεία Intel (και άλλες, π.χ. AMD) | Εταιρεία ARM Holdings (και άλλοι κατασκευαστές) |
| Παράδειγμα ΚΜΕ | Core i7-2600 | Cortex-A9 |
| Ταχύτητα χρονιστή | 3.4 GHz | 1 GHz |
| Κατανάλωση ενέργειας | 95 W | 1 W |
| Χαρακτηριστικά | ΚΜΕ με σύνθετη αρχιτεκτονική, γρήγορη και αποδοτική. | ΚΜΕ με απλουστευμένη αρχιτεκτονική, και χαμηλή κατανάλωση ενέργειας. |
| Πού χρησιμοποιείται συνήθως | Σε επιτραπέζιους προσωπικούς υπολογιστές | Σε κινητά τηλέφωνα και υπολογιστές τύπου tablet. |

Οι δύο αυτές ΚΜΕ δεν μπορούν εύκολα να συγκριθούν, διότι έχουν διαφορετική αρχιτεκτονική και εφαρμογές. Γενικά όμως μπορούμε να συμπεράνουμε ότι η Core i7-2600 εκτελεί περισσότερες εντολές ανά δευτερόλεπτο από την Cortex-A9 (λόγω της μεγαλύτερης ταχύτητας του χρονιστή), αλλά έχει πολύ μεγαλύτερη κατανάλωση ενέργειας.

Μπορούμε, όμως, να συγκρίνουμε την Core i7-2600 με μια άλλη ΚΜΕ της ίδιας αρχιτεκτονικής, π.χ. την ΚΜΕ Core i7-620LM (2.00 GHz, 25 W). Η i7-620LM έχει χρονιστή σε αρκετά χαμηλότερη συχνότητα (περίπου τη μισή), αλλά καταναλώνει περίπου το ¼ της ενέργειας σε σύγκριση με την i7-2600 και χρησιμοποιείται συνήθως σε φορητούς υπολογιστές (laptops, notebooks). Η σύγκριση ΚΜΕ έχει γίνει μια αρκετά δύσκολη διαδικασία, διότι ακόμη και ανάμεσα σε όσες ακολουθούν κάποια συγκεκριμένη αρχιτεκτονική, δεν είναι εύκολο να αποφασίσουμε με τη μελέτη λίγων στοιχείων (π.χ. ταχύτητα χρονιστή, κατανάλωση ενέργειας) πόσο καλύτερη είναι μια από κάποιαν άλλη. Υπάρχουν οργανισμοί που δοκιμάζουν πρακτικά διάφορες ΚΜΕ και υπολογίζουν και παρουσιάζουν στο Διαδίκτυο συγκριτικά στοιχεία (benchmarks), όπου μπορούμε να καταφύγουμε για πιο λεπτομερείς πληροφορίες.

Οι περισσότεροι σύγχρονοι υπολογιστές διαθέτουν συνήθως περισσότερες από μία ΚΜΕ, είτε ως ξεχωριστά ολοκληρωμένα κυκλώματα, είτε (πιο συχνά) ως πολλαπλούς **πυρήνες** στο ίδιο ολοκληρωμένο κύκλωμα. Για παράδειγμα η Core i7-2600 που αναφέρθηκε πιο πάνω περιέχει στην πραγματικότητα 4 ΚΜΕ (πυρήνες-cores). ΟΙ κατασκευαστές ΚΜΕ έχουν προχωρήσει στη δημιουργία **πολυπύρηνων επεξεργαστών** (με 2, 4, 8 ή και περισσότερους πυρήνες), κυρίως για τους πιο κάτω λόγους:

- Για πολλά χρόνια, η βασική εξέλιξη στην τεχνολογία των ηλεκτρονικών υπολογιστών βασιζόταν στην ικανότητα κατασκευής όλο και μικρότερων σε διαστάσεις τρανζίστορ (τα οποία είναι τα βασικά δομικά στοιχεία ενός ηλεκτρονικού υπολογιστή). Αυτό είχε ως αποτέλεσμα την αύξηση της ταχύτητας επεξεργασίας δεδομένων. Τα τελευταία,

όμως, χρόνια το μέγεθος τους έχει φτάσει να είναι τόσο μικρό, που αν τα φτιάξουν ακόμη μικρότερα, τότε θα αρχίσουν να επηρεάζονται από ορισμένα φυσικά φαινόμενα που παρατηρούνται σε πολύ μικρά μεγέθη επηρεάζουν την αξιοπιστία τους και έτσι η ΚΜΕ δεν θα παράγει τα σωστά αποτελέσματα. Έτσι, αντί να προσπαθούν να κατασκευάσουν μια γρηγορότερη ΚΜΕ, οι κατασκευαστές προχωρούν με την κατασκευή πολλών όχι και τόσο γρήγορων σε ένα ολοκληρωμένο κύκλωμα, ώστε να μπορούν να εκτελούνται ταυτόχρονα πολλές εντολές που είναι σχεδόν ισοδύναμο με την αύξηση της ταχύτητας.

- Πολλοί πυρήνες οδηγούν στη μείωση της κατανάλωσης ενέργειας. Αυτό ακούγεται κάπως παράδοξο, αλλά δεν είναι, αφού στην πράξη όταν οι ανάγκες για υπολογισμό είναι περιορισμένες, αυτόματα διακόπτεται η λειτουργία όσων πυρήνων δεν χρειάζονται και έτσι δεν καταναλώνουν ηλεκτρικό ρεύμα.
- Μπορούν να κατασκευαστούν εξειδικευμένοι πυρήνες που να ενεργοποιούνται ανάλογα με τις απαιτήσεις του χρήστη, π.χ. ένας απλός πυρήνας με χαμηλή κατανάλωση ενέργειας, ένας με εξειδικευμένες δυνατότητες για επεξεργασία γραφικών και παιχνίδια, ένας για πολύπλοκους υπολογισμούς με μεγάλη ακρίβεια κ.λπ.

Παράδειγμα:

Έχουμε τις ακόλουθες ΚΜΕ:

ΚΜΕ1: Ταχύτητα χρονιστή 2 GHz και κατανάλωση ενέργειας 40 W (ένας πυρήνας).

ΚΜΕ2: 2 πυρήνες, ο καθένας με ταχύτητα χρονιστή 1 GHz και κατανάλωση ενέργειας 10 W.

Η ΚΜΕ1 εκτελεί εντολές με διπλάσια ταχύτητα από ότι ο κάθε πυρήνας της ΚΜΕ2. Όταν όμως χρειάζεται, η ΚΜΕ2 ενεργοποιεί και τους 2 πυρήνες, εκτελεί ο καθένας 2 εντολές κάθε στιγμή και το σύνολο εντολών ανά δευτερόλεπτο είναι και στις δύο περιπτώσεις το ίδιο. Η κατανάλωση ενέργειας είναι όμως διπλάσια για την ΚΜΕ1 (40 W) από ότι συνολικά στους δύο πυρήνες της ΚΜΕ2 (20 W). Όταν όμως δεν χρειάζεται, η ΚΜΕ2 μπορεί να απενεργοποιεί τον έναν πυρήνα και έτσι να καταναλώνει μόνο 10 W. Έτσι η ΚΜΕ2 είναι πιο ευέλικτη και πιο οικονομική επιλογή από την ΚΜΕ1.

Βασικές Έννοιες

Κεντρική Μονάδα Επεξεργασίας, ΚΜΕ (Central Processing Unit, CPU):

Το εξάρτημα (ηλεκτρονικό ολοκληρωμένο κύκλωμα) που είναι υπεύθυνο για την επεξεργασία των δεδομένων, εκτελώντας τις εντολές από ένα πρόγραμμα. Συχνά αναφέρεται και ως μικροεπεξεργαστής (microprocessor). Αποτελείται από την *Αριθμητική και Λογική Μονάδα*, την *Μονάδα Ελέγχου* και τους *Καταχωρητές*.

Αριθμητική και Λογική Μονάδα (Arithmetic and Logic Unit -ALU):

Μια συλλογή από ηλεκτρονικά κυκλώματα φτιαγμένα για να εκτελούν αριθμητικές πράξεις, συγκρίσεις και λογικούς συλλογισμούς. Αποτελεί μέρος της ΚΜΕ.

Μονάδα Ελέγχου (Control Unit):

Μια συλλογή από ηλεκτρονικά κυκλώματα που ελέγχουν και συντονίζουν τη λειτουργία της ΚΜΕ και των άλλων μερών του υπολογιστή, ώστε να λειτουργούν αρμονικά και αξιόπιστα. Αποτελεί μέρος της ΚΜΕ.

**Καταχωρητές
(Registers):**

Ειδικοί αποθηκευτικοί χώροι που βρίσκονται μέσα στην ΚΜΕ και συνεργάζονται με την Αριθμητική και Λογική Μονάδα για εύκολη και γρήγορη πρόσβαση σε δεδομένα και πληροφορίες. Το μέγεθός τους (δηλαδή από πόσα bit αποτελούνται).

**Χρονοστής ή Ρολόι
(Clock):**

Ηλεκτρονικό εξάρτημα το οποίο στέλνει αδιάκοπα στα μέρη της ΚΜΕ μια σειρά από ηλεκτρικούς παλμούς για να συντονίζονται. Καθορίζει την ταχύτητα με την οποία η ΚΜΕ εκτελεί εντολές από ένα πρόγραμμα. Η συχνότητα των παλμών μετριέται σε Hz (Hertz), π.χ. 1 GHz.

**Κύρια χαρακτηριστικά
μιας ΚΜΕ:**

Το πιο κύριο χαρακτηριστικό μιας ΚΜΕ είναι η **αρχιτεκτονική** της, δηλαδή πώς σχεδιάστηκε από τον κατασκευαστή της. Το δεύτερο βασικό χαρακτηριστικό είναι η **συχνότητα του χρονοστή** της και το τρίτο είναι η **ενέργεια που καταναλώνει** ανά δευτερόλεπτο (ονομάζεται ισχύς) και μετριέται σε Wat (**Watt, W**).

Για ΚΜΕ κάποιας συγκεκριμένης αρχιτεκτονικής, όσο αυξάνεται η συχνότητα του χρονοστή, τόσο αυξάνεται και η κατανάλωση ενέργειας.

Πυρήνας:

Τα τελευταία χρόνια έχει επικρατήσει η πρακτική να τοποθετούν οι κατασκευαστές περισσότερες από μία ΚΜΕ στο ίδιο ολοκληρωμένο κύκλωμα. Αυτές ονομάζονται πυρήνες και προσφέρουν ταυτόχρονη εκτέλεση περισσότερων από μία εντολών.

Πηγές

1. Αράπογλου Α., Μαβόγλου Χ., Οικονομάκος Η., Φύτρος Κ., (2006) *Πληροφορική Α', Β', Γ' Γυμνασίου*, σελ. 109—112, ΟΕΔΒ.
2. Βουτηράς Γ., Κονιδάρη Ε., Κούτρας Μ., Σφώρος Ν., (2003) *Πληροφορική Γυμνασίου*, σελ. 38—39, ΟΕΔΒ
3. Γιαλλουρίδης Κ., Γκιπέρτης Ε., Κόμης Β., Σιδερίδης Α., Σταθόπουλος Κ. (1999) *Εφαρμογές Πληροφορικής Υπολογιστών Α', Β', Γ' Ενιαίου Λυκείου*, σελ. 39—42, ΟΕΔΒ
4. Μωϋσέως Ν., Διονυσίου Μ., (2006) *Βασικές Έννοιες Α' Λυκείου*, σελ. 37—42, ΥΑΠ.
5. Πανεπιστήμιο Μακεδονίας, Τμήμα Πληροφορικής, *Εισαγωγή στην ΚΜΕ*, http://www.it.uom.gr/project/mycomputer/cpu/es_mer11.html
6. Πληροφορίες για ΚΜΕ χαμηλής ισχύος αρχιτεκτονικής ARM, <http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.subset.cortexa.a8/index.html>
7. Σύγκριση ΚΜΕ, Tom's Hardware, <http://www.tomshardware.com/charts/processors,6.html>
8. Σύγκριση ΚΜΕ με βάση την κατανάλωση ενέργειάς τους, http://en.wikipedia.org/wiki/List_of_CPU_power_dissipation
9. Συγκριτικά αποτελέσματα (benchmarks) διάφορων ΚΜΕ, http://www.cpubenchmark.net/cpu_list.php
10. Δρ Παρασκευή Μεντζέλου (Computer Science), Παρουσίαση με θέμα: 'Υλικό Ηλεκτρονικού Υπολογιστή'.

Γ2.3 Κύρια και Βοηθητική Μνήμη

Τι θα μάθουμε σήμερα:

- ◆ Να αναφέρουμε τα είδη κύριας μνήμης και να τα συγκρίνουμε με βάση τα χαρακτηριστικά τους
- ◆ Να περιγράψουμε τον ρόλο του κάθε είδους της κύριας μνήμης
- ◆ Να εξηγούμε τι είναι η βοηθητική μνήμη
- ◆ Να εξηγούμε και να διακρίνουμε τη διαφορά ανάμεσα στην κύρια και τη βοηθητική μνήμη
- ◆ Να αναφέρουμε τον ρόλο και τα χαρακτηριστικά (χωρητικότητα, ταχύτητα ανάγνωσης/εγγραφής δεδομένων, φορητότητα, εγγραψιμότητα, κόστος, κ.λπ.) συνήθων μέσων βοηθητικής μνήμης (σκληροί δίσκοι, μνήμη φλας USB/SD/Memory Stick, CD, DVD, CD-R/RW, DVD-R/RW)
- ◆ Να επιλέγουμε σε παραδείγματα το καταλληλότερο μέσο για την αποθήκευση με βάση το μέγεθος των αρχείων και τα χαρακτηριστικά των διάφορων μέσων βοηθητικής μνήμης.

1. Μνήμη

Όπως έχουμε μάθει, σε καθεμία από τις βασικές λειτουργίες του υπολογιστή (είσοδος, επεξεργασία, έξοδος και αποθήκευση) αντιστοιχούν συγκεκριμένα ηλεκτρονικά κυκλώματα και συσκευές (έχουμε ήδη αναφερθεί στην Κεντρική Μονάδα Επεξεργασίας). Η λειτουργία της αποθήκευσης προσφέρεται από τις διάφορες μονάδες αποθήκευσης, που ονομάζονται **Μνήμη** του υπολογιστή. Η μνήμη είναι ένα από τα κυριότερα μέρη ενός υπολογιστή. Μαζί με την Κεντρική Μονάδα Επεξεργασίας (Central Processing Unit) είναι απαραίτητα για τη λειτουργία ενός υπολογιστικού συστήματος. Η μνήμη του υπολογιστή διακρίνεται σε δύο κατηγορίες: την **Κύρια Μνήμη** και τη **Βοηθητική Μνήμη**.

2. Η Κύρια Μνήμη του υπολογιστή

Έτσι ονομάζονται οι αποθηκευτικοί χώροι με τους οποίους η ΚΜΕ έχει **άμεση επικοινωνία** και από όπου μπορεί να παίρνει εντολές για εκτέλεση και δεδομένα για επεξεργασία, καθώς και να αποθηκεύει πληροφορίες που προκύπτουν από την επεξεργασία. Η κύρια μνήμη είναι οργανωμένη με απλό τρόπο, ως μια σειρά από αριθμημένα πανομοιότυπα «κουτιά» ή «θυρίδες» με συγκεκριμένη χωρητικότητα δεδομένων (π.χ. όλα τα «κουτιά» έχουν χωρητικότητα 1 byte ή 2 byte ή 4 byte ή 8 byte, κ.λπ.).

Για να διαβάσει δεδομένα, η ΚΜΕ χρειάζεται απλά να δώσει στην Κύρια Μνήμη τον αριθμό του «κουτιού» και αυτή θα εντοπίσει και αποστείλει τα δεδομένα που περιέχει. Για να αποθηκεύσει πληροφορίες, η ΚΜΕ χρειάζεται απλά να δώσει στην Κύρια Μνήμη τις πληροφορίες καθώς και τον αριθμό του «κουτιού» όπου πρέπει να τοποθετηθούν. Είναι σημαντικό να αναφέρουμε ότι τα bit από το κάθε «κουτί» αποστέλλονται ταυτόχρονα (παράλληλα), το καθένα μέσα από το δικό του κανάλι (καλώδιο), ώστε η μεταφορά να γίνεται όσο το δυνατό πιο γρήγορα.

Στην Κύρια Μνήμη ανήκουν δύο είδη ηλεκτρονικών κυκλωμάτων, η **Μνήμη Τυχαίας Προσπέλασης (Random Access Memory, RAM)** και η **Μνήμη Μόνο για Ανάγνωση (Read Only Memory, ROM)**.

2.1 Η Μνήμη Τυχαίας Προσπέλασης (Random Access Memory, RAM)

Σε αυτό το κομμάτι της κύριας μνήμης αποθηκεύονται προσωρινά οι εντολές από τα προγράμματα που έχουν ενεργοποιηθεί από τον χρήστη και εκτελούνται από την ΚΜΕ, τα δεδομένα του κάθε προγράμματος και οι πληροφορίες από την επεξεργασία, μέχρι να τελειώσει η εκτέλεση του κάθε προγράμματος. Η αποθήκευση είναι



Εικόνα 18 Μνήμη RAM

προσωρινή και η διακοπή παροχής ηλεκτρικού ρεύματος οδηγεί στη διαγραφή των δεδομένων που περιέχει.

Αποτελείται από ολοκληρωμένα κυκλώματα σε μορφή μικρών καρτών που τοποθετούνται σε ειδικές υποδοχές στη μητρική πλακέτα. Κάθε κάρτα μνήμης RAM χαρακτηρίζεται από τη **χωρητικότητα** της, , το **χρόνο προσπέλασης**, τον **τύπο** της (δηλαδή την αρχιτεκτονική της) και το **κόστος αγοράς** της.

Η **χωρητικότητα** μιας κάρτας μνήμης RAM είναι ο συνολικός αριθμός «κουτιών» για αποθήκευση πληροφοριών που περιέχει. Μετριέται σε MB ή GB (π.χ. 512MB, 2GB, κ.λπ.). Μπορούμε να αυξήσουμε τη συνολική μνήμη RAM τοποθετώντας επιπρόσθετες κάρτες μνήμης.

Παράδειγμα:

Στη μητρική κάρτα ενός συγκεκριμένου υπολογιστή υπάρχουν 4 υποδοχές μνήμης και σε μόνο μία από αυτές έχει τοποθετηθεί μια κάρτα μνήμης με χωρητικότητα 512MB. Μπορούμε να τοποθετήσουμε σε μian άλλη υποδοχή μια δεύτερη κάρτα μνήμης με χωρητικότητα 512MB και έτσι να έχουμε συνολικά 1024MB (=1GB). Θα μπορούσαμε, επίσης, να αφαιρέσουμε την κάρτα μνήμης των 512MB και να συμπληρώσουμε και τις 4 υποδοχές με κάρτες μνήμης των 2GB, ώστε να έχουμε συνολικά 8GB μνήμης RAM.

Ο **χρόνος προσπέλασης** είναι ο χρόνος που απαιτείται από τη στιγμή που η ΚΜΕ ζητά να διαβάσει ή να γράψει κάποιο δεδομένο στη μνήμη RAM, μέχρι να ολοκληρωθεί αυτή η διαδικασία (διάβασμα ή γράψιμο). Συνήθως ο χρόνος αυτός είναι μερικά τρισεκατομμυριοστά του δευτερολέπτου (ns). Αντί για το χρόνο προσπέλασης, συνήθως δίνεται η **ταχύτητα μεταφοράς δεδομένων** σε MB/s (Megabyte ανά δευτερόλεπτο), π.χ. 6400 MB/s.

Ο **τύπος** μιας κάρτας μνήμης RAM, εκτός από τις φυσικές διαστάσεις της κάρτας και τον τύπο υποδοχής για σύνδεσή της, καθορίζει επίσης το μέγεθος των «κουτιών» που την αποτελούν, αλλά και την ταχύτητα με την οποία μπορεί να ανταλλάζει δεδομένα με την ΚΜΕ. Κατά καιρούς έχουν εμφανιστεί διάφοροι τύποι. Ο τύπος μνήμης που έχει επικρατήσει τα τελευταία χρόνια για προσωπικούς υπολογιστές ονομάζεται Double Data Rate Synchronous Dynamic Random Access Memory (DDR SDRAM ή απλά DDR) και οι νεότερες παραλλαγές του, DDR2 και DDR3 που επιτρέπουν ακόμη μεγαλύτερες ταχύτητες μεταφοράς δεδομένων. Στις μνήμες τύπου DDR, το κάθε «κουτί» αποτελείται από 64 bit (8 byte), που ταιριάζει με πολλές σύγχρονες ΚΜΕ, οι οποίες έχουν καταχωρητές με μέγεθος επίσης 64 bit. Η ταχύτητα μεταφοράς δεδομένων αναγράφεται στην κάρτα μνήμης και εκφράζεται σε εκατομμύρια «κουτιά» ανά δευτερόλεπτο, π.χ. DDR-200, DDR2-400, DDR2-667, DDR3-800, κ.λπ. Μπορούμε εύκολα να υπολογίσουμε την ταχύτητα μεταφοράς δεδομένων, π.χ. για την DDR-200 έχουμε $200 \times 8 = 1600$ MB/s, ενώ για τη DDR3-800 έχουμε $800 \times 8 = 6400$ MB/s.

Το κόστος αγοράς για μια κάρτα μνήμης εξαρτάται από τον τύπο, τον χρόνο προσπέλασης και τη χωρητικότητά της. Συνήθως είναι από € 10 μέχρι € 50.



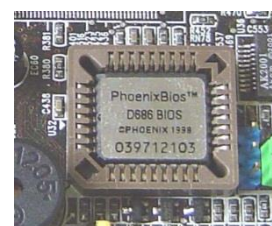
Εικόνα 19 Διάφοροι τύποι μνήμης RAM (σε ανατύπωση από wikipedia.org). Από πάνω προς τα κάτω, DIP, SIPP, SIMM (30-pin), SIMM (72-pin), DIMM (168-pin), DDR DIMM (184-pin).

2.2 Η Μνήμη Μόνο για Ανάγνωση (Read Only Memory , ROM)

Όταν εκκινούμε έναν υπολογιστή, η ΚΜΕ πρέπει να εκτελέσει μια *σειρά από εντολές* (δηλαδή πρόγραμμα), ώστε να ελέγξει ποιες συσκευές είναι συνδεδεμένες, να τις ρυθμίσει, ώστε να είναι έτοιμες για χρήση και στη συνέχεια να ενεργοποιήσει το Λειτουργικό Σύστημα, ώστε ο υπολογιστής να είναι έτοιμος για χρήση. Η σειρά αυτών των εντολών αποτελεί το **Βασικό Σύστημα Εισόδου/Εξόδου (Basic Input/Output System, BIOS)** και δεν μπορεί να τοποθετηθεί στη μνήμη RAM, επειδή θα διαγραφόταν όταν διακόπτεται η παροχή ηλεκτρικού ρεύματος. Για τον σκοπό αυτό χρησιμοποιείται ένα άλλο είδος κύριας μνήμης, η **Μνήμη Μόνο για Ανάγνωση (Read Only Memory , ROM)**.

Η μνήμη ROM λειτουργεί με παρόμοιο τρόπο όπως η μνήμη RAM, με τη διαφορά ότι το περιεχόμενό της είναι μόνιμα αποθηκευμένο από τον κατασκευαστή της και δεν χάνεται όταν διακοπεί το ηλεκτρικό ρεύμα ή όταν κλείνουμε τον υπολογιστή. Ούτε και είναι δυνατή η αποθήκευση σε αυτήν οποιωνδήποτε πληροφοριών κατά την επεξεργασία. Η μνήμη ROM είναι εγκατεστημένη σε μια μητρική κάρτα (20), η χωρητικότητά της είναι προκαθορισμένη και συνήθως δεν ξεπερνά τα 4 MB (ενώ για τη μνήμη RAM είναι συνήθως 4 GB, περίπου 1000 φορές μεγαλύτερη).

Το βασικό χαρακτηριστικό της μνήμης ROM (ότι το περιεχόμενό της αποθηκεύεται όταν την κατασκευάζουν και μετά δεν χάνεται ούτε αλλάζει) περιέχει ταυτόχρονα και ένα ουσιαστικό πρόβλημα: Εάν έγινε κάποιο σφάλμα στο πρόγραμμα που τοποθετήθηκε σε αυτήν από τον κατασκευαστή, δεν υπάρχει τρόπος να διορθωθεί. Τα τελευταία χρόνια οι κατασκευαστές έχουν αντικαταστήσει την «παραδοσιακή» μνήμη ROM με μνήμη Flash. Η μνήμη Flash είναι μια παραλλαγή της ROM, που επιτρέπει με κατάλληλη διαδικασία τη διαγραφή και επανεγγραφή του περιεχομένου της. Με αυτό τον τρόπο, οι χρήστες μπορούν να αντικαταστήσουν το πρόγραμμα BIOS στη μνήμη αυτή με άλλο βελτιωμένο που δημιούργησε ο κατασκευαστής.



Εικόνα 20 Μνήμη ROM

3. Η Βοηθητική Μνήμη

Με την ονομασία βοηθητική μνήμη εννοούμε το σύνολο των περιφερειακών μονάδων αποθήκευσης, όπου οι πληροφορίες αποθηκεύονται οργανωμένα και μακροπρόθεσμα. Στη βοηθητική μνήμη ανήκουν οι σκληροί δίσκοι (εσωτερικοί ή εξωτερικοί), ο δίσκος SSD, η μνήμη φλας (USB, SD, κ.λπ.), οι οπτικοί δίσκοι (CD, CD-R/RW, DVD, DVD-R/RW) και όλες οι άλλες μονάδες μόνιμης αποθήκευσης πληροφοριών.

Αντίθετα με την Κύρια Μνήμη, η ΚΜΕ δεν έχει άμεση πρόσβαση στη Βοηθητική Μνήμη. Για να πάρει ή να αποθηκεύσει δεδομένα στη βοηθητική μνήμη, θα πρέπει να αποστείλει κατάλληλες εντολές στα πολύπλοκα ηλεκτρονικά κυκλώματα που ελέγχουν τη λειτουργία της κάθε περιφερειακής μονάδας αποθήκευσης, τα οποία αφού εντοπίσουν τα δεδομένα, τα αποστέλλουν στην ΚΜΕ μέσα από τα καλώδια που συνδέουν την περιφερειακή μονάδα αποθήκευσης με τη μητρική κάρτα.

Στη συνέχεια παρουσιάζονται τα κύρια είδη περιφερειακών μονάδων αποθήκευσης και τα βασικά χαρακτηριστικά τους, ώστε να μπορείτε να συγκρίνετε και να επιλέξετε το πιο κατάλληλο ανάλογα με τις ανάγκες σας.

4. Ο σκληρός δίσκος (Hard Disk Drive) και ο δίσκος SSD (Solid State Drive)

Πρόκειται για μονάδες αποθήκευσης που έχουν μεγάλη χωρητικότητα και ταχύτητα μεταφοράς δεδομένων. Συνήθως είναι μόνιμα εγκατεστημένοι μέσα στον υπολογιστή.



Εικόνα 21 Ένας σκληρός δίσκος

Ο **HDD (Hard Disk Drive)** είναι ένας μηχανικός σκληρός δίσκος που λειτουργεί με έναν περιστρεφόμενο δίσκο στον οποίο αποθηκεύονται τα δεδομένα μέσω μίας κινούμενης κεφαλής.

Ο **SSD (Solid State Drive)** είναι ένας δίσκος που λειτουργεί όπως ένα κοινό USB stick, αποθηκεύει δηλαδή τις πληροφορίες μέσα σε μικρά τσιπάκια.



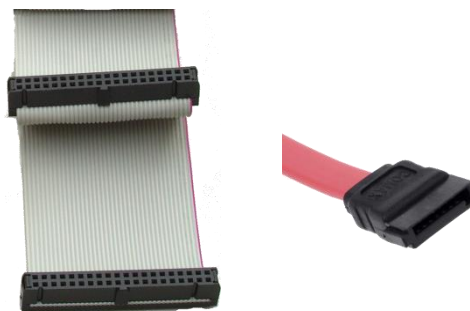
Εικόνα 22 Ένας δίσκος SSD

Σε αντίθεση με τους HDD δίσκους, οι SSD δεν έχουν μηχανικά μέρη με αποτέλεσμα να είναι αθόρυβοι, πιο γρήγοροι και πιο αξιόπιστοι, καταναλώνοντας παράλληλα λιγότερη ενέργεια. Από την άλλη πλευρά, όμως, είναι μικρότεροι σε χωρητικότητα, πιο ακριβοί και έχουν μικρότερη διάρκεια ζωής σε σχέση με τους HDD δίσκους.

4.1 Χαρακτηριστικά ενός δίσκου

Τα κύρια χαρακτηριστικά ενός δίσκου είναι τα ακόλουθα:

- **Χωρητικότητα:** Πόσα byte συνολικά μπορούμε να αποθηκεύσουμε σε αυτόν. Συνήθεις χωρητικότητες είναι 160 GB, 250 GB, 320GB, 500 GB, 1 TB, 2 TB. Υπάρχουν όμως και δίσκοι με μεγαλύτερη χωρητικότητα, π.χ. 8 TB.
- **Ταχύτητα μεταφοράς δεδομένων:** Με ποια ταχύτητα μεταφέρονται τα δεδομένα από και προς τον δίσκο. Μετριέται σε bit ανά δευτερόλεπτο (bits per second, b/s) ή σε byte ανά δευτερόλεπτο (bytes per second, B/s). Η ταχύτητα μεταφοράς δεδομένων για τον σκληρό δίσκο είναι μεταξύ 1 Gb/s και 1,5 Gb/s περίπου (δηλαδή περίπου 100 MB/s μέχρι 150 MB/s), ενώ για τον δίσκο SSD είναι μεταξύ 2 Gb/s και 5,5 Gb/s περίπου (δηλαδή περίπου 200 MB/s μέχρι 550 MB/s).
- **Τύπος σύνδεσης:** Πρόκειται για τον τρόπο σύνδεσης (το καλώδιο που μεταφέρει δεδομένα) του δίσκου με τη μητρική κάρτα. Οι παλαιότεροι σκληροί δίσκοι έχουν σύνδεση IDE (Integrated Drive Electronics) με καλώδιο σαν πλατεία κορδέλα και παράλληλη μεταφορά δεδομένων (πολλά bit ταυτόχρονα), με ταχύτητα μέχρι και 133 MB/s. Ο τύπος σύνδεσης σε πιο πρόσφατους υπολογιστές ακολουθεί το πρότυπο SATA (Serial Advanced Technology Attachment) που χρησιμοποιεί σειριακή μεταφορά δεδομένων (ένα bit κάθε φορά) με ταχύτητα 1,5 Gb/s (150 MB/s) και οι νεότερες εκδόσεις SATA 2.0 (ταχύτητα 3 Gb/s ή 300 MB/s) και SATA 3.0 (ταχύτητα 6 Gbps ή 600 MBps).



Εικόνα 23 Καλώδια σύνδεσης για τύπους IDE (αριστερά) και SATA (δεξιά)

- **Φυσικό μέγεθος:** Οι διαστάσεις του δίσκου. Ανάλογα με το πού θα τοποθετηθεί, θα πρέπει να έχει και το αντίστοιχο φυσικό μέγεθος. Έχουν επικρατήσει οι δίσκοι μεγέθους 3.5 ιντσών (3.5"), που ταιριάζουν σε επιτραπέζιους υπολογιστές και 2.5 ιντσών (2.5"), που είναι μικρότεροι και συνήθως χρησιμοποιούνται σε φορητούς υπολογιστές.

4.2 Εσωτερικοί και εξωτερικοί δίσκοι

Τα τελευταία χρόνια έχουν επικρατήσει δύο είδη δίσκων.

Εσωτερικός δίσκος (internal disk): Είναι ένας δίσκος προσαρμοσμένος μέσα στον υπολογιστή, ο οποίος δεν μπορεί να μετακινηθεί εύκολα από υπολογιστή σε υπολογιστή. Συνδέεται απευθείας με τη μητρική κάρτα, συνήθως με καλώδιο SATA και επιτρέπει τη γρήγορη ανάγνωση και αποθήκευση δεδομένων. Σε επιτραπέζιους υπολογιστές είναι σχετικά εύκολο να προσθέσουμε περισσότερους από έναν εσωτερικούς δίσκους για να αυξήσουμε τη συνολική χωρητικότητα. Σε φορητούς υπολογιστές, λόγω περιορισμένου χώρου και συνδέσεων, αυτό δεν είναι συνήθως δυνατό.

Εξωτερικός δίσκος (external disk): Είναι ένας δίσκος ο οποίος συνδέεται εξωτερικά με τον ηλεκτρονικό υπολογιστή και μπορεί εύκολα να μετακινηθεί από έναν υπολογιστή σε άλλο. Συνήθως συνδέεται μέσω θύρας USB. Η ταχύτητα μεταφοράς δεδομένων στους εξωτερικούς αυτούς δίσκους είναι χαμηλότερη από τους εσωτερικούς.



Εικόνα 24 Εξωτερικός σκληρός δίσκος

5. Οι Οπτικοί δίσκοι (CD, DVD, CD-R/RW, DVD-R/RW)

Σε αντίθεση με τους δίσκους που αναφέραμε πιο πάνω, οι οποίοι χρησιμοποιούν μαγνητισμό για αποθήκευση δεδομένων, οι οπτικοί δίσκοι χρησιμοποιούν ιδιότητες του φωτός. Τα διάφορα είδη οπτικών δίσκων (CD, CD-R, CD-RW, κ.λπ.) διαφέρουν κυρίως στο υλικό και τον τρόπο που είναι φτιαγμένη η επιφάνεια εγγραφής των δεδομένων. Η επιφάνεια αυτή αποτελείται από μια σειρά από μικρά στίγματα που είτε είναι διαφανή είτε αδιαφανή. Με αυτό τον τρόπο κωδικοποιούνται και εγγράφονται το 0 και το 1. Για να διαβάσει τα δεδομένα ο υπολογιστής, ο οπτικός δίσκος τοποθετείται σε ειδική συσκευή/οδηγό (Optical Disk Drive) και μέσω μιας φωτεινής ακτίνας (λέιζερ) γίνεται ανάγνωση των δεδομένων.

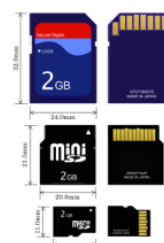


Εικόνα 25 Οπτικός δίσκος

Οι πιο γνωστοί τύποι οπτικών δίσκων είναι το CD-ROM και το DVD-ROM, καθώς και τα εγγράψιμα (CD-R, DVD-R) και επανεγγράψιμα (CD-RW, DVD-RW) αντίστοιχά τους.

6. Αποθηκευτικά μέσα τύπου Μνήμης Φλας (USB, SD)

Με την εμφάνιση των φορητών υπολογιστών, κινητών τηλεφώνων και άλλων φορητών συσκευών, αλλά και με την ανάγκη για ένα φορητό αποθηκευτικό μέσο μεγάλης χωρητικότητας, ευκολίας στην εγγραφή, και μικρότερο από τους επανεγγράψιμους οπτικούς δίσκους, η τεχνολογία ήλθε να προτείνει τη χρήση της μνήμης, φλας ως βάσης για μια σειρά από μικρές φορητές συσκευές.



Η μνήμη φλας αποτελεί μια τεχνολογία που δανείζεται στοιχεία τόσο από τη μνήμη RAM όσο και από τη ROM. Κάτω από κανονικές συνθήκες, τα

Εικόνα 26 Διάφορα είδη μνήμης φλας

δεδομένα που έχουν εγγραφεί διατηρούνται όταν διακοπεί η παροχή ηλεκτρικού ρεύματος, όπως και στη ROM. Εάν θέλουμε να εγγράψουμε δεδομένα, θα πρέπει να προηγηθεί η διαδικασία διαγραφής των δεδομένων από μια σειρά από byte, περίπου όπως γίνεται και με έναν τομέα ενός σκληρού δίσκου ή μιας δισκέτας. Κατόπιν, με την εφαρμογή ηλεκτρικού ρεύματος ψηλότερης τάσης από ότι για διάβασμα, γίνεται η εγγραφή των νέων δεδομένων.

Η μνήμη φλας χαρακτηρίζεται ως αποθηκευτικός χώρος «στερεάς κατάστασης» (solid state), διότι χρησιμοποιεί αποκλειστικά ηλεκτρονικά κυκλώματα (τρανζίστορ) και δεν περιέχει μηχανικά τμήματα, όπως μοτέρ, κεφαλές ανάγνωσης, κ.λπ. Αυτό προσφέρει πλεονεκτήματα σε σχέση με άλλες μονάδες αποθήκευσης, όπως:

- Χαμηλότερη κατανάλωση ενέργειας (τα μηχανικά μέρη καταναλώνουν ενέργεια)
- Λιγότερες πιθανότητες βλάβης από χτυπήματα και ταρακούνημα που είναι ένα γενικό πρόβλημα των άλλων μέσων αποθήκευσης
- Χαμηλός χρόνος αναζήτησης (κάτω από 0.1 ms)
- Υψηλή ταχύτητα μεταφοράς δεδομένων (100 MB/s – 500 MB/s).



Εικόνα 27 Μνήμη φλας τύπου USB flash drive

Τα κύρια μειονεκτήματά της είναι:

- Ακριβότερη ανά GB σε σύγκριση με τον σκληρό δίσκο
- Μικρότερη χωρητικότητα από τον σκληρό δίσκο, συνήθως από 2 GB μέχρι 128 GB
- Περιορισμένος αριθμός εγγραφών στο κάθε byte της (συνήθως 10000-100000 φορές)

Η μνήμη φλας χρησιμοποιείται κυρίως για τα ακόλουθα φορητά μέσα αποθήκευσης:

- **USB Flash Drive:** Το πιο κοινό μέσο για υπολογιστές, διότι έχει υποδοχή για σύνδεση σε θύρα USB, που υπάρχει σχεδόν σε όλους τους υπολογιστές
- **Memory Stick:** Είδος μνήμης για διάφορες φορητές συσκευές (φωτογραφικές συσκευές, κινητά τηλέφωνα), που εισήχθη από την εταιρεία Sony
- **Secure Digital card (SD Card)** και οι παραλλαγές της (π.χ. SDHC): Χρησιμοποιείται σε μεγάλη ποικιλία συσκευών αντί για σκληρό δίσκο, π.χ. υπερ-φορητοί υπολογιστές, και υπολογιστές τύπου «ταμπλέτας», κινητά τηλέφωνα, φωτογραφικές μηχανές, κ.λπ.
- **Compact Flash (CF):** Χρησιμοποιείται σε υπολογιστές και άλλες συσκευές αντί για σκληρό δίσκο, π.χ. υπολογιστές που λειτουργούν ως «τερματικά», συσκευές δικτύου, κ.ά.

7. Σύνοψη

Έχουμε μελετήσει τόσο τα είδη της κύριας όσο και της βοηθητικής μνήμης.

Τα κύρια χαρακτηριστικά τους είναι η χωρητικότητα, η ταχύτητα μεταφοράς δεδομένων, η δυνατότητα εγγραφής/επανεγγραφής, η φορητότητα και το κόστος. Στον ακόλουθο πίνακα παρουσιάζονται οι κύριες μονάδες αποθήκευσης και τα χαρακτηριστικά τους. Για τις μονάδες τύπου flash αναφέρεται μόνο το πιο κοινό είδος, αφού και τα άλλα είδη έχουν παρόμοια χαρακτηριστικά. Αντί για το κόστος, αναγράφεται το κόστος ανά GB για να είναι δυνατή η σύγκριση.

| Είδος | Χωρητικότητα | Ταχύτητα Μεταφ. δεδομένων | Εγγραφή | Επανεγγραφή | Φορητή | Κόστος ανά GB |
|-----------------------------|--------------|---------------------------|---------|-------------|--------|---------------|
| Μνήμη RAM | 1-8 GB | 6400 MB/s | ✓ | ✓ | - | € 15 |
| Σκληρός δίσκος (εσωτερικός) | 160GB-4TB | 150 MB/s | ✓ | ✓ | | € 0,08 |
| Σκληρός δίσκος (εξωτερικός) | 160GB-4TB | 60 MB/s | ✓ | ✓ | ✓ | € 0,10 |
| CD-ROM | 700 MB | 7,8 MB/s | | | ✓ | - |
| CD-R | 700 MB | 7,8 MB/s | ✓ | | ✓ | € 0,70 |
| CD-RW | 700 MB | 7,8 MB/s | ✓ | ✓ | ✓ | € 1,40 |
| DVD-ROM | 4,7 GB | 21 MB/s | | | ✓ | - |
| DVD-R | 4,7 GB | 21 MB/s | ✓ | | ✓ | € 0,14 |
| DVD-RW | 4,7 GB | 21 MB/s | ✓ | | ✓ | € 0,28 |
| USB Flash Drive | 2-128 GB | 100 MB/s-500 MB/s | ✓ | ✓ | ✓ | € 1,5 |

Όπως εύκολα διαπιστώνουμε, η μνήμη RAM είναι ακριβή, αλλά φοβερά πιο γρήγορη στη μεταφορά δεδομένων από ότι τα άλλα μέσα. Αυτός είναι εξάλλου και ο ρόλος της. Ο εσωτερικός σκληρός δίσκος αποτελεί την κύρια γρήγορη και φθηνή λύση για αποθήκευση, ενώ ο εξωτερικός είναι μια πολύ καλή λύση για να διατηρούμε αντίγραφο ασφαλείας σε περίπτωση που υπάρξει πρόβλημα με τον εσωτερικό μας σκληρό δίσκο.

Από τις άλλες μονάδες αποθήκευσης, το DVD-R αναδεικνύεται ως μια καλή λύση για αποθήκευση και αποστολή δεδομένων, λόγω του χαμηλού κόστους και φορητότητας, ενώ το USB Flash Drive κυριαρχεί ως ένας βολικός, αλλά κάπως ακριβός τρόπος για φορητή αποθήκευση δεδομένων.

Εδώ θα πρέπει να αναφερθεί ότι εκτός από τις πιο πάνω μονάδες αποθήκευσης, υπάρχουν και άλλες, με εξειδικευμένη χρήση. Για παράδειγμα, η μαγνητική ταινία (σε μορφή μικρής κασέτας) εξακολουθεί να χρησιμοποιείται ως μέσο για μακροχρόνια φύλαξη παλιών δεδομένων από πολλούς οργανισμούς, διότι έχει αποδειχτεί αξιόπιστη.

Επίσης, χρειάζεται μεγάλη προσοχή στην προστασία των δεδομένων, διότι σε κάποιες περιπτώσεις η απώλειά τους είναι δυσαναπλήρωτη. Οι μονάδες αποθήκευσης είναι συσκευές και κάποτε παθαίνουν βλάβη. Κοινά προβλήματα είναι η καταστροφή των δεδομένων μιας δισκέτας που βρέθηκε κοντά σε ένα ηλεκτρομαγνητικό πεδίο, ή ενός σκληρού δίσκου μετά από τράνταγμα ή πέσιμο (εάν ακουμπήσουν οι κεφαλές την επιφάνειά του, ο δίσκος καταστρέφεται), ή το γδάρισμα της επιφάνειας ενός DVD-R (και η έκθεση του στον ήλιο) ή ακόμη και η φθορά ενός USB Flash Drive, μετά από πολλές εγγραφές στο ίδιο αρχείο (ή και η απώλειά του).

Γενικά, συστήνεται να αποθηκεύουμε τα δεδομένα μας και σε μια δεύτερη μονάδα αποθήκευσης για ασφάλεια. Αυτό δεν είναι πάντοτε εύκολο, ιδιαίτερα σε κεντρικούς υπολογιστές. Εκεί υπάρχουν έξυπνοι μηχανισμοί που επιτρέπουν την ταυτόχρονη εγγραφή σε

περισσότερους από έναν σκληρούς δίσκους (π.χ. συστήματα RAID) και σε περίπτωση βλάβης ο διαχειριστής δεν έχει παρά να αντικαταστήσει τον σκληρό δίσκο (το σύστημα αυτόματα θα αντιγράψει και σε αυτόν όλα τα δεδομένα).

Βασικές Έννοιες

Μνήμη τυχαίας προσπέλασης (Random Access Memory):

Αποθηκεύει προσωρινά πληροφορίες για όση ώρα είναι ανοικτός ο Η/Υ. Όταν κλείσει ο υπολογιστής η RAM χάνει τις πληροφορίες. Χρησιμοποιείται για ανάγνωση και εγγραφή δεδομένων. Ανήκει στην κύρια μνήμη.

Μνήμη Μόνο για Διάβασμα (Read Only Memory):

Ό,τι έχει καταγραφεί εδώ δεν μπορεί να διαγραφεί από τους απλούς χρήστες. Όταν κλείσει ο υπολογιστής τα δεδομένα που βρίσκονται στη ROM δεν διαγράφονται. Τα δεδομένα που βρίσκονται εδώ είναι απαραίτητα για το ξεκίνημα και τη λειτουργία του υπολογιστή και προσφέρονται μόνο για διάβασμα. Ανήκει στην κύρια μνήμη.

Σκληρός Δίσκος (Hard Disk):

Αποθηκεύει μόνιμα πληροφορίες. Υπάρχει ο εξωτερικός σκληρός δίσκος, ο οποίος μπορεί να μεταφερθεί μεταξύ υπολογιστών και ο εσωτερικός, ο οποίος δεν μεταφέρεται. Έχει μεγάλη χωρητικότητα, είναι γρήγορος σε ανάγνωση και εγγραφή δεδομένων και με την πάροδο του χρόνου το κόστος του μειώνεται. Ανήκει στη βοηθητική μνήμη.

SSD (Solid State Drive)

Λειτουργεί όπως ένα κοινό USB. Είναι αθόρυβος, πιο γρήγορος και πιο αξιόπιστος, καταναλώνοντας παράλληλα λιγότερη ενέργεια από τον σκληρο δίσκο. Είναι μικρότερος σε χωρητικότητα, πιο ακριβός και έχει μικρότερη διάρκεια ζωής.

Μνήμη φλας (USB,SD):

Αποθηκεύει μόνιμα πληροφορίες. Μπορεί να μεταφερθεί εύκολα. Χαμηλό κόστος. Γρήγορο σε ανάγνωση και αποθήκευση δεδομένων. Είναι πιο αργό από τον σκληρό δίσκο. Ανήκει στη βοηθητική μνήμη.

Οπτικός δίσκος (CD, DVD, CD-R/RW, DVD-R/RW):

Αποθηκεύει μόνιμα πληροφορίες. Χαμηλό κόστος. Γρήγορο σε ανάγνωση και αποθήκευση. Ανήκει στη βοηθητική μνήμη.

Πηγές

1. Αράπογλου Α., Μαβόγλου Χ., Οικονομάκος Η., Φύτρος Κ., (2006) *Πληροφορική Α', Β', Γ' Γυμνασίου*, σελ. 16—21, 104—112, ΟΕΔΒ.
2. Wikipedia: <http://en.wikipedia.org/wiki/>
3. Wikipedia: http://el.wikipedia.org/wiki/Δισκέτα_υπολογιστή
4. Wikipedia: http://el.wikipedia.org/wiki/Σκληρός_δίσκος
5. Βουρόπουλος Ηλίας, Γιολλάσης Νεκτάριος, Σαράφη Ιωάννης, Ο Υπολογιστής (Πτυχιακή εργασία), <http://www.it.uom.gr/project/mycomputer/Default.htm> Πανεπιστήμιο Μακεδονίας.
6. http://el.wikipedia.org/wiki/Μνήμη_τυχαίας_προσπέλασης
7. http://www.freescale.com/webapp/sps/site/overview.jsp?code=784_LPBB_DDR

Γ2.4 Εκτυπωτές

Τι θα μάθουμε σήμερα:

- ◆ Να αναγνωρίζουμε και να αναφέρουμε τα κύρια χαρακτηριστικά ενός εκτυπωτή (ανάλυση, ταχύτητα, χρώμα, μέγεθος χαρτιού, κόστος, κ.λπ.)
- ◆ Να αναγνωρίζουμε και να αναφέρουμε τις μονάδες μέτρησης της ανάλυσης και της ταχύτητας ενός εκτυπωτή
- ◆ Να συγκρίνουμε εκτυπωτές με βάση την ταχύτητα, την ανάλυση, το μέγεθος τους, κ.λπ.
- ◆ Να επιλέγουμε από έναν κατάλογο τον πιο κατάλληλο εκτυπωτή για κάποιο παράδειγμα που θα δοθεί.

1. Εισαγωγή

Ανάμεσα στις πιο συνηθισμένες περιφερειακές που χρειάζεται να αγοράσει κάποιος ατόμου απέκτησε υπολογιστή, είναι ο εκτυπωτής. Λιγότερο συχνά και ο σαρωτής. Οι εκτυπωτές και οι σαρωτές, παρά τις σημαντικές διαφορές τους, αφού οι πρώτοι αποτελούν μονάδα εξόδου και οι δεύτεροι μονάδα εισόδου, όπως θα διαπιστωθεί έχουν κοινά στοιχεία, γι' αυτό και τους παρουσιάζουμε μαζί, στο ίδιο κεφάλαιο.

2. Είδη εκτυπωτών

Υπάρχουν διάφορα είδη εκτυπωτών, κάποια πολύ εξειδικευμένα.

Σήμερα στην αγορά βρίσκουμε κυρίως εκτυπωτές ψεκασμού μελάνης και εκτυπωτές λέιζερ και κάποτε συναντούμε εκτυπωτές ακίδων.



Εικόνα 28 Εκτυπωτής ακίδων

Οι **εκτυπωτές ακίδων** (dot matrix printers) είναι παλαιότερης τεχνολογίας και βασίζονται σε έναν κύλινδρο που μετακινεί το χαρτί προς τα πάνω, και σε μία κεφαλή εκτύπωσης που μετακινείται δεξιά-αριστερά και περιέχει μία ή δύο σειρές από ακίδες, που ελέγχονται η κάθε μία ανεξάρτητα από την άλλη. Μεταξύ της κεφαλής εκτύπωσης και του χαρτιού υπάρχει μια ταινία ποτισμένη με μελάνι. Όταν οι ακίδες χτυπούν στην ταινία, αφήνουν μια μικρή κουκίδα μελανιού στο χαρτί. Το κάθε γράμμα (ή εικόνα) αποτελείται από σειρές από κουκκίδες. Οι εκτυπωτές ακίδων είναι αρκετά θορυβώδεις κατά τη λειτουργία τους, λόγω του ότι οι ακίδες είναι σαν μικρά σφυράκια που χτυπούν στη μελανοταινία. Για τον λόγο αυτό, η χρήση τους περιορίστηκε σε συγκεκριμένες εφαρμογές.

Οι θερμικοί εκτυπωτές (thermal printers) βασίζονται σε παρόμοια τεχνολογία, αλλά οι ακίδες θερμαίνονται και ακουμπούν απευθείας το χαρτί. Αντί για μελανοταινία, χρησιμοποιούν ειδικό χαρτί ποτισμένο με ειδική ουσία, που όταν θερμαίνεται από τις ακίδες, αποχτά μαύρο χρώμα.

Οι **εκτυπωτές ψεκασμού μελάνης** (ink jet printers) είναι οι πιο διαδεδομένοι σήμερα. Εάν έχετε εκτυπωτή στο σπίτι, είναι πολύ πιθανόν να είναι αυτού του τύπου. Όπως και οι εκτυπωτές ακίδων, έχουν μηχανισμό (κύλινδρο) για μετακίνηση του χαρτιού και κεφαλή εκτύπωσης που μετακινείται δεξιά-αριστερά.



Εικόνα 29 Εκτυπωτής ψεκασμού μελάνης

Αντί για ακίδες, η κεφαλή εκτύπωσης έχει μικροσκοπικές τρυπίτσες που «ψεκάζουν» μικρές σταγόνες (κουκκίδες) από μελάνι στο χαρτί. Η κάθε κουκίδα έχει μέγεθος μικρότερο από το πάχος μιας τρίχας. Αντί για μελανοταινία, έχουν δοχείο με μελάνι, το οποίο είναι τοποθετημένο στην κεφαλή εκτύπωσης. Συνήθως υπάρχουν ένα δοχείο με μαύρο μελάνι και τρία με βασικά χρώματα, κυανό (cyan), ματζέντα (magenta) και κίτρινο (yellow), από τα οποία ο εκτυπωτής συνθέτει

μια μεγάλη ποικιλία χρωμάτων. Έτσι, έχουν τη δυνατότητα να εκτυπώνουν κείμενο, εικόνες, αλλά και φωτογραφίες με σχετικά καλή ποιότητα.

Οι **εκτυπωτές λέιζερ** (laser printers) ονομάζονται έτσι, διότι αντί κεφαλής εκτύπωσης έχουν μια λεπτή ακτίνα φωτός (λέιζερ) που ζωγραφίζει την εικόνα για εκτύπωση σε φωτοευαίσθητο υλικό. Στην πραγματικότητα, η εκτύπωση βασίζεται περισσότερο στον στατικό ηλεκτρισμό, δηλαδή στις δυνάμεις που ασκεί μια συγκέντρωση ηλεκτρικού φορτίου. Για παράδειγμα, εάν τρίψουμε ένα πλαστικό στυλό με μάλλινο ύφασμα συγκεντρώνεται σ' αυτό ηλεκτρικό φορτίο και εάν το πλησιάσουμε σε μικρά χαρτάκια, τα τραβά και κολλάνε στο στυλό. Υπάρχουν δύο είδη ηλεκτρικού φορτίου, θετικό και αρνητικό, τα οποία έλκονται το ένα προς το άλλο, αλλά απωθούν φορτία του ίδιου τύπου. Πώς αυτό το φαινόμενο μπορεί να χρησιμοποιηθεί για εκτύπωση, θα το δούμε στη συνέχεια.



Εικόνα 30 Εκτυπωτής λέιζερ

Οι εκτυπωτές λέιζερ είναι πολύπλοκα μηχανήματα. Η βάση της εκτύπωσης όμως είναι ένα φωτοευαίσθητο τύμπανο (drum), δηλαδή ένας κύλινδρος που έχει την καμπύλη του επιφάνεια καλυμμένη με ειδικό υλικό. Πριν την εκτύπωση της κάθε σελίδας, το τύμπανο φορτίζεται ομοιόμορφα με αρνητικό ηλεκτρικό φορτίο. Στη συνέχεια μια λεπτή ισχυρή ακτίνα φωτός (λέιζερ), πιο λεπτή από το πάχος μιας τρίχας, κατευθύνεται από ένα σύστημα από περιστρεφόμενους καθρέφτες και φακούς στην επιφάνεια του τυμπάνου και τη κτυπά σε λεπτές, πυκνές γραμμές δεξιά-αριστερά, ενώ παράλληλα το τύμπανο περιστρέφεται.

Η ακτίνα λέιζερ ενεργοποιείται ή απενεργοποιείται, «ζωγραφίζοντας» έτσι με φως μικρές κουκκίδες που σχηματίζουν τη σελίδα στον κύλινδρο. Το υλικό με το οποίο είναι καλυμμένος ο κύλινδρος έχει την ιδιότητα να χάνει το ηλεκτρικό του φορτίο στα σημεία όπου έχει χτυπήσει η φωτεινή ακτίνα. Αντί για μελάνι, οι εκτυπωτές λέιζερ χρησιμοποιούν ένα δοχείο με λεπτή σκόνη (toner), παρόμοια με αυτήν που χρησιμοποιείται στις φωτοτυπικές μηχανές. Μικρή ποσότητα σκόνης μεταφέρεται κοντά στο τύμπανο, ενώ μεταφέρεται σε αυτήν αρνητικό ηλεκτρικό φορτίο. Η σκόνη προσκολλάται (όπως τα χαρτάκια στο στυλό) μόνο στις περιοχές όπου χτύπησε προηγουμένως η ακτίνα λέιζερ (απωθείται από τις άλλες περιοχές), σχηματίζοντας την εικόνα της σελίδας στο τύμπανο. Για να μεταφερθεί αυτή στο χαρτί, το χαρτί φορτίζεται με θετικό ηλεκτρικό φορτίο και καθώς περνά από τον κύλινδρο, η λεπτή σκόνη μεταφέρεται στο χαρτί, αλλά εάν τη σκουπίσουμε θα φύγει. Γι' αυτό αμέσως μετά το χαρτί περνά από έναν ζεστό κύλινδρο (fuser) που πιέζει και θερμαίνει τη σκόνη, ώστε αυτή να λειώσει και να κολλήσει οριστικά στο χαρτί, παράγοντας την τυπωμένη εικόνα. Αμέσως μετά το τύμπανο καθαρίζεται από σκόνη και ηλεκτρικό φορτίο, ώστε να είναι έτοιμο για την επόμενη εκτύπωση. Για έγχρωμη εκτύπωση η διαδικασία επαναλαμβάνεται για καθένα από τα τρία βασικά χρώματα σκόνης και για το μαύρο, τα οποία βρίσκονται σε ξεχωριστά δοχεία.

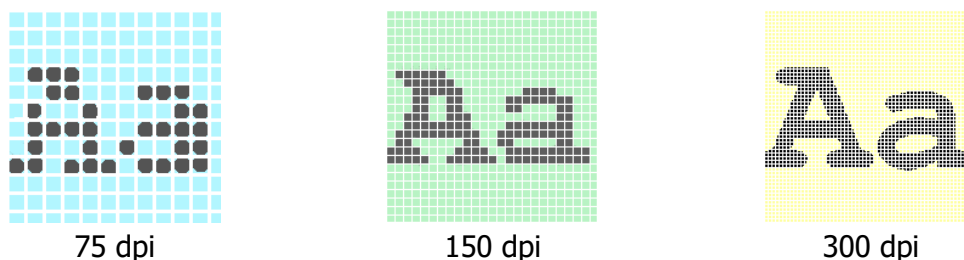
3. Χαρακτηριστικά εκτυπωτών

Ακόμη κι εάν αποφασίζαμε τον τύπο του εκτυπωτή που χρειαζόμαστε, υπάρχουν πολλά μοντέλα στην αγορά. Για να διαλέξουμε τον πιο κατάλληλο θα πρέπει να μελετήσουμε και να συγκρίνουμε τα χαρακτηριστικά τους, όπως η ποιότητα και η ταχύτητα. Υπάρχουν όμως και άλλα τα οποία θα δούμε πιο κάτω.

3.1 Ανάλυση

Και οι τρεις κατηγορίες εκτυπωτών τυπώνουν στην ουσία σειρές από κουκκίδες. Εάν οι κουκκίδες είναι μεγάλες, σε μία συγκεκριμένη απόσταση τυπώνονται λιγότερες κουκκίδες,

ενώ εάν είναι μικρότερες τυπώνονται περισσότερες. Πυκνές κουκκίδες σημαίνουν περισσότερη λεπτομέρεια, και κατά συνέπεια καλύτερη ποιότητα εκτύπωσης. Αυτό το χαρακτηριστικό ονομάζεται **ανάλυση** εκτύπωσης και η μονάδα μέτρησής της είναι **κουκκίδες ανά ίντσα (dots per inch, dpi)**, δηλαδή ο αριθμός κουκκίδων που μπορεί να τυπώνει ο εκτυπωτής σε μια συγκεκριμένη απόσταση (1 ίντσα=2,54 cm).



Εικόνα 31 Ανάλυση εκτύπωσης

Η ανάλυση ενός εκτυπωτή ακίδων είναι γενικά χαμηλότερη από τα άλλα είδη εκτυπωτών (75 dpi). Οι εκτυπωτές ψεκασμού μελάνης και οι εκτυπωτές λέιζερ έχουν ανάλυση από 300dpi και άνω, πολλές φορές 600 dpi ή και 1200 dpi.

3.2 Ταχύτητα

Ένα άλλο βασικό χαρακτηριστικό είναι η ταχύτητα εκτύπωσης, που μετριέται συνήθως σε **σελίδες ανά λεπτό (pages per minute, ppm)**. Για εκτυπωτές που δέχονται συνεχές χαρτί (π.χ. ακίδων), συχνά η ταχύτητα μετριέται σε **χαρακτήρες ανά δευτερόλεπτο (characters per second, cps)**.

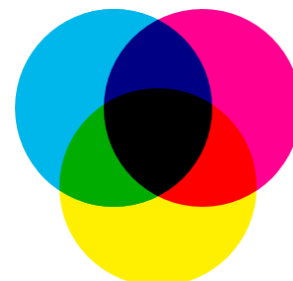
Γενικά οι εκτυπωτές λέιζερ τυπώνουν με μεγαλύτερη ταχύτητα από τους υπόλοιπους. Έχουν επίσης το χαρακτηριστικό ότι η ταχύτητα εκτύπωσής τους είναι σταθερή, είτε τυπώνουν μια σελίδα με λίγο κείμενο, είτε μια ολοσέλιδη φωτογραφία. Αυτό δεν ισχύει για τα άλλα είδη εκτυπωτών, αφού η ταχύτητα εξαρτάται από το πόσο πολύπλοκη ή γεμάτη είναι η σελίδα.

3.3 Δυνατότητα χρώματος

Οι εκτυπωτές ψεκασμού μελάνης που πωλούνται στην αγορά σήμερα είναι κυρίως για χρήση στο σπίτι, όπου παρουσιάζεται συχνά η ανάγκη για εκτύπωση κάποιας έγχρωμης φωτογραφίας ή εικόνας. Έτσι, συνήθως έχουν τη δυνατότητα έγχρωμης εκτύπωσης. Γι' αυτό διαθέτουν ξεχωριστά δοχεία με μελάνι σε τρία «βασικά» χρώματα, κυανό (cyan), ματζέντα (magenta) και κίτρινο (yellow), από τα οποία ο εκτυπωτής συνθέτει όλα τα υπόλοιπα, ψεκάζοντας στο ίδιο σημείο την απαραίτητη ποσότητα από το κάθε χρώμα.

Ο συνδυασμός ίσων ποσοτήτων από τα τρία χρώματα δημιουργεί το μαύρο χρώμα. Επειδή όμως είναι αδύνατο να συνδυαστούν ακριβώς, συνήθως προκύπτει μια σκούρα απόχρωση του καφέ. Γι' αυτόν το λόγο υπάρχει και ένα ξεχωριστό τέταρτο δοχείο με πραγματικό μαύρο χρώμα.

Πολλοί εκτυπωτές λέιζερ, τυπώνουν μόνο ασπρόμαυρα. Υπάρχουν όμως πολλές εφαρμογές όπου το χρώμα δεν είναι απαραίτητο, αλλά η ταχύτητα και η ανάλυση είναι. Παραδείγματα είναι η εκτύπωση επιστολών για πελάτες ενός γραφείου και οι εβδομαδιαίες εκθέσεις που ετοιμάζονται για ενημέρωση του διευθυντή ενός υπουργείου.



Εικόνα 32 Σύνθεση χρωμάτων με συνδυασμό κυανού, ματζέντα και κίτρινου χρώματος

Είναι σημαντικό να αναφερθεί ότι το μελάνι στους εκτυπωτές ψεκασμού μελάνης χρειάζεται κάποιο χρόνο να στεγνώσει και μπορεί να επηρεαστεί από την υγρασία, ενώ η σκόνη στις τυπωμένες σελίδες από έναν εκτυπωτή λέιζερ δεν επηρεάζεται από τέτοια προβλήματα.

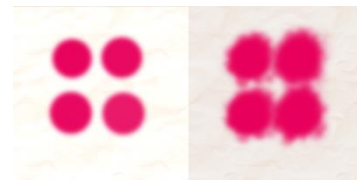
3.4 Είδος και μέγεθος χαρτιού

Οι περισσότεροι εκτυπωτές είναι κατασκευασμένοι για να τυπώνουν σε χαρτί συνηθισμένου μεγέθους (A4). Μπορούν, όμως, να προσαρμοστούν σε χαρτί μικρότερου μεγέθους, καθώς και σε φακέλους. Το μέγιστο μέγεθος χαρτιού είναι το A4. Κάποιοι εκτυπωτές μπορούν να τυπώνουν σε χαρτί μεγαλύτερου μεγέθους, για παράδειγμα σε A3.

Επίσης, υπάρχουν εκτυπωτές που μπορούν αυτόματα να τυπώνουν και στις δύο πλευρές του χαρτιού (duplex).

Οι εκτυπωτές ακίδων έχουν, επίσης, τη δυνατότητα να τυπώνουν σε συνεχές χαρτί (σε ρολό ή σε συνεχόμενα φύλλα). Μπορούν, επίσης, να τυπώνουν ταυτόχρονα σε πολλά αντίγραφα, λόγω του τρόπου λειτουργίας τους (οι ακίδες εξασκούν πίεση στο χαρτί, που μεταφέρεται και στα υπόλοιπα αντίγραφα-καρμπόν). Αυτό είναι χρήσιμο σε κάποιες εφαρμογές, όπως είναι η εκτύπωση τιμολογίων και αποδείξεων, όπου χρειάζεται να κρατείται αντίγραφο για έλεγχο.

Τέλος, χρειάζεται να αναφερθεί ότι το απλό χαρτί γραφείου επιτρέπει στο μελάνι να απλώνει, ενώ παράλληλα έχει σκούρα επιφάνεια, με αποτέλεσμα τα χρώματα να έχουν λιγότερη αντίθεση. Για τους εκτυπωτές αυτούς υπάρχει και ειδικό χαρτί με πιο λεία και λευκή επιφάνεια, κάποτε καλυμμένη με ειδικά υλικά, ώστε να περιορίζουν αυτά τα φαινόμενα, αλλά κοστίζει περισσότερα χρήματα από το συνηθισμένο χαρτί. Υπάρχει, επίσης, ειδικό χαρτί για εκτύπωση φωτογραφιών, που έχει γυαλιστερή επιφάνεια.



Εικόνα 33 Εκτύπωση τεσσάρων κουκκίδων από εκτυπωτή ψεκασμού μελάνης (σε μεγέθυνση), σε ειδικό χαρτί (αριστερά) και σε απλό χαρτί γραφείου (δεξιά)

3.5 Τύπος σύνδεσης

Οι περισσότεροι εκτυπωτές έχουν σύνδεση για θύρα USB, η οποία όπως αναφέρθηκε σε προηγούμενα κεφάλαια έχει επικρατήσει ως η κύρια θύρα σύνδεσης περιφερειακών συσκευών. Κάποιοι παλαιότεροι εκτυπωτές έχουν σύνδεση για παράλληλη θύρα (και επομένως για να τους συνδέσουμε ίσως να χρειαστεί να εγκαταστήσουμε στον υπολογιστή μας κατάλληλη κάρτα επέκτασης, εάν δεν έχει τέτοια θύρα). Σήμερα, οι περισσότεροι εκτυπωτές έχουν κάρτα δικτύου ή ασύρματη κάρτα δικτύου, ώστε να μπορούν να συνδεθούν απευθείας σε δίκτυο υπολογιστών και να εξυπηρετούν πολλούς χρήστες (όπως σε ένα μικρό γραφείο), αλλά είναι κάπως πιο ακριβοί. Υπάρχουν και εκτυπωτές που έχουν περισσότερους από έναν τρόπο σύνδεσης.

Όποια κι αν είναι η περίπτωση, πριν από την αγορά θα πρέπει να εξακριβωθεί ότι ο εκτυπωτής μπορεί να συνδεθεί με τους υπολογιστές που πρόκειται να τον χρησιμοποιούν.

3.6 Θόρυβος

Οι εκτυπωτές ακίδων ανήκουν στην κατηγορία των **κρουστικών εκτυπωτών**, αφού οι ακίδες χτυπούν προκαλώντας αρκετό θόρυβο κατά τη λειτουργία τους, σε αντίθεση με τους εκτυπωτές ψεκασμού μελάνης και ιδιαίτερα τους εκτυπωτές λέιζερ, που ανήκουν στην κατηγορία των **μη κρουστικών εκτυπωτών**, αφού δεν υπάρχει μηχανισμός που να χτυπά.

3.7 Κόστος αγοράς και κόστος εκτύπωσης

Πολλές φορές δίνουμε σημασία μόνο στο **κόστος αγοράς** ενός εκτυπωτή (την τιμή που θα πληρώσουμε για να τον αγοράσουμε). Θα πρέπει, όμως, να λαμβάνουμε υπόψη μας και το **κόστος εκτύπωσης**, διότι σε πολλές περιπτώσεις ξεπερνά κατά πολύ το κόστος αγοράς. Πολλοί προμηθευτές μπορεί να μειώσουν το κόστος αγοράς, διότι γνωρίζουν ότι θα εισπράξουν πολύ περισσότερα από την πώληση μελανιών που θα χρειάζεται να αντικαθιστούνται, διότι εξαντλούνται γρήγορα. Σε άλλες περιπτώσεις, μέρος του μηχανισμού εκτύπωσης είναι ενσωματωμένο με το δοχείο με μελάνι (το σύστημα ψεκασμού σε έναν εκτυπωτή ψεκασμού μελάνης) ή το toner (το τύμπανο σε έναν εκτυπωτή λέιζερ).

Έτσι, φτηνοί στην αγορά τους εκτυπωτές, όπως είναι οι ψεκασμού μελάνης, έχουν ψηλό κόστος εκτύπωσης, αφού με μία σειρά από μελάνια εκτυπώνουν μικρό αριθμό σελίδων. Οι εκτυπωτές λέιζερ συχνά είναι ακριβότεροι στην αγορά τους, αλλά έχουν γενικά χαμηλότερο κόστος εκτύπωσης, αφού με ένα δοχείο με σκόνη (toner) μπορεί να εκτυπώσει πολύ περισσότερες σελίδες. Το κόστος εκτύπωσης συνήθως υπολογίζεται ανά σελίδα.

Παράδειγμα:

| Εκτυπωτής | Τύπος | Κόστος Αγοράς | Κόστος μελανιών/toner | Αριθμός σελίδων που εκτυπώνουν | Κόστος ανά σελίδα |
|-----------|-------------------|---------------|-----------------------|--------------------------------|-------------------|
| A | Ink Jet | € 32 | € 27 (€ 12+€ 15) | 190 | € 0,142 |
| B | Ink Jet | € 90 | €44 (4×€ 11) | 420 | € 0,100 |
| Γ | Laser μαυρόασπρος | € 80 | € 70 | 2500 | € 0,028 |
| Δ | Laser έγχρωμος | € 180 | €300 (4×€ 75) | 2500 | € 0,120 |

Ο εκτυπωτής A είναι ο πιο φθηνός στην αγορά. Ακόμη και τα μελάνια του είναι φθηνότερα, αλλά επειδή εκτυπώνει λιγότερες σελίδες από τους άλλους τρεις εκτυπωτές, το κόστος ανά σελίδα είναι πιο ψηλό από τους άλλους.

Ο εκτυπωτής λέιζερ Γ έχει το χαμηλότερο κόστος εκτύπωσης, έστω και αν η αγορά του στοιχίζει περισσότερα από τον πρώτο. Εάν τυπώνουμε κυρίως ασπρόμαυρα, οι δύο εκτυπωτές λέιζερ δεν έχουν σημαντική διαφορά στο κόστος εκτύπωσης (αλλά θα πρέπει να αναρωτηθεί κάποιος κατά πόσο δικαιολογείται το κόστος αγοράς του για λίγα έγχρωμα αντίγραφα).

Η αγορά ενός εκτυπωτή εξαρτάται σε μεγάλο βαθμό από το κόστος αγοράς του, το κόστος εκτύπωσης ανά σελίδα, τον αριθμό σελίδων που ο χρήστης θέλει να εκτυπώνει κάθε μήνα και τον χρόνο που σκοπεύει να χρησιμοποιήσει τον εκτυπωτή πριν τον αντικαταστήσει. Για παράδειγμα, εάν θέλουμε έναν εκτυπωτή για 3 χρόνια και σκοπεύουμε να τυπώνουμε 1000 σελίδες τον μήνα (όπως ένα μικρό γραφείο), δηλαδή $1000 \times 12 \times 3 = 36000$ σελίδες σε 3 χρόνια. Ο εκτυπωτής A θα μας στοιχίζει ανά σελίδα μέχρι την αντικατάστασή του $(€ 32 + € 0,142 \times 36000) / 36000 = € 0,143$, ενώ ο Γ $(€ 80 + € 0,028 \times 36000) / 36000 = € 0,030$ και ο Γ θα ήταν πιο οικονομική λύση. Εάν όμως ο όγκος εκτύπωσης είναι χαμηλότερος, π.χ. 10 σελίδες τον μήνα (όπως στο σπίτι για να τυπώνουμε που και που), δηλαδή $10 \times 12 \times 3 = 360$ σελίδες σε 3 χρόνια, τότε ο A θα μας στοιχίζει ανά σελίδα μέχρι την αντικατάστασή του $(€ 32 + € 0,142 \times 360) / 360 = € 0,231$, ενώ ο Γ $(€ 80 + € 0,028 \times 360) / 360 = € 0,250$ και ο A θα ήταν πιο οικονομική λύση.

Βασικές Έννοιες

Εκτυπωτής Ακίδων (Dot Matrix Printer): Απλός εκτυπωτής που περιέχει σειρές από ακίδες, που χτυπούν μια μελανοταινία και αφήνουν κουκκίδες στο χαρτί. Αρκετά θορυβώδης, πολύ φθηνή εκτύπωση, αλλά με χαμηλή ποιότητα.

Εκτυπωτής Ψεκασμού Μελάνης (Ink Jet Printer): Ο πιο διαδομένος για χρήση στο σπίτι. Έχει δοχεία με μελάνι και μικροσκοπικές τρυπίτσες που «ψεκάζουν» κουκκίδες από μελάνι στο χαρτί. Συνήθως τυπώνουν έγχρωμα με σχετικά καλή ποιότητα.

Εκτυπωτής Λείζερ (Laser Printer): Έχουν μια λεπτή ακτίνα φωτός (λείζερ) που ζωγραφίζει την εικόνα για εκτύπωση σε φωτοευαίσθητο υλικό..

Χαρακτηριστικά εκτυπωτών

Ανάλυση (Resolution): Με πόση λεπτομέρεια γίνεται η εκτύπωση. Μετριέται σε κουκκίδες ανά ίντσα (dots per inch, dpi), π.χ. 300 dpi.

Ταχύτητα (Speed): Η ταχύτητα εκτύπωσης μετριέται συνήθως σε σελίδες ανά λεπτό (pages per minute, ppm). Για εκτυπωτές που δέχονται συνεχές χαρτί (π.χ. ακίδων), μετριέται σε χαρακτήρες ανά δευτερόλεπτο (characters per second, cps).

Δυνατότητα χρώματος: Έγχρωμη ή ασπρόμαυρη εκτύπωση. Για εκτύπωση απαιτούνται ξεχωριστά δοχεία με χρώμα, κυανό (cyan), ματζέντα (magenta), κίτρινο (yellow), και μαύρο (black), από τα οποία ο εκτυπωτής συνθέτει όλα τα υπόλοιπα.

Είδος και μέγεθος χαρτιού: Για εκτυπωτές, το μεγαλύτερο μέγεθος χαρτιού στο οποίο μπορούν να τυπώσουν. Συνήθως είναι το μέγεθος A4, αλλά κάποτε είναι μεγαλύτερο (π.χ. A3).

Οι εκτυπωτές ακίδων έχουν τη δυνατότητα να τυπώνουν επίσης σε συνεχές χαρτί (ρολό) και σε πολλά αντίτυπα ταυτόχρονα.

Επεξεργασία διπλής όψης (duplex): Όταν ο εκτυπωτής μπορεί αυτόματα να τυπώνει/σαρώνει και στις δύο πλευρές του χαρτιού.

Τύπος σύνδεσης: Οι περισσότεροι εκτυπωτές έχουν σύνδεση για θύρα USB. Κάποιοι παλαιότεροι έχουν σύνδεση για παράλληλη θύρα. Ορισμένοι έχουν κάρτα δικτύου ή ασύρματη κάρτα δικτύου.

Θόρυβος: Οι εκτυπωτές ακίδων προκαλούν αρκετό θόρυβο κατά τη λειτουργία, σε αντίθεση με τους υπόλοιπους. Ανήκουν στην κατηγορία των κρουστικών εκτυπωτών, (οι ακίδες χτυπούν), σε αντίθεση με τους εκτυπωτές ψεκασμού μελάνης και λείζερ, που ανήκουν στην κατηγορία των μη κρουστικών εκτυπωτών.

Κόστος αγοράς: Η τιμή που θα πληρώσουμε όταν αγοράσουμε τον εκτυπωτή.

Κόστος εκτύπωσης: Το κόστος αγοράς μελανιού/toner και άλλων αναλώσιμων υλικών (σε πολλές περιπτώσεις ξεπερνά κατά πολύ το κόστος αγοράς). Για σύγκριση εκτυπωτών υπολογίζεται το κόστος εκτύπωσης ανά σελίδα.

Πηγές

1. Βουτηράς Γ., Κονιδάρη Ε., Κούτρας Μ., Σφώρος Ν., (2003) *Πληροφορική Γυμνασίου*, σελ. 42—48, ΟΕΔΒ
2. Βικιπαίδεια, *Λήμματα για Εκτυπωτές και Σαρωτές*:
<http://el.wikipedia.org/wiki/Εκτυπωτής>
http://en.wikipedia.org/wiki/Dot_matrix_printer
http://en.wikipedia.org/wiki/Thermal_printer
http://en.wikipedia.org/wiki/Inkjet_printer
http://en.wikipedia.org/wiki/Laser_printer
<http://el.wikipedia.org/wiki/Σαρωτής>
http://en.wikipedia.org/wiki/Image_scanner
3. Ιστοχώρος, How Stuff Works, *Λήμματα για Εκτυπωτές και Σαρωτές*:
<http://computer.howstuffworks.com/inkjet-printer.htm>
<http://computer.howstuffworks.com/laser-printer.htm>
<http://computer.howstuffworks.com/scanner.htm>
4. Βίντεο από την εταιρία Lexmark: *Πώς λειτουργούν οι εκτυπωτές*:
<http://www.youtube.com/watch?v=PHxXmjxcAIo>
<http://www.youtube.com/watch?v=WHurJclBPYA>
5. Άλλοι Ιστοχώροι για *Εκτυπωτές*:
<http://www.inkline.gr/>
<http://pacific.jour.auth.gr/hardware/printer.htm>
http://www.ehow.com/how-does_4928286_how-dot-matrix-printer-works.html

Γ2.5 Οθόνες

Τι θα μάθουμε σήμερα:

- ❖ Να αναγνωρίζουμε και να αναφέρουμε τα κύρια χαρακτηριστικά μιας οθόνης (τεχνολογία, ανάλυση, μέγεθος, κόστος, κ.λπ.)
- ❖ Να αναγνωρίζουμε και να αναφέρουμε τις μονάδες μέτρησης της ανάλυσης και του μεγέθους μιας οθόνης
- ❖ Να συγκρίνουμε οθόνες με βάση την ανάλυση και το μέγεθός τους
- ❖ Να επιλέγουμε από έναν κατάλογο την πιο κατάλληλη οθόνη για κάποιο παράδειγμα που θα δοθεί.

1. Εισαγωγή

Η οθόνη είναι η πιο συνηθισμένη μονάδα εξόδου ενός σύγχρονου υπολογιστή. Είναι αδύνατον να φανταστούμε έναν υπολογιστή χωρίς κάποιας μορφής οθόνη, από τον μικρότερο υπολογιστή παλάμης, μέχρι τον μεγαλύτερο υπερυπολογιστή. Στο κεφάλαιο αυτό θα επικεντρωθούμε στις οθόνες που συναντούμε σε προσωπικούς υπολογιστές και στα χαρακτηριστικά τους.

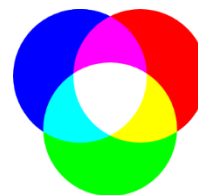


Εικόνα 34 Μια οθόνη αποτελείται από εικονοστοιχεία, το καθένα από αυτά έχει τρεις φωτεινές πηγές, για κόκκινο, πράσινο και μπλε χρώμα

2. Πώς λειτουργεί μια οθόνη

Όλες οι οθόνες, άσχετα από τον τρόπο κατασκευής τους, αποτελούνται από μικρές κουκκίδες που ονομάζονται **εικονοστοιχεία (pixels)** οργανωμένα σε σειρές (γραμμές) και στήλες, και κατάλληλους μηχανισμούς και κυκλώματα για να ρυθμίζουν τη φωτεινότητα του κάθε εικονοστοιχείου.

Σε μια έγχρωμη οθόνη, το κάθε εικονοστοιχείο αποτελείται από τρία μέρη που το καθένα μπορεί να εκπέμπει φως διαφορετικού χρώματος, κόκκινο (red) πράσινο (green) και μπλε (blue). Το μείγμα των χρωμάτων αυτών σε διαφορετικές εντάσεις δημιουργεί όλα τα χρώματα που βλέπουμε.



Εικόνα 35 Σύνθεση χρωμάτων με συνδυασμό κόκκινου, πράσινου και μπλε φωτός

Οι πληροφορίες για τον συνδυασμό χρωμάτων του κάθε εικονοστοιχείου αποστέλλονται στην οθόνη από την κάρτα γραφικών, με τη σειρά. Μόλις συμπληρωθεί η αποστολή πληροφοριών σε όλα τα εικονοστοιχεία, η διαδικασία επαναλαμβάνεται, ώστε ολόκληρη η οθόνη να «ανανεώνεται» πολλές φορές το δευτερόλεπτο (20 ή περισσότερες), ώστε στα μάτια μας να δίνεται η ψευδαίσθηση ότι η εικόνα «κινείται». Έτσι, όταν παρακολουθούμε βίντεο στην οθόνη μας, στην πραγματικότητα βλέπουμε εικόνες να εναλλάσσονται με μεγάλη ταχύτητα.

3. Είδη Οθόνης

Στα διάφορα είδη οθόνης διαφέρει κυρίως ο μηχανισμός δημιουργίας και λειτουργίας των εικονοστοιχείων. Για πολλά χρόνια επικρατούσαν οι οθόνες **Καθοδικού Σωλήνα (Cathode Ray Tube, CRT)**. Τα τελευταία όμως χρόνια έχουν αντικατασταθεί πλήρως από οθόνες υγρών κρυστάλλων (**Liquid Crystal Display, LCD**) και διάφορες παραλλαγές τους.

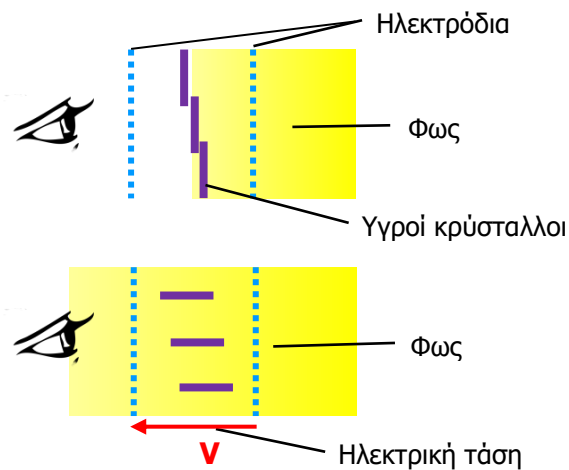
3.1 Οθόνη Υγρών Κρυστάλλων (Liquid Crystal Display, LCD)

Οι οθόνες αυτές είναι αρκετά λεπτές και τις συναντούμε σε όλους τους σύγχρονους υπολογιστές. Η οθόνη είναι καλυμμένη με εικονοστοιχεία που περιέχουν υγρούς κρυστάλλους. Οι υγροί κρύσταλλοι είναι ουσίες που συμπεριφέρονται σαν κρύσταλλοι και είναι διαφανείς στο φως, αλλά καθώς εφαρμόζουμε σε αυτούς ηλεκτρική τάση, αλλάζουν μορφή και εμποδίζουν το φως να περάσει (χρησιμοποιούνται σε υπολογιστικές μηχανές, ρολόγια και άλλες συσκευές). Το κάθε εικονοστοιχείο έχει τρία ξεχωριστά μέρη, που το καθένα έχει μπροστά κόκκινο, πράσινο ή μπλε φίλτρο, έτσι ώστε όταν φωτίζεται από μια φωτεινή πηγή που βρίσκεται πίσω από την οθόνη να εκπέμπει μόνο το φως του συγκεκριμένου χρώματος. Πίσω από το κάθε φίλτρο υπάρχουν οι υγροί κρύσταλλοι και πολύ λεπτά καλώδια για να εφαρμόζεται σε αυτούς ηλεκτρική τάση, ανάλογη με την ένταση του φωτός που χρειάζεται να εκπέμπουν.



Εικόνα 36

Οθόνη υγρών κρυστάλλων



Εικόνα 37

Αρχή λειτουργίας οθόνης υγρών κρυστάλλων

Η φωτεινή πηγή είναι συνήθως σειρές από λαμπτήρες φθορισμού ή σειρές από φωτοδιόδους (light-emitting diodes, LEDs). Οι οθόνες αυτού του τύπου μπορούν να κατασκευαστούν ώστε να είναι πολύ λεπτές, αφού αποτελούνται από τη φωτεινή πηγή και δύο λεπτά διαφανή φύλλα που περιέχουν τους υγρούς κρυστάλλους. Έχουν, επίσης, χαμηλότερη κατανάλωση ενέργειας από τις οθόνες καθοδικού σωλήνα.

Μια παραλλαγή αυτού του τύπου οθόνης, έχει επικρατήσει σήμερα: σε κάθε χρώμα του κάθε εικονοστοιχείου υπάρχει ένα Τρανζίστορ Λεπτής Μεμβράνης (Thin Film Transistor, TFT), από το οποίο προέρχεται και το όνομά του (**LCD-TFT Monitor**), και βοηθά το εικονοστοιχείο να φαίνεται φωτεινότερο.

3.2 Άλλα είδη οθόνης

Οι οθόνες LCD-TFT παρά τη διάδοσή τους έχουν κάποια σοβαρά μειονεκτήματα: Μόνο μέρος του φωτός που παράγεται από την πηγή αξιοποιείται (το υπόλοιπο χάνεται) και καταναλώνεται ενέργεια ακόμη και όταν το εικονοστοιχείο είναι σκοτεινό (για να κρατά τον υγρό κρύσταλλο στην «κλειστή» του μορφή). Τεχνολογικές εξελίξεις και έρευνα σταδιακά οδηγούν σε νέες ιδέες που βρίσκουν τον δρόμο τους στην αγορά, όπως για παράδειγμα υγροί κρύσταλλοι με την ιδιότητα να κρατούν για απεριόριστο χρόνο το «χρώμα» τους, ακόμη και αν αφαιρεθεί εντελώς η ηλεκτρική τάση ή η περισσότερο γνωστή οθόνη Οργανικών Φωτοδιόδων (Organic Light-Emitting Diode, OLED), όπου ο υγρός κρύσταλλος αντικαθιστάται με οργανικές ουσίες που εκπέμπουν φως και δεν χρειάζονται



Εικόνα 38 Ευλύγιστη οθόνη OLED

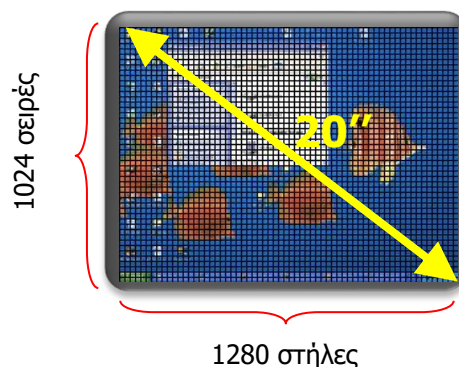
πλέον φωτεινή πηγή. Αυτές ήδη κυκλοφορούν σε φορητές συσκευές με την ονομασία AMOLED (Active Matrix OLED). Μια ενδιαφέρουσα ιδιότητά τους είναι ότι μπορούν να κατασκευαστούν και σε ευλύγιστο υλικό, όπως μεμβράνες, με νέες προοπτικές, για παράδειγμα οθόνες που θα τυλίγονται για να καταλαμβάνουν μικρότερο χώρο.

4. Χαρακτηριστικά οθονών

Οι οθόνες είναι πολύπλοκες συσκευές και έχουν αλληλοεξαρτώμενα τεχνικά χαρακτηριστικά (πολλά από τα οποία δεν δίνονται από τον κατασκευαστή), τα οποία δεν είναι πάντοτε εύκολο να συγκρίνουμε. Τα βασικά χαρακτηριστικά που συναντούμε στην αγορά σχολιάζονται πιο κάτω, αλλά πολλές φορές χρειάζεται να δει κάποιος μια συγκεκριμένη οθόνη σε λειτουργία πριν την αγοράσει.

4.1 Ανάλυση

Στις περισσότερες περιπτώσεις η ανάλυση αναφέρεται με τον αριθμό των εικονοστοιχείων σε κάθε σειρά (στήλες) και τον αριθμό των σειρών. Για παράδειγμα μια οθόνη με ανάλυση 1280×1024 έχει 1280 στήλες και 1024 σειρές από εικονοστοιχεία, ενώ μια με 1600×900 έχει 1600 στήλες και 900 σειρές. Ο τρόπος αυτός



Εικόνα 39 Ανάλυση και μέγεθος οθόνης

περιγραφής της ανάλυσης δεν βοηθά στη σύγκριση οθονών διαφορετικού μεγέθους, διότι δεν περιγράφει πόσες κουκκίδες (εικονοστοιχεία) έχουμε ανά ίντσα. Εάν γνωρίζουμε όμως το μήκος και το πλάτος μιας οθόνης, με τον κατάλληλο υπολογισμό μπορούμε να βρούμε πόσες κουκκίδες αντιστοιχούν ανά ίντσα (**dots per inch, dpi**) και να προχωρήσουμε στη σύγκριση, όπως και με τους εκτυπωτές.

4.2 Μέγεθος οθόνης

Έχει καθιερωθεί να ονομάζεται έτσι το μήκος της διαγωνίου της επιφάνειας των εικονοστοιχείων σε ίντσες. Κοινά μεγέθη είναι 10", 14", 15,6", 17", 19", 20", 24" (τα πρώτα τρία είναι μεγέθη για φορητούς υπολογιστές). Ο τρόπος αυτός μέτρησης είναι συχνά παραπλανητικός, διότι δεν έχουν όλες οι οθόνες το ίδιο σχήμα. Κάποιες είναι πιο τετραγωνισμένες, ενώ άλλες είναι στενόμακρες (wide screen).

4.3 Σχήμα οθόνης (λόγος διαστάσεων)

Οι παλαιότερες οθόνες (κυρίως καθοδικού σωλήνα, αλλά και μερικές υγρών κρυστάλλων) είχαν λόγο διαστάσεων 4:3 (δηλαδή εάν χωρίσουμε το πλάτος σε 4 ίσα μέρη, το ύψος θα είναι ίσο με 3 τέτοια μέρη). Υπήρξαν και οθόνες με λόγο 5:4. Τα τελευταία χρόνια έχουν επικρατήσει οι οθόνες με λόγο διαστάσεων 16:9, με μεγαλύτερο πλάτος σε σχέση με το ύψος τους.



Εικόνα 40 Δύο οθόνες με το ίδιο μέγεθος (20"), αλλά διαφορετικούς λόγους διαστάσεων 16:9 (αριστερά),

Δύο οθόνες με το ίδιο μέγεθος (διαγωνίου) αλλά διαφορετικού σχήματος έχουν και διαφορετική επιφάνεια (η πιο «τετραγωνισμένη» έχει μεγαλύτερη επιφάνεια). Για παράδειγμα, μια οθόνη με λόγο διαστάσεων 4:3 και μέγεθος 20" έχει πλάτος 16" και ύψος 12" ενώ μια οθόνη με λόγο διαστάσεων 16:9 και το ίδιο μέγεθος

έχει πλάτος 17,4" και ύψος 9,8" περίπου. Η πρώτη έχει εμβαδόν 192 τετραγωνικές ίντσες, ενώ η δεύτερη μόνο 170,9.

4.4 Αριθμός χρωμάτων που μπορεί να αναπαραστήσει

Γενικά, υπάρχει η εντύπωση ότι οι περισσότερες οθόνες μπορούν να αναπαραστήσουν με ακρίβεια τα 16 εκατομμύρια περίπου χρωμάτων που αποστέλλει μια κάρτα γραφικών (24 bit, 8 bit για κάθε βασικό χρώμα). Στην πραγματικότητα, τα εσωτερικά ηλεκτρονικά μιας οθόνης πιθανόν να αναπαραστήσουν μόνο 6 bit χρώμα και να αγνοούν τα υπόλοιπα. Έτσι, όμοια («γειτονικά») χρώματα πιθανόν να εμφανίζονται στην οθόνη ως απaráλλαχτα. Για γενική χρήση ενός υπολογιστή, αυτό πιθανόν να μην το παρατηρήσουμε. Ένας γραφίστας όμως, θα διαπιστώσει τις μικρές αυτές διαφορές όταν τυπωθεί μια εικόνα που επεξεργάστηκε, ενώ αυτές δεν φαίνονταν στην οθόνη.

4.5 Ρυθμός ανανέωσης πλαισίων (frame refresh rate)

Πρόκειται για την ταχύτητα με την οποία η οθόνη μπορεί να σχεδιάσει μια ολόκληρη εικόνα (ονομάζεται πλαίσιο ή καρτέ), δηλαδή πόσες φορές το δευτερόλεπτο μπορεί να ξανασχεδιάζεται ολόκληρη η οθόνη. Είναι αρκετά υψηλότερη των 25 ανά δευτερόλεπτο (δηλαδή πάνω από 25 Hz), που είναι το όριο κάτω από το οποίο τα μάτια μας εντοπίζουν την οθόνη να τρεμοπαίζει. Συνήθως είναι 60 Hz, 75 Hz, και σε πολλές σύγχρονες οθόνες ξεπερνά τα 100 Hz.

4.6 Χρόνος ανταπόκρισης

Πρόκειται για τον χρόνο που χρειάζεται ένα εικονοστοιχείο για να αλλάξει από μία φωτεινότητα (ή χρώμα) σε άλλη. Αυτός καθορίζεται από την τεχνολογία κατασκευής των εικονοστοιχείων και είναι γενικά μεγαλύτερος στις φθηνότερες/μέτριων επιδόσεων οθόνες. Μετριέται σε χιλιοστά του δευτερολέπτου (milliseconds, ms) και είναι σημαντικό να είναι μικρός εάν η οθόνη θα χρησιμοποιείται για παιχνίδια, βίντεο ή άλλες εφαρμογές όπου πιθανόν να υπάρχουν γρήγορες μεταβολές στα χρώματα.

4.7 Τύπος σύνδεσης

Οι περισσότερες οθόνες έχουν σύνδεση με θύρα **VGA** (Video Graphics Array), ενώ κάποιες νεότερες έχουν καθαρά ψηφιακή σύνδεση με θύρα **DVI** (Digital Visual Interface) ή ακόμη και σύνδεση ψηλής ευκρίνειας **HDMI** (High Definition Multimedia Interface). Οι περισσότερες οθόνες νέας τεχνολογίας προσφέρουν πάνω από έναν τύπο σύνδεσης.

4.8 Άλλα χαρακτηριστικά

Εκτός από τα πιο πάνω, που είναι βασικά χαρακτηριστικά, υπάρχουν και αρκετά άλλα όπως:

- **Η φωτεινότητα της οθόνης (luminosity):** Πόσο φωτεινά είναι τα χρώματα και σε ποιο βαθμό είναι ορατά π.χ. σε εξωτερικούς χώρους με ηλιοφάνεια. Η μονάδα μέτρησής της είναι cd/m².
- **Η γωνία θέασης (viewing angle):** Είναι η μεγαλύτερη γωνία από την κάθετο στην οθόνη υπό την οποία μπορούμε να βλέπουμε την εικόνα ικανοποιητικά. Μπορεί να είναι διαφορετική ως προς την οριζόντια και την κατακόρυφη. Αφορά κυρίως οθόνες υγρών κρυστάλλων και μετριέται σε μοίρες.
- **Λόγος αντίθεσης (contrast ratio):** Είναι ο λόγος της έντασης του φωτός που εκπέμπεται από ένα εικονοστοιχείο όταν έχει άσπρο χρώμα και μαύρο χρώμα. Για παράδειγμα 300:1 σημαίνει ότι το άσπρο είναι 300 φορές πιο φωτεινό από το μαύρο.
- **Κόστος αγοράς**

Βασικές Έννοιες

| | |
|-------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Εικονοστοιχεία (pixels): | Οι κουκκίδες που αναπαριστούν χρώματα στην οθόνη. Είναι οργανωμένα σε σειρές (γραμμές) και στήλες. Το καθένα αποτελείται από τρία μέρη που εκπέμπουν φως διαφορετικού χρώματος, κόκκινο (red), πράσινο (green) και μπλε (blue). |
| Οθόνη Καθοδικού Σωλήνα (Cathode Ray Tube, CRT): | Οθόνες παλαιότερης τεχνολογίας. |
| Οθόνη Υγρών Κρυστάλλων (Liquid Crystal Display): | Επίπεδη οθόνη την οποία συναντούμε στους σύγχρονους υπολογιστές. Τα εικονοστοιχεία περιέχουν υγρούς κρυστάλλους, που εκπέμπουν ή εμποδίζουν το φως (από λάμπες φθορισμού ή φωτοδιόδους) να περάσει από μέσα τους, ανάλογα με την τάση του ρεύματος που εφαρμόζεται στον καθένα από αυτούς. Έχουν χαμηλή κατανάλωση ρεύματος και γι' αυτό χρησιμοποιούνται σε φορητές συσκευές. |
| Οθόνη LCD-TFT: | Η παραλλαγή της οθόνης υγρών κρυστάλλων που επικράτησε σήμερα. Χρησιμοποιεί σε κάθε εικονοστοιχείο Τρανζίστορ Λεπτής Μεμβράνης (Thin Film Transistor, TFT), που βοηθά να φαίνεται φωτεινότερη. |
| Οθόνη Οργανικών Φωτοδιόδων (Organic Light-Emitting Diode, OLED): | Νέος τύπος οθόνης όπου αντί για υγρό κρύσταλλο υπάρχουν οργανικές ουσίες που εκπέμπουν φως, έχουν πολύ μικρή κατανάλωση ρεύματος και δε χρειάζονται φωτεινή πηγή. |
| | Χαρακτηριστικά οθονών |
| Ανάλυση (Resolution): | Ο αριθμός των εικονοστοιχείων σε κάθε σειρά (στήλες) και ο αριθμός των σειρών, π.χ. 1600×900. |
| Μέγεθος (Size): | Το μήκος της διαγωνίου της επιφάνειας των εικονοστοιχείων σε ίντσες, π.χ. 20". |
| Ρυθμός ανανέωσης πλαισίων (frame refresh rate): | Πόσες φορές το δευτερόλεπτο μπορεί να ξανασχεδιάζεται ολόκληρη η οθόνη. Μετριέται σε Hz,, π.χ. 100 Hz. |
| Τύπος σύνδεσης: | Σε τι είδους θύρα του υπολογιστή συνδέεται, π.χ. με θύρα VGA (Video Graphics Array), DVI (Digital Visual Interface) ή σύνδεση ψηλής ευκρίνειας HDMI (High Definition Multimedia Interface). |
| Χρόνος ανταπόκρισης (response time): | Ο χρόνος που χρειάζεται ένα εικονοστοιχείο για να αλλάξει από μία φωτεινότητα (ή χρώμα) σε άλλη. Μετριέται σε χιλιοστά του δευτερολέπτου, π.χ. 3 ms. |
| Κόστος αγοράς: | Η τιμή που θα πληρώσουμε όταν αγοράσουμε την οθόνη. |

Πηγές

1. Βουτηράς Γ., Κονιδάρη Ε., Κούτρας Μ., Σφώρος Ν., (2003) *Πληροφορική Γυμνασίου*, σελ. 44—45, ΟΕΔΒ
2. Βικιπαίδεια, *Λήμματα για οθόνες*:
http://el.wikipedia.org/wiki/Οθόνη_υπολογιστή
http://en.wikipedia.org/wiki/Computer_monitor
http://en.wikipedia.org/wiki/Liquid_crystal_display

Γ2.6 Πώς Συγκρίνουμε και Πώς Επιλέγουμε Υπολογιστικά Συστήματα

Τι θα μάθουμε σήμερα:

- ◆ Να συγκρίνουμε υπολογιστικά συστήματα με βάση τα χαρακτηριστικά του υλικού (ταχύτητα ΚΜΕ, ποσότητα μνήμης RAM, χωρητικότητα δίσκου, χαρακτηριστικά οθόνης, εκτυπωτή και άλλων μονάδων/περιφερειακών συσκευών) και του λογισμικού που διαθέτουν
- ◆ Να επιλέγουμε από έναν κατάλογο κατάλληλο υπολογιστικό σύστημα (ή μέρη του) για εξυπηρέτηση του χρήστη σε ένα παράδειγμα και να στοιχειοθετούμε την απόφασή μας.

1. Εισαγωγή

Έχουμε ήδη ανακεφαλαιώσει τα βασικά χαρακτηριστικά του υλικού (εξαρτημάτων και περιφερειακών μονάδων) του υπολογιστή. Επίσης, θυμηθήκαμε τον ρόλο του λογισμικού εφαρμογών και του λογισμικού συστήματος (κυρίως του λειτουργικού συστήματος) και πώς αυτά σχετίζονται μεταξύ τους. Παράλληλα, αναφερθήκαμε στον ρόλο του χρήστη, που πέρα από την αλληλεπίδρασή του με τον υπολογιστή μέσα από το λογισμικό καθορίζει και ποιο λογισμικό και ποιο υλικό θα πρέπει να υπάρχει, ώστε να καλύπτονται οι υπολογιστικές ανάγκες του. Ακολούθησε ανάλυση των προβλημάτων που πιθανόν να παρουσιαστούν στο υλικό, το λογισμικό και τον χρήστη και παρουσιάστηκαν τρόποι πρόληψης ή αντιμετώπισής τους. Ο χρήστης, το λογισμικό εφαρμογών, το λογισμικό συστήματος και το υλικό μαζί αποτελούν αυτό που ονομάζουμε υπολογιστικό σύστημα. Για να λειτουργεί ικανοποιητικά το υπολογιστικό σύστημα, θα πρέπει τα μέρη του να συνεργάζονται ομαλά και αποτελεσματικά.

Σε αυτό το κεφάλαιο θα μελετήσουμε τι πρέπει να λαμβάνουμε υπόψη μας όταν επιλέγουμε υλικό και λογισμικό, ώστε να εξυπηρετούνται οι ανάγκες του χρήστη, αλλά να υπάρχει ταυτόχρονα και ομαλή συνεργασία μεταξύ τους, χωρίς να ξεχνάμε ότι το κόστος αγοράς και συντήρησης είναι ένας από τους καθοριστικούς παράγοντες.

2. Ο χρήστης και οι υπολογιστικές ανάγκες του

Ο κάθε άνθρωπος χρησιμοποιεί τον υπολογιστή σύμφωνα με τις δικές του ανάγκες. Ένας μαθητής/Μια μαθήτρια πιθανόν χρησιμοποιεί τον υπολογιστή του/της για να αναζητά πληροφορίες στο Διαδίκτυο, να δημιουργεί παρουσιάσεις για τα μαθήματά του/της, να ακούει μουσική, να παίζει παιχνίδια και να επικοινωνεί με φίλους/φίλες του/της. Ένας υπάλληλος σε μια ασφαλιστική εταιρεία χρειάζεται έναν υπολογιστή που να κρατά ημερολόγιο των συναντήσεων με τους πελάτες του, να στέλνει και να λαμβάνει μηνύματα ηλεκτρονικού ταχυδρομείου, να επιτρέπει τον υπολογισμό του κόστους διάφορων ασφαλιστικών σχεδίων (π.χ. σε υπολογιστικό φύλλο) και να έχει διάρκεια μπαταρίας, ώστε να μη χρειάζεται να τον φορτίζει παρά μόνο στο τέλος κάθε εργάσιμης ημέρας. Ένας αρχιτέκτονας χρειάζεται έναν υπολογιστή με μεγάλη οθόνη, κατάλληλο σχεδιαστικό πρόγραμμα και ποντίκι ακριβείας ή γραφίδα και εκτυπωτή ή σχεδιαστή (plotter) για μεγάλο μέγεθος χαρτιού.

Στην κάθε περίπτωση θα πρέπει να εντοπίζονται τα χαρακτηριστικά και οι δυνατότητες που απαιτούνται (οι προδιαγραφές), για να έχουμε όσο το δυνατόν ένα σύστημα που από τη μία να εξυπηρετεί και να καλύπτει τις ανάγκες/απαιτήσεις του χρήστη και από την άλλη να μην περιλαμβάνει υλικό και λογισμικό που δεν είναι αναγκαίο, αυξάνοντας αχρείαστα το κόστος αγοράς. Συχνά παραλείπουμε να λάβουμε υπόψη το κόστος όταν αγοράζουμε ένα υπολογιστικό σύστημα για προσωπική μας χρήση (ιδιαίτερα εάν θα το πληρώσουν οι γονείς μας). Όταν όμως πρόκειται για επαγγελματική χρήση, μια επιχείρηση ή ένας οργανισμός θα

πρέπει να μελετήσει τις ανάγκες και το κόστος προσεκτικά, αφού οι οικονομικές συνέπειες μπορεί να είναι σοβαρές.

Για παράδειγμα, σε μια εταιρεία η αντικατάσταση 200 υπολογιστών με καινούργιους αξίας €1.000 ο καθένας θα στοιχίσει €200.000. Εάν αγοραστούν φθηνότεροι υπολογιστές αξίας €800 ο καθένας, νοουμένου ότι καλύπτονται οι ανάγκες των χρηστών, θα στοιχίσει €160.000, δηλαδή εξοικονομούνται €40.000.

Παράλληλα, μια λανθασμένη επιλογή με κριτήριο μόνο το κόστος, πιθανόν να αποδειχτεί ζημιογόνα, διότι εάν δεν εξυπηρετούνται οι ανάγκες των χρηστών θα χρειαστούν αντικατάσταση ή αναβάθμιση των υπολογιστικών συστημάτων, επιβαρύνοντας έτσι τον προϋπολογισμό.

3. Επιλογή λογισμικού εφαρμογών και λειτουργικού συστήματος

Το λογισμικό εφαρμογών που απαιτείται προκύπτει από τις ανάγκες του χρήστη, αφού με αυτό θα τις εξυπηρετεί. Οι κύριοι παράγοντες που επηρεάζουν την επιλογή είναι το **κόστος αγοράς** του λογισμικού και πιθανή προηγούμενη εμπειρία του χρήστη ή απαίτηση για συγκεκριμένο λογισμικό. Για παράδειγμα, ένας διευθυντής πωλήσεων, ο οποίος χρειάζεται να δημιουργεί ηλεκτρονικές παρουσιάσεις και είναι εξοικειωμένος με το πρόγραμμα *Microsoft PowerPoint* (το οποίο θα πρέπει να αγοράσει) είναι πιθανόν να μην επιθυμεί να χρησιμοποιήσει το αντίστοιχο πρόγραμμα παρουσιάσεων *LibreOffice Impress*, το οποίο προσφέρεται δωρεάν.

Σε αρκετές περιπτώσεις, ιδιαίτερα με εξειδικευμένο λογισμικό (π.χ. προγράμματα σχεδίασης για αρχιτέκτονες), το κόστος του λογισμικού μπορεί να ξεπεράσει το κόστος του υλικού του υπολογιστή όπου θα εγκατασταθεί.

Ο δημιουργός ενός προγράμματος (λογισμικού) καθορίζει τις **απαιτήσεις (requirements)** του υλικού οι οποίες απαιτούνται για τη λειτουργία του. Σε αρκετές περιπτώσεις δίνονται ξεχωριστά οι ελάχιστες απαιτήσεις (minimum requirements), δηλαδή τα απολύτως απαραίτητα για τη λειτουργία του (αλλά όχι υποχρεωτικά την καλή λειτουργία του). Για το λογισμικό εφαρμογών ως μέρος των απαιτήσεων καθορίζεται και το λειτουργικό σύστημα το οποίο είναι αναγκαίο για τη λειτουργία του.

Παραδείγματα:

- Για εγκατάσταση και χρήση του λειτουργικού συστήματος *Windows 11* οι απαιτήσεις είναι: ΚΜΕ αρχιτεκτονικής x86 με ταχύτητα επεξεργασίας τουλάχιστον 1 GHz και τουλάχιστον 2 πυρήνες, μνήμη RAM με χωρητικότητα τουλάχιστον 4 GB, σκληρό δίσκο με χωρητικότητα τουλάχιστον 64 GB.
- Για εγκατάσταση και χρήση του σχεδιαστικού προγράμματος *AutoCAD 2021* οι απαιτήσεις είναι: Λειτουργικό Σύστημα Microsoft Windows 10 ή νεότερο 64-bit, ΚΜΕ αρχιτεκτονικής x64 με ταχύτητα επεξεργασίας τουλάχιστον 2,5 GHz, μνήμη RAM με χωρητικότητα τουλάχιστον 8 GB (η εισήγηση είναι 16 GB, με 8 GB να είναι οι ελάχιστες προδιαγραφές), σκληρό δίσκο με ελεύθερο χώρο τουλάχιστον 7 GB, ανάλυση οθόνης τουλάχιστον 1920 x 1080.
- Για εγκατάσταση και χρήση του παιχνιδιού *MineCraft* οι απαιτήσεις είναι: Λειτουργικό Σύστημα Microsoft Windows 10 ή νεότερο, ΚΜΕ με ταχύτητα επεξεργασίας τουλάχιστον 3.5 GHz, μνήμη RAM με χωρητικότητα τουλάχιστον 4 GB, σκληρό δίσκο με ελεύθερο χώρο τουλάχιστον 4 GB, κάρτα γραφικών με τουλάχιστον 1 GB Μνήμη RAM.

4. Επιλογή υλικού

Το υλικό του υπολογιστή θα πρέπει να ταιριάζει με τις απαιτήσεις του λογισμικού, όπως αναφέρθηκαν πιο πάνω. Επιπρόσθετα, θα πρέπει να επιλεγούν κατάλληλες περιφερειακές συσκευές ώστε να καλύπτουν τις ανάγκες του χρήστη. Παρά το γεγονός ότι οι περισσότερες περιφερειακές συσκευές σήμερα είναι φτιαγμένες (συμβατές) να λειτουργούν με διάφορα λειτουργικά συστήματα, επιβάλλεται να γίνει έλεγχος, διότι χωρίς κατάλληλο λογισμικό (οδηγό συσκευής) από τον κατασκευαστή για το συγκεκριμένο λειτουργικό σύστημα που επιλέξαμε, η περιφερειακή συσκευή πιθανόν να μην μπορεί να χρησιμοποιηθεί.

Επίσης, θα πρέπει να γίνει προσεκτικός συλλογισμός για τη μελλοντική χρήση του υπολογιστή, διότι υπάρχει πάντοτε η πιθανότητα να προστεθούν νέες ανάγκες του χρήστη μετά την αγορά του υπολογιστικού συστήματος. Ο χρόνος λειτουργίας ενός υπολογιστικού συστήματος πριν να εξεταστεί η αντικατάστασή του είναι συνήθως 6–8 χρόνια.

5. Πώς επιλέγουμε ένα υπολογιστικό σύστημα για κάποιον χρήστη

Παρακάτω θα εξετάσουμε μια συστηματική διαδικασία για επιλογή υπολογιστικού συστήματος (παρόμοια με αυτήν που υιοθετούν επαγγελματίες σύμβουλοι πληροφορικής).

5.1 Καταγραφή των αναγκών του χρήστη

Για να επιλέξουμε ένα υπολογιστικό σύστημα, αρχίζουμε με την καταγραφή των αναγκών του χρήστη. Με βάση τις ανάγκες αυτές, θα φανεί ποιο λογισμικό εφαρμογών απαιτείται, καθώς και κάποιες πληροφορίες, κυρίως για τις περιφερειακές συσκευές που χρειάζονται και για τον συνολικό αποθηκευτικό χώρο του σκληρού δίσκου που απαιτείται. Οι ανάγκες του χρήστη θα μας καθοδηγήσουν στο υλικό και το λογισμικό που απαιτείται. Παρακάτω δίνεται μια σειρά από ερωτήσεις, που απαντώντας τις μπορούμε να καταγράψουμε τις ανάγκες του χρήστη:

- **Χρειάζεται να μεταφέρεται ο υπολογιστής; Πόσο συχνά; Πόση αυτονομία (διάρκεια μπαταρίας) απαιτείται;** (Καθορίζεται εάν χρειάζεται φορητός υπολογιστής ή όχι και ποιος τύπου).
- **Πόσο όγκο και τι είδους δεδομένα χρειάζεται να αποθηκεύονται μόνιμα;** (Καθορίζει το μέγεθος του σκληρού δίσκου ή άλλης βοηθητικής μνήμης).
- **Τι είδους προγράμματα (λογισμικό εφαρμογών) απαιτούνται για τις ανάγκες του χρήστη; Υπάρχουν περιορισμοί ή προτιμήσεις;**
- **Ποια λειτουργικά συστήματα μπορούν να χρησιμοποιηθούν; Υπάρχουν περιορισμοί ή προτιμήσεις;** (Κάποιοι οργανισμοί πιθανόν να ζητούν συγκεκριμένο λειτουργικό σύστημα, για το οποίο υπάρχει εμπειρία και υποστήριξη).
- **Χρειάζεται να γίνεται εκτύπωση/σάρωση έγχρωμη/μαυρόασπρη, ποιας ποιότητας, για ποιο σκοπό, ποιας ποσότητας;**
- **Χρειάζεται συγκεκριμένη οθόνη ή άλλες περιφερειακές συσκευές; Ποιες και με ποιες προδιαγραφές;**
- **Ποιες άλλες λειτουργίες χρειάζονται;** (π.χ. εγγραφή σε CD, επικοινωνία μέσω Διαδικτύου, κ.λπ.)
- **Η κατανάλωση ρεύματος είναι σημαντική για τον χρήστη;**
- **Τι άλλο θα πρέπει να προστεθεί για την προστασία του υλικού, του λογισμικού/των πληροφοριών και του χρήστη;** (Υλικό και λογισμικό για προστασία του υπολογιστικού συστήματος, π.χ. συσκευές και λογισμικό για δημιουργία εφεδρικών αρχείων, πρόγραμμα καταπολέμησης ιών, κ.ά.)

5.2 Επιλογή λογισμικού εφαρμογών και λειτουργικού συστήματος

Με βάση τις ανάγκες του χρήστη που καταγράφηκαν προτείνεται συγκεκριμένο λογισμικό εφαρμογών. Σε περιπτώσεις όπου υπάρχουν εναλλακτικές επιλογές για κάποιο λογισμικό, μπορούν να εξεταστούν παράλληλα και να προταθούν στον χρήστη διαφορετικές εναλλακτικές επιλογές (ή να προταθεί η λύση με το χαμηλότερο κόστος).

Επιλέγοντας το λογισμικό εφαρμογών, σε πολλές περιπτώσεις καθορίζεται αυτόματα και το λειτουργικό σύστημα, από τις προδιαγραφές του λογισμικού εφαρμογών. Σε κάποιες περιπτώσεις όπου ο δημιουργός του λογισμικού εφαρμογών διαθέτει εκδοχές του για διαφορετικά λειτουργικά συστήματα (π.χ. *Mozilla Firefox, Skype*) ίσως χρειαστεί να εξεταστούν οι πιθανοί εναλλακτικοί συνδυασμοί.

Με βάση τις απαιτήσεις τόσο του λογισμικού εφαρμογών όσο και του λειτουργικού συστήματος καθορίζονται οι υπόλοιπες απαιτήσεις για το υλικό (ταχύτητα ΚΜΕ, χωρητικότητα μνήμης RAM και σκληρού δίσκου ή άλλων μέσων αποθήκευσης, κ.λπ.).

5.3 Επιλογή υλικού

Μέσα από τη διαδικασία που ακολουθούμε θα έχουν ήδη συγκεντρωθεί οι απαιτήσεις για το υλικό (ΚΜΕ Μνήμη, σκληρό δίσκο, κάρτα γραφικών, κάρτα δικτύου, περιφερειακές συσκευές, κ.λπ.).

Σε αυτό το στάδιο γίνεται αναζήτηση (από την αγορά) των τεχνικών προδιαγραφών και του κόστους εναλλακτικών επιλογών για αξιολόγηση. Αφού απορρίψουμε τυχόν επιλογές οι οποίες δεν ανταποκρίνονται στις απαιτήσεις, προχωρούμε στην αξιολόγηση και την επιλογή της καταλληλότερης για να προτείνουμε στον χρήστη.

Συχνά προκύπτει να έχουμε συσκευές των οποίων οι προδιαγραφές (τα χαρακτηριστικά τους) είναι αντίστοιχα με το κόστος αγοράς τους. Υπάρχουν όμως περιπτώσεις όπου δεν είναι ξεκάθαρο ποια είναι η καλύτερη ή η φθηνότερη λύση για τον χρήστη. Για παράδειγμα, έστω ότι έχουμε να διαλέξουμε από τους πιο κάτω δύο φορητούς υπολογιστές:

A: CPU 3.7 GHz, RAM 8 GB, HDD 500 GB, CD/DVD, οθόνη 17", τιμή €600

B: CPU 3.9 GHz, RAM 16 GB, HDD 250 GB, CD/DVD, οθόνη 15", τιμή €650

Ο **A** έχει χαμηλότερη ταχύτητα χρονιστή και λιγότερη μνήμη RAM αλλά διαθέτει σκληρό δίσκο μεγαλύτερης χωρητικότητας και μεγαλύτερη οθόνη από τον **B**. Σε τέτοιες περιπτώσεις, για να αποφασίσουμε ποιος είναι ο πιο κατάλληλος, θα πρέπει να αξιολογήσουμε «ζυγίζοντας» σε ποιο βαθμό οι διάφορες προδιαγραφές σχετίζονται με τις ανάγκες του χρήστη. Τέλος, προτείνουμε στον χρήστη με ολοκληρωμένο τρόπο το λογισμικό, το υλικό και το κόστος για το υπολογιστικό σύστημα.

5.4 Παραδείγματα

Παράδειγμα 1

Η Δήμητρα μόλις έχει τελειώσει το Λύκειο και θα αρχίσει τις σπουδές της στον κλάδο των γραφικών τεχνών. Χρειάζεται ένα υπολογιστικό σύστημα, το οποίο να μπορεί να μεταφέρει και να χρησιμοποιεί τόσο στο πανεπιστήμιο όσο και στον χώρο διαμονής της κάθε μέρα. Η οθόνη θα πρέπει να είναι υψηλής ανάλυσης και μεγέθους τουλάχιστο 17" για να μπορεί να εμφανίζει σε αυτό τις διάφορες δημιουργίες της. Επίσης, θα πρέπει να διαθέτει αρκετά μεγάλο εσωτερικό αποθηκευτικό χώρο ώστε να μπορεί να αποθηκεύει τις δημιουργίες της για την τετραετή διάρκεια των σπουδών της (περίπου 30 τον χρόνο, η καθεμιά με μέγεθος 2 GB)

παράλληλα χρειάζεται να γράφει εργασίες και να δημιουργεί παρουσιάσεις, αλλά και να επικοινωνεί με τους γονείς της μέσω Skype (ήχο και εικόνα). Δεν έχει συγκεκριμένες απαιτήσεις για το λογισμικό εφαρμογών, αλλά επιθυμεί να χρησιμοποιεί το λειτουργικό σύστημα Windows. Μπορεί να διαθέσει μέχρι 1000 ευρώ.

Ανάγκες του χρήστη:

- Χρειάζεται φορητός υπολογιστής. Δεν καθορίζεται συγκεκριμένη διάρκεια μπαταρίας.
- Όγκος δεδομένων για μόνιμη αποθήκευση: Δημιουργίες γραφικών 2 GB × 30 τον χρόνο × 4 χρόνια, σύνολο 240 GB (χρειάζεται επιπρόσθετος αποθηκευτικός χώρος για το λειτουργικό σύστημα και το λογισμικό εφαρμογών).
- Λογισμικό εφαρμογών που χρειάζεται: Επεξεργασία γραφικών, επεξεργασία κειμένου, πρόγραμμα παρουσιάσεων, Skype.
- Λειτουργικά συστήματα που μπορούν να χρησιμοποιηθούν: Windows.
- Οθόνη 17".
- Άλλες λειτουργίες: Σύνδεση με το Διαδίκτυο, μικρόφωνο, ηχεία, βιντεοκάμερα.
- Προστασία: Πρόγραμμα καταπολέμησης ιών.

Επιλογή λογισμικού εφαρμογών και λειτουργικού συστήματος:

Από τις ανάγκες προκύπτει ότι χρειάζεται:

- Λειτουργικό σύστημα: *Windows* (π.χ. *Windows 11*, το οποίο έχει απαιτήσεις ΚΜΕ αρχιτεκτονικής x86 με ταχύτητα επεξεργασίας τουλάχιστον 1 GHz με 2 πυρήνες, Μνήμη RAM με χωρητικότητα τουλάχιστον 4 GB, σκληρό δίσκο με χωρητικότητα τουλάχιστον 64 GB).
- Επεξεργαστής κειμένου/πρόγραμμα παρουσιάσεων: *Microsoft Office 365*, €5 ανά μήνα ή εναλλακτικά *LibreOffice v.7*, €0 (και τα δύο χωρίς ιδιαίτερες προδιαγραφές για το υλικό πέρα από χωρητικότητα σκληρού δίσκου περίπου 3 GB).
- Πρόγραμμα γραφικών: *Adobe CC για Windows*, €60 ανά μήνα (ανάλυση οθόνης τουλάχιστον 1024x768, χώρος στον σκληρό δίσκο 4GB, σύνδεση με το Διαδίκτυο) ή εναλλακτικά *Inkscape και Gimp*, €0 (με τις ίδιες απαιτήσεις οθόνης και σκληρού δίσκου).
- *Skype* (χωρίς ιδιαίτερες προδιαγραφές για το υλικό, πέρα από ηχεία, μικρόφωνο και βιντεοκάμερα τα οποία ήδη σημειώθηκαν).
- Πρόγραμμα καταπολέμησης ιών: *Windows Security* (δωρεάν με τα *Windows*).

Από τις επιλογές που έχουν διερευνηθεί προτείνονται όσες είναι υπογραμμισμένες. Το *LibreOffice v.7* προτιμήθηκε διότι δεν έχει κόστος αγοράς, όπως και τα *Inkscape* και *Gimp*, διότι το εναλλακτικό *Adobe CC* από μόνο του ξεπερνά το διαθέσιμο ποσό.

Επιλογή υλικού:

Μελετώντας τα πιο πάνω προκύπτει ότι ο υπολογιστής που απαιτείται πρέπει να είναι φορητός, με ΚΜΕ αρχιτεκτονικής x86 με δύο ή περισσότερους πυρήνες, με ταχύτητα επεξεργασίας τουλάχιστον 1 GHz, Μνήμη RAM με χωρητικότητα τουλάχιστον 4 GB, σκληρό δίσκο με χωρητικότητα τουλάχιστον 256 GB, οθόνη με ανάλυση 1024x768, ηχεία, μικρόφωνο και βιντεοκάμερα (παίρνουμε τις ψηλότερες απαιτήσεις από όλο το λογισμικό).

Έστω ότι αναζητήσαμε από προμηθευτές/καταστήματα πώλησης και πήραμε τις πιο κάτω εισηγήσεις (θεωρούμε ότι όλοι οι υπολογιστές είναι φορητοί και διαθέτουν CD/DVD, ηχεία, μικρόφωνο, βιντεοκάμερα, κάρτα δικτύου και εγκατεστημένο το Windows 11:

Y1: CPU 2,00 GHz (με 2 πυρήνες), RAM 4 GB, HDD 500 GB, οθόνη 17" 1024x768, τιμή €480

Y2: CPU 2,00 GHz (με 2 πυρήνες), RAM 4 GB, HDD 250 GB, οθόνη 15" 1200x800, τιμή €500

Y3: CPU 2,20 GHz (με 2 πυρήνες), RAM 6 GB, HDD 500 GB, οθόνη 17" 1200x800, τιμή €650

Y4: CPU 2,53 GHz (με 2 πυρήνες), RAM 8 GB, HDD 750 GB, οθόνη 17" 1800x1024, τιμή €800

Ο Y2 απορρίπτεται, διότι δεν έχει ικανοποιητική χωρητικότητα σκληρού δίσκου. Οι υπόλοιποι ικανοποιούν τις ανάγκες του χρήστη και με βάση την τιμή μπορούμε να επιλέξουμε τον Y1 ως την πιο οικονομική λύση

Παράδειγμα 2:

Μια εταιρεία που ασχολείται με το λιανικό εμπόριο, εισάγει είδη τροφίμων, τα οποία διαθέτει σε αλυσίδα καταστημάτων. Οι παραγγελίες από τα καταστήματα αποστέλλονται μέσω διαδικτύου (με ηλεκτρονικό ταχυδρομείο) και στα κεντρικά γραφεία της εταιρείας τέσσερα μέλη του γραμματειακού προσωπικού θα καταχωρούν τις παραγγελίες και θα εκδίδουν τιμολόγια, τα οποία θα εκτυπώνονται και θα αποστέλλονται στην αποθήκη για εκτέλεση (η ποιότητα εκτύπωσης δεν χρειάζεται να είναι πολύ καλή, αλλά η ταχύτητα θα πρέπει να είναι τουλάχιστον 20 σελίδες ανά λεπτό). Η αποθήκευση και η εκτύπωση θα γίνεται σε εξειδικευμένο πρόγραμμα αποθήκης το οποίο ήδη αγοράστηκε (και υποστηρίζεται μόνο από λειτουργικό σύστημα Linux) και θα λειτουργεί σε ξεχωριστό κεντρικό υπολογιστή, όπου θα είναι συνδεδεμένος και ο εκτυπωτής για τουλάχιστον 3 χρόνια. Κάθε μέρα στον κεντρικό υπολογιστή θα καταχωρούνται περίπου 250 MB δεδομένων από 250 παραγγελίες. Οι υπολογιστές του γραμματειακού προσωπικού θα πρέπει να διαθέτουν οθόνες μεγέθους τουλάχιστο 20", ώστε να είναι δυνατόν ταυτόχρονα να βλέπουν με ευκολία τα μηνύματα ηλεκτρονικού ταχυδρομείου και τη φόρμα με τις παραγγελίες. Σε ορισμένες περιπτώσεις το γραμματειακό προσωπικό χρειάζεται να αναζητά πληροφορίες πελατών στον παγκόσμιο ιστό και να γράφει και να εκτυπώνει (ασπρόμαυρες) επιστολές με πρόγραμμα επεξεργασίας κειμένου (το προσωπικό εκπαιδεύτηκε στη χρήση του Microsoft Word και αυτό πρέπει να εγκατασταθεί). Η κατανάλωση ρεύματος δεν είναι σε ψηλή προτεραιότητα, αφού η εταιρεία εδράζεται σε βιομηχανική περιοχή και ωφελείται από χαμηλού κόστους ρεύμα. Χρειάζεται να προταθεί η λύση η οποία καλύπτει τις ανάγκες των χρηστών με το χαμηλότερο δυνατό κόστος.

Ανάγκες του χρήστη:

Εδώ υπάρχουν δύο κατηγορίες υπολογιστικών συστημάτων, του γραμματειακού προσωπικού και ο κεντρικός υπολογιστής, οι οποίες θα πρέπει να εξεταστούν ξεχωριστά.

Για το γραμματειακό προσωπικό:

- Χρειάζεται επιτραπέζιος υπολογιστής.
- Λογισμικό εφαρμογών που χρειάζεται: Επεξεργαστής κειμένου, φυλλομετρητής ιστού, πρόγραμμα ηλεκτρονικού ταχυδρομείου.
- Λειτουργικά συστήματα που μπορούν να χρησιμοποιηθούν: *Windows*.
- Οθόνη 20".
- Άλλες λειτουργίες: Σύνδεση με το Διαδίκτυο.
- Προστασία: Πρόγραμμα καταπολέμησης ιών.

Για τον κεντρικό υπολογιστή:

- Χρειάζεται επιτραπέζιος υπολογιστής.

- Όγκος δεδομένων για μόνιμη αποθήκευση: Δημιουργίες γραφικών 250 MB × 365 μέρες × 3 χρόνια, σύνολο 273750 MB, περίπου 268 GB (χρειάζεται επιπρόσθετος αποθηκευτικός χώρος για το λειτουργικό σύστημα και το λογισμικό εφαρμογών).
- Λογισμικό εφαρμογών που χρειάζεται: Εξειδικευμένο πρόγραμμα αποθήκης.
- Λειτουργικά συστήματα που μπορούν να χρησιμοποιηθούν: *Linux*.
- Εκτυπωτής με ταχύτητα τουλάχιστον 20 ppm.
- Άλλες λειτουργίες: Σύνδεση με δίκτυο.
- Προστασία: Προστασία από διακοπή του ηλεκτρικού ρεύματος (UPS), DVD Drive για εφεδρικά αρχεία ασφάλειας.

Επιλογή λογισμικού εφαρμογών και λειτουργικού συστήματος:

Από τις ανάγκες προκύπτει ότι χρειάζεται:

Για το γραμματειακό προσωπικό:

- Λειτουργικό σύστημα: Windows (π.χ. *Windows 11*, το οποίο έχει απαιτήσεις ΚΜΕ αρχιτεκτονικής x86 με ταχύτητα επεξεργασίας τουλάχιστον 1 GHz με 2 πυρήνες, Μνήμη RAM με χωρητικότητα τουλάχιστον 4 GB, σκληρό δίσκο με χωρητικότητα τουλάχιστον 64 GB).
- Επεξεργαστής κειμένου/πρόγραμμα παρουσιάσεων: *Microsoft Office 365*, €5 ανά μήνα ή εναλλακτικά *LibreOffice v.7*, €0 (και τα δύο χωρίς ιδιαίτερες προδιαγραφές για το υλικό πέρα από χωρητικότητα σκληρού δίσκου περίπου 3 GB).
- Φυλλομετρητής ιστού: Edge ή Firefox ή Opera ή Chrome ή Safari (όλοι χωρίς κόστος ή ιδιαίτερες απαιτήσεις).
- Πρόγραμμα καταπολέμησης ιών: Kaspersky Total Security (€60).

Για τον κεντρικό υπολογιστή:

- Λειτουργικό σύστημα: Linux (π.χ. *Ubuntu Server 14LTS*, το οποίο έχει απαιτήσεις ΚΜΕ αρχιτεκτονικής x86 με ταχύτητα επεξεργασίας τουλάχιστον 1 GHz, μνήμη RAM με χωρητικότητα τουλάχιστον 512 MB, σκληρό δίσκο με χωρητικότητα τουλάχιστον 5 GB).
- Εξειδικευμένο πρόγραμμα αποθήκης (ήδη αγοράστηκε, δεν έχει συγκεκριμένες απαιτήσεις).

Από τις επιλογές που έχουν διερευνηθεί προτείνονται όσες είναι υπογραμμισμένες. Ως πρόγραμμα καταπολέμησης ιών θα μπορούσε να χρησιμοποιηθεί το δωρεάν Microsoft Security Essentials, αλλά προτείνεται το Kaspersky, το οποίο παρέχει μεγαλύτερη ασφάλεια.

Επιλογή υλικού:

Μελετώντας τα πιο πάνω προκύπτει ότι ο υπολογιστής που απαιτείται για το γραμματειακό προσωπικό πρέπει να είναι επιτραπέζιος, με ΚΜΕ αρχιτεκτονικής x86 με 2 πυρήνες, με ταχύτητα επεξεργασίας τουλάχιστον 1 GHz, μνήμη RAM με χωρητικότητα τουλάχιστον 4 GB, σκληρό δίσκο με χωρητικότητα τουλάχιστον 64 GB και οθόνη με μέγεθος 20".

Ο κεντρικός υπολογιστής πρέπει να είναι επιτραπέζιος, με ΚΜΕ αρχιτεκτονικής x86, με ταχύτητα επεξεργασίας τουλάχιστον 1 GHz, μνήμη RAM με χωρητικότητα τουλάχιστον 512 GB, σκληρό δίσκο με χωρητικότητα τουλάχιστον 273 GB, εφοδιασμένο με DVD Drive, με εκτυπωτή ταχύτητας 20ppm και με UPS.

Έστω ότι αναζητήσαμε από προμηθευτές/καταστήματα πώλησης και πήραμε τις πιο κάτω εισηγήσεις (θεωρούμε ότι όλοι οι υπολογιστές είναι επιτραπέζιοι και διαθέτουν CD/DVD και κάρτα δικτύου):

HY1: CPU 1,60 GHz (με 2 πυρήνες), RAM 4 GB, HDD 250 GB, οθόνη 17", τιμή €370.

HY2: CPU 1,60 GHz (με 2 πυρήνες), RAM 4 GB, HDD 250 GB, οθόνη 20", τιμή €420.

HY3: CPU 2,20 GHz (με 2 πυρήνες), RAM 4 GB, HDD 250 GB, οθόνη 20", τιμή €550.

HY4: CPU 2,20 GHz (με 2 πυρήνες), RAM 4 GB, HDD 500 GB, οθόνη 17", τιμή €500.

HY5: CPU 2,50 GHz (με 2 πυρήνες), RAM 8 GB, HDD 1 TB, οθόνη 20", τιμή €690

Για το γραμματειακό προσωπικό, ο HY1 και ο HY4 απορρίπτονται, διότι δεν έχουν αρκετά μεγάλη οθόνη. Επειδή στόχος είναι η ικανοποίηση των αναγκών με το χαμηλότερο κόστος, δεν χρειάζεται λεπτομερής αξιολόγηση και επιλέγουμε τον HY2.

Για τον κεντρικό υπολογιστή, οι HY1, HY2 και HY3 απορρίπτονται, διότι δεν έχουν αρκετή χωρητικότητα σκληρού δίσκου. Επομένως επιλέγουμε τον HY4, ο οποίος έχει χαμηλότερο κόστος.

Στην περίπτωση αυτή θα χρειαστούμε και έναν εκτυπωτή. Έστω ότι πήραμε από τους προμηθευτές τις πιο κάτω προδιαγραφές εκτυπωτών:

E1: Ψεκασμού μελάνης, έγχρωμος, 1200dpi, 20ppm, € 0,142 ανά σελίδα, τιμή €80.

E2: Λείζερ, ασπρόμαυρος, 300dpi, 22ppm, € 0,022 ανά σελίδα, τιμή €200.

E3: Λείζερ, ασπρόμαυρος, 600dpi, 24ppm, € 0,024 ανά σελίδα, τιμή €350.

Ο E1 φαίνεται φθηνότερος, αλλά εάν υπολογίσουμε το κόστος για εκτύπωση 250 σελίδων παραγγελιών για κάθε μέρα (περίπου 273750 σελίδες σε 3 χρόνια), τότε το κόστος του E1 είναι € 38953, πολύ ψηλότερο από τον E2 με € 6223 και τον E3 με € 6920. Επομένως επιλέγουμε τον E2 ως τη φθηνότερη επιλογή.

Παρόμοια διαδικασία κανονικά πρέπει να ακολουθηθεί και για το UPS (στοιχίζει περίπου € 60–100).

Βασικές Έννοιες

| | |
|--------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Υπολογιστικές ανάγκες του χρήστη: | Λεπτομερής περιγραφή των αναγκών που έχει ο χρήστης (φορητότητα/αυτονομία, όγκος δεδομένων για μόνιμη αποθήκευση, λογισμικό εφαρμογών, λειτουργικό σύστημα, περιφερειακές συσκευές και προδιαγραφές τους, κατανάλωση ρεύματος, προστασία). Χρησιμοποιούνται για να καθοριστούν οι απαιτήσεις του λογισμικού και οι προδιαγραφές του υλικού. |
| Απαιτήσεις λογισμικού εφαρμογών: | Οι ελάχιστες προδιαγραφές του υλικού και του λειτουργικού συστήματος που απαιτούνται για την ομαλή λειτουργία του λογισμικού εφαρμογών. |
| Απαιτήσεις λειτουργικού συστήματος: | Οι ελάχιστες προδιαγραφές του υλικού που απαιτούνται για την ομαλή λειτουργία του λογισμικού εφαρμογών. |
| Προδιαγραφές υλικού: | Τα χαρακτηριστικά του υλικού που απαιτούνται ώστε να καλύπτονται οι απαιτήσεις του λογισμικού και κατά συνέπεια οι ανάγκες του χρήστη. |
| Επίδοση ανά κόστος: | Δείκτης που υπολογίζεται διαιρώντας τον Δείκτη Επίδοσης με το κόστος της κάθε επιλογής. Μας βοηθά να συγκρίνουμε πόσο κοστίζει η επίδοση. Για συσκευές με αναλώσιμα, πρέπει να προστεθεί το κόστος λειτουργίας και η τιμή, για να υπολογιστεί το κόστος. |

Πηγές

1. Προδιαγραφές Λειτουργικού συστήματος Windows 7. Ανακτήθηκε στις 04.07.2013.
<http://windows.microsoft.com/en-us/windows7/products/system-requirements>
2. Προδιαγραφές σχεδιαστικού προγράμματος AutoCAD 2013. Ανακτήθηκε στις 04.07.2013.
<http://docs.autodesk.com/ACD/2013/ENU/System%20Requirements/index.html>
3. Προδιαγραφές παιχνιδιού Η/Υ SimCity. Ανακτήθηκε στις 04.07.2013.
<https://help.ea.com/article/simcity-system-requirements>

ΕΝΟΤΗΤΑ Γ3 Λειτουργικά Συστήματα

Για την προετοιμασία και συγγραφή του υλικού αυτής της ενότητας εργάστηκαν οι ακόλουθοι/θες Καθηγητές/τριες Πληροφορικής και Επιστήμης Ηλεκτρονικών Υπολογιστών που παρακολούθησαν το πρόγραμμα επιμόρφωσης μάχιμων εκπαιδευτικών για το Νέο Αναλυτικό Πρόγραμμα:

Ζάγγουλος Νικόλαος (Σύμβουλος)

Μάκκουλα Πόλα

Μαραθεύτη Χριστιάνα

Μυλωνάς Σωκράτης (Σύμβουλος)

Νικόλαου Λοΐζος

Ξενοφώντος Ξένιος

Χρυσοστόμου Χρυσόστομος

Γ3.1 Λειτουργικά Συστήματα – Λειτουργίες και Παραδείγματα

Τι θα μάθουμε σήμερα:

- ◊ Να αναγνωρίζουμε τον ρόλο του Λειτουργικού Συστήματος ως συντονιστή του υλικού και του λογισμικού του υπολογιστή
- ◊ Να αναφέρουμε τις βασικές συντονιστικές λειτουργίες του Λειτουργικού Συστήματος (έλεγχος υλικού, διαχείριση πόρων – ΚΜΕ/κύριας μνήμης/συστήματος αρχειοθέτησης/εκτυπωτών/δικτύου, διαχείριση πρόσβασης και χρηστών)
- ◊ Να περιγράψουμε τη σχέση του Λειτουργικού Συστήματος και του Λογισμικού Εφαρμογών
- ◊ Να ονομάζουμε τα βασικότερα Λειτουργικά Συστήματα που χρησιμοποιούνται σήμερα (Microsoft Windows, Linux, Mac OS X, Android, Symbian, κ.λπ.).

1. Λειτουργικό Σύστημα

Το λειτουργικό σύστημα είναι ένα πρόγραμμα το οποίο ελέγχει την εκτέλεση των υπολοίπων προγραμμάτων, διαχειρίζεται τους πόρους του υπολογιστή και παρέχει στα προγράμματα τη δυνατότητα να τρέξουν. Λειτουργεί, δηλαδή, ως ενδιάμεσος μεταξύ του υλικού, των εφαρμογών και του χρήστη. Σχεδόν όλοι οι υπολογιστές, παλάμης, επιτραπέζιοι, υπερυπολογιστές, ακόμη και τα έξυπνα κινητά τηλέφωνα, χρησιμοποιούν κάποιο τύπο λειτουργικού συστήματος.



Εικόνα 42 Λειτουργικό Σύστημα, ενδιάμεσος υλικού και λογισμικού

2. Βασικές Λειτουργίες Λειτουργικών Συστημάτων

Το λειτουργικό σύστημα ελέγχει τη λειτουργία του υπολογιστή συνολικά και χρησιμοποιείται ως υπόβαθρο για την εκτέλεση όλων των υπολοίπων προγραμμάτων, τη διαχείριση των περιφερειακών συσκευών και την εξασφάλιση της επικοινωνίας με τον χρήστη. Παρακάτω παρουσιάζονται οι βασικές λειτουργίες των λειτουργικών συστημάτων:

2.1 Ελέγχει το Υλικό του Συστήματος:

Μια από τις κεντρικές αρμοδιότητες του λειτουργικού συστήματος είναι ο έλεγχος του υλικού. Μέσω του λειτουργικού συστήματος, εξασφαλίζεται η αρμονική συνεργασία και επικοινωνία μεταξύ όλων των συσκευών που απαρτίζουν το υπολογιστικό σύστημα. Για παράδειγμα, εάν χρησιμοποιούμε μια εφαρμογή επεξεργασίας κειμένου και θέλουμε να εκτυπώσουμε μία επιστολή αλλά τυγχάνει ο εκτυπωτής μας να είναι αποσυνδεδεμένος, το λειτουργικό σύστημα θα μας δώσει αυτή την πληροφορία. Επίσης, εάν θέλουμε να συνδέσουμε στο υπολογιστικό μας σύστημα μια νέα περιφερειακή συσκευή, τότε το λειτουργικό σύστημα χρησιμοποιώντας τον κατάλληλο οδηγό συσκευής θα μας δώσει την ευκαιρία να ανταλλάξουμε δεδομένα με τη συγκεκριμένη συσκευή.

2.2 Διαχειρίζεται την ΚΜΕ:

Το λειτουργικό σύστημα αναθέτει στη ΚΜΕ τις διάφορες διεργασίες που πρέπει να εκτελεστούν, ανάλογα με τις απαιτήσεις του προγράμματος που χρησιμοποιείται τη δεδομένη στιγμή.

2.3 Διαχειρίζεται τη Μνήμη:

Το λειτουργικό σύστημα συντονίζει τη χρήση των διάφορων τύπων μνήμης (καταχωρητές, κύρια μνήμη και βοηθητική μνήμη), καταγράφοντας ποια τμήματά τους είναι διαθέσιμα, ποια

είναι δεσμευμένα και, αναλόγως με τις απαιτήσεις των διεργασιών, εκχωρεί ή απελευθερώνει τμήματα μνήμης.

2.4 Διαχειρίζεται το Σύστημα Αρχαιοθέτησης:

Το μέρος εκείνο που γίνεται περισσότερο αντιληπτό από τον χρήστη είναι εκείνο που διαχειρίζεται το σύστημα αρχείων. Το λειτουργικό σύστημα εξασφαλίζει έναν ομοιόμορφο τρόπο για να αντιλαμβανόμαστε την αποθήκευση της πληροφορίας, με τη χρήση της έννοιας αρχείο και φάκελος. Το λειτουργικό σύστημα μάς δίνει τη δυνατότητα να εκτελούμε μια σειρά από ενέργειες σε κάθε αρχείο όπως είναι η δημιουργία, η διαγραφή, η μετονομασία, η μετακίνηση, η αντιγραφή αρχείου και η τροποποίηση αρχείου.

2.5 Διαχειρίζεται το Δίκτυο:

Τα περισσότερα σύγχρονα λειτουργικά συστήματα επιτρέπουν τη σύνδεση ενός υπολογιστή, τόσο σε τοπικά δίκτυα όσο και στο Διαδίκτυο, ενσωματώνοντας στον κώδικά τους την υλοποίηση των απαιτούμενων αντίστοιχων πρωτοκόλλων.

2.6 Διαχειρίζεται την Πρόσβαση των Χρηστών:

Το λειτουργικό σύστημα διαχειρίζεται το επίπεδο πρόσβασης του κάθε χρήστη σε ένα υπολογιστή. Με τη δημιουργία προφίλ χρήστη μπορούμε να καθορίσουμε τη δική μας συλλογή ρυθμίσεων. Τα προφίλ χρηστών, κάθε φορά που θα ενεργοποιούμε ένα λειτουργικό σύστημα, διασφαλίζουν ότι θα χρησιμοποιούνται οι προσωπικές μας προτιμήσεις και ταυτόχρονα, εάν το επιθυμούμε, άλλοι χρήστες δεν θα έχουν πρόσβαση στα δικά μας αρχεία.

3. Σχέση Λειτουργικού Συστήματος και Λογισμικού Εφαρμογών

Το λειτουργικό σύστημα είναι το κύριο μέρος του λογισμικού συστήματος, το οποίο ελέγχει και συντονίζει το υλικό. Όταν κάποιο πρόγραμμα, το οποίο ανήκει στην κατηγορία του λογισμικού εφαρμογών, χρειάζεται να χρησιμοποιήσει κάποια συσκευή, απευθύνεται στο λειτουργικό σύστημα που αναλαμβάνει την επικοινωνία με τη συσκευή εκ μέρους του. Για παράδειγμα, εάν χρησιμοποιούμε μια εφαρμογή επεξεργασίας κειμένου και επιθυμούμε να αποθηκεύσουμε ένα έγγραφο στον σκληρό δίσκο, το λειτουργικό σύστημα θα μεσολαβήσει έτσι ώστε να παραλάβει τα δεδομένα από την εφαρμογή και να επικοινωνήσει με τον σκληρό δίσκο, δίνοντάς του οδηγίες για το πού και με ποια σειρά θα γίνει η αποθήκευση.

Η κάθε εφαρμογή είναι φτιαγμένη για να συνεργάζεται με ένα συγκεκριμένο λειτουργικό σύστημα. Κατά συνέπεια, μια εφαρμογή η οποία λειτουργεί σε έναν υπολογιστή ο οποίος διαθέτει ένα συγκεκριμένο λειτουργικό σύστημα, δεν μπορεί να λειτουργήσει σε έναν άλλο υπολογιστή ο οποίος διαθέτει κάποιο άλλο λειτουργικό σύστημα. Επειδή οι κατασκευαστές λογισμικού γνωρίζουν αυτόν τον περιορισμό, συνήθως ετοιμάζουν διαφορετικές εκδόσεις μιας εφαρμογής για διαφορετικά λειτουργικά συστήματα, ώστε εμείς να επιλέγουμε αυτήν που ταιριάζει με το λειτουργικό σύστημα που διαθέτει ο υπολογιστής μας.

4. Τα Βασικότερα Λειτουργικά Συστήματα

Είναι γεγονός, ότι τις περισσότερες φορές, αγοράζοντας κάποιος έναν υπολογιστή, κάποιο λειτουργικό σύστημα θα είναι ήδη εγκατεστημένο σε αυτόν. Είναι πολύ σημαντικό να γνωρίζουμε όμως, ότι ανάλογα με την κατηγορία στην οποία ανήκει ο υπολογιστής μας και λαμβάνοντας υπόψη τις δικές μας προτεραιότητες και ανάγκες, μπορούμε να επιλέξουμε το λειτουργικό σύστημα που θα διαθέτει.

4.1 Λειτουργικά Συστήματα για Προσωπικούς Υπολογιστές

4.1.1 Windows

Η σειρά εμφανίστηκε πρώτα από τη Microsoft το 1985 για να ανταγωνιστεί το λειτουργικό σύστημα της Apple, το οποίο χρησιμοποιούσε γραφικό περιβάλλον. Τα Microsoft Windows, με τις διαφορετικές εκδόσεις τους, οι οποίες ανήκουν στην κατηγορία του λογισμικού κλειστού κώδικα, επικράτησαν στην παγκόσμια αγορά προσωπικών υπολογιστών. Συνήθως, χρήστες οι οποίοι ενδιαφέρονται να μπορούν να χρησιμοποιούν στον υπολογιστή τους μία διευρυμένη γκάμα εφαρμογών, θα εγκαταστήσουν στον υπολογιστή τους κάποια έκδοση των Windows. Η τελευταία έκδοση των Windows που κυκλοφόρησε στην αγορά είναι η σειρά των Windows 11.



Εικόνα 43 Windows 11

4.1.2 Linux

Η αρχιτεκτονική του Linux είναι βασισμένη στις αρχές του λειτουργικού συστήματος Unix και η ανάπτυξη του είναι χαρακτηριστικό παράδειγμα εθελοντικής συνεργασίας από



Εικόνα 44 Διανομές Linux

διαδικτυακές κοινότητες, ενώ όλο το έργο είναι ανοικτού κώδικα και ελεύθερα προσβάσιμο από όλους για αντιγραφή, τροποποίηση ή αναδιανομή χωρίς περιορισμό. Το Linux κυκλοφορεί σε διαφορετικές διανομές (distributions) Linux, λόγω της μεγάλης δυνατότητας παραμετροποίησης που προσφέρει. Η μεγάλη δυνατότητα παραμετροποίησης σε συνδυασμό με το γεγονός ότι το Linux προσφέρεται δωρεάν, το καθιστά πρώτη επιλογή για άτομα τα οποία ασχολούνται με τον πειραματισμό και τον προγραμματισμό, με στόχο την διεύρυνση των δυνατοτήτων του λειτουργικού συστήματος. Σχεδόν όλες οι διανομές περιλαμβάνουν το πρόγραμμα περιήγησης Mozilla Firefox και τη σουίτα εφαρμογών γραφείου LibreOffice. Δύο από τις πιο διαδεδομένες διανομές είναι το Linux Ubuntu, το Linux Fedora και το srolinux το οποίο βασίζεται στο Ubuntu Lucid και περιέχει όλο το εκπαιδευτικό λογισμικό του Παιδαγωγικού Ινστιτούτου, μαζί με τα βιβλία της Α΄ Γυμνασίου.

4.1.3 Mac OS X

Το Mac OS X είναι μια σειρά γραφικών λειτουργικών συστημάτων που αναπτύσσεται, προωθείται και πωλείται από την Apple Inc. και συμπεριλαμβάνεται σε κάθε καινούριο υπολογιστή Macintosh (Mac). Το Mac OS X είναι η εξέλιξη του αρχικού Mac OS το οποίο ήταν το αρχικό λειτουργικό σύστημα της Apple την περίοδο 1984-1999. Έκτοτε, το Mac OS X έχει δεχτεί τέσσερις αναβαθμίσεις από τις οποίες οι πιο πρόσφατη είναι η Mac OS X v10.7 "Lion" (Λιοντάρι), που κυκλοφόρησε στις 20 Ιουλίου 2011. Είναι γνωστό ότι η Apple, στον τομέα των πολυμεσικών εφαρμογών, διαθέτει ένα προβάδισμα σε σχέση με τους περισσότερους ανταγωνιστές της. Γι' αυτό τον λόγο, χρήστες οι οποίοι έχουν ψηλά στις προτεραιότητές τους τη δυνατότητα χρήσης απαιτητικών σχεδιαστικών εφαρμογών, αναμένεται ότι θα επιλέξουν υπολογιστή Mac και κατά συνέπεια το λειτουργικό Mac OS X.



Εικόνα 45 Mac OS X Lion

4.2 Λειτουργικά Συστήματα για Μεγάλους Υπολογιστές

Οι Μεγάλοι υπολογιστές είναι φτιαγμένοι για να εξυπηρετούν ταυτόχρονα εκατοντάδες χρήστες που συνδέονται μαζί τους. Οι χρήστες, χρησιμοποιούν το πληκτρολόγιο και την οθόνη του προσωπικού υπολογιστή τους για είσοδο δεδομένων και έξοδο πληροφοριών, ενώ η επεξεργασία και η αποθήκευση γίνεται στον Μεγάλο Υπολογιστή. Τα λειτουργικά συστήματα που χρησιμοποιούνται στους μεγάλους υπολογιστές, θα πρέπει να μπορούν να υποστηρίξουν αυτή την απαίτηση. Επιπρόσθετα, πρέπει να μπορούν να διαχειρίζονται τον αυξημένο όγκο δεδομένων και πληροφοριών αποτελεσματικά. Κάποια από τα περισσότερο διαδεδομένα λειτουργικά συστήματα για Μεγάλους υπολογιστές είναι τα εξής: UNIX, Aix και Solaris.



Εικόνα 46
Λειτουργικά Συστήματα
Μεγάλων Υπολογιστών

4.3 Λειτουργικά Συστήματα για Φορητές Συσκευές

Τα τελευταία χρόνια διαπιστώσαμε έναν πραγματικό κατακλυσμό που σχετίζεται με τα λειτουργικά συστήματα για έξυπνα κινητά τηλέφωνα και γενικότερα για τις μικρές-φορητές συσκευές όπως είναι οι υπολογιστές τύπου ταμπλέτας (tablet computers). Επειδή οι υπολογιστικές προδιαγραφές αυτών των συσκευών είναι χαμηλές σε σχέση με τις προδιαγραφές των επιτραπέζιων υπολογιστών, ήταν αναγκαίο να αναπτυχθούν και τα ανάλογα λειτουργικά συστήματα. Παραδείγματα λειτουργικών συστημάτων που χρησιμοποιούνται σε φορητές συσκευές είναι τα εξής: Android, iOS, BlackBerry OS6, Windows Phone 7 και Symbian.

4.3.1 Android

Το λειτουργικό Android αναπτύχθηκε από την Google και σήμερα θεωρείται σε παγκόσμιο επίπεδο το ίδιο δημοφιλές με το iOS της Apple. Είναι η πιο γρήγορα αναπτυσσόμενη σήμερα πλατφόρμα και χρησιμοποιείται από πολλούς κατασκευαστές φορητών συσκευών.

Εικόνα 47
Λειτουργικό
Σύστημα
Android



4.3.2 iOS

Το iOS είναι διαθέσιμο μόνο στις συσκευές της Apple και σήμερα είναι ίσως το πιο γνωστό από όλες τις φορητές συσκευές. Υπάρχουν εκατοντάδες χιλιάδες εφαρμογές διαθέσιμες στο App Store της Apple.

Εικόνα 48
Λειτουργικό
Σύστημα
iOS 4



4.3.3 BlackBerry

Το λειτουργικό σύστημα BlackBerry (διαθέσιμο μόνο σε συσκευές BlackBerry®) είναι διάσημο κυρίως για την παροχή υψηλής ποιότητας υπηρεσιών email και instant messaging μεταξύ BlackBerry συσκευών. Σήμερα έχει περιορισμένη χρήση, αφού έχουν επικρατήσει το Android και το iOS.

Εικόνα 49
Λειτουργικό
Σύστημα
BlackBerry
OS6



4.3.4 Windows Phone

Το λειτουργικό σύστημα για κινητά τηλέφωνα από την Microsoft, Windows Phone δίνει έμφαση στον απλούστερο δυνατό χειρισμό της συσκευής, στη σύνδεση με ιστοτόπους κοινωνικού περιεχομένου καθώς και στην επεξεργασία

Εικόνα 50
Λειτουργικό
Σύστημα
Windows
Phone 7



δεδομένων με γνωστές εφαρμογές της Microsoft όπως το Microsoft Office.

4.3.5 Symbian

Το Symbian ήταν ένα από τα πιο διαδεδομένα λειτουργικά συστήματα για κινητά τηλέφωνα. Πολλά τηλέφωνα Nokia, αλλά και τηλέφωνα άλλων κατασκευαστών είχαν το λειτουργικό αυτό σύστημα. Τα τελευταία χρόνια, όμως, έχουν επικρατήσει άλλα λειτουργικά συστήματα.

Εικόνα 51
Λειτουργικό
Σύστημα
Symbian

symbian
OS

Βασικές Έννοιες

Λειτουργικό Σύστημα:

Είναι λογισμικό (συλλογή από προγράμματα) το οποίο ελέγχει την εκτέλεση των υπολοίπων προγραμμάτων, διαχειρίζεται τους πόρους του υπολογιστή και παρέχει στα προγράμματα την δυνατότητα να τρέξουν (να εκτελεστούν).

Βασικές Λειτουργίες

Λειτουργικών Συστημάτων:

- Ελέγχει το Υλικό του Συστήματος
- Διαχειρίζεται την ΚΜΕ
- Διαχειρίζεται τη Μνήμη
- Διαχειρίζεται το Σύστημα Αρχαιοθέτησης
- Διαχειρίζεται το Δίκτυο
- Διαχειρίζεται την Πρόσβαση των Χρηστών

Παραδείγματα Λειτουργικών Συστημάτων για

Προσωπικούς Υπολογιστές:

- Windows
- Linux
- Mac OS X

Παραδείγματα Λειτουργικών Συστημάτων για Μεγάλους Υπολογιστές:

- UNIX
- Aix
- Solaris

Παραδείγματα Λειτουργικών Συστημάτων για Φορητές Συσκευές:

- Android
- iOS
- BlackBerry
- Windows Phone
- Symbian

ΕΝΟΤΗΤΑ Γ6 Βάσεις Δεδομένων και Ανάπτυξη Πληροφοριακών Συστημάτων

Για την προετοιμασία και τη συγγραφή του υλικού αυτής της ενότητας εργάστηκαν οι ακόλουθοι/θες Καθηγητές/τριες Πληροφορικής και Επιστήμης Ηλεκτρονικών Υπολογιστών που παρακολούθησαν το πρόγραμμα επιμόρφωσης μάχιμων εκπαιδευτικών για το Νέο Αναλυτικό Πρόγραμμα:

Αντωνίου Κωνσταντίνος
Αχιλλέως Χριστάκης
Ευθυμιάδου Έλενα
Ζεβεδαίος Λίνος
Θρασυβούλου Έλενα
Ιωάννου Ιωάννης (Σύμβουλος)
Καλλένος Μιχάλης
Καμιναρίδου Χριστίνα
Κλόκκος Πόλυς
Κόκκινος Κλεάνθης
Κουτούφαρη Αργυρώ
Κυπριανού Χρίστος
Κωνσταντίνου Μελίνα
Μάκκουλα Πόλα
Μιχαήλ Παναγιώτης
Μουμτζή Μαριάννα
Νικολάου Νεόφυτος
Ξενοφώντος Γιώργος

Ξενοφώντος Ξένιος (Σύμβουλος)
Παναγή Παύλος
Παναγίδου-Βέρελ Ιφιγένεια
Παπαχριστοδούλου Χρυστάλλα
Παυλίδου Αλεξία
Παύλου Γεωργία
Πρωτοπαπά Μαρία
Σιακαλλής Χαρίλαος
Τζιώρτας Χρήστος
Φυλακτού Δημήτρης
Φωτάκος Θωμάς
Χατζηγιάννη Αγγέλα
Χαριλάου Μιλτιάδης (Σύμβουλος)
Χατζηκωνσταντή Ανδριανή
Χατζηπαντελής Δημήτρης
Χατζηπολυκάρπου Ιωάννης
Χρυσοστόμου Χρύσης

Γ6.1 Εισαγωγή στις Βάσεις Δεδομένων

Τι θα μάθουμε σήμερα:

- Να ορίζουμε τι είναι μια βάση δεδομένων
- Να διακρίνουμε τα είδη των βάσεων δεδομένων (έντυπη/ηλεκτρονική)
- Να διακρίνουμε και να αναφέρουμε τα πλεονεκτήματα και τα μειονεκτήματα μιας βάσης δεδομένων
- Να αναφέρουμε βάσεις δεδομένων από την καθημερινή ζωή
- Να αναφέρουμε παραδείγματα προγραμμάτων διαχείρισης μηχανογραφημένων βάσεων δεδομένων.

1. Βάση Δεδομένων (Database)

Μια βάση δεδομένων είναι ένα σύνολο από πληροφορίες που σχετίζονται μεταξύ τους και οι οποίες έχουν οργανωθεί με τέτοιο τρόπο ώστε η χρήση τους να είναι γρήγορη και αποτελεσματική.

Καθημερινά, σε πολλές από τις δραστηριότητές μας χρειαζόμαστε πληροφορίες. Γι' αυτό τον λόγο οι βάσεις δεδομένων βρίσκονται παντού: Στο σπίτι, στην εργασία, στα σχολεία, στα καταστήματα, στις κυβερνητικές υπηρεσίες κ.λπ.

Μια βάση δεδομένων μπορεί να υπάρχει σε:

- **έντυπη μορφή** (π.χ. ο τηλεφωνικός κατάλογος) και σε
- **ηλεκτρονική μορφή** (π.χ. οι επαφές μας στο κινητό μας τηλέφωνο).

Στην ενότητα αυτή θα μελετήσουμε τις ηλεκτρονικές ή διαφορετικά τις μηχανογραφημένες βάσεις δεδομένων.

2. Ηλεκτρονικές (μηχανογραφημένες) βάσεις δεδομένων

Η δημιουργία και διαχείριση μηχανογραφημένων βάσεων δεδομένων ήταν ένας από τους κύριους λόγους της χρήσης υπολογιστών από μεγάλους οργανισμούς, από τη δεκαετία του 1960. Αρχικά είχαν προταθεί διάφοροι τρόποι οργάνωσης και διαχείρισης των δεδομένων, βασισμένοι σε διαφορετικά θεωρητικά «μοντέλα».

Από τη δεκαετία του 1970 και μετά επικράτησε η άποψη ότι ο πιο αποτελεσματικός τρόπος οργάνωσης και διαχείρισης των μηχανογραφημένων βάσεων δεδομένων είναι μέσα από ειδικό λογισμικό το οποίο ονομάζεται Σύστημα Διαχείρισης Βάσεων Δεδομένων (ΣΔΒΔ) - **D**atabase **M**anagement **S**ystem (**DBMS**). Το λογισμικό αυτό είναι κατασκευασμένο έτσι ώστε να μπορεί να διαχειρίζεται διαφορετικές βάσεις δεδομένων οι οποίες έχουν σχεδιαστεί και οργανωθεί ακολουθώντας το θεωρητικό «μοντέλο» που προτείνει το ΣΔΒΔ.

Κάποια από τα πιο δημοφιλή συστήματα διαχείρισης μηχανογραφημένων βάσεων δεδομένων είναι και τα ακόλουθα: MySQL, PostgreSQL, SQLite, Microsoft SQL Server, **Microsoft Access**, Oracle, SAP, dBASE, FoxPro, IBM DB2 και FilemakerPro.

Τα επιθυμητά χαρακτηριστικά ενός Συστήματος Διαχείρισης Βάσης Δεδομένων (ΣΔΒΔ) είναι τα ακόλουθα:

- μείωση πλεοναζόντων δεδομένων
- ευκολία στην ανάπτυξη νέων εφαρμογών
- μηχανισμοί ασφαλείας.

Τα σύγχρονα (ΣΔΒΔ) έχουν **εργαλεία** για:

- δημιουργία πινάκων
- εισαγωγή και αλλαγή δεδομένων στους πίνακες μέσα από φόρμες
- άντληση πληροφοριών με τη δημιουργία ερωτημάτων
- εκτύπωση πληροφοριών με τη δημιουργία εκθέσεων

Την ίδια εποχή εξετάστηκαν διάφοροι τρόποι σχεδιασμού και οργάνωσης των δεδομένων και επικράτησε το μοντέλο της **Σχισιακής Βάσης Δεδομένων (Relational Database)**, το οποίο προτείνει ότι μια βάση δεδομένων είναι οργανωμένη σε πίνακες οι οποίοι σχετίζονται μεταξύ τους με βάση κάποια από τα δεδομένα που περιέχουν.

Έτσι, μέχρι σήμερα έχουν επικρατήσει τα Συστήματα Διαχείρισης Σχισιακών Βάσεων Δεδομένων (Relational Database Management System, RDBMS), όπως είναι η *Microsoft Access*.

3. Πλεονεκτήματα των μηχανογραφημένων βάσεων δεδομένων

(1) **Οι πληροφορίες καταχωρούνται και εντοπίζονται πολύ γρήγορα:** Σε μια μηχανογραφημένη βάση δεδομένων οι χρήστες μπορούν να καταχωρούν και να εντοπίζουν πολύ πιο γρήγορα και αποτελεσματικά τις πληροφορίες που θέλουν, ανεξάρτητα από τον τρόπο και τη σειρά καταχώρισης. Για παράδειγμα, μπορούμε να βρούμε πολύ πιο γρήγορα τον αριθμό τηλεφώνου ενός φίλου μας εάν ψάξουμε στις επαφές του κινητού μας παρά στον τηλεφωνικό κατάλογο.

(2) **Ο χειρισμός των πληροφοριών μπορεί να γίνει με μεγαλύτερη ευελιξία:** Σε μια μηχανογραφημένη βάση δεδομένων υπάρχει η δυνατότητα οι πληροφορίες να εξετάζονται με βάση συγκεκριμένα κριτήρια και να ετοιμάζονται λίστες με τις πληροφορίες που πληρούν τα κριτήρια αυτά.

Η γραμματεία ενός σχολείου, για παράδειγμα, θα μπορούσε σε μια μηχανογραφημένη βάση δεδομένων να ετοιμάσει με μεγάλη ευκολία έναν κατάλογο με τα στοιχεία των μαθητών (*π.χ. όνομα, επίθετο, διεύθυνση, αριθμός δικαιολογημένων απουσιών, αριθμός αδικαιολόγητων απουσιών, κ.λπ.*), που πληρούν κάποια κριτήρια (*π.χ. οι αδικαιολόγητες απουσίες τους υπερβαίνουν κάποιο αριθμό*). Για να ετοιμαστεί ο ίδιος κατάλογος σε μια έντυπη βάση δεδομένων θα ήταν πολύ πιο χρονοβόρο και επιπλέον θα υπήρχε αυξημένη πιθανότητα να γίνει λάθος.

(3) **Αποφεύγεται η επανάληψη δεδομένων:** Σε μια (σχισιακή) βάση δεδομένων δεν καταχωρούνται τα ίδια δεδομένα περισσότερες από μια φορά. Για παράδειγμα, το ονοματεπώνυμο και η διεύθυνση ενός πελάτη μιας ασφαλιστικής εταιρείας καταχωρούνται ως μέρος των στοιχείων του πελάτη. Εάν ο πελάτης έχει κάνει ασφάλεια ζωής και ασφάλεια ιδιωτικού οχήματος, δεν θα καταχωρηθούν το όνομα και διεύθυνσή του ξεχωριστά μαζί με τα στοιχεία της κάθε ασφάλειας. Έτσι, εάν μελλοντικά δηλώσει ότι έχει αλλάξει διεύθυνση κατοικίας, αυτόματα θα ισχύει η ίδια διεύθυνση για όλες τις ασφάλειες του πελάτη και δεν θα υπάρχει η πιθανότητα να αποστέλλονται έγγραφα για μια ασφάλεια σε λανθασμένη διεύθυνση.

(4) **Υπάρχει έλεγχος στη καταχώριση των δεδομένων:** Κατά την καταχώριση των δεδομένων σε μια βάση δεδομένων μπορούν να εξασκούνται λογικοί έλεγχοι πάνω σε αυτά. Με αυτό τον τρόπο αποτρέπεται η εισαγωγή παράλογων και λανθασμένων δεδομένων, όπως για παράδειγμα μελλοντικές ημερομηνίες ως ημερομηνίες γεννήσεως, κείμενο αντί για αριθμό, αρνητικοί αριθμοί για ποσότητες προϊόντων, αριθμοί τηλεφώνων με πιο λίγα ή πιο πολλά ψηφία από ότι πρέπει, κ.λπ.

(5) **Υπάρχει δυνατότητα πρόσβασης (προσπέλασης) στα δεδομένα μιας βάσης δεδομένων από πολλούς χρήστες ταυτόχρονα:** Με τη χρήση των δικτύων μπορούν πολλοί χρήστες ταυτόχρονα από διάφορους χώρους να έχουν πρόσβαση στα δεδομένα μιας βάσης δεδομένων.

Για παράδειγμα, η βάση δεδομένων μιας τράπεζας, όπου οι υπάλληλοι από τα διαφορετικά καταστήματα της τράπεζας έχουν πρόσβαση στους ίδιους λογαριασμούς ή η βάση δεδομένων μιας αεροπορικής εταιρείας, όπου οποιοσδήποτε μπορεί μέσω του διαδικτύου να ψάξει για πτήσεις και να κάνει κρατήσεις.

4. Μειονέκτημα των μηχανογραφημένων βάσεων δεδομένων

Το κύριο μειονέκτημα των μηχανογραφημένων βάσεων δεδομένων είναι ο **χρόνος** και το **κόστος** που απαιτείται για να σχεδιαστεί, να αναπτυχθεί και να συντηρηθεί μια βάση δεδομένων. Όσο πιο πολλά είναι τα δεδομένα και όσο πιο πολύπλοκη ή συχνή η επεξεργασία τους, τόσο τα πλεονεκτήματα των μηχανογραφημένων βάσεων δεδομένων οδηγούν στην επιλογή τους αντί των έντυπων βάσεων δεδομένων, σε βαθμό που πολλές φορές μια έντυπη βάση δεδομένων να εξαγεται από μια μηχανογραφημένη (π.χ. ο τηλεφωνικός κατάλογος).

5. Βάσεις Δεδομένων από την καθημερινή μας ζωή

Μερικά **παραδείγματα** βάσεων δεδομένων από την καθημερινή μας ζωή είναι τα εξής:

- Βάσεις Δεδομένων Σχολείων
- Βάσεις Δεδομένων Τραπεζών
- Βάσεις Δεδομένων για διαχείριση της αποθήκης σε Υπεραγορές
- Ιατρικά Συστήματα Διαχείρισης Ασθενών σε Νοσοκομεία
- Συστήματα Κρατήσεων Αεροπορικών Εισιτηρίων
- Βάσεις Δεδομένων Τηλεφωνίας (*Μηχανογραφημένοι Τηλεφωνικοί Κατάλογοι*).

Βασικές Έννοιες***Βάση Δεδομένων
(Database):***

Είναι ένα σύνολο από πληροφορίες που σχετίζονται μεταξύ τους και έχουν οργανωθεί έτσι ώστε η χρήση τους να είναι γρήγορη και αποτελεσματική.

***Έντυπες
Βάσεις Δεδομένων:***

Είναι βάσεις δεδομένων που υπάρχουν σε έντυπη μορφή.

***Ηλεκτρονικές Βάσεις
Δεδομένων:***

Είναι οι μηχανογραφημένες βάσεις δεδομένων, δηλαδή αυτές που βρίσκονται εγκατεστημένες σε ηλεκτρονικούς υπολογιστές.

***Σύστημα Διαχείρισης
Βάσεων Δεδομένων
(ΣΔΒΔ) (Database
Management System,
DBMS):***

Είναι λογισμικό το οποίο είναι κατασκευασμένο έτσι ώστε να μπορεί να διαχειρίζεται διαφορετικές βάσεις δεδομένων οι οποίες έχουν σχεδιαστεί και οργανωθεί με συγκεκριμένο τρόπο.

***Σχεσιακή Βάση Δεδομένων
(Relational Database):***

Τρόπος σχεδιασμού και οργάνωσης των δεδομένων, το οποίο προτείνει ότι μια βάση δεδομένων πρέπει να είναι οργανωμένη σε πίνακες οι οποίοι σχετίζονται μεταξύ τους με βάση κάποια από τα δεδομένα που περιέχουν.

***Σύστημα Διαχείρισης
Σχεσιακών Βάσεων
Δεδομένων (Relational
Database Management
System, RDBMS):***

Είναι λογισμικό το οποίο είναι κατασκευασμένο έτσι ώστε να μπορεί να διαχειρίζεται βάσεις δεδομένων οι οποίες έχουν σχεδιαστεί και οργανωθεί ως Σχεσιακές Βάσεις Δεδομένων. Παραδείγματα ΣΔΒΔ είναι και τα ακόλουθα: MySQL, PostgreSQL, Oracle, SQLite, Microsoft SQL Server, Microsoft Access και SAP.

Γ6.2 Βασικά Δομικά Συστατικά Στοιχεία και Κύρια Αντικείμενα μιας Βάσης Δεδομένων

Τι θα μάθουμε σήμερα:

- Να αναγνωρίζουμε και να περιγράψουμε πώς είναι οργανωμένη μια βάση δεδομένων όσον αφορά τα βασικά δομικά συστατικά στοιχεία της (*πίνακες, εγγραφές και πεδία*)
- Να εντοπίζουμε και να περιγράψουμε τις σχέσεις που διέπουν τους πίνακες σε μια βάση δεδομένων και τα πλεονεκτήματα που προκύπτουν από αυτές
- Να δημιουργούμε και να αποθηκεύουμε μια βάση δεδομένων σε συγκεκριμένη θέση σε μονάδα δίσκου
- Να ανοίγουμε μια βάση δεδομένων από συγκεκριμένη θέση από μονάδα δίσκου
- Να αναγνωρίζουμε τα κύρια αντικείμενα μιας βάσης δεδομένων.

1. Βασικά Δομικά Συστατικά Στοιχεία μιας Βάσης Δεδομένων

1.1 Πίνακας (Table)

Το πιο βασικό δομικό συστατικό στοιχείο μιας βάσης δεδομένων είναι ο **πίνακας**. Η οργάνωση και η αποθήκευση των πληροφοριών (*δεδομένων*) μιας βάσης δεδομένων γίνεται με τη χρήση πινάκων.

Μια βάση δεδομένων μπορεί να αποτελείται από πολλούς πίνακες και σε κάθε πίνακα αποθηκεύονται πληροφορίες σχετικά με ένα συγκεκριμένο θέμα.

Για παράδειγμα, ένας πίνακας μπορεί να περιέχει όλα τα στοιχεία των μελών ενός Κέντρου Νεότητας, ένας άλλος πίνακας να περιέχει όλες τις δραστηριότητες που προσφέρει το Κέντρο Νεότητας κ.λπ.

Ο κάθε πίνακας έχει το δικό του **όνομα** και τα δεδομένα του είναι οργανωμένα σε γραμμές (εγγραφές) και στήλες (πεδία).



| YouthMem | Firstname | Surname | Sex | Birthdate | Address | Town |
|----------|------------|-----------|-----|-----------|------------------|---------|
| 1001 | Αβραάμ | Κυριάκου | A | 12/8/1999 | Καντάρας 19 | Λεμεσός |
| 1002 | Αχιλλέας | Αλέξη | A | 13/8/1999 | Καντάρας 15 | Λεμεσός |
| 1004 | Αλέξανδρος | Νικόλαου | A | 15/8/1999 | Αμμοχώστου 37 | Λεμεσός |
| 1005 | Αλέξανδρος | Αντώνιου | A | 16/8/1999 | Αμμοχώστου 13 | Λεμεσός |
| 1006 | Αλεξάνδρα | Αθανάσιου | Θ | 17/8/1999 | Αμμοχώστου 11 | Λεμεσός |
| 1007 | Αλέξης | Παντελή | A | 18/8/1999 | Μόρφου 26 | Λεμεσός |
| 1008 | Αναστασία | Δέμη | Θ | 19/8/1999 | Μόρφου 16 | Λεμεσός |
| 1009 | Άντρεα | Κυριάκου | Θ | 20/8/1999 | Κυρηνείας 7 | Λεμεσός |
| 1010 | Άντρεα | Γεώργιου | Θ | 21/8/1999 | Ριζοκαρπάσου 912 | Λεμεσός |
| 1011 | Άντρεα | Στυλιανού | Θ | 22/8/1999 | Ριζοκαρπάσου 7 | Λεμεσός |
| 1013 | Ανδρέας | Κόκου | A | 24/8/1999 | Ριζοκαρπάσου 15 | Λεμεσός |
| 1014 | Ανδρέας | Βάσου | A | 25/8/1999 | Αμμοχώστου 27 | Λεμεσός |
| 1015 | Ανδρέας | Εφραίμ | A | 26/8/1999 | Καντάρας 27 | Λεμεσός |
| 1016 | Ανδρέας | Σωκράτη | A | 27/8/1999 | Ριζοκαρπάσου 16 | Λεμεσός |

Ο Πίνακας:
YouthMembers

1.2 Εγγραφή (Record)

Εγγραφή είναι η κάθε **γραμμή** ενός πίνακα μιας βάσης δεδομένων όπου περιέχει όλα τα δεδομένα που περιγράφουν μια συγκεκριμένη καταχώριση του πίνακα και αποτελεί μια ενότητα στοιχείων.

| YouthMem | Firstname | Surname | Sex | Birthdate | Address | Town |
|----------|------------|-----------|-----|-----------|---------------|---------|
| 1001 | Αβραάμ | Κυριάκου | A | 12/8/1999 | Καντάρας 19 | Λεμεσός |
| 1002 | Αγλλέας | Αλέξη | A | 13/8/1999 | Καντάρας 15 | Λεμεσός |
| 1004 | Αλέξανδρος | Νικόλαου | A | 15/8/1999 | Αμμοχώστου 37 | Λεμεσός |
| 1005 | Αλέξανδρος | Αντώνιου | A | 16/8/1999 | Αμμοχώστου 13 | Λεμεσός |
| 1006 | Αλεξάνδρα | Αθανάσιου | Θ | 17/8/1999 | Αμμοχώστου 11 | Λεμεσός |
| 1007 | Αλέξης | Παντελή | A | 18/8/1999 | Μόρφου 26 | Λεμεσός |
| 1008 | Αναστασία | Δέμη | Θ | 19/8/1999 | Μόρφου 16 | Λεμεσός |
| 1009 | Άντρεα | Κυριάκου | Θ | 20/8/1999 | Κυρηνείας 7 | Λεμεσός |

εγγραφή

1.3 Πεδίο (Field)

Πεδίο είναι η κάθε **στήλη** ενός πίνακα μιας βάσης δεδομένων. Σε ένα πεδίο καταχωρούνται οι τιμές για ένα συγκεκριμένο χαρακτηριστικό των εγγραφών του πίνακα. Τα δεδομένα που βρίσκονται στο κάθε πεδίο είναι πάντα του ίδιου τύπου. Κάθε πεδίο έχει πάντα διαφορετικό όνομα.

Για παράδειγμα, σε έναν πίνακα πελατών, **ένα πεδίο** θα είναι τα ονόματα των πελατών, **άλλο πεδίο** τα επώνυμά τους, **άλλο πεδίο** η διεύθυνσή τους και **άλλο πεδίο** τα τηλέφωνα τους.

| YouthMem | Firstname | Surname | Sex | Birthdate | Address | Town |
|----------|------------|-----------|-----|-----------|---------------|---------|
| 1001 | Αβραάμ | Κυριάκου | A | 12/8/1999 | Καντάρας 19 | Λεμεσός |
| 1002 | Αγλλέας | Αλέξη | A | 13/8/1999 | Καντάρας 15 | Λεμεσός |
| 1004 | Αλέξανδρος | Νικόλαου | A | 15/8/1999 | Αμμοχώστου 37 | Λεμεσός |
| 1005 | Αλέξανδρος | Αντώνιου | A | 16/8/1999 | Αμμοχώστου 13 | Λεμεσός |
| 1006 | Αλεξάνδρα | Αθανάσιου | Θ | 17/8/1999 | Αμμοχώστου 11 | Λεμεσός |
| 1007 | Αλέξης | Παντελή | A | 18/8/1999 | Μόρφου 26 | Λεμεσός |
| 1008 | Αναστασία | Δέμη | Θ | 19/8/1999 | Μόρφου 16 | Λεμεσός |
| 1009 | Άντρεα | Κυριάκου | Θ | 20/8/1999 | Κυρηνείας 7 | Λεμεσός |

πεδίο

2. Ειδικές Σχέσεις μεταξύ πινάκων (Relationships)

Όπως έχουμε δει πιο πάνω, μια βάση δεδομένων μπορεί να αποτελείται από περισσότερους από έναν πίνακα και κάθε πίνακας αποθηκεύει δεδομένα για ένα συγκεκριμένο θέμα.

Οι διάφοροι πίνακες της βάσης δεδομένων μπορούν να σχετίζονται μεταξύ τους με μια ειδική σχέση. Αυτή η ειδική σχέση, η οποία ονομάζεται **Relationship**, είναι μια λογική σύνδεση μεταξύ δύο πινάκων που έχουν τουλάχιστον ένα κοινό πεδίο (δηλαδή ένα πεδίο του οποίου τα δεδομένα είναι κοινά για κάποιες εγγραφές στον ένα πίνακα και για κάποιες εγγραφές στον άλλο πίνακα).

Η δημιουργία αυτής της σχέσης ανάμεσα σε δύο πίνακες μιας βάσης δεδομένων, μας επιτρέπει να αντλήσουμε στοιχεία και πληροφορίες που ανήκουν και στους δύο πίνακες.

Αυτό έχει ως αποτέλεσμα να έχουμε τα πιο κάτω **πλεονεκτήματα**:

- Δεν χρειάζεται να καταχωρίσουμε τα ίδια δεδομένα πολλές φορές
- Γίνεται ευκολότερη η διαχείριση των δεδομένων (*ενημέρωση, διαγραφή κ.λπ.*)
- Μικρότερος όγκος δεδομένων, πιο γρήγορη αναζήτηση
- Μειώνεται το περιθώριο σφαλμάτων
- Περισσότερη σαφήνεια στην οργάνωση των δεδομένων.

Τα **είδη** των ειδικών σχέσεων που υπάρχουν μεταξύ των πινάκων είναι:

2.1 Ειδική Σχέση: Ένα-Προς-Ένα (One-To-One, 1-1)

Κάθε εγγραφή από τον ΠΡΩΤΟ πίνακα σχετίζεται με μόνο μία εγγραφή από τον ΔΕΥΤΕΡΟ πίνακα, αλλά και αντίστροφα.

2.2 Ειδική Σχέση: Ένα-Προς-Πολλά (One-To-Many, 1-∞)

Κάθε εγγραφή από τον ΠΡΩΤΟ πίνακα σχετίζεται με πολλές εγγραφές από τον ΔΕΥΤΕΡΟ πίνακα, αλλά κάθε εγγραφή από το ΔΕΥΤΕΡΟ πίνακα, σχετίζεται με μόνο μία εγγραφή από τον ΠΡΩΤΟ πίνακα.

Ένα **παράδειγμα** μιας ειδικής σχέσης Ένα-προς-Πολλά, είναι στη βάση δεδομένων του Κέντρου Νεότητας (YouthCentre.accdb). Εδώ υπάρχουν δύο πίνακες:

- Ο πρώτος πίνακας που έχει το όνομα **Activities** περιέχει την αθλητική δραστηριότητα με την οποία ασχολείται το κάθε μέλος.
- Ο δεύτερος πίνακας που έχει το όνομα **YouthMembers** περιέχει τα ονόματα των μελών του Κέντρου Νεότητας.
- Οι δύο πίνακες συνδέονται μεταξύ τους με την ειδική σχέση Ένα-Προς-Πολλά. Δηλαδή, κάθε μέλος από τον πίνακα **YouthMembers** λαμβάνει μέρος μόνο σε μία δραστηριότητα του πίνακα **Activities**, ενώ σε κάθε δραστηριότητα υπάρχουν πολλά μέλη.

| ActivityCode | Description | Registration | Click to Add | | | | |
|--------------|--------------|--------------|--------------|------------|------------------|----------|-----------|
| 1 | Ποδόσφαιρο | 100,00 € | | | | | |
| YouthMem | Firstname | Surname | Sex | Birthdate | Address | Town | TelNumber |
| 1001 | Αβραάμ | Κυριάκου | A | 12/8/1999 | Καντάρας 19 | Λεμεσός | 99651298 |
| 1002 | Αχιλλέας | Αλέξη | A | 13/8/1999 | Καντάρας 15 | Λεμεσός | 99274730 |
| 1004 | Αλέξανδρος | Νικόλαου | A | 15/8/1999 | Αμμοχώστου 37 | Λεμεσός | 99434069 |
| 1127 | Λένα | Γκία | Θ | 4/3/1998 | Ριζοκαρπάσου 901 | Λάρνακα | 99827728 |
| 1128 | Λιούσι | Ρομπέρτ | Θ | 5/3/1998 | Καντάρας 10 | Λάρνακα | 99804020 |
| 1215 | Ζήνα | Ερρίκου | Θ | 19/10/2002 | Ριζοκαρπάσου 35 | Λεμεσός | 99974523 |
| * | | | | | | | |
| 2 | Καλαθόσφαιρα | 80,00 € | | | | | |
| YouthMem | Firstname | Surname | Sex | Birthdate | Address | Town | TelNumber |
| 1005 | Αλέξανδρος | Αντώνιου | A | 16/8/1999 | Αμμοχώστου 13 | Λεμεσός | 99174661 |
| 1027 | Άννα | Γιώργου | Θ | 10/2/2000 | Ριζοκαρπάσου 13 | Λεμεσός | 99262876 |
| 1045 | Χαραλαμπία | Τάσου | Θ | 28/2/2000 | Ριζοκαρπάσου 911 | Λευκωσία | 99127935 |
| 1060 | Χρυστάλλα | Νικόλαου | Θ | 14/3/2000 | Καντάρας 35 | Λευκωσία | 99157570 |
| 1185 | Ραφαέλλα | Μιχαήλ | Θ | 19/9/2002 | Μόρφου 40 | Λευκωσία | 99210223 |
| * | | | | | | | |
| 3 | Πετόσφαιρα | 75,00 € | | | | | |
| YouthMem | Firstname | Surname | Sex | Birthdate | Address | Town | TelNumber |
| 1006 | Αλεξάνδρα | Αθανάσιου | Θ | 17/8/1999 | Αμμοχώστου 11 | Λεμεσός | 99986377 |
| 1010 | Άντρεα | Γεώργιου | Θ | 21/8/1999 | Ριζοκαρπάσου 912 | Λεμεσός | 99222077 |
| 1028 | Άννα | Γεννάδιου | Θ | 11/2/2000 | Κυρηνείας 31 | Λεμεσός | 99645647 |
| 1181 | Πετρίνα | Κυριάκου | Θ | 15/9/2002 | Αμμοχώστου 21 | Λευκωσία | 99156880 |
| 1182 | Προκόπης | Σάββα | A | 16/9/2002 | Κυρηνείας 18 | Λευκωσία | 99927797 |

3. Κύρια Αντικείμενα Βάσης Δεδομένων (Objects)

3.1 Πίνακας (Table)

Όπως έχουμε ήδη αναφέρει, ένας **πίνακας** είναι μια συλλογή δεδομένων που περιγράφουν ομοειδή αντικείμενα.

3.2 Φόρμα (Form)

Οι **φόρμες** είναι ένας εύκολος τρόπος για την καταχώριση και την επεξεργασία των δεδομένων στους πίνακες. Μια φόρμα είναι άμεσα συνδεδεμένη με έναν ή περισσότερους πίνακες και εμφανίζει συγκεντρωμένα τα στοιχεία που αφορούν την κάθε εγγραφή. Τα δεδομένα τα παίρνει από έναν πίνακα ή τα αποθηκεύει σε έναν πίνακα.

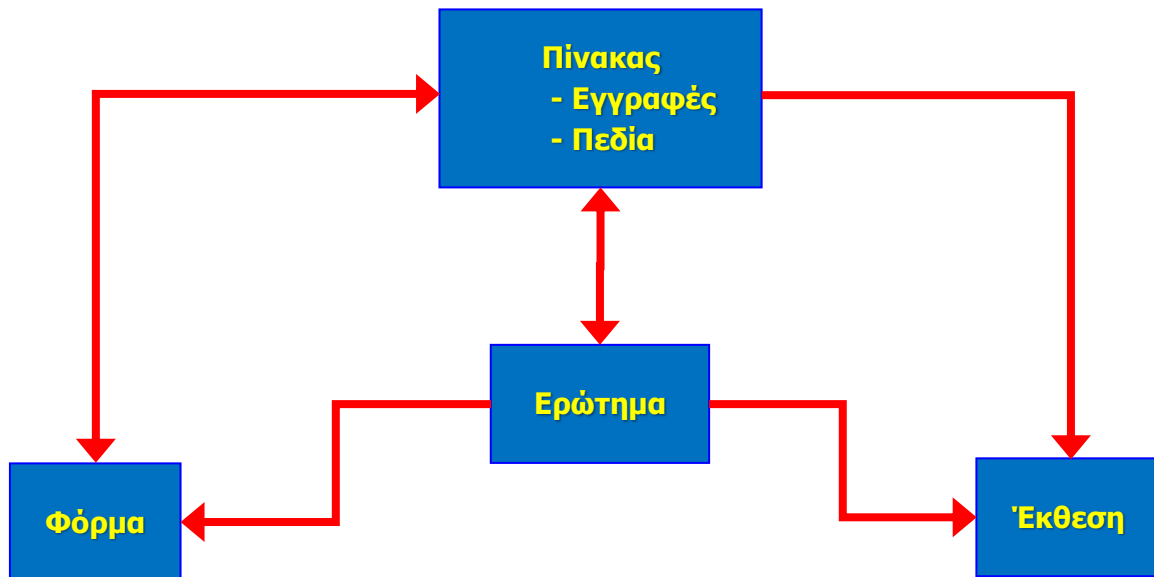
3.3 Ερωτήματα (Queries)

Με τη βοήθεια των **ερωτημάτων** και χρησιμοποιώντας τα κατάλληλα κριτήρια, μπορούμε να εντοπίζουμε και να παρουσιάζουμε τα δεδομένα που επιθυμούμε από μια βάση δεδομένων.

3.4 Εκθέσεις (Reports)

Οι **εκθέσεις** αποτελούν έναν αποτελεσματικό τρόπο για την παρουσίαση και την εκτύπωση επιλεγμένων πληροφοριών μιας βάσης δεδομένων. Μας επιτρέπουν να ομαδοποιούμε τα δεδομένα σε διάφορα επίπεδα, να καθορίζουμε τη διάταξή τους και να τα μορφοποιούμε κατάλληλα.

4. Η Δομή μιας Βάσης Δεδομένων



5. Εφαρμογή Βάσεων Δεδομένων (Microsoft Access)

Όπως έχει ήδη αναφερθεί, για τη δημιουργία, διαχείριση και συντήρηση των ηλεκτρονικών βάσεων δεδομένων χρησιμοποιούμε ειδικά λογισμικά που ονομάζονται **Συστήματα Διαχείρισης Βάσης Δεδομένων (ΣΔΒΔ) - Database Management Systems (DBMS)**. Ένα από αυτά είναι η Microsoft Access.

6. Ξεκίνημα της Εφαρμογής Βάσεων Δεδομένων

(1) **Start**→**All Programs**→**Microsoft Office**→ **Microsoft Access**.

ή

Εάν υπάρχει ήδη το εικονίδιο της εφαρμογής στην επιφάνεια εργασίας (**Desktop**)

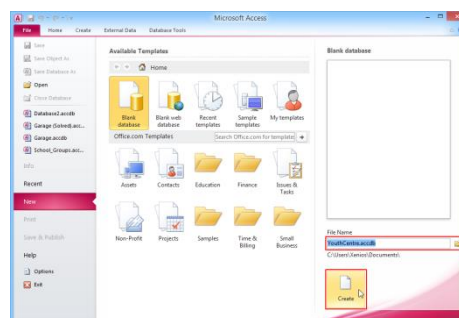


, μπορούμε απλώς να κάνουμε διπλό κλικ πάνω του.

7. Δημιουργία νέας κενής βάσης δεδομένων (Blank database)

Για τη δημιουργία μιας νέας βάσης δεδομένων, εφόσον έχει ήδη ξεκινήσει η εφαρμογή, κάνουμε τα ακόλουθα:

(1) Από την καρτέλα File επιλέγουμε την εντολή **New** (εάν δεν είναι ήδη προεπιλεγμένη) και από την περιοχή **Available Templates**, το εικονίδιο Blank database και μετά κλικ στο κουμπι **Create**.

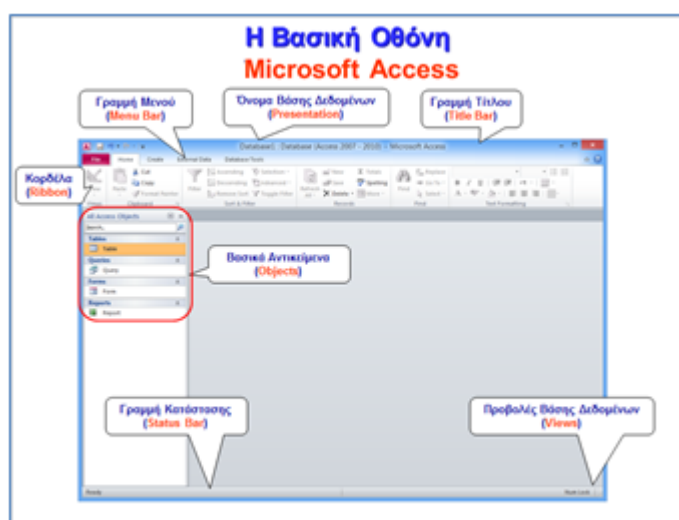


ή

Μπορούμε επίσης να χρησιμοποιήσουμε τον συνδυασμό πλήκτρων **CTRL + N**.

(2) Ακολούθως, κάτω δεξιά στο παράθυρο πληκτρολογούμε το όνομα που θέλουμε να δώσουμε στη βάση δεδομένων μας, επιλέγουμε το χώρο που θέλουμε να αποθηκευτεί και κάνουμε κλικ στο κουμπι **Create**.

(3) Στη συνέχεια, θα ανοίξει το παράθυρο της Microsoft Access, από το οποίο και θα γίνει η ανάπτυξη της βάσης δεδομένων.

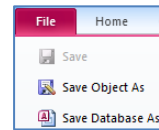



8. Αποθήκευση βάσης δεδομένων (Save)

Εάν θα αποθηκεύσουμε τη βάση δεδομένων μας σε Φορητή Μνήμη (Flash Memory), τοποθετούμε τη Φορητή Μνήμη μας σε μια θύρα USB.

Η προεπιλεγμένη μορφή αρχείου όταν αποθηκεύουμε είναι το **Microsoft Access Database**, το οποίο δίνει την προέκταση **.accdb** στο αρχείο.

- (1) Από την καρτέλα **File** επιλέγουμε την εντολή **Save Database As** για να αποθηκεύσουμε τη βάση δεδομένων μας, ή όταν θέλουμε να την αποθηκεύσουμε με άλλο όνομα ή σε άλλη θέση (*εναλλακτικά χρησιμοποιούμε τον συνδυασμό πλήκτρων **ALT + F, A***).



- Χρησιμοποιούμε την εντολή **File→Save** για να αποθηκεύσουμε μία ήδη αποθηκευμένη βάση δεδομένων, με το ίδιο όνομα και στην ίδια θέση μετά από αλλαγές που κάναμε.
- Αντί της εντολής **File→Save** μπορούμε εναλλακτικά να κάνουμε κλικ στο κουμπί  ή πατάμε τον συνδυασμό πλήκτρων **CTRL + S**.

(2) Καθορίζουμε το όνομα (**Filename**), τη θέση (**Save in**) και τον τύπο (**Save as type**).

(3) Τέλος, κάνουμε κλικ στο κουμπί **Save** για να το αποθηκεύσουμε.

9. Άνοιγμα βάσης δεδομένων (Open)

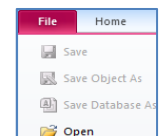
Για να ανοίξουμε μια υφιστάμενη βάση δεδομένων, εφόσον έχει ήδη ξεκινήσει η εφαρμογή, τότε:

- (1) Από την καρτέλα **File** επιλέγουμε την εντολή **Open**.

ή

Μπορούμε επίσης να χρησιμοποιήσουμε τον συνδυασμό πλήκτρων **CTRL + O**.

(2) Από το παράθυρο που θα ανοίξει, εντοπίζουμε και ανοίγουμε τη βάση δεδομένων που επιθυμούμε.

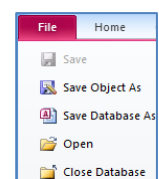


10. Κλείσιμο βάσης δεδομένων (Close Database)

- (1) Από την καρτέλα **File** επιλέγουμε την εντολή **Close Database**.

ή

Μπορούμε επίσης να χρησιμοποιήσουμε τον συνδυασμό πλήκτρων **ALT + F, C**.

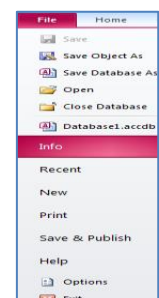


11. Κλείσιμο Εφαρμογής Βάσεων Δεδομένων (Exit)

- (1) Από την καρτέλα **File** επιλέγουμε την εντολή **Exit**.

ή








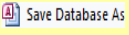





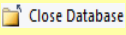




Μπορούμε επίσης να χρησιμοποιήσουμε τον συνδυασμό πλήκτρων **ALT + F, X**.



Βασικές Έννοιες

| | |
|-------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Πίνακας (Table): | Η οργάνωση των πληροφοριών μιας βάσης δεδομένων γίνεται με τη χρήση πινάκων. Ο κάθε <u>πίνακας</u> έχει το δικό του όνομα και αποτελείται από γραμμές και στήλες. Στον πίνακα τα δεδομένα είναι οργανωμένα σε <u>στήλες</u> (πεδία) και σε <u>γραμμές</u> (εγγραφές). |
| Εγγραφή (Record): | <u>Εγγραφή</u> είναι η κάθε γραμμή ενός πίνακα μιας βάσης δεδομένων η οποία περιέχει όλα τα δεδομένα που περιγράφουν μια συγκεκριμένη καταχώριση του πίνακα και αποτελεί μια ενότητα στοιχείων. |
| Πεδίο (Field): | <u>Πεδία</u> είναι οι στήλες ενός πίνακα μιας βάσης δεδομένων στις οποίες καταχωρούνται οι τιμές για ένα συγκεκριμένο χαρακτηριστικό των εγγραφών του πίνακα. |
| Ειδικές Σχέσεις πινάκων (Relationships): | Κάθε πίνακας αποτελεί ομάδα από στοιχεία που έχουν μια σχέση μεταξύ τους. Οι πίνακες όμως μπορεί να έχουν και αυτοί μεταξύ τους μια ειδική σχέση. Αυτή η <u>ειδική σχέση</u> ονομάζεται Relationship . |
| Κύρια Αντικείμενα Βάσης Δεδομένων (Objects): | Πίνακες (Tables) Φόρμες (Forms) Ερωτήματα (Queries) Εκθέσεις (Reports) |

Υπόμνημα

| <u>Περιγραφή</u> (Description) | <u>Καρτέλα</u> (Tab) | <u>Εικονίδιο</u> (Icon) | <u>Πλήκτρα</u> (Keys) |
|-----------------------------------------------------|------------------------------------------------------------------------------------|------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|
| Δημιουργία νέας βάσης δεδομένων (Blank database) |  |  |  |
| Αποθήκευση βάσης δεδομένων (Save) |  |  |  |
| Αποθήκευση βάσης δεδομένων ως (Save Database As) |  |  |  |
| Άνοιγμα βάσης δεδομένων (Open) |  |  |  |
| Κλείσιμο βάσης δεδομένων (Close Database) |  |  |  |
| Κλείσιμο Εφαρμογής Βάσεων Δεδομένων (Exit) |  |  |  |

Γ6.3 Πίνακες

Τι θα μάθουμε σήμερα:

- ❖ Να δημιουργούμε και να αποθηκεύουμε έναν πίνακα σε μια βάση δεδομένων και να καθορίζουμε τα πεδία και τους τύπους δεδομένων των πεδίων
- ❖ Να καθορίζουμε τις ιδιότητες των πεδίων ενός πίνακα
- ❖ Να αναφέρουμε τα κριτήρια επιλογής προσδιορισμού ενός πρωτεύοντος κλειδιού και να καθορίζουμε το κατάλληλο πεδίο ως πρωτεύον κλειδί
- ❖ Να χρησιμοποιούμε την εναλλαγή προβολών πίνακα (Design View και Datasheet View)
- ❖ Να διαγράφουμε ένα πίνακα.

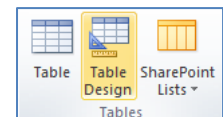
1. Πίνακας (Table)

Ο πίνακας είναι μια συλλογή δεδομένων που περιγράφουν ομοειδή (που έχουν σχέση μεταξύ τους) αντικείμενα. Όλα τα δεδομένα που καταχωρούμε σε μια βάση δεδομένων **αποθηκεύονται σε έναν ή περισσότερους πίνακες** και για αυτό θεωρούνται τα βασικότερα στοιχεία μιας βάσης δεδομένων.

Για **παράδειγμα**, ένας πίνακας μπορεί να περιέχει όλα τα στοιχεία των πελατών μας, ένας άλλος να περιέχει τα στοιχεία των προμηθευτών μας, κ.λπ.

2. Δημιουργία Πίνακα με τη χρήση της Προβολής Σχεδίασης (Table Design)

(1) Από την κάρτελα **Create** και την ομάδα **Tables** επιλέγουμε το εικονίδιο **Table Design**.



(2) Στο παράθυρο που εμφανίζεται, και από την **πρώτη γραμμή** του πλαισίου του **Field Name** πληκτρολογούμε το όνομα του πεδίου που θέλουμε να χρησιμοποιήσουμε.

(3) Στο πλαίσιο του **Data Type** επιλέγουμε από τη λίστα τον τύπο δεδομένων του πεδίου που θέλουμε να χρησιμοποιήσουμε.

(4) Στο πλαίσιο του **Description** πληκτρολογούμε την περιγραφή-σχόλια του πεδίου που θέλουμε να χρησιμοποιήσουμε.

| Field Name | Data Type | Description (Optional) |
|--------------|------------|----------------------------------------------|
| Όνομα πελάτη | Short Text | Εδώ γράφουμε τα ονόματα των πελατών μας |
| Διεύθυνση | Short Text | Εδώ γράφουμε τις διευθύνσεις των πελατών μας |

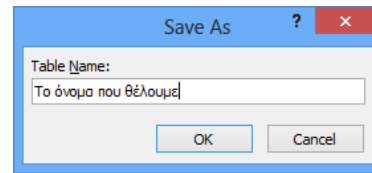
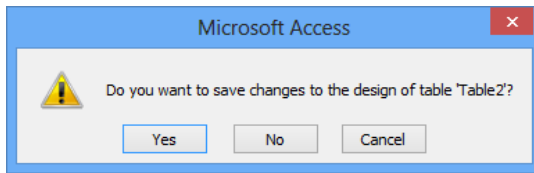
Field Properties

| | |
|---------------------|------------|
| General (αποκρυφ) | |
| Field Size | 255 |
| Format | |
| Input Mask | |
| Caption | |
| Default Value | |
| Validation Rule | |
| Validation Text | |
| Required | No |
| Allow Zero Length | Yes |
| Indexed | No |
| Unicode Compression | No |
| IME Mode | No Control |
| IME Sentence Mode | None |
| Text Align | General |

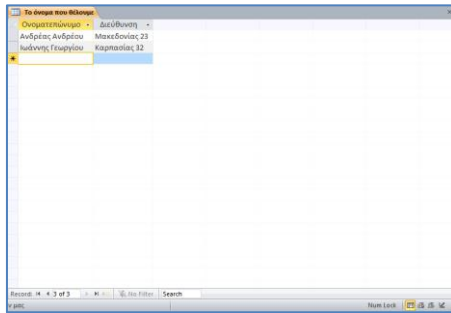
The maximum number of characters you can enter in the field. The largest maximum you can set is 255. Press F1 for help on field size.

(5) Ακολούθως, στη **δεύτερη γραμμή**, πληκτρολογούμε ένα νέο πεδίο με τον νέο τύπο δεδομένων του και τη νέα περιγραφή του και συνεχίζουμε και στις υπόλοιπες γραμμές μέχρι να τελειώσουμε με τα πεδία που θέλουμε να χρησιμοποιήσουμε.

- (6) Στη συνέχεια, κλείνουμε τον πίνακα, του καθορίζουμε ένα **περιγραφικό όνομα** και αποθηκεύουμε τον πίνακά μας.

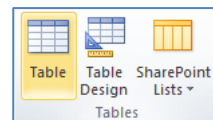



- (7) Τέλος, αρχίζουμε την πληκτρολόγηση των εγγραφών που θέλουμε να χρησιμοποιήσουμε στον πίνακά μας.

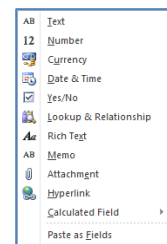



3. Δημιουργία Πίνακα με τη χρήση της Προβολής Φύλλου Δεδομένων (Table)

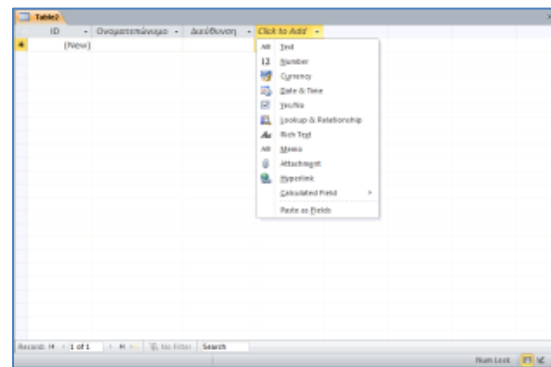
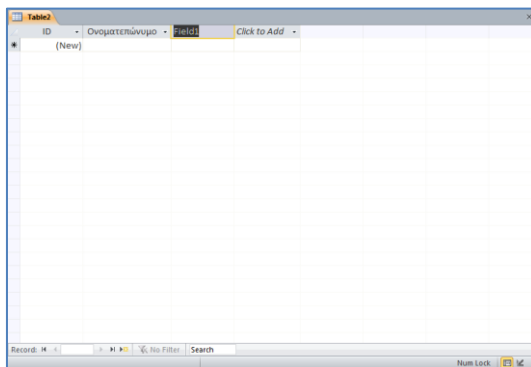
- (1) Από την καρτέλα Create και την ομάδα Tables επιλέγουμε το εικονίδιο Table.



- (2) Ακολούθως, κάνουμε κλικ στο **βελάκι**  δίπλα από το κουμπί **Click to Add** και επιλέγουμε πρώτα τον τύπο δεδομένων του πεδίου που θέλουμε χρησιμοποιήσουμε.

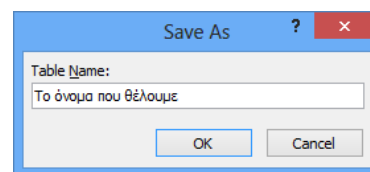
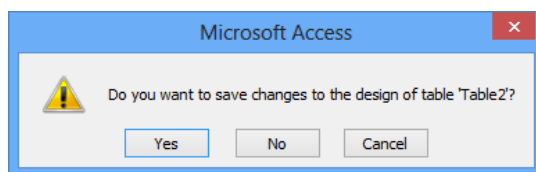


- (3) Στη συνέχεια, στο πλαίσιο του **Field1**  πληκτρολογούμε το όνομα του πεδίου που θέλουμε να χρησιμοποιήσουμε.

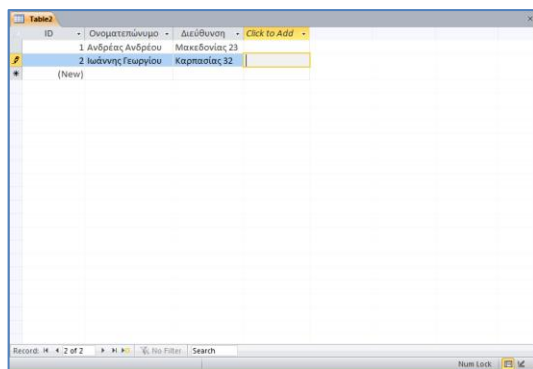


Ακολούθως, πατώντας το πλήκτρο **ENTER** μας πάει στο **επόμενο πεδίο** που θέλουμε να προσθέσουμε και συνεχίζουμε μέχρι να τελειώσουμε με τα πεδία που θέλουμε να χρησιμοποιήσουμε.

- (4) Στη συνέχεια, κλείνουμε τον πίνακα, του καθορίζουμε ένα **περιγραφικό όνομα** και αποθηκεύουμε τον πίνακά μας.



(5) Τέλος, αρχίζουμε την πληκτρολόγηση των εγγραφών που θέλουμε να χρησιμοποιήσουμε στον πίνακά μας.



4. Κανόνες Ονομασίας Πεδίων

Όταν δημιουργούμε έναν πίνακα καλό είναι να χρησιμοποιούμε **περιγραφικά ονόματα πεδίων**, ώστε να είναι κατανοητά και από άλλους χρήστες της βάσης δεδομένων:

- (1) Μπορούν να περιλαμβάνουν οποιονδήποτε συνδυασμό χαρακτήρων και αριθμών, **εκτός από** διάφορα σύμβολα π.χ. την τελεία (.), το θαυμαστικό (!) και τις αγκύλες ([]).
- (2) ΔΕΝ επιτρέπεται να αρχίζουν με ΚΕΝΟ διάστημα.

5. Τύποι Δεδομένων (Data Type)

Ο τύπος δεδομένων ορίζει το **είδος των δεδομένων** που θα μπορεί ο χρήστης να καταχωρίσει στο πεδίο. Χρησιμοποιώντας σωστά τους τύπους δεδομένων μπορούμε να περιορίσουμε τις λανθασμένες καταχωρίσεις στη βάση δεδομένων και να διασφαλίσουμε τη σωστή λειτουργία της.

Η Microsoft Access, όπως και τα περισσότερα ΣΔΒΔ υποστηρίζει τους εξής βασικούς **τύπους δεδομένων**:

- **Short Text** Κείμενο μικρής έκτασης
- **Number** Αριθμός
- **Date/Time** Ημερομηνία/Ωρα
- **Currency** Χρηματικό ποσό
- **AutoNumber** Αυτόματη αρίθμηση
- **Yes/No** Ναι/Όχι ή Σωστό/Λάθος ή Αληθές/Ψευδές

Η Microsoft Access, υποστηρίζει επίσης κάποιους επιπρόσθετους (εξειδικευμένους) τύπους δεδομένων:

- **Long Text** Κείμενο μεγάλης έκτασης

- **OLE Object** Αντικείμενο OLE - Αντικείμενα από άλλες εφαρμογές (*Εικόνα, Ήχο, Βίντεο*)
- **Hyperlink** Υπερσύνδεσμος (*Εισαγωγή συνδέσμου με ιστοσελίδα στο Διαδίκτυο*)
- **Attachment** Επισυναπτόμενο αρχείο
- **Calculated** Μαθηματικός υπολογισμός πεδίων
- **Lookup Wizard...** Καθορισμός τιμών στο πεδίο (*Λίστα Επιλογών*)

Παρακάτω ακολουθεί σύντομη περιγραφή των χαρακτηριστικών για τον κάθε τύπο πεδίου:

- (α) **Short Text** - Μπορεί να περιέχει **μέχρι και 255** χαρακτήρες κειμένου, σύμβολα ή αριθμούς.
- (β) **Number** - Μπορεί να περιέχει **αριθμούς** για να τους χρησιμοποιήσουμε για μαθηματικές πράξεις.
- (γ) **Date/Time** - Κατάλληλος για την καταχώριση δεδομένων που αναφέρονται σε **ημερομηνίες και ώρες**.
- (δ) **Currency** - Κατάλληλος τύπος δεδομένων για την καταχώριση δεδομένων που αναφέρονται σε **χρηματικά ποσά**.
- (ε) **AutoNumber** - Παίρνει μοναδικές αριθμητικές τιμές οι οποίες **καθορίζονται αυτόματα** από την εφαρμογή.
- (στ) **Yes/No** - Μας επιτρέπει να δημιουργούμε πεδία που περιέχουν **λογικές τιμές** (*Αληθές ή Ψευδές, Ναι ή Όχι*).
- (ζ) **Long Text** - Μπορεί να περιέχει **μέχρι και 65.535** χαρακτήρες κειμένου, σύμβολα ή αριθμούς.
- (η) **OLE Object** - Μας επιτρέπει να εισάγουμε αντικείμενα από άλλες εφαρμογές όπως **εικόνα, ήχο και βίντεο**.
- (θ) **Hyperlink** - Κατάλληλος για εισαγωγή διευθύνσεων **υπερσυνδέσμων**.
- (ι) **Attachment** - Κατάλληλος για εισαγωγή **επισυναπτόμενων αρχείων**.
- (ια) **Calculated** - Περιέχει **μαθηματικές πράξεις** για υπολογισμούς πεδίων.
- (ιβ) **Lookup Wizard...** - Περιέχει **προκαθορισμένες τιμές** όπως είναι μια λίστα επιλογών.

6. Ιδιότητες Πεδίων (Field Properties)

Οι ιδιότητες των πεδίων μάς επιτρέπουν να προσαρμόσουμε τον τρόπο με τον οποίο το σύστημα διαχείρισης βάσεων δεδομένων **αποθηκεύει, εμφανίζει και χειρίζεται** τα δεδομένα ενός πεδίου όπως:

- (1) Πόσους χαρακτήρες μπορούν να δεχτούν τα πεδία κειμένου.
- (2) Εάν ένα πεδίο πρέπει οπωσδήποτε να περιέχει κάποια τιμή.
- (3) Ποια μορφή πρέπει να έχουν τα δεδομένα ενός πεδίου.

Π.χ. οι βασικότερες **ιδιότητες των πεδίων** στη Microsoft Access είναι:

- **Field Size** Μέγεθος πεδίου
- **Format** Μορφή (τρόπος εμφάνισης)
- **Input Mask** Μάσκα εισαγωγής
- **Caption** Λεζάντα (Ταμπέλα)
- **Indexed** Ευρετήριο
- **Default Value** Προεπιλεγμένη (αρχική) τιμή
- **Validation Rule** Κανόνας επικύρωσης ή εγκυρότητας
- **Validation Text** Κείμενο επικύρωσης ή εγκυρότητας
- **Required** Απαιτείται (Εάν είναι υποχρεωτικό ή όχι)
- **Decimal Places** Δεκαδικές θέσεις ή Δεκαδικά ψηφία

Παρακάτω ακολουθεί σύντομη περιγραφή των χαρακτηριστικών για την κάθε ιδιότητα των πεδίων:

- (α) **Field Size** - Μας επιτρέπει να καθορίσουμε το **μέγιστο πλήθος** των χαρακτήρων του πεδίου.
- (β) **Format** - Καθορίζουμε τη **μορφή** με την οποία θα παρουσιάζονται τα δεδομένα στον χρήστη.
- (γ) **Input Mask** - Καθορίζουμε τη **μορφή** στην οποία θα μπορεί να καταχωρεί τα δεδομένα ο χρήστης.
- (δ) **Caption** - Καθορίζουμε το **όνομα** που θα εμφανίζεται στον πίνακα αντί για το αρχικό όνομα του πεδίου.
- (ε) **Indexed** – Μας επιτρέπει να εφαρμόσουμε **γρήγορη αναζήτηση** δεδομένων μέσω ενός πεδίου.
- (στ) **Default Value** - Καθορίζουμε την τιμή που θα καταχωρείται **αυτόματα** στο συγκεκριμένο πεδίο σε κάθε νέα εγγραφή.
- (ζ) **Validation Rule** - Εισάγουμε **κριτήρια** στο πεδίο προκειμένου να περιορίσουμε τις τιμές που μπορεί να δέχεται.
- (η) **Validation Text** - Ορίζουμε το **μήνυμα σφάλματος** που θα εμφανίζεται όταν οι τιμές που καταχωρούνται δεν πληρούν τα κριτήρια του κανόνα επικύρωσης.
- (θ) **Required** - Επιλέγουμε εάν θα είναι **υποχρεωτική** η καταχώριση δεδομένων στο πεδίο ή αν θα επιτρέπεται να το αφήνουμε κενό.
- (ι) **Decimal Places** - Καθορίζουμε το πλήθος των **δεκαδικών ψηφίων** που θα εμφανίζονται στα δεξιά της υποδιαστολής.

7. Πρωτεύον Κλειδί (Primary Key)

Η επιλογή του πρωτεύοντος κλειδιού σε έναν πίνακα είναι μια από τις βασικότερες επιλογές που έχουμε να κάνουμε. Το πρωτεύον κλειδί είναι ένα **πεδίο** ή **συνδυασμός πεδίων** του πίνακα, που πρέπει πάντα να περιέχει/ουν έγκυρα μοναδικά δεδομένα, με αποτέλεσμα να

διασφαλίζεται η μοναδικότητα κάθε εγγραφής του πίνακα. Με άλλα λόγια, εάν έχουμε έναν πίνακα και γνωρίζουμε, την τιμή του πρωτεύοντος κλειδιού, θα πρέπει να μπορούμε να εντοπίσουμε το πολύ μία εγγραφή η οποία να έχει αυτή την τιμή. Αυτό σημαίνει ότι δεν θα υπάρχουν δύο ή περισσότερες εγγραφές στον πίνακα που να έχουν την ίδια τιμή για το πρωτεύον κλειδί.

Γενικά, ως πρωτεύον κλειδί ενός πίνακα επιλέγουμε ένα πεδίο που περιέχει κάποιο κωδικό ή αριθμό που πρέπει να είναι **μοναδικός για κάθε εγγραφή** του πίνακα, π.χ.: αριθμός Ταυτότητας, αριθμός μητρώου μαθητή κ.λπ. Θα πρέπει να είμαστε προσεκτικοί στην επιλογή μας, διότι πεδία όπου η καταχώριση είναι προαιρετική δεν είναι κατάλληλα.

Σε κάποιες περιπτώσεις όπου δεν υπάρχει ένα πεδίο για να λειτουργήσει ως πρωτεύον κλειδί, πιθανόν να υπάρχει ένας συνδυασμός πεδίων. Για παράδειγμα, στη βάση δεδομένων ενός σχολείου, στον πίνακα των απουσιών των μαθητών καταχωρούνται ο αριθμός μητρώου, η ημερομηνία απουσίας, η περίοδος απουσίας, το μάθημα και κατά πόσον είναι δικαιολογημένη ή όχι. Κανένα από αυτά τα πεδία από μόνο του δεν είναι κατάλληλο ως πρωτεύον κλειδί, γι αυτό και χρησιμοποιούμε τον συνδυασμό (αριθμός μητρώου, ημερομηνία, περίοδος) ως πρωτεύον κλειδί στον πίνακα αυτό.

Εάν στον πίνακά μας δεν έχουμε κάποιο πεδίο ή συνδυασμό πεδίων που να είναι μοναδικά σε κάθε εγγραφή, τότε η επιλογή του πεδίου αυτόματης αρίθμησης (AutoNumber) είναι ιδανική για τον ορισμό πρωτεύοντος κλειδιού, αφού οι τιμές του είναι εξ' ορισμού μοναδικές.

Παράδειγμα:

Για τον πίνακα των μελών ενός κέντρου νεότητας που φαίνεται στην επόμενη σελίδα, το τηλέφωνο (πεδίο «TelNumber») μπορεί να φαίνεται κατάλληλο ως πρωτεύον κλειδί, αλλά επειδή πιθανό να υπάρχουν μέλη που να μην έχουν το δικό τους τηλέφωνο ή να μοιράζονται (με τα αδέρφια τους) το ίδιο τηλέφωνο δεν μπορεί να το χρησιμοποιήσουμε. Για τον ίδιο λόγο το πεδίο Email είναι ακατάλληλο. Επίσης, ο συνδυασμός των πεδίων Όνομα και Επώνυμο («Firstname» και «Surname») δεν είναι κατάλληλος, διότι υπάρχει η πιθανότητα να έχουμε μέλη με το ίδιο όνομα και επώνυμο (εξάλλου δεν συνηθίζεται να χρησιμοποιούμε πεδία αυτού του τύπου και έκτασης ως πρωτεύον κλειδί). Η καλύτερη επιλογή γι' αυτό τον πίνακα είναι η δημιουργία ενός πεδίου («YouthMemberID») με τύπο δεδομένων AutoNumber με σκοπό να λειτουργήσει ειδικά ως πρωτεύον κλειδί.

| YouthMemberID | Firstname | Surname | Sex | Birthdate | Address | Town | TelNumber | Email | ActivityCode |
|---------------|-----------|--------------|-----|-----------|------------------|---------|-----------|------------------------------------------------------------------|--------------|
| 1013 | Ανδρέας | Κόκου | A | 24/8/1999 | Ριζοκαρπάσου 15 | Λεμεσός | 99715805 | Andre4@gmail.com | 8 |
| 1014 | Ανδρέας | Βάσου | A | 25/8/1999 | Αμμοχώστου 27 | Λεμεσός | 99492649 | Andrew@otenet.com.cy | 9 |
| 1015 | Ανδρέας | Εφραίμ | A | 26/8/1999 | Καντάρας 27 | Λεμεσός | 99404434 | Andrew3@gmail.com | 10 |
| 1016 | Ανδρέας | Σωκράτη | A | 27/8/1999 | Ριζοκαρπάσου 16 | Λεμεσός | 99545302 | Andrew2@gmail.com | 10 |
| 1017 | Ανδρέας | Σταύρου | A | 28/8/1999 | Καντάρας 21 | Λεμεσός | 99839582 | Andrew@cytanet.com.cy | 10 |
| 1018 | Ανδρέας | Γεώργιου | A | 1/2/2000 | Μόρφου 19 | Λεμεσός | 99764158 | Andrew1@gmail.com | 8 |
| 1019 | Ανδρέας | Γεώργιου | A | 2/2/2000 | Κυρηγείας 29 | Λεμεσός | 99857363 | Andrew@gmail.com | 9 |
| 1020 | Ανδρέας | Κωνσταντίνου | A | 3/2/2000 | Καντάρας 18 | Λεμεσός | 99557156 | Andrew@yahoo.com | 9 |
| 1021 | Ανδρέας | Πέτρου | A | 4/2/2000 | Ριζοκαρπάσου 913 | Λεμεσός | 99651574 | Andrew@yahoo.gr | 7 |
| 1022 | Άντρια | Ανδρέα | Θ | 5/2/2000 | Μόρφου 35 | Λεμεσός | 99510430 | Andria@cytanet.com.cy | 6 |
| 1023 | Ανδριανή | Νεοκλή | Θ | 6/2/2000 | Αμμοχώστου 3 | Λεμεσός | 99233241 | Andriani@gmail.com | 7 |
| 1024 | Άγγελος | Λεωνίδα | A | 7/2/2000 | Ριζοκαρπάσου 919 | Λεμεσός | 99881071 | Angel@cytanet.com.cy | 5 |

Παρόλο που η Microsoft Access μας δίνει τη δυνατότητα να δημιουργήσουμε πίνακες χωρίς πρωτεύον κλειδί, αυτό γίνεται σε πολύ σπάνιες περιπτώσεις. Η Microsoft Access, όπως και άλλα παρόμοια ΣΔΒΔ, χρησιμοποιεί τα πεδία πρωτεύοντος κλειδιού για να συνδέσει τα

δεδομένα από έναν πίνακα με αντίστοιχα σε άλλον. Κατά συνέπεια, επειδή τα δεδομένα ενός πίνακα δεν είναι απομονωμένα από τα δεδομένα των άλλων πινάκων σε μια βάση δεδομένων, αλλά συνδέονται μεταξύ τους, σε κάθε πίνακα στη βάση δεδομένων πρέπει καθορίζεται το πρωτεύον κλειδί.

8. Πώς καθορίζουμε το Πρωτεύον Κλειδί

(1) Επιλέγουμε τον πίνακα και κάνουμε **διπλό κλικ** πάνω του για να τον ανοίξουμε πρώτα σε προβολή φύλλου δεδομένων (Datasheet View).

(2) Από την καρτέλα Home και την ομάδα Views επιλέγουμε το εικονίδιο Design View για να ανοίξουμε τον πίνακά μας σε προβολή σχεδίασης.

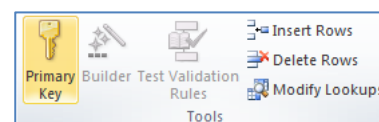


ή

Επιλέγουμε τον πίνακα και κάνουμε δεξί κλικ πάνω του και επιλέγουμε την εντολή **Design View** για να τον ανοίξει σε προβολή σχεδίασης.

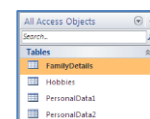
(3) Στη συνέχεια, επιλέγουμε το **κατάλληλο πεδίο ή πεδία** που θέλουμε να ορίσουμε ως πρωτεύον κλειδί του πίνακά μας.

(4) Από την καρτέλα Table Tools→**Design** και την ομάδα Tools επιλέγουμε το εικονίδιο Primary Key για να ορίσουμε με αυτό τον τρόπο το πρωτεύον κλειδί στον πίνακά μας που θέλουμε να χρησιμοποιήσουμε.



9. Διαγραφή Πίνακα (Delete Table)

(1) Από την περιοχή όλων των αντικειμένων της εφαρμογής (All Access Objects) κάτω από τον τίτλο **Tables**, όπου εμφανίζονται όλοι οι πίνακες που έχουν δημιουργηθεί στη βάση δεδομένων μας, επιλέγουμε τον πίνακα που θέλουμε να διαγράψουμε.



(2) Ακολούθως, κάνουμε δεξί κλικ και επιλέγουμε την εντολή **Delete** για να τον διαγράψουμε από τη βάση δεδομένων μας.

ή

Από την καρτέλα Home και την ομάδα Records επιλέγουμε το εικονίδιο Delete.



ή

Μπορούμε επίσης να χρησιμοποιήσουμε το πλήκτρο **Delete**.

10. Προβολές για Πίνακα (Views)


10.1 Προβολή Σχεδίασης (Design View)

(1) Από την περιοχή όλων των αντικειμένων της εφαρμογής (All Access Objects) κάτω από τον τίτλο **Tables**, όπου εμφανίζονται όλοι οι πίνακες που έχουν δημιουργηθεί στη βάση δεδομένων μας, επιλέγουμε τον πίνακα που θέλουμε και κάνουμε **διπλό κλικ** πάνω του για να τον ανοίξουμε πρώτα σε προβολή φύλλου δεδομένων (Datasheet View).

- (2) Ακολουθως, από την καρτέλα Home και την ομάδα Views επιλέγουμε το εικονίδιο Design View για να ανοίξουμε τον πίνακά μας σε προβολή σχεδίασης.

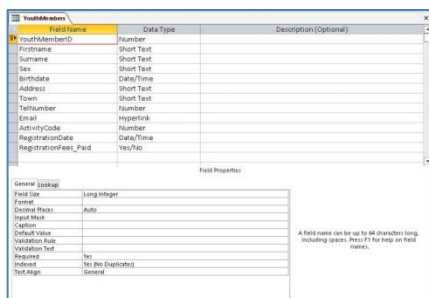


ή

Κάνουμε κλικ στο κουμπί  από τη γραμμή κατάστασης στην κάτω δεξιά γωνιά της οθόνης μας.

ή

Από την περιοχή όλων των αντικειμένων της εφαρμογής (All Access Objects) κάτω από τον τίτλο **Tables**, όπου εμφανίζονται όλοι οι πίνακες που έχουν δημιουργηθεί στη βάση δεδομένων μας, επιλέγουμε τον πίνακα που θέλουμε και κάνουμε **δεξί κλικ** πάνω του και επιλέγουμε την εντολή **Design View** για να τον ανοίξει σε προβολή σχεδίασης.

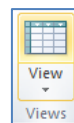


10.2 Προβολή Φύλλου Δεδομένων (Datasheet View)


- (1) Από την περιοχή όλων των αντικειμένων της εφαρμογής (All Access Objects) κάτω από τον τίτλο **Tables**, όπου εμφανίζονται όλοι οι πίνακες που έχουν δημιουργηθεί στη βάση δεδομένων μας, επιλέγουμε τον πίνακα που θέλουμε και κάνουμε **διπλό κλικ** πάνω του για να τον ανοίξουμε σε προβολή φύλλου δεδομένων (Datasheet View).

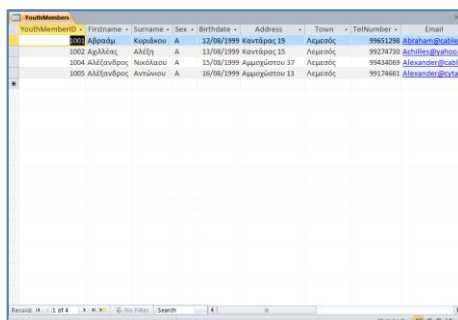
ή

Από την καρτέλα Home και την ομάδα Views επιλέγουμε το εικονίδιο Datasheet View για να ανοίξουμε τον πίνακά μας σε προβολή φύλλου δεδομένων.



ή

Κάνουμε κλικ στο κουμπί  από τη γραμμή κατάστασης στην κάτω δεξιά γωνιά της οθόνης μας.



Υπόμνημα

| Περιγραφή (Description) | Καρτέλα (Tab) | Εικονίδιο (Icon) | Ομάδα (Group) | Παράθυρο Διαλόγου (Dialog Box) | Πλήκτρα (Keys) |
|------------------------------------------------------------------------------|-------------------------|----------------------------|-------------------------|------------------------------------------|--------------------------|
| Δημιουργία Πίνακα με τη χρήση της Προβολής Σχεδίασης (Table Design) | Create | | | | |
| Δημιουργία Πίνακα με τη χρήση της Προβολής Φύλλου Δεδομένων (Table) | Create | | | | |
| Προβολή Σχεδίασης (Design View) | Home | | | | |
| Προβολή Φύλλου Δεδομένων (Datasheet View) | Home | | | | |
| Δημιουργία Πρωτεύοντος Κλειδιού (Primary Key) | Table Tools Design | | | | |
| Διαγραφή Πίνακα (Delete Table) | Home | | | | |

Γ6.4 Εγγραφές


Τι θα μάθουμε σήμερα:

- ◆ Να καταχωρούμε, να επεξεργαζόμαστε και να διαγράφουμε εγγραφές σε πίνακα
- ◆ Να μετακινούμαστε μεταξύ των εγγραφών σε πίνακα.

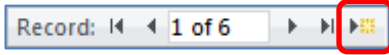
1. Εισαγωγή, Επεξεργασία, Διαγραφή και Μετακίνηση μεταξύ των εγγραφών σε ένα πίνακα

Μετά τη δημιουργία ενός πίνακα, μπορούμε να εισαγάγουμε, να επεξεργαστούμε και να διαγράψουμε εγγραφές σε έναν πίνακα απευθείας ή να μετακινηθούμε μεταξύ των εγγραφών του πίνακα, κάνοντας **διπλό κλικ** πάνω του και ανοίγοντάς τον σε προβολή φύλλου δεδομένων (**Datasheet View**).

1.1 Εισαγωγή Εγγραφής σε έναν πίνακα (New Record)

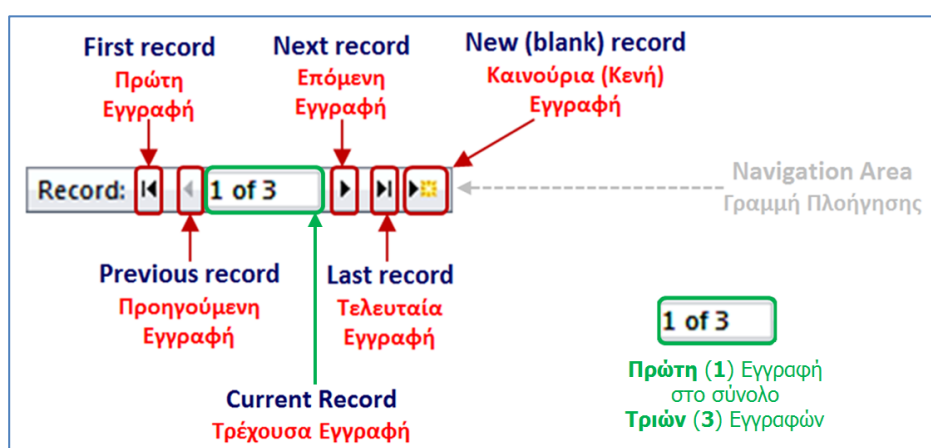
- (1) Κάνουμε δεξί κλικ σε μια οποιαδήποτε γραμμή (εγγραφή) του πίνακά μας.
- (2) Ακολούθως, κάνουμε κλικ στο κουμπι  και η νέα μας εγγραφή **εισάγεται** τελευταία στον πίνακά μας.

ή

Από τη γραμμή πλοήγησης (**Navigation Area**)  κάνουμε κλικ στο εικονίδιο **New (blank) record** και καταχωρούμε τη νέα μας εγγραφή ως τελευταία στον πίνακά μας.

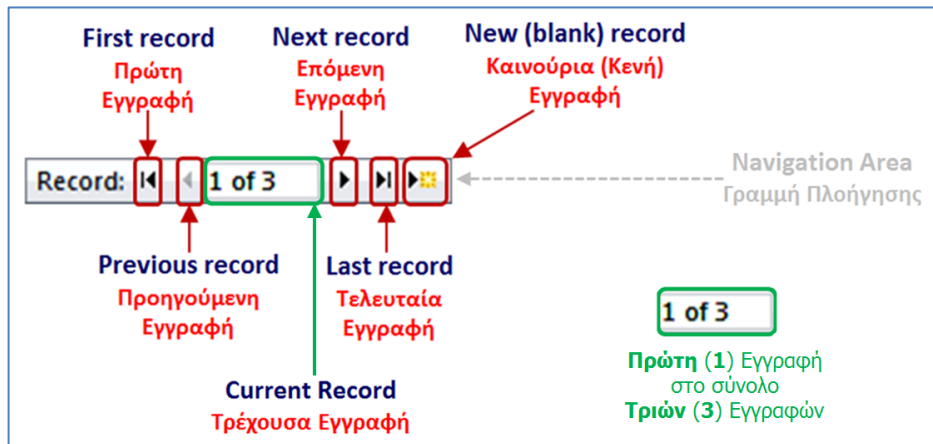
1.2 Μετακίνηση μεταξύ των εγγραφών σε έναν πίνακα

- (1) Μπορούμε να μετακινηθούμε μεταξύ των εγγραφών ενός πίνακα, κάνοντας κλικ σε ένα από τα εικονίδια της γραμμής πλοήγησης πιο κάτω:




1.3 Επεξεργασία Εγγραφής σε έναν πίνακα

- (1) Μπορούμε να πάμε σε οποιαδήποτε εγγραφή του πίνακά μας και να την επεξεργαστούμε (*αλλαγή, διόρθωση κ.λπ.*), κάνοντας κλικ σε ένα από τα εικονίδια της γραμμής πλοήγησης πιο κάτω:



1.4 Διαγραφή Εγγραφής σε έναν πίνακα (Delete Record)


- (1) Επιλέγουμε την **εγγραφή** που θέλουμε να διαγράψουμε από τον πίνακά μας.
- (2) Από την καρτέλα **Home** και την ομάδα **Records** επιλέγουμε το εικονίδιο **Delete**.
- (3) Ακολουθως, κάνουμε κλικ στο κουμπί  και η εγγραφή μας **διαγράφεται** από τον πίνακά μας.

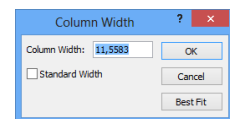


2. Μορφοποίηση Πίνακα


Μετά τη δημιουργία ενός πίνακα, μπορούμε να τον μορφοποιήσουμε, κάνοντας **διπλό κλικ** πάνω του και ανοίγοντάς τον σε προβολή φύλλου δεδομένων (**Datasheet View**).

2.1 Αλλαγή Πλάτους Στήλης-Πεδίου (Field Width)

- (1) Κάνουμε δεξί κλικ στη στήλη (πεδίο) που θέλουμε να αυξομειώσουμε το πλάτος της.
- (2) Ακολουθως, κάνουμε κλικ στο κουμπί  και στο παράθυρο που εμφανίζεται πληκτρολογούμε το **επιθυμητό πλάτος** που θέλουμε να χρησιμοποιήσουμε.



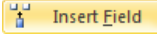
ή

Κάνουμε κλικ στο **δεξιό όριο** μιας στήλης και όταν ο δείκτης του ποντικιού πάρει τη μορφή γραμμής με τα δύο αντίθετα βέλη, **σύρουμε ανάλογα** προς τα αριστερά ή προς τα δεξιά για να αλλάξουμε το πλάτος του. 

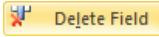
ή

Κάνουμε **διπλό κλικ** στο **δεξιό όριο** μιας στήλης και τα περιεχόμενα της στήλης προσαρμόζονται αυτόματα, ανάλογα με το περιεχόμενό της.

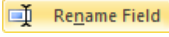
2.2 Εισαγωγή Πεδίου (Insert Field)

- (1) Κάνουμε δεξί κλικ σε μια στήλη (**πεδίο**) που πριν από αυτή θέλουμε να εισαγάγουμε ένα νέο πεδίο στον πίνακά μας.
- (2) Ακολούθως, κάνουμε κλικ στο κουμπί  και το νέο μας πεδίο **εισάγεται** στον πίνακά μας.
- (3) Στη συνέχεια, δίνουμε μια κατάλληλη περιγραφή (ονομασία) στο νέο πεδίο, διαγράφοντας αυτή που έδωσε το σύστημα.

2.3 Διαγραφή Πεδίου (Delete Field)

- (1) Κάνουμε δεξί κλικ στη στήλη (**πεδίο**) που θέλουμε να διαγράψουμε από τον πίνακά μας.
- (2) Ακολούθως, κάνουμε κλικ στο κουμπί  και το πεδίο μας **διαγράφεται** από τον πίνακά μας.

2.4 Μετονομασία Πεδίου (Rename Field)

- (1) Κάνουμε δεξί κλικ στη στήλη (**πεδίο**) του οποίου το όνομα θέλουμε να αλλάξουμε στον πίνακά μας.
- (2) Ακολούθως, κάνουμε κλικ στο κουμπί  και το πεδίο μας **μετονομάζεται** στον πίνακά μας.

3. Ταξινόμηση Δεδομένων (Sort)

Σε μια ηλεκτρονική βάση δεδομένων υπάρχει η δυνατότητα χρήσης της λειτουργίας της ταξινόμησης δεδομένων για να ρυθμίσουμε τη σειρά με την οποία θα εμφανίζονται οι διάφορες εγγραφές.

Με την κατάλληλη **ταξινόμηση δεδομένων** σε έναν πίνακα, μια φόρμα, ένα ερώτημα ή μια έκθεση, διευκολύνεται η κατανόηση των δεδομένων μας και επιτυγχάνεται η γρήγορη και πιο ασφαλής εξαγωγή συμπερασμάτων από μια βάση με μεγάλο όγκο δεδομένων. Επίσης, επιτυγχάνεται ο γρήγορος εντοπισμός συγκεκριμένων δεδομένων που θέλουμε.

Για **παράδειγμα**, ένας κατάλογος προϊόντων που περιλαμβάνει τα προϊόντα με αλφαβητική σειρά ή με βάση την τιμή μονάδας είναι πολύ ευκολότερος στη χρήση του από ότι ένας κατάλογος που περιλαμβάνει τα στοιχεία με τυχαία σειρά.

Γενικά, μπορούμε να ταξινομήσουμε τα δεδομένα με:

- **Αλφαβητική σειρά** ταξινομώντας τα, αρχίζοντας από το
 - **A** έως το **Ω** (*Αύξουσα/Ascending*) ή
 - **Ω** έως το **A** (*Φθίνουσα/Descending*).
- **Αριθμητική σειρά** ταξινομώντας τα, αρχίζοντας από το
 - **μικρότερο** στον **μεγαλύτερο αριθμό** (*Αύξουσα/Ascending*) ή
 - **μεγαλύτερο** στον **μικρότερο αριθμό** (*Φθίνουσα/Descending*).

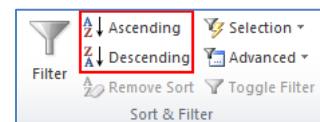
3.1 Ταξινόμηση (Sort) σε Πίνακα, Φόρμα, Ερώτημα και Έκθεση

(1) Ανάλογα:

- Εάν θέλουμε να ταξινομήσουμε πίνακα, τότε επιλέγουμε τον πίνακα και κάνουμε **διπλό κλικ** πάνω του για να τον ανοίξουμε πρώτα σε προβολή φύλλου δεδομένων (Datasheet View).
- Εάν θέλουμε να ταξινομήσουμε φόρμα, τότε επιλέγουμε τη φόρμα και κάνουμε **διπλό κλικ** πάνω της για να την ανοίξουμε πρώτα σε προβολή φόρμας (Form View).
- Εάν θέλουμε να ταξινομήσουμε ερώτημα, τότε επιλέγουμε το ερώτημα και κάνουμε **διπλό κλικ** πάνω του για να το ανοίξουμε πρώτα σε προβολή φύλλου δεδομένων (Datasheet View).
- Εάν θέλουμε να ταξινομήσουμε έκθεση, τότε επιλέγουμε την έκθεση και κάνουμε **δεξί κλικ** πάνω της και επιλέγουμε την εντολή **Layout View** για να την ανοίξουμε σε προβολή διάταξης έκθεσης (Layout View).

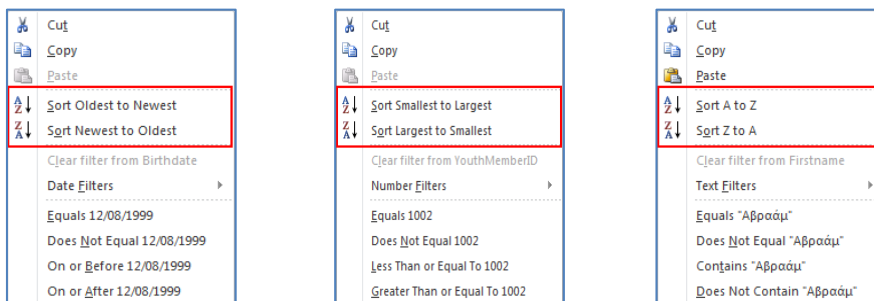
(2) Στη συνέχεια, κάνουμε κλικ στο **πεδίο** το οποίο θέλουμε να ταξινομήσουμε.

(3) Ακολούθως, από την καρτέλα Home και την ομάδα Sort & Filter επιλέγουμε το εικονίδιο Ascending ή το εικονίδιο Descending για να **ταξινομήσουμε** το πεδίο μας είτε σε αύξουσα σειρά είτε σε φθίνουσα σειρά.



ή

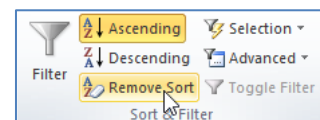
Κάνουμε δεξί κλικ στο **πεδίο** το οποίο θέλουμε να ταξινομήσουμε.



Ακολούθως, κάνουμε κλικ σε ένα από τα πιο πάνω κουμπιά για να **ταξινομήσουμε** το πεδίο μας είτε σε αύξουσα σειρά είτε σε φθίνουσα σειρά.

3.2 Κατάργηση Ταξινόμησης σε Πίνακα, Φόρμα, Ερώτημα και Έκθεση (Remove Sort)

(1) Από την καρτέλα Home και την ομάδα Sort & Filter επιλέγουμε το εικονίδιο Remove Sort για να καταργήσουμε την ταξινόμηση που χρησιμοποιήσαμε προηγουμένως στο πεδίο μας.



4. Χρήση της Εντολής για την Αναζήτηση Πληροφοριών (Find)

Η εντολή αναζήτησης πληροφοριών (Find) χρησιμοποιείται για τη **γρήγορη εύρεση πληροφοριών** από έναν πίνακα ή από μια φόρμα. Αυτό μπορούμε να το πετύχουμε είτε

όταν βρισκόμαστε σε προβολή φύλλου δεδομένων (Datasheet View) του πίνακα, είτε σε προβολή φόρμας (Form View) της φόρμας. Να σημειώσουμε ότι η εντολή αναζήτησης πληροφοριών είναι ένας προσωρινός τρόπος εύρεσης πληροφοριών και για αυτό τον λόγο **ΔΕΝ μπορεί να αποθηκευτεί** για μελλοντική χρήση.

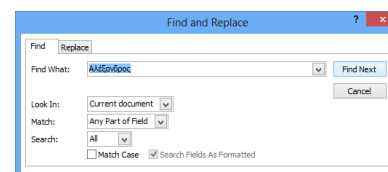
- (1) Επιλέγουμε τον πίνακα ή τη φόρμα και κάνουμε **διπλό κλικ** πάνω του/της για να τον/την ανοίξουμε πρώτα σε προβολή φύλλου δεδομένων (Datasheet View) / προβολή φόρμας (Form View).
- (2) Ακολουθώντας, από την καρτέλα Home και την ομάδα Find επιλέγουμε το εικονίδιο Find.



ή

Μπορούμε επίσης να χρησιμοποιήσουμε τον συνδυασμό πλήκτρων **CTRL + F**.

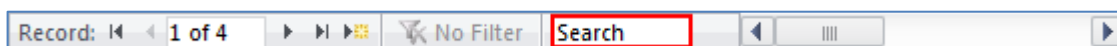
- (3) Από το παράθυρο που εμφανίζεται, στο πλαίσιο του **Find What:** πληκτρολογούμε τη λέξη ή τον αριθμό ή την ημερομηνία που αναζητούμε να βρούμε στις εγγραφές μας.



- (4) Από τη λίστα του **Look In:** επιλέγουμε το Current document και από τη λίστα του **Match:** επιλέγουμε το Any Part of Field.
- (5) Στη συνέχεια, κάνουμε κλικ στο κουμπί **Find Next** για να μας πάει με αυτό τον τρόπο στην **πρώτη εγγραφή** που θα περιέχει αυτά τα δεδομένα από τις εγγραφές μας.
- (6) Τέλος, κάνοντας ξανά κλικ στο κουμπί **Find Next** μας πάει στην **επόμενη εγγραφή** που θα περιέχει αυτά τα δεδομένα από τις εγγραφές μας.

ή

Από τη γραμμή πλοήγησης (Navigation Area) στο πλαίσιο της αναζήτησης (Search) πληκτρολογούμε τη λέξη ή τον αριθμό ή την ημερομηνία που αναζητούμε να βρούμε στις εγγραφές μας και μας πάει με αυτό τον τρόπο στην **πρώτη εγγραφή** που θα περιέχει αυτά τα δεδομένα από τις εγγραφές μας.




Ακολουθώντας, πατώντας το πλήκτρο **ENTER** μας πάει στην **επόμενη εγγραφή** που θα περιέχει αυτά τα δεδομένα από τις εγγραφές μας μέχρι να φτάσει και στην τελευταία εγγραφή.

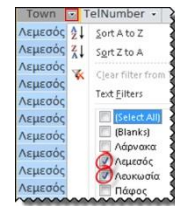
5. Χρήση Φίλτρων για την Αναζήτηση Πληροφοριών (Filter)

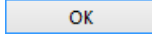
Ένας άλλος τρόπος **γρήγορης εύρεσης πληροφοριών** από έναν πίνακα ή από μια φόρμα είναι τα φίλτρα (Filter). Αυτό μπορούμε να το πετύχουμε είτε όταν βρισκόμαστε σε προβολή φύλλου δεδομένων (Datasheet View) του πίνακα, είτε σε προβολή φόρμας (Form View) της φόρμας. Να σημειώσουμε ότι τα φίλτρα είναι και αυτός ένας προσωρινός τρόπος εύρεσης πληροφοριών και για αυτό τον λόγο **ΔΕΝ μπορούν να αποθηκευτούν** για μελλοντική χρήση.

5.1 Δημιουργία Φίλτρων

(1) Επιλέγουμε τον πίνακα ή τη φόρμα και κάνουμε **διπλό κλικ** πάνω του/της για να τον/τη ανοίξουμε πρώτα σε προβολή φύλλου δεδομένων (Datasheet View) / προβολή φόρμας (Form View).

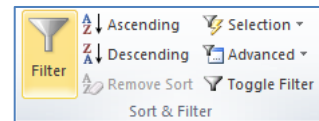
(2) Ακολούθως, κάνουμε κλικ στο **βελάκι**  δίπλα από το πεδίο το οποίο θέλουμε να φιλτράρουμε, (π.χ. *εάν θέλουμε να βρούμε από το πεδίο **Town** τις εγγραφές τις οποίες έχουν για πόλη τη Λεμεσό και τη Λευκωσία*) και επιλέγουμε τις δυο πόλεις.



(3) Τέλος, κάνουμε κλικ στο κουμπι  για να εφαρμόσουμε το φίλτρο.

ή

Από την καρτέλα Home και την ομάδα Sort & Filter επιλέγουμε το εικονίδιο Filter για να μπορέσουμε να φιλτράρουμε με τον ίδιο τρόπο όπως και πιο πάνω.

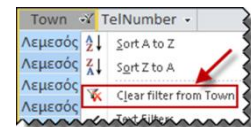


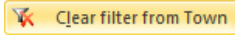
(*Σημείωση: Θα εμφανιστεί το εικονίδιο του φίλτρου δίπλα από το πεδίο.*) 

5.2 Αφαίρεση Φίλτρων (Clear Filter)

(1) Επιλέγουμε τον πίνακα ή τη φόρμα και κάνουμε **διπλό κλικ** πάνω του/της για να τον/τη ανοίξουμε πρώτα σε προβολή φύλλου δεδομένων (Datasheet View) / προβολή φόρμας (Form View).

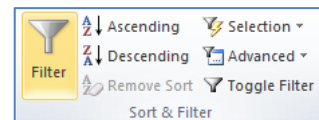
(2) Ακολούθως, κάνουμε κλικ στο **βελάκι**  δίπλα από το πεδίο του οποίου θέλουμε να αφαιρέσουμε το φιλτράρισμα.



(3) Τέλος, επιλέγουμε το  για να αφαιρέσουμε το φίλτρο.

ή

Από την καρτέλα Home και την ομάδα Sort & Filter επιλέγουμε το εικονίδιο Filter για να μπορέσουμε να αφαιρέσουμε το φιλτράρισμα με τον ίδιο τρόπο όπως και πιο πάνω.



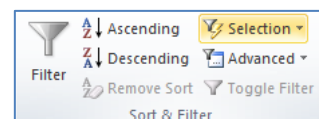
Σημείωση: Θα εμφανιστεί το εικονίδιο του φίλτρου δίπλα από το πεδίο. 

5.3 Φίλτρα κατ' Επιλογή (Selection Filter)

(1) Επιλέγουμε τον πίνακα ή τη φόρμα και κάνουμε **διπλό κλικ** πάνω του/της για να τον/την ανοίξουμε πρώτα σε προβολή φύλλου δεδομένων (Datasheet View) / προβολή φόρμας (Form View).

(2) Ακολούθως, κάνουμε κλικ σε ένα **πεδίο** ή κάνουμε διπλό κλικ σε **μέρος του πεδίου** που θέλουμε να φιλτράρουμε.

(3) Ακολούθως, από την καρτέλα Home και την ομάδα Sort & Filter επιλέγουμε το εικονίδιο Selection και επιλέγουμε στη συνέχεια, ένα από τα πιο κάτω ανάλογα με



την περίπτωση:

Για κάθε είδος πεδίου που επιλέγουμε για φιλτράρισμα υπάρχουν και τα ανάλογα φίλτρα όπως το **TEXT Filters**, **NUMBER Filters** και **DATE Filters**.

Τα φίλτρα κατ' επιλογή εφαρμόζονται σε ένα πεδίο και έχουν τη μορφή:

Κριτήριο τιμή_σύγκρισης

Για παράδειγμα, έστω ότι θέλουμε να επιλέξουμε (να φιλτράρουμε) μόνο τις εγγραφές από τον πίνακα των μελών του κέντρου νεότητας (YouthMembers) για τις οποίες το επίθετο (Surname) τελειώνει σε «ου». Για να το πετύχουμε, στο πεδίο Surname προσθέτουμε φίλτρο (Home ⇒ Sort & Filter) επιλέγοντας το κριτήριο **Ends With** και την τιμή σύγκρισης «**ου**».

Ο πίνακας στην επόμενη σελίδα παρουσιάζει τα κριτήρια που μπορούν να χρησιμοποιηθούν για κάθε τύπο δεδομένων πεδίου.

| Κριτήριο | Επιλέγει μόνο τις εγγραφές οι οποίες στο συγκεκριμένο πεδίο έχουν δεδομένα που ... | Εφαρμόζεται σε πεδία τύπου |
|---------------------------------|------------------------------------------------------------------------------------|----------------------------|
| Equals | είναι ίσα με την τιμή σύγκρισης | Text, Number, Date |
| Does Not Equal | δεν είναι ίσα με με την τιμή σύγκρισης | Text, Number, Date |
| Begins With | αρχίζουν με την τιμή σύγκρισης | Text |
| Does Not Begin With | δεν αρχίζουν με την τιμή σύγκρισης | Text |
| Contains | έχουν μέσα τους την τιμή σύγκρισης | Text |
| Does Not Contain | δεν έχουν μέσα τους την τιμή σύγκρισης | Text |
| Ends With | τελειώνουν με την τιμή σύγκρισης | Text |
| Does Not End With | δεν τελειώνουν με την τιμή σύγκρισης | Text |
| Less Than or Equal To | είναι μικρότερα από ή ίσα με την τιμή σύγκρισης | Number |
| Greater Than or Equal To | είναι μεγαλύτερα από ή ίσα με την τιμή σύγκρισης | Number |
| Between | είναι ανάμεσα στις τιμές σύγκρισης | Date, Number |

Πιο κάτω δίνονται παραδείγματα εφαρμογής των φίλτρων με βάση τον πίνακα **YouthMembers**.

| YouthMembers | | | | | | | |
|---------------|------------|------------|-----|-----------|---------------|----------|-----------|
| YouthMemberID | Firstname | Surname | Sex | Birthdate | Address | Town | TelNumber |
| 1002 | Αβραάμ | Κυριάκου | A | 8/12/1999 | Καντάρας 19 | Λεμεσός | 99651298 |
| 1015 | Αχιλλέας | Αλέξη | A | 8/13/1999 | Καντάρας 15 | Λεμεσός | 99274730 |
| 1018 | Ελένη | Άγγελου | Θ | 2/5/1998 | Καντάρας 11 | Λευκωσία | 99898162 |
| 1024 | Ελεάνα | Πολύκαρπου | Θ | 2/7/1998 | Κυρηνείας 37 | Λευκωσία | 99610499 |
| 1027 | Αλέξανδρος | Νικόλαου | A | 8/15/1999 | Αμμοχώστου 37 | Λεμεσός | 99434069 |
| 1052 | Άννα | Νίκου | Θ | 2/9/2000 | Καντάρας 25 | Λεμεσός | 99216150 |
| 1087 | Χρήστος | Γκέλα | A | 2/29/2000 | Καντάρας 7 | Λευκωσία | 99521594 |
| 1089 | Χρίστος | Δήμου | A | 3/13/2000 | Αμμοχώστου 17 | Λευκωσία | 99551229 |
| 1091 | Ελένη | Γιώργιου | Θ | 2/6/1998 | Μόρφου 6 | Λεμεσός | 99780312 |
| 1099 | Άντρεα | Κυριάκου | Θ | 8/20/1999 | Κυρηνείας 7 | Λεμεσός | 99786239 |

| Πεδίο | Κριτήριο | Τιμή | Εμφανίζει μόνο τις εγγραφές με YouthMemeberID |
|---------------|---------------------------------|-------------------|---------------------------------------------------------|
| Firstname | Equals To | «Ελένη» | 1018 και 1091 |
| Birthdate | Does Not Equal To | «8/12/1999» | 1015, 1018, 1024, 1027, 1052, 1087, 1089, 1091 και 1099 |
| Surname | Begins With | «Γ» | 1087, και 1091 |
| Address | Does Not Begin With | «Κ» | 1027, 1089 και 1091 |
| Town | Contains | «ΕΥΚΩ» | 1018, 1024, 1087 και 1089 |
| Town | Does Not Contain | «ΕΜ» | 1018, 1024, 1087 και 1089 |
| Surname | Ends With | «ΟΥ» | 1002, 1018, 1024, 1027, 1052,1089 και 1091 |
| Surname | Does Not End With | «ΟΥ» | 1015 και 1087 |
| YouthMemberID | Less Than or Equal To | «1024» | 1002, 1015, 1018 και 1024 |
| Birthdate | Greater Than or Equal To | «9/2/2000» | 1052, 1087 και 1089 |
| YouthMemberID | Between | «1002» και «1027» | 1002, 1015, 1018, 1024 και 1027 |

| Υπόμνημα | | | | | |
|---------------------------------------------------------------------------------|-------------------------|----------------------------|-------------------------|------------------------------------------|--------------------------|
| Περιγραφή (Description) | Καρτέλα (Tab) | Εικονίδιο (Icon) | Ομάδα (Group) | Παράθυρο Διαλόγου (Dialog Box) | Πλήκτρα (Keys) |
| Εισαγωγή Εγγραφής (New Record) | Home | | | | |
| Γραμμή Πλοήγησης (Navigation Area) | | | | | |
| Πρώτη Εγγραφή (First record) | | | | | |
| Προηγούμενη Εγγραφή (Previous record) | | | | | |
| Τρέχουσα Εγγραφή (Current Record) | 1 of 6 | | | | |
| Επόμενη Εγγραφή (Next record) | | | | | |
| Τελευταία Εγγραφή (Last record) | | | | | |
| Νέα Κενή Εγγραφή (New (blank) record) | | | | | |
| Μετακίνηση του δρομέα στην ίδια την εγγραφή (Επόμενο και Προηγούμενο) | | | | | |
| Μετακίνηση του δρομέα στην ίδια την εγγραφή (Επόμενο) | | | | | |
| Μετακίνηση του δρομέα στην ίδια την εγγραφή (Προηγούμενο) | | | | | |
| Διαγραφή Εγγραφής (Delete Record) | Home | | | | |

Γ6.5 Σχέσεις Πινάκων

Τι θα μάθουμε σήμερα:

- ◊ Να περιγράψουμε την ανάγκη χρήσης και το σκοπό των σχέσεων μεταξύ πινάκων σε μια βάση δεδομένων (*πλεονεκτήματα*)
- ◊ Να διακρίνουμε τα είδη των σχέσεων πινάκων «ένα προς ένα» και «ένα προς πολλά».

1. Εισαγωγή

Όπως ήδη γνωρίζουμε μια βάση δεδομένων είναι μια **οργανωμένη συλλογή** από συσχετιζόμενα δεδομένα. Πιο συγκεκριμένα, σε μια Σχεσιακή Βάση Δεδομένων τα δεδομένα είναι οργανωμένα σε πίνακες οργανωμένους σε πεδία. Κάθε πίνακας αποθηκεύει δεδομένα για ένα συγκεκριμένο θέμα (αντικείμενο, πρόσωπο ή έννοια), οργανωμένα σε εγγραφές. Συνήθως μια βάση δεδομένων περιλαμβάνει περισσότερους από έναν πίνακες.

Παράδειγμα:

Η βάση δεδομένων για ένα σχολείο περιλαμβάνει τον πίνακα με τα δεδομένα των μαθητών, τον πίνακα με τα δεδομένα των καθηγητών, τον πίνακα με τα δεδομένα των μαθημάτων, τον πίνακα με τα δεδομένα των τμημάτων τον πίνακα με τα δεδομένα των απουσιών και πολλούς άλλους.

Σε μια (Σχεσιακή) βάση δεδομένων, οι διάφοροι πίνακες μπορούν να σχετίζονται και αυτοί μεταξύ τους με ειδικές σχέσεις οι οποίες ονομάζονται **Relationships**. Αυτές οι σχέσεις είναι που συνδέουν τους πίνακες μεταξύ τους και δημιουργούν την ανάγκη να ανήκουν στην ίδια βάση δεδομένων. Ο κάθε πίνακας σε μια βάση δεδομένων πιθανό να σχετίζεται με έναν ή περισσότερους άλλους πίνακες. Η κάθε σχέση εμφανίζεται συνήθως ως μια λογική σύνδεση μεταξύ δύο πινάκων και υποδηλώνεται με την παρουσία και στους δύο πίνακες πεδίων με κοινά δεδομένα.

Παράδειγμα:

Στη βάση δεδομένων του σχολείου, ο πίνακας των μαθητών σχετίζεται με τον πίνακα των τμημάτων (στον πίνακα των μαθητών, μαζί με τα προσωπικά τους στοιχεία είναι καταχωρημένο και το τμήμα στο οποίο ανήκουν). Επίσης, ο πίνακας των καθηγητών σχετίζεται με τα τμήματα στα οποία είναι υπεύθυνοι (στον πίνακα των τμημάτων συμπεριλαμβάνεται ο αριθμός αναφοράς του/της κάθε καθηγητή/τριας). Αντίστοιχα, ο πίνακας των απουσιών περιλαμβάνει τον αριθμό μητρώου του/της μαθητή/τριας που έκανε την απουσία, μαζί με την ημερομηνία και την περίοδο.

Αυτή η ύπαρξη αυτών των σχέσεων μεταξύ των πινάκων, αλλά και ο απλός τρόπος αναπαράστασής τους είναι ένα από τους κύριους λόγους που οι Σχεσιακές Βάσεις Δεδομένων έχουν επικρατήσει. Στην πραγματικότητα υπάρχει ένα στέρεο θεωρητικό υπόβαθρο για τις Σχεσιακές Βάσεις Δεδομένων, το οποίο οδήγησε στη διατύπωση συστηματικών μεθόδων στην υλοποίησή τους. Αυτό μας δίνει «συνταγές» για τον καλύτερο τρόπο διαχωρισμού των δεδομένων σε πίνακες και τον καλύτερο τρόπο διατύπωσης των σχέσεων μεταξύ τους, τα οποία οδηγούν σε σημαντικά πλεονεκτήματα.

2. Πλεονεκτήματα που προκύπτουν από τις Σχέσεις μεταξύ των Πινάκων

Η δημιουργία αυτών των σχέσεων ανάμεσα σε πίνακες μιας βάσης δεδομένων, μάς επιτρέπει να αντλήσουμε τα σχετιζόμενα στοιχεία που ανήκουν στους διάφορους πίνακες με διάφορους τρόπους ανάλογα με τις ανάγκες που προκύπτουν (συνήθως με ένα κατάλληλο ερώτημα, query).

Η ύπαρξη σχετιζόμενων πινάκων έχει ως αποτέλεσμα να έχουμε τα πιο κάτω πλεονεκτήματα:

- Δεν χρειάζεται να καταχωρούμε τα ίδια δεδομένα πολλές φορές. Καταχωρούνται από μία φορά στον κατάλληλο πίνακα και τα ανακτούμε όταν χρειάζεται μέσω των σχέσεων των πινάκων.
- Γίνεται ευκολότερη η διαχείριση των δεδομένων (*ενημέρωση, διαγραφή κ.λπ.*), αφού το κάθε δεδομένο καταχωρείται σε έναν μόνο πίνακα.
- Ο συνολικός όγκος των δεδομένων στη βάση δεδομένων είναι όσο μικρότερος γίνεται, αφού δεν υπάρχει αχρείαστη επανάληψη δεδομένων.
- Μειώνεται το περιθώριο σφαλμάτων αφού σε κάθε αλλαγή ή διαγραφή κάποιας εγγραφής επηρεάζεται συνήθως μόνο ένας πίνακας.
- Υπάρχει περισσότερη σαφήνεια στην οργάνωση των δεδομένων, αφού τόσο η έννοια του πίνακα, όσο και ο τρόπος συσχέτισης είναι κατανοητά, ακόμη και σε μη επαγγελματίες της Πληροφορικής και της Επιστήμης των Ηλεκτρονικών Υπολογιστών.
- Εύκολα μπορούν να προστεθούν νέοι πίνακες ή να δημιουργηθούν νέες σχέσεις μεταξύ των πινάκων σε μια βάση δεδομένων, καθώς και να συνδυαστούν οι πίνακες και οι σχέσεις τους με νέους τρόπους για την εξαγωγή νέων πληροφοριών.

3. Ο ρόλος του Πρωτεύοντος Κλειδιού στον συσχετισμό των πινάκων

Πρωτεύον Κλειδί (**Primary Key**) έχει οριστεί να είναι το **πεδίο** (ή τα πεδία) που προσδιορίζει μοναδικά κάθε εγγραφή στον πίνακα (*δηλαδή έχει ξεχωριστή, διαφορετική τιμή από οποιαδήποτε άλλη εγγραφή του πίνακα*).

Το Πρωτεύον Κλειδί χρησιμοποιείται συνήθως για τη δημιουργία της σχέσης μεταξύ **κύριου πίνακα** (*Πρωτεύων Πίνακας - Primary Table*) και **συσχετιζόμενου πίνακα** (*Δευτερεύων Πίνακας - Secondary or Related Table*), ώστε να ξέρουμε σίγουρα σε ποια εγγραφή αναφερόμαστε.

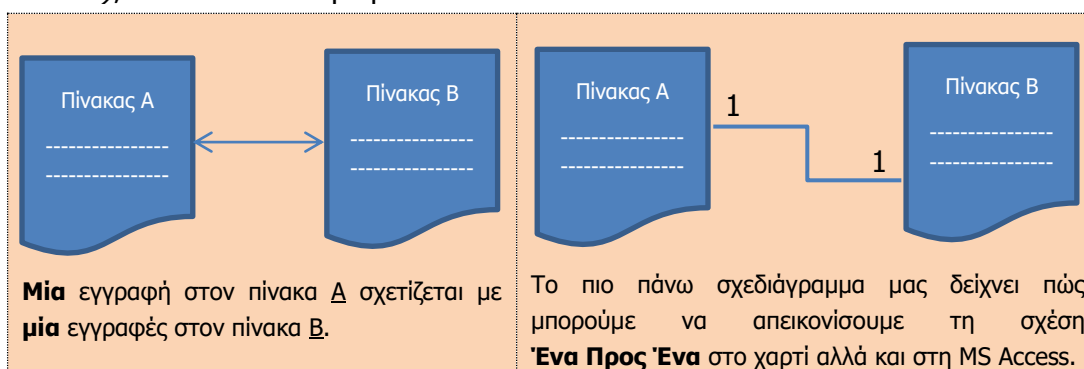
4. Είδη Ειδικών Σχέσεων (Relationships)

Τα **είδη** των ειδικών σχέσεων που υπάρχουν μεταξύ των πινάκων είναι:

- Ένα-Προς-Ένα (**One-To-One**)
- Ένα-Προς-Πολλά (**One-To-Many**)
- Πολλά-Προς-Πολλά (**Many-To-Many**) (Θα παρουσιαστεί αργότερα)

4.1 Ένα-Προς-Ένα (One-To-One, 1-1)

Κάθε εγγραφή από τον ΠΡΩΤΟ πίνακα (**Πρωτεύων Πίνακας - Primary Table**) σχετίζεται με μόνο μία εγγραφή από τον ΔΕΥΤΕΡΟ πίνακα (**Δευτερεύων Πίνακας - Secondary Table**), αλλά και αντίστροφα.



Παράδειγμα σε μια βάση δεδομένων **σχολείου**:

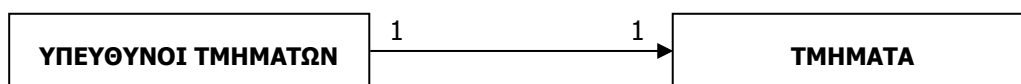
- Πίνακες:
 - **ΥΠΕΥΘΥΝΟΙ ΤΜΗΜΑΤΩΝ**
 - **ΤΜΗΜΑΤΑ**

Για κάθε **τμήμα** του σχολείου υπάρχει **ένας υπεύθυνος καθηγητής** και θέλουμε να γνωρίζουμε τα στοιχεία του. Άρα υπάρχει η ανάγκη του συσχετισμού των δύο πινάκων.

- (1) Αρχίζοντας πάντοτε από το ΕΝΑ αντικείμενο του ΠΡΩΤΟΥ πίνακα, ρωτάτε με πόσα αντικείμενα του ΔΕΥΤΕΡΟΥ πίνακα μπορεί να έχει σχέση, δηλαδή:

1 ΥΠΕΥΘΥΝΟΣ ΚΑΘΗΓΗΤΗΣ με πόσα ΤΜΗΜΑΤΑ μπορεί να έχει σχέση;

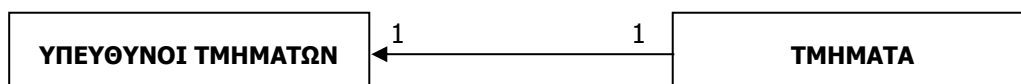
= 1 ΤΜΗΜΑ



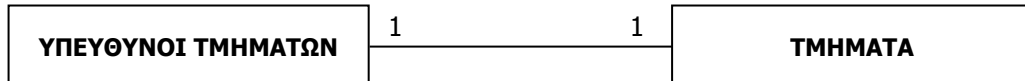
- (2) Στη συνέχεια, αρχίζοντας πάλιν από το ΕΝΑ αντικείμενο του ΔΕΥΤΕΡΟΥ πίνακα, ρωτάτε με πόσα αντικείμενα του ΠΡΩΤΟΥ πίνακα μπορεί να έχει σχέση:

1 ΤΜΗΜΑ με πόσους ΥΠΕΥΘΥΝΟΥΣ ΚΑΘΗΓΗΤΕΣ μπορεί να έχει σχέση;

= 1 ΥΠΕΥΘΥΝΟ ΚΑΘΗΓΗΤΗ



(3) Άρα το τελικό αποτέλεσμα της πιο πάνω διεργασίας είναι:



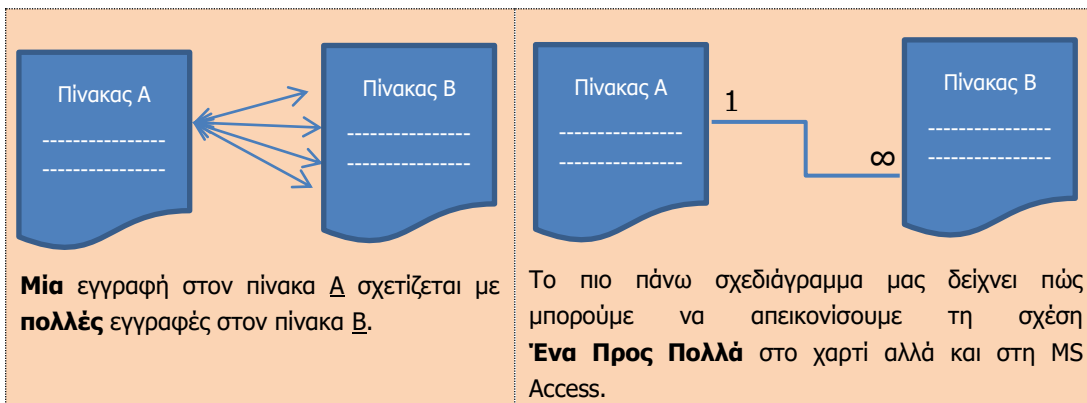
Άλλα **παραδείγματα** σχέσεων Ένα-Προς-Ένα από την καθημερινότητά μας είναι:

- η σχέση που υπάρχει μεταξύ μιας **Πόλης** και ενός **Δημάρχου**.
- η σχέση που υπάρχει μεταξύ μιας **Χώρας** και του **Προέδρου** της.

Για να μπορεί να δημιουργηθεί αυτό το είδος σχέσης μεταξύ των πινάκων θα πρέπει να υπάρχει στον έναν από τους δυο ένα πεδίο που να αντιστοιχεί στο πρωτεύον κλειδί του άλλου (π.χ. στον πίνακα των τμημάτων να υπάρχει ένα πεδίο όπου θα καταχωρούνται τιμές οι οποίες αντιστοιχούν σε αυτές που υπάρχουν στο πρωτεύον κλειδί του υπεύθυνου τμήματος, ή αντίστοιχα, στον πίνακα των υπεύθυνων τμήματος να υπάρχει ένα πεδίο όπου θα καταχωρείται τιμές οι οποίες αντιστοιχούν σε αυτές που υπάρχουν στο πρωτεύον κλειδί του τμήματος, για να συσχετίζονται οι αντίστοιχες εγγραφές).

4.2 Ένα-Προς-Πολλά (One-To-Many, 1-∞)

Κάθε εγγραφή από τον ΠΡΩΤΟ πίνακα (**Πρωτεύων Πίνακας - Primary Table**) σχετίζεται με πολλές εγγραφές από το ΔΕΥΤΕΡΟ πίνακα (**Δευτερεύων Πίνακας - Secondary Table**), αλλά κάθε εγγραφή από το ΔΕΥΤΕΡΟ πίνακα, σχετίζεται με μόνο μία εγγραφή από τον ΠΡΩΤΟ πίνακα (**Πρωτεύων Πίνακας - Primary Table**).



Παράδειγμα σε μια βάση δεδομένων **σχολείου**:

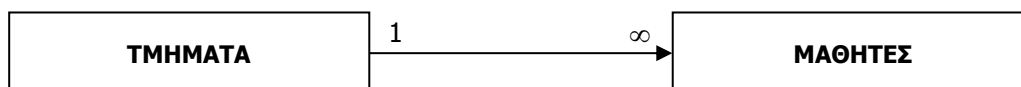
- Πίνακες:
 - **ΤΜΗΜΑΤΑ**
 - **ΜΑΘΗΤΕΣ**

Για κάθε **τμήμα** θέλουμε να γνωρίζουμε τα στοιχεία των **μαθητών** που ανήκουν σε αυτό, αλλά και για κάθε **μαθητή** τα στοιχεία του **τμήματος** που ανήκει, για αυτό υπάρχει η ανάγκη του συσχετισμού των δύο πινάκων.

(1) Αρχίζοντας πάντοτε από το ΕΝΑ αντικείμενο του ΠΡΩΤΟΥ πίνακα, ρωτάτε με πόσα αντικείμενα του ΔΕΥΤΕΡΟΥ πίνακα μπορεί να έχει σχέση, δηλαδή:

1 ΤΜΗΜΑ με πόσους ΜΑΘΗΤΕΣ μπορεί να έχει σχέση;

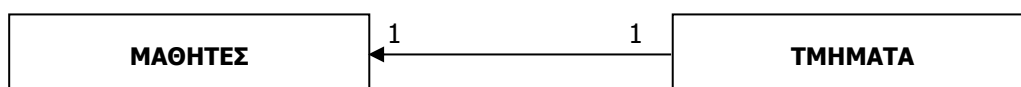
= ΠΟΛΛΟΥΣ ΜΑΘΗΤΕΣ



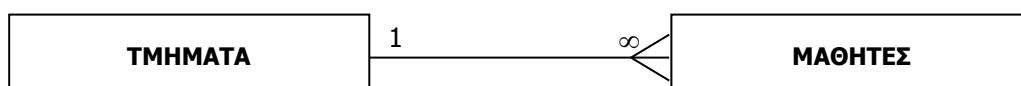
Στη συνέχεια, αρχίζοντας πάλι από το ΕΝΑ αντικείμενο του ΔΕΥΤΕΡΟΥ πίνακα, ρωτάτε με πόσα αντικείμενα του ΠΡΩΤΟΥ πίνακα μπορεί να έχει σχέση:

1 ΜΑΘΗΤΗΣ μπορεί να έχει σχέση με πόσα ΤΜΗΜΑΤΑ;

= 1 ΤΜΗΜΑ



(2) Άρα το τελικό αποτέλεσμα της πιο πάνω διεργασίας είναι:



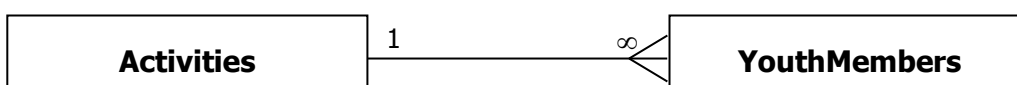
Άλλα **παράδειγματα** σχέσεων Ένα-Προς-Πολλά από την καθημερινότητά μας είναι:

- η σχέση της **Μητέρας** με τα **Παιδιά** της. Μία μητέρα μπορεί να έχει πολλά παιδιά, ένα παιδί όμως, έχει μόνο μία βιολογική μητέρα.
- η σχέση που υπάρχει μεταξύ μιας **Πόλης** και των **Κατοίκων** της. Η Πόλη έχει πολλούς Κατοίκους, ο κάθε Κάτοικος όμως ζει σε μία μόνο Πόλη.

Για να μπορεί να δημιουργηθεί αυτό το είδος σχέσης μεταξύ των πινάκων θα πρέπει να υπάρχει στον πίνακα από τη μεριά του «Πολλά» ένα πεδίο που να αντιστοιχεί στο πρωτεύον κλειδί του πίνακα από τη μεριά του «Ένα» (π.χ. στον πίνακα των μαθητών να υπάρχει ένα πεδίο όπου θα καταχωρούνται τιμές οι οποίες αντιστοιχούν σε αυτές που υπάρχουν στο πρωτεύον κλειδί του τμήματος για να συσχετίζονται οι αντίστοιχες εγγραφές).

Παράδειγμα:

Για να γίνει καλύτερα κατανοητός ο μηχανισμός των σχέσεων μεταξύ των πινάκων, θα χρησιμοποιήσουμε το παράδειγμα της βάσης δεδομένων του Κέντρου Νεότητας. Υπάρχει ο πίνακας των Μελών («YouthMembers») και ο πίνακας των Δραστηριοτήτων (Activities). Κάθε μέλος μπορεί να επιλέξει μία δραστηριότητα, ενώ κάθε δραστηριότητα περιλαμβάνει πολλά μέλη:



Πιο κάτω παρουσιάζονται οι πίνακες αυτοί, όπου σημειώνονται με διαφορετικό χρώμα οι εγγραφές από τους δύο πίνακες οι οποίες σχετίζονται μεταξύ τους. Φαίνεται ξεκάθαρα η σχέση «Ένα-Προς-Πολλά», όπως διατυπώθηκε πιο πάνω:

| YouthMemberID | Firstname | Surname | Sex | Birthdate | Town | ActivityCode |
|---------------|------------|----------|-----|-----------|----------|--------------|
| 1036 | Αντώνης | Ηλία | A | 19/2/2000 | Λεμεσός | 10 |
| 1037 | Αργύρης | Ανδρέα | A | 20/2/2000 | Λεμεσός | 10 |
| 1038 | Αρτεμισία | Αντώνιου | Θ | 21/2/2000 | Λεμεσός | 9 |
| 1039 | Βασίλης | Πέτρου | A | 22/2/2000 | Λεμεσός | 8 |
| 1040 | Βασίλειος | Κυριάκου | A | 23/2/2000 | Λεμεσός | 7 |
| 1041 | Μπόρις | Σαμψών | A | 24/2/2000 | Λευκωσία | 6 |
| 1042 | Καλλιόπη | Μάριου | Θ | 25/2/2000 | Λευκωσία | 5 |
| 1043 | Καλλιόπη | Ιωάννη | Θ | 26/2/2000 | Λευκωσία | 4 |
| 1044 | Καρολίνα | Άμερ | Θ | 27/2/2000 | Λευκωσία | 3 |
| 1045 | Χαραλαμπία | Τάσου | Θ | 28/2/2000 | Λευκωσία | 2 |
| 1046 | Χρήστος | Γκέλα | A | 29/2/2000 | Λευκωσία | 1 |
| 1047 | Χρήστος | Ηρακλή | A | 1/3/2000 | Λευκωσία | 3 |

| ActivityCode | Description | RegistrationFees |
|--------------|--------------------------|------------------|
| 1 | Ποδόσφαιρο | 100,00 € |
| 2 | Καλαθόσφαιρα | 80,00 € |
| 3 | Πετόσφαιρα | 75,00 € |
| 4 | Επιτραπέζια Αντισφαίριση | 60,00 € |
| 5 | Σκάκι | 75,00 € |
| 6 | Σκι | 100,00 € |
| 7 | Χειροσφαίριση | 60,00 € |
| 8 | Παπασία | 180,00 € |
| 9 | Τένις | 120,00 € |
| 10 | Μπάντμιντον | 90,00 € |

Βασικές Έννοιες

Πρωτεύον κλειδί (Primary Key):

Πεδίο ή συνδυασμός πεδίων ενός πίνακα του οποίου το περιεχόμενο καθορίζει με μοναδικό τρόπο την κάθε εγγραφή του πίνακα. Χρησιμοποιείται συνήθως για τη δημιουργία μιας σχέσης μεταξύ των εγγραφών δύο πινάκων, ώστε να γνωρίζουμε ακριβώς ποιες εγγραφές σχετίζονται μεταξύ τους και με ποιο τρόπο.

Ειδικές Σχέσεις πινάκων (Relationships):

Κάθε πίνακας αποτελεί ομάδα από στοιχεία που αναφέρονται σε κάποιο πρόσωπο, αντικείμενο ή έννοια. Οι εγγραφές σε έναν πίνακα σε μια βάση δεδομένων μπορεί σχετίζονται με εγγραφές σε έναν άλλο πίνακα. Αυτή η ειδική σχέση ονομάζεται **Relationship**.

Είδη Ειδικών Σχέσεων:

Ένα Προς Ένα (**One-To-One, 1-1**)
Ένα Προς Πολλά (**One-To-Many, 1-∞**)

Ένα Προς Ένα (One-To-One) (1-1):

Κάθε εγγραφή από τον ΠΡΩΤΟ πίνακα σχετίζεται με μόνο μία εγγραφή από τον ΔΕΥΤΕΡΟ πίνακα, αλλά και αντίστροφα.

Ένα Προς Πολλά (One-To-Many) (1-∞):

Κάθε εγγραφή από τον ΠΡΩΤΟ πίνακα σχετίζεται με πολλές εγγραφές από το ΔΕΥΤΕΡΟ πίνακα, αλλά κάθε εγγραφή από τον ΔΕΥΤΕΡΟ πίνακα, σχετίζεται με μόνο μία εγγραφή από τον ΠΡΩΤΟ πίνακα.

Γ6.6 Δημιουργία και Διαγραφή Σχέσεων Μεταξύ δύο Πινάκων

Τι θα μάθουμε σήμερα:

- ◆ Να δημιουργούμε και να διαγράφουμε σχέσεις μεταξύ δύο πινάκων
- ◆ Να καταχωρούμε εγγραφές στο δεύτερο συσχετιζόμενο πίνακα.

1. Εισαγωγή

Έχουμε ήδη μάθει πώς να δημιουργούμε σχέσεις μεταξύ των πινάκων πάνω στο χαρτί. Σε αυτό το μάθημα θα εξηγηθεί ο τρόπος που θα δημιουργηθούν αυτές οι σχέσεις μεταξύ των πινάκων στη Microsoft Access.

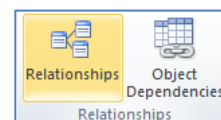
Για να δημιουργήσουμε αυτές τις ειδικές σχέσεις (Relationships) πρέπει να έχουμε ήδη δημιουργημένη τη βάση δεδομένων με τουλάχιστον δύο πίνακες που σχετίζονται μεταξύ τους. Οι σχέσεις σε μια βάση δεδομένων συνήθως, ορίζονται **πριν** την καταχώριση δεδομένων στους πίνακες. Εάν καταχωρίσουμε δεδομένα σε έναν πίνακα και στη συνέχεια, επιχειρήσουμε να συμπεριλάβουμε τον πίνακα σε κάποια σχέση, τότε μπορεί να αντιμετωπίσουμε προβλήματα στη δημιουργία της σχέσης, όταν στα πεδία τα οποία επιχειρούμε να συσχετίσουμε υπάρχουν σε κάποιες εγγραφές του ενός πίνακα τιμές οι οποίες δεν αντιστοιχούν με τις τιμές στο δεύτερο πίνακα.

2. Δημιουργία σχέσης Ένα-Προς-Πολλά μεταξύ δύο πινάκων (One-To-Many, 1-∞)

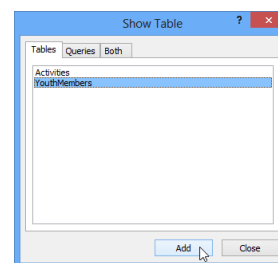
Η διαδικασία που ακολουθούμε για να δημιουργήσουμε μια σχέση μεταξύ δύο πινάκων είναι η εξής:

(1) Αποφασίζουμε πρώτα ποιοι πίνακες θα συμπεριληφθούν στη σχέση που θέλουμε να χρησιμοποιήσουμε και βεβαιωνόμαστε ότι αυτοί οι πίνακες είναι **κλειστοί**.

(2) Από την καρτέλα Database Tools και την ομάδα Relationships επιλέγουμε το εικονίδιο Relationships για να ανοίξουμε το παράθυρο των σχέσεων. Μαζί του ανοίγει και το πλαίσιο διαλόγου εμφάνισης πινάκων, και έτσι μπορούμε να δημιουργήσουμε τη σχέση που θέλουμε να χρησιμοποιήσουμε.



(3) Στη συνέχεια, από το πλαίσιο διαλόγου εμφάνισης πινάκων επιλέγουμε τον/τους **πίνακα/ες** ή το/τα **ερώτημα/τα** που θέλουμε να χρησιμοποιήσουμε και κάνουμε κλικ στο κουμπί για να το/τα **προσθέσουμε** στο παράθυρο των σχέσεών μας.

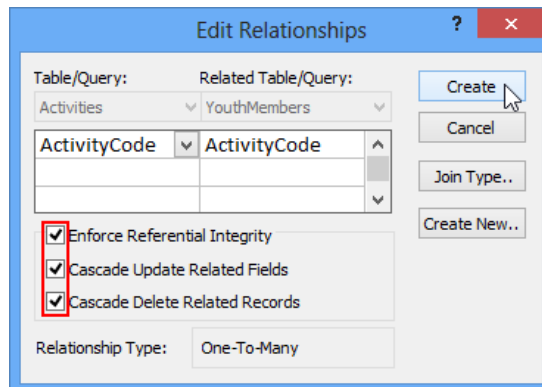


(4) Ακολούθως, κάνουμε κλικ στο κουμπί για να **κλείσουμε** το πλαίσιο διαλόγου εμφάνισης πινάκων.

(5) Μετά, επιλέγουμε το πεδίο που είναι το πρωτεύον κλειδί (ActivityCode) από τον πρωτεύοντα πίνακα (Activities) της σχέσης, το **σύρουμε** πάνω στο συσχετιζόμενο

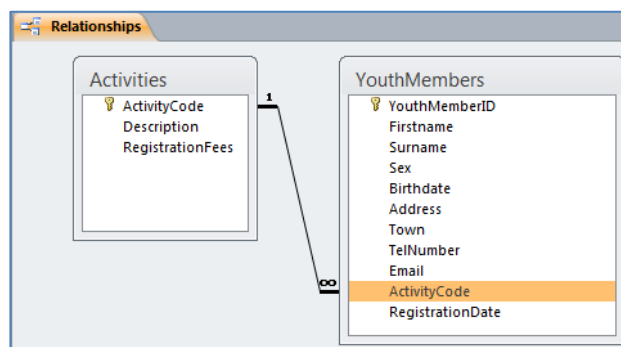
πεδίο (**ActivityCode**) του δευτερεύοντος πίνακα (**YouthMembers**) και το αφήνουμε.

(6) Στο παράθυρο της επεξεργασίας των σχέσεων που εμφανίζεται, επιλέγουμε τα πιο κάτω:

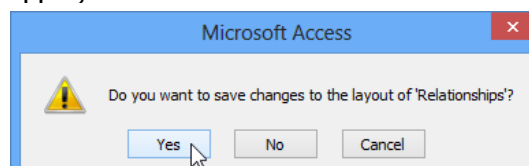
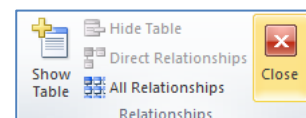


- **Enforce Referential Integrity** (Ενεργοποίηση Ακεραιότητας Αναφορών) - Το επιλέγουμε για να βεβαιωθούμε ότι τα συνδεδεμένα πεδία δεν περιέχουν τιμές που δεν ταιριάζουν.
- **Cascade Update Related Fields** (Διαδοχική Ενημέρωση Συσχετιζόμενων Πεδίων) - Το επιλέγουμε για να ενημερωθούν όλες οι συσχετιζόμενες εγγραφές σε περίπτωση που αλλάξουμε την τιμή του πρωτεύοντος κλειδιού μιας εγγραφής.
- **Cascade Delete Related Records** (Διαδοχική Διαγραφή Συσχετιζόμενων Εγγραφών) - Το επιλέγουμε όταν στην περίπτωση που διαγράψουμε την εγγραφή στον πρώτο πίνακα, επιθυμούμε να διαγράφονται και όλες οι συσχετιζόμενες εγγραφές στον άλλο πίνακα.

(7) Ακολούθως, κάνουμε κλικ στο κουμπί **Create** και η σχέση μας θα δημιουργηθεί.



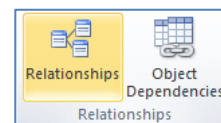
(8) Τέλος, από την καρτέλα **Relationship Tools**→**Design** και την ομάδα **Relationships** επιλέγουμε το εικονίδιο **Close** για να κλείσουμε το παράθυρο των σχέσεων και αποθηκεύουμε στη συνέχεια, τη σχέση μας.



3. Διαγραφή Σχέσης μεταξύ δύο πινάκων (Delete Relationship)

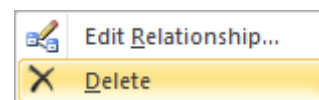
Η διαδικασία που ακολουθούμε για να διαγράψουμε μια σχέση μεταξύ δύο πινάκων είναι η εξής:

- (1) Από την καρτέλα Database Tools και την ομάδα Relationships επιλέγουμε το εικονίδιο Relationships για να ανοίξουμε το παράθυρο των σχέσεων.



- (2) Στη συνέχεια, κάνουμε **δεξί κλικ** πάνω στη γραμμή της σχέσης που συνδέει τους δύο πίνακες που θέλουμε να διαγράψουμε.


- (3) Ακολούθως, κάνουμε κλικ στο κουμπί **Delete**, επιβεβαιώνουμε την πρόθεσή μας για διαγραφή της σχέσης, κάνοντας κλικ στο κουμπί **Yes** και η σχέση μας έτσι **διαγράφεται**.



4. Καταχώριση Εγγραφών στον δεύτερο συσχετιζόμενο πίνακα

Για να καταχωρίσουμε εγγραφές στον δεύτερο συσχετιζόμενο πίνακα μπορούμε να ακολουθήσουμε την ίδια διαδικασία όπως και πριν, όταν οι πίνακες δεν σχετιζονταν. Μας παρέχεται όμως επιπρόσθετα, η δυνατότητα από τον πρώτο πίνακα να βλέπουμε και να καταχωρούμε στοιχεία στον δεύτερο συσχετιζόμενο πίνακα όπως πιο κάτω:

Ανοίγουμε και βλέπουμε τις εγγραφές του δεύτερου πίνακα πατώντας στο συν (+) που εμφανίζεται μπροστά από κάθε εγγραφή του πρώτου πίνακα. Έτσι, στο παράδειγμά μας, ανοίγοντας τον πίνακα δραστηριοτήτων **Activities** μπορούμε να βλέπουμε και τα μέλη που ανήκουν στην κάθε δραστηριότητα. Επίσης, μπορούμε να εισάγουμε νέα μέλη στον πίνακα **YouthMembers** μέσω της προβολής του πίνακα δραστηριοτήτων **Activities**.

- (1) Επιλέγουμε τον πρωτεύοντα πίνακα (Activities) της σχέσης και κάνουμε **διπλό κλικ** πάνω του για να τον ανοίξουμε σε προβολή φύλλου δεδομένων (Datasheet View).
- (2) Ακολούθως, κάνουμε κλικ στο κουμπί  που βρίσκεται αριστερά από την εγγραφή που θέλουμε να καταχωρίσουμε τις σχετικές εγγραφές και **γράφουμε** την εγγραφή που θέλουμε να χρησιμοποιήσουμε.

| ActivityCode | Description | RegistrationFees | Click to Add |
|--------------|--------------------------|------------------|--------------|
| 1 | Ποδόσφαιρο | 100,0 € | |
| 2 | Καλαθόσφαιρα | 80,0 € | |
| 3 | Πετάνσφαιρα | 75,0 € | |
| 4 | Επιτραπέζια Αντισφαίριση | 60,0 € | |
| 5 | Σκάκι | 75,0 € | |
| 6 | Σκι | 100,0 € | |
| 7 | Χειροσφαίριση | 60,0 € | |
| 8 | Ιππασία | 180,0 € | |
| 9 | Τένις | 120,0 € | |
| 10 | Μπάδμιντον | 90,0 € | |
| 11 | Καράτε | 120,0 € | |
| | (New) | | |

Υπόμνημα

Περιγραφή
(Description)

Καρτέλα
(Tab)

Εικονίδιο
(Icon)

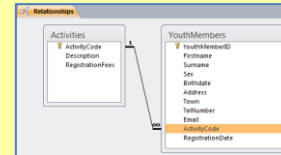
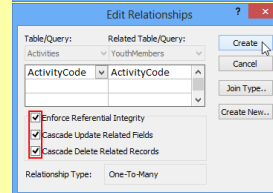
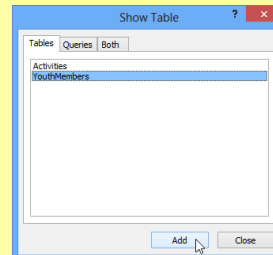
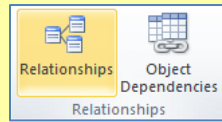
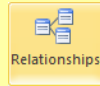
Ομάδα
(Group)

Παράθυρο Διαλόγου
(Dialog Box)

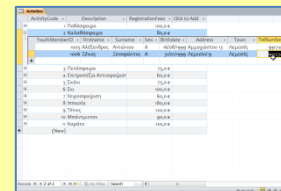
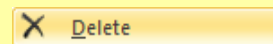
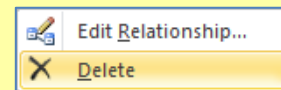
Πλήκτρα
(Keys)

Δημιουργία Σχέσης
Ένα Προς Πολλά
μεταξύ δύο πινάκων
**(One-To-Many,
1-∞)**

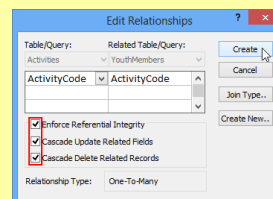
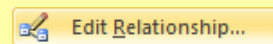
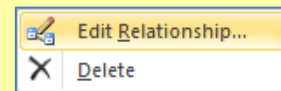
Database Tools



Δεξί κλικ



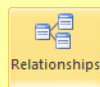
Δεξί κλικ



Καταχώριση
Εγγραφών
στο δεύτερο
συσχετιζόμενο πίνακα

Επεξεργασία Σχέσεων
(Edit Relationship)

Database Tools



Γ6.7 Φόρμες Εισαγωγής Δεδομένων

Τι θα μάθουμε σήμερα:

- ◆ Να περιγράψουμε την ανάγκη χρήσης της φόρμας εισαγωγής δεδομένων (*πλεονεκτήματα*)
- ◆ Να δημιουργούμε μια απλή φόρμα εισαγωγής δεδομένων
- ◆ Να καταχωρούμε, να επεξεργαζόμαστε και να διαγράφουμε εγγραφές σε έναν πίνακα χρησιμοποιώντας τη φόρμα εισαγωγής δεδομένων
- ◆ Να μετακινούμαστε μεταξύ των εγγραφών σε φόρμα (*Επόμενη/Προηγούμενη, Πρώτη/Τελευταία*).

1. Φόρμα Εισαγωγής Δεδομένων (Form)

Ένας πιο **φιλικός** και πιο **εύκολος** τρόπος για την καταχώριση, διαγραφή και επεξεργασία των δεδομένων σε μια βάση δεδομένων, είναι η χρήση των φορμών εισαγωγής δεδομένων (Forms).

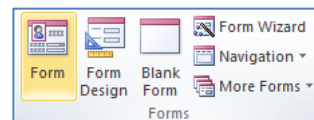
Κάθε φόρμα είναι συνδεδεμένη με έναν ή περισσότερους πίνακες, μέσα στους οποίους καταχωρεί τα δεδομένα (εγγραφές). Επίσης, εμφανίζει από τους πίνακες τα ήδη καταχωρημένα δεδομένα για σκοπούς πληροφόρησης ή και τροποποίησης.

Η δυνατότητα σχεδίασης και μορφοποίησης μιας φόρμας με βάση τις ανάγκες του χρήστη της βάσης δεδομένων, την κάνει πολύ πιο εύχρηστη και πιο φιλική προς τον χρήστη.

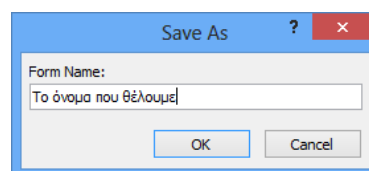
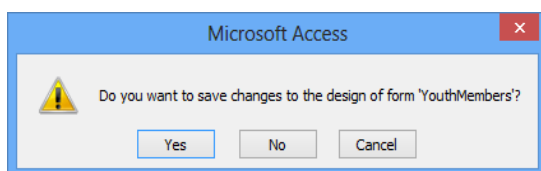
2. Δημιουργία Φόρμας με το Εργαλείο Αυτόματης Φόρμας (Form)

(1) Επιλέγουμε τον πίνακα.

(2) Από την καρτέλα Create και την ομάδα Forms επιλέγουμε το εικονίδιο Form και η φόρμα μας δημιουργείται με έναν **αυτόματο** σχετικά τρόπο.



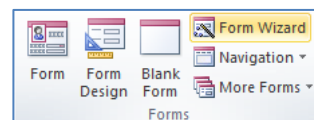
(3) Τέλος, κλείνουμε τη φόρμα, της καθορίζουμε ένα **περιγραφικό όνομα** και αποθηκεύουμε τη φόρμα μας.







3. Δημιουργία Φόρμας με τη χρήση του Οδηγού Φορμών (Form Wizard)

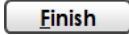
(1) Επιλέγουμε τον πίνακα.

(2) Από την καρτέλα Create και την ομάδα Forms επιλέγουμε το εικονίδιο Form Wizard.



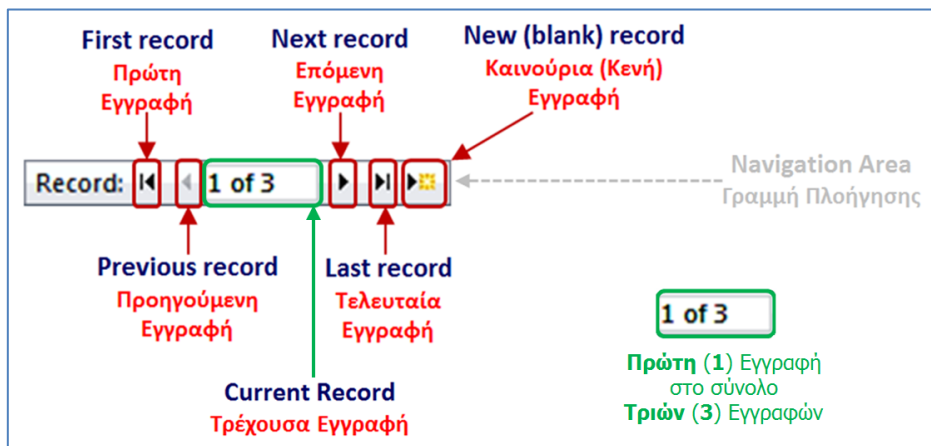
(3) Ακολούθως, κάνουμε κλικ στο κουμπί  για να μεταφέρουμε ΟΛΑ τα πεδία στο πλαίσιο **Selected Fields:** ή κάνουμε κλικ στο κουμπί  για να μεταφέρουμε ΕΝΑ προς ΕΝΑ τα πεδία στο πλαίσιο **Selected Fields:** και μετά κάνουμε κλικ στο κουμπί  για να εμφανίσουμε το επόμενο πλαίσιο διαλόγου.

(4) Στη συνέχεια, επιλέγουμε την **επιθυμητή διάταξη** για τη φόρμα μας και κάνουμε ξανά κλικ στο κουμπί  για να εμφανίσουμε το επόμενο πλαίσιο διαλόγου.


- (5) Τέλος, καθορίζουμε ένα **περιγραφικό όνομα** και κάνουμε κλικ στο κουμπί  για να **αποθηκεύσουμε** τη φόρμα μας.

4. Εισαγωγή, Επεξεργασία και Μετακίνηση μεταξύ των εγγραφών σε μια φόρμα εισαγωγής δεδομένων

- (1) Επιλέγουμε τη φόρμα και κάνουμε **διπλό κλικ** πάνω της για να την ανοίξουμε πρώτα σε προβολή φόρμας (**Form View**).
- (2) Ακολούθως, κάνουμε κλικ σε ένα από τα εικονίδια της γραμμής πλοήγησης πιο κάτω:



5. Διαγραφή Εγγραφής σε μια φόρμα (Delete Record)

- (1) Επιλέγουμε τη φόρμα και κάνουμε **διπλό κλικ** πάνω της για να την ανοίξουμε πρώτα σε προβολή φόρμας (**Form View**).
- (2) Στη συνέχεια, επιλέγουμε την **εγγραφή** που θέλουμε να διαγράψουμε από τη φόρμα μας.
- (3) Από την καρτέλα Home και την ομάδα Records επιλέγουμε το εικονίδιο Delete.
- (4) Ακολούθως, κάνουμε κλικ στο κουμπί  για να **διαγράψουμε** την εγγραφή που θέλουμε από τη φόρμα μας.




6. Προβολές για Φόρμα (Views)

6.1 Προβολή Σχεδίασης (Design View)

- (1) Από την περιοχή όλων των αντικειμένων της εφαρμογής (**All Access Objects**) κάτω από τον τίτλο **Forms**, όπου εμφανίζονται όλες οι φόρμες που έχουν δημιουργηθεί στη βάση δεδομένων μας, επιλέγουμε τη φόρμα που θέλουμε και κάνουμε **διπλό κλικ** πάνω της για να την ανοίξουμε πρώτα σε προβολή φόρμας (**Form View**).
- (2) Ακολούθως, από την καρτέλα Home και την ομάδα Views επιλέγουμε το εικονίδιο Design View για να ανοίξουμε τη φόρμα μας σε προβολή σχεδίασης.

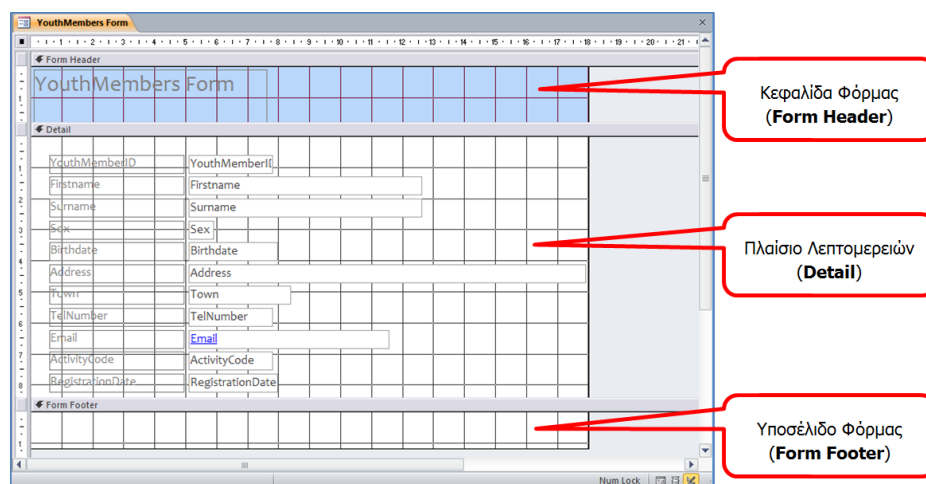


ή

Κάνουμε κλικ στο κουμπί  από τη γραμμή κατάστασης στην κάτω δεξιά γωνιά της οθόνης μας.

ή

Από την περιοχή όλων των αντικειμένων της εφαρμογής (All Access Objects) κάτω από τον τίτλο **Forms**, όπου εμφανίζονται όλες οι φόρμες που έχουν δημιουργηθεί στη βάση δεδομένων μας, επιλέγουμε τη φόρμα που θέλουμε και κάνουμε **δεξί κλικ** πάνω της και επιλέγουμε την εντολή **Design View** για να την ανοίξει σε προβολή σχεδίασης.



6.2 Προβολή Φόρμας (Form View)


(1) Από την περιοχή όλων των αντικειμένων της εφαρμογής (All Access Objects) κάτω από τον τίτλο **Forms**, όπου εμφανίζονται όλες οι φόρμες που έχουν δημιουργηθεί στη βάση δεδομένων μας, επιλέγουμε τη φόρμα που θέλουμε και κάνουμε **διπλό κλικ** πάνω της για να την ανοίξουμε σε προβολή φόρμας (Form View).

ή

Από την καρτέλα Home και την ομάδα Views επιλέγουμε το εικονίδιο Form View για να ανοίξουμε τη φόρμα μας σε προβολή φόρμας.



ή

Κάνουμε κλικ στο κουμπί  από τη γραμμή κατάστασης στην κάτω δεξιά γωνιά της οθόνης μας.

7. Μορφοποίηση Φόρμας

Μια φόρμα εισαγωγής δεδομένων μετά που θα τη δημιουργήσουμε, έχουμε τη δυνατότητα να τη μορφοποιήσουμε και να την τροποποιήσουμε έτσι ώστε να την κάνουμε πιο φιλική και ύχρηστη στον χρήστη της βάσης δεδομένων μας.

Για κάθε πεδίο της φόρμας έχουμε μια ετικέτα (Label) στην οποία αναγράφεται η περιγραφή του πεδίου και ένα πλαίσιο κειμένου (Text Box) στο οποίο ο χρήστης της φόρμας μπορεί να καταχωρεί ή να εμφανίζει την τιμή του αντίστοιχου πεδίου για κάθε εγγραφή.



Μπορούμε να μορφοποιήσουμε μια φόρμα, κάνοντας **διπλό κλικ** πάνω της και ανοίγοντάς την πρώτα σε προβολή φόρμας (Form View).

7.1 Αλλαγή Διαστάσεων της ενεργού περιοχής της φόρμας


7.1.1 Αλλαγή Πλάτους της ενεργού περιοχής της φόρμας

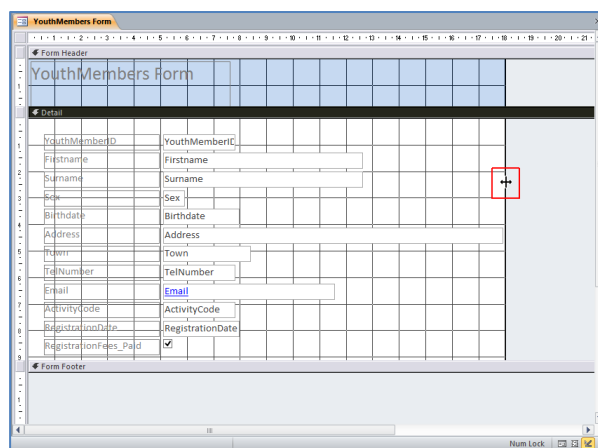
- (1) Από την καρτέλα Home και την ομάδα Views επιλέγουμε το εικονίδιο Design View για να ανοίξουμε τη φόρμα μας σε προβολή σχεδίασης.



ή


Επιλέγουμε τη φόρμα και κάνουμε δεξί κλικ πάνω της και επιλέγουμε την εντολή **Design View** για να την ανοίξει σε προβολή σχεδίασης.

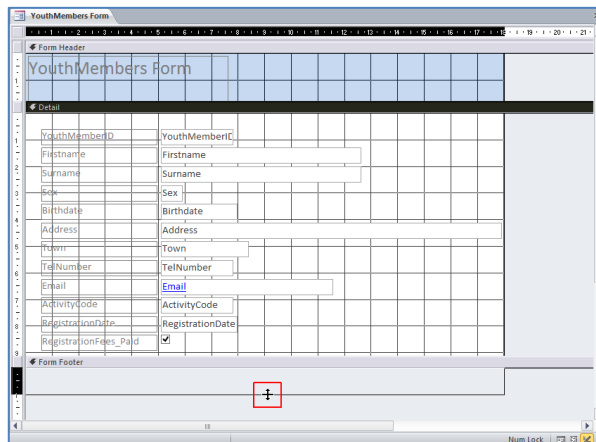
- (2) Στη συνέχεια, κάνουμε κλικ στη δεξιά πλευρά του περιγράμματος της ενεργού περιοχής της φόρμας.
- (3) Όταν ο δείκτης του ποντικιού πάρει τη μορφή γραμμής με τα δύο αντίθετα βέλη, , **σύρουμε ανάλογα** προς τα αριστερά ή προς τα δεξιά για να αλλάξουμε το πλάτος της ενεργού περιοχής της φόρμας.




7.1.2 Αλλαγή Ύψους της ενεργού περιοχής της φόρμας

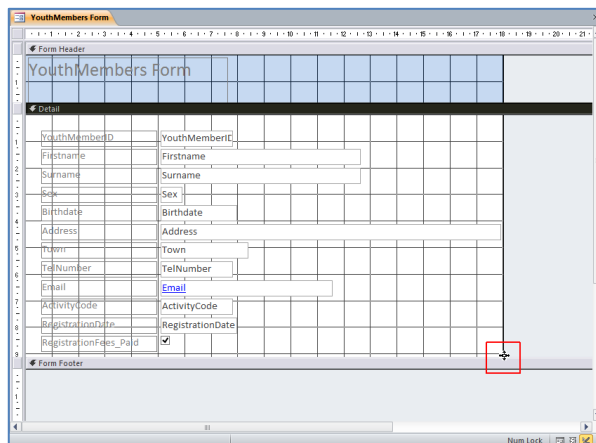
- (1) Κάνουμε κλικ στην κάτω πλευρά του περιγράμματος της ενεργού περιοχής της φόρμας.

- (2) Όταν ο δείκτης του ποντικιού πάρει τη μορφή γραμμής με τα δύο αντίθετα βέλη, , **σύρουμε ανάλογα** προς τα πάνω ή προς τα κάτω για να αλλάξουμε το ύψος της ενεργού περιοχής της φόρμας.



7.1.3 Αλλαγή Πλάτους και Ύψους της ενεργού περιοχής της φόρμας ταυτόχρονα


- (1) Κάνουμε κλικ στην **κάτω δεξιά πλευρά του περιγράμματος** της ενεργού περιοχής της φόρμας.
- (2) Όταν ο δείκτης του ποντικιού πάρει τη μορφή γραμμής με τα δύο αντίθετα βέλη, , **σύρουμε ανάλογα διαγώνια** προς τα πάνω ή προς τα κάτω για να αλλάξουμε τις διαστάσεις της ενεργού περιοχής της φόρμας.

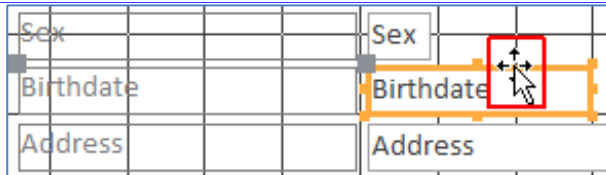


7.2 Μετακίνηση Στοιχείων Ελέγχου


- (1) Κάνουμε κλικ σε ένα στοιχείο ελέγχου (είτε στο πλαίσιο κειμένου είτε στην ετικέτα) πάνω όμως στο περίγραμμα του (όχι σε κάποια λαβή του).

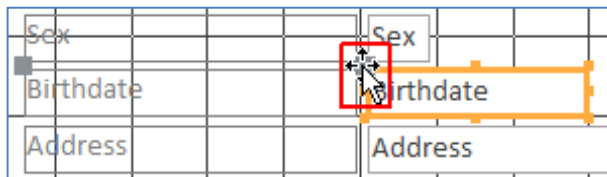
Σημείωση: Γύρω από το περίγραμμά του θα εμφανιστούν μικρά πορτοκαλί τετραγωνάκια τα οποία ονομάζονται λαβές (handles).

Όταν ο δείκτης του ποντικιού πάρει τη μορφή με τα τέσσερα αντίθετα βέλη, , **σύρουμε ανάλογα** προς την κατεύθυνση που θέλουμε να το μετακινήσουμε.




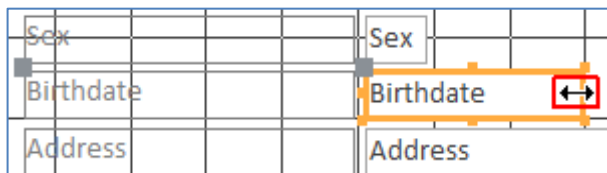
7.3 Μετακίνηση του Πλαισίου Κειμένου ή της Ετικέτας ξεχωριστά

- (1) Κάνουμε κλικ σε ένα στοιχείο ελέγχου (είτε στο πλαίσιο κειμένου είτε στην ετικέτα) πάνω όμως στο γκρίζο τετραγωνάκι του.
- (2) Όταν ο δείκτης του ποντικιού πάρει τη μορφή με τα τέσσερα αντίθετα βέλη, , **σύρουμε ανάλογα** προς την κατεύθυνση που θέλουμε να το μετακινήσουμε.



7.4 Αλλαγή Διαστάσεων του Πλαισίου Κειμένου ή της Ετικέτας ξεχωριστά

- (1) Κάνουμε κλικ σε ένα στοιχείο ελέγχου (είτε στο πλαίσιο κειμένου είτε στην ετικέτα) γύρω όμως από το περίγραμμά τους πάνω σε κάποια από τις λαβές του.
- (2) Όταν ο δείκτης του ποντικιού πάρει τη μορφή διπλού βέλους , **σύρουμε ανάλογα** προς τα αριστερά ή δεξιά ή πάνω ή κάτω ή διαγώνια για να του αλλάξουμε τις διαστάσεις του.

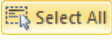


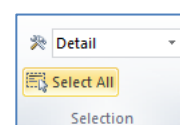
7.5 Μορφοποίηση Στοιχείων Ελέγχου (Formatting)

- (1) Κάνουμε κλικ σε ένα στοιχείο ελέγχου (είτε στο πλαίσιο κειμένου είτε στην ετικέτα) πάνω όμως στο γκρίζο τετραγωνάκι του.
- (2) Ακολουθώς, από την καρτέλα Form Design Tools→**Format** και την ομάδα Font κάνουμε κλικ σε μια από τις διαθέσιμες επιλογές μορφοποίησης που υπάρχουν και επιλέγουμε τη μορφοποίηση που θέλουμε να χρησιμοποιήσουμε.



7.6 Επιλογή Όλων των Αντικειμένων (Select All)

- (1) Από την καρτέλα Form Design Tools→**Format** και την ομάδα Selection επιλέγουμε το εικονίδιο  για να επιλέξουμε όλα τα αντικείμενα που υπάρχουν στη φόρμα.



ή

Μπορούμε επίσης να χρησιμοποιήσουμε τον συνδυασμό πλήκτρων



7.7 Διαγραφή Πεδίου από τη φόρμα (Delete Field)

- (1) Κάνουμε κλικ σε ένα από τα πλαίσια κειμένου πάνω όμως στο γκρίζο τετραγωνάκι του.

(*Σημείωση: Το πρόγραμμα θα αφαιρέσει και το πλαίσιο κειμένου και την ετικέτα του μαζί από τη φόρμα σας.*)

- (2) Ακολούθως, από την καρτέλα Home και την ομάδα Records επιλέγουμε το εικονίδιο Delete για να το διαγράψουμε από τη φόρμα.



ή

Κάνουμε δεξί κλικ και επιλέγουμε την εντολή **Delete** για να το διαγράψουμε από τη φόρμα.

ή

Μπορούμε επίσης να χρησιμοποιήσουμε το πλήκτρο **Delete**.

7.8 Προσθήκη Πεδίου στη φόρμα (Add Existing Fields)

- (1) Από την καρτέλα Form Design Tools→**Design** και την ομάδα Tools επιλέγουμε το εικονίδιο .


- (2) Ακολούθως, από το παράθυρο που εμφανίζεται επιλέγουμε το πεδίο που θέλουμε να χρησιμοποιήσουμε και το **σύρουμε** πάνω στη φόρμα.



7.9 Προσθήκη Ετικέτας στη φόρμα (Label)

- (1) Από την καρτέλα Form Design Tools→**Design** και την ομάδα Controls επιλέγουμε το εικονίδιο .

- (2) Ακολούθως, κάνουμε κλικ σε ένα οποιοδήποτε

σημείο πάνω στη φόρμα όπου ο δείκτης του ποντικιού θα μετατραπεί σε  και γράφουμε μέσα στον χώρο που θα δημιουργήσει αυτό που θέλουμε.



Υπόμνημα

| Περιγραφή (Description) | Καρτέλα (Tab) | Εικονίδιο (Icon) | Ομάδα (Group) | Παράθυρο Διαλόγου (Dialog Box) | Πλήκτρα (Keys) |
|------------------------------------------------------------------------|-------------------------|----------------------------|-------------------------|------------------------------------------|--------------------------|
| Δημιουργία Φόρμας με το Εργαλείο Αυτόματης Φόρμας (Form) | | | | | |
| Δημιουργία Φόρμας με τη χρήση του Οδηγού Φορμών (Form Wizard) | | | | | |
| Γραμμή Πλοήγησης (Navigation Area) | | | | | |
| Πρώτη Εγγραφή (First record) | | | | | |
| Προηγούμενη Εγγραφή (Previous record) | | | | | |
| Τρέχουσα Εγγραφή (Current Record) | | | | | |
| Επόμενη Εγγραφή (Next record) | | | | | |
| Τελευταία Εγγραφή (Last record) | | | | | |
| Νέα Κενή Εγγραφή (New (blank) record) | | | | | |
| Διαγραφή Εγγραφής (Delete Record) | | | | | |

Μετακίνηση του
δρομέα στην ίδια την
εγγραφή
**(Επόμενο και
Προηγούμενο)**



Μετακίνηση του
δρομέα στην ίδια την
εγγραφή
(Επόμενο)



Μετακίνηση του
δρομέα στην ίδια την
εγγραφή
(Προηγούμενο)

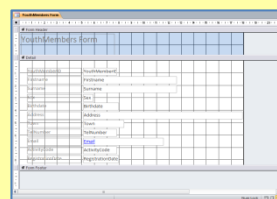


+



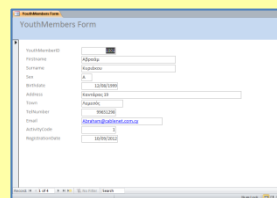
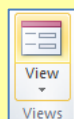
Προβολή
Σχεδίασης
(Design View)

Home



Προβολή
Φόρμας
(Form View)

Home



Γ6.8 Ερωτήματα

Τι θα μάθουμε σήμερα:

- ◆ Να περιγράφουμε την ανάγκη χρήσης των ερωτημάτων σε μια βάση δεδομένων (*πλεονεκτήματα*)
- ◆ Να δημιουργούμε, να τροποποιούμε, να αποθηκεύουμε και να διαγράφουμε απλά ερωτήματα
- ◆ Να εφαρμόζουμε, να προσθέτουμε και να διαγράφουμε κριτήρια σε ένα ερώτημα (*χρήση σταθερών τιμών και συγκριτικών τελεστών: =, <, >, <>, <=, >=, between ...and ...*)
- ◆ Να εφαρμόζουμε, να προσθέτουμε και να διαγράφουμε κριτήρια σε ένα ερώτημα (*χρήση λογικών τελεστών και χαρακτήρων μπαλαντέρ: AND, OR, NOT και *, ?*)
- ◆ Να ταξινομούμε αποτελέσματα ερωτημάτων βάσει των πεδίων σε αύξουσα/φθίνουσα, αλφαβητική/αριθμητική σειρά (*Ascending/Descending*).

1. Ερωτήματα (Queries)

Μια βάση δεδομένων μπορεί να περιέχει πολλούς πίνακες συνδεδεμένους μεταξύ τους, με χιλιάδες εγγραφές στον καθένα. Είναι, λοιπόν, πολύ δύσκολο να βρει ο χρήστης συγκεκριμένα δεδομένα που ψάχνει.

Η χρήση των φίλτρων ή της εντολής αναζήτησης που έχουν ήδη αναφερθεί, βοηθάνε σε κάποιο βαθμό αλλά έχουν και πολλά μειονεκτήματα, όπως το ότι δεν διατηρούνται και το ότι δεν μπορούν να πάρουν δεδομένα από δύο ή περισσότερους πίνακες. Λύση στα πιο πάνω προβλήματα δίνουν τα **Ερωτήματα (Queries)**.

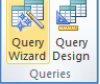
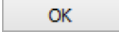





Πιο κάτω παραθέτουμε μερικές από τις δυνατότητες (πλεονεκτήματα) των **ερωτημάτων**:

- Μπορούμε να επιλέξουμε μόνο συγκεκριμένα πεδία ενός πίνακα και να τα παρουσιάσουμε με τη σειρά που θέλουμε.
- Μπορούμε να θέσουμε κριτήρια και να εμφανιστούν μόνο οι εγγραφές που ικανοποιούν αυτά τα κριτήρια.
- Μπορούμε να τα αποθηκεύσουμε στη βάση δεδομένων και να τα εκτελέσουμε όποτε τα χρειαστούμε.
- Μπορούμε να αντλούμε δεδομένα από έναν ή περισσότερους πίνακες.
- Μπορούν, επίσης, να κάνουν και υπολογισμούς.

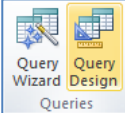
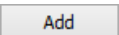
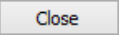
Παίρνουμε ως **παράδειγμα** τη βάση δεδομένων ενός σχολείου, η οποία υποθέτουμε ότι έχει τρεις πίνακες. Έναν πίνακα που περιέχει τα **προσωπικά στοιχεία** των μαθητών, άλλο που περιέχει τους **βαθμούς** και έναν τρίτο που περιέχει τις **απουσίες**:

- (1) Με τη χρήση ενός ερωτήματος από τον πίνακα με τα προσωπικά στοιχεία μπορούμε να πάρουμε κάποια από τα προσωπικά δεδομένα των μαθητών, όπως το ονοματεπώνυμο και το τηλέφωνο των γονέων τους, ενώ από τον πίνακα των απουσιών μπορούμε να εντοπίσουμε ποιοι μαθητές έχουν πρόβλημα απουσιών.
- (2) Με ένα άλλο ερώτημα από τον πίνακα με τους βαθμούς μπορούμε να βρούμε τους άριστους μαθητές κ.λπ.

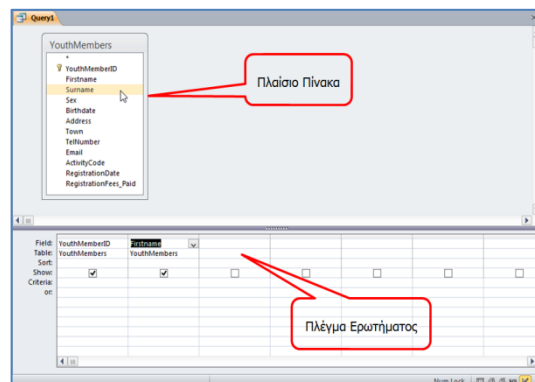
2. Δημιουργία Ερωτήματος με τη χρήση του Οδηγού Ερωτημάτων (Query Wizard)

- (1) Επιλέγουμε τον πίνακα.
- (2) Από την καρτέλα **Create** και την ομάδα **Queries** επιλέγουμε το εικονίδιο **Query Wizard**. 
- (3) Στη συνέχεια, επιλέγουμε **Simple Query Wizard** και κάνουμε κλικ στο κουμπί  για να εμφανίσουμε το επόμενο πλαίσιο διαλόγου.
- (4) Ακολούθως, κάνουμε κλικ στο κουμπί  για να μεταφέρουμε ΟΛΑ τα πεδία στο πλαίσιο **Selected Fields:** ή κάνουμε κλικ στο κουμπί  για να μεταφέρουμε ΕΝΑ προς ΕΝΑ τα πεδία στο πλαίσιο **Selected Fields:** και μετά κάνουμε κλικ στο κουμπί  για να εμφανίσουμε το επόμενο πλαίσιο διαλόγου.
- (5) Στη συνέχεια, επιλέγουμε την **επιθυμητή διάταξη** (Αναλυτικά ή Περιληπτικά) για το ερώτημά μας και κάνουμε ξανά κλικ στο κουμπί  για να εμφανίσουμε το επόμενο πλαίσιο διαλόγου.
- (6) Τέλος, καθορίζουμε ένα **περιγραφικό όνομα** και κάνουμε κλικ στο κουμπί  για να **αποθηκεύσουμε** το ερώτημά μας.


3. Δημιουργία Ερωτήματος με τη χρήση της Προβολή Σχεδίασης (Query Design)

- (1) Επιλέγουμε τον πίνακα.
- (2) Από την καρτέλα **Create** και την ομάδα **Queries** επιλέγουμε το εικονίδιο **Query Design**. 
- (3) Στη συνέχεια, από το πλαίσιο διαλόγου εμφάνισης πινάκων (**Show Table**) επιλέγουμε τον/τους **πίνακα/ες** που θέλουμε να χρησιμοποιήσουμε και κάνουμε κλικ στο κουμπί  για να τον/τους **προσθέσουμε** στην προβολή σχεδίασης (**Design View**) του ερωτήματός μας.
- (4) Ακολούθως, κάνουμε κλικ στο κουμπί  για να **κλείσουμε** το πλαίσιο διαλόγου εμφάνισης πινάκων.
- (5) Στο **πλαίσιο πίνακα**, κάνουμε **διπλό κλικ** στο πρώτο πεδίο για να το **προσθέσουμε** στην πρώτη στήλη του πλέγματος ερωτημάτων μας.

(*Σημείωση: Στο πάνω μισό τμήμα του παραθύρου μπορούμε να δούμε το **πλαίσιο πίνακα** με τα πεδία του. (Το * στην αρχή του καταλόγου αντιπροσωπεύει όλα τα πεδία.) Στο κάτω μισό τμήμα του παραθύρου βρίσκεται το πλέγμα ενός πίνακα – ονομάζεται **πλέγμα ερωτήματος**, στο οποίο μπορούμε να δημιουργήσουμε και να τροποποιήσουμε το ερώτημά μας με οπτικό τρόπο.*)

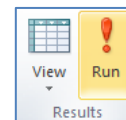


(6) Στη συνέχεια, κάνουμε **διπλό κλικ** στο δεύτερο πεδίο για να το **προσθέσουμε** στη δεύτερη στήλη του πλέγματος ερωτημάτων μας με τον ίδιο τρόπο.

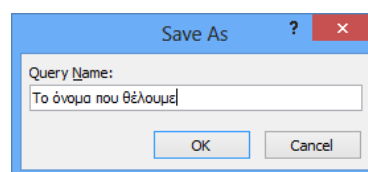
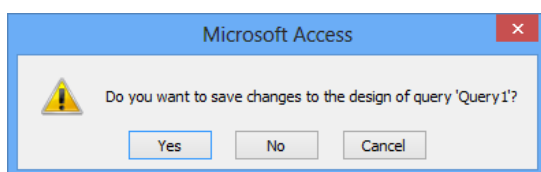
(*Σημείωση: Εάν προσθέσουμε λάθος πεδίο, κάνουμε κλικ στο πλήκτρο  για να το διαγράψουμε.*)

(7) Συνεχίζουμε με τον ίδιο τρόπο όπως πιο πάνω μέχρι να **προσθέσουμε** όλα τα πεδία που θέλουμε να χρησιμοποιήσουμε στο ερώτημά μας.

(8) Ακολούθως, από την καρτέλα Query Tools→**Design** και την ομάδα Results επιλέγουμε το εικονίδιο Run για να εκτελέσει το ερώτημα η Microsoft Access και να εμφανίσει το αποτέλεσμα.



(9) Τέλος, κλείνουμε το ερώτημα, του καθορίζουμε ένα **περιγραφικό όνομα** και αποθηκεύουμε το ερώτημα μας.



4. Τροποποίηση Ερωτήματος

(1) Από την περιοχή όλων των αντικειμένων της εφαρμογής (All Access Objects) κάτω από τον τίτλο **Queries**, επιλέγουμε το ερώτημα που θέλουμε να τροποποιήσουμε και κάνουμε **διπλό κλικ** πάνω του για να το ανοίξουμε πρώτα σε προβολή φύλλου δεδομένων (Datasheet View).

(2) Ακολούθως, από την καρτέλα Home και την ομάδα Views επιλέγουμε το εικονίδιο Design View για να ανοίξουμε το ερώτημά μας σε προβολή σχεδίασης.

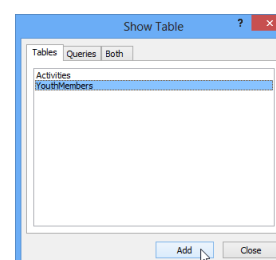
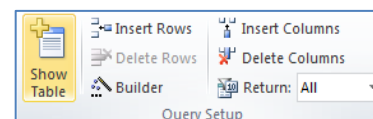


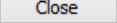
ή

Επιλέγουμε το ερώτημα από τη συγκεκριμένη περιοχή, κάνουμε δεξί κλικ πάνω του και επιλέγουμε την εντολή **Design View** για να το ανοίξει σε προβολή σχεδίασης.

(3) Στην περίπτωση που θέλουμε να **προσθέσουμε** ακόμη έναν πίνακα ή ακόμη ένα ερώτημα στο νέο μας ερώτημα, κάνουμε τις πιο κάτω ενέργειες:

- Από την καρτέλα Query Tools→**Design** και την ομάδα Query Setup επιλέγουμε το εικονίδιο Show Table για να ανοίξουμε το πλαίσιο διαλόγου εμφάνισης πινάκων.
- Στη συνέχεια, από το πλαίσιο διαλόγου εμφάνισης πινάκων επιλέγουμε τον/τους **πίνακα/ες** ή το/τα **ερώτημα/τα** που θέλουμε να χρησιμοποιήσουμε και κάνουμε κλικ στο κουμπί **Add** για να το/τα **προσθέσουμε** στην προβολή σχεδίασης του ερωτήματός μας.

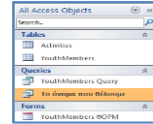


- Ακολουθως, κάνουμε κλικ στο κουμπί  για να **κλείσουμε** το πλαίσιο διαλόγου εμφάνισης πινάκων.

(4) Στη συνέχεια, κλείνουμε το ερώτημά, αποθηκεύοντας όμως **ξανά** τις νέες αλλαγές που κάναμε στο ερώτημά μας.

5. Διαγραφή Ερωτήματος (Delete Query)

(1) Από την περιοχή όλων των αντικειμένων της εφαρμογής (All Access Objects) κάτω από τον τίτλο **Queries**, επιλέγουμε το ερώτημα που θέλουμε να διαγράψουμε.

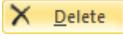


(2) Ακολουθως, κάνουμε δεξί κλικ και επιλέγουμε την εντολή **Delete** για να το διαγράψουμε από τη βάση δεδομένων μας.

ή

Από την καρτέλα Home και την ομάδα Records επιλέγουμε το εικονίδιο Delete.



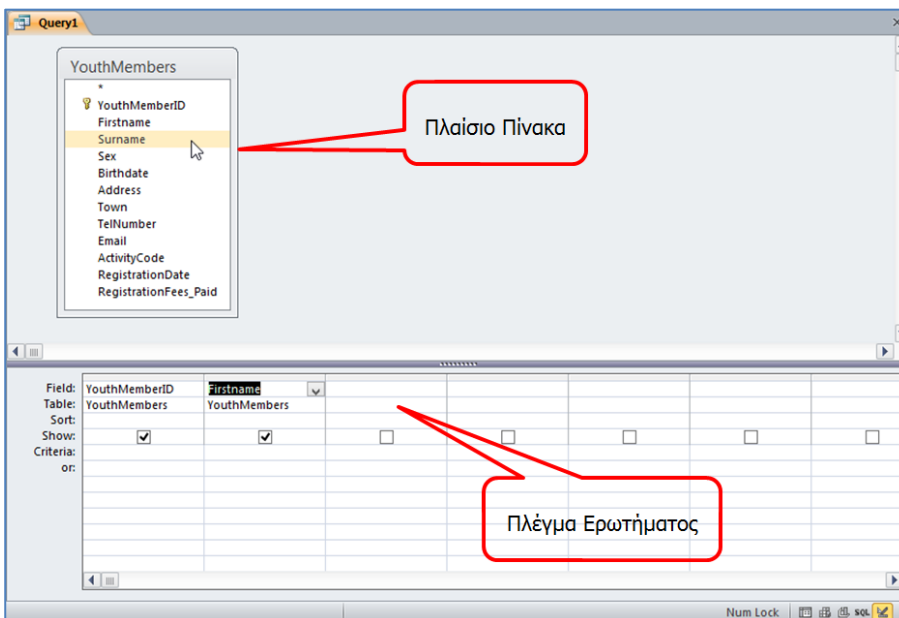
Ακολουθως, κάνουμε κλικ στο κουμπί  για να το **διαγράψουμε** από τη βάση δεδομένων μας.

ή

Μπορούμε, επίσης, να χρησιμοποιήσουμε το πλήκτρο **Delete**.

6. Το Παράθυρο της Προβολής Σχεδίασης ενός Ερωτήματος

Το παράθυρο της προβολής σχεδίασης ενός ερωτήματος χωρίζεται σε **δύο τμήματα**:



(1) Στο πάνω μισό τμήμα του παραθύρου μπορούμε να δούμε το **πλαίσιο πίνακα** με τα πεδία του στο οποίο βασίζεται το ερώτημα. Το ερώτημα του δικού μας παραδείγματος βασίζεται σε έναν μόνο πίνακα, γι' αυτό και βλέπουμε μόνο μια λίστα πεδίων. (Το * στην αρχή του καταλόγου αντιπροσωπεύει όλα τα πεδία.)

(2) Στο κάτω μισό τμήμα του παραθύρου βρίσκεται το πλέγμα ενός πίνακα – ονομάζεται **πλέγμα ερωτήματος**, στο οποίο μπορούμε να δημιουργήσουμε και να τροποποιήσουμε το ερώτημά μας με οπτικό τρόπο όπως πιο κάτω:

- **Field:** Παρουσιάζει τα πεδία που θα χρησιμοποιήσουμε.
- **Table:** Παρουσιάζει τον/τους πίνακα/ες που θα χρησιμοποιήσουμε.
- **Sort:** Καθορίζουμε την ταξινόμηση που θα χρησιμοποιήσουμε όπως:
 - **Ascending** (ΑΥΞΟΥΣΑ ταξινόμηση),
 - **Descending** (ΦΘΙΝΟΥΣΑ ταξινόμηση),
 - **(not sorted)** (ΔΙΧΩΣ ταξινόμηση).
- **Show:** Καθορίζουμε εάν το πεδίο θα είναι ορατό ή όχι όταν εκτελέσουμε το ερώτημα.
- **Criteria:** Καθορίζουμε τα κριτήρια, δηλαδή τους περιορισμούς που ορίζουμε στα ερωτήματα, ώστε να μας βρίσκουν μόνο τις εγγραφές που μας ενδιαφέρουν.

7. Κριτήρια Ερωτημάτων (Criteria)

Τα κριτήρια είναι οι **περιορισμοί** που ορίζουμε στα ερωτήματα, ώστε να βρίσκουν από τη Βάση Δεδομένων **μόνο τις εγγραφές** που μας ενδιαφέρουν όπως:

| Σταθερές Τιμές | Έκφραση | Περιγραφή |
|----------------|-----------|------------|
| Λέξη ή Γράμμα | Λεμεσός | Ίση με ... |
| Αριθμός | 2012 | Ίση με ... |
| Ημερομηνία | 31/1/1967 | Ίση με ... |

| Συγκριτικοί Τελεστές | Έκφραση | Περιγραφή |
|----------------------|----------------------------------------|-------------------------|
| Between ...and... | Between 1/1/2013 and 31/12/2013 | Μεταξύ ... και ... |
| = | =35 | Ίση με ... |
| > | >36 | Μεγαλύτερη από ... |
| < | <40 | Μικρότερη από ... |
| <> | <>0 | Διάφορο του ... (άνισο) |
| >= | >=50 | Μεγαλύτερη ή ίση με ... |
| <= | <=46 | Μικρότερη ή ίση με ... |

| Λογικοί Τελεστές | Έκφραση | Περιγραφή |
|------------------|-------------|------------------------------------|
| AND | <50 AND >40 | και ο ένας περιορισμός και ο άλλος |
| OR | 48 OR 52 | ή ο ένας περιορισμός ή ο άλλος |
| NOT | NOT Ελένη | Εκτός από ... |

| Χαρακτήρες Μπαλαντέρ | Έκφραση | Περιγραφή |
|----------------------|--------------|------------------------------------------------------------------------------|
| * | N* | Αντιστοιχεί σε κανένα ή περισσότερους χαρακτήρες |
| ? | N???? | Αντιστοιχεί με οποιονδήποτε μεμονωμένο χαρακτήρα |
| [] | [XK]ριστίνα | Αντιστοιχεί σε κάποιο από τους χαρακτήρες στις αγκύλες |
| ! | [!XK]ριστίνα | Αντιστοιχεί με οποιονδήποτε χαρακτήρα εκτός από τους χαρακτήρες στις αγκύλες |
| - | [A-E] | Αντιστοιχεί με κάθε χαρακτήρα που βρίσκεται στην περιοχή μέσα στις αγκύλες |
| # | 2#5 | Αντιστοιχεί με οποιονδήποτε μεμονωμένο αριθμητικό χαρακτήρα |

Υπόμνημα

Περιγραφή
(Description)

Καρτέλα
(Tab)

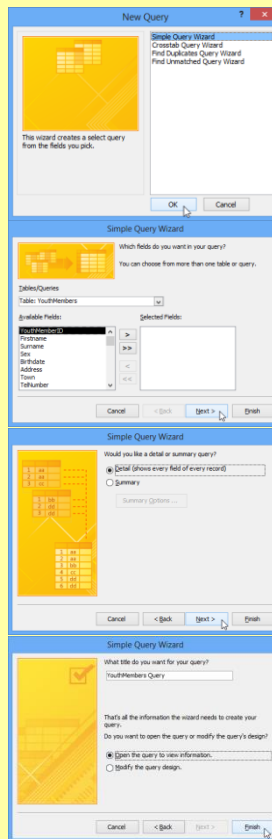
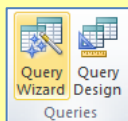
Εικονίδιο
(Icon)

Ομάδα
(Group)

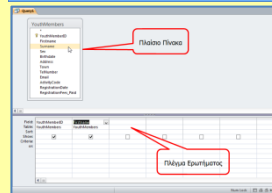
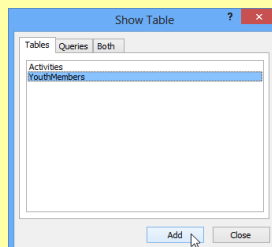
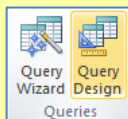
Παράθυρο Διαλόγου
(Dialog Box)

Πλήκτρα
(Keys)

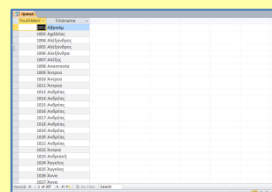
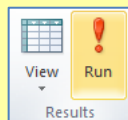
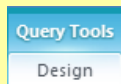
Δημιουργία Ερωτήματος με τη χρήση του Οδηγού Ερωτημάτων (**Query Wizard**)



Δημιουργία Ερωτήματος με τη χρήση της Προβολής Σχεδίασης (**Query Design**)



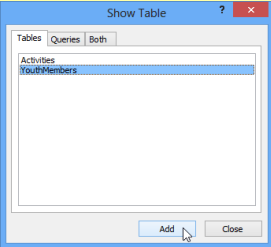
Εκτέλεση Ερωτήματος (**Run**)



Εμφάνιση Πινάκων σε Ερώτημα (Show Table)

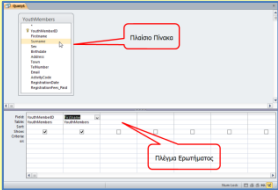
Query Tools Design Show Table

Insert Rows Insert Columns Delete Rows Delete Columns Builder Return: All Query Setup

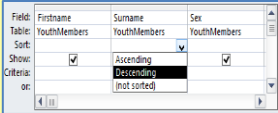


Τροποποίηση Ερωτήματος

Home View Views



Ταξινόμηση Αποτελεσμάτων των Ερωτημάτων (Sort)



Διαγραφή Ερωτήματος (Delete Query)

Home Delete

Refresh All New Save Delete Records Totals Spelling More

Delete Delete Record Delete Column Delete

Γ6.9 Ερωτήματα από Δύο Συσχετιζόμενους Πίνακες

Τι θα μάθουμε σήμερα:

- ◆ Να δημιουργούμε ερωτήματα από δύο συσχετιζόμενους πίνακες.

1. Ερωτήματα από δύο Συσχετιζόμενους Πίνακες (Queries in Related Tables)

Τα ερωτήματα από **δύο συσχετιζόμενους πίνακες** αποτελούν έναν από τους πιο αποτελεσματικούς τρόπους εξαγωγής συγκεκριμένων πληροφοριών από μια συσχετιζόμενη βάση δεδομένων.

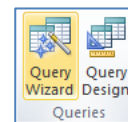
Όταν υπάρχουν συσχετισμοί μεταξύ δύο πινάκων, οι πληροφορίες που μπορούν να παραχθούν από ένα ερώτημα περιέχουν ήδη τους συσχετισμούς ανάμεσα στα δεδομένα των δύο πινάκων, έχοντας έτσι το πλεονέκτημα της **ταυτόχρονης παρουσίασης συσχετιζόμενων πληροφοριών** από δύο ή περισσότερους πίνακες.

Αυτό είναι και το **τεράστιο πλεονέκτημα** της δημιουργίας ερωτημάτων από δύο συσχετιζόμενους πίνακες: η ταυτόχρονη συγκέντρωση, αντιπαραβολή και παρουσίαση συσχετιζόμενων πληροφοριών από μια βάση δεδομένων για περαιτέρω επεξεργασία ή αποθήκευση.

2. Δημιουργία ερωτήματος με τη χρήση του Οδηγού Ερωτημάτων (Query Wizard)

(1) Επιλέγουμε τον πίνακα.

(2) Από την καρτέλα **Create** και την ομάδα **Queries** επιλέγουμε το εικονίδιο **Query Wizard**.



(3) Στη συνέχεια, επιλέγουμε **Simple Query Wizard** και κάνουμε κλικ στο κουμπί για να εμφανίσουμε το επόμενο πλαίσιο διαλόγου.

(4) Ακολούθως, κάνουμε κλικ στο κουμπί για να μεταφέρουμε ΟΛΑ τα πεδία στο πλαίσιο **Selected Fields:** ή κάνουμε κλικ στο κουμπί για να μεταφέρουμε ΕΝΑ προς ΕΝΑ τα πεδία στο πλαίσιο **Selected Fields:** και μετά κάνουμε κλικ στο κουμπί για να εμφανίσουμε το επόμενο πλαίσιο διαλόγου.

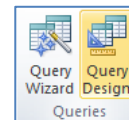
(5) Στη συνέχεια, επιλέγουμε την **επιθυμητή διάταξη** (Αναλυτικά ή Περιληπτικά) για το ερώτημά μας και κάνουμε ξανά κλικ στο κουμπί για να εμφανίσουμε το επόμενο πλαίσιο διαλόγου.

(6) Τέλος, καθορίζουμε ένα **περιγραφικό όνομα** και κάνουμε κλικ στο κουμπί για να **αποθηκεύσουμε** το ερώτημά μας.

3. Δημιουργία ερωτήματος με τη χρήση της Προβολής Σχεδίασης (Query Design)

(1) Επιλέγουμε τον πίνακα.

(2) Από την καρτέλα **Create** και την ομάδα **Queries** επιλέγουμε το εικονίδιο **Query Design**.



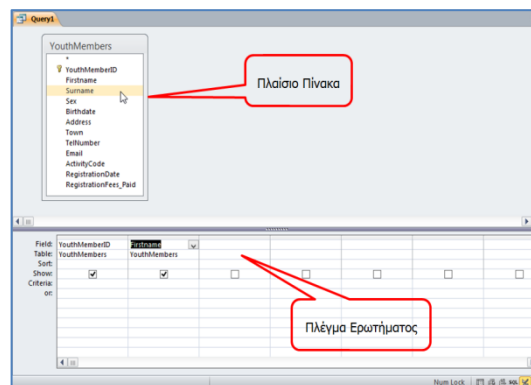
(3) Στη συνέχεια, από το πλαίσιο διαλόγου εμφάνισης πινάκων (**Show Table**) επιλέγουμε τον/τους **πίνακα/ες** που θέλουμε να χρησιμοποιήσουμε και κάνουμε κλικ στο κουμπί για να τον/τους **προσθέσουμε** στην προβολή σχεδίασης (**Design View**) του ερωτήματός μας.

(4) Ακολούθως, κάνουμε κλικ στο κουμπί για να **κλείσουμε** το πλαίσιο διαλόγου εμφάνισης πινάκων.

(5) Στο **πλαίσιο πίνακα**, κάνουμε **διπλό κλικ** στο πρώτο πεδίο για να το **προσθέσουμε** στην πρώτη στήλη του πλέγματος ερωτημάτων μας.

*(Σημείωση: Στο πάνω μισό τμήμα του παραθύρου μπορούμε να δούμε το **πλαίσιο πίνακα** με τα πεδία του. (Το * στην αρχή του καταλόγου αντιπροσωπεύει όλα τα πεδία.)*

*Στο κάτω μισό τμήμα του παραθύρου βρίσκεται το πλέγμα ενός πίνακα – ονομάζεται **πλέγμα ερωτήματος**, στο οποίο μπορούμε να δημιουργήσουμε και να τροποποιήσουμε το ερώτημά μας με οπτικό τρόπο.)*

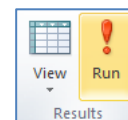


(6) Στη συνέχεια, κάνουμε **διπλό κλικ** στο δεύτερο πεδίο για να το **προσθέσουμε** στη δεύτερη στήλη του πλέγματος ερωτημάτων μας με τον ίδιο τρόπο.

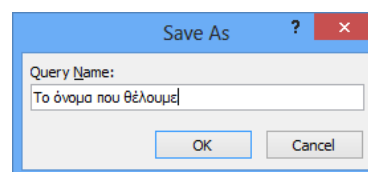
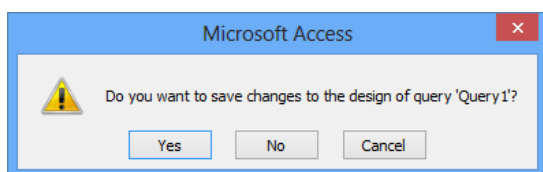
*(Σημείωση: Εάν προσθέσουμε λάθος πεδίο, κάνουμε κλικ στο πλήκτρο **Delete** για να το διαγράψουμε.)*

(7) Συνεχίζουμε με τον ίδιο τρόπο όπως πιο πάνω μέχρι να **προσθέσουμε** όλα τα πεδία που θέλουμε να χρησιμοποιήσουμε στο ερώτημά μας.

(8) Ακολούθως, από την καρτέλα **Query Tools**→**Design** και την ομάδα **Results** επιλέγουμε το εικονίδιο **Run** για να εκτελέσει το ερώτημα η Microsoft Access και να εμφανίσει το αποτέλεσμα.



(9) Τέλος, κλείνουμε το ερώτημα, του καθορίζουμε ένα **περιγραφικό όνομα** και αποθηκεύουμε το ερώτημά μας.



Υπόμνημα

Περιγραφή
(Description)

Καρτέλα
(Tab)

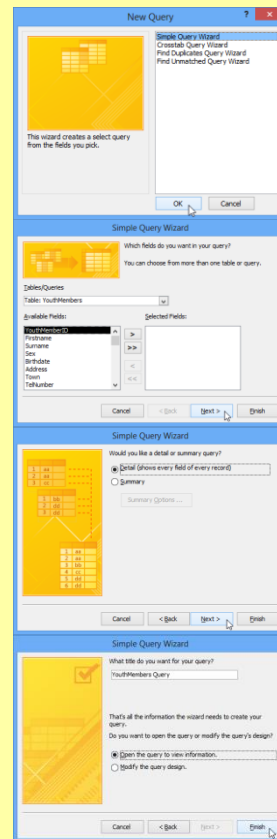
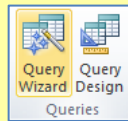
Εικονίδιο
(Icon)

Ομάδα
(Group)

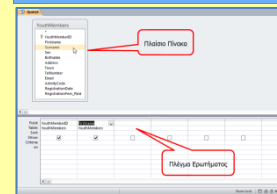
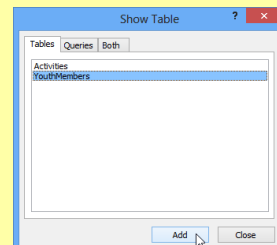
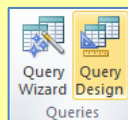
Παράθυρο Διαλόγου
(Dialog Box)

Πλήκτρα
(Keys)

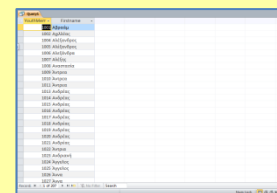
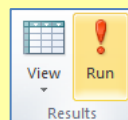
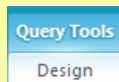
Δημιουργία Ερωτήματος με τη χρήση του Οδηγού Ερωτημάτων (**Query Wizard**)

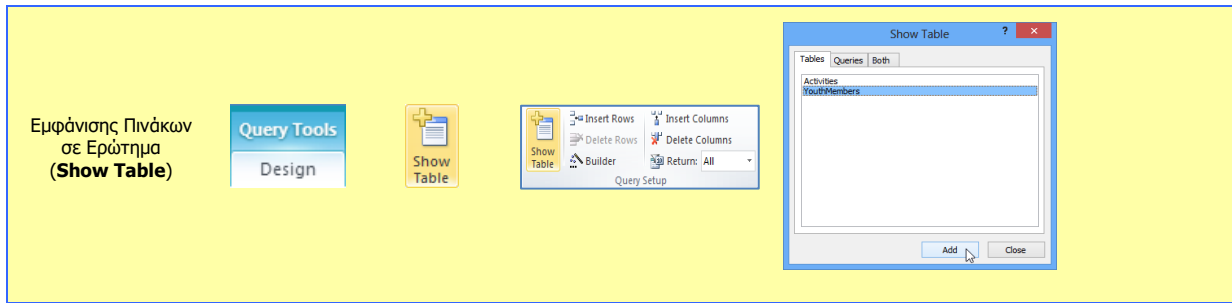


Δημιουργία Ερωτήματος με τη χρήση της Προβολής Σχεδίασης (**Query Design**)



Εκτέλεση Ερωτήματος (**Run**)





Γ6.10 Εκθέσεις

Τι θα μάθουμε σήμερα:

- ◆ Να περιγράψουμε την ανάγκη χρήσης των εκθέσεων σε μια βάση δεδομένων τόσο από πίνακες όσο και από ερωτήματα (*πλεονεκτήματα*)
- ◆ Να δημιουργούμε, να αποθηκεύουμε, να εκτυπώνουμε και να διαγράφουμε απλές εκθέσεις.

1. Εκθέσεις (Reports)

Συχνά, χρειάζεται να επιλέξουμε, να οργανώσουμε και να τυπώσουμε συγκεκριμένα στοιχεία της βάσης δεδομένων μας. Για αυτό τον λόγο χρειαζόμαστε τις **εκθέσεις ή αναφορές** οι οποίες είναι εργαλεία που έχουν ως βασικό σκοπό τους την αποτελεσματική παρουσίαση των δεδομένων μας σε **έντυπη μορφή**, έτοιμα προς εκτύπωση.

Μια έκθεση αφού τη δημιουργήσουμε και την αποθηκεύσουμε, μπορούμε να την εκτελέσουμε (τρέξουμε). Η ενέργεια αυτή έχει πάντα ως αποτέλεσμα την απεικόνιση των δεδομένων που υπάρχουν στη βάση δεδομένων τη συγκεκριμένη στιγμή.

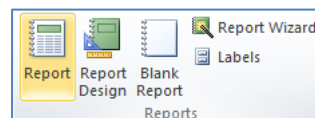
Κάθε έκθεση μπορεί να μορφοποιηθεί έτσι ώστε να παρουσιάζει τις πληροφορίες με τον καλύτερο δυνατό τρόπο σε ό,τι αφορά την ανάγνωση και τη χρήση τους.

Οι εκθέσεις δημιουργούνται συνήθως για **εκτύπωση** ή για την **προβολή** τους στην οθόνη. Μπορούμε επίσης, το αποτέλεσμα τους να το αποθηκεύσουμε σε ένα αρχείο ή να το κάνουμε εξαγωγή σε κάποιο άλλο πρόγραμμα ή να το αποστείλουμε ως μήνυμα ηλεκτρονικού ταχυδρομείου.

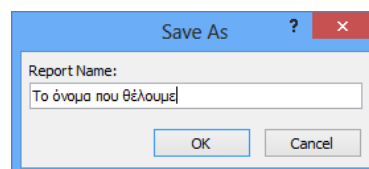
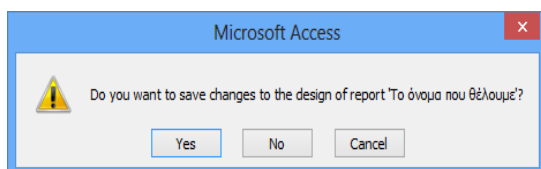
2. Δημιουργία Έκθεσης με το Εργαλείο Αυτόματης Έκθεσης (Report)

(1) Επιλέγουμε το ερώτημα ή τον πίνακα για το/τον οποίο θέλουμε να δημιουργήσουμε μια έκθεση.

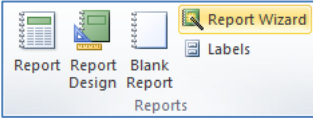
(2) Από την καρτέλα **Create** και την ομάδα **Reports** επιλέγουμε το εικονίδιο **Report** και η έκθεσή μας δημιουργείται με ένα **αυτόματο** σχετικά τρόπο.



(3) Τέλος, κλείνουμε την έκθεση, της καθορίζουμε ένα **περιγραφικό όνομα** και αποθηκεύουμε την έκθεσή μας.




3. Δημιουργία Έκθεσης με τη χρήση του Οδηγού Εκθέσεων (Report Wizard)


- (1) Επιλέγουμε το ερώτημα ή τον πίνακα για το/τον οποίο θέλουμε να δημιουργήσουμε μια έκθεση και ...
- (2) Από την καρτέλα **Create** και την ομάδα **Reports** επιλέγουμε το εικονίδιο **Report Wizard**.
 
- (3) Ακολούθως, κάνουμε κλικ στο κουμπί **>>** για να μεταφέρουμε ΟΛΑ τα πεδία στο πλαίσιο **Selected Fields:** ή κάνουμε κλικ στο κουμπί **>** για να μεταφέρουμε ΕΝΑ προς ΕΝΑ τα πεδία στο πλαίσιο **Selected Fields:** και μετά κάνουμε κλικ στο κουμπί **Next >** για να εμφανίσουμε το επόμενο πλαίσιο διαλόγου.
- (4) Στη συνέχεια, επιλέγουμε ένα από τα πεδία βάσει του οποίου θέλουμε να **ομαδοποιήσουμε** την έκθεσή μας και κάνουμε κλικ στο κουμπί **>** για να το μεταφέρουμε στο πλαίσιο στα δεξιά και κάνουμε ξανά κλικ στο κουμπί **Next >** για να εμφανίσουμε το επόμενο πλαίσιο διαλόγου.
- (5) Στη συνέχεια, μπορούμε επιλέγοντας μέχρι και τέσσερα πεδία, να **ταξινομήσουμε** την έκθεσή μας ανάλογα, σε αύξουσα ή φθίνουσα αλφαβητική/αριθμητική σειρά βάσει του πεδίου και κάνουμε ξανά κλικ στο κουμπί **Next >** για να εμφανίσουμε το επόμενο πλαίσιο διαλόγου.
- (6) Στη συνέχεια, επιλέγουμε την **επιθυμητή διάταξη** μαζί με τον **επιθυμητό προσανατολισμό** της έκθεσής μας και κάνουμε ξανά κλικ στο κουμπί **Next >** για να εμφανίσουμε το επόμενο πλαίσιο διαλόγου.
- (7) Τέλος, καθορίζουμε ένα **περιγραφικό όνομα** και κάνουμε κλικ στο κουμπί **Finish** για να **αποθηκεύσουμε** την έκθεσή μας.

4. Προβολές για Έκθεση (Views)

4.1 Προβολή Σχεδίασης (Design View)

- (1) Από την περιοχή όλων των αντικειμένων της εφαρμογής (**All Access Objects**) κάτω από τον τίτλο **Reports**, όπου εμφανίζονται όλες οι εκθέσεις που έχουν δημιουργηθεί στη βάση δεδομένων μας, επιλέγουμε την έκθεση που θέλουμε και κάνουμε **διπλό κλικ** πάνω της για να την ανοίξουμε πρώτα σε προβολή έκθεσης (**Report View**).
- (2) Ακολούθως, από την καρτέλα **Home** και την ομάδα **Views** επιλέγουμε το εικονίδιο **Design View** για να ανοίξουμε την έκθεσή μας σε προβολή σχεδίασης.
 

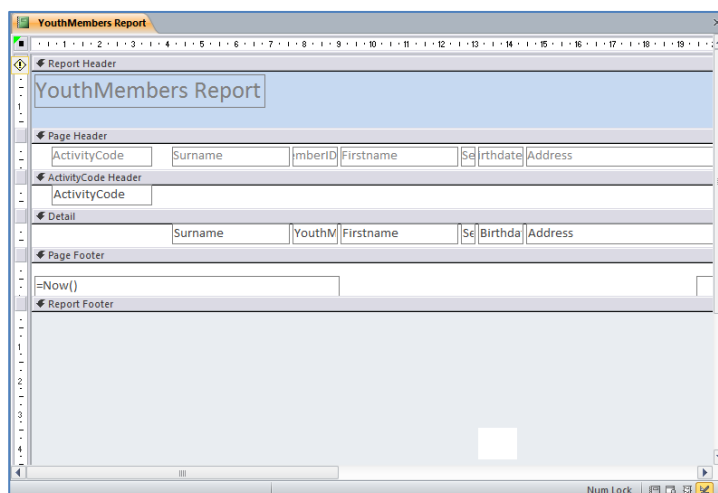
ή

Κάνουμε κλικ στο κουμπί  από τη γραμμή κατάστασης στην κάτω δεξιά γωνιά της οθόνης μας.

ή

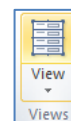
Από την περιοχή όλων των αντικειμένων της εφαρμογής (**All Access Objects**) κάτω από τον τίτλο **Reports**, όπου εμφανίζονται όλες οι εκθέσεις που έχουν δημιουργηθεί στη βάση δεδομένων μας, επιλέγουμε την έκθεση που θέλουμε

και κάνουμε **δεξί κλικ** πάνω της και επιλέγουμε την εντολή **Design View** για να την ανοίξει σε προβολή σχεδίασης.




4.2 Προβολή Διάταξης Έκθεσης (Layout View)

- (1) Από την περιοχή όλων των αντικειμένων της εφαρμογής (**All Access Objects**) κάτω από τον τίτλο **Reports**, όπου εμφανίζονται όλες οι εκθέσεις που έχουν δημιουργηθεί στη βάση δεδομένων μας, επιλέγουμε την έκθεση που θέλουμε και κάνουμε **διπλό κλικ** πάνω της για να την ανοίξουμε πρώτα σε προβολή έκθεσης (Report View).
- (2) Ακολούθως, από την καρτέλα Home και την ομάδα Views επιλέγουμε το εικονίδιο Layout View για να ανοίξουμε την έκθεσή μας σε προβολή διάταξης έκθεσης.

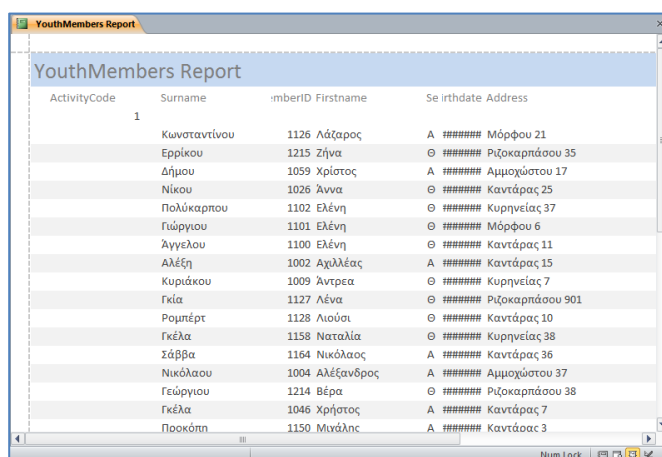


ή

Κάνουμε κλικ στο κουμπί  από τη γραμμή κατάστασης στην κάτω δεξιά γωνιά της οθόνης μας.

ή

Από την περιοχή όλων των αντικειμένων της εφαρμογής (**All Access Objects**) κάτω από τον τίτλο **Reports**, όπου εμφανίζονται όλες οι εκθέσεις που έχουν δημιουργηθεί στη βάση δεδομένων μας, επιλέγουμε την έκθεση που θέλουμε και κάνουμε **δεξί κλικ** πάνω της και επιλέγουμε την εντολή **Layout View** για να την ανοίξει σε προβολή διάταξης έκθεσης.



5. Τροποποίηση Έκθεσης

(1) Από την περιοχή όλων των αντικειμένων της εφαρμογής (All Access Objects) κάτω από τον τίτλο **Reports**, επιλέγουμε την έκθεση που θέλουμε να τροποποιήσουμε και κάνουμε **διπλό κλικ** πάνω της για να την ανοίξουμε πρώτα σε προβολή έκθεσης (Report View).

(2) Ακολούθως, από την καρτέλα Home και την ομάδα Views επιλέγουμε το εικονίδιο Design View για να ανοίξουμε την έκθεσή μας σε προβολή σχεδίασης και κάνουμε τις αλλαγές που θέλουμε ακριβώς με τον ίδιο τρόπο που κάναμε και σε μια φόρμα.



ή

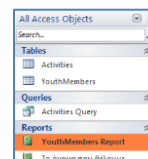
Επιλέγουμε την έκθεση από τη συγκεκριμένη περιοχή, κάνουμε δεξί κλικ πάνω της και επιλέγουμε την εντολή **Design View** για να την ανοίξει σε προβολή σχεδίασης και κάνουμε τις αλλαγές που θέλουμε ακριβώς με τον ίδιο τρόπο που κάναμε και σε μια φόρμα.

(3) Στη συνέχεια, κλείνουμε την έκθεση, αποθηκεύοντας όμως **ξανά** τις νέες αλλαγές που κάναμε στην έκθεσή μας.

6. Διαγραφή Έκθεσης (Delete Report)

(1) Από την περιοχή όλων των αντικειμένων της εφαρμογής (All Access Objects) κάτω από τον τίτλο **Reports**, επιλέγουμε την έκθεση που θέλουμε να διαγράψουμε.


(2) Ακολούθως, κάνουμε δεξί κλικ και επιλέγουμε την εντολή **Delete** για να τη διαγράψουμε από τη βάση δεδομένων μας.



ή

Από την καρτέλα Home και την ομάδα Records επιλέγουμε το εικονίδιο Delete.



Ακολούθως, κάνουμε κλικ στο κουμπί  για να τη **διαγράψουμε** από τη βάση δεδομένων μας.

ή

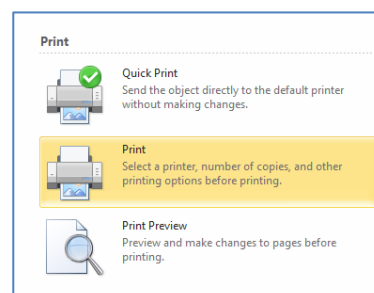
Μπορούμε, επίσης, να χρησιμοποιήσουμε το πλήκτρο .

7. Εκτύπωση Έκθεσης (Print Report)

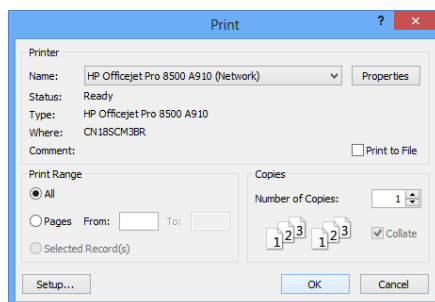
(1) Επιλέγουμε την έκθεση που θέλουμε να εκτυπώσουμε.

(2) Από την καρτέλα File επιλέγουμε την εντολή **Print**.

- Κάνουμε κλικ στο εικονίδιο Quick Print για να αποστείλουμε την έκθεση απευθείας στον προεπιλεγμένο εκτυπωτή χωρίς να ορίσουμε επιλογές εκτυπωτή.
- Κάνουμε κλικ στο εικονίδιο Print για να



ανοίξουμε ένα παράθυρο διαλόγου από το οποίο μπορούμε να επιλέξουμε εκτυπωτή, να καθορίσουμε τον αριθμό των αντιγράφων κ.λπ.



ή

Μπορούμε επίσης να χρησιμοποιήσουμε τον συνδυασμό πλήκτρων **CTRL + P**.

- Κάνουμε κλικ στο εικονίδιο Print Preview για να ανοίξουμε την έκθεσή μας σε Προβολή Προεπισκόπησης της Εκτύπωσης.

Υπόμνημα

| <u>Περιγραφή</u> (Description) | <u>Καρτέλα</u> (Tab) | <u>Εικονίδιο</u> (Icon) | <u>Ομάδα</u> (Group) | <u>Παράθυρο Διαλόγου</u> (Dialog Box) | <u>Πλήκτρα</u> (Keys) |
|-----------------------------------------------------------------------------|-------------------------|----------------------------|-------------------------|------------------------------------------|--------------------------|
| Δημιουργία Έκθεσης με το Εργαλείο Αυτόματης Έκθεσης (Report) | | | | | |
| Δημιουργία Έκθεσης με τη χρήση του Οδηγού Εκθέσεων (Report Wizard) | | | | | |
| | | | | | |
| | | | | | |

| | | | | |
|----------------------------------------------------------|-------------|--------------|--|--|
| <p>Προβολή Σχεδίασης (Design View)</p> | <p>Home</p> | | | |
| <p>Προβολή Διάταξης Έκθεσης (Layout View)</p> | <p>Home</p> | | | |
| <p>Τροποποίηση Έκθεσης</p> | <p>Home</p> | | | |
| <p>Διαγραφή Έκθεσης (Delete Report)</p> | <p>Home</p> | | | |
| <p>Εκτύπωση Έκθεσης (Print Report)</p> | <p>File</p> | <p>Print</p> | | |

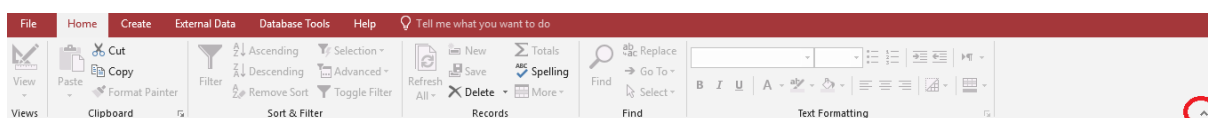
Γ6.11 Επιπρόσθετες Λειτουργίες

Τι θα μάθουμε σήμερα:


- ◆ Να χρησιμοποιούμε τις διαθέσιμες λειτουργίες βοήθειας
- ◆ Να εμφανίζουμε και να αποκρύπτουμε τις διαθέσιμες γραμμές εργαλείων
- ◆ Να επαναφέρουμε και να ελαχιστοποιούμε την κορδέλα
- ◆ Να εξάγουμε τα δεδομένα ενός πίνακα ή τα αποτελέσματα ενός ερωτήματος σε αρχείο υπολογιστικών φύλλων, XML
- ◆ Να αλλάζουμε τον προσανατολισμό της σελίδας, το μέγεθος του χαρτιού και τα περιθώρια εκτύπωσης
- ◆ Να εκτυπώνουμε όλες ή επιλεγμένες εγγραφές.

1. Ελαχιστοποίηση κορδέλας (ribbon)

Για να **ελαχιστοποιήσουμε** πατούμε το κουμπί που βλέπουμε στην πιο κάτω εικόνα. Εναλλακτικά, μπορούμε να πατήσουμε τον συνδυασμό πλήκτρων **CTRL+F1**.



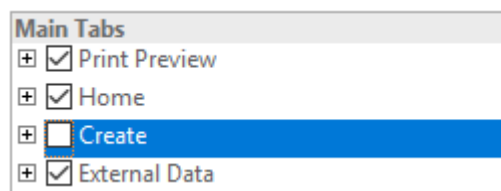
2. Επαναφορά κορδέλας (ribbon)

Για να επαναφέρουμε την κορδέλα πατούμε το κουμπί  ή χρησιμοποιούμε τον συνδυασμό πλήκτρων **CTRL+F1**.

3. Εμφάνιση και απόκρυψη των γραμμών εργαλείων

Για να αποκρύψουμε μια γραμμή εργαλείων ακολουθούμε τα πιο κάτω βήματα:

- (1) Από το μενού **File** επιλέγουμε **Options**
- (2) Στην περίπτωση που η γραμμή εργαλείων που θέλουμε να αποκρύψουμε έχει πράγμα το οποίο σημαίνει ότι εμφανίζεται το κύριο παράθυρο της εφαρμογής,, πατούμε για να εξαφανιστεί το , όπως φαίνεται και στην πιο κάτω εικόνα.



- (3) Πατούμε το κουμπί **OK**.

Σημείωση: Για να εμφανιστεί και πάλι η γραμμή εργαλείων στο κύριο παράθυρο της εφαρμογής, επαναλαμβάνουμε την πιο πάνω διαδικασία, με τη διαφορά ότι θα πρέπει να εμφανίζεται το μπροστά από το όνομα της γραμμής εργαλείων.

4. Χρήση της Βοήθειας (Help)

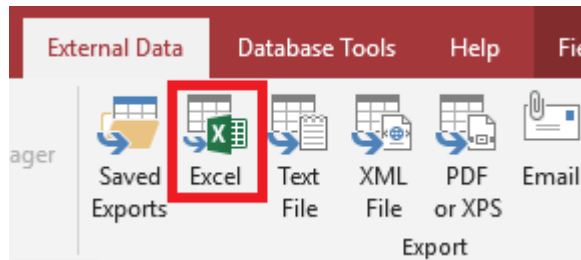
Όπως έχουμε μάθει και στις άλλες εφαρμογές, όταν επιθυμούμε να αναζητήσουμε οτιδήποτε δεν γνωρίζουμε σχετικά με την εφαρμογή, έχουμε την ευκαιρία να χρησιμοποιήσουμε τη διαθέσιμη λειτουργία βοήθειας που υπάρχει.

Για να ενεργοποιήσουμε την βοήθεια της Microsoft Access, μπορούμε να πατήσουμε το κουμπί **Help** που βρίσκεται στη γραμμή εργαλείων, είτε να πατήσουμε το κουμπί **F1** από το πληκτρολόγιο.

5. Εξαγωγή στοιχείων πίνακα ή ερωτήματος

Η Microsoft Access μας δίνει τη δυνατότητα να εξαγάμε τα στοιχεία ενός πίνακα ή ενός ερωτήματος στο υπολογιστικό φύλλο ή και σε άλλες μορφές. Για να το πετύχουμε ακολουθούμε τα πιο κάτω βήματα:

- (1) Από το μενού **External Data**, επιλέγουμε **Excel**, όπως φαίνεται και στην πιο κάτω εικόνα:

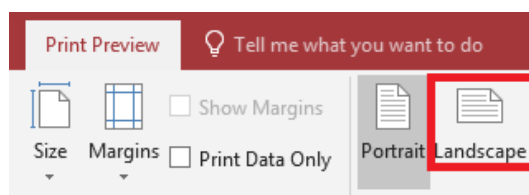


- (2) Στο παράθυρο που εμφανίζεται καθορίζουμε (i) το χώρο αποθήκευσης του αρχείου που θα δημιουργηθεί και (ii) το όνομα του αρχείου.
- (3) Πατούμε το κουμπί **OK**.

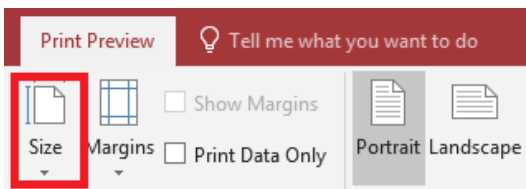
6. Αλλαγή του προσανατολισμού της σελίδας, των περιθωρίων και του μεγέθους του χαρτιού

Για να αλλάξουμε τον προσανατολισμό της σελίδας, τα περιθώρια και το μέγεθος του χαρτιού ακολουθούμε τα πιο κάτω βήματα:

- (1) Από το μενού **File**, επιλέγουμε **Print** και στη συνέχεια **Print Preview**.
- (2) Στην περίπτωση που θέλουμε να αλλάξουμε τον προσανατολισμό επιλέγουμε **Landscape** (για οριζόντιο) ή **Portrait** (για κάθετο).



- (3) Αν θέλουμε να αλλάξουμε το μέγεθος του χαρτιού πατούμε στην επιλογή **Size**.


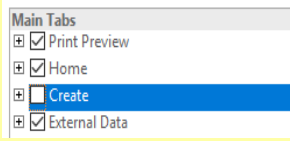
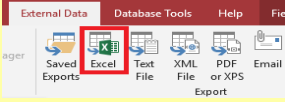
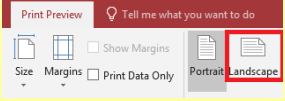
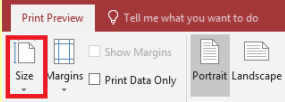


- (4) Τέλος, αν θέλουμε να αλλάξουμε τα περιθώρια, επιλέγουμε **Page Setup**.

7. Εκτύπωση συγκεκριμένων εγγραφών

Στην περίπτωση που θέλουμε να εκτυπώσουμε συγκεκριμένες εγγραφές του πίνακα ή του ερωτήματος ακολουθούμε τα πιο κάτω βήματα:

- (1) Επιλέγουμε τις εγγραφές
- (2) Από το μενού **File**, επιλέγουμε **Print** και πάλι **Print**.
- (3) Στη συνέχεια επιλέγουμε **Selected Records**.

| Υπόμνημα | | | | | |
|------------------------------------------------|-------------------------|----------------------------|-----------------------------|--------------------------------------------------------------------------------------|--------------------------|
| <u>Περιγραφή</u> (Description) | <u>Καρτέλα</u> (Tab) | <u>Εικονίδιο</u> (Icon) | <u>Ομάδα</u> (Group) | <u>Παράθυρο Διαλόγου</u> (Dialog Box) | <u>Πλήκτρα</u> (Keys) |
| Ελαχιστοποίηση και Επαναφορά Κορδέλας (Ribbon) | Home | | |  | CTRL+F1 |
| Εμφάνιση και Απόκρυψη των γραμμών εργαλείων | File | Options | |  | |
| Χρήση βοήθειας (Help) | Home | Help | | | F1 |
| Εξαγωγή στοιχείων πίνακα ή ερωτήματος | External Data | Excel | |  | |
| Αλλαγή προσανατολισμού σελίδας | File | Print Preview | Landscape ή Portrait |  | |
| Αλλαγή του μεγέθους του χαρτιού | File | Print Preview | Size |  | |

ΕΝΟΤΗΤΑ Γ7 Αλγοριθμική Σκέψη, Προγραμματισμός και Σύγχρονες Εφαρμογές Πληροφορικής

Για την προετοιμασία και τη συγγραφή του υλικού αυτής της ενότητας εργάστηκαν οι ακόλουθοι/θες Καθηγητές/τριες Πληροφορικής και Επιστήμης Ηλεκτρονικών Υπολογιστών που παρακολούθησαν το πρόγραμμα επιμόρφωσης μάχιμων εκπαιδευτικών για το Νέο Αναλυτικό Πρόγραμμα:

Αλκείδου Μαρίνα
Ανδρέου Δημήτρης
Αντωνίου Αντώνης Φ.
Αχιλλέως Κλειώ
Γεωργίου Αντώνης
Δημητρίου Λευτέρης
Δημητρίου Παναγιώτα
Διονυσίου Μιχάλης (Σύμβουλος)
Ζαχαρίου Κυριάκος
Θεοδώρου Ηλίας (Σύμβουλος)
Κουρούσις Αντώνης
Μάκκουλα Πόλα
Μακρή Σωτικοπούλου Ελένη
Μέσσιου Παρασκευή

Νικολάου Νικολέττα
Οικονομίδης Γιώργος
Παναγίδης Μιχάλης
Παπαδοπούλου Άνθη
Παπαλυσάνδρου Αλέξανδρος (Σύμβουλος)
Παρπέρης Νικόλας
Παυλικκάς Παύλος (Σύμβουλος)
Πετρή Ιωσήφ
Συμεωνίδης Κωνσταντίνος
Τσιμούρης Χαράλαμπος
Φωτίου Παύλος
Χατζηγιάννης Γιώργος
Χατζηλοΐζου Μαρία
Χατζηπαντελής Δημήτρης

Γ7.1 Εισαγωγή στους Αλγορίθμους

Τι θα μάθουμε σήμερα:

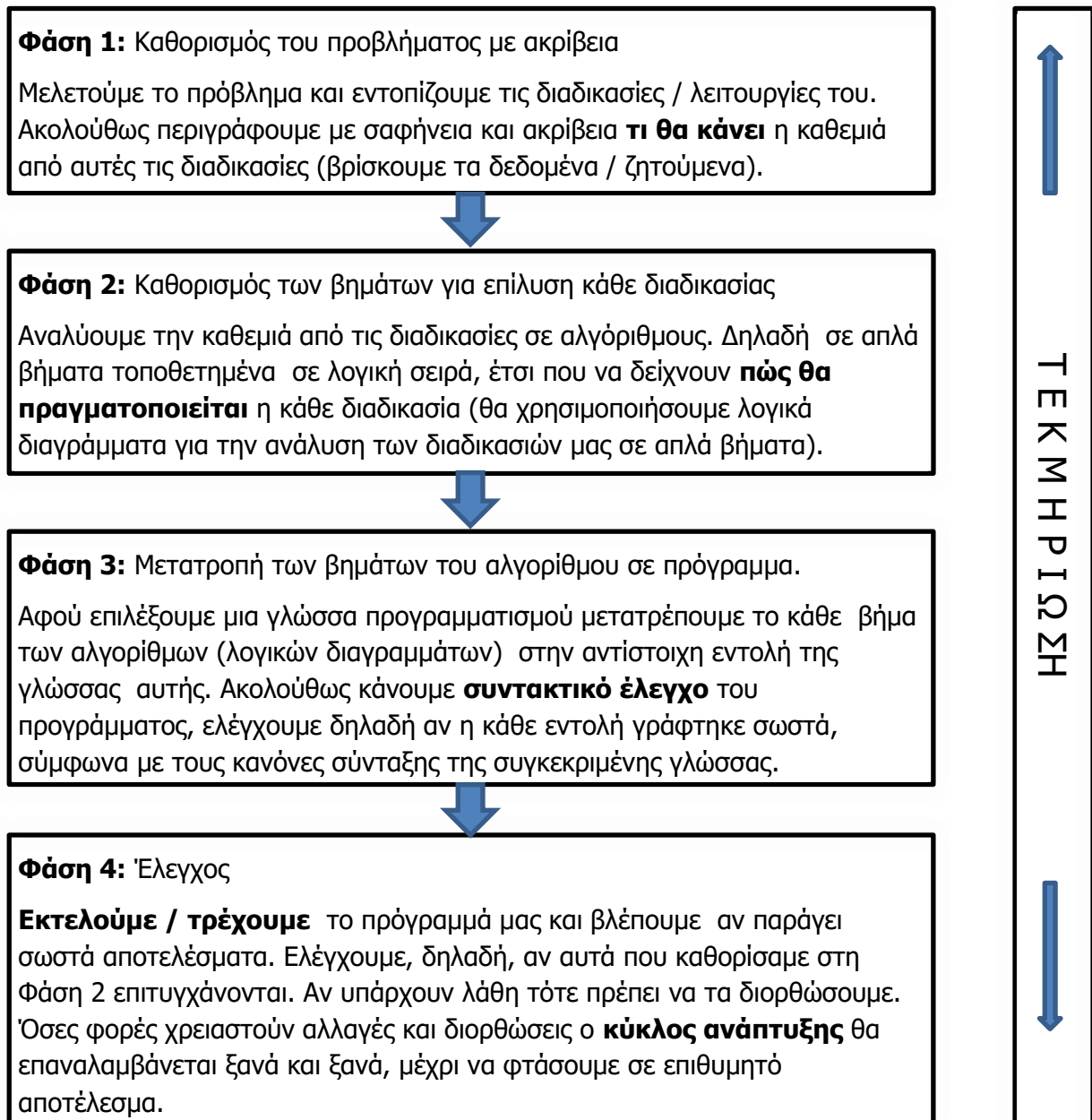
- ◆ Να αναφέρουμε και να εξηγήσουμε τα στάδια ανάπτυξης ενός προγράμματος
- ◆ Να εξηγήσουμε την έννοια του αλγορίθμου
- ◆ Να αναφέρουμε απλούς αλγόριθμους από την καθημερινότητά μας
- ◆ Να αξιολογούμε μια σειρά από βήματα/εντολές και να εξηγήσουμε αν αποτελούν ή όχι αλγόριθμο.
- ◆ Να αξιολογούμε έτοιμα παραδείγματα και να διακρίνουμε σε ποια από αυτά παρατηρούνται ή όχι τα χαρακτηριστικά ενός σωστού αλγορίθμου (Σαφήνεια/καθοριστικότητα, αποτελεσματικότητα, περατότητα)
- ◆ Να αναφέρουμε τρόπους διατύπωσης ενός αλγορίθμου (λεκτική περιγραφή/ψευδοκώδικας, λογικό διάγραμμα)
- ◆ Να αναλύουμε απλά προβλήματα σε αλγόριθμο χρησιμοποιώντας τη λεκτική μέθοδο
- ◆ Να βρίσκουμε τον πιο αποδοτικό αλγόριθμο σε ένα πρόβλημα (π.χ. ταξινόμηση αριθμών με διαφορετικούς τρόπους - πιο σύντομη διαδρομή για ένα οδηγό να φτάσει στον προορισμό του).

1. Ο Κύκλος Ανάπτυξης Εφαρμογής – Ποια βήματα/φάσεις πρέπει να ακολουθήσουμε για να δημιουργήσουμε ένα πρόγραμμα

Τα προγράμματα δημιουργούνται από ανθρώπους και όπως όλες οι τεχνολογικές κατασκευές (π.χ. κτίρια, αυτοκίνητα, γέφυρες, ηλεκτρικές συσκευές), για να λειτουργούν σωστά θα πρέπει να σχεδιαστούν προσεκτικά, να κατασκευαστούν όπως έχουν σχεδιαστεί και να ελεγχθούν πριν τα χρησιμοποιήσουμε. Σκεφτείτε την κατασκευή ενός μεγάλου κτιρίου. Χωρίς διάφορες λεπτομερείς μελέτες, σχεδιαγράμματα από πολιτικούς μηχανικούς, αρχιτέκτονες, ηλεκτρολογικά και πολλά άλλα σχέδια, ο εργολάβος δεν μπορεί καν να ξεκινήσει δουλειά. Για να διορθωθεί μια κατασκευή που δεν έχει σχεδιαστεί σωστά, συνήθως απαιτεί την εξασφάλιση νέων υλικών και πολύ κόπο για την επιδιόρθωσή της, συχνά με ψηλό κόστος, πράγμα που αναγκάζει τον κατασκευαστή να τη σχεδιάσει πολύ προσεκτικά. Το λογισμικό, σε αντίθεση με άλλες κατασκευές, δεν απαιτεί την αγορά υλικών και συχνά δίνεται η εντύπωση ότι δεν απαιτείται σχεδιασμός. Στην πραγματικότητα, όμως, το κόστος εντοπισμού και επίλυσης λαθών στο λογισμικό, ιδιαίτερα εάν δεν έχει σχεδιαστεί σωστά, είναι μια δύσκολη, μακροχρόνια και δαπανηρή διαδικασία.

Βλέπουμε, λοιπόν, ότι όλες οι σοβαρές κατασκευές γίνονται με καλή προεργασία και σχεδιασμό. Το ίδιο συμβαίνει και στην πληροφορική. Για τη δημιουργία ενός προγράμματος (λογισμικού) χρειάζεται να περάσουμε μέσα από μια διαδικασία που περιλαμβάνει διάφορες φάσεις / στάδια τα οποία ονομάζονται Κύκλος Ανάπτυξης Αλγοριθμικής Εφαρμογής.

Στη συνέχεια του κεφαλαίου θα δούμε αναλυτικά όλες τις φάσεις του κύκλου ανάπτυξης και στο τέλος θα μπορούμε να δημιουργήσουμε απλά προγράμματα.



Εικόνα 55 Οι φάσεις / στάδια του Κύκλου Ανάπτυξης Προγράμματος

Στην Εικόνα 55 γίνεται αναφορά σε πολλές διαδικασίες/λειτουργίες. Τα μεγάλα, σύνθετα προγράμματα αποτελούνται από πολλές ανεξάρτητες λειτουργίες. Όλα όμως τα προγράμματα που θα μελετήσουμε στη συνέχεια θα αποτελούνται από μία μόνο διαδικασία/λειτουργία.

Τεκμηρίωση

Τεκμηρίωση είναι το σύνολο του γραπτού υλικού που συνοδεύει ένα πρόγραμμα. Δεν είναι ακόμα μια φάση στην ανάπτυξη ενός προγράμματος, αλλά μία παράλληλη διαδικασία που χρησιμοποιείται σε όλα τα στάδια ανάπτυξης ενός προγράμματος. Μεταξύ άλλων, η τεκμηρίωση, περιλαμβάνει:

Για τον χρήστη: το εγχειρίδιο χρήσης (user manual), το οποίο μας καθοδηγεί με βήμα προς βήμα οδηγίες, ώστε να χρησιμοποιήσουμε το πρόγραμμα.

Για τον προγραμματιστή: περιλαμβάνει όλα τα εργαλεία τα οποία θα κάνουν την ανάπτυξη ενός νέου προγράμματος ή την τροποποίηση ενός υφισταμένου προγράμματος ευκολότερη.

Παράδειγμα

Ένα από τα δημοφιλέστερα παιχνίδια είναι το FIFA που κατασκευάζεται από την εταιρεία *EA Sports*. Στις εκδόσεις του παιχνιδιού πριν από το 2012 παρουσιαζόταν το εξής πρόβλημα. Σε περίπτωση δυναμικής διεκδίκησης της μπάλας (π.χ. τάκλιν) φαινόταν ότι ο ένας παίκτης «έμπαινε» μέσα στον άλλο. Επίσης ο τρόπος με τον οποίον αλληλεπιδρούσαν οι παίκτες στις διάφορες διεκδικήσεις ήταν προβλέψιμος. Τα προβλήματα διορθώθηκαν στο FIFA 2012 με τη δημιουργία του **Impact Engine**, ενός προγράμματος (μιας διαδικασίας/λειτουργίας) το οποίο να διαχειρίζεται την αλληλεπίδραση μεταξύ των παικτών.



Εικόνα 56 Το impact engine του FIFA 2012

Στον παρακάτω πίνακα περιγράφονται με πολύ συνοπτικό τρόπο τα βήματα του κύκλου ανάπτυξης του Impact Engine.

Φάση 1: Η εταιρεία (EA Sports), αναγνώρισε το πρόβλημα. Αυτό έγινε τόσο από το προσωπικό της εταιρείας όσο και από τους χρήστες που άσκησαν κριτική. Αφού έγινε αναγνώριση του προβλήματος, οι αναλυτές της εταιρείας κατέγραψαν με σαφήνεια το **τι** έπρεπε να γίνει.

Φάση 2: Έχοντας μπροστά τους την ανάλυση του προβλήματος, τα άτομα της ομάδας ανάπτυξης του παιχνιδιού άρχισαν να ψάχνουν και να σχεδιάζουν τις λύσεις.

Φάση 3: Σε αυτή τη φάση οι προγραμματιστές της εταιρείας υλοποίησαν τις λύσεις που δόθηκαν από την ομάδα ανάπτυξης.

Φάση 4: Τέλος, ειδικοί gamers ανέλαβαν τον έλεγχο. Έπαιξαν το παιχνίδι πολλές ώρες προσπαθώντας να εντοπίσουν τυχόν λάθη στο Impact Engine.

Είναι φανερό ότι αν μια από τις φάσεις του κύκλου ανάπτυξης δεν σχεδιαστεί σωστά, τότε ολόκληρο το πρόγραμμα θα αποτύχει.

2. Τι είναι αλγόριθμος

Αλγόριθμο ονομάζουμε έναν αριθμό από βήματα γραμμένα σε λογική σειρά, τα οποία χρειάζονται για την εκτέλεση μιας εργασίας ή για την επίλυση ενός προβλήματος.

Η λέξη «αλγόριθμος» προέρχεται από μία μελέτη του Πέρση μαθηματικού του 8ου αιώνα μ.Χ. Αλ Χουαρίζμι (Abu Ja'far Mohammed ibn Musa Al-Khowarismi). Πέντε αιώνες αργότερα η μελέτη μεταφράστηκε στα Λατινικά και άρχισε με τη φράση "Algoritmus dixit" (ο Αλγόριθμος είπε...). Έτσι το όνομά του ταυτίστηκε με τη συστηματική, λεπτομερή και ακριβή περιγραφή του τρόπου επίλυσης κάποιου προβλήματος, η οποία είναι απαραίτητη στη διαδικασία δημιουργίας ενός προγράμματος για τον υπολογιστή.



Εικόνα 57 Ο Abu Ja'far Mohammed ibn Musa Al-Khowarismi σε αιγυπτιακό γραμματόσημο.

2.1 Παραδείγματα αλγορίθμων

Στη ζωή μας, σε καθημερινή βάση, εκτελούμε δεκάδες απλές ή σύνθετες διαδικασίες που μπορούν να περιγραφούν με αλγόριθμους. Οι διαδικασίες αυτές δεν είναι απαραίτητο να σχετίζονται με ηλεκτρονικούς υπολογιστές. Μερικές τέτοιες διαδικασίες (αλγόριθμοι) είναι οι ακόλουθες:

- Ο υπολογισμός του εμβαδού ενός γεωμετρικού σχήματος (π.χ. ενός ορθογωνίου)
- Εύρεση του μεγαλύτερου ή μικρότερου αριθμού από μία ομάδα αριθμών
- Η αποστολή ηλεκτρονικού μηνύματος μέσω του ηλεκτρονικού μας υπολογιστή
- Ο έλεγχος των φώτων της τροχαίας σε μία πόλη.
- Η πορεία που ακολουθεί καθημερινά ένας διανομέας εφημερίδων για να διανείμει τις εφημερίδες
- Ένα ηλεκτρονικό παιχνίδι με πολλά στάδια.

Τρία από τα πιο πάνω παραδείγματα αλγορίθμων αναλύονται σε απλά βήματα πιο κάτω:

Υπολογισμός εμβαδού ορθογωνίου

1. Διάβασε το μήκος ορθογωνίου
2. Διάβασε το πλάτος ορθογωνίου
3. Υπολόγισε το εμβαδόν (Εμβαδόν = μήκος \times πλάτος)
4. Τύπωσε το εμβαδόν

Εύρεση του μεγαλύτερου αριθμού από μια ομάδα 3 διαφορετικών αριθμών

1. Διάβασε τους 3 αριθμούς (a,b,c)
2. Αν $a > b$ και $a > c$ τότε
3. Τύπωσε a
4. Αλλιώς αν $b > a$ και $b > c$ τότε
5. Τύπωσε b
6. Αλλιώς
7. Τύπωσε c

Αποστολή ηλεκτρονικού μηνύματος (e-mail)

1. Μπαίνω στην ιστοσελίδα του παροχέα μου.
2. Πληκτρολογώ το User Name μου
3. Πληκτρολογώ το Password μου
4. Επιλέγω «Νέο Μήνυμα»
5. Πληκτρολογώ ή επιλέγω την ηλεκτρονική διεύθυνση του παραλήπτη
6. Γράφω το μήνυμα
7. Επιλέγω «Αποστολή»
8. Επιλέγω «Έξοδος»

Πρέπει όμως να τονίσουμε ότι μια σειρά από βήματα δεν αποτελούν απαραίτητα αλγόριθμο. Για παράδειγμα:

Οι κανόνες σχολικής συμπεριφοράς στην τάξη.

1. Ακούω προσεκτικά τον/την καθηγητή/τρια μου!
2. Προσφέρω τη βοήθειά μου στους συμμαθητές μου!
3. Σέβομαι και αγαπώ τους συμμαθητές μου!
4. Σηκώνω το χέρι μου όταν θέλω να μιλήσω!
5. Ακούω με προσοχή τον συμμαθητή μου!
6. Ακολουθώ τις οδηγίες του/της καθηγητή/τριας μου!
7. Φροντίζω για την καθαριότητα της τάξης μου!

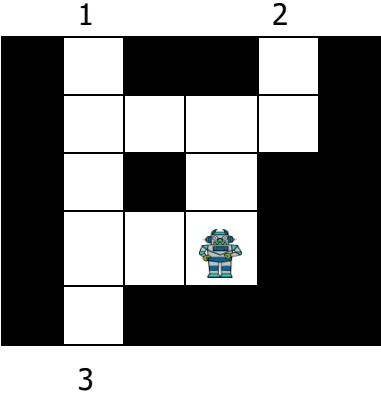
Οι οδηγίες αυτές δεν αποτελούν αλγόριθμο, είναι ασύνδετες μεταξύ τους, δεν έχουν κάποια λογική σειρά και δεν οδηγούν στην παραγωγή κάποιου συγκεκριμένου αποτελέσματος.

3. Τα χαρακτηριστικά ενός αλγορίθμου

Είναι ξεκάθαρο ότι τα βήματα του προηγούμενου παραδείγματος δεν αποτελούν αλγόριθμο. Ένας αλγόριθμος δεν αποτελείται από μια οποιαδήποτε σειρά βημάτων ή εντολών αλλά μία σειρά από βήματα ή εντολές σε **λογική σειρά** και πρέπει επιπρόσθετα να έχει τα πιο κάτω χαρακτηριστικά:

- **Σαφήνεια / Καθοριστικότητα:** Κάθε βήμα (εντολή) του αλγορίθμου πρέπει να είναι απλό και να είναι διατυπωμένο με σαφήνεια, ώστε να μην υπάρχει αμφιβολία για τον τρόπο με τον οποίο θα εκτελεσθεί. Πρέπει, επίσης, να μπορεί να εκτελεσθεί με ακρίβεια μέσα σε συγκεκριμένα χρονικά πλαίσια (πεπερασμένος). Για παράδειγμα, η εντολή «προθερμαίνω τον φούρνο» δεν είναι σαφής, διότι δεν προσδιορίζει για πόσο χρονικό διάστημα προθερμαίνω τον φούρνο.
- **Περατότητα:** Κάθε αλγόριθμος πρέπει πάντα να τελειώνει ύστερα από την εκτέλεση ενός αριθμού βημάτων, με άλλα λόγια δεν μπορεί να συνεχίζει την εκτέλεσή του για πάντα.
- **Αποτελεσματικότητα:** Να δίνει συγκεκριμένα αποτελέσματα, τα οποία να μπορούν να εξακριβωθούν από έναν άνθρωπο, εκτελώντας με «μολύβι και χαρτί» τις εντολές / βήματα του αλγορίθμου. Επίσης οι εντολές αυτές θα πρέπει να είναι διατυπωμένες έτσι ώστε να μπορούν να εκτελεστούν από αυτόν στον οποίο απευθύνεται ο αλγόριθμος.

Ο πιο κάτω πίνακας παρουσιάζει μερικά παραδείγματα «αλγορίθμων» που δεν έχουν όλα τα πιο πάνω χαρακτηριστικά.

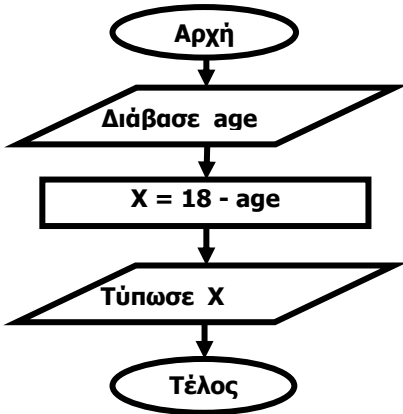
| | |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Βρες τον μικρότερο από 3 αριθμούς</p> <ol style="list-style-type: none"> 1. Διαβάζω αριθμούς 2. Βρίσκω τον μικρότερο 3. Τυπώνω τον μικρότερο | <p>Πρόβλημα σαφήνειας/καθοριστικότητας</p> <p>(Πόσους αριθμούς διαβάζω; Πώς βρίσκω τον μικρότερο;)</p> |
| <p>Το ρομπότ που βλέπετε προσπαθεί να βγει από τον λαβύρινθο σε μια από τις εξόδους 1, 2 ή 3. Αν καταφέρει να βγει τότε ο αλγόριθμος τερματίζεται. Μπορεί να κινείται μόνο μέσα από άσπρα κουτάκια και είναι προγραμματισμένο να κινείται ως εξής:</p> <ol style="list-style-type: none"> 1. Κάνε επιτόπου στροφή δεξιά 2. Προχώρα 2 κουτιά ευθεία μπροστά 3. Αν βγεις σε κάποια από τις 3 εξόδους σταμάτα. Διαφορετικά επανέλαβε τα βήματα 1 και 2. <div style="text-align: center;">  </div> | <p>Πρόβλημα περατότητας</p> <p>(Ο αλγόριθμος δεν θα σταματήσει ποτέ γιατί το ρομπότ θα κάνει κύκλους μέσα στον λαβύρινθο και δεν θα καταφέρει να βγει σε μια από τις εξόδους).</p> |
| <p>Υπολογισμός μέσου όρου 20 αριθμών</p> <ol style="list-style-type: none"> 1. Διάβασε 20 αριθμούς 2. Υπολόγισε τον μέσο όρο 3. Τύπωσε τον μέσο όρο. | <p>Πρόβλημα αποτελεσματικότητας</p> <p>Η 2^η εντολή δεν είναι απλή και δεν μπορεί να εκτελεστεί απευθείας από τον υπολογιστή. Πρέπει να υπολογιστεί πρώτα το άθροισμα των 20 αριθμών και μετά ο μέσος τους όρος.</p> |

4. Τρόποι διατύπωσης αναπαράστασης αλγορίθμων

Έναν αλγόριθμο μπορούμε να τον διατυπώσουμε με τρεις τρόπους: Λεκτική Περιγραφή, Ψευδοκώδικα (pseudocode), και Λογικό Διάγραμμα (flowchart).

- Λεκτική περιγραφή: Περιγραφή του αλγορίθμου σε βήματα χρησιμοποιώντας λέξεις από την καθημερινή μας ζωή.
- Ψευδοκώδικας: Πιο αυστηρή διατύπωση του αλγορίθμου κατά την οποία χρησιμοποιούνται καθορισμένες εντολές/λέξεις και συμβολισμοί.
- Λογικό Διάγραμμα: Ένας γραφικός τρόπος παρουσίασης ενός αλγορίθμου, που αποτελείται από σύμβολα με συγκεκριμένη σημασία, που συνδέονται μεταξύ τους με βέλη, που δείχνουν τη ροή της εκτέλεσης.

Παράδειγμα : Σε πόσα χρόνια ένας ανήλικος θα δικαιούται να βγάλει δίπλωμα οδήγησης;

| Λεκτική Περιγραφή | Ψευδοκώδικας | Λογικό Διάγραμμα |
|-------------------------------------------------------------------------------------------|-----------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. Διάβασε την ηλικία 2. Αφαίρεσε την ηλικία από τον αριθμό 18 3. Τύπωσε τη διαφορά | 1. Διάβασε age 2. $X = 18 - \text{age}$ 3. Τύπωσε X |  <pre> graph TD A([Αρχή]) --> B[/Διάβασε age/] B --> C[X = 18 - age] C --> D[/Τύπωσε X/] D --> E([Τέλος]) </pre> |

5. Αποδοτικός Αλγόριθμος

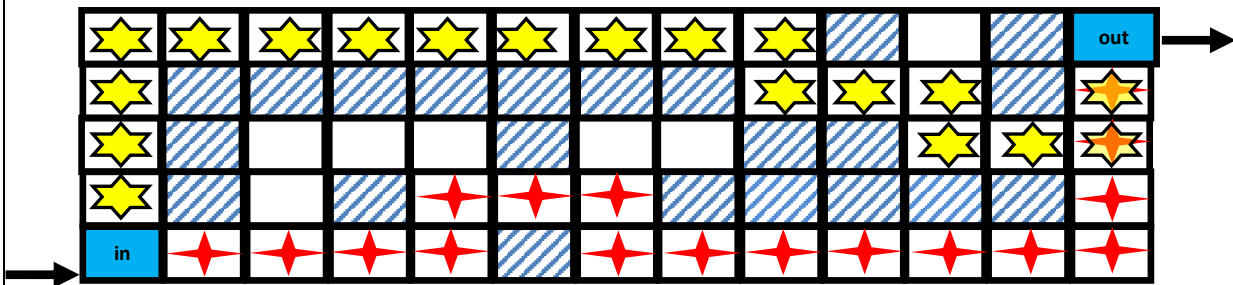
Συνήθως παρατηρούμε ότι πολλά προβλήματα της καθημερινότητάς μας επιλύονται με περισσότερους από έναν τρόπους. Για παράδειγμα:

- Για να περπατήσουμε από το σπίτι μέχρι το σχολείο μας. Συνήθως, υπάρχουν περισσότερες από μια διαδρομές που μπορούμε να ακολουθήσουμε.
- Η λύση ενός προβλήματος στα μαθηματικά. Πολύ συχνά βλέπουμε ότι κάποιος συμμαθητής μας έλυσε την ίδια άσκηση με διαφορετικό τρόπο από τον δικό μας.

Σε όλες τις περιπτώσεις είναι λογικό να ψάχνουμε για την καλύτερη και αποδοτικότερη λύση. Έτσι, θα ακολουθήσουμε τη συντομότερη διαδρομή για να πάμε στο σχολείο μας και θα λύσουμε την άσκηση των μαθηματικών με τον τρόπο που θεωρούμε ευκολότερο. Παρομοίως στην πληροφορική μπορεί να υπάρχουν περισσότεροι από ένας αλγόριθμοι για τη λύση του ίδιου προβλήματος. Ένας αλγόριθμος είναι πιο **αποδοτικός** από κάποιον άλλον αν εκτελείται γρηγορότερα και κάνει χρήση όσο το δυνατόν λιγότερου χώρου μνήμης.

Πιο κάτω παρουσιάζονται τρία παραδείγματα προβλημάτων. Για το κάθε πρόβλημα προτείνονται δύο διαφορετικοί αλγόριθμοι οι οποίοι αποτελούν λύσεις στο πρόβλημα. Όπως εξηγείται, αν συγκρίνουμε τους αλγόριθμους αυτούς συχνά κάποιος είναι αποδοτικότερος από τον άλλον.

Παράδειγμα 1 : για να εισέλθει κάποιος από την είσοδο (in) και να εξέλθει από την έξοδο (out) έχει δύο επιλογές (α) ακολουθώντας τη διαδρομή με τα αστεράκια, (β) ακολουθώντας τη διαδρομή με τους σταυρούς. Ποια επιλογή / διαδρομή είναι πιο αποδοτική;



Απάντηση: η διαδρομή (β) γιατί είναι συντομότερη 17 βήματα ενώ η (α) χρειάζεται 19.

Παράδειγμα 2: Ετοιμασία μέτριου frappe. Ποιος από τους δύο αλγόριθμους είναι αποδοτικότερος;

- | | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ol style="list-style-type: none"> 1. Βάζουμε λίγο νερό σε ένα ποτήρι 2. Βάζουμε 1 κουταλάκι στιγμιαίου καφέ 3. Βάζουμε 1 κουταλάκι ζάχαρη 4. Ανακατεύουμε με το μίξερ 5. Προσθέτουμε και άλλο νερό 6. Βάζουμε λίγο γάλα 7. Βάζουμε 2-3 παγάκια | <ol style="list-style-type: none"> 1. Βάζουμε λίγο νερό σε ένα ποτήρι 2. Βάζουμε 1 κουταλάκι ζάχαρη 3. Βάζουμε 1 κουταλάκι στιγμιαίου καφέ 4. Ανακατεύουμε με το μίξερ 5. Προσθέτουμε και άλλο νερό 6. Βάζουμε 2-3 παγάκια 7. Βάζουμε λίγο γάλα |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Απάντηση: κανένας, και οι δύο έχουν την ίδια απόδοση, απλά κάποια από τα βήματα τους δεν είναι στην ίδια σειρά.

Παράδειγμα 3: εξεύρεση του ελάχιστου κοινού πολλαπλάσιου (ΕΚΠ) δύο ακεραίων αριθμών a και β όπου $a \leq \beta$ (για παράδειγμα το ΕΚΠ των αριθμών 3 και 4 είναι το 12).

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|---------------|----------|-----------|----------|----------------------------------------------------------|----------|-----------------------------------------------------------|----------|----------------------------------------------------|----------|-----|--|--------------------------------------------------------------|----------|---------------------------------------------------------------|----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|----------|---------------|----------|-----------|----------|----------------------------------------------------------|----------|------------------------------------------------------------|----------|--------------------------------------------------------------|----------|
| <ol style="list-style-type: none"> 1. Διάβασε έναν αριθμό a 2. Διάβασε ένα μεγαλύτερο αριθμό β 3. Όρισε τον ακέραιο X ίσο με β 4. Αν ο X διαιρείται ακριβώς και από τους δύο ακεραίους a και β τότε τύπωσε τον X (που είναι το ΕΚΠ) και σταμάτα διαφορετικά αύξησε τον X κατά μια μονάδα και επανέλαβε το βήμα 4. | <ol style="list-style-type: none"> 1. Διάβασε έναν αριθμό a 2. Διάβασε ένα μεγαλύτερο αριθμό β 3. Όρισε τον ακέραιο X ίσο με β 4. Αν ο X διαιρείται ακριβώς και από τους δύο ακεραίους a και β τότε τύπωσε τον X (που είναι το ΕΚΠ) και σταμάτα διαφορετικά αύξησε τον X κατά β μονάδες και επανέλαβε το βήμα 4. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <p>Π.χ Είσοδος $a=3$ $\beta=4$</p> <table> <tbody> <tr><td>(1) $a=3$</td><td>(Βήμα 1)</td></tr> <tr><td>(2) $\beta=4$</td><td>(Βήμα 2)</td></tr> <tr><td>(3) $X=4$</td><td>(Βήμα 3)</td></tr> <tr><td>(4) 4 δεν διαιρείται με 3, αλλά με 4 άρα $X=5$</td><td>(Βήμα 4)</td></tr> <tr><td>(5) 5 δεν διαιρείται με 3, ούτε με 4, άρα $X=6$</td><td>(Βήμα 4)</td></tr> <tr><td>(6) 6 διαιρείται με 3, αλλά όχι με 4, άρα $X=7$</td><td>(Βήμα 4)</td></tr> <tr><td>...</td><td></td></tr> <tr><td>(11) 10 δεν διαιρείται με 3, ούτε με 4, άρα $X=12$</td><td>(Βήμα 4)</td></tr> <tr><td>(12) 12 διαιρείται με 3 και με 4, άρα ΕΚΠ=12</td><td>(Βήμα 4)</td></tr> </tbody> </table> <p>Εδώ χρειάζονται 12 βήματα.</p> | (1) $a=3$ | (Βήμα 1) | (2) $\beta=4$ | (Βήμα 2) | (3) $X=4$ | (Βήμα 3) | (4) 4 δεν διαιρείται με 3, αλλά με 4 άρα $X=5$ | (Βήμα 4) | (5) 5 δεν διαιρείται με 3, ούτε με 4, άρα $X=6$ | (Βήμα 4) | (6) 6 διαιρείται με 3, αλλά όχι με 4, άρα $X=7$ | (Βήμα 4) | ... | | (11) 10 δεν διαιρείται με 3, ούτε με 4, άρα $X=12$ | (Βήμα 4) | (12) 12 διαιρείται με 3 και με 4, άρα ΕΚΠ=12 | (Βήμα 4) | <p>Π.χ Είσοδος $a=3$ $\beta=4$</p> <table> <tbody> <tr><td>(1) $a=3$</td><td>(Βήμα 1)</td></tr> <tr><td>(2) $\beta=4$</td><td>(Βήμα 2)</td></tr> <tr><td>(3) $X=4$</td><td>(Βήμα 3)</td></tr> <tr><td>(4) 4 δεν διαιρείται με 3, αλλά με 4 άρα $X=9$</td><td>(Βήμα 4)</td></tr> <tr><td>(5) 9 διαιρείται με 3, αλλά όχι με 4, άρα $X=12$</td><td>(Βήμα 4)</td></tr> <tr><td>(6) 12 διαιρείται με 3 και με 4, άρα ΕΚΠ=12</td><td>(Βήμα 4)</td></tr> </tbody> </table> <p>Εδώ χρειάζονται μόνο 6 βήματα. Άρα ο αλγόριθμος είναι πιο αποδοτικός. (Η διαφορά γίνεται ακόμη πιο μεγάλη εάν οι αριθμοί a και β είναι μεγαλύτεροι, αφού το βήμα 4 επαναλαμβάνεται πολύ περισσότερες φορές στον αλγόριθμο αριστερά παρά στον αλγόριθμο πιο πάνω.)</p> | (1) $a=3$ | (Βήμα 1) | (2) $\beta=4$ | (Βήμα 2) | (3) $X=4$ | (Βήμα 3) | (4) 4 δεν διαιρείται με 3, αλλά με 4 άρα $X=9$ | (Βήμα 4) | (5) 9 διαιρείται με 3, αλλά όχι με 4, άρα $X=12$ | (Βήμα 4) | (6) 12 διαιρείται με 3 και με 4, άρα ΕΚΠ=12 | (Βήμα 4) |
| (1) $a=3$ | (Βήμα 1) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (2) $\beta=4$ | (Βήμα 2) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (3) $X=4$ | (Βήμα 3) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (4) 4 δεν διαιρείται με 3, αλλά με 4 άρα $X=5$ | (Βήμα 4) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (5) 5 δεν διαιρείται με 3, ούτε με 4, άρα $X=6$ | (Βήμα 4) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (6) 6 διαιρείται με 3, αλλά όχι με 4, άρα $X=7$ | (Βήμα 4) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (11) 10 δεν διαιρείται με 3, ούτε με 4, άρα $X=12$ | (Βήμα 4) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (12) 12 διαιρείται με 3 και με 4, άρα ΕΚΠ=12 | (Βήμα 4) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (1) $a=3$ | (Βήμα 1) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (2) $\beta=4$ | (Βήμα 2) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (3) $X=4$ | (Βήμα 3) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (4) 4 δεν διαιρείται με 3, αλλά με 4 άρα $X=9$ | (Βήμα 4) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (5) 9 διαιρείται με 3, αλλά όχι με 4, άρα $X=12$ | (Βήμα 4) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| (6) 12 διαιρείται με 3 και με 4, άρα ΕΚΠ=12 | (Βήμα 4) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Βασικές Έννοιες

Κύκλος Ανάπτυξης μιας Εφαρμογής/ενός Προγράμματος:

Τα βήματα/φάσεις που πρέπει να ακολουθήσουμε για να δημιουργήσουμε ένα πρόγραμμα.

1^η Φάση—Καθορισμός του προβλήματος με ακρίβεια:

Μελετούμε προσεκτικά το πρόβλημά μας για να εντοπίσουμε τις διαδικασίες/λειτουργίες που το αποτελούν και περιγράφουμε με σαφήνεια και ακρίβεια **τι θα κάνει** η καθεμιά από αυτές τις διαδικασίες, ποιά **δεδομένα** δέχεται και ποιές **πληροφορίες (ζητούμενα)** δημιουργεί.

2^η Φάση—Καθορισμός των βημάτων επίλυσης της κάθε διαδικασίας:

Αναλύουμε την κάθε διαδικασία σε **αλγόριθμο** ώστε να διατυπώνεται με σαφήνεια και ακρίβεια **πώς θα πραγματοποιείται** η κάθε διαδικασία.

| | |
|---------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 3^η Φάση–Μετατροπή των βημάτων σε πρόγραμμα: | Μετατρέπουμε τον αλγόριθμο σε πρόγραμμα, μεταφράζοντάς τον σε σειρές από εντολές σε μια γλώσσα προγραμματισμού. |
| 4^η Φάση–Έλεγχος προγράμματος: | Εκτελούμε το πρόγραμμα και ελέγχουμε κατά πόσο κάνει αυτά που καταγράψαμε στη 2 ^η φάση. |
| Τεκμηρίωση: | Είναι το σύνολο του γραπτού υλικού που συνοδεύει ένα πρόγραμμα. Περιλαμβάνει οδηγίες για τον χρήστη και σημαντικές πληροφορίες για τον προγραμματιστή. |
| Αλγόριθμος: | Είναι ένας αριθμός από βήματα ή εντολές, γραμμένα σε λογική σειρά, τα οποία χρειάζονται για την εκτέλεση μιας εργασίας ή για την επίλυση ενός προβλήματος. |
| Χαρακτηριστικά Αλγορίθμου: | <p>Σαφήνεια/Καθοριστικότητα: Κάθε βήμα (εντολή) του αλγορίθμου πρέπει να είναι απλό και να είναι διατυπωμένο με σαφήνεια. Επίσης, πρέπει να εκτελείται με ακρίβεια μέσα σε συγκεκριμένα πλαίσια.</p> <p>Περατότητα: Κάθε αλγόριθμος πρέπει πάντα να τελειώνει ύστερα από την εκτέλεση ενός αριθμού βημάτων, με άλλα λόγια δεν μπορεί να συνεχίζει την εκτέλεσή του για πάντα.</p> <p>Αποτελεσματικότητα: Να δίνει συγκεκριμένα αποτελέσματα, τα οποία να μπορούν να εξακριβωθούν από έναν άνθρωπο, εκτελώντας με «μολύβι και χαρτί» τις εντολές / βήματα του αλγορίθμου. Επίσης οι εντολές αυτές θα πρέπει να είναι διατυπωμένες έτσι ώστε να μπορούν να εκτελεστούν από αυτόν στον οποίο απευθύνεται ο αλγόριθμος.</p> |
| Λεκτική περιγραφή: | Περιγραφή ενός αλγορίθμου σε βήματα χρησιμοποιώντας λέξεις από την καθημερινή μας ζωή. |
| Ψευδοκώδικας: | Πιο αυστηρή διατύπωση του αλγορίθμου όπου χρησιμοποιούνται καθορισμένες εντολές/λέξεις και συμβολισμοί. |
| Λογικό Διάγραμμα: | Ένας γραφικός τρόπος παρουσίασης ενός αλγορίθμου, με γεωμετρικά σχήματα. |






Γ7.2 Αναπαράσταση Αλγορίθμων με Λογικό Διάγραμμα

Τι θα μάθουμε σήμερα:

- ◆ Να αναλύουμε ένα πρόβλημα και να καταγράφουμε τα Δεδομένα, τα Ζητούμενα (Πληροφορίες) και την Επεξεργασία
- ◆ Να σχεδιάζουμε τα σύμβολα που χρησιμοποιούνται σε ένα λογικό διάγραμμα και να εξηγούμε τη χρήση του καθενός από αυτά
- ◆ Να χρησιμοποιούμε το κατάλληλο σχήμα για την αναπαράσταση μιας εντολής
- ◆ Να διακρίνουμε σταθερές και μεταβλητές ανάμεσα στα βήματα του αλγορίθμου.

1. Λογικά Διαγράμματα

Στο προηγούμενο κεφάλαιο εξετάσαμε την έννοια του αλγορίθμου και αναφέραμε διάφορους τρόπους αναπαράστασης του. Η μέθοδος που θα χρησιμοποιήσουμε στα επόμενα μαθήματα είναι αυτή των λογικών διαγραμμάτων. Τα **λογικά διαγράμματα (flowcharts)** είναι ένας σχηματικός τρόπος για να αναπαριστούμε τη λύση ενός προβλήματος. Αποτελούνται από ένα σύνολο γεωμετρικών σχημάτων, όπου το καθένα δηλώνει μία συγκεκριμένη ενέργεια. Τα γεωμετρικά σχήματα ενώνονται μεταξύ τους με βέλη, που δηλώνουν τη σειρά εκτέλεσης των ενεργειών αυτών. Τα κυριότερα γεωμετρικά σχήματα που χρησιμοποιούμε είναι τα εξής:

| Σύμβολο | Χρήση |
|-------------------------------------------------------------------------------------|----------------------------------------|
|  | Αρχή / Τέλος |
|  | Είσοδος Δεδομένων / Έξοδος Πληροφοριών |
|  | Πράξη / Επεξεργασία |
|  | Ερώτηση/ Απόφαση |
|  | Πορεία Ενεργειών |

Τα κυριότερα πλεονεκτήματα των λογικών διαγραμμάτων είναι η περιεκτικότητά τους, δηλαδή η επεξήγηση πολύπλοκων εννοιών με σύντομο και απλό τρόπο, καθώς και η ανεξαρτησία που παρέχουν όσον αφορά τη γλώσσα, αφού είναι εύκολο να γίνουν κατανοητά ακόμα και μεταξύ ατόμων που μιλούν διαφορετική γλώσσα. Επίσης, ο καθένας από εμάς θα περιγράψει τη λύση ενός προβλήματος με διαφορετικό τρόπο, αν χρησιμοποιήσει φυσική γλώσσα. Με τα λογικά διαγράμματα υπάρχει μια δομή που πρέπει να ακολουθείται και μας δίνει συγκεκριμένα αποτελέσματα.

2. Δεδομένα, Επεξεργασία, Πληροφορίες (Ζητούμενα),

Έχουμε αναφέρει ότι το πρώτο στάδιο ανάπτυξης ενός προγράμματος είναι η κατανόηση του προβλήματος. Η κατανόηση του προβλήματος είναι η σωστή και πλήρης αποσαφήνιση των δεδομένων και ζητούμενων του προβλήματος.

Η πρώτη μας ενέργεια λοιπόν για να επιλύσουμε ένα πρόβλημα, είναι η καταγραφή των δεδομένων. Τα δεδομένα είναι τα στοιχεία που μας είναι γνωστά και μπορούν να μας βοηθήσουν στη λύση του προβλήματος.

Σε κάθε πρόβλημα ψάχνουμε να βρούμε την απάντηση σε μια ερώτηση. Αυτό που ψάχνουμε είναι το ζητούμενο (πληροφορίες).

Η διαδικασία μέσω της οποίας βρίσκουμε τις πληροφορίες από τα δεδομένα, ονομάζεται επεξεργασία.



3. Μεταβλητές

Έχουμε αναφέρει ότι ένα από τα χαρακτηριστικά ενός αλγορίθμου είναι η γενικότητα, δηλαδή να μπορεί να επιλύσει παρόμοιας φύσης προβλήματα. Αν θέλω για παράδειγμα να δημιουργήσω ένα πρόγραμμα που θα βρίσκει το εμβαδόν ενός τετραγώνου, θα πρέπει το πρόγραμμά μου να μπορεί να υπολογίσει το αποτέλεσμα ανεξάρτητα από το μήκος της πλευράς που θα του δίνω. Είναι φανερό ότι για διαφορετικές τιμές της πλευράς θα έχω και διαφορετικά αποτελέσματα. Στο παρόν στάδιο μπορούμε να ονομάσουμε μεταβλητή σε έναν αλγόριθμο τα δεδομένα/πληροφορίες που αλλάζουν (μεταβάλλονται) κατά τη διάρκεια της εκτέλεσης του αλγορίθμου. Σε μεταγενέστερο στάδιο θα δούμε ότι οι μεταβλητές είναι θέσεις στην κύρια μνήμη του υπολογιστή μας που αποθηκεύουν προσωρινά δεδομένα/πληροφορίες.

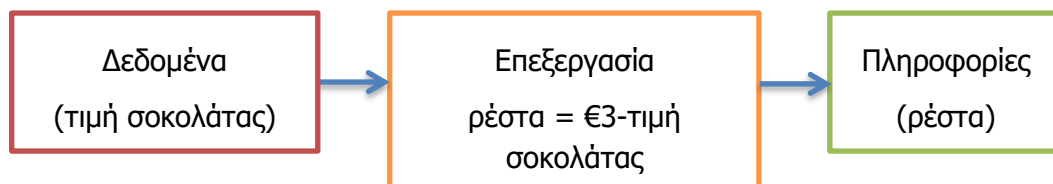
4. Αναπαράσταση αλγορίθμου με τη χρήση λογικών διαγραμμάτων

Τα βήματα που πρέπει να ακολουθήσουμε για να σχεδιάσουμε ένα λογικό διάγραμμα είναι τα ακόλουθα:

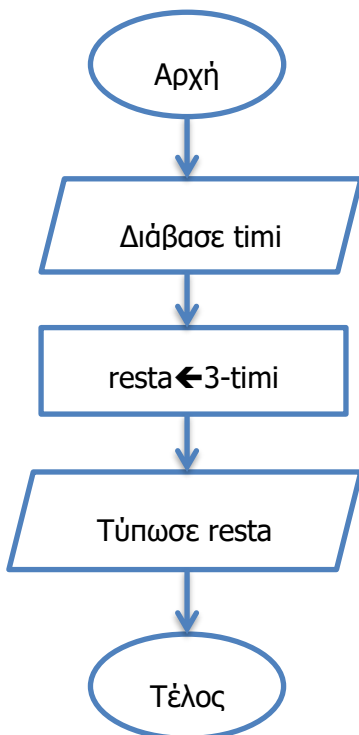
- (1) Τα λογικά διαγράμματα πρέπει πάντα να έχουν μία αρχή και ένα τέλος.
- (2) Βρίσκουμε τα δεδομένα (εισόδους) και χρησιμοποιούμε το κατάλληλο σχήμα (παραλληλόγραμμο γράφοντας μέσα τη λέξη **Διάβασε**).
- (3) Βρίσκουμε τις διαδικασίες (αριθμητικές πράξεις – αποφάσεις) που πρέπει να χρησιμοποιήσουμε για να πάρουμε τις πληροφορίες/ ζητούμενα.
- (4) Παρουσιάζουμε τις πληροφορίες (παραλληλόγραμμο γράφοντας μέσα τη λέξη **Τύπωσε**).

Παράδειγμα 1

Βρίσκεστε στην καντίνα του σχολείου σας και θέλετε να αγοράσετε μια σοκολάτα. Ο καντινιέρης σας πληροφορεί για την τιμή της σοκολάτας. Στο πορτοφόλι σας έχετε €3. Πόσα ρέστα θα πάρετε;



Μετατρέποντας το πιο πάνω σχήμα σε λογικό διάγραμμα έχουμε:



Μερικές παρατηρήσεις για το λογικό διάγραμμα:

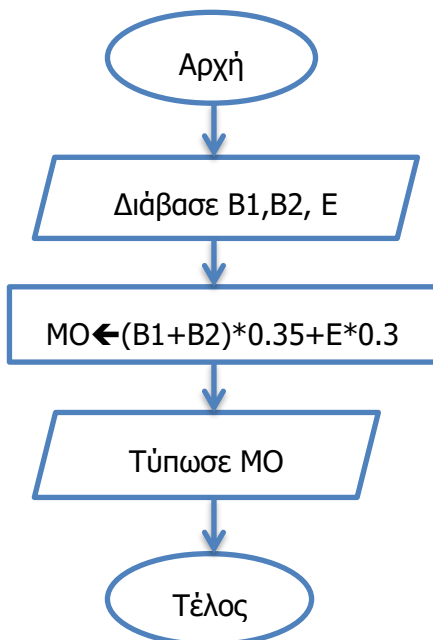
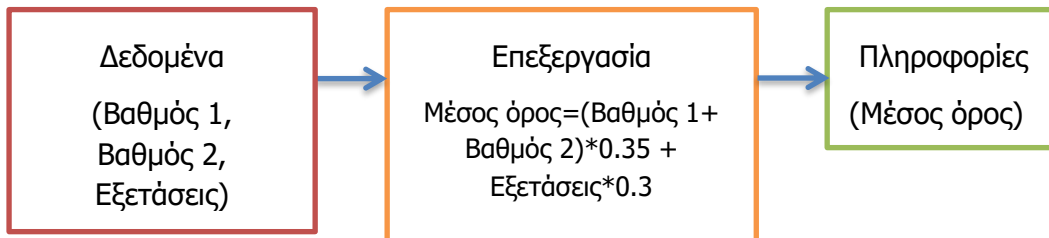
- (1) Για σκοπούς ευκολίας αντικαταστήσαμε το «τιμή σοκολάτας» με τη λέξη *timi*
- (2) Οι μόνες λέξεις στα ελληνικά που χρησιμοποιούμε είναι Αρχή, Τέλος, Διάβασε, Τύπωσε. Για οτιδήποτε άλλο χρησιμοποιούμε λατινικούς χαρακτήρες. Θα μπορούσαμε ακόμα να αντικαταστήσουμε τις ελληνικές λέξεις με τις λέξεις Start, End, Read, Print/Write.
- (3) Το \leftarrow μπορεί να αντικατασταθεί από το =
- (4) Στη θέση των λέξεων *timi* και *resta* θα μπορούσαμε να βάλουμε οτιδήποτε (π.χ. X, Y). Είναι όμως καλή ιδέα να χρησιμοποιούμε λέξεις που περιγράφουν τα δεδομένα και τις πληροφορίες.
- (5) Μεταβλητές του αλγορίθμου μας είναι η *timi* και τα *resta*.

Παράδειγμα 2

Ο μέσος όρος στο μάθημα των Νέων Ελληνικών υπολογίζεται ως εξής:

Μέσος όρος= (Βαθμός 1^{ου} τετραμήνου + Βαθμός 2^{ου} τετραμήνου) * 0.35 + Βαθμός εξετάσεων * 0.3

Η καθηγήτρια σάς ενημερώνει για τους βαθμούς σας. Ποιος θα είναι ο μέσος όρος σας;





Για να σχεδιάσουμε το λογικό διάγραμμα βρίσκουμε τα δεδομένα και τα ζητούμενα. Στο πρόβλημά μας δεδομένα είναι οι βαθμοί των τετραμήνων και ο βαθμός των εξετάσεων. Αυτά είναι και τα στοιχεία που θα **διαβάσει** το πρόγραμμά μας. Ζητούμενο είναι ο μέσος όρος. Για να υπολογίσουμε τον μέσο όρο πρέπει να χρησιμοποιήσουμε τον μαθηματικό τύπο που μας έχει δοθεί στην εκφώνηση. Ο μέσος όρος είναι η πληροφορία που θα **τυπωθεί**.

Μεταβλητές του πιο πάνω αλγορίθμου είναι οι 3 βαθμοί (Β1, Β2, Ε) και ο μέσος όρος (ΜΟ).

Βασικές Έννοιες**Λογικό
Διάγραμμα:**

Είναι ένας σχηματικός τρόπος για την αναπαράσταση ενός αλγορίθμου. Στα λογικά διαγράμματα χρησιμοποιούμε τα παρακάτω σχήματα:

| Σύμβολο | Χρήση |
|-----------------------------------------------------------------------------------|----------------------------------------|
|  | Αρχή / Τέλος |
|  | Είσοδος δεδομένων / Έξοδος πληροφοριών |
|  | Πράξη / Επεξεργασία |
|  | Ερώτηση/Απόφαση |
|  | Πορεία εκτέλεσης |

Δεδομένα:

Τα στοιχεία σε ένα πρόβλημα που μας είναι γνωστά και μπορούν να μας βοηθήσουν στη λύση του προβλήματος.

Επεξεργασία:

Η διαδικασία μέσω της οποίας βρίσκουμε τις πληροφορίες.

Πληροφορίες:

Τα ζητούμενα του προβλήματος. Προκύπτουν μετά την επεξεργασία των δεδομένων.

Μεταβλητή:

Μεταβλητή σε έναν αλγόριθμο ονομάζουμε ένα στοιχείο (δεδομένο/ πληροφορία) που δεν έχει σταθερή τιμή. Όταν αναφερόμαστε στον όρο μεταβλητή για ένα πρόγραμμα, εννοούμε τη θέση μνήμης που δεσμεύεται στην κύρια μνήμη του υπολογιστή (RAM) και χρησιμοποιείται για να αποθηκεύει προσωρινά δεδομένα και πληροφορίες του προγράμματός μας.

Γ7.3 Προβλήματα Ακολουθιακής Δομής

Τι θα μάθουμε σήμερα:

- ◆ Να προτείνουμε τη λύση ενός προβλήματος ακολουθιακής δομής, χρησιμοποιώντας λογικό διάγραμμα
- ◆ Να χρησιμοποιούμε την προκαταρκτική εκτέλεση για να ελέγξουμε την ορθότητα του αλγορίθμου μας.

1. Ακολουθιακή Δομή

Η ακολουθιακή δομή χρησιμοποιείται για την αντιμετώπιση απλών προβλημάτων, όπου είναι δεδομένη η σειρά εκτέλεσης ενός συνόλου ενεργειών/ εντολών. Η πρώτη ενέργεια που πρέπει να εκτελέσει ένας αλγόριθμος σε αυτά τα προβλήματα είναι η είσοδος δεδομένων. Ακολουθεί η επεξεργασία των δεδομένων και τέλος γίνεται η παρουσίαση των πληροφοριών.

Ένα παράδειγμα ακολουθιακής δομής από την καθημερινή ζωή είναι η εκτέλεση μιας συνταγής για να φτιάξουμε ένα φαγητό. Παίρνουμε τα υλικά (δεδομένα), τα μαγειρεύουμε (επεξεργασία) και έχουμε έτοιμο το φαγητό (αποτέλεσμα/ πληροφορία).

2. Προκαταρκτική Εκτέλεση

Προκαταρκτική εκτέλεση είναι η διαδικασία στην οποία εκτελούμε τις εντολές του αλγορίθμου, όπως θα τις εκτελούσε ο ηλεκτρονικός υπολογιστής. Σε κάθε βήμα σημειώνουμε τις τιμές των μεταβλητών ή των αποφάσεων που αλλάζουν. Χρησιμοποιούμε συνήθως ένα πίνακα στον οποίο καταχωρούμε:

- (1) Τις αρχικές τιμές των μεταβλητών καθώς και τις τιμές τους μετά από κάθε αλλαγή.
- (2) Τις αποφάσεις και τις τιμές τους όταν αλλάζουν.
- (3) Την παρουσίαση των μηνυμάτων και των αποτελεσμάτων του αλγορίθμου. Δηλαδή τι θα δούμε στην οθόνη του υπολογιστή, τις πληροφορίες που παράγονται στην έξοδο.

Παράδειγμα

Θα γράψουμε την προκαταρκτική εκτέλεση του λογικού διαγράμματος που δόθηκε στο προηγούμενο κεφάλαιο και υπολόγιζε τον μέσο όρο στο μάθημα των Νέων Ελληνικών. Όπως αναφέραμε ήδη οι μεταβλητές μας είναι οι βαθμοί των 2 τετραμήνων, των εξετάσεων και ο μέσος όρος. Για να ελέγξουμε τον αλγόριθμό μας θέτουμε σαν αρχικές τιμές $B1=20$, $B2=19$ και $E=18$. Δημιουργούμε τον παρακάτω πίνακα και τον συμπληρώνουμε εκτελώντας τον αλγόριθμο μας βήμα προς βήμα.

| Μεταβλητές | | | | Παρουσίαση |
|------------|----|----|-------|------------|
| B1 | B2 | E | MO | 19.05 |
| 20 | 19 | 18 | | |
| | | | 19.05 | |

3. Μαθηματικές εκφράσεις

Οι μαθηματικές πράξεις στην πληροφορική συμβολίζονται με συγκεκριμένα σύμβολα.

| Μαθηματική πράξη | Μαθηματικός Συμβολισμός | Συμβολισμός στους αλγορίθμους | Παράδειγμα | Αποτέλεσμα |
|------------------|-------------------------|-------------------------------|------------|------------|
| Πρόσθεση | $A + B$ | $A + B$ | $5+2$ | 7 |
| Αφαίρεση | $A - B$ | $A - B$ | $7 - 3$ | 4 |
| Πολλαπλασιασμός | $A \times B$ | $A * B$ | $5*3$ | 15 |
| Διαίρεση | $A \div B$ | A / B | $5/2$ | 2.5 |

Κάποιες πράξεις έχουν μεγαλύτερη προτεραιότητα από άλλες. Αυτό σημαίνει ότι όταν έχουμε έναν υπολογισμό με πολλές πράξεις δεν ξεκινούμε από την πρώτη αλλά πρέπει να δούμε ποια έχει προτεραιότητα. Η προτεραιότητα πράξεων είναι η ίδια με τα μαθηματικά και είναι η ακόλουθη:

- (1) Παρενθέσεις
- (2) Πολλαπλασιασμός, Διαίρεση
- (3) Πρόσθεση, Αφαίρεση.

Σε περίπτωση που έχουμε πράξεις με την ίδια προτεραιότητα τότε εκτελείται αυτή που βρίσκεται πιο αριστερά.

Παραδείγματα

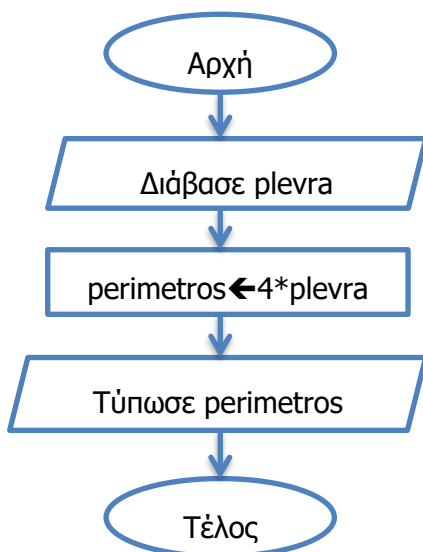
- (α) $7+12*2-1 = 7+24-1 = 30$
- (β) $6*4/2-1 = 24/2-1 = 12-1 = 11$
- (γ) $4*4*2-2 = 16*2-2 = 30$
- (δ) $(7+3)*2 = 10*2 = 20$

4. Προβλήματα με μία (1) είσοδο και έξοδο

Πρόκειται για τα απλούστερα προβλήματα που θα συναντήσετε. Ακολουθούν δύο παραδείγματα. Για καθένα από τα παραδείγματα θα σχεδιάζουμε το λογικό διάγραμμα και θα γράφουμε την προκαταρκτική εκτέλεση.

Παράδειγμα 1

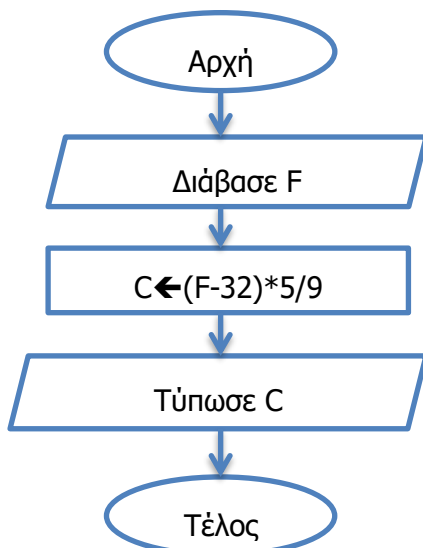
Να παρουσιάσετε τον αλγόριθμο που υπολογίζει την περίμετρο ενός τετραγώνου και να γράψετε την προκαταρκτική εκτέλεση για αρχική τιμή της πλευράς = 4



| Μεταβλητές | | Παρουσίαση |
|------------|------------|------------|
| plevra | perimetros | 16 |
| 4 | | |
| | 16 | |

Παράδειγμα 2

Στην Κύπρο η μονάδα μέτρησης της θερμοκρασίας που χρησιμοποιούμε είναι οι βαθμοί Κελσίου (°C). Υπάρχουν όμως άλλες χώρες, όπως οι ΗΠΑ, που χρησιμοποιούν τους βαθμούς Φάρεναϊτ (°F). Ο τρόπος μετατροπής των βαθμών Φάρεναϊτ σε βαθμούς Κελσίου δίνεται από τον τύπο $C = (F - 32) * 5/9$. Να σχεδιάσετε το λογικό διάγραμμα και να γράψετε την προκαταρκτική εκτέλεση για αρχική τιμή $F = 98.6$.



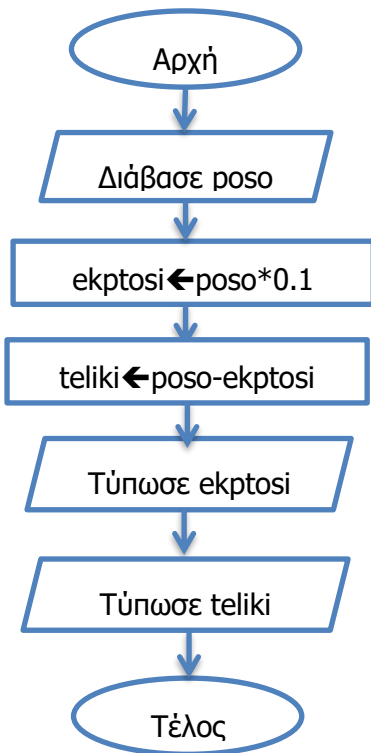
| Μεταβλητές | | Παρουσίαση |
|------------|----|------------|
| F | C | 37 |
| 98.6 | | |
| | 37 | |

5. Προβλήματα με μία (1) είσοδο και περισσότερες εξόδους

Η μεθοδολογία δεν αλλάζει σε σχέση με την προηγούμενη κατηγορία. Οι αλλαγές που θα παρατηρήσουμε είναι ότι θα έχουμε περισσότερες μεταβλητές και περισσότερα στοιχεία στην παρουσίαση.

Παράδειγμα 3

Ένα κατάστημα ρούχων κάνει έκπτωση 10% σε όλους τους πελάτες του. Να σχεδιάσετε το λογικό διάγραμμα που δέχεται το αρχικό ποσό και τυπώνει την έκπτωση και την τελική τιμή. Τέλος να γράψετε την προκαταρκτική εκτέλεση για αρχική τιμή poso=200.



| Μεταβλητές | | | Παρουσίαση |
|------------|---------|--------|------------|
| poso | ekptosi | teliki | 20 |
| 200 | | | 180 |
| | 20 | | |
| | | 180 | |

6. Προβλήματα με περισσότερες εισόδους και 1 έξοδο

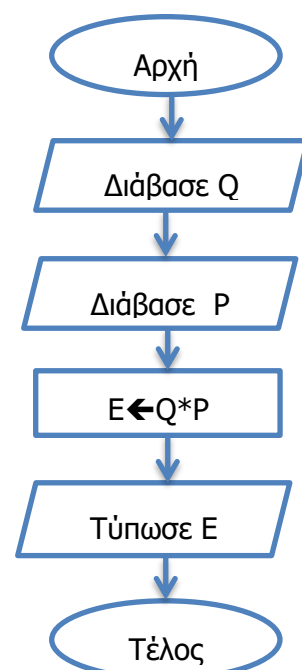
Σε αυτή τη κατηγορία προβλημάτων θα έχουμε περισσότερες από μία εισόδους (περισσότερες μεταβλητές στο Διάβασε) και μόνο μία έξοδο.

Παράδειγμα 4

Ένας έμπορος πώλησε κάποια ποσότητα (Q) από πορτοκάλια προς κάποια τιμή (P).

Πόση θα είναι η εισπραξη (E); Να σχεδιάσετε το λογικό διάγραμμα και να γράψετε την προκαταρκτική εκτέλεση για Q=100, P= 1.50

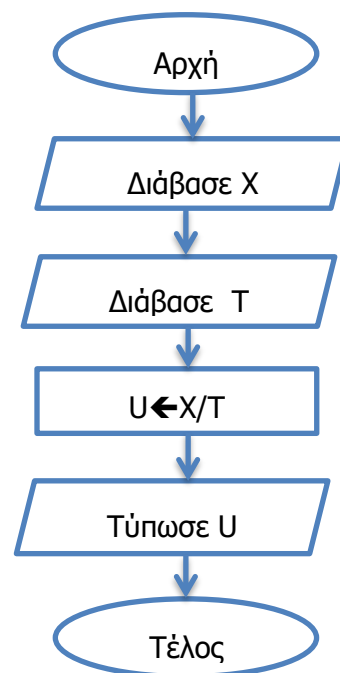
| Μεταβλητές | | | Παρουσίαση |
|------------|------|-----|------------|
| Q | P | E | 150 |
| 100 | | | |
| | 1.50 | | |
| | | 150 | |



Παράδειγμα 5

Η ταχύτητα ενός σώματος δίνεται από τον τύπο $U=X/T$ (ταχύτητα = μετατόπιση/ χρόνος). Για παράδειγμα, αν ένα σώμα διένυσε μια απόσταση 100 μέτρα μέσα σε 2 δευτερόλεπτα τότε σημαίνει ότι η ταχύτητα του ήταν 50 m/s. Να σχεδιάσετε το λογικό διάγραμμα για το πιο πάνω πρόβλημα.

| Μεταβλητές | | | Παρουσίαση |
|------------|---|----|------------|
| X | T | U | 50 |
| 100 | | | |
| | 2 | | |
| | | 50 | |

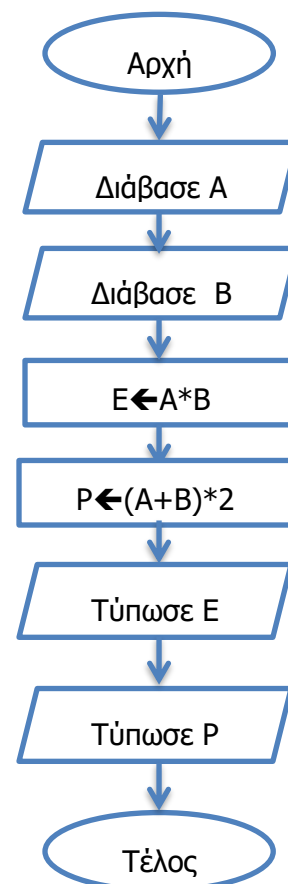
**7. Προβλήματα με περισσότερες εισόδους και εξόδους**

Σε αυτή την κατηγορία προβλημάτων ο αλγόριθμος μας θα δέχεται πολλαπλές εισόδους και θα παράγει πολλαπλά αποτελέσματα.

Παράδειγμα 6

Να σχεδιάσετε το λογικό διάγραμμα που δέχεται το μήκος (A) και το πλάτος (B) ενός ορθογωνίου και υπολογίζει και τυπώνει το εμβαδόν (E) και την περίμετρο (P) του. Το εμβαδόν ενός ορθογωνίου δίνεται από τον τύπο $E=A*B$, ενώ η περίμετρος από τον τύπο $P=(A+B)*2$. Να γράψετε την προκαταρκτική εκτέλεση για αρχικές τιμές $A=10$ και $B=5$.

| Μεταβλητές | | | | Παρουσίαση |
|------------|---|----|----|------------|
| A | B | E | P | 50 |
| 10 | | | | 30 |
| | 5 | | | |
| | | 50 | | |
| | | | 30 | |

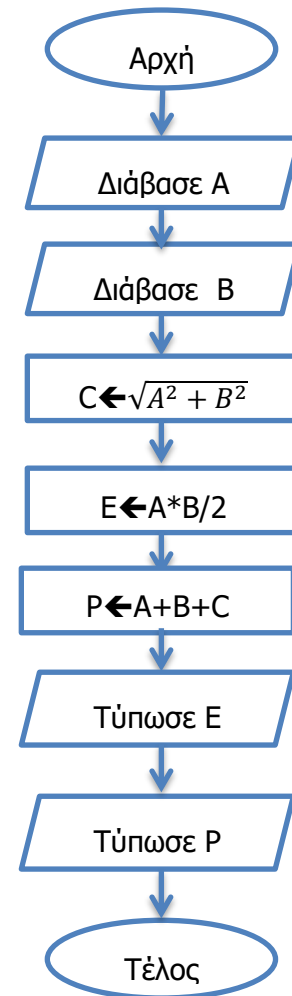


Παράδειγμα 7

Να σχεδιάσετε το λογικό διάγραμμα που υπολογίζει το εμβαδόν και την περίμετρο ενός ορθογωνίου τριγώνου όταν είναι γνωστές οι 2 κάθετες. Για να υπολογίσετε την περίμετρο πρέπει να υπολογίσετε την υποτείνουσα χρησιμοποιώντας το πυθαγόρειο θεώρημα.

(Σημείωση: $E=A*B/2$, $P=A+B+C$, $C=\sqrt{A^2 + B^2}$)

| Μεταβλητές | | | | | Παρουσίαση |
|------------|---|---|---|----|------------|
| A | B | C | E | P | 6 12 |
| 3 | | | | | |
| | 4 | | | | |
| | | 5 | | | |
| | | | 6 | | |
| | | | | 12 | |



Βασικές Έννοιες**Ακολουθιακή
Δομή:**

Κατηγορία προβλημάτων τα οποία δεν περιέχουν αποφάσεις. Σε αυτού του είδους τα προβλήματα η λύση ακολουθεί το μοντέλο **Είσοδος Δεδομένων → Επεξεργασία → Παρουσίαση Πληροφοριών**

**Προκαταρκτική
Εκτέλεση:**

Είναι η εκτέλεση του αλγορίθμου στο χαρτί, όπως θα τον εκτελούσε ο υπολογιστής.

**Τελεστές –
Προτεραιότητα
Πράξεων:**

Για την αναπαράσταση μαθηματικών εκφράσεων χρησιμοποιούμε τους ακόλουθους τελεστές

| Μαθηματική πράξη | Μαθηματικός Συμβολισμός | Συμβολισμός στους αλγορίθμους |
|------------------|-------------------------|-------------------------------|
| Πρόσθεση | $A + B$ | $A + B$ |
| Αφαίρεση | $A - B$ | $A - B$ |
| Πολλαπλασιασμός | $A \times B$ | $A * B$ |
| Διαίρεση | $A \div B$ | A / B |

Η προτεραιότητα των πιο πάνω τελεστών είναι:

- (1) Παρενθέσεις
- (2) Πολλαπλασιασμός, Διαίρεση
- (3) Πρόσθεση, Αφαίρεση.

Γ7.4 Προβλήματα Δομής Διακλάδωσης

Τι θα μάθουμε σήμερα:

- ◆ Να διακρίνουμε τις διαφορές μεταξύ ακολουθιακής δομής και δομής διακλάδωσης
- ◆ Να μελετούμε προβλήματα και να εξηγούμε αν απαιτείται ή όχι δομή διακλάδωσης
- ◆ Να δίνουμε παραδείγματα από την καθημερινότητα όπου γίνεται χρήση κάποιου είδους δομής διακλάδωσης
- ◆ Να σχεδιάζουμε λογικά διαγράμματα με μία συνθήκη, τόσο χωρίς το else όσο και με εντολές στο else
- ◆ Να χρησιμοποιούμε τη μέθοδο της προκαταρκτικής εκτέλεσης για την εξαγωγή των αποτελεσμάτων των λογικών διαγραμμάτων.

1. Δομή διακλάδωσης ή Δομή επιλογής

1.1 Δομή διακλάδωσης και καθημερινότητα

Πολλές φορές τυγχάνει να εμπλακούμε σε απλά προβλήματα τα οποία εμπεριέχουν «διλήμματα» της καθημερινής μας ζωής. Επίσης, όλοι έτυχε να πούμε κάποιες φράσεις όπως π.χ. «αν διαβάσω, θα γράψω καλά στο διαγώνισμα» ή «αν έχω περισσότερες από 50 αδικαιολόγητες απουσίες, θα μείνω στάσιμος».

Σε αυτές τις φράσεις, όπως και σε πολλές άλλες τις οποίες χρησιμοποιούμε στον καθημερινό μας λόγο, μπορούμε να αναγνωρίσουμε το γεγονός ότι ανάλογα με το τι θα γίνει ή τι θα επιλέξουμε, θα υπάρξει διαφορετικό αποτέλεσμα. Αυτή η έννοια στην οποία γίνεται κάποια σύγκριση (π.χ. να πάρουμε δύο δεδομένα και να αποφασίσουμε εάν το πρώτο είναι ίσο, μεγαλύτερο ή μικρότερο από το δεύτερο) ή κάποιος λογικός συλλογισμός (π.χ. συλλογισμός που απαντιέται με «Αληθής» ή «Ψευδής»), ονομάζεται συνθήκη.

Κάποια απτά παραδείγματα από την καθημερινή μας ζωή όπου υπάρχει κάποια συνθήκη, φαίνονται στον πιο κάτω πίνακα:

| Φράση | Συνθήκη | Τιμή (Αληθής/Ψευδής) | Αποτέλεσμα |
|-----------------------------------------------------------------------------------------------|------------------------------------------|-------------------------|-----------------------------------------|
| Αν βρέξει, θα πάρω ομπρέλα. | Αν βρέξει | Αληθής Ψευδής | Θα πάρω ομπρέλα Δεν θα πάρω ομπρέλα |
| Αν αγοράσω προϊόντα αξίας πέραν των 100 ευρώ θα έχω έκπτωση 15% διαφορετικά θα έχω έκπτωση 7% | Αν το ποσό είναι μεγαλύτερο από 100 ευρώ | Αληθής Ψευδής | Θα έχω έκπτωση 15% Θα έχω έκπτωση 7% |

1.2 Διαφορές ανάμεσα στη δομή ακολουθίας και στη δομή διακλάδωσης

Όλα τα προβλήματα δεν λύνονται μόνο με τη διαδοχική εκτέλεση κάποιων συγκεκριμένων εντολών. Πολλές φορές πρέπει να ελεγχθεί μια κατάσταση και ανάλογα να αποφασίσουμε αν θα συνεχίσουμε προς τη μία ή την άλλη κατεύθυνση τις ενέργειές μας.

Η ακολουθιακή δομή εντολών ή δομή ακολουθίας χρησιμοποιείται για την αντιμετώπιση απλών προβλημάτων, όπου είναι δεδομένη η σειρά εκτέλεσης ενός συνόλου ενεργειών. Ένα απλό παράδειγμα από την καθημερινή μας ζωή είναι η ακολουθία οδηγιών μίας συνταγής

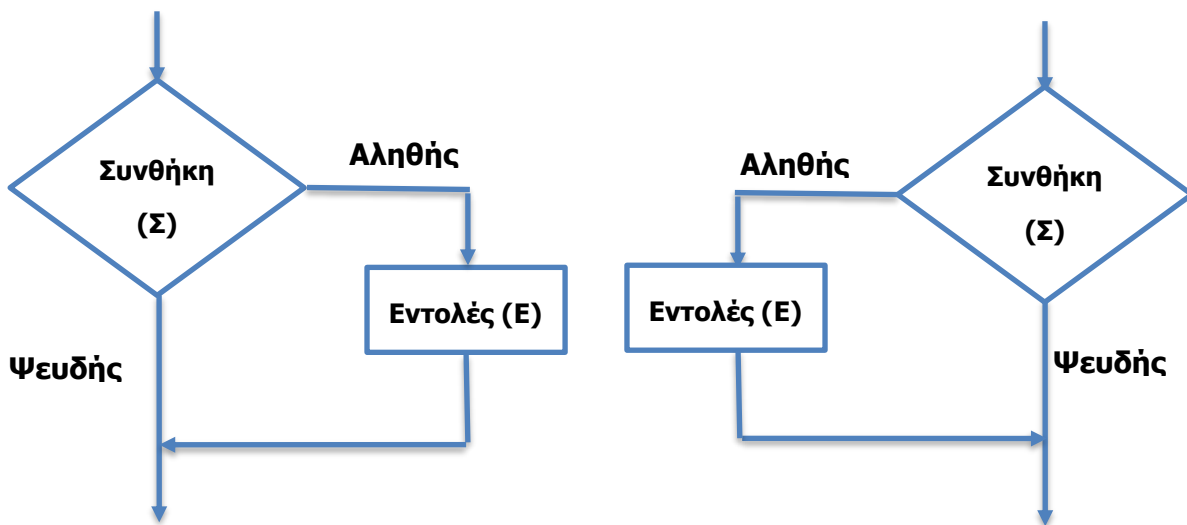
μαγειρικής με στόχο την κατασκευή ενός φαγητού. Τα βήματα και οι ποσότητες που πρέπει να ακολουθηθούν είναι συγκεκριμένα και οι οδηγίες απόλυτα καθορισμένες και σαφείς.

Στη πραγματικότητα, όμως, πολύ λίγα προβλήματα μπορούν να επιλυθούν με τον τρόπο της ακολουθιακής δομής ενεργειών. Συνήθως, τα προβλήματα έχουν κάποιες ιδιαιτερότητες και δεν ισχύουν τα ίδια βήματα για κάθε περίπτωση. Η πλέον συνηθισμένη περίπτωση είναι να λαμβάνονται κάποιες αποφάσεις με βάση κάποια δεδομένα κριτήρια, που μπορεί να είναι διαφορετικά για κάθε διαφορετικό στιγμιότυπο ενός προβλήματος. Οι καθημερινές απλές μας ενέργειες περιέχουν αυτή τη διαδικασία επιλογής με βάση κάποια κατάσταση. Για παράδειγμα, το πρόβλημα της προετοιμασίας μας για έξοδο σχετίζεται με τις καιρικές συνθήκες. Έτσι λέμε ότι «αν βρέχει, θα πάρω ομπρέλα, αλλιώς θα πάρω καπέλο». Η συνθήκη εδώ είναι το «αν βρέχει», ενώ η απόφαση είναι είτε να «πάρω ομπρέλα» είτε να «πάρω καπέλο» με βάση την «τιμή» της συνθήκης.

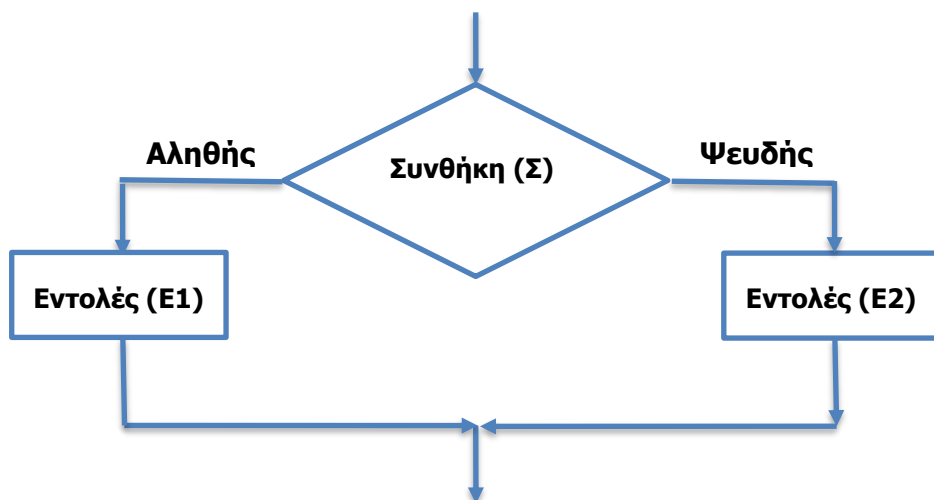
1.3 Η δομή διακλάδωσης:

Η δομή διακλάδωσης χρησιμοποιείται, όταν κατά την εκτέλεση του αλγορίθμου θα πρέπει να ελέγξουμε μια κατάσταση και ανάλογα να εκτελέσουμε ή όχι κάποιες εντολές.

Στο λογικό διάγραμμα, η εντολή επιλογής εμφανίζεται ως μία διακλάδωση στην πορεία της εκτέλεσης των εντολών. Μέσα σε έναν ρόμβο διατυπώνεται μια συνθήκη, δηλαδή μια έκφραση, που μπορεί να απαντηθεί με Αληθής (True) ή Ψευδής (False), και ανάλογα με την απάντηση ακολουθείται η κατάλληλη διαδρομή, όπως φαίνεται πιο κάτω.



Υπάρχει και μια δεύτερη μορφή της δομής διακλάδωσης, όπου ελέγχουμε μια κατάσταση και, αν ισχύει μια συνθήκη, εκτελούμε ένα σύνολο εντολών, ενώ, αν δεν ισχύει η συνθήκη, εκτελούμε ένα διαφορετικό σύνολο εντολών. Σε αυτή την περίπτωση το λογικό διάγραμμα έχει τη μορφή του παρακάτω σχήματος.



Γενικά, η διαδικασία της επιλογής περιλαμβάνει τον έλεγχο κάποιας συνθήκης που μπορεί να έχει δύο τιμές (Αληθής ή Ψευδής) και ακολουθεί η απόφαση εκτέλεσης κάποιας ενέργειας με βάση την τιμή της λογικής αυτής συνθήκης.

Εάν η συνθήκη είναι αληθής εκτελείται το σκέλος του **Αληθής**, ενώ στην αντίθετη περίπτωση που η συνθήκη είναι ψευδής, εκτελείται το σκέλος του **Ψευδής**. Σε πολλές περιπτώσεις αντί του **Αληθής** και του **Ψευδής** μπορεί να δείτε τους όρους ΟΡΘΟ / ΛΑΘΟΣ, ΝΑΙ/ΟΧΙ ή τους Αγγλικούς Όρους TRUE / FALSE. Για παράδειγμα θα μπορούσαμε να ρωτήσουμε «Είσαι μεγαλύτερος των 18 χρόνων»; Η απάντηση σε αυτή την ερώτηση είναι ΝΑΙ ή ΟΧΙ. Αν η απάντηση είναι ΟΧΙ, η σημασία της είναι ότι το άτομο είναι Μικρότερο ή Ίσο με 18 χρόνια. Στις συνθήκες γενικά χρησιμοποιούμε τα συγκριτικά σύμβολα, όπως φαίνεται πιο κάτω.

Πίνακας Συγκριτικών Συμβόλων

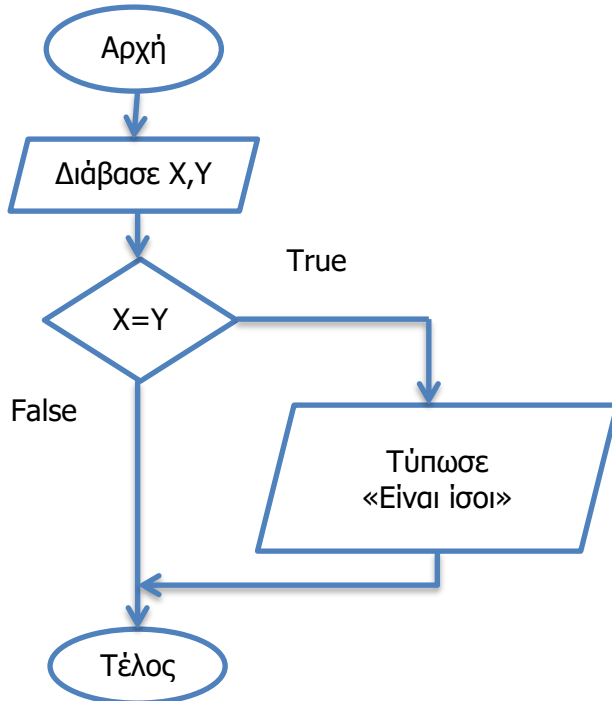
| | |
|-------------------------|----|
| Ισότητα | = |
| Μεγαλύτερο | > |
| Μεγαλύτερο ή Ίσο | >= |
| Μικρότερο | < |
| Μικρότερο ή Ίσο | <= |
| Άνισο | <> |

Παραδείγματα Συνθήκης χρησιμοποιώντας συγκριτικά σύμβολα

| Συνθήκη | Αποτέλεσμα (Αληθής/Ψευδής) |
|----------------|-----------------------------------|
| 3>2 | Αληθής |
| 10<=10 | Αληθής |
| 8<4 | Ψευδής |
| 5<>5 | Ψευδής |

Παράδειγμα 1

Να σχεδιάσετε λογικό διάγραμμα το οποίο να διαβάζει 2 αριθμούς **X**, **Y**. Αν οι 2 αριθμοί είναι ίσοι, τότε να τυπώνει το μήνυμα «**Είναι ίσοι**». Να γράψετε την προκαταρκτική εκτέλεση για αρχικές τιμές α) X=5, Y=5 και β) X=5, Y=6



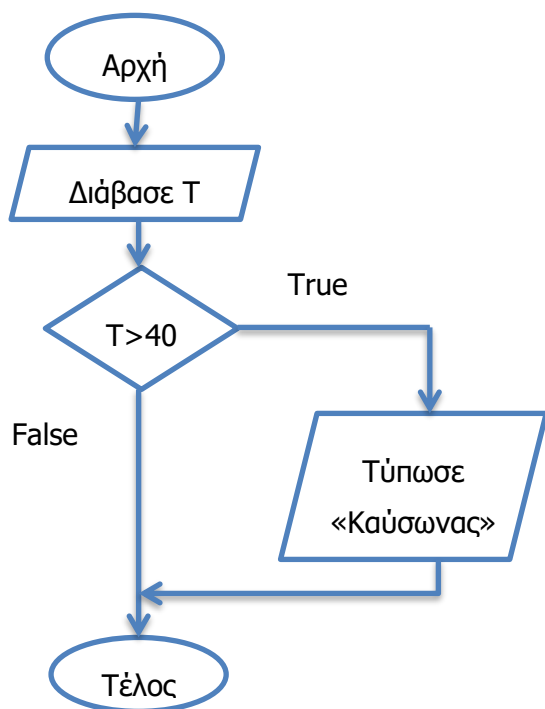
| Μεταβλητές | | Απόφαση | | Παρουσίαση |
|------------|---|---------|------------|------------|
| X | Y | X=Y | True/False | Είναι ίσοι |
| 5 | | | | |
| | 5 | | | |
| | | 5=5 | True | |

| Μεταβλητές | | Απόφαση | | Παρουσίαση |
|------------|---|---------|------------|------------|
| X | Y | X=Y | True/False | |
| 5 | | | | |
| | 6 | | | |
| | | 5=6 | False | |

Παρατηρούμε ότι στην προκαταρκτική εκτέλεση έχουμε μια επιπλέον περιοχή για να γράφουμε τις αποφάσεις (δηλαδή τα αποτελέσματα των συνθηκών). Για κάθε απόφαση που υπάρχει στον αλγόριθμό μας τοποθετούμε δυο στήλες, μια για τη συνθήκη και μια για το αποτέλεσμα της συνθήκης. Προσέξτε, επίσης, ότι σε αυτή την υποκατηγορία προβλημάτων δομής διακλάδωσης (χωρίς εντολές στο Όχι/False), υπάρχει περίπτωση το πρόγραμμα που θα υλοποιήσουμε για κάποιες περιπτώσεις να μην εμφανίζει κανένα αποτέλεσμα.

Παράδειγμα 2

Να σχεδιάσετε λογικό διάγραμμα το οποίο να διαβάζει τη θερμοκρασία μιας ημέρας (T). Αν η θερμοκρασία είναι μεγαλύτερη από 40, τότε να τυπώνει το μήνυμα «**Καύσωνας**». Να γράψετε την προκαταρκτική εκτέλεση για αρχικές τιμές α) T=45 και β) T=40.

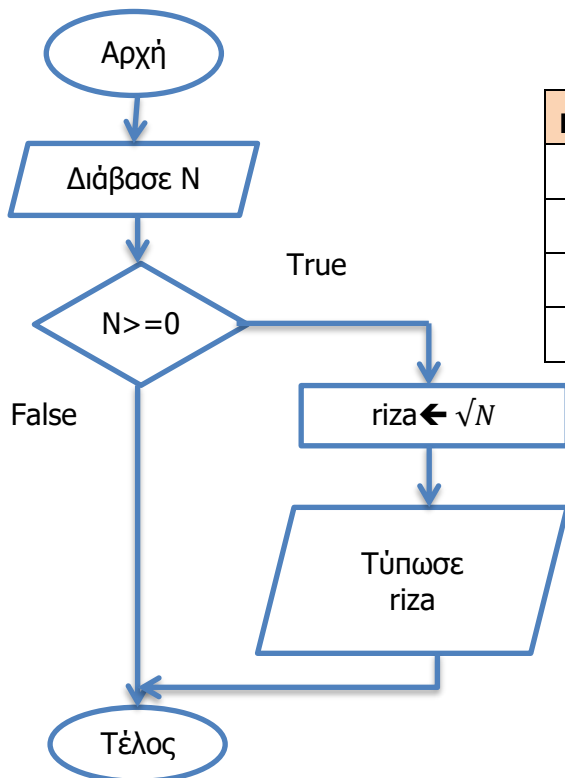


| Μεταβλητές | | Απόφαση | | Παρουσίαση |
|------------|---------|------------|--|------------|
| T | T > 40 | True/False | | |
| 45 | | | | |
| | 45 > 40 | True | | |

| Μεταβλητές | | Απόφαση | | Παρουσίαση |
|------------|---------|------------|--|------------|
| T | T > 40 | True/False | | |
| 40 | | | | |
| | 40 > 40 | False | | |

Παράδειγμα 3

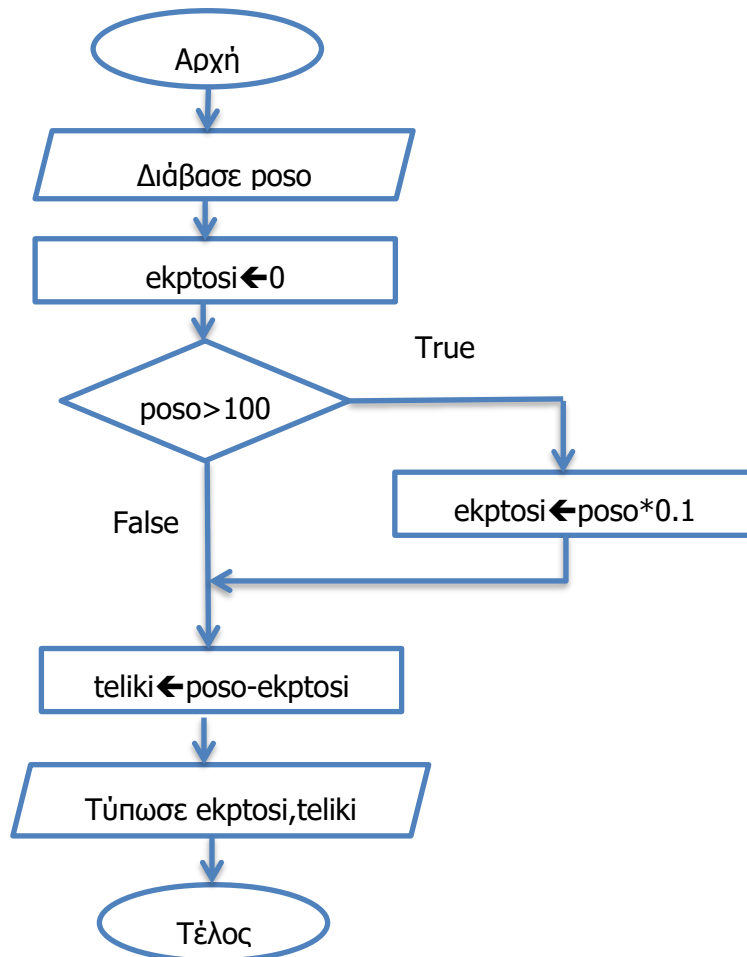
Να σχεδιάσετε λογικό διάγραμμα το οποίο να διαβάζει έναν αριθμό (N). Αν ο αριθμός είναι θετικός (≥ 0) τότε υπολογίζει και τυπώνει την τετραγωνική ρίζα του αριθμού. Να γράψετε την προκαταρκτική εκτέλεση για N=4.



| Μεταβλητές | | Απόφαση | | Παρουσίαση |
|------------|------|---------|------------|------------|
| N | riza | N >= 0 | True/False | |
| 4 | | | | |
| | | 4 >= 0 | True | |
| | 2 | | | |

Παράδειγμα 4

Σε παράδειγμα που είχαμε δώσει στην ακολουθιακή δομή, δημιουργήσαμε τον αλγόριθμο για ένα κατάστημα που έκανε έκπτωση 10% σε όλους τους πελάτες του. Να ξανασχεδιάσετε το λογικό διάγραμμα για το πιο πάνω πρόβλημα, ώστε το κατάστημα να κάνει έκπτωση μόνο σε όσους πελάτες κάνουν αγορές πέραν των 100 ευρώ.



Σε αντίθεση με τα προηγούμενα παραδείγματα εδώ βλέπουμε ότι θα έχουμε αποτέλεσμα στην παρουσίαση ανεξαρτήτως αν ισχύει ή όχι η συνθήκη.

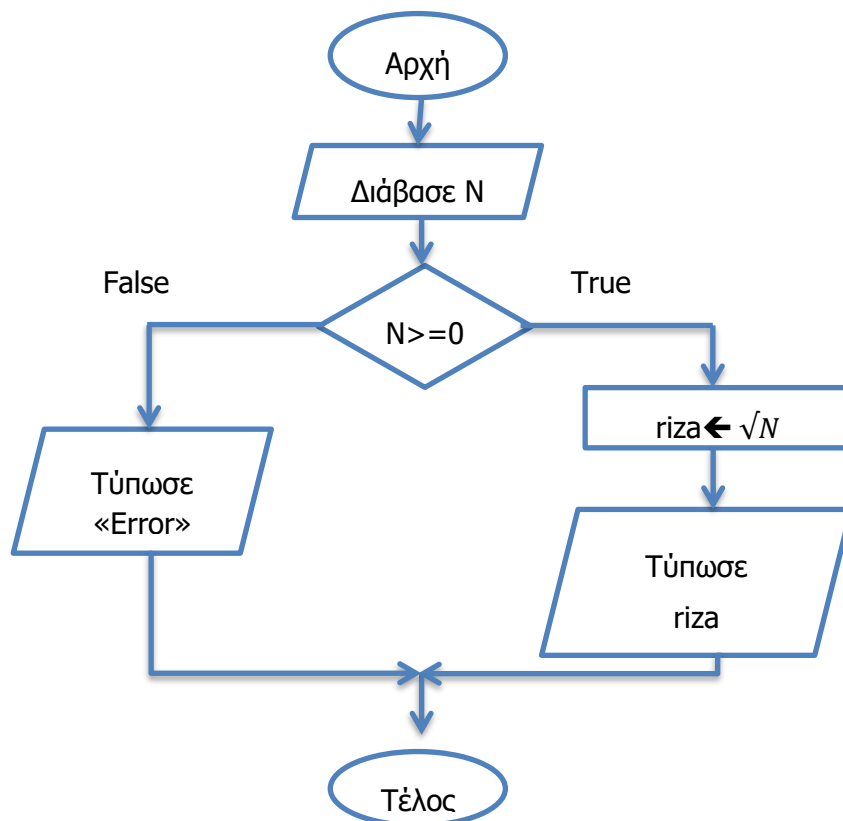
1.5 Σχεδιασμός λογικών διαγραμμάτων με μια απλή συνθήκη (με εντολές για το Ψευδής/False)

Η απουσία εντολών για την περίπτωση που η συνθήκη είναι ψευδής (false), δεν είναι κάτι συνηθισμένο, σε αλγορίθμους/ προγράμματα με μια απλή συνθήκη. Ο λόγος είναι ότι περιμένουμε πάντα από το πρόγραμμά μας να μας ενημερώσει για κάτι (να έχει κάποια έξοδο πληροφοριών). Στα πιο πάνω παραδείγματα θα θέλαμε να ξέρουμε ότι οι αριθμοί «δεν είναι ίσοι» (Παράδειγμα 1), «δεν είναι καύσωνας» (Παράδειγμα 2) και «δεν υπάρχει τετραγωνική ρίζα» (παράδειγμα 3).

Η αντιμετώπιση των προβλημάτων με εντολές για την περίπτωση που η συνθήκη είναι ψευδής δεν διαφέρει πολύ με τα προβλήματα που ήδη έχουμε δει. Πρέπει να εντοπίσουμε τα δεδομένα και τα ζητούμενα (πληροφορίες), να βρούμε τη συνθήκη ελέγχου και τις εντολές που θα εκτελούνται όταν θα ισχύει και όταν δεν θα ισχύει η συνθήκη.

Παράδειγμα 5

Να τροποποιήσετε το Παράδειγμα 3 του προηγούμενου υποκεφαλαίου, ώστε να τυπώνει μήνυμα «Error» στην περίπτωση που δεν υπάρχει τετραγωνική ρίζα.

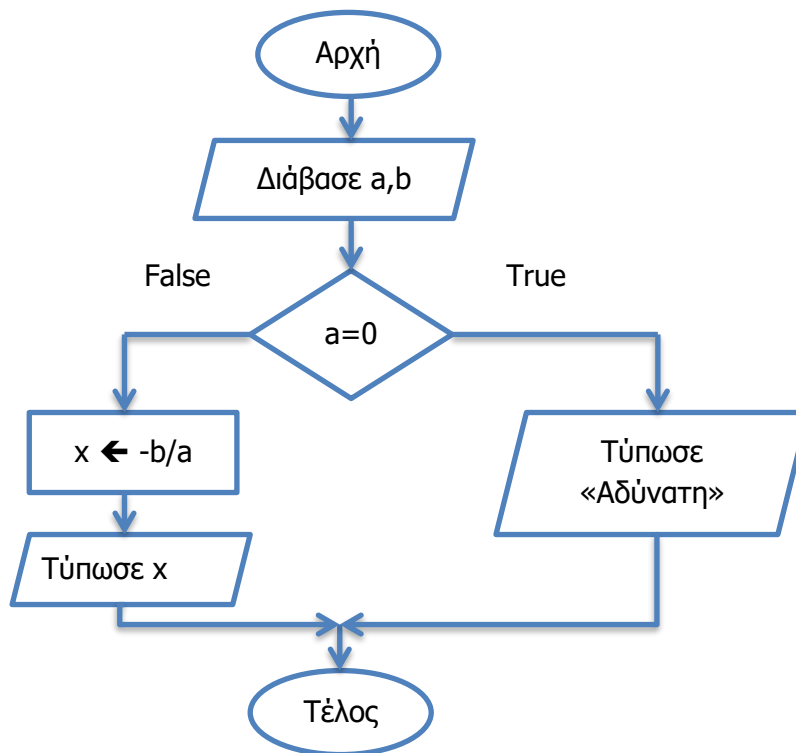


Ας δοκιμάσουμε να γράψουμε την προκαταρκτική εκτέλεση για αρχική τιμή N=-4. Παρατηρούμε ότι πλέον ο αλγόριθμός μας θα τυπώνει «Error» στην περίπτωση που δεν ισχύει η συνθήκη.

| Μεταβλητές | | Απόφαση | | Παρουσίαση |
|------------|------|---------|------------|------------|
| N | riza | N >= 0 | True/False | Error |
| -4 | | | | |
| | | -4 >= 0 | False | |

Παράδειγμα 6

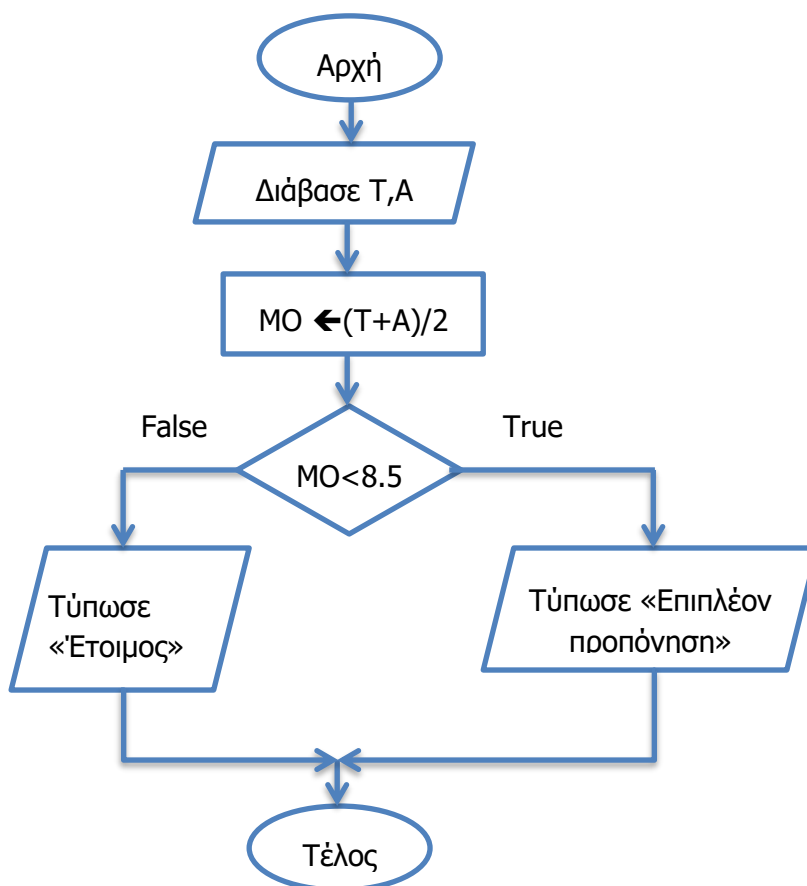
Ένας συμμαθητής σας δυσκολεύεται να βρίσκει τη λύση της εξίσωσης α' βαθμού $ax+b=0$. Να σχεδιάσετε το λογικό διάγραμμα για το πιο πάνω πρόβλημα που θα δέχεται το a και b και θα υπολογίζει το x. Σε περίπτωση που δοθεί $a=0$ να τυπώνει το μήνυμα «Αδύνατη». Τέλος, να γράψετε την προκαταρκτική εκτέλεση για την εξίσωση $2x+4=0$. (Σημείωση: Η λύση της εξίσωσης είναι $x=-b/a$). Να γράψετε την προκαταρκτική εκτέλεση για αρχικές τιμές $a=2$ και $b=4$.



| Μεταβλητές | | | Απόφαση | | Παρουσίαση |
|------------|---|----|---------|------------|------------|
| a | B | x | a=0 | True/False | -2 |
| 2 | | | | | |
| | 4 | | | | |
| | | | 2=0 | False | |
| | | -2 | | | |

Παράδειγμα 7

Ο προπονητής μιας ποδοσφαιρικής ομάδας έχει παρατηρήσει ότι οι παίκτες της ομάδας του δεν είναι σε φόρμα. Αποφάσισε λοιπόν να τους εξετάσει τόσο σε θέματα ταχύτητας, όσο και σε θέματα αντοχής, υποβάλλοντάς τους σε μία άσκηση ταχύτητας (T) και σε μια άσκηση αντοχής (A). Σε καθεμιά από τις ασκήσεις ο ποδοσφαιριστής παίρνει μια βαθμολογία με άριστα το 10. Αν ο μέσος όρος των 2 ασκήσεων είναι μικρότερος του 8.5, τότε ο ποδοσφαιριστής πρέπει να περάσει από επιπλέον προπόνηση διαφορετικά θεωρείται ότι είναι έτοιμος. Να σχεδιάσετε το λογικό διάγραμμα για το πιο πάνω πρόβλημα.

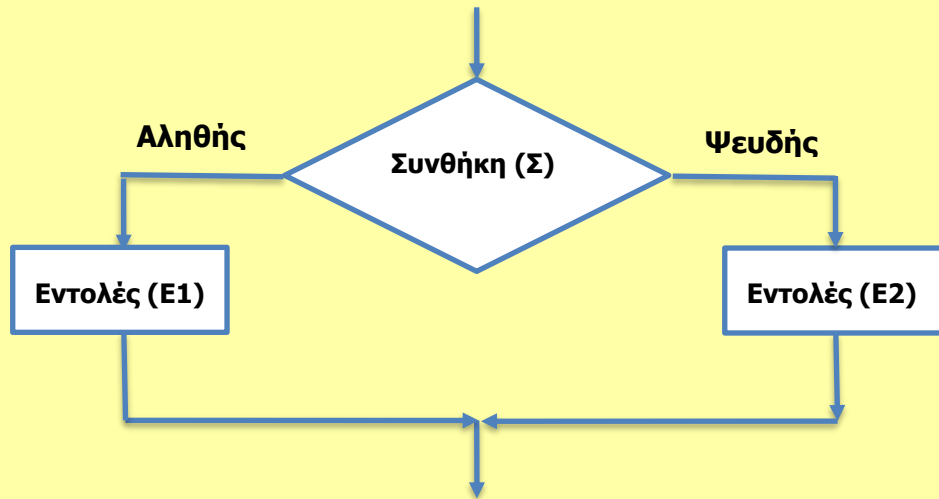


Βασικές Έννοιες

Δομή

Διακλάδωσης:

Δομή διακλάδωσης έχουμε όταν σε έναν αλγόριθμο πρέπει να εκτελεστεί μία από δύο εναλλακτικές σειρές εντολών με βάση μια απόφαση, ελέγχοντας μια λογική συνθήκη. Το γεωμετρικό σχήμα που χρησιμοποιούμε στα λογικά διαγράμματα για να δείξουμε ότι το πρόγραμμα πρέπει να μία απόφαση είναι ο ρόμβος.



Γ7.5 Γλώσσες Προγραμματισμού

Τι θα μάθουμε σήμερα:

- ◊ Να εξηγήσουμε τι είναι γλώσσα προγραμματισμού
- ◊ Να διακρίνουμε τη διαφορά μεταξύ αλγορίθμου και γλώσσας προγραμματισμού
- ◊ Να διακρίνουμε τις διάφορες γλώσσες προγραμματισμού και να αναφέρουμε τα κύρια χαρακτηριστικά τους.

1. Εισαγωγή

Γλώσσα προγραμματισμού είναι μια τεχνητή γλώσσα που λειτουργεί ως μέσο με το οποίο μπορούμε να μετατρέψουμε έναν αλγόριθμο σε πρόγραμμα, ώστε να εκτελείται από τον υπολογιστή. Οποιοδήποτε πρόγραμμα βλέπετε στον υπολογιστή σας, είτε πρόκειται για παιχνίδι είτε για λειτουργικό σύστημα, έχει δημιουργηθεί χρησιμοποιώντας μια γλώσσα προγραμματισμού.

Επειδή όπως θα εξηγήσουμε στη συνέχεια, η περιγραφή των εντολών με τρόπο που να μπορεί να τις εκτελεί ο υπολογιστής είναι δύσκολη για τον άνθρωπο, οι γλώσσες προγραμματισμού λειτουργούν ως ένα ενδιάμεσο στάδιο, ώστε η περιγραφή του αλγορίθμου να γίνεται με ευκολία και ακρίβεια και στη συνέχεια, με αυτόματο τρόπο, να μεταφράζεται σε μορφή που να μπορεί να εκτελεστεί από την ΚΜΕ.

2. Γλώσσες Προγραμματισμού

Όπως και οι φυσικές γλώσσες των ανθρώπων, έτσι και αυτές έχουν λεξιλόγιο και σύνταξη, τα οποία πρέπει να ακολουθούνται πιστά και χαρακτηρίζονται ως **Γλώσσες Υψηλού Επιπέδου**, αφού είναι πιο κοντά στην ανθρώπινη γλώσσα παρά στη «γλώσσα» της ΚΜΕ. Μερικές από τις πιο γνωστές γλώσσες προγραμματισμού Υψηλού Επιπέδου είναι οι ακόλουθες:

- **Visual Basic:** Χρησιμοποιείται για προγράμματα που θα «τρέξουν» σε Windows. Εύκολη στην εκμάθησή της.
- **C/C++:** Πρόκειται για γλώσσες με μεγάλες δυνατότητες που στόχο έχουν τη δημιουργία πολύπλοκων προγραμμάτων (τα Windows είναι «γραμμένα» κατά κύριο λόγο σε C++).
- **Pascal:** Γλώσσα που χρησιμοποιείται κατά κύριο λόγο για εκπαιδευτικούς λόγους. Προτιμάται για το «λεξιλόγιο» της που είναι πολύ κοντά στη φυσική γλώσσα.
- **Java:** Κύριο χαρακτηριστικό αυτής της γλώσσας είναι ότι μπορεί να δημιουργήσει εφαρμογές ανεξαρτήτως πλατφόρμας (λειτουργικού συστήματος). Οι περισσότερες εφαρμογές για κινητά τηλέφωνα (π.χ. Android) έχουν δημιουργηθεί με Java.
- **Perl:** Αρχικά δημιουργήθηκε για γρήγορη ανάλυση δεδομένων και δημιουργία εκθέσεων. Παρέχει εύκολους τρόπους αναζήτησης και μετατροπής δεδομένων από μια μορφή σε άλλη. Χρησιμοποιείται κυρίως από διαχειριστές συστημάτων και για αυτοματοποιήσεις σε ιστοσελίδες.
- **Python:** Γλώσσα προγραμματισμού η οποία δημιουργήθηκε με στόχο το ξεκάθαρο και απλό συντακτικό. Χρησιμοποιείται για αυτοματοποιήσεις σε ιστοσελίδες, ως γλώσσα αυτοματισμού σε εφαρμογές και ως εκπαιδευτική γλώσσα.

Παρακάτω παρουσιάζονται οι εντολές στις πιο πάνω γλώσσες για το πρόγραμμα που υπολογίζει την περίμετρο ενός τετραγώνου.

| | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre>#include <iostream> using namespace std; int main() { int a, P; cin >> a; P = 4 * a; cout << P; return 0; }</pre> <p style="text-align: center;">(α) C++</p> | <pre>Private Sub cmdCalculate_Click() Dim a As Integer Dim P As Integer a = InputBox("Δώσε την πλευρά") P = 4 * a MsgBox P End Sub</pre> <p style="text-align: center;">(β) Visual Basic</p> |
| <pre>a=int(input('Δώσε πλευρά')) P = 4*a print(P)</pre> <p style="text-align: center;">(γ) Python</p> | <pre>Import java.util.*; public class tetragono { public static void main(String[] args){ Scanner in = new Scanner(System.in); int a, P; a = in.nextInt(); P = 4 * a; System.out.println(P); } }</pre> <p style="text-align: center;">(δ) Java</p> |

Εικόνα 65 Πρόγραμμα το οποίο υπολογίζει την περίμετρο ενός τετραγώνου:

(α) σε γλώσσα προγραμματισμού C++, (β) σε γλώσσα προγραμματισμού Visual Basic
 (γ) σε γλώσσα προγραμματισμού Python, (δ) σε γλώσσα προγραμματισμού Java

Οι Γλώσσες Υψηλού Επιπέδου είναι σχετικά εύκολες στην εκμάθηση και επιτρέπουν στον/στην προγραμματιστή/τρια να δημιουργεί πολύπλοκα προγράμματα σε σχετικά σύντομο χρονικό διάστημα, αυξάνοντας την παραγωγικότητά του/της. Παρόλο που οι πιο πάνω γλώσσες έχουν διαφορές, όταν πρόκειται για απλά προγράμματα βλέπουμε ότι έχουν πολλές ομοιότητες.

Στις Γλώσσες Υψηλού Επιπέδου, πολλά στοιχεία που χρησιμοποιούνται συχνά (υπολογισμοί, συναρτήσεις, κουμπιά, παράθυρα, κ.λπ.) υπάρχουν ως έτοιμα τμήματα προγραμμάτων, ενώ παράλληλα παρέχονται πολλά χρήσιμα εργαλεία για την ανάπτυξη προγραμμάτων.

Βασικές Έννοιες

Γλώσσα Προγραμματισμού:

Είναι τρόπος περιγραφής ενός αλγορίθμου σε πρόγραμμα με σκοπό να μεταφραστεί σε μορφή που να εκτελείται από την ΚΜΕ του υπολογιστή.

Γλώσσα Υψηλού Επιπέδου:

Γλώσσα περιγραφής ενός αλγορίθμου παραπλήσια με την ανθρώπινη, με δικό της συντακτικό και γραμματική.

Παραδείγματα: Pascal, Python, C/C++, Visual Basic, Java, Perl.

Πηγές

1. Αράπογλου Α., Μαβόγλου Χ., Οικονομάκος Η., Φύτρος Κ. (2006) *Πληροφορική Α', Β', Γ' Γυμνασίου*, σελ. 182—184, ΟΕΔΒ.

Τι θα μάθουμε σήμερα:

- ◆ Να χρησιμοποιούμε το περιβάλλον προγραμματισμού του *Pame* για να ανοίγουμε (Open), να αποθηκεύουμε (Save), να αποσφαλματώνουμε και να εκτελούμε (Run) προγράμματα στη γλώσσα προγραμματισμού Python
- ◆ Να επιλέγουμε κατάλληλο τύπο δεδομένων για τις μεταβλητές ενός προγράμματος
- ◆ Να χρησιμοποιούμε εντολές εισόδου και εξόδου
- ◆ Να επιλέγουμε και να τοποθετούμε τις κατάλληλες εντολές με τη σωστή σειρά, για να μετατρέψουμε έναν αλγόριθμο σε πρόγραμμα
- ◆ Να εκτελούμε το πρόγραμμα και να διαπιστώνουμε κατά πόσο εκτελείται σωστά.

1. Εισαγωγή

Η γλώσσα προγραμματισμού Python δημιουργήθηκε από τον Ολλανδό Γκίντο βαν Ρόσσομ (Guido van Rossum) στο ερευνητικό κέντρο Centrum Wiskunde & Informatica (CWI) το 1989 και κυκλοφόρησε για πρώτη φορά το 1991.

Η γλώσσα αυτή είναι ιδιαίτερα δημοφιλής για διδασκαλία προγραμματισμού, καθώς η σύνταξη της είναι σχετικά εύκολη για μάθηση και παράλληλα προωθεί την παραγωγή δομημένων προγραμμάτων. Διακρίνεται λόγω του ότι έχει πολλές βιβλιοθήκες που διευκολύνουν ιδιαίτερα αρκετές συνηθισμένες εργασίες και για την ταχύτητα εκμάθησής της. Επιπρόσθετα, η Python έχει χρησιμοποιηθεί ευρέως για την ανάπτυξη εφαρμογών ανοικτού λογισμικού και είναι μία από τις γλώσσες που χρησιμοποιούνται περισσότερο από την κοινότητα ανοικτού λογισμικού. Επιπλέον, στηρίζει αλλά και στηρίζεται από την ανάπτυξη των τεχνολογιών του διαδικτύου νέας γενιάς. Τέλος, θεωρείται ως μια σύγχρονη γλώσσα προγραμματισμού η οποία αναβαθμίζεται συνεχώς.

Για σκοπούς διδασκαλίας της ενότητας αυτής θα χρησιμοποιήσουμε την **Python 3** και συγκεκριμένα την έκδοση **3.8.6**. Μπορούμε να κατεβάσουμε την Python στον υπολογιστή μας από τον σύνδεσμο <https://www.python.org/>. Εκεί θα βρούμε ένα ολοκληρωμένο περιβάλλον, το IDLE, το οποίο μπορούμε να χρησιμοποιήσουμε για να γράψουμε, να αποσφαλματώσουμε και να εκτελέσουμε το πρόγραμμά μας.



Εικόνα 66 Το λογότυπο της Python

2. Ενεργοποίηση του ολοκληρωμένου περιβάλλοντος ανάπτυξης της Python

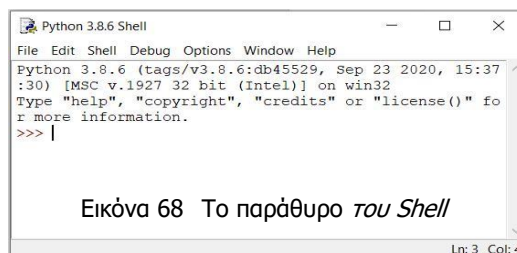
Το ολοκληρωμένο περιβάλλον (**IDLE**) της Python αποτελείται από το παράθυρο του **Shell** (διερμηνευτής) και το παράθυρο του **Editor**.

Κάνουμε διπλό κλικ στο εικονίδιο της Python που βρίσκεται στην επιφάνεια εργασίας ή από το μενού **Start** επιλέγουμε το ολοκληρωμένο περιβάλλον ανάπτυξης (**IDLE**) της Python.



Εικόνα 67 Το εικονίδιο του IDLE της Python

Ακολούθως, εμφανίζεται στην οθόνη μας το παράθυρο του Shell.



Εικόνα 68 Το παράθυρο του Shell

3. Δημιουργία νέου αρχείου

Από το μενού **File** επιλέγουμε **New File**. Ανοίγει ένα νέο παράθυρο (Editor Window) στο οποίο μπορούμε να γράψουμε τον κώδικά μας.

4. Αποθήκευση αρχείου

Αφού γράψουμε τον κώδικά μας στο Editor Window, αποθηκεύουμε το αρχείο επιλέγοντας το μενού **File** και την επιλογή **Save As**.

Σημείωση: Τα αρχεία της Python πρέπει να έχουν επέκταση ***.py**

5. Άνοιγμα αρχείου

Από το μενού **File** επιλέγουμε **Open**. Ανοίγει ένα νέο παράθυρο (Editor Window) στο οποίο εμφανίζεται ο κώδικας του προγράμματός μας.

6. Εκτέλεση προγράμματος

Αφού γράψουμε και αποθηκεύσουμε το πρόγραμμά μας, το εκτελούμε έτσι ώστε να δούμε κατά πόσο δίνει σωστά αποτελέσματα. Από το μενού **Run** του Editor Window επιλέγουμε **Run Module**. Τα αποτελέσματα της εκτέλεσης του προγράμματος παρουσιάζονται στο παράθυρο του Shell.

The screenshot shows two windows side-by-side. The left window is a Python editor with the following code:

```
x=int(input('Δώστε τον πρώτο αριθμό: '))
y=int(input('Δώστε τον δεύτερο αριθμό: '))
z=x+y
print('Το άθροισμα είναι:',z)
```

The right window is the Python 3.8.6 Shell, showing the execution output:

```
Python 3.8.6 (tags/v3.8.6:db45529, Sep 23 2020, 15:37:30)
[MSC v.1927 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more
information.
>>>
===== RESTART: C:\Users\dimit\OneDrive\Desktop\Γ Συμπ
άσιου\ex1.py =====
Δώστε τον πρώτο αριθμό: 6
Δώστε τον δεύτερο αριθμό: 8
Το άθροισμα είναι: 14
>>>
```

Εικόνα 69 Εκτέλεση προγράμματος

7. Δημιουργία Προγράμματος

7.1 Μεταβλητές

Όπως αναφέραμε σε προηγούμενο κεφάλαιο της ενότητας, στα λογικά διαγράμματα υπάρχει η έννοια της μεταβλητής. Σε αντίθεση με άλλες γλώσσες προγραμματισμού, στην Python δεν χρειάζεται να δηλωθεί μια μεταβλητή, αλλά δημιουργείται με ανάθεση τιμής. Δηλαδή, της δίνουμε ένα όνομα και στη συνέχεια της εκχωρούμε μια τιμή. Η γενική σύνταξη της εκχώρησης τιμής είναι:

μεταβλητή = έκφραση

Π.χ.

Plevra = 4

Στη γλώσσα Python υπάρχουν ορισμένοι κανόνες που πρέπει να ακολουθήσουμε σχετικά με το όνομα μιας μεταβλητής:

- Το όνομα μιας μεταβλητής μπορεί να περιέχει μόνο αλφαριθμητικούς χαρακτήρες και κάτω παύλες (A-z, 0-9, και _).
- Το όνομα μιας μεταβλητής πρέπει να ξεκινά με γράμμα ή την κάτω παύλα.
- Το όνομα μιας μεταβλητής δεν μπορεί να ξεκινά με αριθμό.
- Οι κεφαλαίοι και οι πεζοί χαρακτήρες είναι διαφορετικοί, άρα οι age, Age και AGE είναι τρεις διαφορετικές μεταβλητές.
- Συνηθίζουμε να δίνουμε όνομα σχετικό με το περιεχόμενο της μεταβλητής.
- Στην Python υπάρχουν δεσμευμένες λέξεις (reserved words), οι οποίες χρησιμοποιούνται για τη σύνταξη εντολών και δεν μπορούν να χρησιμοποιηθούν ως ονομασία μιας μεταβλητής. Μερικές από αυτές είναι: **input, print, int, float, string, if, else, for, while**.

Στις μεταβλητές αναθέτονται τιμές διάφορων τύπων δεδομένων, όμως αυτές οι τιμές μπορούν να αλλάζουν κατά τη διάρκεια εκτέλεσης του προγράμματος. Τρεις βασικοί τύποι δεδομένων που θα χρησιμοποιήσουμε είναι οι εξής:

- **Integer:** Αντιπροσωπεύει ακέραιους αριθμούς. Για αυτό κάθε δεδομένο που δηλώνεται ως integer είναι ένας ακέραιος αριθμός.
- **Float:** Αντιπροσωπεύει τους πραγματικούς (δεκαδικούς) αριθμούς.
- **String:** Αντιπροσωπεύει αλφαριθμητικά (ακολουθίες χαρακτήρων, κείμενο).

7.2 Είσοδος Δεδομένων

Για είσοδο δεδομένων χρησιμοποιούμε τη συνάρτηση `input()`, η οποία έχει την πιο κάτω μορφή:

```
Μεταβλητή = input('Μήνυμα')
```

Η μεταβλητή παίρνει την τιμή που δίνει ο χρήστης μέσω του πληκτρολογίου, αφού εμφανιστεί στην οθόνη του το μήνυμα που πληκτρολογούμε μέσα στα εισαγωγικά. Ο τύπος δεδομένων της μεταβλητής είναι αλφαριθμητικός (string), ανεξάρτητα με το τι θα πληκτρολογήσει ο χρήστης.

Στην περίπτωση που θέλουμε τα δεδομένα εισόδου να μετατραπούν σε αριθμούς έτσι ώστε να χρησιμοποιηθούν σε πράξεις, τότε πρέπει να τα μετατρέψουμε χρησιμοποιώντας τις συναρτήσεις `int()` και `float()`.

```
number1 = int(input('Δώσε έναν ακέραιο αριθμό:'))
number2 = float(input('Δώσε έναν πραγματικό αριθμό:'))
```

7.3 Έξοδος Πληροφοριών

Για εκτύπωση αποτελεσμάτων στην οθόνη χρησιμοποιούμε τη συνάρτηση `print()`, η οποία έχει την πιο κάτω μορφή:

```
print('Μήνυμα')
```

ή

print (μεταβλητή)

Τα μηνύματα πρέπει να περικλείονται σε εισαγωγικά, είτε μονά (' ') είτε διπλά (" "). Για σκοπούς ομοιομορφίας των σημειώσεων χρησιμοποιούνται μόνο τα μονά εισαγωγικά. Μπορούμε να εμφανίσουμε συνδυασμό μηνυμάτων και μεταβλητών χωρισμένα με κόμμα:

```
print('Εμβαδόν=',E)
print(E, P)
print(E)
print ('Το εμβαδόν και η περίμετρος είναι: ',E,P)
```

Είναι σημαντικό να αναφέρουμε, επίσης, την εκτύπωση ενός **πραγματικού αριθμού με συγκεκριμένο αριθμό δεκαδικών ψηφίων**:

```
print('Συνολική τιμή προϊόντων','%.2f' % total)
```

7.4 Επεξεργασία

Μια βασική λειτουργία σε ένα πρόγραμμα είναι η πραγματοποίηση κάποιου υπολογισμού, το αποτέλεσμα του οποίου (η τιμή) να εκχωρηθεί σε κάποια μεταβλητή για να χρησιμοποιηθεί αργότερα. Το μέρος υπολογισμού της τιμής αυτής ονομάζεται **έκφραση** και γίνεται χρησιμοποιώντας αριθμούς, τις τιμές άλλων μεταβλητών, σύμβολα τελεστών των διάφορων μαθηματικών πράξεων και παρενθέσεις. Η εκχώρηση τιμής σε μια μεταβλητή γίνεται με το σύμβολο = . Η γενική σύνταξη της εκχώρησης τιμής είναι:

μεταβλητή = έκφραση

Π.χ.

```
perimetros = 4*plevra
mesos_oros = (bathmos1+bathmos2+bathmos3)/3
```

Οι βασικοί αριθμητικοί τελεστές που υποστηρίζονται από την Python είναι:

| | |
|---|-----------------|
| + | πρόσθεση |
| - | αφαίρεση |
| * | πολλαπλασιασμός |
| / | διαίρεση |

Η σειρά με την οποία γίνονται οι πράξεις αλλάζει με τη χρήση παρενθέσεων. Να χρησιμοποιείτε συχνά παρενθέσεις, γιατί κάνουν πιο σωστή τη σύνταξη του κώδικα και σας προφυλάσσουν από λάθη. Η σειρά με την οποία ενεργούν οι τελεστές είναι:

| | |
|------|---------------------------|
| () | παρενθέσεις |
| *, / | πολλαπλασιασμός, διαίρεση |
| +, - | πρόσθεση, αφαίρεση |

Όταν οι τελεστές έχουν την ίδια προτεραιότητα, τότε οι πράξεις εκτελούνται με τη σειρά, από αριστερά προς δεξιά.

8. Σφάλματα

Σφάλμα ή λάθος θεωρείται οτιδήποτε επηρεάζει τη σωστή εκτέλεση του προγράμματος. Διαχωρίζονται σε δύο βασικές κατηγορίες:

Συντακτικά Λάθη: αφορούν παραβιάσεις του συντακτικού της γλώσσας π.χ. μία εντολή έχει γραφτεί συντακτικά λάθος. Όταν το πρόγραμμα περιέχει συντακτικά λάθη δεν μπορεί να πραγματοποιηθεί η εκτέλεσή του μέχρι να διορθωθούν.

Παραδείγματα

```
print 'Hello'           λείπουν οι παρενθέσεις
Print ('Hello')        λανθασμένη εντολή (print και όχι Print)
```

Λογικά Λάθη: είναι δύσκολα στην ανίχνευσή τους. Είναι λάθη στον κώδικα του προγράμματος που έχουν ως αποτέλεσμα εσφαλμένη ή απρόβλεπτη συμπεριφορά. Είναι ένας τύπος σφάλματος που συμβαίνει κατά τη διάρκεια εκτέλεσης του προγράμματος που μπορεί απλά να παράξει λάθος αποτέλεσμα ή να προκαλέσει τον πρόωρο τερματισμό του προγράμματος.

Παράδειγμα

Το πιο κάτω πρόγραμμα δέχεται δύο ακέραιους αριθμούς και υπολογίζει και τυπώνει το άθροισμά τους.

```
number1 = int(input('Δώσε τον πρώτο αριθμό:'))
number2 = int(input('Δώσε τον δεύτερο αριθμό:'))
sum = number1-number2
print(sum)
```

Το πρόγραμμα υπολογίζει τη διαφορά και όχι το άθροισμα.

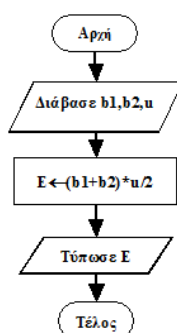
9. Ακολουθιακή Δομή

9.1 Μετατροπή Λογικού Διαγράμματος σε Πρόγραμμα

Για να μετατρέψουμε ένα πρόβλημα σε πρόγραμμα, όπως ήδη έχουμε δει, πρέπει πρώτα να ακολουθήσουμε τα στάδια ανάπτυξης και να σχεδιάσουμε το λογικό διάγραμμα που μας δίνει τη σχηματική λύση του προβλήματος. Αν το λογικό διάγραμμα που σχεδιάσαμε είναι σωστό τότε το μόνο που μένει είναι να το μεταφράσουμε στη γλώσσα προγραμματισμού που θέλουμε.

Παράδειγμα 1

Έστω ότι έχουμε το παρακάτω λογικό διάγραμμα. Ακολουθούμε τα παρακάτω βήματα για να το μετατρέψουμε σε πρόγραμμα στη γλώσσα προγραμματισμού **Python**.



Ξεκινώντας από την αρχή μετατρέπουμε το κάθε σχήμα σε εντολή, όπως φαίνεται στον πιο κάτω πίνακα.

| Σχήμα | Εντολή |
|-------|----------------------------------------------------------------------------------------------------|
| | |
| | <pre>b1 = float(input('Δώσε b1:')) b2 = float(input('Δώσε b2:')) u = float(input('Δώσε u:'))</pre> |
| | <pre>E = (b1+b2) * u / 2</pre> |
| | <pre>print(E)</pre> |
| | |

10. Η εντολή if...else

Για να υλοποιήσουμε προβλήματα δομής διακλάδωσης πρέπει να χρησιμοποιήσουμε την εντολή `if...else`. Η εντολή ελέγχει μια συνθήκη, αν η συνθήκη είναι αληθής (`true`) τότε εκτελείται μια ομάδα εντολών, διαφορετικά (`false`) εκτελείται μια άλλη ομάδα εντολών.

Σύνταξη εντολής

```
if <συνθήκη>:
```

```
    Ομάδα εντολών 1
```

```
else:
```

```
    Ομάδα εντολών 2
```

Για να πληκτρολογήσουμε τις εντολές της ομάδας εντολών 1 θα πρέπει να χρησιμοποιήσουμε εσοχή (indent). Η εσοχή θα πρέπει να είναι η ίδια για όλες τις εντολές της ομάδας.

Η ομάδα εντολών 2 δεν είναι υποχρεωτική. Μπορεί να έχουμε δομή διακλάδωσης χωρίς `else`.

Στις συνθήκες γενικά χρησιμοποιούμε τα συγκριτικά σύμβολα, όπως φαίνεται πιο κάτω:

Πίνακας Συγκριτικών Συμβόλων

| | |
|------------------|----|
| Ισότητα | == |
| Μεγαλύτερο | > |
| Μεγαλύτερο ή Ίσο | >= |
| Μικρότερο | < |
| Μικρότερο ή Ίσο | <= |
| Άνισο | != |

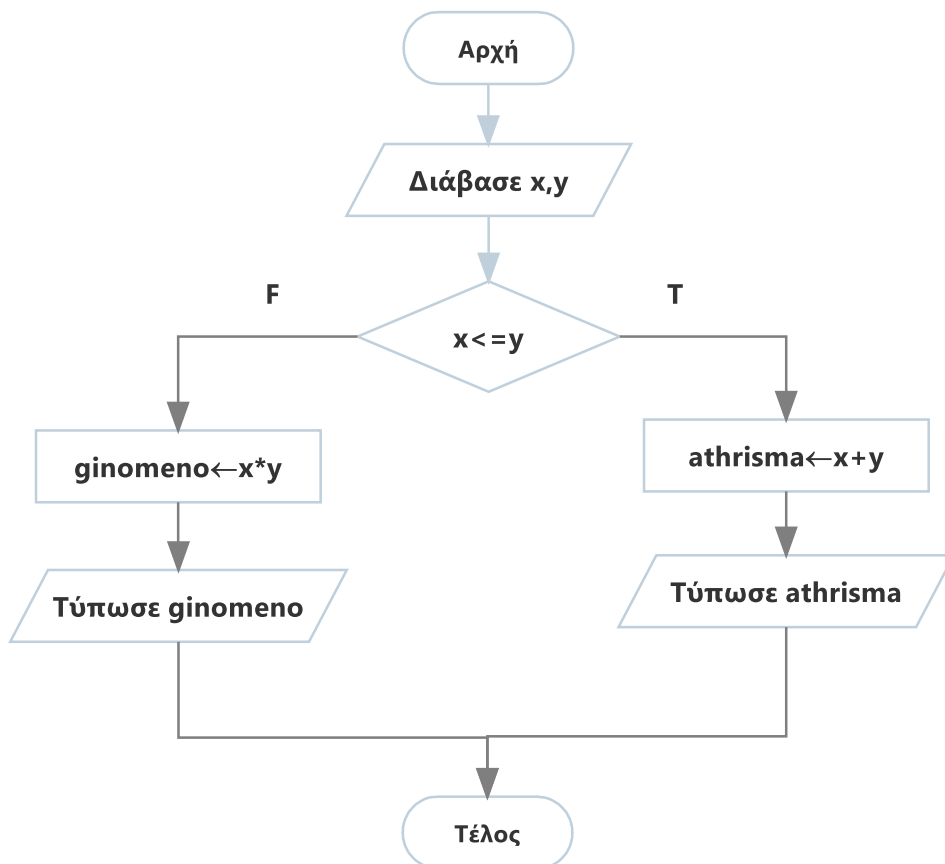
Παράδειγμα 1

Να δημιουργήσετε ένα πρόγραμμα το οποίο να δέχεται δυο ακέραιους αριθμούς και να τυπώνει τον μεγαλύτερο.

| Πρόγραμμα | Επεξήγηση |
|------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre> a = int(input('Δώσε τον 1ο αριθμό:')) b = int(input('Δώσε τον 2ο αριθμό:')) if a>b: print(a) else: print(b) </pre> | <p>Διάβασε 2 αριθμούς.</p> <p>Αν ο a είναι μεγαλύτερος από τον b τύπωσε τον a</p> <p>διαφορετικά τύπωσε τον b</p> |

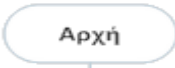
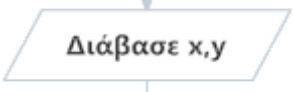
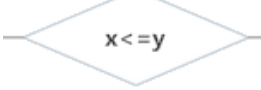

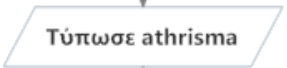


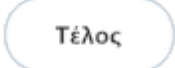
Παράδειγμα 2

Να δημιουργήσετε ένα πρόγραμμα το οποίο να δέχεται δύο πραγματικούς αριθμούς. Αν ο πρώτος αριθμός είναι μικρότερος ή ίσος με τον δεύτερο τότε να υπολογίζεται και να τυπώνεται το άθροισμά τους, αλλιώς να υπολογίζεται και να τυπώνεται το γινόμενό τους.



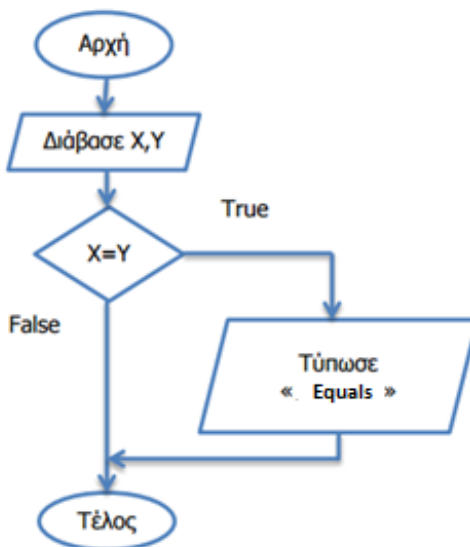
Γ.7. Μ4 – Γ.7.Μ13 Γ.7 Αλγοριθμική Σκέψη, Προγραμματισμός και Σύγχρονες Εφαρμογές

Ξεκινώντας από την αρχή μετατρέπουμε το κάθε σχήμα σε εντολή, όπως φαίνεται στον πιο κάτω πίνακα.

| Σχήμα | Εντολή |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------|
|  | |
|  | <pre>x = float(input('Δώσε x:')) y = float(input('Δώσε y:'))</pre> |
|  | <pre>if x<=y:</pre> |
| <p>T</p>   | <pre> athrisma=x+y print(athrisma)</pre> |
| <p>F</p>   | <pre>else: ginomeno=x*y print(ginomeno)</pre> |
|  | |

Παράδειγμα 3

Να δημιουργήσετε πρόγραμμα το οποίο να δέχεται δύο ακέραιους αριθμούς. Αν οι αριθμοί είναι ίσοι να τυπώνει το μήνυμα «Equals».



Ξεκινώντας από την αρχή μετατρέπουμε το κάθε σχήμα σε εντολή, όπως φαίνεται στον πιο κάτω πίνακα.

| Σχήμα | Εντολή |
|-------|------------------------------------------------------------------------------------|
| | |
| | <code>X = int(input('Δώσε X: '))</code> <code>Y = int(input('Δώσε Y: '))</code> |
| | <code>if X==Y:</code> |
| | <code>print('Equals')</code> |
| | |

Βασικές Έννοιες

| | |
|-------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Μεταβλητή: | Θέση μνήμης που χρησιμοποιούμε για να αναθέσουμε μια τιμή, η οποία όμως μπορεί να αλλάζει κατά τη διάρκεια εκτέλεσης του προγράμματος. |
| int: | Τύπος δεδομένων ο οποίος αντιπροσωπεύει τους ακέραιους αριθμούς. |
| float: | Τύπος δεδομένων ο οποίος αντιπροσωπεύει τους πραγματικούς αριθμούς. |
| string: | Τύπος δεδομένων ο οποίος δέχεται αλφαριθμητικά δεδομένα. |
| input: | Εντολή για είσοδο δεδομένων. Π.χ. <code>X = int(input('Δώσε X:'))</code> |
| print: | Εντολή για έξοδο πληροφοριών. Π.χ. <code>print(embadon)</code> |
| = | Σύμβολο εντολής εκχώρησης τιμής (επεξεργασία, υπολογισμός). Η τιμή της έκφρασης στο δεξί μέρος αποθηκεύεται στη μεταβλητή στο αριστερό μέρος. Π.χ. <code>a = b*(c+d)</code> |
| if-else: | Εντολή η οποία αναπαριστά τη δομή διακλάδωσης. Π.χ. <code>if a>b: c = a+b d = 4*a else: c = a-b d = 4*b</code> |

Υπόμνημα

| Λειτουργία | Μενού |
|--------------------|------------------------------|
| Νέο αρχείο | File⇒New File |
| Αποθήκευση αρχείου | File⇒Save File⇒Save As... |
| Άνοιγμα αρχείου | File⇒Open... |
| Εκτέλεση (run) | Run⇒Run Module |

