ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

Πληροφορική για Νηπιαγωγούς

ΤΖΕΝΗ ΠΑΓΓΕ
Επίκ. Καθηγήτρια

Δρ ΠΑΝΟΣ ΜΑΚΡΗΣ

ΙΩΑΝΝΙΝΑ 2000

ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

Πληροφορική για Νηπιαγωγούς

ΤΖΕΝΗ ΠΑΓΓΕ
Επίκ. Καθηγήτρια

Δρ ΠΑΝΟΣ ΜΑΚΡΗΣ

ΙΩΑΝΝΙΝΑ 2000

Μακρής Πάνος - Παγγέ Τζένη

Πληροφορική για Νηπιαγωγούς

Μάρτιος 2000

Γ' Έκδοση

Προηγούμενες εκδόσεις: Α' Έκδοση 1997
Β' Έκδοση 1998

ISBN 960-233-034-1

Πρόλογος

Τις τελευταίες δεκαετίες οι Ηλεκτρονικοί Υπολογιστές (Η/Υ) χρησιμοποιούνται παράλληλα σε όλους σχεδόν τους τομείς της καθημερινής μας ζωής και στο χώρο της επιστήμης ως επιστημονικά εργαλεία έρευνας. Σ' αυτούς στηρίζεται βασικά η Πληροφορική, η οποία βρίσκεται σε ραγδαία ανάπτυξη. Δεν υπάρχει σήμερα τομέας της επιστήμης χωρίς έστω και την απλή εφαρμογή της. Η θέση της μάλιστα άρχισε να γίνεται αναγκαία και στην εκπαίδευση από το 1980 και μετά.

Ωστόσο, η Πληροφορική είναι φυσικό συνεχώς να εξελίσσεται με την συνεξέλιξη της γενικότερης τεχνολογίας. Το γεγονός αυτό έχει άμεσο αντίκτυπο στο πώς εφαρμόζεται η Πληροφορική στην εκπαίδευση και πώς διδάσκεται στα σχολεία. Γιατί η Πληροφορική σαν μάθημα θεωρείται αναγκαία όχι μόνο για τους φοιτητές και τους μαθητές των ανωτέρων τάξεων της εκπαίδευσης, αλλά και για τους μαθητές των κατωτέρων βαθμίδων της εκπαίδευσης.

Είναι γεγονός ότι υπάρχει μια αρκετά εκτεταμένη βιβλιογραφία για το αντικείμενο της Πληροφορικής γενικά. Αυτό που δεν υπάρχει, είναι μια άφθονη και προσιτή βιβλιογραφία εξειδικευμένη για τους φοιτητές και μάλιστα των Παιδαγωγικών Τμημάτων. Το βιβλίο μας έρχεται να καλύψει αυτήν την ανάγκη και μάλιστα στο Παιδαγωγικό Τμήμα Νηπιαγωγών. Το βιβλίο περιλαμβάνει βασικές γνώσεις με τις οποίες εισάγεται ο φοιτητής στην Πληροφορική από την άποψη που τον αφορά, γι' αυτό και στη διάταξη της ύλης τηρήθηκε μια ειδική σειρά, για καθαρά παιδαγωγικούς λόγους. Στην αρχή παρατίθενται ορισμένες βασικές γνώσεις και αρχές της Πληροφορικής, κατόπιν εξετάζεται αναλυτικά ο Ηλεκτρονικός Υπολογιστής (Η/Υ), στη συνέχεια αναπτύσσονται η παράλληλη επεξεργασία τα δίκτυα και οι επικοινωνίες και τέλος περιγράφεται αναλυτικά η πρακτική εφαρμογή του, και ειδικά θέματα σχετικά με τη χρήση του υπολογιστή στην εκπαίδευση.
Βεβαίως για τη διευκόλυνση του αναγνώστη, στο τέλος του βιβλίου, αναφέρονται στο Παράρτημα, στοιχεία γλωσσών προγραμματισμού Η/Υ και παρατίθεται σχετική βιβλιογραφία.

Το βιβλίο έχει ενιαία μορφή και δομή. Ωστόσο, για τα κεφάλαια σχετικά με την αρχιτεκτονική και το σύστημα του Η/Υ την ειδικότερη επιμέλεια και ευθύνη φέρει ο κ. Πάνος Μακρής.

Πάνος Μακρής και Τζένη Παγγέ, 2000

Περιεχόμενα

ΠΑΡΑΡΤΗΜΑΤΑ - Ο ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΤΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΜΕΡΟΣ Ι

Ο ΑΝΘΡΩΠΟΣ

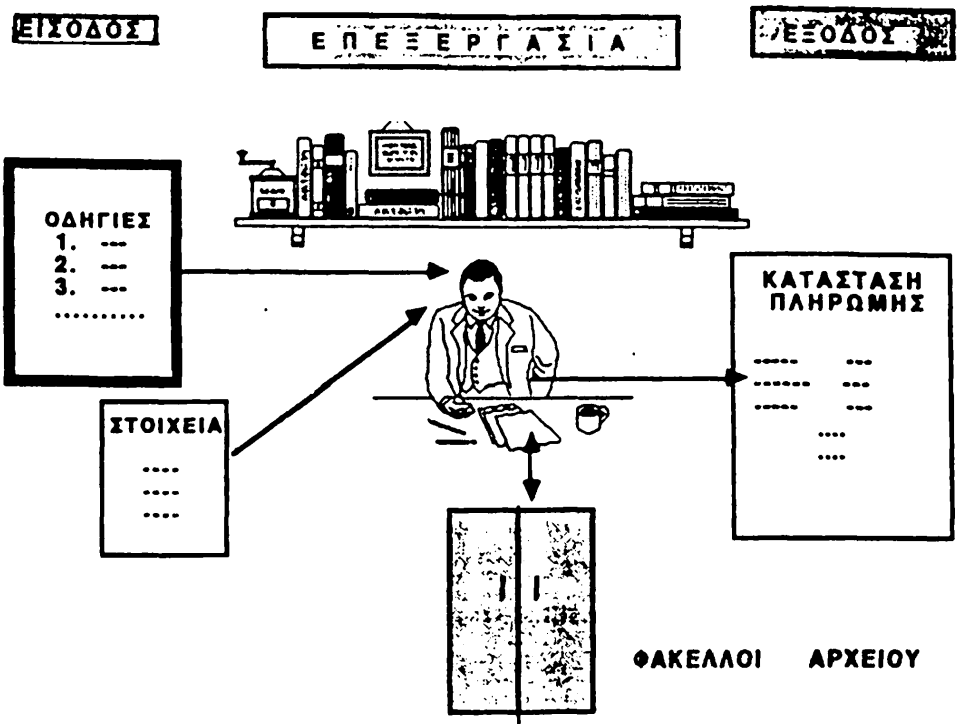
Η ΠΛΗΡΟΦΟΡΙΑ

Ο ΥΠΟΛΟΓΙΣΤΗΣ

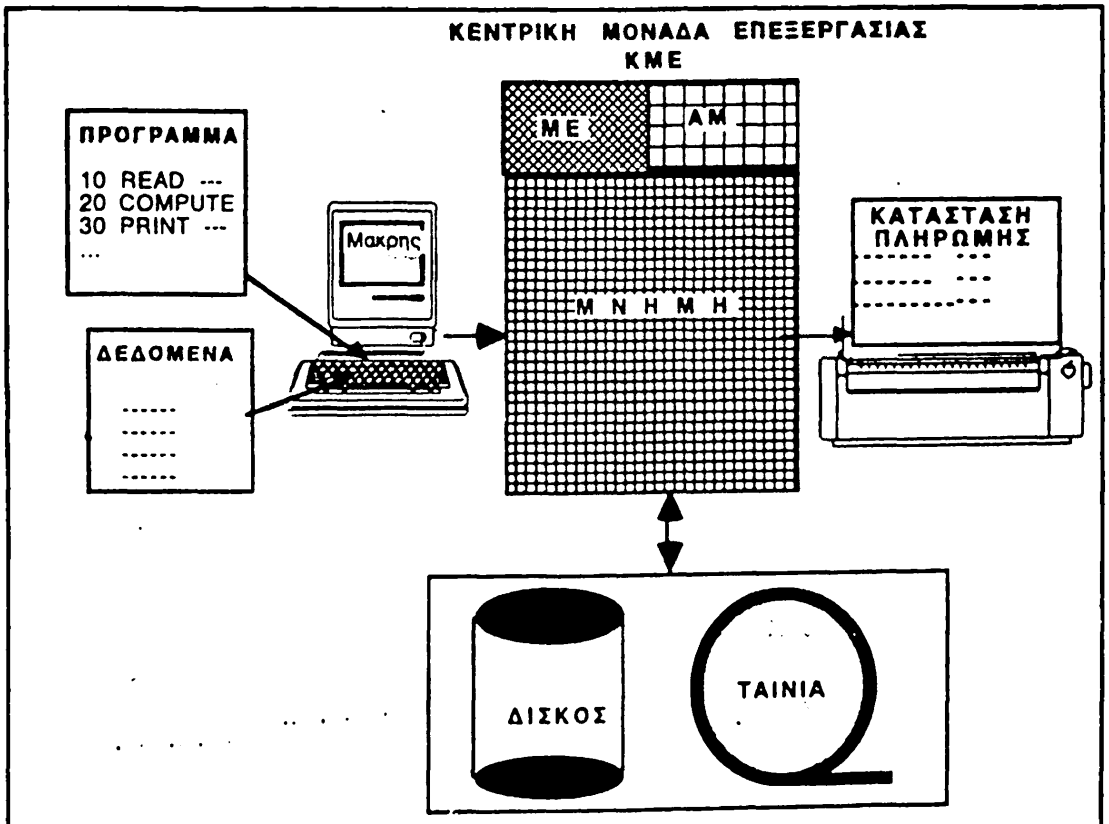
1 Ο ΑΝΘΡΩΠΟΣ ΚΑΙ Ο ΥΠΟΛΟΓΙΣΤΗΣ

Ασφαλώς θα έχετε δεί πολλούς υπολογιστές μικρούς αλλά και μεγαλύτερους, μηχανήματα ακατανόητα, σιδερένια κουτιά, οθόνες, φωτάκια να αναβοσβύνουν, καλώδια.Θα έχετε ακούσει πολλά για τη *λογική* του υπολογιστή, για τις *αποφάσεις* του υπολογιστή. Ισως συνηθίσατε στην ιδέα και στο σχήμα τους και τα πήρατε σαν δεδομένα. Αυτός είναι ο υπολογιστής θα είπατε, πίστευε και μή ερεύνα, έτσι πρέπει να είναι όπως τα αυτοκίνητα είναι έτσι όπως είναι, οι οικιακές συσκευές και τόσα άλλα πράγματα στη ζωή μας και μείς οι ίδιοι οι άνθρωποι στο κάτω της γραφής. Αναρωτηθήκατε όμως αλήθεια ποτέ γιατί ο υπολογιστής είναι έτσι και όχι αλλιώς; Πώς και γιατί αυτοί οι σοφοί που φροντίζουν πάντα πρίν από μας για μάς τον έφτιαξαν έτσι και όχι αλλιώς;

Ας πάρουμε όμως τα πράγματα από την αρχή. Ολοι ξέρουμε πως η μορφή και το περιεχόμενο κάθε εργαλείου που φτιάχνει ο άνθρωπος υπαγορεύονται από την ανάγκη που προορίζεται να ικανοποιήσει και φυσικά από τις τεχνικές κατασκευαστικές δυνατότητες της κάθε εποχής. Ο υπολογιστής προορίζεται να βοηθήσει τον άνθρωπο στις διανοητικές του λειτουργίες, είναι δηλαδή ένα μηχανικό βοήθημα διανοητικής λειτουργίας όπως υπάρχουν μηχανικά βοηθήματα όρασης, ακοής κλπ. Τον βοηθάει στο να φέρει σε πέρας μια δουλειά διανοητική : διεκπεραίωση μιας διαδικασίας, λήψη μιας απόφασης κλπ.



Σχ. 1.1.α Χειρογραφική διαδικασία



Σχ. 1.1.β Μηχανογραφική Διαδικασία

Για να καταλάβουμε λοιπόν πώς και με ποιά κριτήρια ο υπολογιστής φτιάχτηκε έτσι όπως φτιάχτηκε θα προσπαθήσουμε να παρουσιάσουμε με ένα παράδειγμα τον τρόπο με τον οποίο βοηθάει τον άνθρωπο στη διεκπεραίωση μιας απλής κλασικής διαδικασίας όπως είναι η μισθοδοσία.

Διεκπεραίωση μιας διαδικασίας

Ο υπολογισμός της μισθοδοσίας σε μια επιχείρηση

Η μισθοδοσία (ο υπολογισμός συντάξεων κλπ) είναι μια κλασική περίπτωση χρησιμοποίησης υπολογιστών σε μια επιχείρηση. Ο στόχος της διαδικασίας είναι ξεκάθαρος : η κατάρτιση των μισθοδοτικών καταστάσεων με τους καθαρούς μισθούς (και κάποια ανάλυση) των υπαλλήλων. Ας δούμε λοιπόν πρώτα με κάποια συντομία πώς γινότανε η διαδικασία αυτή παλιότερα αποκλειστικά από τους υπαλλήλους μισθοδοσίας και πώς γίνεται σήμερα στις εταιρίες που έχουν υπολογιστή (Υ).

Απο τον άνθρωπο-Χειρογραφική διαδικασία

Για να κάνει τη δουλειά του ο αρμόδιος υπάλληλος μισθοδοσίας χρειάζεται δύο πράγματα:

α. Μερικά βασικά-πρωτογενή στοιχεία για κάθε υπάλληλο της επιχείρησης (πχ. ονοματεπώνυμο, βασικό μισθό, επιδόματα, κρατήσεις κλπ)
β. Ενα σύνολο οδηγιών που θα του λένε πώς θα επεξεργαστεί τα παραπάνω πρωτογενή στοιχεία για κάθε υπάλληλο προκειμένου να καταλήξει στο ζητούμενο (καθαρό πληρωτέο). Οι οδηγίες προκύπτουν από κατάλληλη ανάλυση των νόμων και λοιπών κανονισμών που διέπουν τη μισθοδοσία των διαφόρων κατηγοριών υπαλλήλων της εταιρίας.

Ενα υπεραπλουστευμένο σύνολο τέτοιων οδηγιών θα μπορούσε να είναι το παρακάτω:
1. Πάρε τα βασικά στοιχεία ενός υπαλλήλου
2. Πρόσθεσε τον βασικό του μισθό με τα επιδόματά του
3. Αφαίρεσε τις κρατήσεις του από το παραπάνω άθροισμα. Το αποτέλεσμα είναι το καθαρό πληρωτέο
4. Γράψε στην κατάσταση πληρωμής το ονοματεπώνυμο του υπόψη υπαλλήλου μαζί με το παραπάνω αποτέλεσμα

5. Ξανάρχισε από το βήμα 1 για τον επόμενο υπάλληλο και συνέχισε μέχρι να τελειώσουν όλοι οι υπάλληλοι της εταιρίας.

Για να γίνει τώρα η δουλειά θα πρέπει ο υπάλληλος να πάρει τις παραπάνω οδηγίες απο το χαρτί όπου είναι γραμμένες και να τις βάλει μέσα στο μυαλό του. Η είσοδος αυτή στο μυαλό του ανθρώπου γίνεται με τα κατάλληλα όργανα εισόδου (μάτια).
Εχοντας τώρα ο άνθρωπος τις οδηγίες στο μυαλό του μπορεί να προχωρήσει στην επεξεργασία. Για να το κάνει αυτό ουσιαστικά αρχίζει και εκτελεί τις οδηγίες μία προς μία με τη σειρά που είναι γραμμένες μέχρι να τελειώσουν. Η πρώτη οδηγία του ζητάει να βάλει μέσα στο μυαλό του τα πρωτογενή στοιχεία ενός υπαλλήλου. Η είσοδος αυτή γίνεται πάλι με το γνωστό όργανο εισόδου (μάτια) από την κατάσταση όπου είναι γραμμένα τα βασικά στοιχεία των υπαλλήλων. Στη συνέχεια προχωράει στις υπόλοιπες οδηγίες. Κάνοντας αυτά που του λένε οι οδηγίες προφανώς καταλήγει στο ζητούμενο, δηλαδή στην περίπτωσή μας στον υπολογισμό του καθαρού πληρωτέου. Η τέταρτη οδηγία τώρα (βήμα 4) του ζητάει να παρουσιάσει στον έξω κόσμο το αποτέλεσμα που βρήκε και έχει ακόμα στο μυαλό του. Αυτή η έξοδος αποτελεσμάτων από το μυαλό προς τον έξω κόσμο (πχ κατάσταση μισθοδοσίας) γίνεται με το κατάλληλο όργανο εξόδου (πχ χέρι-μολύβι-γράψιμο). Ιδιαίτερη σημασία έχει η τελευταία οδηγία (βήμα 5) που δείχνει την *μηχανική* επαναληπτικότητα, την ρουτίνα και το χρονοβόρο της διαδικασίας.

Βλέπουμε λοιπόν οτι για να διεκπεραιωθεί η εργασία της μισθοδοσίας χρειάζονται οι εξής επι μέρους διαδικασίες :
- εισαγωγή των οδηγιών και των κατά περίπτωση απαραίτητων βασικών στοιχείων απο τον έξω κόσμο στο μυαλό του ανθρώπου
- εκτέλεση των οδηγιών μέσα στο μυαλό του ανθρώπου, η οποία απαιτεί την κατάλληλη επεξεργασία των στοιχείων και τον υπολογισμό του ζητουμένου αποτελέσματος.
- έξοδος των αποτελεσμάτων απο το μυαλό του ανθρώπου στον έξω κόσμο.

Και κάτι ακόμα για να ολοκληρώσουμε την παρουσίαση. Πολλές φορές ο αρμόδιος υπάλληλος της μισθοδοσίας μπορεί να χρειαστεί για να κάνει τη δουλειά του μερικά λεπτομερειακά στοιχεία για κάποιο υπάλληλο.Τότε θα διαπιστώσει οτι δεν αρκούν τα βασικά στοιχεία που έχει πρόχειρα για κάθε

υπάλληλο αλλά θα πρέπει να ανατρέξει στον αναλυτικό φάκελλό του. Οι φάκελλοι αυτοί περιέχουν όλα τα σχετικά με κάθε υπάλληλο έγγραφα και στοιχεία και φυλάγονται σε μεγάλα ντουλάπια. Οι όγκοι αυτοί των αναλυτικών στοιχείων είναι τα λεγόμενα αρχεία.

Από τα παραπάνω συνάγεται πως για να φέρει ο άνθρωπος σε πέρας τη δουλειά αυτή, όπως και κάθε παρόμοια, πρέπει να διαθέτει τα αντίστοιχα μέσα-όργανα εισαγωγής στοιχείων (πχ μάτια, αυτιά), επεξεργασίας στοιχείων (πχ μυαλό) και εξόδου αποτελεσμάτων (πχ στόμα για να τα πεί, χέρια για να τα γράψει). Τέλος σε περίπτωση που χρειαστεί να ανατρέξει σε αρχεία θα πρέπει να έχει διαθέσιμα και τα απαραίτητα μέσα τήρησης των αρχείων αυτών (πχ. χαρτί, ντουλάπια).

Απο τη μηχανή-Μηχανογράφηση

Τα παραπάνω βέβαια ίσχυαν στην περίπτωση που τη δουλειά την έκανε ο άνθρωπος. Μια πιο προσεκτική παρατήρηση όμως θα έδειχνε πως τα στάδια διεκπεραίωσης μιας εργασίας όπως τα διατυπώσαμε παραπάνω δηλ. η εισαγωγή των στοιχείων, η επεξεργασία τους και η έξοδος των αποτελεσμάτων είναι μια φυσιολογική αλληλουχία διαδικασιών που θα πρέπει να ισχύει ανεξάρτητα από το ποιός κάνει την εργασία. Είτε άνθρωπος λοιπόν είτε μηχανή κάνει τη δουλειά θα πρέπει να διαθέτει κάποια μέσα (όργανα-μηχανήματα) για εισαγωγή των στοιχείων, επεξεργασία τους, έξοδο των αποτελεσμάτων και τήρηση των αρχείων.

Προκειμένου λοιπόν για μηχανή εκείνο που έχει σημασία είναι να βρεθεί ο κατάλληλος τρόπος να κατασκευαστούν μηχανήματα που θα εξομοιώνουν τις ανθρώπινες λειτουργίες εισαγωγής -επεξεργασίας -εξόδου στοιχείων. Πώς δηλαδή θα μπορεί να φτιαχτεί ένα μηχάνημα που να *διαβάζει* στοιχεία απο τον έξω κόσμο (μηχανικό μάτι ας πούμε), άλλο που να επεξεργάζεται στοιχεία (μηχανικό μυαλό), άλλο που να *γράφει* στοιχεία (μηχανικό χέρι) και άλλο που να κρατάει αρχεία (ηλεκτρονικό ντουλάπι). Εκεί μετράει πράγματι η φαντασία του κατασκευαστή και φυσικά η πρόοδος της επιστήμης και της τεχνολογίας. Αυτό ήταν το ζητούμενο εδώ και πολλά χρόνια, το πρόβλημα ήταν να βρεθεί μια τεχνικά εφικτή λύση. Μια τέτοια λύση άρχισε να παίρνει σάρκα και οστά στην Αμερική από τα τέλη της δεκαετίας του 40 και αυτή η λύση ονομάστηκε computer ή ηλεκτρονικός εγκέφαλος ή ηλεκτρονικός υπολογιστής ή απλά υπολογιστής (Υ).

Από τότε η προσπάθεια συνεχίζεται με στόχο την κατασκευή μηχανημάτων που θα κάνουνε τις αντίστοιχες λειτουργίες όσο γίνεται πιό φυσικά - ανθρώπινα, διευκολύνοντας έτσι την επικοινωνία ανθρώπου μηχανής. Δηλαδή μηχανήματα που θα μπορούν να *διαβάζουν* τα στοιχεία κατευθείαν από το χαρτί όπου είναι γραμμένα ή να τα *ακούνε* όταν κάποιος άνθρωπος τους τα διαβάζει και μετά να *γράφουν* ή να *μιλάνε* τα αποτελέσματα. Αυτά βέβαια αφορούν στην έρευνα και στους στόχους που υπάρχουν, στο βιβλίο αυτό θα αναφερθούμε κυρίως στο τί υπάρχει σήμερα από πλευράς μηχανημάτων εισόδου, επεξεργασίας, εξόδου και αρχείων.
Ας δούμε λοιπόν τώρα τί γίνεται όταν είναι μια μηχανή εκείνη που κάνει τη δουλειά της μισθοδοσίας. Καταρχή θα πρέπει να πούμε οτι αν έχουμε μια μηχανή να μας κάνει μια δουλειά λέμε οτι έχουμε μηχανογράφηση της δουλειάς αυτής. Η μηχανογράφηση στη διεθνή ορολογία λέγεται data processing. Στα τελευταία χρόνια η μηχανογράφηση συχνά αποδίδεται με το γενικότερο όρο πληροφορική (αγγλικά informatics). Εδώ θα πρέπει να πούμε οτι επειδή η ανάπτυξη των υπολογιστών (Υ) άρχισε στην Αμερική μεγάλο μέρος της μηχανογραφικής ορολογίας έχει διατυπωθεί στην αγγλική και στη συνέχεια διεθνοποιηθεί. Κατά καιρούς οι διάφορες χώρες προσπάθησαν να διαμορφώσουν τοπικές ορολογίες με μεγαλύτερη ή μικρότερη επιτυχία. Το ίδιο έγινε και στη χώρα μας από μεμονωμένους φορείς που δεν κατάφεραν όμως να καταλήξουν σε μιά κοινά αποδεκτή ορολογία. Στο βιβλίο μας θα χρησιμοποιούμε συχνά και τον αγγλικό-διεθνή όρο για να διευκολύνουμε εκείνους που κάποια στιγμή θα χρειαστεί να μελετήσουν αγγλική βιβλιογραφία .

Ας ξαναγυρίσουμε όμως στη μισθοδοσία. Μια δουλειά, ένα σύστημα γενικότερα που διεκπεραιώνει ο Υ (στην περίπτωσή μας η μισθοδοσία) στη μηχανογραφική ορολογία λέγεται εφαρμογή. Ας έλθουμε τώρα στην ίδια την εργασία την εφαρμογή της μισθοδοσίας. Η δουλειά βέβαια καθεαυτή και οι απαιτήσεις της (η έκδοση μισθοδοτικών καταστάσεων) παραμένει η ίδια είτε άνθρωπος είτε μηχανή την εκτελούν. Ιδια ακόμα παραμένει η ανάγκη να έχουμε τα βασικά στοιχεία και τις οδηγίες όπως και στην περίπτωση του ανθρώπου. Οπως και ο άνθρωπος έτσι και ο υπολογιστής θα πρέπει να *διαβάσει* τα στοιχεία και τις οδηγίες στη μνήμη του, να εκτελέσει τις οδηγίες μία προς μία στο *μυαλό* του και τέλος να *γράψει* τα αποτελέσματα. Και εδώ στην εκτέλεση από τον Υ της τελευταίας οδηγίας, της οδηγίας της επαναληπτικότητας (βήμα 5) βλέπουμε την χρησιμότητα

του Υ. Βλέπουμε δηλαδή πώς βγάζουμε τη ρουτίνα της επανάληψης από τον άνθρωπο και τη δίνουμε στο μηχάνημα που την κάνει αγόγγυστα και μάλιστα πολύ πιό γρήγορα από εκείνον.

Τα βασικά στοιχεία (δηλ τα στοιχεία που εισάγομε στον Υ) στη μηχανογραφική ορολογία λέγονται δεδομένα ή data. Ας δούμε τώρα τις οδηγίες. Εδώ υπεισέρχεται και άλλος ένας παράγοντας: η γλώσσα όπου θα πρέπει να είναι γραμμένες οι οδηγίες. Οπως δηλαδή στην περίπτωση του ανθρώπου οι οδηγίες θα πρέπει να είναι γραμμένες στη γλώσσα εκείνου που θα τις εκτελέσει (ελληνικά αν αυτός είναι έλληνας ,αγγλικά αν είναι άγγλος κλπ) έτσι και τώρα στην περίπτωση της μηχανής θα πρέπει να είναι γραμμένες σε κατάλληλη γλώσσα που καταλαβαίνει η μηχανή. Οι οδηγίες γραμμένες σε γλώσσα που καταλαβαίνει η μηχανή είναι το λεγόμενο πρόγραμμα. Κάθε μια οδηγία ξεχωριστά λέγεται εντολή του προγράμματος.

Οι γλώσσες που καταλαβαίνουν οι υπολογιστές και στις οποίες μπορούν να γραφούν τα προγράμματα λέγονται γλώσσες προγραμματισμού. Κανονικά η *φυσική* γλώσσα των Υ είναι πολύ δύσκολη για τον άνθρωπο (κάτι σαν τα κινέζικα ας πούμε) γιαυτό από πολύ νωρίς ο άνθρωπος προσπάθησε να *μάθει* τον Υ να *μιλάει* ξένες γιαυτόν αλλά πιό οικείες για τον ίδιο (τον άνθρωπο) γλώσσες. Γλώσσες δηλαδή που να μοιάζουν με τις ανθρώπινες φυσικές γλώσσες και μιά και η προσπάθεια άρχισε στην Αμερική καθιερώθηκε η γλώσσα αυτή να είναι η αγγλική. Ετσι δημιουργήθηκε μια ολόκληρη σειρά από γλώσσες προγραμματισμού που μοιάζουν με τα αγγλικά. Τέτοιες γλώσσες είναι η BASIC, COBOL, FORTRAN, PASCAL, ADA κλπ

Ο Υπολογιστής

Ας δούμε όμως τώρα τον υπολογιστή τον ίδιο. Οπως είπαμε και παραπάνω για να κάνει μιά δουλειά θα χρειαστεί μηχανήματα εισαγωγής στοιχείων, επεξεργασίας, εξόδου αποτελεσμάτων ή με άλλα λόγια μηχανήματα επικοινωνίας με τον έξω κόσμο (είσοδος-έξοδος), εσωτερικής επεξεργασίας καθώς και μηχανήματα αρχείων.

Τα μηχανήματα εισόδου έχουν περάσει πολλές αλλαγές από τότε που πρωτοπαρουσιάστηκε ο Υ και ασφαλώς θα περάσουν κι άλλες. Τη στιγμή αυτή υπάρχουν αρκετά μηχανήματα εισόδου αλλά το πιό διαδεδομένο είναι

το λεγόμενο τερματικό εφοδιασμένο με οθόνη και πληκτρολόγιο. Το καθαρά μηχάνημα εισόδου αν θέλουμε να κυριολεκτήσουμε είναι το πληκτρολόγιο με το οποίο ο άνθρωπος πληκτρολογεί ό,τι θέλει να βάλει μέσα στον Υ (προγράμματα, στοιχεία). Το πληκτρολόγιο είναι συνδεδεμένο με τον κυρίως Υ και έτσι τα στοιχεία περνάνε κατευθείαν στο εσωτερικό του Υ για να γίνει η επεξεργασία τους.

Το εσωτερικό αυτό του Υ, τα μηχανήματα δηλαδή επεξεργασίας (το μυαλό) του Υ ονομάζονται Κεντρική Μονάδα. Η ΚΜ αποτελείται από τη λεγόμενη Μνήμη (memory ή store) όπου αποθηκεύονται τα προγράμματα και τα στοιχεία που πρόκειται να επεξεργαστούν και τον Επεξεργαστή (Processor) ή Κεντρική Μονάδα Επεξεργασίας - ΚΜΕ (αγγλικά Central Processing Unit - CPU). Ο Επεξεργαστής αποτελείται από τη λεγόμενη Μονάδα Ελέγχου - ΜΕ (Control Unit) που έχει την αποστολή να παίρνει τις εντολές του προγράμματος από τη μνήμη μία-μία και να τις εκτελεί. Τέλος έχουμε την Αριθμητική Μονάδα - ΑΜ (Arithmetic Unit) που αναλαμβάνει να κάνει τις αριθμητικές πράξεις που ζητάνε οι εντολές.

Σαν μηχανήματα εξόδου θεωρούμε εκείνα τα μηχανήματα που επιτρέπουν στον Υ να μας παρουσιάζει τα αποτελέσματα της επεξεργασίας των στοιχείων που έκανε στην Κεντρική Μονάδα. Τέτοια μηχανήματα είναι ο γνωστός εκτυπωτής όπως επίσης και η οθόνη του τερματικού που είδαμε παραπάνω.

Σαν μηχανήματα αρχείων για την αποθήκευση μεγάλων όγκων δεδομένων (δηλ. το ηλεκτρονικό ισοδύναμο των ντουλαπιών) χρησιμοποιούνται οι μαγνητικοί δίσκοι και οι μαγνητικές ταινίες. Κάτι ανάλογο όπως θα δούμε παρακάτω των δίσκων του πικ-απ και των ταινιών του μαγνητοφώνου ή του κασετόφωνου αντίστοιχα.

Καταλήγοντας μπορούμε να πούμε οτι κάθε σύστημα υπολογιστή από τον πιο μεγάλο μέχρι τον πιο μικρό αποτελείται αναγκαστικά από μηχανήματα εισόδου, επεξεργασίας, εξόδου και (αν χρειάζεται) αρχείων. Εκείνο που διαφέρει και χαρακτηρίζει τα μεγάλα και τα μικρά συστήματα Υ είναι η πολυπλοκότητα, ο όγκος και ο αριθμός μηχανημάτων απο την κάθε κατηγορία που αποτελούν το σύστημα του υπολογιστή. Το σύνολο των μηχανημάτων αυτών λέγεται με μια λέξη εξοπλισμός ή υλικό του συστήματος του Υ. Ακόμα συναντάται συχνά με το αγγλικό-διεθνή όρο

hardware (χάρντγουερ) που υποδηλώνει στα αγγλικά το σκληρό (hard), το σιδερικό.

Σε αντιδιαστολή τα προγράμματα ονομάστηκαν software λόγω της *μαλακής*-soft φύσης τους (μπορούμε εύκολα να τα αλλάξουμε όταν αλλάξουμε κάποιες εντολές τους). Ελληνικά αποδόθηκαν σαν λογικό ή λογισμικό. Στο παράδειγμά μας της μισθοδοσίας οι πέντε οδηγίες γραμμένες σε γλώσσα προγραμματισμού είναι ένα software.

Είδη Υπολογιστών

Οι Υπολογιστές έχει επικρατήσει να διακρίνονται σε τρείς μεγάλες κατηγορίες τους:
- Μεγάλους (mainframes)
- Μεσαίους (minis) και τους
- Μικρούς (micros)

Οι μικροί υπολογιστές με τους οποίους θα ασχοληθούμε έχουν την δική τους ιστορία:
- άρχισαν από οικιακοί (home)
- εξελίχθηκαν σε Υ για γραφεία και μικρές επιχειρήσεις
- με τα νέα μοντέλα ανεβαίνουν με προοπτική να γεφυρώσουν το χασμα με τους μεγάλους υπολογιστές.

Σήμερα υπάρχουν δύο βασικά στάνταρ στους μικρούς υπολογιστές:
- οι υπολογιστές IBM και συμβατοί που έχουν το 80% της αγοράς και έχει καθιερωθεί να ονομάζονται PC ή προσωπικοί υπολογιστές και
- οι υπολογιστές της Apple οι λεγόμενοι Macintosh.

Στη συνέχεια του βιβλίου μας θα ασχοληθούμε με τα PC.

ΕΞΕΛΙΞΗ ΤΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
(Ιστορική αναδρομή)

3500 π.Χ.: Ο άβακας (Πέτρινη πλάκα 20εκ. επί 30εκ. με σκαλισμένα πάνω της μικρά αυλάκια στα οποία κινούνταν μικρές σφαίρες) Κατά τον μεσαίωνα ο άβακας είχε χρησιμοποιηθεί τόσο στην Ευρώπη όσο και στην Ασία και τον αραβικό κόσμο.

150 π.Χ.: Ο αστρολάβος των Κυθήρων

1614: Ο λογαριθμικός κανόνας ή πολλαπλασιαστική μηχανή

1642: Ο Blaise Pascal και η Pascalin (πρώτη ψηφιακή αριθμομηχανή με σύστημα οδοντωτών τροχών που έκανε προσθαφαιρέσεις) Η μηχανή αυτή χρησιμοποιούσε μέχρι οκτώ στήλες ψηφίων.

1822: Η "Δορυφορική μηχανή" του Babbage που υπολόγιζε τριγωνομετρικές και λογαριθμικές συναρτήσεις και είναι η πρώτη μηχανή που σχεδιάστηκε για να μπορεί να χρησιμοποιεί προγράμματα με συγκεκριμένες εντολές.

1833: Η "Αναλυτική μηχανή" του Babbage που είχε τα χαρακτηριστικά των σύγχρονων Η/Υ. (Η είσοδος των στοιχείων γινόταν με διάτρητες κάρτες).

1839: Ο George Boole αντιστοίχισε σύμβολα άλγεβρας και λογικής. Το σύστημα του Boole αποτέλεσε την βάση αυτού που είναι σήμερα γνωστό ως άλγεβρα του Boole.

1890: Ο Herman Hollerith και η IBM (κατασκεύασε μια μηχανή με την βοήθεια της οποίας έγινε η απογραφή πληθυσμού των ΗΠΑ)

1936: Ο Allan Turing και η "Καθολική Μηχανή" (έκανε συγκεκριμένο είδος υπολογισμών)

1938: Ο Konrad Zuse και τα μοντέλα Η/Υ Z1 (1938), Z2 (1938), Z3 (1941), Z4 (1944) που τα κατασκεύασε στη Γερμανία

1943: Ο Howard Aiken και ο "MARK 1" (Υπολογιστής αυτόματης διαδοχής πράξεων που κατασκευάστηκε στις ΗΠΑ και βασιζόταν στις αρχές του ηλεκτρομαγνητισμού)

1943: Ο Allan Turing και ο "Κολοσσός" (Ψηφιακοί Η/Υ που κατασκευάστηκαν για λογαριασμό των συμμάχων, αφού αυτοί πρώτα έσπασαν τον κώδικα της Γερμανικής υπολογιστικής μηχανής Enigma)

1943: Οι John Mauchly, John Eckert και ο "ENIAC" (Electronic Numerical Integrator And Calculator) Ολοκληρώθηκε το 1946 και είναι ο πρώτος ψηφιακός ηλεκτρονικός υπολογιστής. Παράλληλα στην Αγγλία είχε αναπτυχθεί μια ειδικής χρήσης υπολογιστική μηχανή ονομαζόμενη 'Κολοσσός' και προοριζόταν για την αποκρυπτογρά-

φηση κωδικών που είχαν δημιουργηθεί από γερμανικές ηλεκτρομηχανικές συσκευές.

1944: Ο John von Neumann και ο υπολογιστής "EDVAC" στις ΗΠΑ. Ο υπολογιστής αυτός είχε τη δυνατότητα να αποθηκεύει ένα πρόγραμμα στο οποίο οι εντολές διαχειρίζονταν με τον ίδιο τρόπο όπως και τα αριθμητικά δεδομένα. Ολοκληρώθηκε η μηχανή αυτή το 1950.

1949: Οι John Mauchly, John Eckert και ο "EDSAC" (Electronic Delay-Storage Automatic Computer) στην Αγγλία

1949: Η μηχανή "AGE" του Allan Turing

1950: Η μηχανή "MANIAC" του John von Neumann

1951: Οι John Mauchly και John Eckert κατασκεύασαν τον "UNIVAC 1" (Universal Automatic Computer) για την υπηρεσία απογραφής. Ο υπολογιστής αυτός όπως όλοι της γενιάς του χρησιμοποιούσε λυχνίες. Η κύρια μνήμη του αποτελείτο από γραμμές καθυστέρησης υδραργύρου, όπου μια λυχνία υδραργύρου χρησίμευε ως ακουστικό μέσο δια μέσου του οποίου οι ηχητικοί παλμοί που αντιπροσώπευαν τα ψηφία 0,1 διαβιβάζονταν πάνω σε καθορισμένη απόσταση. Είναι ο πρώτος υπολογιστής που χειρίζονταν με άνεση αριθμητικές και αλφαβητικές πληροφορίες.

Η μηχανή "LEO",

1962: Εμφανίζονται τα CDC-1604 και NCR-304. Η υπολογιστική μονάδα του CDC-1604 περιείχε 25.000 κρυσταλλοτριόδους

1968: Εμφανίζονται τα IBM-370 και Cyber της CDC

1974: Ο μικροεπεξεργαστής 8080

1975: Διάθεση του ALTAIR 8800 - Εμφάνιση του μικροεπεξεργαστή Z-80

1976: Διάθεση του υπερυπολογιστή CRAY-1

1976: Ίδρυση της εταιρείας APPLE

1979: Αρχίζει η έρευνα από τους Ιάπωνες για την ανάπτυξη συστημάτων Τεχνητής Νοημοσύνης

1979: Διάθεση του 68000 της Motorola

1981: Διάθεση του IBM-PC με χρήση του μικροεπεξεργαστή 8088

1981: Διάθεση του λειτουργικού συστήματος MS-DOS

1983: Αφετηρία για την Τεχνητή Νοημοσύνη

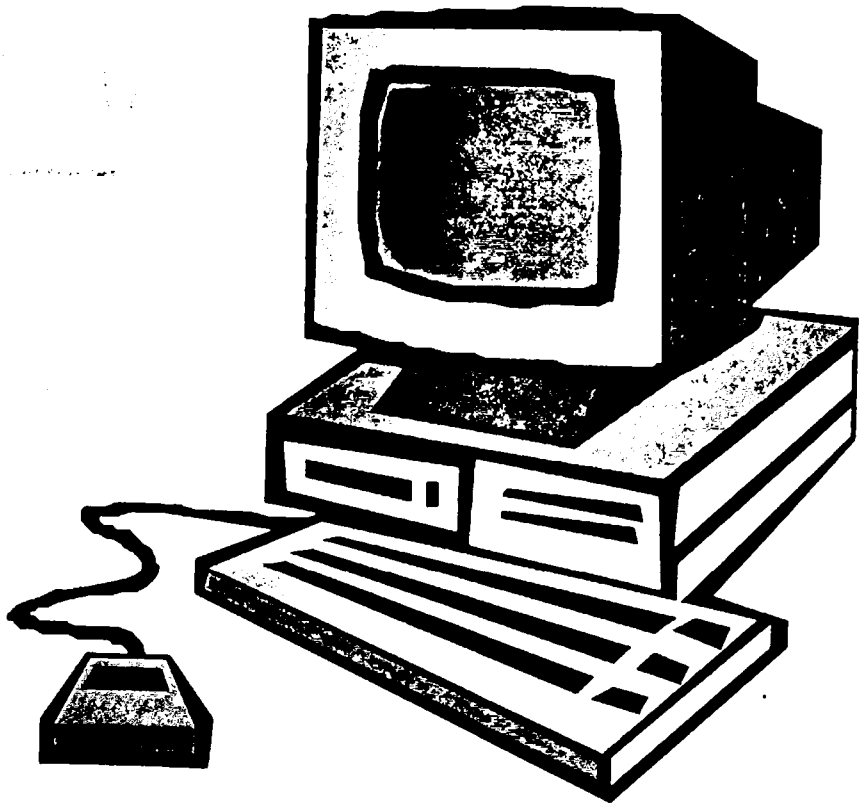
1983: Αφετηρία για τις οπτικές ίνες και τους οπτικούς δίσκους

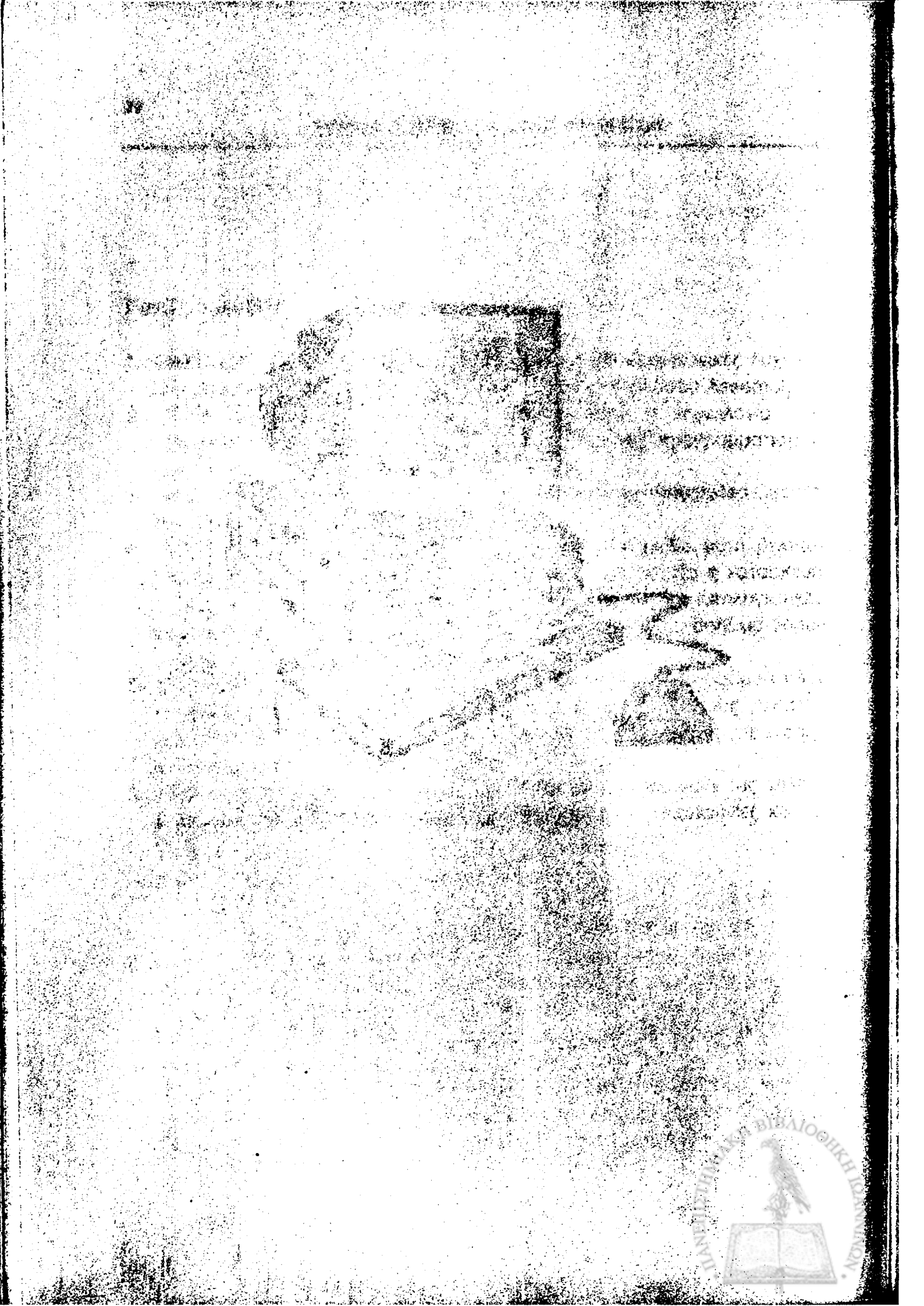
1985: Διάθεση των υπερυπολογιστών CRAY-2, S810-20, SX-2 και VP-400

1993: Διάθεση του μικροεπεξεργαστή pentium

Γενιές των Η/Υ

1. 1946-1957. Οι Η/Υ χρησιμοποιούν αποκλειστικά θερμοϊονικές λυχνίες. Ο προγραμματισμός γίνεται σε γλώσσα μηχανής ή γλώσσα Assembly.
2. 1958-1960. Χρησιμοποιείται για πρώτη φορά η τεχνολογία των Transistors στους Η/Υ. Αναπτύσσονται οι Γλώσσες προγραμματισμού υψηλού επιπέδου Cobol, Fortran κ.α.
3. 1963-1968.Οι Η/Υ χρησιμοποιούν την τεχνολογία των Integrated Circuits (ολοκληρωμένα κυκλώματα).
4. 1970-1978. Τα ολοκληρωμένα κυκλώματα την περίοδο αυτή βελτιώνονται, τοποθετούνται εκατοντάδες τρανζίστορ και αρχίζει η κατασκευή μεγάλων υπολογιστών. Επιπλέον, αναπτύσσονται οι υπολογιστικές μηχανές τσέπης και ανταλλάσσονται πληροφορίες με τη βοήθεια δορυφόρων.
5. 1979-1990. Εμφανίζονται την περίοδο αυτή τα μεγάλου βαθμού ολοκληρωμένα κυκλώματα και αναπτύσσεται η τεχνολογία της ολοκληρωμένης λογικής. Αναπτύσσονται επίσης νέες γλώσσες προγραμματισμού όπως οι Lisp, Prolog, Logo.
6. 1990-Σήμερα. Κατά την περίοδο αυτή γίνεται η συνένωση της τηλεόρασης με τον Η/Υ. Αναπτύσσονται παράλληλα οι επικοινωνίες και τα δίκτυα.





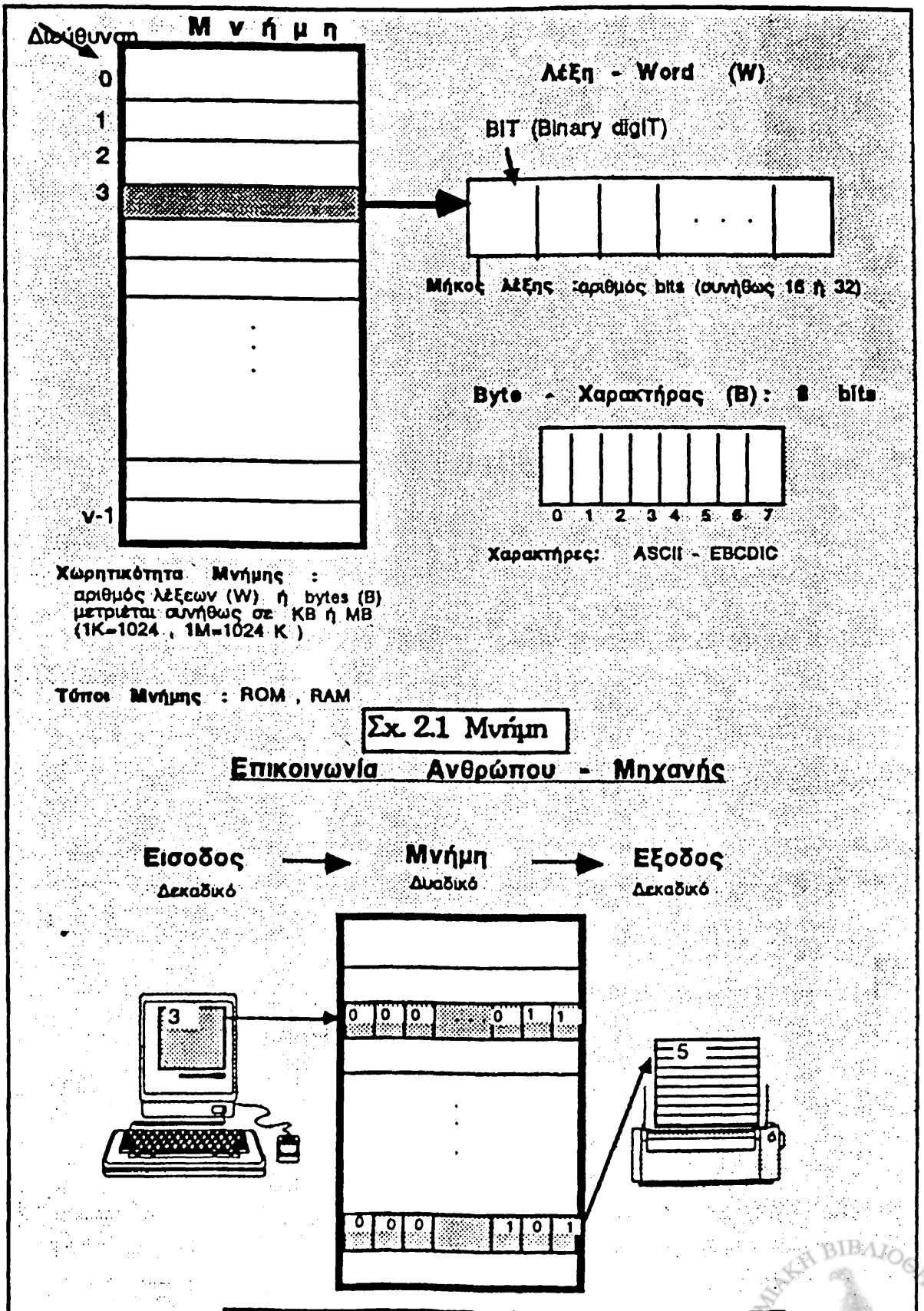
2 ΑΠΟ ΤΗ ΘΕΩΡΙΑ ΣΤΗΝ ΠΡΑΞΗ

Υπολογιστές και Δυαδικό σύστημα

Μιλήσαμε στο προηγούμενο κεφάλαιο για εισαγωγή στοιχείων στον Υ, απομνημόνευση στη Μ, επεξεργασία στον Ε κλπ και προσπαθήσαμε να συσχετίσουμε τις λειτουργίες αυτές με τις αντίστοιχες ανθρώπινες λειτουργίες. Εύκολο να το λέει κανείς, δύσκολο όμως να το πραγματοποιήσει. Εδώ έπρεπε ο άνθρωπος να επιστρατεύσει όλη τη φαντασία και την εφευρετικότητά ώστε να αξιοποιήσει υλικά που υπάρχουν στη φύση, τις δυνατότητες που του δίνει η τεχνολογία και να φτιάξει νέους μηχανισμούς που θα επιτυγχάνουν τους παραπάνω στόχους.

Πώς λειτουργεί η Μνήμη

Ας δούμε από πιό κοντά για παράδειγμα τη βασική λειτουργία της απομνημόνευσης - καταχώρησης στοιχείων στη Μ και μέσα από αυτή ας προσπαθήσουμε να απομυθοποιήσουμε τον τρόπο λειτουργίας ολόκληρου του συστήματος του Υ.
Αλλά ας πάρουμε τα πράγματα από την αρχή.
Πώς φανταζόμαστε τη μνήμη του Υ. Η πρώτη εικόνα που έρχεται στους πιό πολλούς είναι μιας μεγάλης ντουλάπας με ράφια. Από 'κεί ξεκινούν οι εξομοιώσεις. Κάθε ράφι μπορεί να θεωρηθεί σαν θέση μνήμης στην οποία μπαίνει και ένα στοιχείο που βάζουμε στη μνήμη, είτε δεδομένο είναι αυτό (αριθμός, λέξη) είτε εντολή προγράμματος. Η θέση μνήμης λέγεται στη μηχανογραφική-πληροφορική ορολογία και λέξη (word). Οι θέσεις μνήμης αριθμούνται με αύξουσα αρίθμηση από την πρώτη μέχρι την τελευταία. Ο αύξων αριθμός κάθε θέσης μνήμης την προσδιορίζει μέσα στη μνήμη και



λέγεται διεύθυνση (address) της υπόψη θέσης της μνήμης. Οι διευθύνσεις βοηθάνε τον Υ να κρατάει λογαριασμό για το πού τοποθετείται το κάθε τι που εισέρχεται στη μνήμη.
Μια μορφή της μνήμης φαίνεται στο Σχ.2.1 όπου διακρίνεται η μνήμη με τα περιεχόμενά της: οδηγίες (πρόγραμμα) και δεδομένα.

Υπολογιστές και δυαδικό σύστημα. Όποιος έχει ασχοληθεί έστω και λίγο με τους Υ θα έχει ασφαλώς ακούσει τη φράση κλισέ πως ο Υ λειτουργεί στο δυαδικό σύστημα. Από 'κεί και πέρα Υ και Δυαδικό σύστημα πάνε πακέτο και όσο για το τι και το πώς, πίστευε και μη ερεύνα. Εδώ θα προσπαθήσουμε με λίγα λόγια να εξηγήσουμε μερικά πράγματα, προχωρώντας σε ακόμα μεγαλύτερο βάθος την ανάλυση που αρχίσαμε παραπάνω για να φτάσουμε στην ουσία του αρχικού ερωτήματός μας: πώς τέλος πάντων καταχωρείται ένα στοιχείο του έξω κόσμου στο εσωτερικό του Υ.

Στον εγκέφαλο του ανθρώπου υπάρχουν οι στοιχειώδεις μονάδες (νευρόνια κλπ), που καθιστούν δυνατή αυτή την καταχώρηση και τη μετέπειτα επεξεργασία. Εδώ τί υλικά και μηχανισμούς θα χρησιμοποιήσουμε; Το ζητούμενο ήταν μια λύση τεχνικά εφικτή αλλά και απλή και οικονομική, κατάλληλη για μαζική παραγωγή.

Η απάντηση δεν ήταν εύκολη και ήλθε σαν αποτέλεσμα μιας σειράς επιστημονικών-θεωρητικών-μαθηματικών αλλά και τεχνολογικών εξελίξεων. Οταν ήλθε όμως φάνηκε απλή, όπως όλες οι μεγάλες ανακαλύψεις (αφού ανακαλυφθούν), κάτι σαν το αβγό του Κολόμβου. Για να καταλάβουμε καλύτερα τη συλλογιστική που ακολουθήθηκε ας θεωρήσουμε πρώτα την περίπτωση της απομνημόνευσης αριθμών. Αφού κάθε αριθμός αποτελείται από ψηφία είναι ευνόητο να περιμένουμε πως και η θέση μνήμης θα αποτελείται από επί μέρους υπό-θέσεις κάτι σαν μικρές κυψελίδες - θυρίδες σε κάθε μια από τις οποίες θα τοποθετούνταν και ένα ψηφίο του αριθμού μας. Το ερώτημα τώρα είναι από τι υλικό θα πρέπει να είναι φτιαγμένες αυτές οι κυψελίδες ώστε να μπορούν να κρατάνε τα ψηφία που τοποθετούμε. Είναι προφανές πως αφού τα ψηφία των αριθμών μας μπορεί να είναι οτιδήποτε από 0 μέχρι 9 και τα υλικά που θα χρησιμοποιήσουμε θα πρέπει να μπορούν να λαμβάνουν τουλάχιστο 10 διαφορετικές καταστάσεις, ώστε να μπορούμε να αντιστοιχήσουμε έναν αριθμό-ψηφίο από το 0 μέχρι το 9 σε κάθε μία κατάσταση. Υλικά όμως τόσο πολλών καταστάσεων είναι δύσκολο να βρεθούν στη φύση και ακόμα

και στην περίπτωση που βρίσκονταν θα ήταν ακριβά και δύσκολο να αξιοποιηθούν. Δύσκολο λοιπόν στην πράξη να βάλουμε τους αριθμούς μας αυτούσιους μέσα στη μνήμη. Η ανάγκη για χρησιμοποίηση υλικών πολλών καταστάσεων είναι αυτό που το δυσκολεύει. Αφού λοιπόν υπάρχουν δυσκολίες να βρούμε τα υλικά που θέλουμε όπως τα θέλουμε, ας πάμε αντίστροφα και ας δούμε τι μπορεί να μας δώσει η φύση εύκολα και οικονομικά και πώς και αν μπορούμε να αξιοποιήσουμε για το σκοπό που τα θέλουμε. Δεν χρειάζεται μεγάλη προσπάθεια για να διαπιστώσουμε πως στη φύση αφθονούν τα υλικά δύο καταστάσεων (που μπορούν δηλαδή ανά πάσα στιγμή και υπό ορισμένες προϋποθέσεις να βρίσκονται σε μία από δύο δυνατές καταστάσεις). Αυτή η διαπίστωση έδωσε και την πρώτη ιδέα.

Ήταν φανερό πως θα έπρεπε να αξιοποιηθεί η αρχή του διακόπτη που αναγνωρίζει δύο καταστάσεις (ανοικτός - κλειστός, on-off, φως-σκοτάδι). Η κατασκευή λοιπόν κυψελίδων δύο καταστάσεων ή δυαδικών κυψελίδων (μέσα στις θέσεις μνήμης) θα απλοποιούσε τα πράγματα γιατί η πρώτη ύλη υπάρχει όπως είπαμε άφθονη στη φύση και οι δύο καταστάσεις που θα παράγουν οι κυψελίδες αυτές είναι ξεκάθαρες και εύκολο να ανιχνευθούν.

Το δεύτερο θεμελιώδες ερώτημα είναι πώς θα μπορέσουμε να αντιστοιχήσουμε τη φυσική αυτή κατάσταση των διακοπτών με τα στοιχεία του έξω κόσμου (δεδομένα κλπ) που θέλουμε να καταχωρήσουμε. Η απάντηση είναι πάλι απλή : αντιστοιχούμε την μία κατάσταση του διακόπτη, ας πούμε την *αναμμένη* κατάσταση, με τον αριθμό *1* και την άλλη (τη "σβυστή") με το *0*. Δηλαδή από την έννοια του δυαδικού φυσικού στοιχείου οδηγούμαστε στην έννοια του αριθμητικού δυαδικού ψηφίου (0 ή 1). Το δυαδικό αυτό ψηφίο και κατ' επέκταση και το αντίστοιχό του δυαδικό στοιχείο λέγεται διεθνώς bit (από το αγγλικό binary digit)

Ας θεωρήσουμε τώρα πως μία θέση μνήμης αποτελείται από έναν αριθμό τέτοιων διακοπτών. Οι διακόπτες αυτοί μπορεί να είναι "αναμμένοι" ή "σβυστοί". Είναι προφανές πως η αλληλουχία αυτή των αναμμένων και σβυστών διακοπτών αντιστοιχείται με αλληλουχία από *1* και *0* bits, π.χ. 1001110. Αυτή όμως η αλληλουχία ψηφίων δεν είναι τίποτα άλλο παρά ένας αριθμός. Ο αριθμός αυτός λέγεται δυαδικός γιατί είναι εκφρασμένος σε μια αριθμητική γλώσσα (ή αριθμητικό σύστημα κατά τη μαθηματική του έκφραση), που έχει δύο μόνο ψηφία το 0 και το 1 και λέγεται δυαδικό σύστημα. Το δυαδικό σύστημα είναι το σύστημα που *καταλαβαίνει* ο

υπολογιστής, όπως ο άνθρωπος καταλαβαίνει ένα άλλο σύστημα που έχει τα γνωστά μας δέκα ψηφία 0,1,2,...,9 και λέγεται δεκαδικό σύστημα. Και στο δικό μας το σύστημα, το δεκαδικό, ένας αριθμός π.χ. ο 453689 δεν είναι τίποτα άλλο από μια αλληλουχία ψηφίων αποδεκτών από το δεκαδικό σύστημα.

Τα διάφορα αριθμητικά συστήματα (δυαδικό, δεκαδικό κλπ) επικοινωνούν μεταξύ τους δηλαδή ένας αριθμός στο δεκαδικό σύστημα μπορεί να μετατραπεί στον αντίστοιχο και ισοδύναμό του δυαδικό και αντίστροφα. Αυτό βέβαια αποδεικνύεται μαθηματικά (υπάρχει και η σχετική μέθοδος μετατροπής), αλλά μπορεί να προσεγγιστεί και λογικά έτσι ώστε να γίνει κατανοητό ότι ο *δεκαδικός* ανθρώπινος κόσμος και ο *δυαδικός* των υπολογιστών αποτελούν δύο διαφορετικές εκφράσεις της μιας και μοναδικής πραγματικότητας.

Όλοι έχουμε την έννοια του αριθμού που είναι μία και μοναδική, π.χ. όταν ακούμε για 3 πρόβατα δημιουργείται η αντίστοιχη εικόνα στο μυαλό μας. Το 3 τώρα μπορεί (αριθμητικά) να αποδοθεί με άπειρους τρόπους στις διάφορες αριθμητικές γλώσσες ή αριθμητικά συστήματα, έτσι μπορούμε να το πούμε 3 στο δεκαδικό σύστημα, 11 στο δυαδικό κλπ, η έννοια του τρία όμως είναι μία και μοναδική. Μπορούμε μάλιστα με μαθηματικό τρόπο να μεταβούμε από τη μια παράσταση στην άλλη. Όταν έχουμε π.χ. έναν αριθμό στο δεκαδικό σύστημα, μπορούμε να δημιουργήσουμε τον ισοδύναμό του στο δυαδικό και αντίστροφα.

Η μνήμη και το τρανζίστορ - ψηφιακός Υ

Μετά τη θεωρία προχωρούμε να απαντήσουμε σε μερικά τεχνολογικά ερωτήματα. Τί είδους διακόπτες χρησιμοποιούνται στην κατασκευή Υ. Ανάλογα με τις τεχνολογικές εξελίξεις σε κάθε εποχή χρησιμοποιήθηκαν διάφοροι τύποι διακοπτών (ρελαί, δίοδοι λυχνίες). Σήμερα χρησιμοποιούνται τα τρανζίστορς. Το τρανζίστορ είναι ένας κρύσταλλος (πυρίτιο ή silicon) που αν διοχετεύσεις ρεύμα κατά μία φορά το αφήνει να περάσει ενώ αν του το δώσεις από την άλλη φορά το διακόπτει. Βλέπουμε δηλαδή ότι ανάλογα με τη φορά του διοχετευομένου ρεύματος το τρανζίστορ είτε αφήνει το ρεύμα να περάσει (είναι *αναμμένο*) είτε όχι (είναι *σβυστό*). Πράγματι αν *δούμε* μια θέση της μνήμης από πιο κοντά θα διαπιστώσουμε πως δεν

είναι μονοκόμματη αλλά αποτελείται από έναν αριθμό τέτοιων τρανζίστορς από τα οποία άλλα είναι *αναμμένα* και άλλα *σβυστά*.

Το τσιπ

Είδαμε στα παραπάνω τη μνήμη σαν ντουλάπα με ράφια, που όπως είπαμε στη συνέχεια μπορεί να είναι εκατομμύρια. Φανταζόμαστε ασφαλώς χώρους με τερατώδεις διαστάσεις. Από την άλλη μεριά ρίχνουμε μια ματιά στο μικροσκοπικό PC μας και με το δίκιο μας απορούμε. Πώς εξηγείται;

Είπαμε στην αρχή πως η τεχνολογία της μνήμης βασίζεται στην αρχή του διακόπτη και στις ημιαγωγικές ιδιότητες του τρανζίστορ. Με τη βοήθεια του τρανζίστορ κατασκευάζεται το στοιχειώδες κύκλωμα του δυαδικού ψηφίου - bit (δυαδικό ψηφίο 0 ή 1). Το κύκλωμα αυτό εκτός από τρανζίστορ αποτελείται από πυκνωτές και αντιστάσεις.

Μία μνήμη, όπως είναι προφανές, αποτελείται από ένα σύνολο τέτοιων στοιχειωδών κυκλωμάτων. Παλιά τα στοιχεία αυτά καταλάμβαναν μεγάλο χώρο και κατ' ακολουθία οι Υ ήταν ογκώδεις και ακριβοί. Οι εξελίξεις όμως της τεχνολογίας τα τελευταία χρόνια, πρώτα με τα τυπωμένα κυκλώματα και στη συνέχεια με τα κυκλώματα υψηλής πυκνότητας (μεγάλος αριθμός κυκλωμάτων συμπυκνωμένων σε μικρό χώρο) είχαν σαν αποτέλεσμα να μικρύνει δραματικά ο όγκος και το κόστος των Υ και να αυξηθεί η χωρητικότητά τους.
Εκείνο που εκμεταλλεύτηκε η τεχνολογία κατασκευής των Υ ήταν οι ιδιότητες των ημιαγωγών και οι τεχνικές *επέμβασης* στη μοριακή δομή τους. Αυτό που κατάφεραν ήταν να χρησιμοποιήσουν ένα τέτοιο κρύσταλλο και συγκεκριμένα το silicon (υψηλή καθαρότητα και αφθονία στη φύση) και με την εφαρμογή κατάλληλων φωτοχημικών μεθόδων να δημιουργήσουν τεχνητές ανωμαλίες στη δομή του σε προκαθορισμένα σημεία. Οι τεχνητές αυτές ανωμαλίες έκαναν τα συγκεκριμένα σημεία να *συμπεριφέρονται* κατά περίπτωση σαν τρανζίστορς, πυκνωτές ή αντιστάσεις. Τα *στοιχεία* αυτά κατάλληλα συνδεόμενα μεταξύ τους με αυλάκια αλουμινίου αποτελούσαν τα ισοδύναμα των δυαδικών κυκλωμάτων της μνήμης. Για να γίνουμε λίγο πιο συγκεκριμένοι, η μεθοδολογία που ακολουθούσαν ήταν να παίρνουν τεμάχια silicon 10 cm μήκος περίπου και 0.5 mm πάχος να *εγχαράζουν* χιλιάδες πανομοιότυπα κυκλώματα μνήμης ή ΜΕ ή ΑΜ και μετά να το κόβουν σε μικρά τετράγωνα τεμαχίδια 3mm το καθένα. Αυτά τα

μικρά τεμαχίδια ονομάστηκαν microchips ή απλά chips και το καθένα είχε μερικές χιλιάδες τέτοια κυκλώματα. Ανάλογα με το είδος των κυκλωμάτων τα τσιπς διακρίνονταν σε τσιπς μνήμης, τσιπς ΜΕ, τσιπς ΑΜ κλπ.

Η Κάρτα

Πολλές φορές χρειάζεται να συνδέσουμε μερικά τσιπς μεταξύ τους για να κάνουν μια πιο σύνθετη εργασία. Τέτοια τσιπς συγκεντρωμένα σε μία πλακέτα που κάνουν μια συγκεκριμένη εργασία συχνά αποδίδεται και σαν κάρτα. Στον Υ έχουμε πολλά παραδείγματα τέτοιων καρτών, όπως θα δούμε και στη συνέχεια του βιβλίου. Π.χ. τσιπς μνήμης και το τσιπ του μΕ μαζί με άλλα συναφή αποτελούν τη βασική κάρτα του Υ που συχνά αποδίδεται με τον όρο μητρική κάρτα motherboard. Αλλες κάρτες είναι οι κάρτες ελέγχου (controllers) και προσαρμογής (adapters) των διαφόρων περιφερειακών (πχ. η κάρτα οθόνης), κλπ.

ΑΝΑΛΥΣΗ 2.1

(Μπορεί να παραλειφθεί σε πρώτη ανάγνωση)

Αριθμητικά συστήματα

Οι αριθμητικές γλώσσες στα μαθηματικά λέγονται αριθμητικά συστήματα και ορίζονται από το πλήθος των βασικών ψηφίων που αναγνωρίζουν και το οποίο λέγεται βάση του αριθμητικού συστήματος. Έτσι έχουμε το γνωστό μας δεκαδικό σύστημα που αναγνωρίζει τα γνωστά μας δέκα ψηφία 0,1,2,...,9 και έχει βάση το 10. Με τον ίδιο τρόπο μπορούμε να ορίσουμε το οκταδικό σύστημα σαν ένα σύστημα με βάση το 8 και οκτώ ψηφία τα 0,1,...,7, το δυαδικό σύστημα με βάση το 2 και δύο ψηφία το 0,1 κλπ.
Ενας αριθμός εκφρασμένος σε ένα σύστημα δεν είναι τίποτα άλλο από μια αλληλουχία ψηφίων αποδεκτών από το σύστημα αυτό. Για παράδειγμα ο αριθμός 45389 μπορεί να θεωρηθεί σαν ένας αριθμός στο δεκαδικό σύστημα, γιατί όλα τα ψηφία του είναι αποδεκτά από το δεκαδικό σύστημα, δεν μπορεί όμως να θεωρηθεί οκταδικός αριθμός γιατί τα ψηφία του 8 και 9 δεν αναγνωρίζονται από το οκταδικό σύστημα. Με τον ίδιο τρόπο μια αλληλουχία δυαδικών ψηφίων πχ. 1011 λέγεται δυαδικός αριθμός.

Επικοινωνία μεταξύ συστημάτων

(α) Μετατροπή από το Δυαδικό στο Δεκαδικό

τάξη ψηφίων
(3) (2) (1) (0)
1 1 0 1 Βάση : 2

1101 =
$1x2^3 + 1x2^2 + 0x2^1 + 1x2^0 =$
8+4+0+1 = 13

Κάθε δυαδικός αριθμός μπορεί να μετατραπεί στον αντίστοιχό του δεκαδικό, αν πολλαπλασιάσουμε κάθε ψηφίο του με τη βάση του δυαδικού συστήματος (2) υψωμένη στην τάξη του ψηφίου (τη σειρά του) μέσα στον αριθμό και προσθέσουμε τα γινόμενα. Η τάξη του ψηφίου βρίσκεται αν αριθμήσουμε τα ψηφία από το τέλος του αριθμού προς την αρχή αρχίζοντας από το 0.

(β) Μετατροπή από το Δεκαδικό στο Δυαδικό

13|2
1 6|2
0 3|2
1 1|2
1 0

μας δίνει τον 1101

Για τη μετατροπή από δεκαδικό σε δυαδικό αριθμό η μέθοδος συνίσταται σε διαδοχικές διαιρέσεις του "δεκαδικού" αριθμού δια του 2 μέχρι να φτάσουμε σε μηδενικό πηλίκο. Τα υπόλοιπα των διαιρέσεων (που θα είναι 0 ή 1) λαμβανόμενα από το τελευταίο προς το πρώτο δίνουν τον αντίστοιχο δυαδικό αριθμό.

Αρίθμηση και πράξεις στο Δυαδικό Σύστημα

Ισχύουν οι ίδιοι νόμοι όπως στο δεκαδικό σύστημα. Αν προσέξουμε την αρίθμηση στο δεκαδικό 0,1,2,...9,10,11,...99,100 κοκ. θα δούμε πως πηγαίνουμε σε διψήφιο αριθμό όταν ο μονοψήφιος έχει εξαντλήσει τα όριά του (9). Σε τριψήφιο όταν τα ψηφία του διψήφιου γίνονται φθάνουν τα όριά τους (99) κλπ.
Ετσι και στο δυαδικό αριθμούμε: 0,1, 10,11, 100,101,110,111, 1000 κοκ. Αν κάνουμε τις μετατροπές θα δούμε την αντίστοιχη σειρά στο δεκαδικό: 0,1, 2,3, 4,5,6,7, 8 κοκ

ΑΝΑΛΥΣΗ 2.1 (συνέχεια)

Πόσες τιμές μπορεί να πάρει ένας δυαδικός αριθμός ν ψηφίων

Ας δούμε πρώτα τι συμβαίνει με τους δεκαδικούς αριθμούς. Πόσες διαφορετικές τιμές μπορεί να πάρει ένας διψήφιος δεκαδικός αριθμός; Η απάντηση είναι φυσικά 100. Γιατί; Γιατί η μικρότερη τιμή είναι 00 και η μεγαλύτερη 99. Άρα οι διαφορετικές τιμές είναι 0, 1, 2, 3,...., 99 δηλ 100 τιμές.

Η μεγαλύτερη τιμή ενός πολυψήφιου αριθμού σε οποιοδήποτε σύστημα είναι εκείνη που προκύπτει όταν κάθε ψηφίο πάρει τη μεγαλύτερη τιμή του στο αντίστοιχο σύστημα. Η τιμή αυτή είναι 9 στο δεκαδικό σύστημα, 1 στο δυαδικό κλπ.

Πόσες τιμές για παράδειγμα μπορεί να πάρει ένας διψήφιος δυαδικός αριθμός (δύο bits); Ο μικρότερος φυσικά είναι 00 (0 στο δεκαδικό) και ο μεγαλύτερος 11 (3 στο δεκαδικό) άρα οι διαφορετικές τιμές είναι τέσσερις οι: 00, 01 (1), 10 (2), 11 (3), άλλο περιθώριο δεν υπάρχει μια και το καθένα από τα δύο δυαδικά ψηφία εξάντλησε τις δυνατότητες αύξησής του (δηλ το 1). Επομένως τα δύο bits μπορούν να πάρουν τέσσερις τιμές.

Τώρα ας δούμε πώς μπορούμε να βρίσκουμε εύκολα το πλήθος αυτών των τιμών που μπορεί να πάρει ένας ν-ψήφιος αριθμός σε οποιοδήποτε αριθμητικό σύστημα.
Αν προσέξουμε από τα παραπάνω παραδείγματα ο διψήφιος δεκαδικός αριθμός μπορεί να πάρει 100 τιμές ή 10^2. Ο διψήφιος δυαδικός 4 τιμές ή 2^2. Και στις δύο περιπτώσεις παρατηρούμε πως οι τιμές προκύπτουν από έναν εκθετικό αριθμό που έχει βάση τη βάση του αριθμητικού συστήματος του υπόψη αριθμού και εκθέτη το πλήθος των ψηφίων του.

Στο δυαδικό σύστημα λοιπόν ένας ν-ψήφιος αριθμός δηλ αριθμός με ν bits "χωράει" 2^v τιμές. Ετσι έχουμε πχ ότι τα 10 bits μπορούν να χωρέσουν (να μετρήσουν) 2^{10} ή 1024 αριθμούς (από το 0 μέχρι το 1023) και τα 8 bits 2^8 ή 256 αριθμούς.

Οι πράξεις στο δυαδικό σύστημα

Στο δυαδικό σύστημα έχουμε τις τέσσερις πράξεις όπως και στο δεκαδικό. Οι πράξεις αυτές είναι η πρόσθεση, η αφαίρεση, ο πολλαπλασιασμός και η διαίρεση.

Στην πρόσθεση δυαδικών αριθμών ισχύουν οι παρακάτω κανόνες:

1+0=1
0+1=1
0+0=0
1+1=0 και κρατούμενο 1

Στην αφαίρεση δυαδικών αριθμών ισχύουν οι παρακάτω κανόνες:

1-0=1
1-1=0
0-0=0
0-1=1 και κρατούμενο 1

Στον πολλαπλασιασμό και την διαίρεση αφού κάνουμε κανονικά τις πράξεις εφαρμόζουμε τους παραπάνω κανόνες της πρόσθεσης και αφαίρεσης.

3 ΠΛΗΡΟΦΟΡΙΑ ΚΑΙ ΥΠΟΛΟΓΙΣΤΗΣ

Πώς καταχωρείται η πληροφορία στον Υπολογιστή. Η έννοια της κωδικοποίησης.

Είδαμε στο προηγούμενο κεφάλαιο πως όλες οι πληροφορίες που θα επεξεργασθεί ο Υ θα πρέπει να τοποθετηθούν στη Μνήμη και μάλιστα υπό αριθμητική - δυαδική μορφή. Αυτό σημαίνει πως όλες οι πληροφορίες για να εισαχθούν στη μνήμη και επεξεργαστούν από έναν τέτοιο Υ θα πρέπει πρώτα να μετατραπούν σε δυαδικούς αριθμούς.

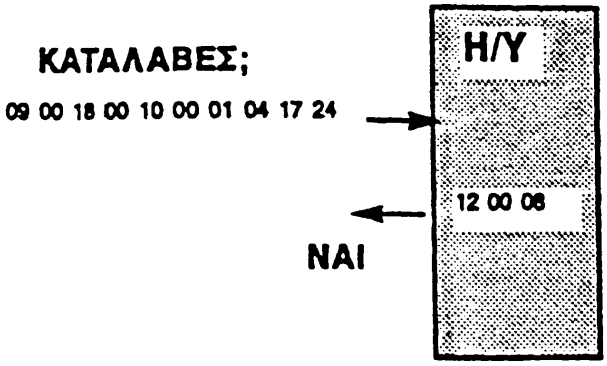
Ψηφιακές και αναλογικές πληροφορίες

Οι πληροφορίες υπάρχουν σε διάφορες μορφές, αριθμοί, κείμενα, εικόνες, ήχοι κλπ, μπορούμε όμως να τις κατατάξουμε σε δύο βασικές κατηγορίες: τη διακριτή ή ψηφιακή και τη συνεχή ή αναλογική.

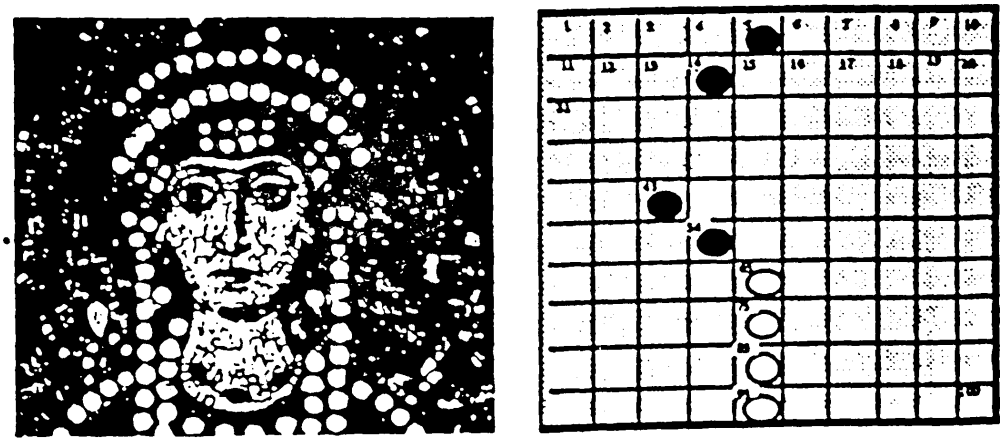
Οι ψηφιακές πληροφορίες αποτελούνται από ένα σύνολο διακριτών (αυτόνομων) στοιχειωδών πληροφοριών π.χ. οι αριθμοί αποτελούνται από σύνολο ξεχωριστών ψηφίων, τα κείμενα από σύνολο χαρακτήρων κλπ.

Οι αναλογικές πληροφορίες αποτελούνται από ένα σύνολο άπειρων θεωρητικά στοιχειωδών πληροφοριών τα όρια των οποίων δεν είναι διακριτά αλλά συγχέονται με συνεχή, αναλογικό τρόπο δημιουργώντας ένα σύνθετο αποτέλεσμα π.χ. μια εικόνα αποτελείται από άπειρα χρώματα και αποχρώσεις τα όρια των οποίων είναι δυσδιάκριτα, ο ήχος (φωνή, μουσική) από άπειρες τιμές έντασης, συχνότητας κλπ.

Α	00	Ξ	13
Β	01	Ο	14
Γ	02	Π	15
Δ	03	Ρ	16
Ε	04	Σ	17
Ζ	05	Τ	18
Η	06	Υ	19
Θ	07	Φ	20
Ι	08	Χ	21
Κ	09	Ψ	22
Λ	10	Ω	23
Μ	11	;	24
Ν	12	...	



Σχ. 3.1Κωδικοποίηση Χαρακτήρων



Σχ. 3.2 Κωδικοποίηση Ψηφιδωτού

Πως οι πληροφορίες εισέρχονται στον Υπολογιστή - Κωδικοποίηση

Όπως είπαμε και παραπάνω για την εισαγωγή μιας πληροφορίας στον Υ απαιτείται η μετατροπή της σε δυαδική ψηφιακή αριθμητική μορφή: δυαδικοποίηση.
Αφού όμως από το δεκαδικό μας σύστημα μπορούμε να πάμε εύκολα στο δυαδικό (όπως είδαμε στο προηγούμενο κεφάλαιο) τότε εκείνο που αρκεί να καταφέρουμε για να μπορέσουμε να εισαγάγουμε μία πληροφορία στον Υ είναι να τη μετατρέψουμε σε αριθμό του δεκαδικού μας συστήματος. Η αριθμοποίηση αυτή της πληροφορίας λέγεται κωδικοποίηση.

Ας δούμε λοιπόν πώς επιτυγχάνεται η κωδικοποίηση αυτή για τις ψηφιακές και πώς για τις αναλογικές πληροφορίες.

Οι Ψηφιακές Πληροφορίες - Αριθμοί - κείμενα

Αριθμοί. Για τους αριθμούς τα πράγματα είναι απλά. Θέλουμε για παράδειγμα να εισαγάγουμε στη μνήμη ένα αριθμητικό στοιχείο π.χ. τον βασικό μισθό ενός υπαλλήλου που είναι έστω 30000. Πληκτρολογούμε τον αριθμό και αυτόματα αυτός μετατρέπεται στον αντίστοιχό του δυαδικό ο οποίος και *καταχωρείται* σε μία θέση μνήμης (το δυαδικό σήμα 1 *ανάβει* το αντίστοιχο τρανζίστορ ενώ το 0 το *σβύνει*). Προβλήματα ανακύπτουν στη λεπτομέρεια, όταν θέλουμε να καταχωρήσουμε τους αριθμούς με όλες τους τις ιδιότητες (θετικοί, αρνητικοί, πολύ μεγάλοι, πολύ μικροί δηλ δεκαδικοί μεγάλης ακριβείας). Και για αυτές τις περιπτώσεις έχει βρεθεί λύση που μπορεί ο ενδιαφερόμενος αναγνώστης να βρει στα ειδικά βιβλία.

Κείμενο: Ενα κείμενο αποτελείται από χαρακτήρες (αλφαβητικούς, κεφαλαίους και πεζούς, αριθμητικούς, σημεία στίξεως κλπ., ό,τι εν πάση περιπτώσει υπάρχει σε μία γραφομηχανή).
Πώς θα μπορέσουμε να βάλουμε ένα κείμενο στον Υ; Με την κωδικοποίηση. Δηλαδή παίρνουμε το σύνολο των δυνατών χαρακτήρων σε ένα κείμενο (αλφαβητικών, αριθμητικών, σημείων στίξης κλπ) και σε κάθε χαρακτήρα αντιστοιχούμε έναν αριθμό όπως π.χ. φαίνεται στο Σχ. 3.1. Με το τρόπο αυτό ένα κείμενο μπορεί να αποδοθεί σαν μια σειρά από αριθμούς και έτσι να καταχωρηθεί στη Μνήμη.

Πώς γίνεται η κωδικοποίηση. Στην αρχή ξεκαθαρίζουμε πόσοι είναι οι διαφορετικοί χαρακτήρες που μπορούν να χρησιμοποιηθούν σε ένα κείμενο. Τις μέτρησαν οι Αμερικάνοι σύμφωνα με τις δικές τους ανάγκες και θεώρησαν πως με 128 χαρακτήρες καλύπτονται. Τους έδωσαν αριθμούς από το 0 μέχρι το 127 και ξεμπέρδεψαν. Με την εξάπλωση των Υ και την ανάγκη να εξυπηρετηθούν και άλλοι ειδικοί χαρακτήρες αλλά και άλλα αλφάβητα (π.χ. ελληνικό) διατέθηκαν άλλοι τόσοι κωδικοί (από το 128 μέχρι το 255) για την εξυπηρέτησή τους. Ετσι έχουμε αντιστοιχήσει κάθε χαρακτήρα με ένα τριψήφιο δεκαδικό αριθμό από το 0 μέχρι το 255. Οι κωδικοί αυτοί όμως για να μπούν στον Υ πρέπει να μετατραπούν σε δυαδικούς. Οι τριψήφιοι δεκαδικοί 0-255 μετατρέπονται σε 8-ψήφιους δυαδικούς. Αυτή η ομάδα των 8 bits (που κωδικοποιεί δυαδικά ένα χαρακτήρα) έχει επικρατήσει να λέγεται byte.

Πρότυπα κωδικοποίησης. Για να υπάρχει τώρα μια τάξη γενικά αποδεκτή σχετικά με το ποιος κωδικός αντιστοιχεί σε κάθε χαρακτήρα έχουν δημιουργηθεί κατά καιρούς διάφορα συστήματα κωδικοποίησης. Τελικά όμως έχουν επικρατήσει και εφαρμόζονται δύο. Η πρότυπη τυποποίηση ASCII (American Standard Code for Information Interchange) του αμερικάνικου κέντρου τυποποίησης και η κωδικοποίηση EBCDIC (Extended Binary Coded Decimal Interchange Code) της IBM. Ο κώδικας ASCII περιέχει 94 γραφικούς χαρακτήρες που τυπώνονται και άλλους 34 χαρακτήρες που χρησιμοποιούνται για διάφορες λειτουργίες ελέγχου. Οι κωδικοποιήσεις αυτές καθορίζουν αυτές τις αντιστοιχίες π.χ. στην ASCII ο χαρακτήρας C αντιστοιχεί στον αριθμό 67, το *= * στο 61, το λατινικό γράμμα A με 1000001, κλπ.
Οι τύποι χαρακτήρων ελέγχου είναι τρεις:
οι διαμορφωτές μορφής,
οι διαχωριστές πληροφορίας και
οι χαρακτήρες ελέγχου-επικοινωνίας.

Οι Αναλογικές Πληροφορίες - Η έννοια της Ψηφιοποίησης

Και μέχρι εδώ καλά. Η κωδικοποίηση των ψηφιακών πληροφοριών είναι εύκολη υπόθεση. Μήπως όμως αυτό σημαίνει πως ο Υ περιορίζεται μόνο στην επεξεργασία αριθμητικών πληροφοριών; Η πείρα μας από τους Υ όπου βλέπουμε στις οθόνες τους στο σπίτι μας ή στις βιτρίνες των ειδικών

καταστημάτων, εικόνες φωτογραφικής τελειότητας και ακόμα και μουσική στερεοφωνικής πιστότητας αποδεικνύει το αντίθετο.
Πώς επιτυγχάνεται η απεικόνιση στη μνήμη συνεχών πληροφοριών; Μετατρέποντάς τις σε ψηφιακές!
Αναλύουμε δηλ. τη συνεχή πληροφορία σε πεπερασμένο πλήθος διαδοχικών διακριτών καταστάσεων ή με άλλα λόγια την ψηφιοποιούμε. Ένα βυζαντινό ψηφιδωτό είναι χαρακτηριστικό παράδειγμα εφαρμοσμένης ορατής ψηφιοποίησης εικόνας. Η ψηφιοποίηση έχει ήδη έλθει στη ζωή μας και όλοι έχουμε εμπειρίες ψηφιοποιήσεων ακόμα και αν δεν το έχουμε συνειδητοποιήσει π.χ. ψηφιοποίηση του χρόνου (ψηφιακά ρολόγια), του ήχου (CD, ψηφιακά τηλέφωνα).

Η διαδικασία λοιπόν για την εισαγωγή μιας συνεχούς πληροφορίας στον Υ είναι:
α. ψηφιοποίηση σε διακριτές καταστάσεις
β. κωδικοποίηση δηλ. αριθμοποίηση των καταστάσεων αυτών
γ. δυαδικοποίηση δηλ. μετατροπή των αριθμών στο δυαδικό σύστημα

Γενικά μπορούμε να πούμε πως είναι ιδιαίτερα σημαντικό να επιτύχουμε εισαγωγή στον Υ όσο γίνεται πιο πολλών κατηγοριών στοιχείων γιατί από κει και πέρα ανοίγονται τεράστιες δυνατότητες δυναμικής αξιοποίησής τους. Μπορούμε δηλ. μετά να επιδράσουμε επάνω τους με όποιο τρόπο θέλουμε ή με άλλα λόγια να κάνουμε τις επεξεργασίες που θέλουμε χρησιμοποιώντας τις κατάλληλες οδηγίες (προγράμματα). Αν έχουμε για παράδειγμα εικόνες μπορούμε, να τις τροποποιήσουμε να τις περιστρέψουμε, να τις μεγεθύνουμε, να τις μικρύνουμε να απομονώσουμε ορισμένα σημεία για περαιτέρω εξέταση κλπ. Καταλαβαίνουμε πόση σημασία έχει κάτι τέτοιο σε μερικές εφαρμογές όπως είναι η Ιατρική και τι μπορούμε να πετύχουμε αν καταχωρήσουμε στον Υ ακτινογραφίες , εικόνες από το ηλεκτρονικό μικροσκόπιο κλπ.

Παράδειγμα ψηφιοποίησης - Πώς εισάγουμε εικόνα στον Υ

Χαρακτηριστικό παράδειγμα ψηφιοποίησης συνεχούς πληροφορίας (εικόνας) είναι τα γνωστά μας ψηφιδωτά δηλ. εικόνες που αποτελούνται από πεπερασμένο σύνολο διάκριτων στοιχείων - ψηφίδων.
Και πώς καταχωρούμε ένα τέτοιο ψηφιδωτό στη Μνήμη; Με κατάλληλη κωδικοποίηση των στοιχείων (ψηφίδων) που απαρτίζουν την εικόνα. Πώς;

Αν θεωρήσουμε πως μια τέτοια εικόνα αποδίδεται από ένα πλαίσιο με αριθμημένα τετραγωνάκια μεγέθους ψηφίδας μπορούμε να παραστήσουμε κάθε ψηφίδα με τον αριθμό στο τετραγωνάκι που βρίσκεται (Σχ. 3.2). Κάνουμε δηλ. αυτό που ονομάσαμε κωδικοποίηση. Με τον τρόπο αυτό παριστάνουμε ολόκληρη την εικόνα σαν μια σειρά αριθμών.
Για ένα μαυρόασπρο μάλιστα ψηφιδωτό που έχουμε ψηφίδες άσπρες και μαύρες θα πρέπει μαζί με την κωδικοποιημένη θέση της κάθε ψηφίδας να προσδιορίσουμε και το χρώμα της ψηφίδας δηλαδή αν πρόκειται για άσπρη ή μαύρη ψηφίδα. Και αυτό μπορεί εύκολα να κωδικοποιηθεί αν πούμε την άσπρη ψηφίδα σαν 0 και τη μαύρη σαν 1, οπότε κάθε ψηφίδα θα παρίσταται σαν δύο αριθμοί: τη θέση (π.χ. 0 - 100) και το χρώμα (0 ή 1). Για έγχρωμες ψηφίδες πρέπει να κωδικοποιήσουμε και τα χρώματα.

Αφού καταλάβαμε αυτό προχωράμε. Καλά με τα ψηφιδωτά, αλλά εδώ βλέπουμε εικόνες φωτογραφικής τελειότητας. Από κεί και πέρα είναι θέμα κλίμακας. Αν μειώσουμε το μέγεθος των ψηφίδων αυξάνοντας αντίστοιχα τον αριθμό τους θα έχουμε όλο και πιο τέλεια εικόνα. Και αν το μέγεθος της ψηφίδας γίνει κουκίδα και οι αποστάσεις μεταξύ τους μικρότερη από αυτή που μπορεί να ξεχωρίσει το μάτι μας, τότε έχουμε την ψευδαίσθηση *συνεχούς* φωτογραφικής εικόνας. Αυτό βέβαια σημαίνει πολύ περισσότερα σημεία, μεγαλύτερη μνήμη ταχύτητα απεικόνισης κλπ, που αυξάνουν το κόστος.

Και καλά για τις φωτογραφίες, τις βλέπετε στις οθόνες των Υ. Γνωρίζετε όμως πως εκτός από την απεικόνιση εικόνων έχουν αρχίσει να κυκλοφορούν και φωτογραφικές μηχανές που βγάζουν φωτογραφίες χωρίς φιλμ; Αντί φίλμ έχουν μνήμη. Τραβάτε τη φωτογραφία, και αυτή αυτόματα ψηφιοποιείται και κωδικοποιείται σε νούμερα στη μνήμη της φωτογραφικής μηχανής. Στη συνέχεια χωρίς χημικά και εμφανίσεις μπορεί να εμφανιστεί στην οθόνη ενός Υ ή να τυπωθεί κατάλληλα σε εκτυπωτή! Το πρόβλημα είναι η τιμή. Αν δεν διαθέτετε τα χρήματα για να την αγοράσετε τώρα, τότε αρκεί να έχετε λίγη υπομονή και να περιμένετε. Τις φωτογραφικές αυτές μηχανές τις χρησιμοποιούμε και για να στέλνουμε φωτογραφίες μέσω Internet.

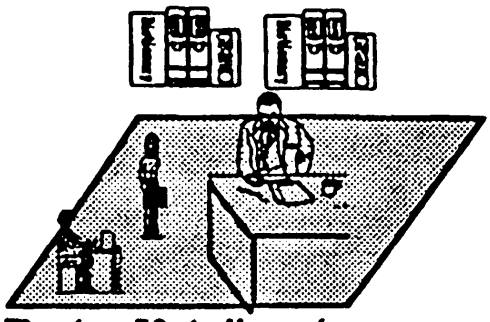
4 Ο ΥΠΟΛΟΓΙΣΤΗΣ ΚΑΙ ΟΙ ΕΦΑΡΜΟΓΕΣ ΤΟΥ

Παρόλες τις υπεραπλουστεύσεις του το πρώτο κεφάλαιο μας βοήθησε να απομυθοποιήσουμε τον Υ και να αναγνωρίσουμε τις δύο βασικές έννοιες της πληροφορικής:
- το μηχάνημα δηλ. τον Υπολογιστή (Υ) και
- τη χρήση του (εφαρμογές)

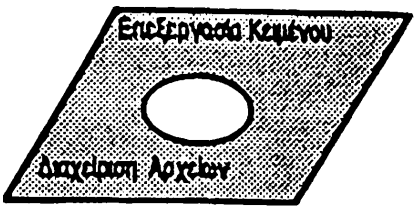
Η σχέση του Υ και της χρήσης του είναι ανάλογη με την πιο οικεία σε μας σχέση ενός κτιρίου και της χρήσης για την οποία προορίζεται (σπίτι, γραφείο, βιομηχανία, νοσοκομείο, ξενοδοχείο κλπ). Είναι προφανές πως δεν αρκεί να υπάρχει απλά το κτίριο για να μπορεί και να λειτουργήσει. Χρειάζονται υπηρεσίες σχετικές με τη χρήση. Οι υπηρεσίες αυτές θα παρέχονται από τους κατάλληλους ανθρώπους και θα είναι δύο ειδών:
- Γενικές υπηρεσίες (καθαριότητας, συντήρησης κλπ)
- Ειδικές υπηρεσίες, σχετικές με τη χρήση του π.χ. λογιστικές, νοσοκομειακές, ξενοδοχειακές κλπ.

Τα τρία επίπεδα εμφανίζονται στο σχήμα 4.1α. Κάθε επίπεδο δίνει το απαραίτητο υπόβαθρο στο αμέσως επόμενο να λειτουργήσει. Αν παρατηρήσουμε κάπως καλύτερα, μπορούμε να πούμε πως το κτίριο και οι γενικές λειτουργικές υπηρεσίες του, παρέχουν την υποδομή για να λειτουργήσουν οι κύριες χρηστικές υπηρεσίες. Αυτό σημαίνει πως μπορούμε να ομαδοποιήσουμε τα δύο πρώτα επίπεδα και να τα θεωρήσουμε σαν το πλήρες σύστημα που διατίθεται για χρήση.

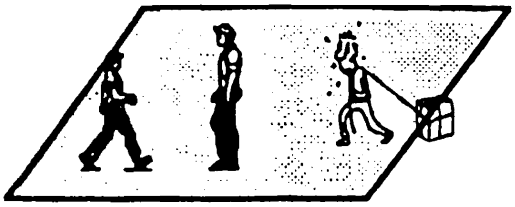
Στο χώρο της πληροφορικής έχουμε αντίστοιχα:
- στη θέση του κτιρίου, τον Υπολογιστή



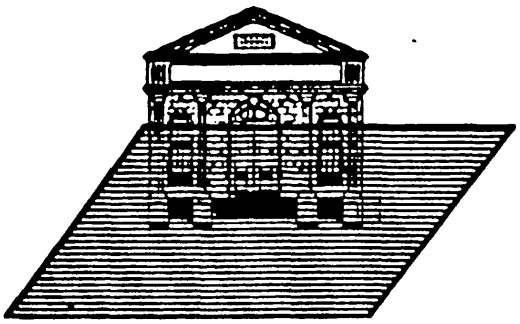
Πλατφόρμα Ειδικών Υπηρεσιών
(νοσοκομειακών, ξενοδοχειακών κλπ. ανάλογα με τη χρήση του κτιρίου)



Πλατφόρμα Εφαρμογών - Πακέτων



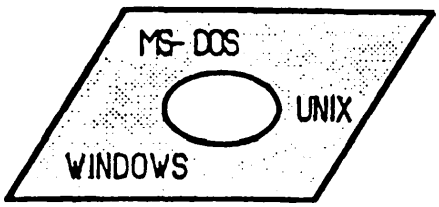
Πλατφόρμα Γενικών Υπηρεσιών
(συντήρηση κλπ)



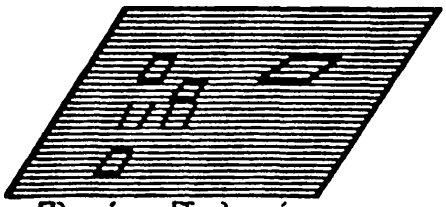
Πλατφόρμα Κυρίου

(α)
Τα τρία επίπεδα στα οποία αναλύεται η λειτουργία ενός οργανισμού - επιχείρησης.

Μέσα στο πλαίσιο εμφανίζεται το "λειτουργικό" κτίριο που διατίθεται στους χρήστες για τις εργασίες τους



Πλατφόρμα Λειτουργικού Συστήματος



Πλατφόρμα Εξοπλισμού

(β)
Τα τρία επίπεδα στα οποία αναλύεται η λειτουργία ενός μηχανογραφικού συστήματος

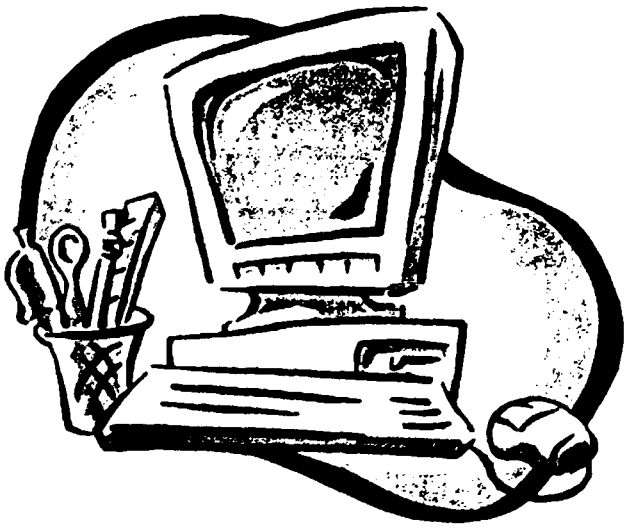
Μέσα στο πλαίσιο εμφανίζεται το Σύστημα του Υπολογιστή που διατίθεται στους χρήστες για τις εφαρμογές τους

Σχ.4.1 Επίπεδα Συστήματος πληροφορικής
Αντιστοίχιση με επιχείρηση

- στη θέση των υπηρεσιών, τα προγράμματα, τα οποία κατ' αντιστοιχία με τα παραπάνω διακρίνονται σε
    - γενικά προγράμματα ελέγχου του εξοπλισμού, εξυπηρέτησης χρηστών κλπ που ονομάζονται Λειτουργικό Σύστημα (ΛΣ)
    - ειδικά προγράμματα σχετικά με την επιθυμητή χρήση του Υ που ονομάζονται εφαρμογές

Έτσι έχουμε κατ' αντιστοιχία την αλληλουχία των επιπέδων του σχ. 4.1β. Πάλι κατ' αντιστοιχία με τα παραπάνω, συχνά θεωρούμε τα δύο πρώτα επίπεδα εξοπλισμό και ΛΣ αναπόσπαστο κομμάτι απαραίτητο για την ανάπτυξη εφαρμογών και το θεωρούμε το σύστημα του Υ το οποίο διατίθεται στους χρήστες για να αναπτύξουν τις εφαρμογές τους δηλ να γράψουν τα ειδικά προγράμματα για τη χρήση που τους ενδιαφέρει.

Η ανωτερότητα του Υ σε σχέση με το παράδειγμα της επιχείρησης είναι η ευελιξία του. Συνήθως τα κτίρια φτιάχνονται για τη χρήση που προορίζονται και παρουσιάζουν μια σχετική ανελαστικότητα στην αλλαγή της χρήσης τους. Με τους Υ τα πράγματα είναι πιο ευέλικτα. Φτιάχνεις ένα γενικό μηχάνημα, το εφοδιάζεις με το κατάλληλο ΛΣ και στη συνέχεια προσδιορίζεις τη χρήση του γράφοντας τα κατάλληλα προγράμματα εφαρμογών. Αν θέλεις να αλλάξεις τη χρήση του απλά αλλάζεις τα προγράμματα εφαρμογών. Ετσι το ίδιο μηχάνημα μπορεί να είναι γραμματεία, γραφείο δακτυλογραφήσεων, λογιστήριο, γραφείο μισθοδοσίας, και ό,τι άλλο φανταστεί ο γόνιμος νους του χρήστη - προγραμματιστή του.



5 Ο ΥΠΟΛΟΓΙΣΤΗΣ ΚΑΙ Ο ΧΡΗΣΤΗΣ ΤΟΥ

Ολοι λίγο πολύ γνωρίζουμε πως ένα σύστημα PC μοιάζει με το εικονιζόμενο στο Σχ. 5.1. Τα τρία βασικά του μέρη είναι:

(Α) το *βασικό κουτί* όπου γίνονται όλες οι εσωτερικές επεξεργασίες
Είναι ένα κουτί που συγκεντρώνει σε μια βολική μορφή συσκευασίας όλες τις εσωτερικές, υπό την ευρεία έννοια, μονάδες του Υ δηλ μονάδες που δε χρειάζεται ο χρήστης να χειρισθεί άμεσα (δεν έχει άμεση επικοινωνία). Τέτοιες μονάδες είναι η Μνήμη και ο Επεξεργαστής, οι μονάδες Δίσκων κλπ. Ο λόγος της συσκευασίας αυτής είναι πρακτικός δηλ. το σύστημα του PC να αποτελείται από όσο γίνεται πιο λίγα *κομμάτια*.

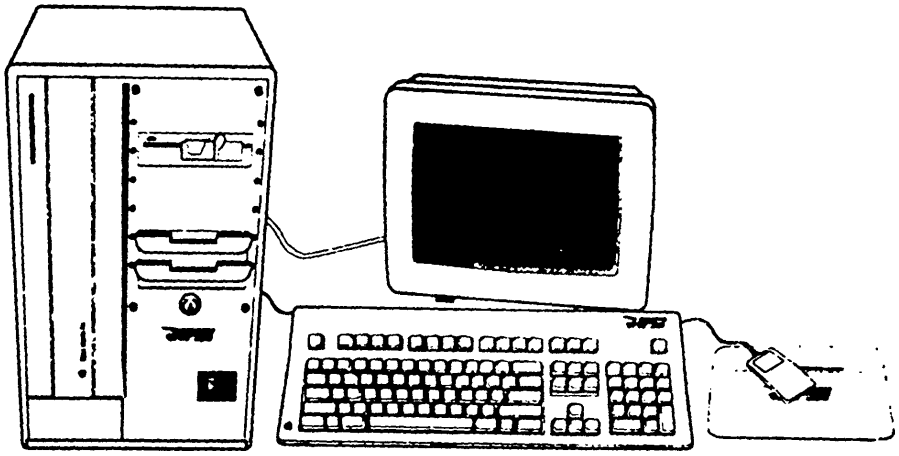
(Γ) το *Πληκτρολόγιο* (Keyboard) για να εισάγουμε στοιχεία

(Β) η *Οθόνη* (Monitor) για να βλέπουμε τα αποτελέσματα

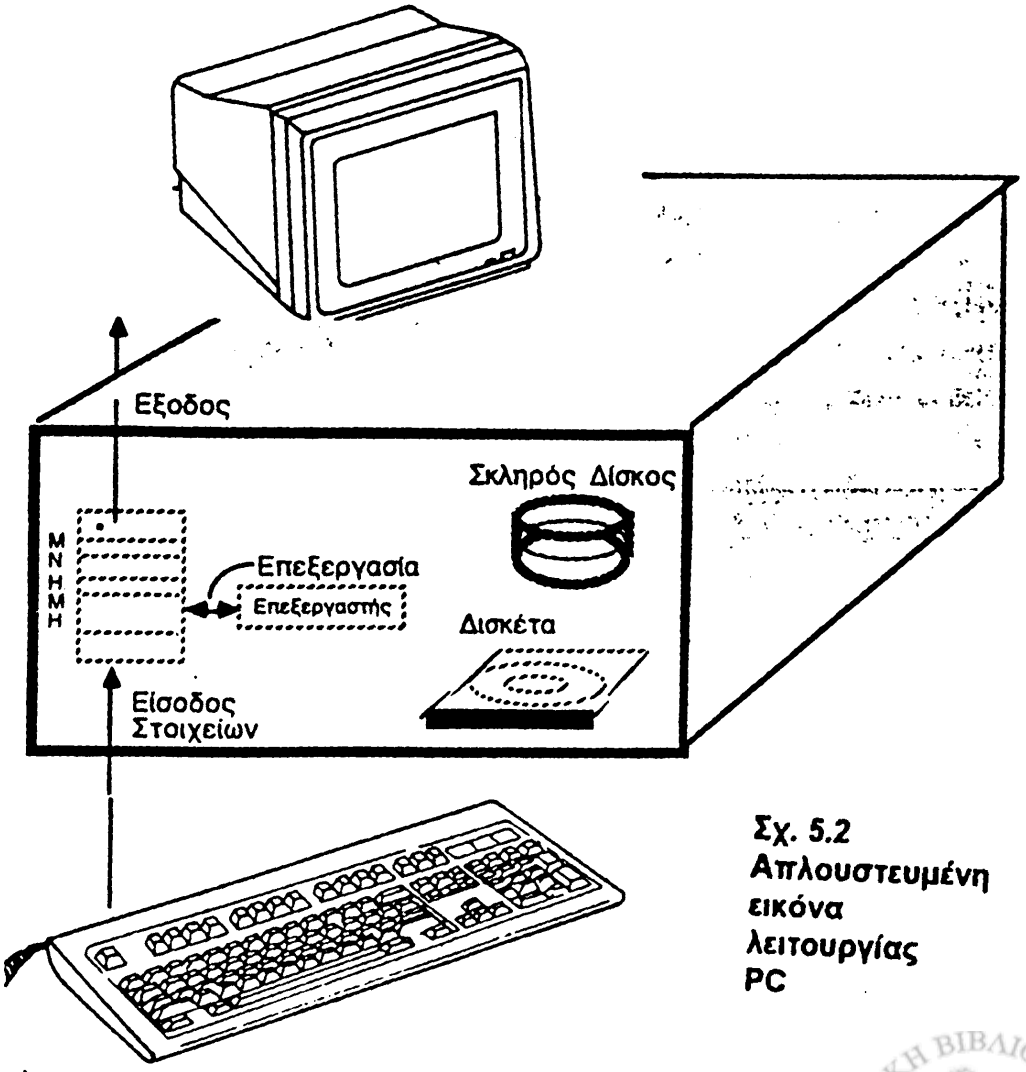
Μια εγκατάσταση σχεδόν πάντα συμπληρώνεται με:

- το *ποντίκι*, που αποτελεί βασικό εργαλείο για τη χρήση των νέων λειτουργικών συστημάτων με γραφικά (WINDOWS) και των εφαρμογών τους, και φυσικά
- τον *εκτυπωτή*

Αν λοιπόν η κεντρική μονάδα ήταν *διάφανη* θα βλέπαμε την εικόνα του Σχ. 5.2.



Σχ. 5.1 Ενα σύστημα PC



Σχ. 5.2
Απλουστευμένη
εικόνα
λειτουργίας
PC

Πώς Λειτουργεί

Ενας στοιχειώδης τρόπος λειτουργίας του Υ είναι:

- Εισαγωγή στοιχείων (π.χ. ένα κείμενο, ένα γράμμα, στοιχεία μισθοδοσίας) από το Πληκτρολόγιο (για στοιχεία που μπαίνουν στον Υ για πρώτη φορά)
- Μεταφορά και Τοποθέτηση στη μνήμη
- Επεξεργασία στοιχείων με συνεχές πήγαιν' έλα μεταξύ μνήμης και επεξεργαστή
- Εξοδος αποτελεσμάτων στην οθόνη ή στον εκτυπωτή.

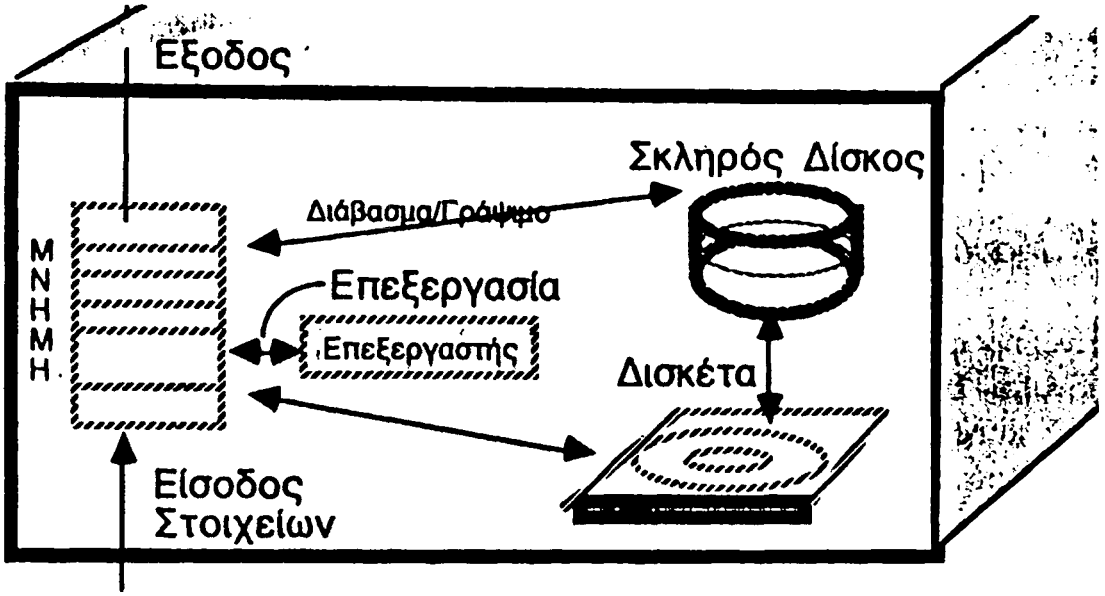
Λαμβάνοντας υπόψη και τους δίσκους η διαδικασία τροποποιείται γιατί τώρα όταν πληκτρολογήσουμε ένα αρχείο στον Υπολογιστή μπορούμε να το βάλουμε στον δίσκο και να το κρατήσουμε. Αν λοιπόν θέλουμε αργότερα να το ξαναχρησιμοποιήσουμε για άλλη επεξεργασία δε χρειάζεται να το ξαναπληκτρολογήσουμε. Το εισάγουμε στη μνήμη από το δίσκο. Η ολοκληρωμένη λοιπόν διαδικασία είναι (σχ. 5.3):

(1) Εισαγωγή στον Υ
(α) από το πληκτρολόγιο (για νέα στοιχεία)
(β) από το δίσκο (για στοιχεία που έχουν εισαχθεί παλαιότερα)
(γ) από δισκέτα, CD
(ε) από scanner

(2) *Μεταφορά και Τοποθέτηση στη μνήμη*

(3) Επεξεργασία στοιχείων (συνεχές πήγαιν' έλα) μεταξύ μνήμης και Επεξεργαστή

(4) Εξοδος στοιχείων
(α) στην οθόνη ή στον εκτυπωτή ή/και
(β) στον δίσκο για διατήρηση και ενδεχόμενη νέα χρήση τους.



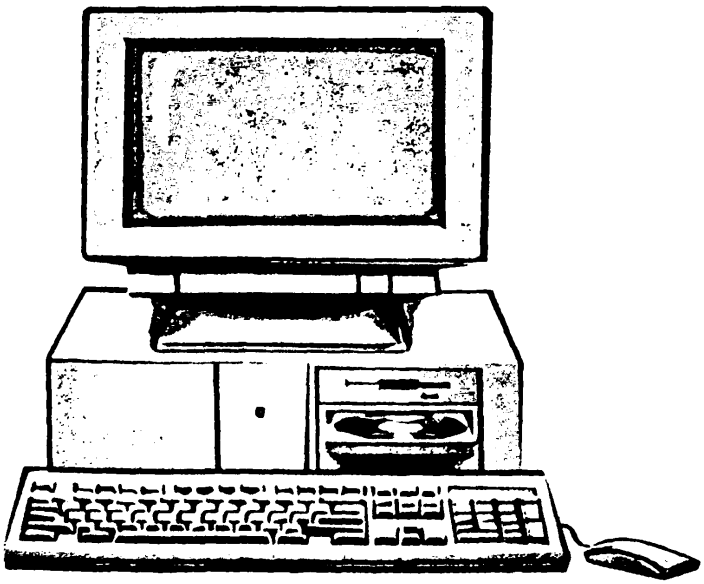
Σχ. 5.3 Ολοκληρωμένη παραστατική εικόνα λειτουργίας PC

Είδη PC

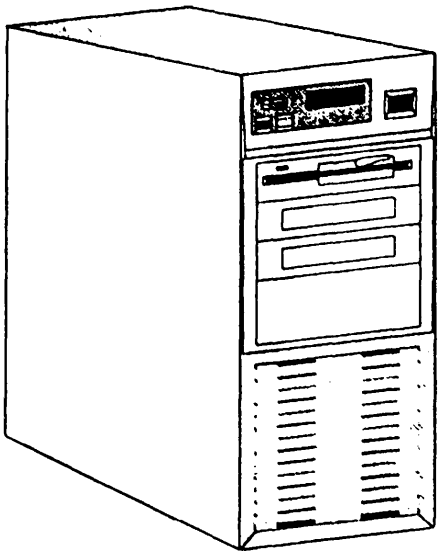
Υπάρχουν πολλές κατατάξεις PC, ανάλογα με την ισχύ τους, τον τύπο του επεξεργαστή τους, τις διαστάσεις τους κλπ
Στο σημείο αυτό και με τις γνώσεις που έχουμε θα αναφερθούμε στη βασική και πιο οφθαλμοφανή κατάταξη σύμφωνα με τις διαστάσεις τους. Τα PC λοιπόν διακρίνονται στα μεγάλα που μπαίνουν πάνω στο τραπέζι και στα μικρά που είναι φορητά.

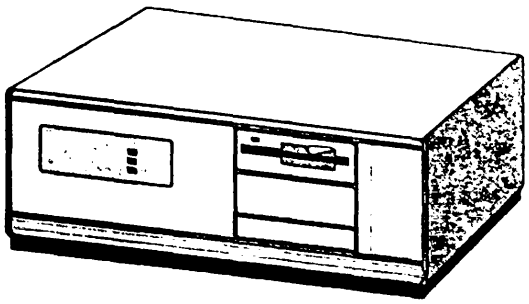
- Τα **Μεγάλα**, χαρακτηρίζονται από το βασικό κουτί τους που είναι στενόμακρο και μπορεί να τοποθετηθεί όρθιο στο πάτωμα ή πάνω στο γραφείο και ονομάζονται TOWER (πύργοι). Ανάλογα με το μέγεθος του κουτιού διακρίνουμε τα μεγάλα TOWER, που χωράνε πιο πολλές εσωτερικές μονάδες και τα μικρότερα.

Παλιότερα υπήρχαν και τα λεγόμενα DESKTOP (επιτραπέζια) στα οποία το κουτί ήταν περίπου τετραγωνικό και συνήθως επάνω του τοποθετούνταν η οθόνη όπως φαίνεται στις παρακάτω εικόνες. (Εικόνα 5.1, 5.2)



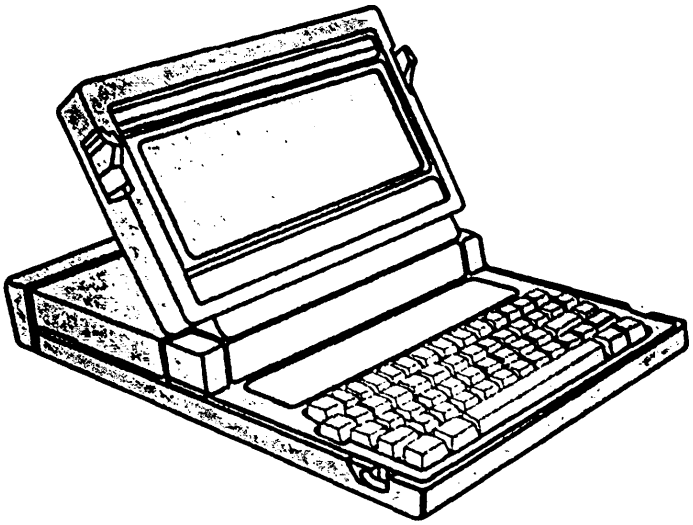
Εικόνα 5.1. (α) Desktop.





(β) Tower

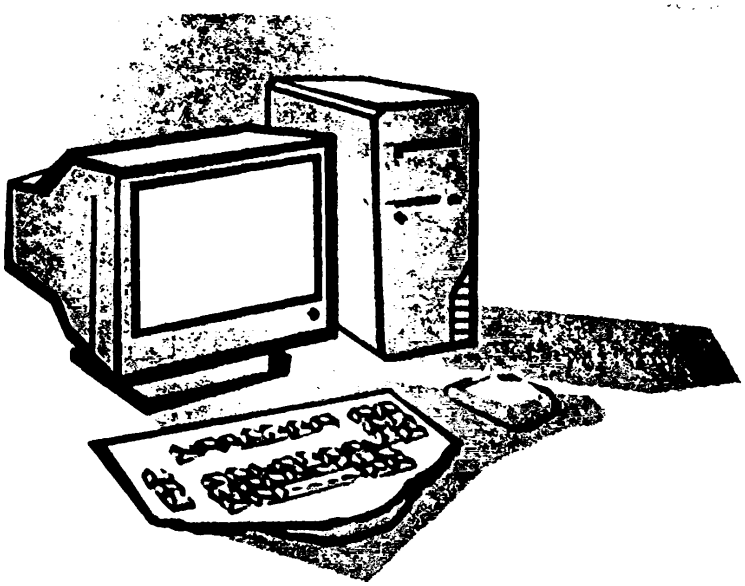
- Τα **Φορητά** (Portable) ή βιβλία (NOTEBOOK), ή "επιγονάτια" (laptop, από το ότι μπορείς να τα δουλέψεις και πάνω στα γόνατα) είναι μικρά, σχήματος βιβλίου, πολύ πετυχημένα και δημοφιλή τα τελευταία χρόνια, ζυγίζουν 3-4 κιλά, δουλεύουν με επαναφορτιζόμενη μπαταρία, συνήθως γύρω στις 3 ώρες, αλλά και με ρεύμα.



Εικόνα 5.2. Φορητός Η/Υ

- Τα **μικρά «τσέπης»** - PDA (Personal Digital Assistant): Νέα γενιά των PC με προοπτική πλατιάς κατανάλωσης. Χαρακτηρίζεται από ριζοσπαστική αντιμετώπιση (πρωτοποριακό Η/W και νέο ΛΣ). Χαρακτηριστικό

παράδειγμα το Newton της Apple που ανακοινώθηκε το καλοκαίρι 1992. Παρά τις ελπίδες που επενδύθηκαν στους Υ αυτούς η μέχρι τώρα εμπορική τους πορεία δεν τις δικαιώνει. Η προσπάθεια όμως συνεχίζεται και νέα μοντέλα από πολλές εταιρείες γιαπωνέζικες κυρίως εμφανίζονται στην αγορά.



ΜΕΡΟΣ ΙΙ

ΤΟ ΣΥΣΤΗΜΑ

ΤΟΥ ΥΠΟΛΟΓΙΣΤΗ

Εξοπλισμός - Προγράμματα

ΠΑΡΑΛΛΗΛΙΑ

6 Η ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΟΥ ΥΠΟΛΟΓΙΣΤΗ

Δομή και διασύνδεση των μονάδων του Υπολογιστή

Στο προηγούμενο κεφάλαιο είδαμε τον Υ όπως φαίνεται απέξω, και πήραμε κάποια γεύση από την τυπική λειτουργία στο εσωτερικό του δηλ. τη διακίνηση των πληροφοριών ανάμεσα στις διάφορες μονάδες του. Πώς γίνεται η διακίνηση αυτή, πώς είναι συνδεδεμένες οι μονάδες μεταξύ τους ώστε να επικοινωνούν και να λέμε ότι ο Υ μας λειτουργεί; Κρίσιμα αλλά και εύλογα ερωτήματα με τα οποία ασχολείται η λεγόμενη αρχιτεκτονική υπολογιστών, θα μας απασχολήσουν στο κεφάλαιο αυτό.

Πρώτες Εννοιες Επικοινωνιών

Το θέμα της επικοινωνίας των πληροφοριών τόσο μέσα στον Υ (μεταξύ εσωτερικών μονάδων), όσο και έξω από τον Υ (σύνδεση μακρινών αποστάσεων, δίκτυα) είναι από τα πιο κρίσιμα στην επιστήμη των Υ.
Εδώ θα ασχοληθούμε κυρίως με την επικοινωνία στο εσωτερικό των Υ. Η επικοινωνία με τον έξω κόσμο έχει αποκτήσει τεράστιο ενδιαφέρον τα τελευταία χρόνια λόγω της ολοένα αυξανόμενης τάσης δικτύωσης των Υ σε παγκόσμιο επίπεδο (INTERNET). Η επκοινωνία αυτή παρουσιάζει τις δικές της ιδιαιτερότητες που απορρέουν από την ανάγκη για το πάντρεμα δύο απαιτητικών επιστημών, της πληροφορικής και των τηλεπικοινωνιών.

Πριν όμως προχωρήσουμε στα βαθιά ας δώσουμε μερικές γενικές έννοιες επικοινωνίας.

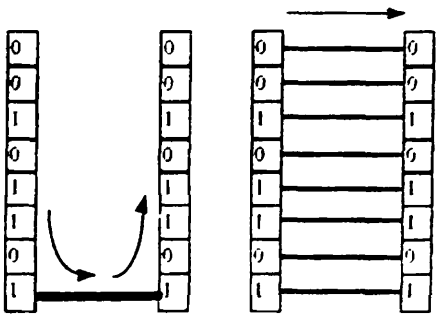
Μετάδοση πληροφορίας. Εχουμε πει πως κάθε μεμονωμένη πληροφορία στον Υ (αριθμός, χαρακτήρας κειμένου κλπ) αποτελείται από bits. Μετάδοση της πληροφορίας από μία μονάδα σε άλλη σημαίνει μετάδοση των δυαδικών της ψηφίων (bits) ένα προς ένα από τη μία μονάδα στην άλλη. Οι πληροφορίες κυκλοφορούν υπό μορφή ομάδων από bits (π.χ. ομάδων των 8 bits δηλ bytes, των 16, 32 bits κλπ).

Δέκτης και Πομπός. Η μονάδα από όπου ξεκινάει η πληροφορία λέγεται πομπός ενώ η μονάδα προορισμού της δέκτης.

Μέσα Επικοινωνίας - Buses και Καλώδια. Για να πραγματοποιηθεί αυτή η κυκλοφορία θα πρέπει να υπάρχει ένας αγωγός (ένα ή περισσότερα καλώδια) που να συνδέει τις αντίστοιχες μονάδες.

Ταχύτητα Επικοινωνίας - bandwidth. Η ταχύτητα μεταφοράς πληροφοριών μέσα σε ένα αγωγό μετριέται όπως είναι φυσικό σε bits per sec ή bps. Κάθε αγωγός χαρακτηρίζεται από τη μεγίστη ταχύτητα bits που επιτρέπει. Συχνά η ταχύτητα ενός αγωγού αναφέρεται και σαν bandwidth.

Σειριακή - Παράλληλη Επικοινωνία. Η μετάδοση πληροφορίας μεταξύ δύο μονάδων σημαίνει μεταφορά των bits της πληροφορίας. Για τη μεταφορά αυτή υπάρχουν δύο τρόποι (βλ σχ. 6.1):



- είτε να υπάρχει ένας αγωγός για κάθε bit, οπότε όταν αρχίσει η μεταφορά όλα τα bits μεταδίδονται ταυτόχρονα και τότε μιλάμε για παράλληλη επικοινωνία,
- είτε να υπάρχει ένας μόνο αγωγός που να συνδέει τις δύο μονάδες, οπότε αναγκαστικά όλα τα bits θα πρέπει να περιμένουν να περάσουν με τη σειρά τους από τον ένα αγωγό και τότε μιλάμε για σειριακή επικοινωνία.

Σχ. 6.1 Τύποι Επικοινωνίας

Είναι προφανές πως η παράλληλη επικοινωνία είναι πιό γρήγορη από τη σειριακή. Είναι όμως και πιο ακριβή γιατί απαιτεί πολλούς αγωγούς (καλώδια). Σε μικρές αποστάσεις αυτό δεν είναι πρόβλημα για το λόγο αυτό στο εσωτερικό του Υ ο κανόνας είναι η παράλληλη επικοινωνία. Εξω όμως από τον Υ για μικρές μεν αποστάσεις μπορεί να έχουμε και παράλληλη σύνδεση (αν και συνήθως προτιμάται η

σειριακή), για μεγάλες όμως αποστάσεις η σειριακή επικοινωνία είναι ο κανόνας.

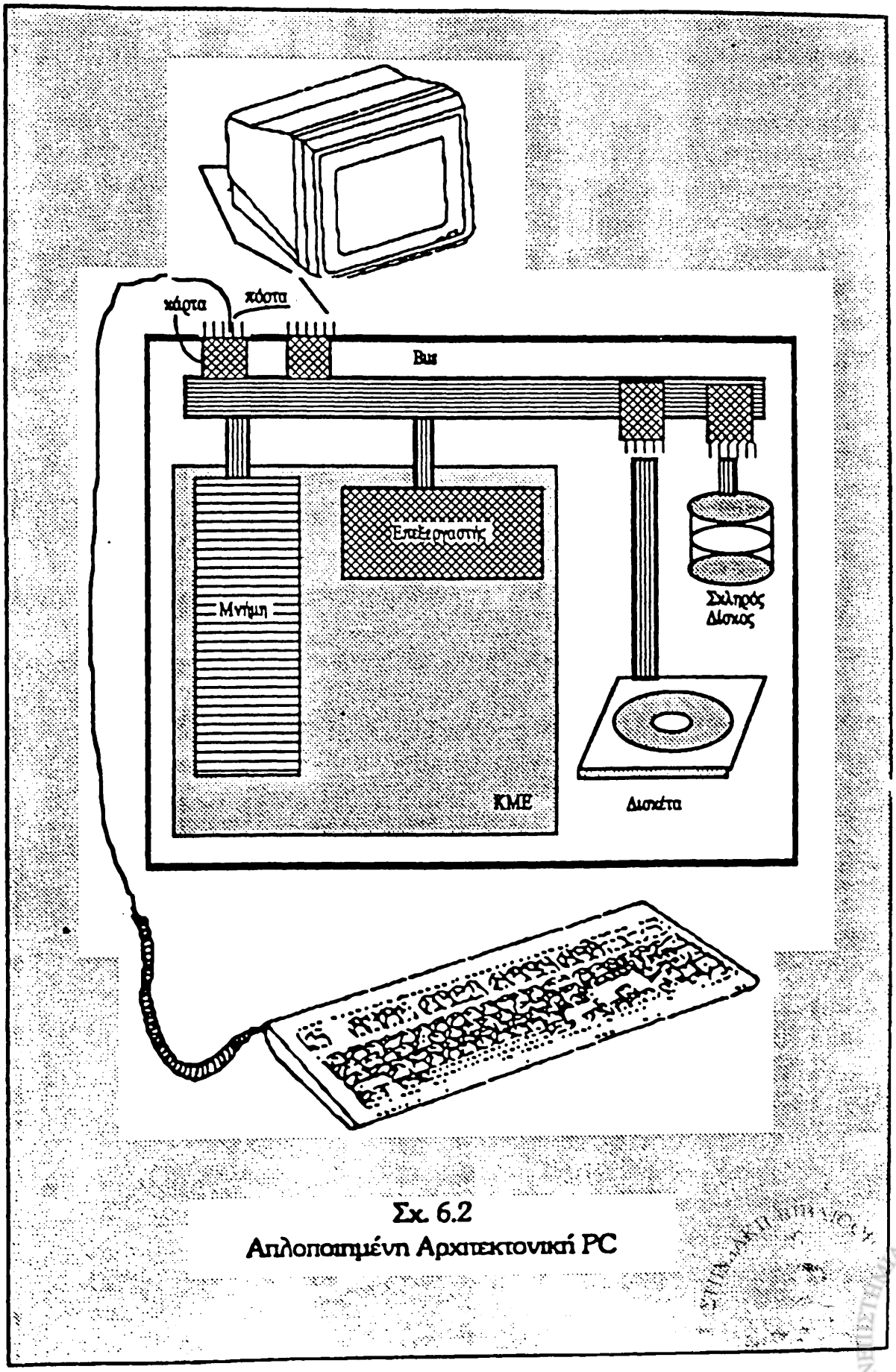
Πρωτόκολλο επικοινωνίας. Για να είναι αξιόπιστη η μετάδοση μιας πληροφορίας (δηλ. να παραληφθεί από το Δέκτη ακριβώς όπως στάλθηκε από τον πομπό) θα πρέπει να υπάρχουν κάποιοι κανόνες που να διέπουν τη μεταφορά. Οι κανόνες αυτοί επιτρέπουν στα μηχανήματα που επικοινωνούν να "συνεννοούνται" για το πότε θα αρχίζει και πότε τελειώνει η μεταξύ τους επικοινωνία, να ελέγχουν τη σωστή μετάδοση σε τακτά διαστήματα κλπ. Οι κανόνες αυτοί για τη περίπτωση της εσωτερικής επικοινωνίας είναι απλοί και υλοποιούνται κυρίως με ηλεκτρονικές διατάξεις. Στην επικοινωνία όμως μακρινών αποστάσεων τα πράγματα δυσκολεύουν, οι κανόνες γίνονται πιό σύνθετοι και η υλοποίησή τους απαιτεί και εξοπλισμό (κάρτες επικοινωνίας) αλλά και προγράμματα που πρέπει να υπάρχουν και στις δύο μονάδες που επικοινωνούν (πομπό και δέκτη).

Η διασύνδεση των μονάδων - Το Κυκλοφοριακό Σύστημα του Υ

Το Bus: Η επικοινωνία μεταξύ των μονάδων στο εσωτερικό του Υ γίνεται παράλληλα όπως είδαμε παραπάνω. Ο αγωγός αυτός επικοινωνίας αποτελείται από πολλά καλώδια και έχει επικρατήσει να ονομάζεται bus (στη πραγματικότητα δεν είναι ένας απλός αγωγός αλλά περιλαμβάνει και άλλες βοηθητικές διατάξεις για τον έλεγχο των συνδεδεμένων μονάδων). Το bus με άλλα λόγια είναι μια κεντρική *λεωφόρος* κυκλοφορίας πληροφοριών με πολλές "λουρίδες κυκλοφορίας" (καλώδια) που διαπερνά όλες τις μονάδες του Υ.

Χαρακτηριστικά Μεγέθη του Bus

Δυναμικότητα του Bus -Αριθμός λωρίδων (γραμμών, αγωγών). Οπως και σε μια πραγματική λεωφόρο έτσι και στο bus σημαντικό ρόλο στην επιτυχία του σαν γρήγορου "αγωγού" πληροφοριών έχει ο αριθμός των "λωρίδων" κυκλοφορίας του, δηλ. των καλωδίων από τα οποία αποτελείται.
Οσο πιο πολλά καλώδια λοιπόν έχει ένα bus, τόσο πιο γρήγορα διακινεί τις πληροφορίες, είναι δηλ πιο ισχυρό.



Σχ. 6.2
Απλοποιημένη Αρχιτεκτονική PC

Επειδή από κάθε καλώδιο περνάει και ένα bit, συνήθως την ισχύ του bus τη μετράμε σε αριθμό (ταυτόχρονα μεταδιδόμενων) bits π.χ. λέμε 8-μπιτο, 16-μπιτο, 32-μπιτο bus.

Ταχύτητα του Bus. Μετριέται με τον αριθμό των διακινουμένων bits ή bytes το δευτερόλεπτο και η μονάδα μέτρησης είναι MBPS (Mega Bytes Per Sec δηλ εκατομμύρια bytes το δευτερόλεπτο).

Εσωτερική Εικόνα του Υ

Αν "περάσουμε" τον Υ από *ακτίνες Χ* , για να φύγουνε τα περιττά στολίδια και να μείνουν τα απαραίτητα θα δούμε την *ακτινογραφία* του συστήματος PC που φαίνεται στο Σχ. 6.2.

Το bus. Η γενική αρχιτεκτονική φιλοσοφία του Υ προβλέπει πως το bus αποτελεί τη "σπονδυλική στήλη" του υπολογιστή και πάνω του "κρεμιούνται" οι διάφορες μονάδες του Υ. Ολη η λειτουργία του Υ διεξάγεται με ένα αδιάκοπο πήγαιν' έλα πληροφοριών από τα μηχανήματα εισόδου π.χ. Πληκτρολόγιο, προς τη Μνήμη (μέσω Επεξεργαστή), από εκεί επεξεργασία στον Επεξεργαστή, αποτελέσματα στη Μνήμη και τέλος έξοδός τους από τη Μνήμη προς τα μηχανήματα εξόδου.

Τοπικός δίαυλος (local bus). Οι τοπικοί δίαυλοι (ΤΔ) είναι μία μέθοδος που επιχειρεί να δώσει λύση στην ανάγκη αυξημένης ταχύτητας επικοινωνίας μεταξύ συγκεκριμένων περιφερειακών και του Ε. Η σκέψη ήταν απλή, συνδέουμε τα περιφερειακά που έχουν αυξημένες απαιτήσεις ταχύτητας (π.χ. οθόνες και δίσκοι), απευθείας με τον Επεξεργαστή επιτυγχάνοντας έτσι ταχύτητες επικοινωνίας ίδιες με την ταχύτητα του Επεξεργαστή.

Κάρτες διασύνδεσης και πόρτες. Η σύνδεση κάθε μονάδας με το bus δε γίνεται άμεσα αλλά μέσω μιας ειδικής, για κάθε περιφερειακό, διάταξης διασύνδεσης (interface). Τα κυκλώματα αυτά διασύνδεσης τοποθετούνται σε μία κάρτα συνήθως μαζί με άλλα κυκλώματα που αφορούν το ίδιο περιφερειακό (π.χ. κυκλώματα ελέγχου, controller). Η κάρτα αυτή είναι χαρακτηριστική για το Π και λέγεται κάρτα του Π. Η συνεχής ελαχιστοποίηση των κυκλωμάτων έχει επιτρέψει να τοποθετούνται σε μία κάρτα κυκλώματα πολλών Π (πολυκάρτες), όπως ακόμα και να ενσωματώνονται στη βασική πλακέτα του Υ που περιλαμβάνει τον Ε και τη

Μ και λέγεται μητρική πλακέτα (motherboard). Οι κάρτες αυτές από τη μέσα μεριά συνδέονται με το bus και από την έξω με το αντίστοιχο περιφερειακό, με τη βοήθεια ειδικής υποδοχής (πρίζας) που λέγεται πολύ παραστατικά, πόρτα (port). Στην πόρτα αυτή προσαρμόζεται το φις του καλωδίου του περιφερειακού.
Βλέπουμε λοιπόν πως για ένα περιφερειακό (Π) π.χ. το πληκτρολόγιο όταν χτυπάμε ένα πλήκτρο τα αντίστοιχα σήματα ακολουθούν την πορεία:
Περιφερειακό - καλώδιο Περιφερειακού- φις - πόρτα - κάρτα - bus - και μέσω Επεξεργαστή καταλήγουν στη μνήμη

Κάθε PC που αγοράζουμε έχει έτοιμες πόρτες για τη σύνδεση των βασικών περιφερειακών : πληκτρολογίου, οθόνης, εκτυπωτή. Οι πόρτες αυτές επιτρέπουν την δίοδο των πληροφοριών (των bits τους) σειριακά ή παράλληλα και λέγονται αντίστοιχα σειριακές ή παράλληλες.

Εκτός όμως από τις έτοιμες πόρτες (που προυποθέτουν πως θα υπάρχουν τοποθετημένες οι αντίστοιχες κάρτες) συνήθως προβλέπει και ελεύθερες θέσεις (σχισμές, slots) στο πίσω μέρος του PC για την τοποθέτηση καρτών εξειδικευμένων περιφερειακών που μπορεί να αγοράσει κανείς από το εμπόριο προκειμένου να συνδέσει το περιφερειακό που τον ενδιαφέρει.

Ανοικτή - κλειστή αρχιτεκτονική

Η φιλοσοφία αυτή που προαναφέραμε και προβλέπει ένα bus και σχισμές - slots με γνωστές, τυποποιημένες προδιαγραφές, στις οποίες μπορούμε να συνδέσουμε όποιο Π επιθυμούμε, αρκεί να διαθέτουμε (ή να κατασκευάσουμε) τη κατάλληλη κάρτα προσαρμογής, ονομάζεται ανοικτή αρχιτεκτονική. Η φιλοσοφία αυτή προωθήθηκε από την IBM με το περίφημο PC.
Αντίθετα η μέχρι τότε πρακτική προέβλεπε τη δημιουργία κλειστών συστημάτων δηλ συστημάτων Υ με αυστηρά καθορισμένη σύνθεση από συγκεκριμένες μονάδες, που ο χρήστης δεν μπορούσε να αλλάξει.

Η Δομή του Υ

Στο σύστημα του Υ, όπως το είδαμε παραπάνω, μπορούμε εύκολα να ξεχωρίσουμε την Κεντρική Μονάδα με τις βασικές εσωτερικές μονάδες (Ε, Μ, Β) και τα διάφορα Υποσυστήματα των Περιφερειακών (οθόνη, Δίσκος

κλπ). Κάθε Υ/Σ αποτελείται από το ίδιο το Περιφερειακό και από τα ηλεκτρονικά κυκλώματα που υποστηρίζουν και ελέγχουν τη λειτουργία του. Τα κυκλώματα αυτά άλλοτε είναι ενσωματωμένα στο Π και άλλοτε είναι σε χωριστή κάρτα που τοποθετείται κατάλληλα στη μητρική. Ο μηχανισμός ελέγχου κάθε Π συχνά περιλαμβάνει ένα είδος επεξεργαστή που ανάλογα με τη περίπτωση μπορεί να είναι απλός ή εξελιγμένος και ισχυρός όσο σχεδόν και ο κεντρικός επεξεργαστής και η Μνήμη. Αυτά λέγονται τοπικός Ε και Μ για να τα αντιδιαστείλουμε από το Κ Ε και Μ. Ο ρόλος των τοπικών αυτών Ε και Μ είναι ευεργετικός για τη λειτουργία του όλου συστήματος γιατί επέρχεται μια αποκέντρωση της επεξεργασίας και δεν επιβαρύνεται αποκλειστικά ο ΚΕ με όλες τις εργασίες.

Μετά από αυτά διακρίνουμε τα ακόλουθα λειτουργικά αυτοδύναμα κομμάτια σε ένα σύστημα Υ:

- **την Κεντρική Μονάδα** που περιλαμβάνει κυρίως τα εσωτερικά κομμάτια του Υ δηλ. τον Επεξεργαστή, τη Μνήμη, το Bus κλπ
- **το Υποσύστημα (Υ/Σ) Δίσκου** που περιλαμβάνει το Δίσκο και τη Κάρτα ελέγχου του
- **το Υ/Σ Οθόνης** που περιλαμβάνει την Οθόνη και την Κάρτα του λεγόμενου προσαρμοστή (adapter) της
- **το Υ/Σ Αργών Περιφερειακών** που περιλαμβάνει τις έτοιμες πόρτες σειριακές και παράλληλες και τα στάνταρ "αργά" περιφερειακά (εκτυπωτές, πληκτρολόγιο, ποντίκι, μόντεμ που όπως θα δούμε αργότερα χρειάζεται για τις μακρινές επικοινωνίες).

7 ΚΕΝΤΡΙΚΗ ΜΟΝΑΔΑ

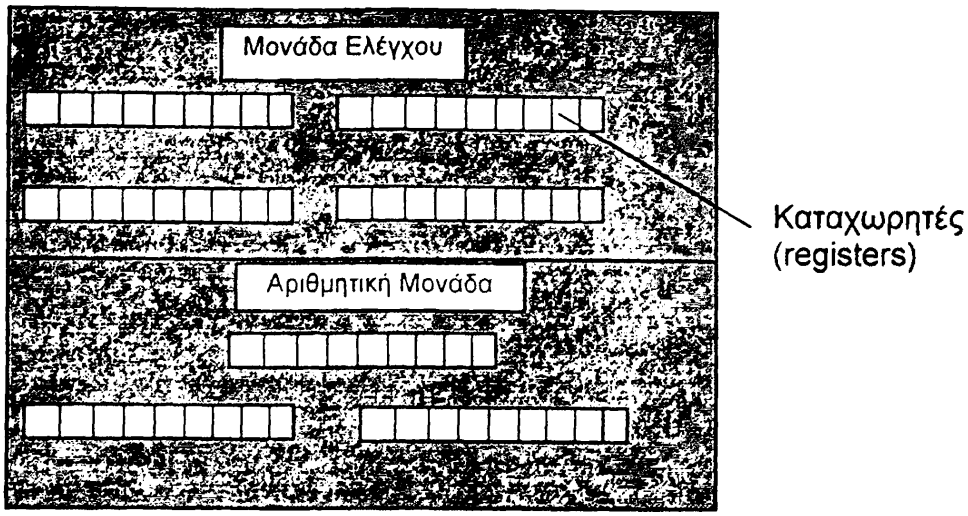
Ο Επεξεργαστής και η Μνήμη

Ο Επεξεργαστής

Ο επεξεργαστής είναι η μονάδα που διεκπεραιώνει τις επεξεργασίες των στοιχείων που έχουν καταχωρηθεί στη μνήμη. Ο τρόπος λειτουργίας είναι:

- ο Ε ζητάει στοιχεία (οδηγίες και δεδομένα) από τη μνήμη που του χρειάζονται για τις επεξεργασίες του και συγκεκριμένα:
    - οδηγίες. Καλεί κάθε μία οδηγία του προγράμματος με τη σειρά από τη μνήμη, την τοποθετει σε ειδική θέση του (καταχωρητή), την αναγνωρίζει και την εκτελεί.
    - δεδομένα. Αν η οδηγία αυτή χρειάζεται κάποια δεδομένα για να εκτελεσθεί (π.χ. πρόσθεση του Βασικού Μισθού με το Επίδομα που είδαμε στο προηγούμενο κεφάλαιο), τότε ο Ε καλεί τα δεδομένα αυτά από τη μνήμη και τα τοποθετεί σε ειδικές θέσεις (καταχωρητές)
- ο Ε επεξεργάζεται κατάλληλα τα στοιχεία που έχει τοποθετήσει στους καταχωρητές του

Βλέπουμε λοιπόν πως και ο Ε ουσιαστικά κάνει ό,τι και ένας τεχνίτης με τον *πάγκο εργασίας* του. Για παράδειγμα ένας ξυλουργός: παίρνει ξύλα από την αποθήκη του (*μνήμη*), τα τοποθετεί στις ειδικές θέσεις του πάγκου του (*καταχωρητές*) και τα κόβει, πλανίζει, καρφώνει (*επεξεργάζεται*) και μετά τα τοποθετεί στην αποθήκη, επεξεργασμένα (ή μισοεπεξεργασμένα με σκοπό να τα ξαναπάρει όταν έλθει η ώρα και να συνεχίσει μέχρι ολοκλήρωσης της επεξεργασίας (τελικό προϊόν, τραπέζι, καρέκλα, σαλόνι κλπ).



Σχ. 7.1 Σχηματική Διάταξη Επεξεργαστή

chip μΕ: Η υλοποίηση των δυνατοτήτων αυτών γίνεται με τη βοήθεια μεγάλου αριθμού κυκλωμάτων περισσότερο σύνθετων από της μνήμης. Ολα τα κυκλώματα αυτά περιέχονται σε ένα τσιπ που ονομάζεται τσιπ μικροεπεξεργαστή (microprocessor chip).

Τα χαρακτηριστικά του μΕ

Ποια είναι τα βασικά μεγέθη που χαρακτηρίζουν την ισχύ ενός μΕ;
Γενικά μπορούμε να πούμε πως για κάθε μονάδα του Υ δύο είναι τα βασικά μεγέθη που χαρακτηρίζουν τη δύναμή της, πόσο γρήγορη είναι και πόσο "μεγάλη" είναι. Τα στοιχεία αυτά μετριούνται διαφορετικά για κάθε μονάδα. Για τον Ε έχουμε τα εξής μεγέθη:

- ***Ταχύτητα Ε.*** Μετριέται με το πόσο πολλές στοιχειώδεις εργασίες κάνει ο Ε το sec ή αλλιώς όπως λέμε από το ρυθμό λειτουργίας του Ε. Ο ρυθμός αυτός εξαρτάται από το εσωτερικό ρολόι του Ε. Σε κάθε "χτύπο" του ρολογιού ο Ε εκτελεί μια στοιχειώδη λειτουργία. Ο αριθμός των χτύπων ανά δευτερόλεπτο ονομάζεται συχνότητα του ρολογιού. Η μονάδα μέτρησης της συχνότητας είναι ένας χτύπος ανά sec και ονομάζεται 1 Herz (Hz). Ενα ρολόι Ε δουλεύει με ταχύτητα πολλών εκατομμυρίων χτύπων το δευτερόλεπτο ή πολλών MHz (Mega Herz). Αντίστοιχα λοιπόν καθιερώθηκε και η ταχύτητα του Ε να μετριέται με την ταχύτητα του ρολογιού του σε MHz.

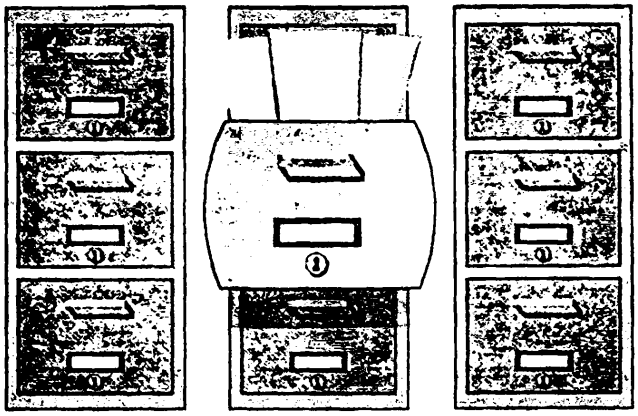
- ***Μέγεθος Ε.*** Μετριέται με το μέγεθος των καταχωρητών του σε bits. Βασικό μέγεθος που συνήθως είναι ίσο με το μήκος της λέξης της μνήμης. Το στάνταρ σήμερα είναι η 32-μπιτη τεχνολογία.

Ιστορική και Τεχνολογική Εξέλιξη των μικρο-Επεξεργαστών PC

Η εξέλιξη των μΕ κυριαρχείται από την πρωτοπόρο εταιρία INTEL. Η μεγάλη της ώρα σήμανε το 1981, όταν η IBM της ανέθεσε να κατασκευάσει ένα μΕ για την σειρά μικρο- υπολογιστών που ετοίμαζε. Ο πρώτος αυτός μΕ ονομάστηκε 8088 και ήταν 8-μπιτος. Η επιτυχία των PC, έδωσε στην INTEL ένα σημαντικό προβάδισμα στην κατασκευή μΕ που εκείνη γρήγορα το μετέτρεψε σε ουσιαστικό μονοπώλιο της αγοράς των μΕ. Τη σειρά των μΕ της η INTEL την ονόμασε 80χ86 με τη τιμή του χ να αυξάνεται για κάθε ισχυρότερο μοντέλο (2,3 κλπ).
Σημείο καμπής ο 30386 ή απλά 386 (το 1987) που έδωσε στους μΕ χαρακτηριστικά μεγάλων Υ (32-μπιτη τεχνολογία, διαχείριση μνήμης, πολυπρογραμματισμό κλπ).

Η Μνήμη

Η Κεντρική Μνήμη ή μνήμη RAM όπως έχει καθιερωθεί να λέγεται είναι η βασική μνήμη που ενδιαφέρει το χρήστη. Είναι η μνήμη στην οποία τοποθετούνται τα προγράμματα και δεδομένα του χρήστη προκειμένου να επεξεργαστούν από τον επεξεργαστή. Η μνήμη αυτή έχει την ιδιότητα να χάνει τα περιεχόμενά της, όταν σβήσουμε τον Υ μας.
Αυτό σημαίνει πως όταν θέλουμε να κρατήσουμε κάποια στοιχεία για μετέπειτα χρήση θα πρέπει να τα τοποθετήσουμε κάπου που θα είναι ασφαλή (στο δίσκο). Η μνήμη δηλ δεν είναι τίποτα άλλο παρά ένα πρόχειρο σημειωματάριο, ο μαυροπίνακας της αίθουσας διδασκαλίας, που μας χρησιμεύει για να κάνουμε κάποιες πράξεις και να βγάζουμε κάποια αποτελέσματα, τα οποία όμως στη συνέχεια πρέπει να περάσουμε στον καθαρό δίσκο (σκληρό) για να τα κρατήσουμε. Στο σκληρό δίσκο η μνήμη τοποθετεί τα στοιχεία σε συγκεκριμένες θέσεις και λειτουργεί με τον ίδιο τρόπο που αρχειοθετούμε τα στοιχεία μας σε μία βιβλιοθήκη όπως φαίνεται από την παρακάτω εικόνα.



Χαρακτηριστικά Στοιχεία της Μνήμης

Μήκος λέξης. Είπαμε παραπάνω πως μια θέση μνήμης λέγεται λέξη της μνήμης και αποτελείται από τα bits. Ο αριθμός των bits μιας λέξης ονομάζεται μήκος της λέξης (βλ. σχ. 2.1 στο κεφ.2). Οσο πιο μεγάλο είναι το μήκος της λέξης τόσο πιο *δυνατός* θεωρείται ο PC. Υπάρχουν PC με λέξεις των 8, 16, 32 bits.

Byte. Ιδιαίτερη θέση στην ορολογία της μνήμης έχει το λεγόμενο byte που είναι μια ομάδα από 8 bits. Οπως όμως είδαμε και παραπάνω σε 8 bits μπαίνει ένας χαρακτήρας, για το λόγο αυτό τα bytes λέγονται συχνά και χαρακτήρες.

Χωρητικότητα. Το πλήθος των bytes ονομάζεται χωρητικότητα της μνήμης. Η χωρητικότητα της μνήμης εκφράζεται σε χιλιάδες (συμβολίζεται Κ) ή εκατομμύρια (συμβολ. Μ) bytes (συμβολ Β). Για να είμαστε πιο ακριβείς 1 Κ δεν είναι ακριβώς 1000 αλλά 1024 δηλ. όσα χωράει ένας δεκαψήφιος δυαδικός αριθμός (ομάδα των 10 bits), όπως είδαμε σε προηγούμενη ενότητα.

Υλοποίηση της Μνήμης - Τσιπς μνήμης

Η μνήμη, δηλ. τα κυκλώματα που την υλοποιούν κατασκευάζονται υπό μορφή τσιπς.

Χαρακτηριστικά ισχύος του τσιπ μνήμης

Τα βασικά χαρακτηριστικά ενός τσιπ μνήμης είναι:

- *η χωρητικότητά του σε bits* (πόσα στοιχειώδη κυκλώματα του 1 bit έχει) και

- *η ταχύτητά του* που συναντάται και σαν Χρόνος Πρόσβασης ή Access Time. Αναφέρεται στο χρόνο που περνάει από τη στιγμή που ο Ε ζητάει ένα στοιχείο από τη μνήμη μέχρι τη στιγμή που η μνήμη θα του το έχει έτοιμο να το πάρει. Ο χρόνος πρόσβασης στα σύγχρονα τσιπς μετριέται σε δισεκατομμυριοστά του sec - nanoseconds (ns).

Είδη της Μνήμης - Chips RAM και ROM

Μνήμη RAM. Η κύρια μνήμη του Υ που είδαμε παραπάνω επιτρέπει στο χρήστη να διαβάζει αυτά που είναι γραμμένα μέσα αλλά και να γράφει. Ονομάζεται μνήμη RAM (Random Access Memory) και υλοποιείται με τα τσιπς RAM. Το μέγεθος της είναι ένας σημαντικός παράγοντας για να εκτιμήσουμε τις δυνατότητες του υπολογιστή μας.

Μνήμη ROM. Μόνο για Διάβασμα. Η μνήμη ROM ή Read Only Memory χρησιμοποιείται για μόνιμη καταχώρηση στοιχείων. Επιτρέπει το διάβασμα των περιεχομένων της αλλά, αντίθετα με τη RAM, δεν επιτρέπει στον χρήστη να γράψει σ' αυτήν (ή να αλλάξει τα περιεχόμενά της) με τα προγράμματά του. Ακόμη έχει την ιδιότητα να διατηρεί τα περιεχόμενά της και μετά από το κλείσιμο του Υ. Η μνήμη με τις ειδικές αυτές ιδιότητες είναι πιο ακριβή από τη RAM και διατίθεται σε μικρότερες χωρητικότητες από τη RAM. Η ROM έρχεται *γραμμένη* από τον κατασκευαστή του Υ και χρησιμοποιείται για την τοποθέτηση των πιο βασικών για τη λειτουργία του Υ προγραμμάτων και στοιχείων.
Ακόμη έχουμε και ROM που μπορούν να αλλάξουν ή να προγραμματιστούν και ονομάζονται:
EROM και
EPROM.

Υπερταχεία Μνήμη - Cache. Η μνήμη RAM είναι πιο αργή από τον Επεξεργαστή, όπως θα δούμε στο επόμενο κεφάλαιο. Αυτό σημαίνει πως όταν ζητάει κάτι ο Ε από τη μνήμη θα περιμένει μέχρι να του παραδοθεί. Για να μειωθούν οι καθυστερήσεις οι σχεδιαστές των Υ χρησιμοποιούν σε μερικές περιπτώσεις ένα άλλο είδος μνήμης, τη μνήμη cache. Η μνήμη αυτή έχει ταχύτητες που προσεγγίζουν εκείνες του Επεξεργαστή αλλά είναι πολύ ακριβή και έτσι δεν μπορούμε να την χρησιμοποιήσουμε σε μεγάλες χωρητικότητες σαν Κύρια Μνήμη.

Μπορεί όμως να χρησιμοποιηθεί σε μικρές χωρητικότητες σαν βοηθητική της RAM για την τοποθέτηση εκείνων των εντολών του προγράμματος και των δεδομένων που χρησιμοποιούνται πιο συχνά σε κάθε δεδομένη στιγμή της ροής του προγράμματος (το σύστημα μπορεί να τα προσδιορίζει). Έχει διαπιστωθεί πως η χρήση cache επιταχύνει σημαντικά την απόδοση του συστήματος γι' αυτό χρησιμοποιείται στα συστήματα που έχουν αυξημένες ανάγκες ταχύτητας επεξεργασίας.

8 ΥΠΟΣΥΣΤΗΜΑ ΔΙΣΚΩΝ

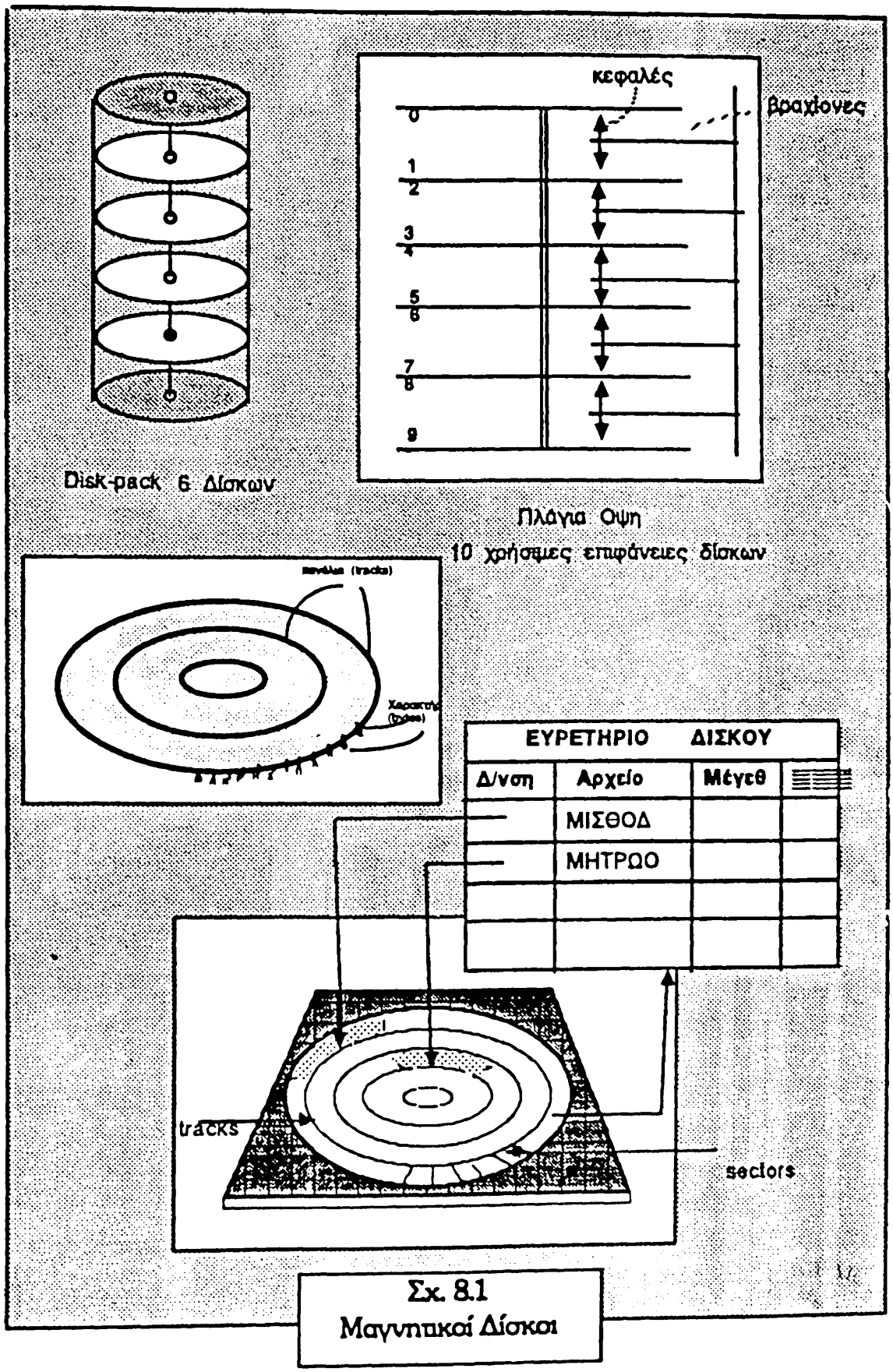
Ο Δίσκος και ο Ελεγκτής του

Ενα Υ/Σ Δίσκου αποτελείται από το Δίσκο και την κάρτα ελέγχου του. Το Υ/Σ Δίσκου επιτρέπει την αποθήκευση (*γράψιμο* από τη μνήμη στο δίσκο) και ανάληψη (*διάβασμα* από το δίσκο στη μνήμη) πληροφοριών σε ένα σύστημα Υπολογιστή.

Ο Δίσκος

Λέγοντας συντομογραφικά *δίσκο* εννοούμε το δίσκο όπου είναι γραμμένα τα στοιχεία μας και το μηχάνημα που γράφει σ' αυτόν (από τη Μ) και διαβάζει από αυτόν, για να τα μεταφέρει στη Μ. Η λειτουργία του μηχανογραφικού δίσκου είναι παρόμοια με εκείνη του πικ-απ. Υπάρχει όπως και στο πικ-απ ο δίσκος που περιστρέφεται περί άξονα και η κεφαλή μέσω της οποίας "διαβάζονται" οι πληροφορίες (δεδομένα αντί ήχων) που είναι γραμμένες στο δίσκο. Ακόμη, στο μηχανογραφικό δίσκο, "γράφονται" πληροφορίες (όπως στα κασετόφωνα). Ο δίσκος τοποθετείται σε ένα μηχάνημα που λέγεται οδηγός δίσκου (disc drive).

Και οι δίσκοι, όπως και όλο το υπόλοιπο σύστημα του Υ, λειτουργούν στο δυαδικό σύστημα. Και εδώ έχουμε υλικά δύο καταστάσεων που σε αντίθεση με τη ΚΜ δεν είναι ηλεκτρονικά - τρανζίστορ αλλά είναι είτε μαγνητικά (οπότε έχουμε τους μαγνητικούς δίσκους) είτε οπτικά (οπτικοί δίσκοι αντίστοιχοι των γνωστών κόμπακτ ντισκς για μουσική) (Εικόνα 8.1).



Σχ. 8.1
Μαγνητικοί Δίσκοι

Μαγνητικοί Δίσκοι

Οι μαγνητικοί δίσκοι διακρίνονται στους μεταλλικούς, τους λεγόμενους σκληρούς δίσκους και στους πλαστικούς δίσκους ή εύκαμπτους δίσκους, τους γνωστούς φλόππυ (floppy).
Οι επιφάνειες των δίσκων είναι καλυμμένες με κατάλληλο μαγνητικό υλικό που μπορεί να αναγνωρίζει δύο καταστάσεις (μαγνητισμένη - μη μαγνητισμένη) και έτσι δίνει τη δυνατότητα τοποθέτησης στοιχείων (bytes) σε δυαδική μορφή όπως η μνήμη.
Η τοποθέτηση των στοιχείων στους δίσκους γίνεται κατά μήκος ομόκεντρων περιφερειών (Σχ. 8.1). Αν δούμε πράγματι από πιο κοντά την επιφάνεια ενός δίσκου θα διαπιστώσουμε ότι αποτελείται από ομόκεντρες περιφέρειες μαγνητικού υλικού, τα tracks (από 40 στους μικρούς μέχρι μερικές εκατοντάδες στους μεγάλους) στα οποία τοποθετούνται τα προς αρχειοθέτηση στοιχεία, σε διαδοχικά bytes - χαρακτήρες. Ενα τρακ χωρίζεται σε ισομεγέθη τόξα, που ονομάζονται sectors (10-100 ανά track). Κάθε sector αποτελείται (συνήθως) από 512 bytes. Το sector αποτελεί τη μονάδα μεταφοράς μεταξύ δίσκου και μνήμης. Οταν λέμε πως ένα αρχείο είναι στο δίσκο, εννοούμε πως όλες οι εγγραφές του, η μία μετά την άλλη χαρακτήρα-χαρακτήρα, καταχωρούνται σε διαδοχικά τρακς του δίσκου. Ο αριθμός των χαρακτήρων που χωράει ένας δίσκος (με την παραπάνω γενική έννοια), μας δίνει την χωρητικότητα του δίσκου και μετριέται σε εκατομμύρια χαρακτήρες (MBytes-MB). Η χωρητικότητα ενός δίσκου εξαρτάται από άλλα δευτερεύοντα χαρακτηριστικά των δίσκων, όπως είναι η πυκνότητα των χαρακτήρων σε ένα τρακ (μετριέται σε Bytes Per Inch-BPI) και η πυκνότητα των τρακς στο δίσκο (Tracks Per Inch-TPI)

Μηχάνημα Δίσκου (disk drive)

Η ανάγνωση/εγγραφή πληροφοριών του δίσκου από/προς τη μνήμη γίνεται όταν τον τοποθετήσουμε στο κατάλληλο μηχάνημα-μονάδα δίσκων (όπως κατ' αναλογία συμβαίνει με τους δίσκους του πικ-άπ). Το μηχάνημα αυτό περιστρέφει το δίσκο με σταθερή ταχύτητα (συνήθως 3600 στροφές το λεπτό για τους σκληρούς δίσκους) και οι πληροφορίες διαβάζονται ή γράφονται με τη βοήθεια κατάλληλα κινούμενων κεφαλών. Οι κεφαλές αυτές ανάγνωσης/εγγραφής βρίσκονται στο άκρο βραχιόνων για να διευκολύνεται η πρόσβαση σε οποιαδήποτε περιοχή του δίσκου.

Πώς βρίσκουμε τις πληροφορίες στο Δίσκο - Ευρετήριο Δίσκου

Οι πληροφορίες στο δίσκο δεν είναι χύμα αλλά ομαδοποιημένες σε ενότητες ομοειδών πληροφοριών που λέγονται. Σε κάθε αρχείο πληροφοριών δίνεται ένα όνομα με το οποίο αναγνωρίζονται στον Υ. Είναι προφανές ότι σε ένα δίσκο μεγάλης χωρητικότητας μπορούμε να τοποθετήσουμε πολλά αρχεία. Για να μπορούν όμως οι χρήστες να βρίσκουν τα αρχεία που θέλουν, κάθε δίσκος είναι εφοδιασμένος με το Ευρετήριο Αρχείων ή File Directory όπως λέγεται συνήθως που περιέχει για κάθε αρχείο το όνομά του, τη διεύθυνσή του μέσα στο δίσκο (διεύθυνση του πρώτου sector του) και το μέγεθός του δηλαδή τον αριθμό συνεχόμενων sectors που καταλαμβάνει (βλ. Σχ. 8.1).

Αφού εντοπίσουμε μέσα στο δίσκο το αρχείο που μας ενδιαφέρει η αναζήτηση στοιχείων (εγγραφών) σε αυτό μπορεί να γίνει σειριακά. Μπορεί όμως να γίνει και άμεσα με την τοποθέτηση (με τον κατάλληλο μηχανισμό) της κεφαλής ανάγνωσης/εγγραφής απευθείας στην πληροφορία που θέλουμε (όπως και με τα τραγούδια των δίσκων του πικ-απ). Επειδή ο δίσκος παρέχει αυτή τη δυνατότητα της άμεσης πρόσβασης στην επιθυμητή πληροφορία, χαρακτηρίζεται και σαν άμεσο ή Τυχαίας Προσπέλασης- Random Access μέσο αρχειοθέτησης.

Χαρακτηριστικά Δίσκων

Χωρητικότητα του δίσκου: Ο αριθμός των χαρακτήρων (bytes) που μπορεί να *χωρέσει* ένας δίσκος και μετριέται σε εκατομμύρια (MB) ή και δισεκατομμύρια (GB) χαρακτήρες.

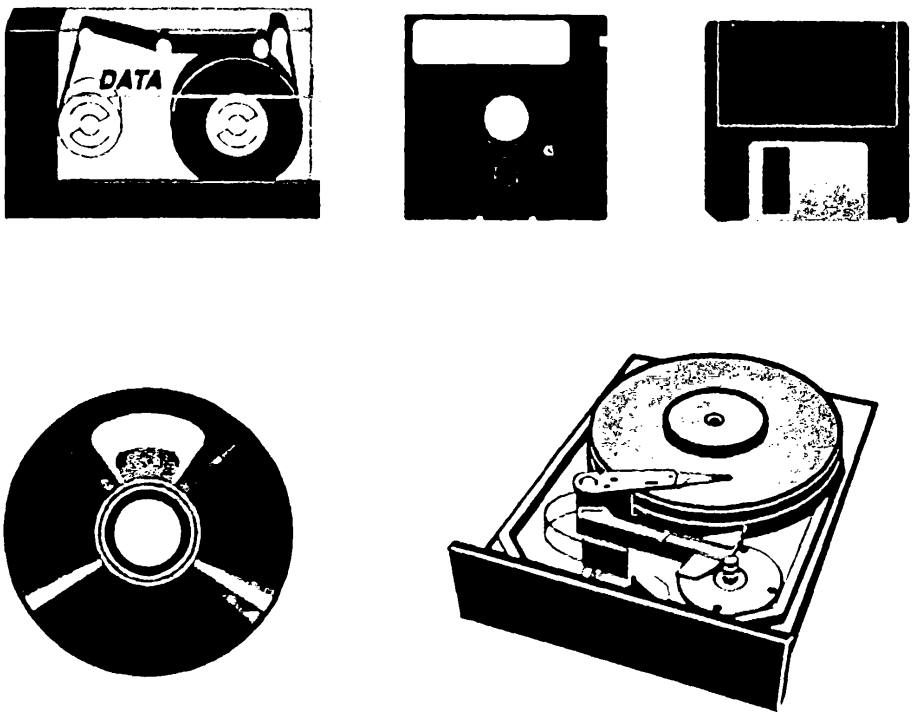
Ταχύτητα. Τα στοιχεία που φυλάγονται στους δίσκους πρέπει να μεταφερθούν στη μνήμη για να γίνει η επεξεργασία τους. Χαρακτηριστικό μέγεθος μεταφοράς στοιχείων είναι η ταχύτητα μεταφοράς (transfer rate) που μετριέται σε εκατομμύρια χαρακτήρες το δευτερόλεπτο (MBPS) και μπορεί, στα καλά μηχανήματα δίσκων, να φτάσει τα 6MBPS.
Ενα άλλο μέτρο της ταχύτητας ενός δίσκου είναι ο *χρόνος πρόσβασης* ή *access time* που δίνει τον χρόνο που χρειάζεται ο μηχανισμός του δίσκου για τον εντοπισμό της πληροφορίας σε ένα τρακ. Μετριέται σε msec.

Είδη μαγνητικών δίσκων

Οι μαγνητικοί δίσκοι διακρίνονται στους λεγόμενους σκληρούς (hard discs) και στους μαλακούς (floppy discs) (Εικόνα 8.1.).

Οι ***μαλακοί δίσκοι*** ή δισκέτες είναι πλαστικοί δίσκοι καλυμένοι από μαγνητικό υλικό και υπάρχουν σε δύο παραλλαγές τους μεγάλους διαμέτρου 5.25" (τείνουν να εξαφανιστούν) και τους μικρούς 3.5".
Το μηχάνημα που τους παίζει βρίσκεται μέσα στο κουτί του Υ και επικοινωνεί με τον έξω κόσμο με μία οριζόντια σχισμή για να μπορεί ο χρήστης να τοποθετεί τη δισκέτα του. Οι δισκέτες που χρησιμοποιούνται κυρίως σήμερα είναι 3.5" των 1.44MB ή 2.88 MB.

Οι ***σκληροί δίσκοι***: είναι φτιαγμένοι από αλουμίνιο και δεν πουλιούνται χύμα αλλά ενσωματωμένοι δύο, τρεις ή περισσότεροι μαζί με το μηχάνημα που τους *παίζει* (σαν μικρό τζουκ-μπόξ). Οι χωρητικότητές τους ποικίλλουν από 300 MB μέχρι πολλές εκατοντάδες MB ή ακόμα και δισεκατομμύρια bytes (GB).



Εικόνα 8.1. Δίσκοι και disk drive Η/Υ.

Οπτικοί Δίσκοι

Τα τελευταία χρόνια όλο και περισσότερο συζητούνται οι οπτικοί δίσκοι κυρίως λόγω της τεράστιας ικανότητας καταχώρησης στοιχείων που διαθέτουν. Η πρώτη γενιά των οπτικών δίσκων ανακαλύφθηκε από την εταιρεία Φίλιπς βασισμένη στην ίδια τεχνολογία με τους γνωστούς κόμπακτ ντίσκς και ονομάστηκε CD-ROM (Compact Disc Read Only Memory).
Ο τρόπος κατασκευής των δίσκων αυτών ήταν ανάλογος των κόμπακτ δίσκων: Με τη χρησιμοποίηση ισχυρής δέσμης ακτίνων laser δημιουργούμε μικροσκοπικές (και ανεξίτηλες) οπές στην επιφάνεια ενός δίσκου-προτύπου (master disk). Οι οπές αυτές λέγονται pits και αντιστοιχούν στο "1", ενώ τα ανέπαφα από τις ακτίνες σημεία λέγονται lands και αντιστοιχούν στο "0". Ετσι καταχωρούνται οι πληροφορίες (bytes) υπό δυαδική - οπτική μορφή. Από τον έτσι δημιουργούμενο δίσκο, κατασκευάζουμε δίσκο-μήτρα, με βάση τον οποίο στη συνέχεια κατασκευάζουμε αντίγραφα τους λεγόμενους CD ROM δίσκους.
Οι δίσκοι αυτοί έχουν το βασικό πλεονέκτημα των μεγάλων χωρητικοτήτων (πάνω από 500 MB) και το μειονέκτημα ότι επιτρέπουν στο χρήστη μόνο να τους διαβάσει και όχι να γράψει σ' αυτούς όπως μπορεί με τους μαγνητικούς δίσκους. Τα τελευταία χρόνια τα μηχανήματα που γράφουν σε CD-ROM έχουν φτηνύνει σημαντικά οπότε μπορεί κανένας εύκολα να δημιουργήσει τους δικούς του CD.

Η δεύτερη γενιά των οπτικών δίσκων, οι λεγόμενοι WORM (Write Once Read Many), έδινε στους χρήστες κενούς δίσκους (όπως συνέβαινε με τους συνηθισμένους μαγνητικούς) και τους επέτρεπε να γράφουν (δηλ να δημιουργούν δικά τους αρχεία). Δεν τους επέτρεπε όμως να τα σβύνουν και να γράφουν άλλα (όπως στους μαγνητικούς δίσκους). Μικρό το κακό δεδομένης της τεράστιας χωρητικότητας των δίσκων αυτών.

Η τρίτη γενιά προφανώς αναφέρεται στην κατασκευή οπτικών δίσκων που να επιτρέπουν εγγραφή/σβύσιμο όπως ακριβώς οι μαγνητικοί δίσκοι. Πρόσφατα κυκλοφόρησαν τέτοιοι δίσκοι που κατασκευάζονται με συνδυασμό οπτικών και μαγνητικών μεθόδων.

Θα *αντικαταστήσουν οι οπτικοί τους μαγνητικούς δίσκους κάποτε;* Στο ερώτημα αυτό κανένας δεν μπορεί να δώσει σαφή απάντηση παρόλο που πολλοί τεχνικοί το θεωρούν πολύ δύσκολο. Υπάρχει πραγματικά η άποψη πως οι επιδόσεις των οπτικών δίσκων (ταχύτητες) είναι, και παρά τις όποιες εξελίξεις, θα εξακολουθήσουν να είναι σημαντικά κατώτερες των

αντίστοιχων των μαγνητικών. Το μεγάλο πλεονέκτημα των οπτικών είναι οι χωρητικότητές τους, που σημαίνει πως η σημασία τους θα αυξάνεται συνεχώς σε εφαρμογές απαιτητικές σε όγκους στοιχείων.

Ελεγκτές Δίσκων - Controllers

Η κάρτα του δίσκου περιλαμβάνει όλα τα κυκλώματα διασύνδεσης, ελέγχου και υποστήριξης λειτουργίας του δίσκου.
Ο καλός σχεδιασμός της κάρτας αυτής μπορεί να αυξήσει σημαντικά την ταχύτητα μεταφοράς στοιχείων και γενικά τις επιδόσεις του δίσκου.
Εχουν αναπτυχθεί αρκετά πρότυπα τέτοιων καρτών από τα οποία τα επικρατέστερα είναι δύο:
- Το πιο γενικό SCSI που μπορεί να συνδέσει πολλούς δίσκους με τον κεντρικό Υ αλλά και άλλα μηχανήματα (ταινίες, CD-ROM κλπ). Είναι κατάλληλο για επαγγελματικές εγκαταστάσεις. Θα πρέπει να προσέξουμε ότι αν έχουμε κάρτα SCSI οι δίσκοι που θα προμηθευόμαστε να είναι συμβατοί με την κάρτα που διαθέτουμε.
- Το πιο διαδεδομένο είναι το IDE πρότυπο. Εδώ υπάρχει η ιδιορρυθμία τα κυκλώματα ελέγχου να είναι ενσωματωμένα με το δίσκο που αγοράζουμε και έτσι η σύνδεση να είναι πιο άμεση. Στο πρότυπο αυτό μπορούμε να συνδέσουμε μέχρι δύο δίσκους. Αυτό βέβαια βάζει ένα περιορισμό στο μέγιστο της χωρητικότητας δίσκων που μπορούμε να έχουμε στο σύστημά μας. Οι εξελίξεις όμως στην τεχνολογία δίσκων (φθηνοί και μεγάλοι δίσκοι πάνω από 1GB) έχουν περιορίσει σημαντικά το πρόβλημα.
- Το πρόβλημα της μέγιστης επιτρεπόμενης χωρητικότητας δίσκων περιορίζεται ακόμα περισσότερο με το νέο πρότυπο EIDE (Extended IDE) που αποτελεί εξέλιξη του προηγούμενου και μας επιτρέπει να συνδέσουμε δύο ζευγάρια δίσκων.

9 ΥΠΟΣΥΣΤΗΜΑ ΟΘΟΝΗΣ

Η Οθόνη και η Κάρτα της

Και τώρα θα μιλήσουμε για τις οθόνες. Το βασικό μηχάνημα εξόδου του Υ. Πόσοι δεν έχουν παρακινηθεί να μπουν στη περιπέτεια της πληροφορικής παρακινημένοι από τις νεκρές φύσεις με τα ζωντανά χρώματα και τις εξωτικές καλλονές που φιγουράρουν στις οθόνες των μαγαζιών της Στουρνάρα. Και από μέσα τα τσιπς και τα bits και τα bytes με το αδιάκοπο πήγαιν΄ έλα ...

Αρχές λειτουργίας της απεικόνισης

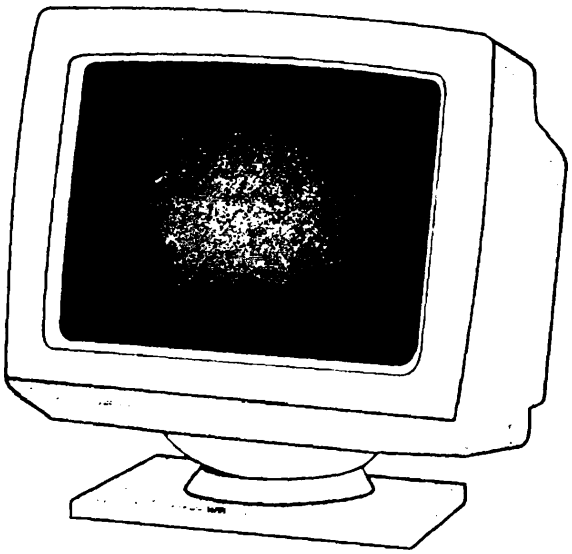
Πώς ο χρήστης λέει στον Υ τι θέλει να σχεδιάσει. Ο χρήστης εισάγει κατάλληλα στον Υ το σχήμα που θέλει (π.χ. ένα έτοιμο σχήμα από το χαρτί με τη βοήθεια scanner μηχάνημα που θα δούμε στο επόμενο κεφάλαιο, είτε σχεδιάζοντας στην οθόνη το σχήμα που θέλει) στη συνέχεια με τη βοήθεια κατάλληλου προγράμματος το σχήμα αυτό ψηφιοποιείται, δηλ φτιάχνεται το ψηφιδωτό του στη μνήμη με bits 0 και 1 (το 1 αντιστοιχεί σε ψηφίδα). Με βάση το ψηφιδωτό αυτό ο Ε στέλνει οδηγίες στο μηχανισμό σάρωσης της οθόνης, ώστε να αποδώσει τα 1 με φωτισμένα σημεία στις κατάλληλες θέσεις της οθόνης.

Από τι αποτελείται

Το Υποσύστημα Οθόνης αποτελείται από την Οθόνη και την Κάρτα προσαρμογής της ή adapter card.

Οθόνη - Monitor

Οπως παρατηρούμε εκ πρώτης όψεως μία οθόνη PC μοιάζει με συσκευή τηλεόρασης (Εικόνα 9.1). Η αρχή λειτουργίας της είναι η ίδια. Βασίζεται στην αρχή του λεγόμενου καθοδικού σωλήνα ή CRT - Cathode Ray Tube. Υπάρχει ένα κανόνι (εκτοξευτής) ηλεκτρονίων που παράγει δέσμη ηλεκτρονίων και σαρώνει μία οθόνη καλυμμένη με φωσφόρο από αριστερά προς τα δεξιά και από πάνω προς τα κάτω. Κατά τη σάρωση το "κανόνι" βομβαρδίζει την οθόνη με ηλεκτρόνια (τα οποία προσπίπτοντα στο φώσφορο της οθόνης σχηματίζουν φωτεινά σημεία) στα σημεία που πρέπει για να σχηματισθεί το επιθυμητό σχήμα.
Η σάρωση και ο βομβαρδισμός από τη δέσμη, στα κατάλληλα σημεία της οθόνης ελέγχονται από ένα ειδικό μηχανισμό, που είναι τοποθετημένος στη πορεία της δέσμης και δημιουργεί το κατάλληλο ηλεκτρονικό πεδίο.
Με τον τρόπο αυτό αποτυπώνουμε οποιοδήποτε σχήμα στην οθόνη είτε εικόνα είναι αυτό είτε κείμενο, αν σκεφθούμε πως κάθε χαρακτήρας μπορεί να εκληφθεί σα σχήμα που δημιουργείται με τον φωτισμό των κατάλληλων (γειτονικών) σημείων.



Εικόνα 9.1. Οθόνη Η/Υ

Πώς έχουμε **χρώμα** στις απεικονίσεις μας;
Η θεωρία λέει ότι υπάρχουν τρία βασικά χρώματα κόκκινο-πράσινο-μπλέ (RGB-red, green, blue) και κάθε χρωματική απόχρωση (από τις άπειρες που υπάρχουν) είναι αποτέλεσμα της μίξης των τριών αυτών χρωμάτων στις σωστές αναλογίες. Ο αριθμός των χρωματικών αποχρώσεων σε μία

οθόνη ποικίλλει από 16 χρώματα μέχρι πολλά εκατομμύρια ή και δισεκατομμύρια ακόμα χρώματα. Για συνηθισμένες γραφικές εφαρμογές αρκούν οι σχετικά φτηνές έγχρωμες οθόνες των 256 χρωμάτων. Για επαγγελματικές όμως εφαρμογές σχεδίασης, εκδόσεων (π.χ. εξώφυλλων περιοδικών) και γενικά εφαρμογές CAD/CAM (Computer Aided Design/Computer Aided Manufacturing) όπως αποκαλούνται, απαιτούνται πανάκριβες οθόνες έγχρωμης φωτογραφικής απεικόνισης (με δυνατότητες τουλάχιστον 16 εκατομμυρίων χρωμάτων). Η σύγχρονη τεχνολογία έχει δώσει οθόνες και 4 δισεκατομμυρίων χρωμάτων.

Στην περίπτωση έγχρωμων οθονών υπάρχουν τρεις μονάδες εκπομπής - από μία για τα βασικά χρώματα κόκκινο-πράσινο-μπλέ (RGB-red, green, blue). Κάθε χρωματική απόχρωση από τις άπειρες που υπάρχουν είναι αποτέλεσμα της μίξης των τριών αυτών χρωμάτων στις σωστές αναλογίες.

Εικονοστοιχεία - pixels

Το πλήθος των σημείων πάνω στην οθόνη στα οποία μπορεί να προσπέσει η ηλεκτρονική δέσμη εξαρτώνται από την λεπτότητα της δέσμης και την ακρίβεια μετατόπισης στα διαδοχικά σημεία κατά τη σάρωση. Ονομάζονται φωτεινά σημεία ή εικονοστοιχεία ή με τον διεθνή όρο pixels, παραφθορά από το picture elements. Όσο πιο πολλά είναι τα pixels, τόσο πιο ακριβής είναι η αναπαράσταση μιας εικόνας, αλλά και πιο ακριβή η οθόνη.

Χαρακτηριστικά Οθόνης

Μέγεθος οθόνης. Το μέγεθος της οθόνης δίνεται σε ίντσες και μετράει τη διαγώνιο της οθόνης. Υπάρχουν οθόνες μικρές 11" ή 12", μεσαίες 14" - 17" και μεγάλες 19"- 21".

Ανάλυση - Είδη. Ο αριθμός των pixel της οθόνης. Συνήθως εμφανίζεται σαν το γινόμενο δύο αριθμών, ο πρώτος των οποίων δίνει τον αριθμό των pixel σε κάθε οριζόντια γραμμή και ο δεύτερος των αριθμό των γραμμών στην οθόνη π.χ. 640x480 σημαίνει ότι στην υπόψη οθόνη διακρίνουμε 640 pixels σε κάθε οριζόντια γραμμή επί 480 γραμμές δηλ. 307.200 pixels. Συνήθως οι διαστάσεις (και αντίστοιχα ο αριθμός των pixels) οριζόντιας προς κατακόρυφη πλευρά μιας οθόνης ακολουθεί το πρότυπο 4/3.

Υπάρχει στενή σχέση μεταξύ ανάλυσης και διαστάσεων της οθόνης. Για κάθε διάσταση οθόνης υπάρχει και η ανάλυση που δίνει τα καλύτερα

αποτελέσματα. Μια μικρή οθόνη δεν μπορεί να αποδώσει καλά υψηλές αναλύσεις, γιατί η πυκνότητα θα είναι πολύ μεγάλη και τα στοιχεία πολύ μικρά. Για οθόνες 14-15" η σωστή ανάλυση είναι 800X600, ενώ καλή είναι και η κάπως μεγαλύτερη 1024X768.
Κάθε οθόνη έχει μία μέγιστη δυνατότητα ανάλυσης, η αξιοποίησή της όμως εξαρτάται από τις δυνατότητες της Κάρτας της, όπως θα δούμε παρακάτω.

Κάρτα Προσαρμογής - Adapter Card

Εξηγώντας τη διαδικασία απεικόνισης στην αρχή του κεφαλαίου είπαμε πως το ψηφιδωτό της επιθυμητής εικόνας δημιουργείται στη μνήμη και στη συνέχεια ο Ε το προωθεί για απεικόνιση στην οθόνη. Κάτι τέτοιο είναι ιδιαίτερα χρονοβόρο για τον Ε, ο οποίος έχει και άλλες δουλειές να κάνει αλλά και τη μνήμη επιβαρύνει. Για το λόγο αυτό θεωρήθηκε σκόπιμο να αποκεντρωθεί η διαδικασία αυτή και να ανατεθεί σε ένα ειδικό υποσταθμό απεικόνισης. Ο υποσταθμός αυτός θα έχει ειδική μνήμη για την καταχώρηση του ψηφιδωτού (Video RAM - VRAM) και ειδικό επεξεργαστή απεικόνισης (Video Controller) για την προώθησή του για εμφάνιση στην οθόνη. Ο υποσταθμός αυτός λέγεται κάρτα προσαρμογής (adapter card). Η ποιότητα και η ταχύτητα της απεικόνισης εξαρτάται άμεσα από την ισχύ της Κάρτας προσαρμογής (ισχυρός επεξεργαστής και γρήγορη και μεγάλη VRAM).

Υπάρχουν διάφορα πρότυπα καρτών. Πιο διαδεδομένη είναι η λεγόμενη Super VGA (Video Graphics Array) με video RAM των 1 ή 2MB για 16-μπιτη οθόνη (δυνατότητα απεικόνισης 64 Κ χρωμάτων) και διακριτική ικανότητα 1024x768.

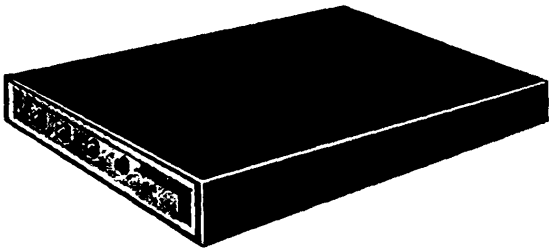
10 ΥΠΟΣΥΣΤΗΜΑ ΑΡΓΩΝ ΠΕΡΙΦΕΡΕΙΑΚΩΝ

Περιφερειακά Εισόδου / Εξόδου
Input / Output (I/O)

Από τι αποτελείται

Το Υποσύστημα των αργών Περιφερειακών περιλαμβάνει:

- χαμηλής ταχύτητας περιφερειακά εισόδου (πληκτρολόγιο, ποντίκι) (Εικον.10.2,3),
- εξόδου (εκτυπωτές) (Εικον. 10.4,5),
- σύνδεσης του Υ με μακρινά δίκτυα (modem) (Εικον.10.1).
- πόρτες σειριακές και παράλληλες στις οποίες συνδέονται τα περιφερειακά αυτά. Οι πόρτες αυτές προβλέπονται σαν στάνταρ σε κάθε PC και βρίσκονται στο πίσω μέρος του κουτιού. Τα κυκλώματα για τις σειριακές και παράλληλες αυτές διασυνδέσεις είναι ενσωματωμένα στη μητρική πλακέτα. Στις παράλληλες πόρτες συνδέονται τα πιο γρήγορα από τα Π αυτά (π.χ. εκτυπωτές), ενώ στις σειριακές τα πιο αργά (π.χ. πληκτρολόγια, ποντίκι, μόντεμ). Επειδή όμως η σειριακή διασύνδεση είναι και η πιο τυποποιημένη και διαδεδομένη για το λόγο αυτό και πολλά από τα "παράλληλα" Π έχουν δυνατότητα και σειριακής διασύνδεσης.



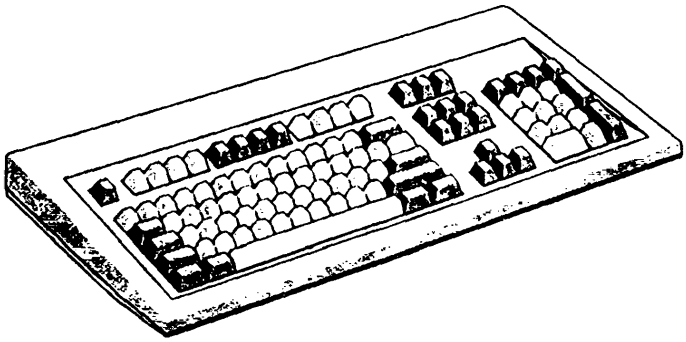
Εικόνα 10.1 Modem

Εδώ θα πρέπει να σημειώσουμε πως επειδή τα Π αυτά συνδέονται απευθείας (μέσω της πόρτας τους) με τον Υ, έχουν το μηχανισμό ελέγχου τους ενσωματωμένο με αυτά. Ετσι έχουμε στο εσωτερικό τους ειδικές κάρτες με επεξεργαστή και μνήμη που σε πολλές περιπτώσεις (π.χ. εκτυπωτές laser) είναι ιδιαίτερα ισχυροί.

Περιφερειακά Εισόδου

Πληκτρολόγιο

Το Πληκτρολόγιο είναι το καθιερωμένο μέσο εισαγωγής στοιχείων στο PC.



Εικόνα 10.2. Πληκτρολόγιο

Πώς λειτουργεί;

Κάτω από την επιφάνεια των πλήκτρων σε ένα Πληκτρολόγιο υπάρχει ένα ολόκληρο δίκτυο οριζόντιων και κάθετων ηλεκτρικών αγωγών, έτσι ώστε κάθε πλήκτρο να είναι τοποθετημένο πάνω από μία διασταύρωση αγωγών

(οι οποίοι και προσδιορίζουν το πλήκτρο). Το πάτημα ενός πλήκτρου ενεργοποιεί ηλεκτρικά τους αντίστοιχους αγωγούς. Η εξακρίβωση του πλήκτρου που πατήθηκε γίνεται με τη βοήθεια ενός ειδικού μικροεπεξεργαστή, που υπάρχει μέσα στο πληκτρολόγιο και ο οποίος σαρώνει τους αγωγούς συνεχώς και επισημαίνει τους ενεργοποιημένους οπότε και προσδιορίζει τη θέση του πλήκτρου και δίνει τον αύξοντα αριθμό του πλήκτρου. Εδώ πρέπει να πούμε πως όλα τα πλήκτρα είναι αριθμημένα από το σύστημα και κάθε πλήκτρο έχει το δικό του εσωτερικό αύξοντα αριθμό (scan code).

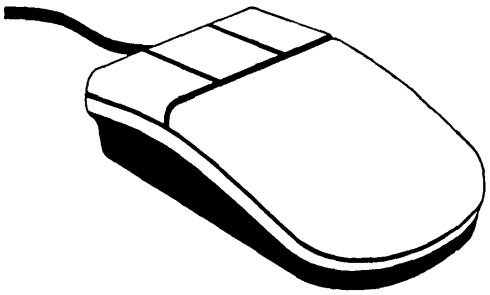
Έχοντας προσδιορίσει το πλήκτρο, εκείνο που απομένει είναι να δοθεί η σημασία του (γράμμα, αριθμός, ή άλλο ειδικό σύμβολο) δηλ το σύμβολο που είναι γραμμένο στο αντίστοιχο πλήκτρο στο πληκτρολόγιο. Αυτό επιτυγχάνεται με τη βοήθεια ενός ειδικού πίνακα που περιέχει για κάθε πλήκτρο τον αύξοντα αριθμό του και τη σημασία του. Με τον πίνακα αυτόν μπορούμε να δώσουμε όποια σημασία θέλουμε στα πλήκτρα.

Αυτή η σημασία του πλήκτρου που προκύπτει από τον πίνακα μετά το πάτημα κάθε πλήκτρου, κατάλληλα κωδικοποιημένη σαν συρμός από 8 bits (δηλ ένα byte) τοποθετείται στη Μνήμη του PC μας.

Παράλληλα με την τοποθέτηση στη Μνήμη ο Υ στέλνει το χαρακτήρα και στην οθόνη του PC ώστε να ελέγχει ο χειριστής την ορθότητα της πληκτρολόγησης. Η εμφάνιση αυτή στην οθόνη των πληκτρολογούμενων χαρακτήρων ονομάζεται ηχώ - echo.

Ποντίκι

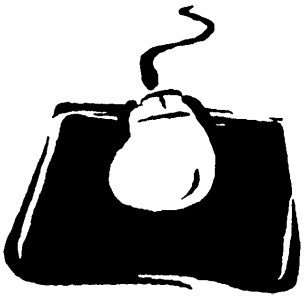
Με τις τελευταίες εξελίξεις των γραφικών παραθυρικών φιλικών ΛΣ, το ποντίκι έγινε απαραίτητο εργαλείο της φιλικής επικοινωνίας χρήστη - Υ με το οποίο ο χρήστης επιλέγει στοιχεία επί της οθόνης (Εικόνα 10.3).



Εικόνα 10.3. Ποντίκι

Πώς λειτουργεί;

Τα απλά μηχανικά ποντίκια αποτελούνται από ένα πλαστικό κουτί σαν πλάκα σαπουνιού μέσα στην οποία είναι μια μπάλα. Όταν το ποντίκι κινείται σε μία επίπεδη επιφάνεια η μπάλα γυρίζει και τα κυκλώματα μέσα υπολογίζουν τη κίνηση. Ο ελεγκτής του Π παρακολουθεί τη κίνηση και παρέχει πληροφορίες για την εκάστοτε θέση του Π η οποία απεικονίζεται στην οθόνη με τη μορφή απλής κουκίδας ή τόξου κλπ. Τα Π διαθέτουν και τουλάχιστο ένα κουμπί που όταν πατηθεί λέει στο πρόγραμμα με το οποίο συνεργάζεται το σημείο στο οποίο δείχνει και το πρόγραμμα προβαίνει αντίστοιχα σε συγκεκριμένη προβλεπόμενη ενέργεια. Το Π είναι το πιο κατάλληλο μηχάνημα για την επιλογή ενεργειών από μενού και τη μεταφορά αντικειμένων στην οθόνη.

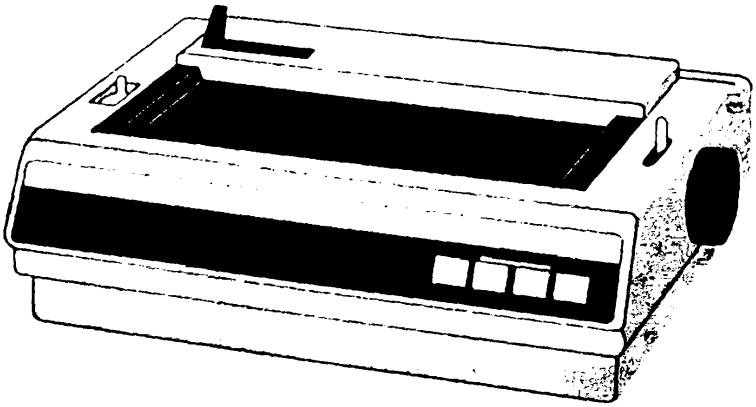


Περιφερειακά Εξόδου

Εκτυπωτές

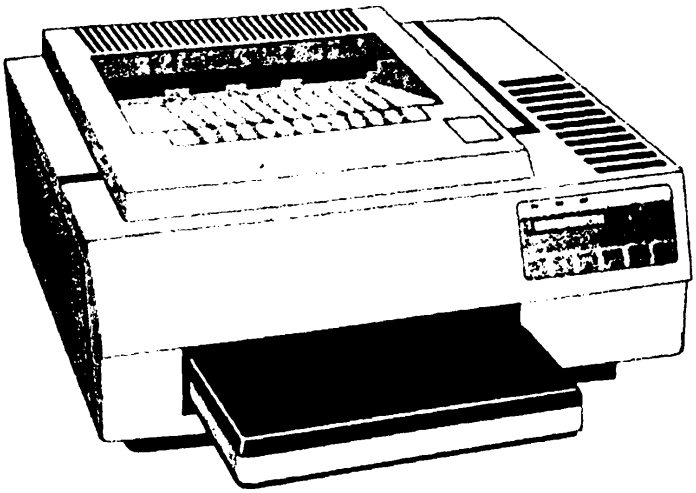
Οι εκτυπωτές των PC διακρίνονται σε

- **εκτυπωτές γραμμής** που τυπώνουν μια γραμμή τη φορά. Σε αυτούς περιλαμβάνονται οι *κρουστικοί* (dot matrix), και οι *ψεκασμού* (ink jet) (Εικόνα 10.4) και



Εικόνα 10.4. Εκτυπωτής γραμμής.

- **εκτυπωτές σελίδας** που μαζεύουν στη μνήμη τους τα στοιχεία ολόκληρης της σελίδας και την τυπώνουν μαζεμένη (laser printers) (Εικόνα 10.5).



Εικόνα 10.5 Εκτυπωτής σελίδας.

Εκτυπωτές γραμμής

Κρουστικοί Εκτυπωτές

Οι κρουστικοί εκτυπωτές ή dot matrix είναι εξελιγμένες γραφομηχανές άνευ δακτυλογράφου. Τη θέση του δακτυλογράφου έχει το κατάλληλο πρόγραμμα που είναι μέσα στον Υ.
Οι εκτυπωτές αυτοί τυπώνουν με την βοήθεια μιας κεφαλής που αποτελείται από ένα πλέγμα ακίδων. Εκτύπωση ενός γράμματος γίνεται με την κατάλληλη ενεργοποίηση εκείνων των ακίδων που αντιστοιχούν στο σχήμα του υπόψη γράμματος και το κτύπημα στο χαρτί μέσω μιας μελανοταινίας.

Χαρακτηριστικά:
- *αριθμός ακίδων* που προσδιορίζει και την ποιότητα της εκτύπωσης. Κυκλοφορούν κυρίως δύο κατηγορίες οι φτηνοί των 9 ακίδων και οι ποιοτικοί των 24 ακίδων. Στους φτηνούς υπάρχει συνήθως η δυνατότητα διπλής εκτύπωσης, δηλ κάθε γραμμή να τυπώνεται δυο φορές με ελαφρά μετατοπισμένη τη θέση των ακίδων τη δεύτερη φορά, ώστε να τυπώνουν στα ενδιάμεσα της πρώτης φοράς. Η δυνατότητα αυτή συνήθως αποδίδεται με τον όρο NLQ- Near Letter Quality.
- *Ταχύτητα* που μετριέται σε cps - characters per sec.
- *Μέγεθος* μικροί (για χαρτί Α4) και μεγάλοι για μεγάλο μηχ/κό χαρτί (Α3)

Πλεονεκτήματα. Χαμηλή τιμή, εκτύπωση γραφικών, εκτύπωση πολλαπλών αντιτύπων απαραίτητο για λογιστήρια.
Μειονεκτήματα: Αυξημένος θόρυβος και το περιορισμένο χρώμα.

Εκτυπωτές Ψεκασμού - Ink jet

Η αρχή λειτουργίας μοιάζει με των κρουστικών η διαφορά είναι πως στη θέση της κεφαλής με τις ακίδες έχουμε ένα κινητό δοχείο μελάνης με έναν αριθμό από μικρές οπές στη θέση των ακίδων. Η εκτύπωση γίνεται με την κατάλληλη ενεργοποίηση έγχυσης μελάνης πάνω στο χαρτί από συγκεκριμένες οπές, ανάλογα με το σχήμα του προς εκτύπωση χαρακτήρα. Χρησιμοποιώντας κατάλληλα μελανοδοχεία με τα τρία βασικά χρώματα (ή 4) τα οποία προκειμένου για την εκτύπωση είναι, και κατάλληλες τεχνικές μίξης τους μπορούμε να έχουμε εντυπωσιακές έγχρωμες εκτυπώσεις.

Χαρακτηριστικά

- *ποιότητα εκτύπωσης,* μετριέται σε αριθμό κουκκίδων ανά ίντσα (dpi). Σήμερα διατίθενται στην αγορά φθηνοί εκτυπωτές με 360 αλλά και 720 dpi.
- *ταχύτητα*

Πλεονεκτήματα: Ποιοτική εκτύπωση, πολύ καλό χρώμα και φθηνή τιμή. Κάποια προσοχή πρέπει να δίνεται στη ποιότητα του χαρτιού για να μην απλώνει το μελάνι.
Μειονεκτήματα: Δεν μπορεί να δώσει πολλαπλά αντίγραφα (χρήσιμο για λογιστήρια κλπ).

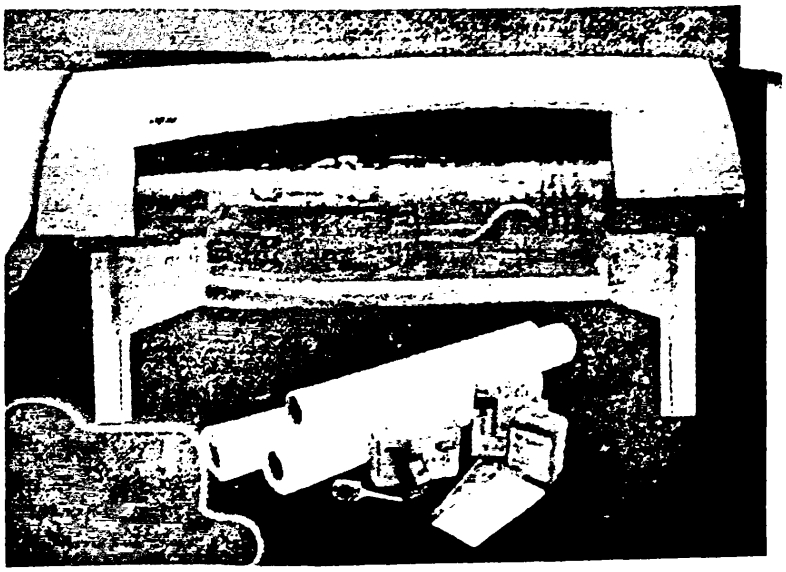
Εκτυπωτές Σελίδων

Laser Printers

Τρόπος λειτουργίας : Οι εκτυπωτές αυτοί είναι ουσιαστικά εξειδικευμένοι υπολογιστές από μόνοι τους, με δικούς τους επεξεργαστές (16-μπιτους, 32-μπιτους) και μνήμη πολλών MBytes. Εχουν δική τους γλώσσα περιγραφής των περιεχομένων μιας σελίδας (η πιο γνωστή είναι η Postscript).
Η προς εκτύπωση σελίδα μεταφέρεται στη Μνήμη του Εκτυπωτή. Όχι σαν αλληλουχία χαρακτήρων αλλά η εικόνα της bit προς bit όπως ήταν στην οθόνη. Για την εκτύπωση από τη Μ του Ε χρησιμοποιείται η ξηρογραφική μέθοδος :
- ανάλογα με τα περιεχόμενα της σελίδας στη μνήμη του Εκτυπωτή ενεργοποιείται κατάλληλα ακτίνα laser που προσπίπτει σε ένα περιστρεφόμενο τύμπανο καλυμμένο με φωτοαγώγιμη ουσία η οποία έχει την ιδιότητα αν φωτιστεί με ακτίνα laser να δημιουργεί (στο σημείο φωτισμού) ηλεκτρικό φορτίο. Με το τρόπο αυτό αποτυπώνονται πάνω στο τύμπανο τα περιεχόμενα μιας ολόκληρης σελίδας.
- αμέσως μετά επιρρίπτεται μελανόσκονη πάνω στην ηλεκτρικά σχηματισμένη εικόνα του τυμπάνου. Η μελανόσκονη έχει την ιδιότητα να επικάθεται πάνω στα φορτισμένα σημεία δημιουργώντας έτσι την εικόνα σε μελάνη που στη συνέχεια αποτυπώνεται σε χαρτί.

Χαρακτηριστικά στοιχεία των εκτυπωτών αυτών είναι :
- *η διακριτική ικανότητα εκτύπωσης*, που μετριέται σε σημεία ανά ίντσα - dots per inch - dpi
- η *ταχύτητα* λειτουργίας σε σελίδες ανά λεπτό. Οι ισχυροί laser για μεγάλους Υ μπορούν να πετύχουν ταχύτητες εκτύπωσης από 12 μέχρι 120 σελίδες το λεπτό (εκφρασμένη σε γραμμές ανά λεπτό η ταχύτητα αυτή μπορεί να δώσει γύρω στις 10000 LPM). Στα μέσα της δεκαετίας του 80 κυκλοφόρησαν φθηνότερες εκδόσεις των laser printers με σημαντικά μικρότερες ταχύτητες, κατάλληλες για σύνδεση με micros. Σήμερα δε αποτελούν αναπόσπαστο μέρος του εξοπλισμού των επιτραπέζιων εκδοτικών συστημάτων (Desk Top Publishing-DTP).

Plotters ή σχεδιαστικά μηχανήματα.
Τα μηχανήματα αυτά χρησιμοποιούνται για την εκτύπωση κυρίως σχεδίων. Τα χρησιμοποιούν κατά κύριο λόγο οι πολιτικοί μηχανικοί, τοπογράφοι, αρχιτέκτονες, χημικοί σχεδιαστές.



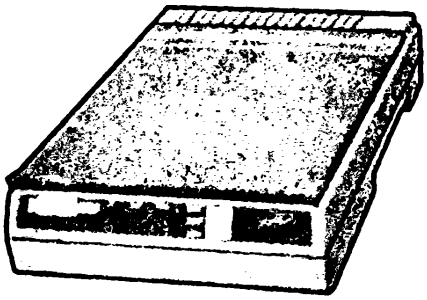
Εικόνα 10.6 Plotters.

Λοιπά Περιφερειακά

Μονάδες Αναγνώρισης Εντύπων Χαρακτήρων - Scanners

Τα μηχανήματα αυτά επιτρέπουν την απευθείας εισαγωγή στο PC σχεδίων, φωτογραφιών κλπ αλλά και κειμένων (Εικόνα 10.6). Για την αναγνώριση των χαρακτήρων των κειμένων αυτών και την καταχώρησή τους στη μνήμη του Υ χρησιμοποιούνται διάφορες τεχνικές με πιο γνωστή την τεχνική OCR (Optical Character Recognition). Στη περίπτωση αυτή εκτός από τον scanner απαιτείται και ένα ειδικό πρόγραμμα που μετατρέπει τους τυπωμένους χαρακτήρες σε μηχανογραφικούς (ASCII).
Η όλη διαδικασία είναι απλή και μοιάζει με φωτοτύπηση, με τον scanner στη θέση του φωτοτυπικού μηχανήματος.
Και για μεν τα σχήματα τα αποτελέσματα είναι πολύ ικανοποιητικά. Για τα κείμενα όμως υπάρχουν πολλά τεχνικά προβλήματα και τα αποτελέσματα εξαρτώνται από την καθαρότητα των χαρακτήρων την ποιότητα του χαρτιού στο οποίο είναι γραμμένα κλπ. Για να ξεπεραστούν τα προβλήματα αυτά, απαιτούνται πολύ ισχυροί scanners αλλά και προγράμματα.

scanner χειρός



scanner

Εικόνα 10.7. Scanners.

Πάντως τη στιγμή αυτή κυκλοφορούν στην αγορά πολλοί και φθηνοί σκάνερς με πολύ καλά αποτελέσματα στην εισαγωγή σχεδίων, και μέτρια στην εισαγωγή κειμένων. Μάλιστα πρόσφατα κυκλοφόρησαν στην αγορά και μικροί scanners ... χειρός. Ο χρήστης κρατάει τον scanner του, τον τοποθετεί πάνω στο κείμενο και τον μετακινεί κατάλληλα ώστε να σαρώσει την επιθυμητή περιοχή.
Ένα παρεμφερές μηχάνημα είναι ο digitizer για την εισαγωγή χαρτών κλπ. σε ψηφιακή μορφή με τη βοήθεια των συντεταγμένων του.

11 ΤΟ ΛΟΓΙΣΜΙΚΟ ΤΟΥ ΥΠΟΛΟΓΙΣΤΗ

Αρχεία και Προγράμματα

Στις προηγούμενες παραγράφους εξετάσαμε συνοπτικά τον εξοπλισμό ή hardware του υπολογιστή. Για να χρησιμοποιήσουμε όμως τον εξοπλισμό αυτό, για τη μηχανογραφική διεκπεραίωση μιας διαδικασίας (εφαρμογής) όπως είδαμε και στο Κεφ.1 είναι απαραίτητα τα δεδομένα που θα επεξεργαστούμε και οι οδηγίες για το πώς θα τα επεξεργαστούμε.

Οδηγίες και δεδομένα αποτελούν το άλλο μεγάλο μέρος το *μαλακό* (soft) ενός μηχανογραφικού συστήματος σε αντιδιαστολή με το *σκληρό* (hard) του εξοπλισμού.
Οι οδηγίες και τα δεδομένα αποτελούν την πρώτη ύλη για τη διεκπεραίωση μιας εφαρμογής και προσδιορίζουν μπορούμε να πούμε την εφαρμογή. Τα στοιχεία αυτά τοποθετούνται μέσα στη μνήμη του Υ και στη συνέχεια διακινούνται μέσα από τις διάφορες μονάδες του εξοπλισμού και κυρίως μεταξύ μνήμης και δίσκων όπως είδαμε στο κεφάλαιο 5 (βλ Σχ. 5.2) μέχρι να ολοκληρωθεί η επεξεργασία τους και να προκύψουν τα αποτελέσματα.

Αρχεία

Και μέχρι εδώ καλά. Δεδομένα και οδηγίες πρέπει να εισάγονται στον Υ και να διακινούνται ανάμεσα στις μονάδες του προκειμένου να γίνουν οι επεξεργασίες. Το ερώτημα τώρα είναι : πώς υφίστανται στον Υ και πώς διακινούνται; Χύμα, σκόρπια ή συγκροτημένα, ομαδοποιημένα, οργανωμένα; Η απάντηση είναι εύκολη. Ο,τι γίνεται στον Υ και τη μηχανογράφηση γενικότερα πρέπει να γίνεται συστηματικά και οργανωμένα.

1. Πάρε τα βασικά στοιχεία ενός υπαλλήλου
2. Πρόσθεσε τον βασικό του μισθό με τα επιδόματά του
3. Αφαίρεσε τις κρατήσεις του από το παραπάνω άθροισμα. Το αποτέλεσμα είναι το καθαρό πληρωτέο
4. Γράψε στην κατάσταση πληρωμής το ονοματεπώνυμο του υπόψη υπαλλήλου μαζί με το παραπάνω αποτέλεσμα
5. Ξανάρχισε από το βήμα 1 για τον επόμενο υπάλληλο και συνέχισε μέχρι να τελειώσουν όλοι οι υπάλληλοι της εταιρίας.

Οδηγίες σε φυσική γλώσσα

```
10 READ ONOMA$, BASMISQOS,
        EPIDOMA, KRATHSEIS
20 IF ONOMA$ = "" THEN
                GOTO 60
30 PLHRWTEO = BASMISQOS +
        EPIDOMA - KRATHSEIS
40 PRINT ONOMA$, PLHRWTEO
50 GOTO 10
60 END
```

Πρόγραμμα σε BASIC

Σχ. 11.1 Αρχεία Προγραμμάτων

Ονοματεπώνυμο	Βασ.Μισθός	Επίδομα	Κρατήσεις
ΚΩΝΣΤΑΝΤΙΝΙΔΗΣ ΣΑΚΗΣ	100000	60000	50000
ΒΟΥΓΙΟΥΚΛΗΣ ΓΙΩΡΓΟΣ	80000	50000	40000
ΚΟΡΡΕ ΑΝΝΑ	60000	40000	30000
ΜΥΡΤΙΔΗΣ ΔΙΟΝΥΣΗΣ	70000	45000	30000

Σχ. 11.2 Αρχεία Δεδομένων

Αγαπητοί/ές Αναγνώστες/ριες,

Ενα απλό κείμενο που αποτελείται από ένα σύνολο λέξεων και παραγράφων λέμε πως αποτελεί ένα αρχείο κειμένου. Τα αρχεία αυτά χαρακτηρίζονται από χαμηλό βαθμό οργάνωσης και αποτελούν την βάση των λεγόμενων εφαρμογών επεξεργασίας κειμένου.

Σχ. 11.3 Αρχεία κειμένου

Είδαμε στο προηγούμενο κεφάλαιο πως αν θέλουμε να διασώσουμε οδηγίες και δεδομένα που βρίσκονται στη μνήμη, πρέπει να τα τοποθετήσουμε στο δίσκο. Πώς όμως πρέπει να τα τοποθετήσουμε προκειμένου να τα ξαναβρούμε όταν τα χρειαστούμε; Αν τα βάλουμε χύμα ασφαλώς θα τα χάσουμε. Προκύπτει λοιπόν η ανάγκη μιας συγκροτημένης διακίνησης.

Για το λόγο αυτό οδηγίες και δεδομένα είναι οργανωμένα σε ενότητες ομοειδών στοιχείων που λέγονται αρχεία. Κάθε αρχείο έχει το όνομά του, που του το δίνει ο ίδιος ο χρήστης, όταν το δημιουργεί (π.χ. όταν εισάγει τα αντίστοιχα στοιχεία στον Υ από το πληκτρολόγιο). Το αρχείο αναγνωρίζεται από το όνομά του οπουδήποτε και αν βρίσκεται μέσα στον Υ.
Ανάλογα με το περιεχόμενό τους υπάρχουν αρχεία οδηγιών (προγράμματα) και αρχεία δεδομένων. Για παράδειγμα:
- ένα σύνολο οδηγιών (πρόγραμμα) μισθοδοσίας ενός οργανισμού αποτελεί ένα αρχείο προγράμματος με το όνομα έστω MISQOD. Το σύνολο των οδηγιών που είδαμε στο προηγούμενο κεφάλαιο (Σχ. 11.1) αποτελεί ένα πρόγραμμα ή ένα αρχείο (προγράμματος).
- Ενα σύνολο μισθοδοτικών δεδομένων μιας εταιρείας που περιλαμβάνουν το Ον/μο του κάθε υπαλλήλου, τον Βασ. Μισθό, τα επιδόματα και τις κρατήσεις (με τη σειρά αυτή) αποτελούν ένα αρχείο δεδομένων με το όνομα έστω MISQDATA (Σχ. 11.2).

Οταν λοιπόν ζητάμε να μεταφερθεί ένα αρχείο μεταξύ μνήμης και δίσκου ή να τυπωθεί (μεταφορά στον εκτυπωτή), δίνουμε το όνομά του και όλα τα περιεχόμενά του μεταφέρονται μαζί.

Αρχεία Δεδομένων

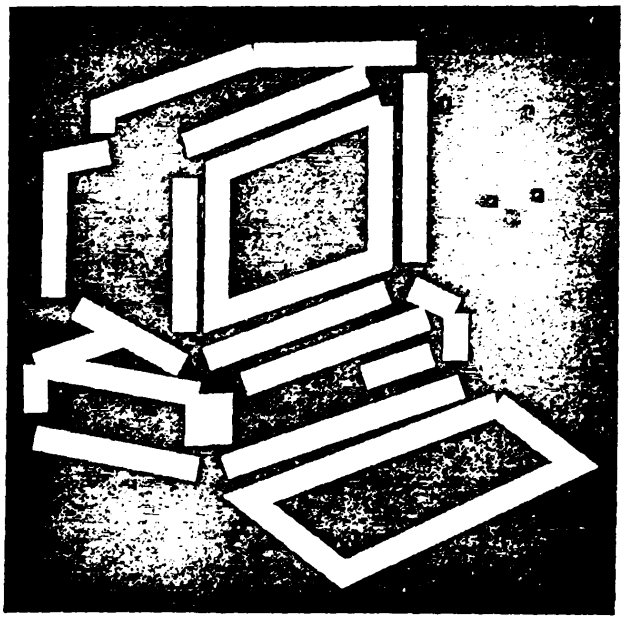
Εκτός από τη βασική διαφοροποίηση σε αρχεία προγραμμάτων και αρχεία δεδομένων υπάρχει διαφοροποίηση και μεταξύ των αρχείων δεδομένων ανάλογα με το είδος των δεδομένων και του βαθμού οργάνωσης των περιεχομένων τους. Έτσι έχουμε:

- **αρχεία** που περιέχουν **δεδομένα** για αριθμητικές επεξεργασίες (data processing), όπως στο παραπάνω παράδειγμα του αρχείου μισθοδοτικών δεδομένων από το οποίο ζητούμε να υπολογισθεί το καθαρό πληρωτέο για κάθε υπάλληλο, τα αρχεία αυτά λέγονται αρχείο δεδομένων (data processing files ή απλά data files). Χαρακτηρίζονται από υψηλό βαθμό οργάνωσης, δηλ κάθε αρχείο αποτελείται από εγγραφές της ίδιας δομής,

μία για κάθε υπάλληλο. Κάθε εγγραφή αποτελείται από τα συστατικά της μέρη που λέγονται πεδία. Τα πεδία είναι τοποθετημένα στην εγγραφή με προκαθορισμένη σειρά και το κάθε πεδίο πιάνει ορισμένο αριθμό χαρακτήρων. Στο σχ. 11.2 φαίνεται ένα τέτοιο αρχείο κάθε εγγραφή του οποίου αποτελείται από τα πεδία Ον/μο του κάθε υπαλλήλου, τον Βασ. Μισθό, τα επιδόματα και τις κρατήσεις.

- αρχεία που περιέχουν **κείμενα** (π.χ. ένα γράμμα ή έγγραφο), τότε έχουμε αρχεία κειμένων (text files). Χαρακτηρίζονται από χαμηλό βαθμό οργάνωσης. Τα περιεχόμενα είναι μια σειρά από χαρακτήρες κειμένου οργανωμένες απλά σε λέξεις και παραγράφους (βλ Σχ. 11.3).

- αρχεία που περιέχουν **εικόνες** (πχ ένα αρχιτεκτονικό σχέδιο, μία μακέτα γραφίστα κλπ), τότε έχουμε αρχεία εικόνων ή γραφικών όπως λέγονται (graphics files), κοκ. (Σχ. 11.4) .



Σχ.11.4

Ευρετήρια Δίσκων

Ολα τα αρχεία τελικά καταλήγουν στο δίσκο του συστήματος όπου και κρατούνται και από όπου φορτώνονται στη μνήμη όταν κληθούν για επεξεργασία. Κάθε δίσκος διαθέτει ένα κατάλογο - Ευρετήριο - Directory με όλα τα αρχεία που περιέχει. Ο χρήστης έχει πρόσβαση στον κατάλογο περιεχομένων των δίσκων του συστήματος, ώστε να γνωρίζει τι περιέχουν και να μπορεί να επιλέγει το αρχείο που τον ενδιαφέρει.

Δημιουργία Αρχείων και επεξεργασία τους - Πακέτα προγραμμάτων

Πως φτιάχνουμε αρχεία. Για παράδειγμα πώς φτιάχνουμε ένα γράμμα. Δυστυχώς δεν είναι τόσο απλό όσο φαίνεται, με την έννοια πως δεν μπορούμε στα καλά καθούμενα να αρχίσουμε να πατάμε πλήκτρα στο Π και να φτιάχνουμε αρχεία. Δεν είναι όμως και δύσκολο φτάνει να γνωρίζεις τη μεθοδολογία. Εκείνο που χρειάζεται είναι ένα (έτοιμο) πρόγραμμα που πρέπει να βρίσκεται στη μνήμη και θα μας βοηθήσει να δημιουργήσουμε το αρχείο που θέλουμε. Για κάθε κατηγορία αρχείων υπάρχουν αντίστοιχα προγράμματα. Για το γράμμα π.χ. υπάρχουν τα προγράμματα επεξεργασίας κειμένων.

Τα προγράμματα αυτά δεν περιορίζονται στο να μας βοηθήσουν να φτιάξουμε ένα αρχείο, αλλά μας βοηθούν να το επεξεργαστούμε (αλλάξουμε, τυπώσουμε κλπ).

Τα προγράμματα αυτά λέγονται πακέτα και διατίθενται σε μεγάλη ποικιλία στην αγορά.

Περισσότερα όμως στην αμέσως επόμενη παράγραφο.

Προγράμματα

Πρόγραμμα: Οπως είδαμε και στην εισαγωγή ένα πρόγραμμα δεν είναι τίποτα άλλο από ένα κείμενο οδηγιών προορισμένων να εκτελεστούν από τον Υ σε μια γλώσσα που καταλαβαίνει ο Υ. Μια τέτοια γλώσσα είναι η BASIC που θα εξετάσουμε στο Παράρτημα Α. Στο Σχ. 11.1 υπάρχει ένα πρόγραμμα BASIC.

Γλώσσα προγραμματισμού: Οι γλώσσες που καταλαβαίνουν οι Υ λέγονται Γλώσσες Προγραμματισμού. Κάθε ΓΠ χαρακτηρίζεται από το αλφάβητό της, τη γραμματική της και το συντακτικό της, όπως συμβαίνει και με τις φυσικές γλώσσες του ανθρώπου. Η διαφορά είναι πως οι ΓΠ δεν είναι τόσο πλούσιες σε λεξιλόγιο και ευέλικτες συντακτικά όσο οι φυσικές. Τις γλώσσες προγραμματισμού τις κατασκευάζουν οι άνθρωποι με χαρακτηριστικά τέτοια που να ευνοούν διάφορες κατηγορίες εφαρμογών. Για το λόγο αυτό υπάρχουν πολλές ΓΠ. Παρόμοια κατάσταση υπάρχει και με τις φυσικές γλώσσες του ανθρώπου. Λέγαμε παλιά η αγγλική γλώσσα είναι η γλώσσα του εμπορίου, η γαλλική η γλώσσα της διπλωματίας, η γερμανική του δικαίου (νομική) και της φιλοσοφίας εννοώντας πως οι γλώσσες αυτές εξελίχθηκαν και δημιουργήθηκαν από λαούς που είχαν την αντίστοιχη έφεση και έφτιαξαν γλώσσες - εργαλεία έκφρασης που να τους διευκολύνουν να εκφράσουν την έφεσή τους αυτή.
Αντίστοιχα με τις ΓΠ έχει επικρατήσει να θεωρείται η COBOL η γλώσσα για τις εμπορικές εφαρμογές, η FORTRAN και η PASCAL για τις επιστημονικές εφαρμογές κλπ.

Πρόγραμμα και γνώση. Όπως ο άνθρωπος ενεργεί με βάση τη γνώση που έχει για το αντίστοιχο θέμα, έτσι και όλες οι *ενέργειες*, ό,τι κάνει δηλ. ο Υ, γίνεται με βάση το αντίστοιχο πρόγραμμα. Το πρόγραμμα δηλ δίνει τη γνώση στον Υ για κάποιο θέμα και ο εξοπλισμός (hardware) την ικανότητα - δυνατότητα να την εφαρμόσει. Η διαφορά είναι πως ενώ για τον άνθρωπο η γνώση κατακτάται με κόπο και χρόνο η γνώση για τον Υ αποκτάται *ακαριαία* από τη στιγμή που το αντίστοιχο πρόγραμμα θα φορτωθεί στη μνήμη του (βέβαια τον μπελά θα τον τραβήξει πάλι ο άνθρωπος που θα χρειαστεί να γράψει το πρόγραμμα).

Για να μπορέσει λοιπόν να χρησιμοποιηθεί ο Υ σαν *υπάλληλος* μισθοδοσίας θα πρέπει να τοποθετηθεί στη μνήμη του και να εκτελεσθεί το

πρόγραμμα μισθοδοσίας όπως είδαμε στην εισαγωγή. Με τον ίδιο τρόπο ο Υ αποκτά γνώσεις λογιστή με το πρόγραμμα λογιστηρίου, αποθηκάριου, γραμματέως με τα αντίστοιχα προγράμματα γραμματειακής υποστήριξης κλπ. Έχετε δει στις μικρές επιχειρήσεις - προσωπικές που ένας άνθρωπος γνωρίζει ολόκληρο σχεδόν το κύκλωμα εργασιών. Κατεβαίνει στην αποθήκη *φοράει* το καπέλο του αποθηκάριου, ανεβαίνει στο λογιστήριο γίνεται λογιστής, πάει στη γραμματεία γράφει την αλληλογραφία κοκ. Έτσι και ο Υ σε μια επιχείρηση μπορεί να κάνει όλες τις δουλειές, αρκεί να έχουν τοποθετηθεί τα αντίστοιχα προγράμματα στη μνήμη. Το πρόγραμμα που εκτελείται κάθε φορά του δίνει και την αντίστοιχη ιδιότητα.

Είδη Προγραμμάτων: Όπως είδαμε και στο κεφ.4 υπάρχουν δύο βασικά είδη προγραμμάτων τα προγράμματα εσωτερικής λειτουργίας του Υ τα λεγόμενα Λειτουργικά Συστήματα και τα προγράμματα εφαρμογών.

Λειτουργικά Συστήματα

Το ΛΣ ελέγχει και αξιοποιεί τους πόρους (εξοπλισμό, αρχεία) του συστήματος και τα προσφέρει στους χρήστες για να κάνουν τη δουλειά τους. Για να διευκολύνει τους χρήστες να χρησιμοποιήσουν το σύστημα του Υ και να επικοινωνήσουν με τους πόρους του, παρέχει ένα σύνολο από βασικές υπηρεσίες. Τους δίνει πχ. τη δυνατότητα να δουν τα περιεχόμενα ενός δίσκου (να ανοίξουν το Ευρετήριό του), τα περιεχόμενα ενός αρχείου, την κατάσταση ενός περιφερειακού, να ξεκινήσουν την εκτέλεση ενός προγράμματος κλπ.

Επικοινωνία με το χρήστη - user interface. Οι υπηρεσίες του ΛΣ παρέχονται με τρόπο που να διευκολύνει τον χρήστη να τις χρησιμοποιήσει. Ο τρόπος αυτός επικοινωνίας χρήστη - ΛΣ συνήθως αποδίδεται με τον όρο user interface. Διάφορα ΛΣ χρησιμοποιούν διάφορους τρόπους επικοινωνίας. Μια κατηγορία ΛΣ χρησιμοποιεί τη μέθοδο των εντολών. Δηλ για κάθε υπηρεσία που παρείχαν υπήρχε ο αντίστοιχος τύπος εντολής που έπρεπε να χρησιμοποιήσει ο χρήστης (να γράψει με το πληκτρολόγιο) για να την καταλάβει το ΛΣ και να τη διεκπεραιώσει. Οι εντολές είχαν σαν βάση ένα αγγλικό ρήμα ενδεικτικό της αντίστοιχης υπηρεσίας (OPEN, CLOSE,

COPY, MOVE κλπ). Τα ΛΣ που επικοινωνούν με εντολές αποδίδονται με τον όρο CUI - Command User Interface.
Η πιο σύγχρονη αντίληψη απαιτεί ένα πιο φιλικό τρόπο επικοινωνίας. Να δίνει στο χρήστη μια παραστατική, γραφική, εικονογραφημένη" απεικόνιση του συστήματος του Υ και των πόρων του, καθώς και των δυνατών υπηρεσιών. Τα ΛΣ αυτά χαρακτηρίζονται σαν γραφικά και αποδίδονται με τον όρο GUI - Graphics User Interface. Για τη χρήση των συστημάτων αυτών δεν αρκεί το πληκτρολόγιο απαιτείται και το ποντίκι. Πρώτη η εταιρεία APPLE χρησιμοποίησε τέτοιο ΛΣ για τους Υ MACINTOSH. Σήμερα τα PC χρησιμοποιούν τα WINDOWS είτε το πιο σύγχρονο WINDOWS 95 είτε το παλιότερο αλλά ακόμα σε χρήση WINDOWS 3.1

Ταυτόχρονη εκτέλεση πολλών εργασιών - multitasking. Στην αρχή τα παλιότερα ΛΣ των PC (όπως το DOS) επέτρεπαν την εκτέλεση μιας μόνο εργασίας κάθε φορά. Χρησιμοποιούσες πχ. ένα πρόγραμμα επεξεργασίας λογιστικών φύλλων, του έδινες μια δύσκολη σειρά υπολογισμών να κάνει που μπορεί να κρατούσε 5 ή 10 λεπτά, έπρεπε να περιμένεις να τελειώσει πριν κάνεις οτιδήποτε άλλο με τον Υ σου. Τα πιο καινούρια ΛΣ σου επιτρέπουν να εκτελείς πολλές εργασίες ταυτόχρονα π.χ. ενώ δουλεύει το πρόγραμμα με τους υπολογισμούς σου, εσύ μπορείς να αρχίσεις να γράφεις ένα γράμμα σε ένα επεξεργαστή κειμένου και, ενώ αυτό τυπώνεται, να αρχίσεις ένα τρίτο κοκ. Τα ΛΣ αυτά χαρακτηρίζονται σαν πολυεργασιακά - multitasking. Χαρακτηριστικά παραδείγματα είναι το W95 αλλά και το W3.1.

Παράλληλη επεξεργασία. Οι πιο σύγχρονοι και ισχυροί υπολογιστές διαθέτουν περισσότερους από ένα επεξεργαστές επιτυγχάνοντας έτσι πραγματική ταυτόχρονη (παράλληλη) επεξεργασία των εφαρμογών τους (βλ. κεφ.13). Για να γίνει αυτό δυνατό απαιτούνται ιδιαίτερα ανεπτυγμένα ΛΣ όπως το UNIX, το WINDOWS NT κ.ά.

Πακέτα προγραμμάτων

Όπως θα θυμούνται οι παλιότεροι στα (όχι και τόσο) παλιά χρόνια ό,τι ήθελε να φορέσει ένας άνθρωπος (ρούχα, παπούτσια κλπ) γινόταν *επί μέτρω* όπως έλεγαν δηλ. ο αντίστοιχος μάστορας του έπαιρνε μέτρα και

του το έφτιαχνε στα μέτρα του. Τώρα βέβαια ήλθε η τυποποίηση και όλα αγοράζονται έτοιμα. Το επί μέτρω δεν έχει καταργηθεί ακόμα, αλλά στοιχίζει. Στα επί μέτρω καταλήγουν όσοι έχουν κάποιο ιδιαίτερο φυσικό πρόβλημα (και όσοι έχουν πολλά λεφτά και θέλουν να ξεχωρίζουν).

Κάτι αντίστοιχο έγινε και με τα προγράμματα. Παλιά κάθε επιχείρηση έβαζε τους προγραμματιστές της να φτιάχνουν προγράμματα για τις δικές της αποκλειστικά εφαρμογές. Στη συνέχεια όταν διαπιστώθηκε πως οι ανάγκες σε μερικές τουλάχιστον εφαρμογές είναι κοινές και μπορούν να γενικευθούν, τότε αναλάμβαναν κάποια γραφεία προγραμματισμού (software houses) να γράψουν τέτοια γενικευμένα προγράμματα που μπορούσαν να καλύψουν τις ανάγκες ενός μεγάλου φάσματος χρηστών και να τα πουλάνε.
Αυτά τα τυποποιημένα - γενικευμένα προγράμματα λέγονται πακέτα.
Πώς γράφεται ένα πακέτο; Αφού επιλεγεί το είδος της εφαρμογής που μπορεί να τυποποιηθεί, κάθεται ο προγραμματιστής και γράφει το πρόγραμμα στη γλώσσα που θεωρεί πιο κατάλληλη για το είδος της εφαρμογής. Στη συνέχεια το πληκτρολογεί στον Υ, το τοποθετεί στη μνήμη και στη συνέχεια στο δίσκο. Αφού το δοκιμάσει, το διορθώσει κλπ και βεβαιωθεί πως δουλεύει σωστά, το τοποθετεί σε μία ή περισσότερες δισκέτες (ανάλογα με το πόσο μεγάλο είναι) και το διαθέτει στην αγορά μαζί με τις οδηγίες χρήσης του (το τεχνικό εγχειρίδιο, manual). Ο ενδιαφερόμενος χρήστης αγοράζει το πακέτο (δισκέτες και manual), φορτώνει το πρόγραμμα από τις δισκέτες στο σκληρό εσωτερικό δίσκο του PC του και μαθαίνει να το χρησιμοποιεί με τη βοήθεια του manual.

Μια από τις πρώτες κατηγορίες εφαρμογών που τυποποιήθηκαν και γράφτηκαν πακέτα ήταν οι εφαρμογές αυτοματοποίησης πράξεων σε καταστάσεις - λογιστικά φύλλα (spreadsheets) όπως καθιερώθηκε να ονομάζονται, οι εφαρμογές επεξεργασίας κειμένων (word processing) που έφεραν επανάσταση στο γραφείο και ήταν ο καταλύτης για την εισαγωγή των ΠΥ στις μικρές επιχειρήσεις και τη ραγδαία ανάπτυξή τους, οι εφαρμογές επεξεργασίας εικόνων για γραφίστες (graphics).

Είδη πακέτων

Υπάρχουν πολλές κατηγορίες πακέτων κάθε μία από τις οποίες αντιστοιχεί σε μία κατηγορία εφαρμογών. Για κάθε κατηγορία εφαρμογών υπάρχουν

πολλά πακέτα από τα οποία μπορεί να επιλέξει κανείς. Τα διάφορα είδη πακέτων μπορούν να ομαδοποιηθούν σε δύο βασικές ομάδες τα Γενικής Χρήσης πακέτα και τα Ειδικά (κάθε κατηγορία πακέτων λειτουργεί με τους αντίστοιχους τύπους αρχείων) :

Α. Γενικής χρήσης. Απευθύνονται σε κάθε ενδιαφερόμενο. Τα κυριότερα είδη είναι :

- επεξεργασίας κειμένων (Word Processing)
- επεξεργασίας λογιστικών φύλλων - spreadsheets
- επεξεργασίας αριθμητικών δεδομένων - αρχείων όπως είναι τα γνωστά πακέτα Διαχείρισης Βάσεων Δεδομένων (Συστήματα Διαχείρισης Βάσεων Δεδομένων - ΣΔΒΔ ή Data Base Management Systems - DBMS)
- επεξεργασίας σχεδίων και εικόνων - graphics
- επεξεργασίας κινούμενων εικόνων - animation
- επεξεργασίας ήχων - μουσική
- στατιστικών επεξεργασιών, π.χ. spss κλπ

Β. Ειδικής χρήσης. Απευθύνονται σε εξειδικευμένους χρήστες:
- πακέτα για εμπορικές επιχειρήσεις, για νοσοκομεία, δικηγορικά γραφεία, ιατρεία κλπ.

Πακέτα και εξοπλισμός

Κάθε πακέτο έχει τις απαιτήσεις του σε μνήμη και ταχύτητα επεξεργαστή (ισχύ επεξεργαστή) για να λειτουργεί ικανοποιητικά. Είναι προφανές πως ο χρήστης που αγοράζει ένα Υ σκοπεύοντας να τον χρησιμοποιήσει για μια κατηγορία εφαρμογών θα πρέπει να προβλέψει, ώστε το σύστημα που θα πάρει να ανταποκρίνεται στις ελάχιστες (και κάτι παραπάνω) απαιτήσεις του πακέτου του.

12 Ο ΚΟΣΜΟΣ ΤΟΥ ΥΠΟΛΟΓΙΣΤΗ

Με μια μικρή αφαίρεση και κάποιες ίσως τολμηρές παρομοιώσεις μπορούμε να δούμε τον κόσμο του PC σαν κάτι ανάλογο του δικού μας κόσμου. Έχουμε λοιπόν:

- τη χώρα, δηλ τον εξοπλισμό, δηλ το PC μας που εξετάσαμε ήδη

- την κυβέρνηση που διοικεί τη χώρα, ώστε να προκύψει ένα λειτουργικό κράτος. Το αντίστοιχο είναι το Λειτουργικό Σύστημα. Η κυβέρνηση χρησιμοποιεί τους πόρους της χώρας, προσφέρει στους πολίτες τις απαραίτητες βασικές υπηρεσίες και καθορίζει το πλαίσιο (νόμοι, κανονισμοί) μέσα στο οποίο κινείται η οικονομική και κοινωνική ζωή των πολιτών. Αντίστοιχη είναι και η αποστολή του ΛΣ σ' έναν Υ.

- τις επιχειρήσεις παροχής υπηρεσιών, δηλ τα πακέτα εφαρμογών. Δημιουργούνται από τους πολίτες μέσα στο πλαίσιο των νόμων και διατάξεων που προβλέπονται από την κυβέρνηση. Προσφέρουν εξειδικευμένες υπηρεσίες σε μια μεγάλη κατηγορία εφαρμογών. Εχουμε τα εστιατόρια, ξενοδοχεία, βιβλιοπωλεία, εκδοτικές εταιρείες, επιχειρήσεις δακτυλογράφησης, λογιστικές κλπ.
Είναι προφανές πως αν κάποιος θέλει να μεταφέρει την επιχείρησή του σε άλλο κράτος, όπου υπάρχει άλλη κυβέρνηση και άλλοι νόμοι, θα πρέπει να προσαρμόσει ανάλογα την επιχείρησή του, δηλ. δεν θα μπορεί να την μεταφέρει ακριβώς όπως ήταν. Αντίστοιχα λειτουργούν και τα πακέτα εφαρμογών τα οποία μπορούν να θεωρηθούν σαν επιχειρήσεις παροχής υπηρεσιών με αυτοματοποιημένη - self service λειτουργία (όπως π.χ. κάτι επιχειρήσεις που διαθέτουν χώρους με πλυντήρια και πηγαίνουν οι

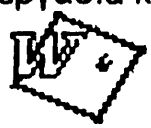
εργένηδες και πλένουν μόνοι τα ρούχα τους). Τα πακέτα φτιάχνονται από τους δημιουργούς τους με βάση τους κανόνες και νόμους και προδιαγραφές ενός συγκεκριμένου ΛΣ και επομένως μόνο κάτω από αυτό το σύστημα μπορούν να λειτουργήσουν. Για να λειτουργήσουν σε άλλο ΛΣ χρειάζονται μετατροπές. Συχνά πετυχημένα πακέτα δημιουργούνται σε διαφορετικές εκδόσεις για να τρέξουν σε διαφορετικά δημοφιλή ΛΣ.

- τους πολίτες, δηλ τους χρήστες, όλους εμάς. Και μάλιστα για τη χώρα του PC μας, εμείς είμαστε ο ένας και μοναδικός προνομιούχος πολίτης. Οι πολίτες μπορούν να ζητήσουν τις υπηρεσίες που προσφέρει το κράτος - ΛΣ ακολουθώντας τη διαδικασία που προβλέπεται. Η διαδικασία αυτή εξαρτάται από το είδος του κράτους που έχει διαμορφωθεί από το ΛΣ. Σε άλλα οι υπηρεσίες ζητούνται με εντολές και σε άλλα μέσα από το παραστατικό γραφικό περιβάλλον, όπως είδαμε στο προηγούμενο κεφάλαιο.

Για να πολιτογραφηθούμε πολίτες σ' αυτή την ευνομούμενη χώρα του PC, θα πρέπει να μελετήσουμε τους βασικούς νόμους που τη διέπουν (Λειτουργικά συστήματα) και τις βασικές επιχειρήσεις παροχής υπηρεσιών που μπορούμε να έχουμε:

Από πλευράς ΛΣ το πιό διαδεδομένο τη στιγμή αυτή είναι το Windows 98 και κυκλοφορούν Η/Υ με 2 ΛΣ.

Από πλευράς εφαρμογών οι πιό διαδεδομένες είναι οι εφαρμογές που καλύπτονται από το πολυπακέτο MS-Office της Microsoft που περιλαμβάνει:
α. Επεξεργασία κειμένου με το MS-WORD

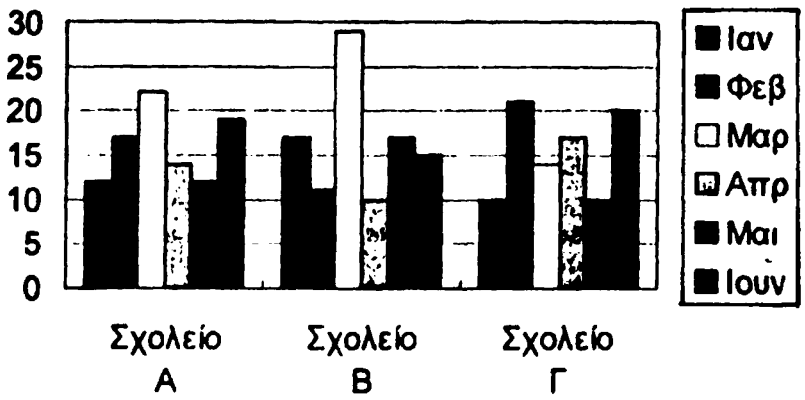


Winword.exe

β. Επεξεργασία λογιστικών φύλλων - spreadsheet με το EXCEL
Πιο κάτω υπάρχουν ένα φύλλο από δεδομένα περασμένα στο excel και το αντίστοιχο γράφημά του. Τα δεδομένα αφορούν αριθμό παιδιών που πρώτευσαν, ανά μήνα σε τρία σχολεία (Σχολείο Α, Σχολείο Β, Σχολείο Γ).

	Σχολείο Α	Σχολείο Β	Σχολείο Γ
Ιαν	12	17	10
Φεβ	17	11	21
Μαρ	22	29	14
Απρ	14	10	17
Μαι	12	17	10
Ιουν	19	15	20

Δεδομένα γραμμένα σε φύλλο του Excel



Γραφική απεικόνιση από τα παραπάνω δεδομένα

γ. Επεξεργασία βάσεων δεδομένων με το ACCESS.

δ. Προετοιμασία παρουσιάσεων με διαφάνειες . με άλλα μέσα με το Power Point.
Πιο κάτω σας δείχνουμε μικρή παρουσίαση που σχεδιάστηκε με το Power Point για να μας δείξει τα μέσα μεταφοράς.

Μεταφορικά Μέσα

Τα Μεταφορικά μέσα
τα χρησιμοποιούμε για επικοινωνίες

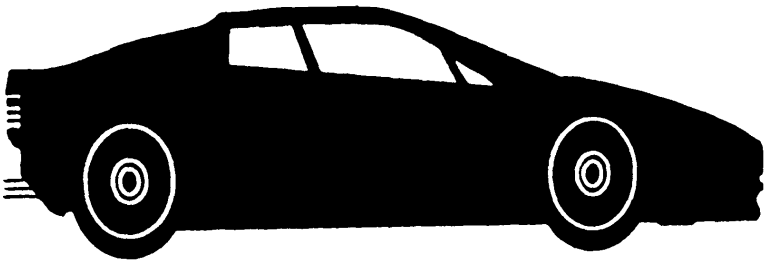
Στην ξηρά

Στον αέρα

Στη θάλασσα

Διαφάνεια 1η

αυτοκίνητο



Διαφάνεια 2η

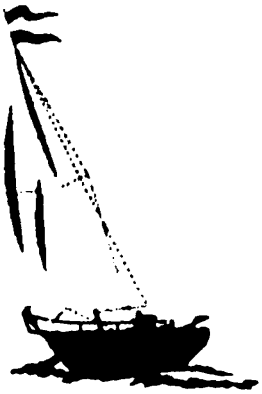
Μεταφορικά μέσα

Το αεροπλάνο

Διαφάνεια 3η

Μεταφορικά μέσα

Ιστιοφόρο



Διαφάνεια 4η

13 ΠΑΡΑΛΛΗΛΗ ΕΠΕΞΕΡΓΑΣΙΑ

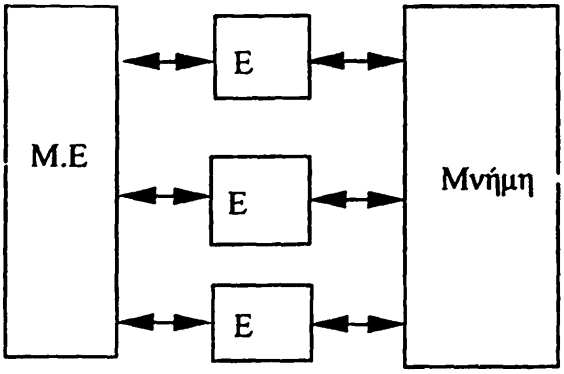
Με τον όρο παράλληλη επεξεργασία εννοούμε μια διαδικασία με την οποία γίνονται πολλές εργασίες συγχρόνως.
Παράλληλος υπολογιστής είναι ένας υπολογιστής που αρχιτεκτονικά είναι έτσι κατασκευασμένος ώστε να έχει τη δυνατότητα να κάνει παράλληλη επεξεργασία. Στον αρχιτεκτονικό σχεδιασμό του υπολογιστή προβλέπεται και ένας επιπλέον μηχανισμός συλλογής των επιμέρους εργασιών που προκύπτουν από την παράλληλη επεξεργασία, με ένα δίκτυο εσωτερικής επικοινωνίας.

Υπάρχουν πολλές και διαφορετικές αρχιτεκτονικές στους παράλληλους υπολογιστές. Παρά τις πολλές προσπάθειες που έγιναν με στόχο την ταξινόμηση των διαφόρων τύπων παράλληλων υπολογιστών καμία δεν επέτυχε πλήρως. Μια ταξινόμηση που έγινε από τον Flynn (1972) διαχωρίζει τους υπολογιστές σύμφωνα με τον τρόπο που η μηχανή σχετίζει τις οδηγίες της με τα δεδομένα που επεξεργάζεται. Η ταξινόμηση αυτή έχει ως ακολούθως:

1. SISD (Single Instruction Single Data Stream) (Μια οδηγία -Μια εκτέλεση)
2. SIMD (Single Instruction Multi Data Stream)
3. MISD (Multiple Instruction Single Data)
4. MIMD (Multiple Instruction Multiple Data)

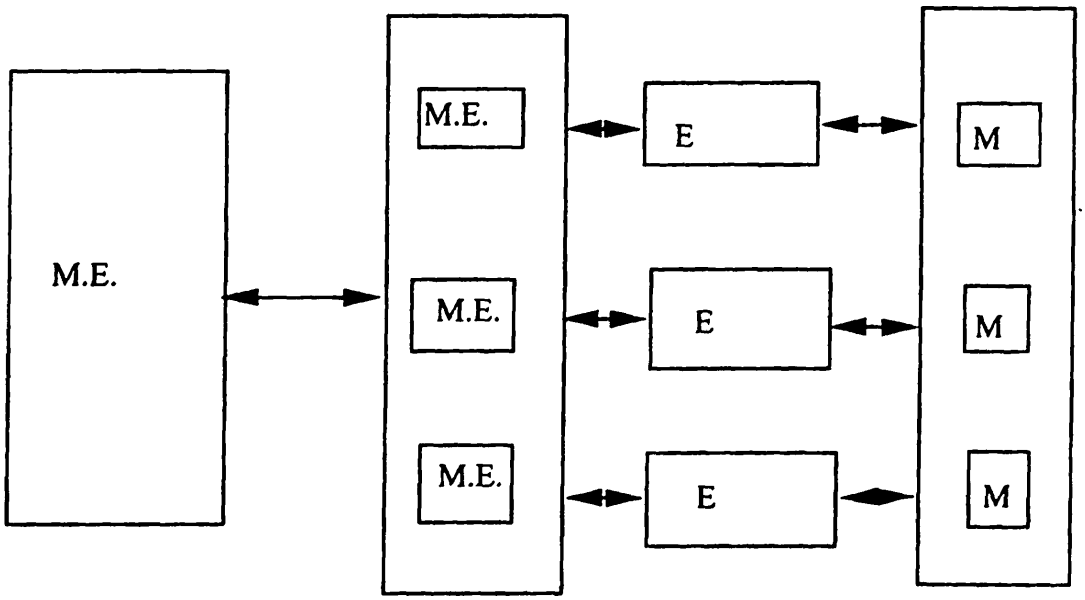
Αναλυτικότερα:

1. Οι SIMD υπολογιστές τυπικά εκτελούν μια ομάδα οδηγιών (εντολών) με έναν αριθμό από απλές μονάδες επεξεργασίας που η κάθε μια κάνει τις ίδιες οδηγίες στα δικά της δεδομένα (διάγραμμα ε1) .



Διάγραμμα ε1. Διαγραμματική παράσταση σε υπολογιστή SIMD.

2. Στους MIMD υπολογιστές κάθε επεξεργαστής έχει τη δική του μονάδα ελέγχου και μπορεί είτε να έχει και τη δική του μνήμη είτε να μοιράζεται μια κοινή μνήμη (διάγραμμα ε2).



Διάγραμμα ε2. Διαγραμματική παράσταση σε υπολογιστή MIMD. M.E (Μονάδα Ελέγχου), E(Επεξεργαστής), M(Μνήμη).

Στην περίπτωση που ο υπολογιστής έχει κοινή μνήμη, όπως οι Gray-2, Encore Multimax, Alliant FX/80, τότε όλοι οι επεξεργαστές επικοινωνούν με τη μνήμη αυτή.

Στην περίπτωση που ο υπολογιστής έχει τοπική ή μοιρασμένη μνήμη, όπως οι Intel iPSC/1, Ametek S14 Hypercube, τότε όλοι οι επεξεργαστές επικοινωνούν απ' ευθείας με τη δική τους μνήμη.

Τόσο στους SIMD όσο και στους MIMD υπολογιστές, ο συντονισμός των επεξεργαστών είναι αναγκαίος διότι συνήθως ένας προγραμματιστής σε παράλληλους υπολογιστές σπάει το πρόγραμμα σε μικρότερα υποπροβλήματα για να τα επεξεργαστούν οι επεξεργαστές του υπολογιστή οι οποίοι πρέπει να επικοινωνούν μεταξύ τους. Ο συντονισμός-επικοινωνία των επεξεργαστών είναι σύγχρονος ή ασύγχρονος. Στα συστήματα με κοινή μνήμη έχουμε σύγχρονη επικοινωνία ενώ στα συστήματα με επεξεργαστές που έχουν τοπική μνήμη έχουμε ασύγχρονη επικοινωνία και ο συντονισμός τους γίνεται με τη μέθοδο μεταφοράς μηνύματος ή MP (Message-passing). Με τον όρο μεταφορά μηνύματος εννοούμε τη διαδικασία κατά την οποία τα δεδομένα που δίνουμε στις μεταβλητές ενός υποπρογράμματος μετακινούνται και δίνονται στις μεταβλητές ενός άλλου υποπρογράμματος. Το MP δεν ενδιαφέρεται για τις τιμές που έχουν τα δεδομένα που μετακινούνται, παρά μόνο για τη μεταφορά τους ως διαδικασία. Το βασικό σημείο στο MP είναι ότι υποστηρίζει διαφορετικές αρχιτεκτονικές υπολογιστών με κύριο στόχο οι επεξεργαστές να επικοινωνούν μεταξύ τους στέλνοντας ο ένας στον άλλο μηνύματα. Τα προγράμματα που είναι γραμμένα σε MP μπορούν να τρέξουν και σε υπολογιστές με τοπική ή κοινή μνήμη. Στην καθημερινή ζωή για τη μεταφορά μηνυμάτων έχουμε τα γνωστά σε όλους fax, τηλέφωνα, γράμματα, ράδιο. Οπως τα τηλέφωνα έχουν διαφορετικούς τρόπους μεταφοράς μηνυμάτων το καθένα έτσι και στο MP υπάρχουν διαφορετικοί τρόποι επικοινωνίας. Στο MP για τη μεταφορά των μηνυμάτων χωρίς λάθη και περικοπές τους από τον αποδέκτη, είναι απαραίτητα να είναι εγκατεστημένα:

α. ένα σύστημα επικοινωνίας

β. ένας διευθυνσιογράφος.

Για τη σωστή μεταφορά μηνυμάτων το MP πρέπει να έχει τις εξής πληροφορίες:

1. Ποιος επεξεργαστής στέλνει την πληροφορία
2. Πού βρίσκονται τα δεδομένα στον επεξεργαστή που στέλνει την πληροφορία
3. Τι είδους δεδομένα θα μεταφερθούν
4. Πόσα δεδομένα υπάρχουν
5. Ποιος επεξεργαστής παίρνει την πληροφορία
6. Σε ποιο μέρος του επεξεργαστή, που θα πάρει τα δεδομένα, θα πάνε τα δεδομένα
7. Τι ποσότητα δεδομένων μπορεί να καταχωρήσει ο επεξεργαστής που θα λάβει τα δεδομένα

Επικοινωνίες

Σύγχρονη επικοινωνία: Στη σύγχρονη επικοινωνία ο επεξεργαστής που στέλνει το μήνυμα περιμένει έως ότου ενημερωθεί από τον αποδέκτη ότι έλαβε το μήνυμα.

Ασύγχρονη επικοινωνία: Στην ασύγχρονη επικοινωνία ο επεξεργαστής που στέλνει το μήνυμα γνωρίζει μόνο πότε έφυγε το μήνυμα προς τον αποδέκτη. Παράδειγμα ασύγχρονης επικοινωνίας είναι όταν στέλνουμε ένα γράμμα με το ταχυδρομείο, που ξέρουμε μόνο πότε ταχυδρομήσαμε το γράμμα.

Είδη επικοινωνιών

Δύο είδη επικοινωνιών έχουμε το:
ένας-προς-ένα δηλαδή ένας στέλνει και ένας λαμβάνει
και το: ένας-προς-πολλούς.

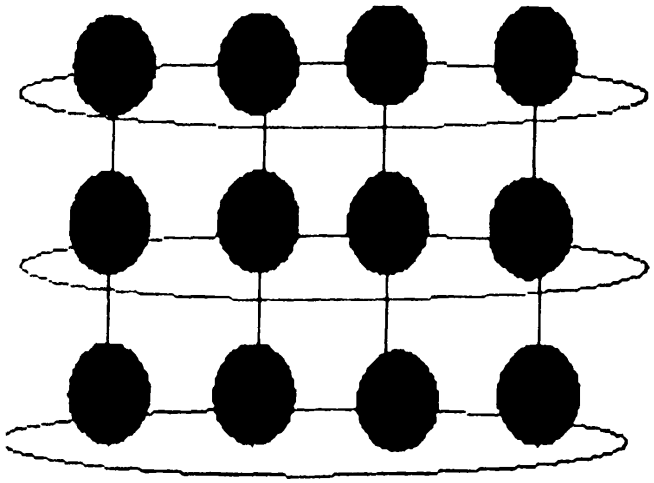
MPI (message passing interface)

Το MPI είναι ένας αλγόριθμος, γραμμένος σε πρόγραμμα στις γλώσσες προγραμματισμού C ή FORTRAN με σκοπό την επικοινωνία μεταξύ επεξεργαστών με διαφορετικές ή όμοιες αρχιτεκτονικές, καλώντας διάφορες MPI υπορουτίνες.

Τοπολογίες

Ο όρος τοπολογία στους παράλληλους υπολογιστές αναφέρεται στον τρόπο με τον οποίο είναι συνδεδεμένοι οι επεξεργαστές στα MIMD και σε μερικά SIMD συστήματα. Υπάρχουν διάφορες τοπολογίες μεταξύ των οποίων είναι οι γραμμικές αλυσίδες, οι δακτύλιοι και οι βρόγχοι. Κάθε τοπολογία έχει μοναδικές και ξεχωριστές ιδιότητες επικοινωνίας των επεξεργαστών μεταξύ τους (διάγραμμα τ1). Υπάρχουν δύο ειδών

τοπολογίες: οι καρτεσιανές και οι γραφικές. Στις καρτεσιανές τοπολογίες κάθε διαδικασία είναι συνδεδεμένη με την γειτονική της και ορίζεται με τις καρτεσιανές συντεταγμένες . Οι γραφικές τοπολογίες καλύπτονται από τη θεωρία περί γραφημάτων.

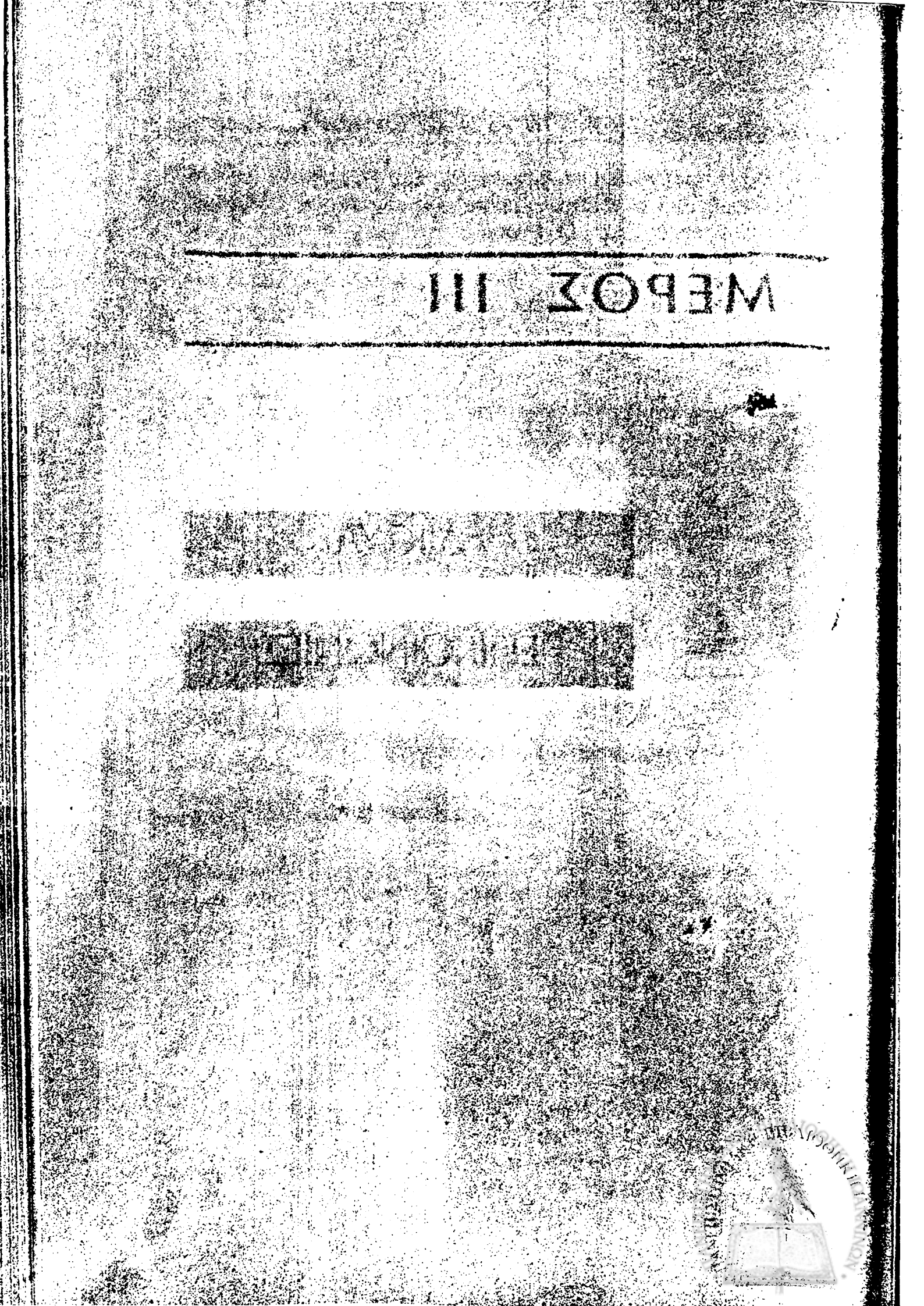


Διάγραμμα τ1: Παράδειγμα μιας 2-διάστατης τοπολογίας.

ΜΕΡΟΣ ΙΙΙ

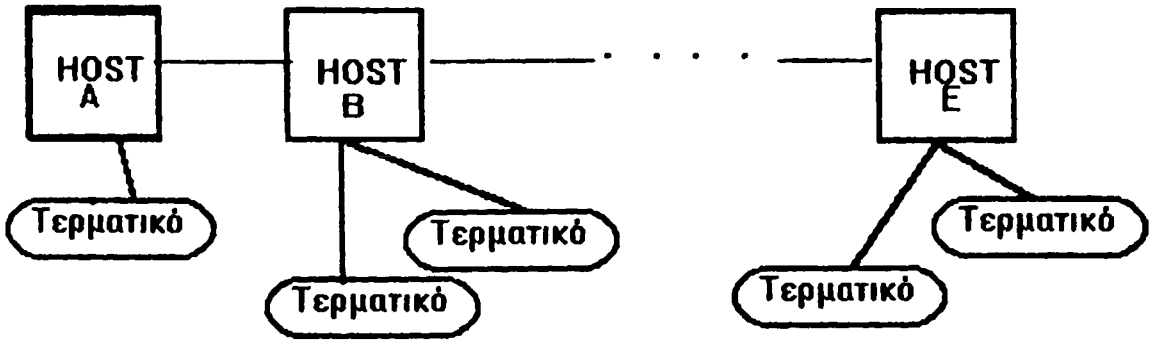
ΔΙΚΤΥΑ

ΕΠΙΚΟΙΝΩΝΙΕΣ



14 ΔΙΚΤΥΑ ΥΠΟΛΟΓΙΣΤΩΝ

Δίκτυο Η/Υ είναι μια ομάδα Η/Υ που επικοινωνούν και συνεργάζονται μεταξύ τους σε τοπικό ή μη επίπεδο (εικόνα 14.1). Οι αυτόνομοι υπολογιστές που είναι συνδεδεμένοι μεταξύ τους και συνεργάζονται ισότιμα, χωρίς να είναι κάτω από τον έλεγχο ενός Η/Υ, λέγονται Hosts του δικτύου. Ο κάθε Host μπορεί να έχει και τα τερματικά του, όπως φαίνεται στο διάγραμμα 14.1.



Διάγραμμα 14.1: Σύνδεση των Hosts με τα τερματικά τους σε δίκτυο.

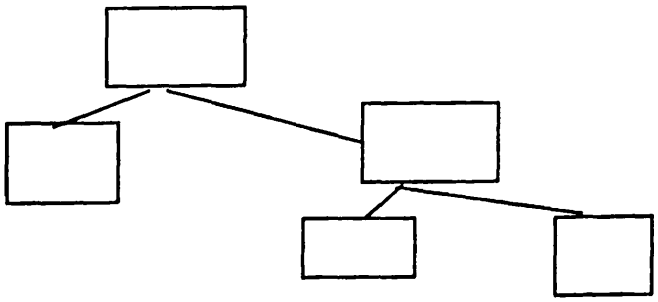
Ανάλογα με την έκταση που καταλαμβάνουν τα δίκτυα διακρίνουμε δύο περιπτώσεις, το WAN και το LAN.
Στα δίκτυα εξετάζουμε γενικότερα τόσο την τοπολογία τους όσο και την αρχιτεκτονική τους. Σε κάθε ένα από αυτά τα δίκτυα (το WAN και το LAN)

για την επικοινωνία των υπολογιστών μεταξύ τους χρησιμοποιείται μια ορισμένη τοπολογία και αρχιτεκτονική.
Με την τοπολογία εξετάζουμε τη φυσική διάταξη από τα καλώδια που συνδέουν τους κόμβους του δικτύου. Οι κυριότερες τοπολογίες είναι:

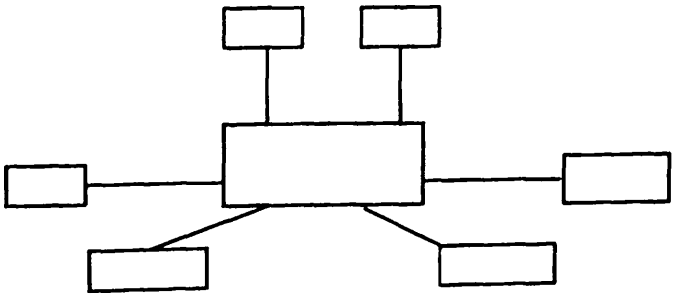
1. Ο γραμμικός δίαυλος (έχει σχήμα γραμμής)
2. Ο δακτύλιος (έχει σχήμα δακτυλίου)
3. Ο αστέρας (έχει σχήμα αστέρα)
4. Οι υβριδικές τοπολογίες (συνδυασμός μιας ή περισσοτέρων από τις παραπάνω τοπολογίες).

Στις αρχές της δεκαετίας του '70 η IBM κατασκεύασε το SNA (System Network Architecture) για τη λειτουργία ενός συστήματος επικοινωνιών, ανεξάρτητα από τα μηχανήματα που το αποτελούν. Την IBM ακολούθησαν άλλες εταιρείες όπως ή DEC με το Digital Network Architecture (DNA), ή CDC με το CDC NET κ.λ.π. Με τον τρόπο αυτό ο χρήστης δεν χρειάζεται να γνωρίζει το SNA πώς λειτουργεί, αλλά μέσω του τερματικού του διατυπώνει το τι θέλει να μάθει, το SNA το παραλαμβάνει αυτό το αίτημα και το παραδίδει στον Host ο οποίος απαντάει και σε λίγα δευτερόλεπτα μέσω SNA παρουσιάζονται στην οθόνη του χρήστη τα αποτελέσματα.
Γενικότερα όμως με την αρχιτεκτονική του δικτύου εξετάζεται η διάταξη των δικτύων με σκοπό την καλύτερη απόδοσή τους.
Πιο κάτω αναφέρονται ενδεικτικά οι κυριότερες αρχιτεκτονικές σχεδιασμού δικτύων που προσφέρουν έναν ορθολογικό σχεδιασμό με στόχο τη βελτιστοποίηση της αποδοτικότητας των υπολογιστών και είναι:

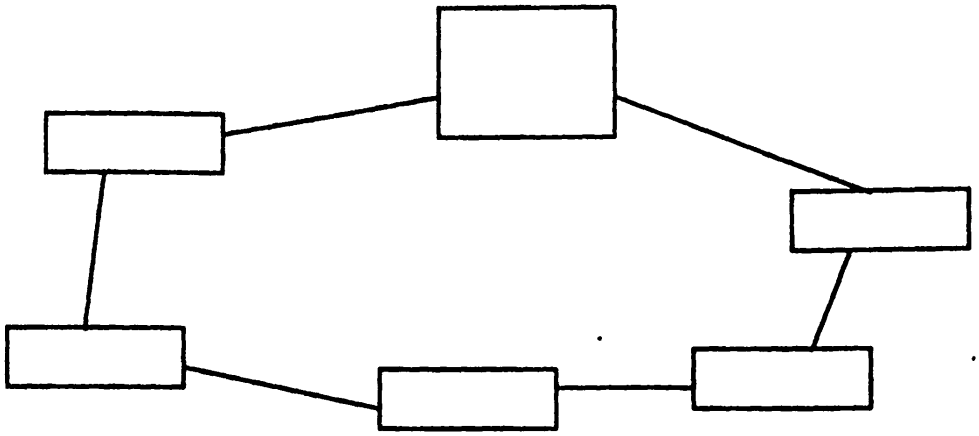
1. Αρχιτεκτονική δομής δέντρου (ιεραρχική)



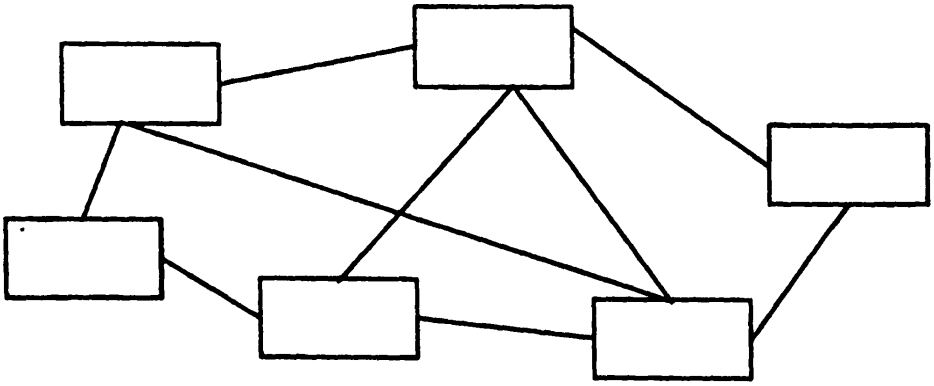
2. Αρχιτεκτονική δομής αστέρος



3. Αρχιτεκτονική δομής δακτυλίου



4. Αρχιτεκτονική διανεμημένης δομής



Είδη δικτύων

α) WAN (Wide Area Networks): Στο WAN οι αποστάσεις των Hosts μεταξύ τους είναι μερικά Km σε αντίθεση με το LAN που αφορά μικρές αποστάσεις. Στην περίπτωση αυτή τοποθετούνται ενδιάμεσα επικοινωνιακά κέντρα ή πολυπλέκτες. Η χρήση των πολυπλεκτών είναι σημαντική και η θέση τους είναι πριν από τη γραμμή επικοινωνίας και μετά από το modem.

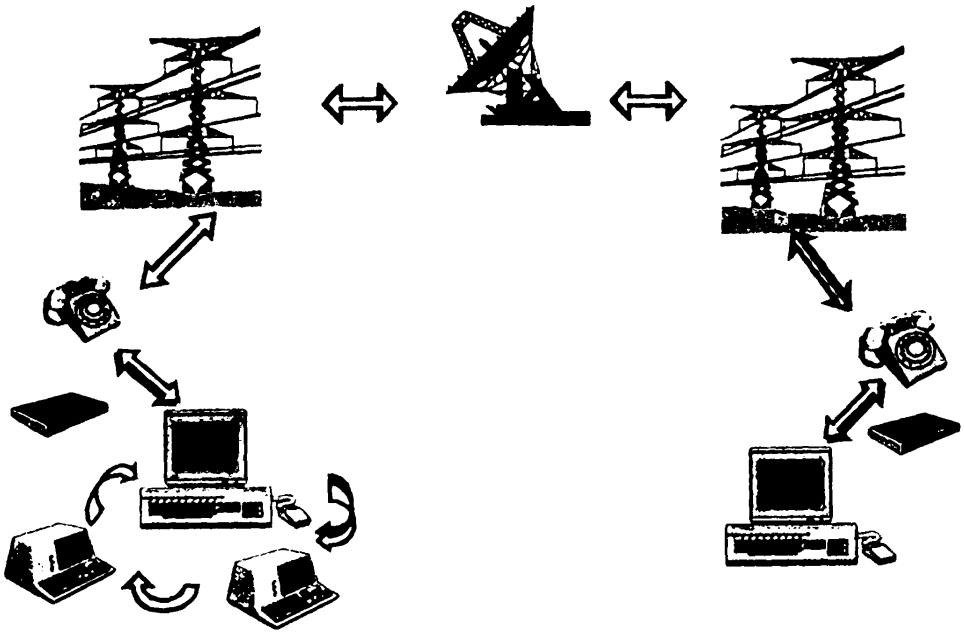
β) LAN (Local Area Networks): Οι αποστάσεις μεταξύ των Hosts είναι μικρές και εκεί για τη σύνδεση των Η/Υ μεταξύ τους, χρησιμοποιούνται κυρίως καλώδια. Παρέχουν επάρκεια στην επικοινωνία μεταξύ προσωπικών Η/Υ, mini computers, mainframes και περιφερειακών σε μια ορισμένη περιοχή.

Σε ένα δίκτυο LAN μπορεί να υπάρχει ένας κεντρικός controller-ελεγκτής Η/Υ που κανονίζει την επικοινωνία των Hosts μεταξύ τους. Πολλές φορές όμως δεν υπάρχει τέτοιος Η/Υ -controller- και οι Η/Υ του δικτύου αυτοδιαχειρίζονται. Έτσι, η επικοινωνία μεταξύ τους γίνεται είτε με την random access (τυχαία προσπέλαση), είτε με την token ring (προσπέλαση μέσω του δακτυλίου).

Τέλος, για τη σύνδεση των Η/Υ μεταξύ τους συνήθως χρησιμοποιείται το Ethernet (δίκτυο ανοικτού τύπου, όπου το καλώδιο είναι επίμηκες) ή το Token ring (όπου το καλώδιο έχει σχήμα δακτυλιδιού) και το υποστηρίζει η IBM. Σε δίκτυα ανοικτού τύπου δεν υπάρχει επιβεβαίωση ότι το σήμα έχει φτάσει στον προορισμό του και ακόμη δεν είναι εύκολη η διόρθωση λαθών μετάδοσης.

Έτσι το LAN ανάλογα με τον τύπο του καλωδίου που χρησιμοποιείται για τη σύνδεση μεταξύ των Host, διακρίνεται σε: α) bus και β) ring. Πολλές φορές η αγορά είναι εκείνη που καθορίζει τον τύπο Η/Υ που κατασκευάζονται και τον τρόπο σύνδεσης μεταξύ τους. Για παράδειγμα αναφέρουμε την General Motors στην Αμερική, που αγοράζοντας μεγάλο αριθμό Η/Υ, "καθορίζει" τις προδιαγραφές τους και τον τρόπο σύνδεσης μεταξύ τους.

Ένα άλλο είδος δικτύων που επιτρέπει την ταυτόχρονη μετάδοση ομιλίας με ψηφιακή μορφή, έχει τη δυνατότητα επικοινωνίας υπολογιστών και μπορεί να μεταδώσει προγράμματα τηλεόρασης λέγεται MAN (Metropolitan Area Networks).



Εικόνα 14.1. Δίκτυα υπολογιστών.

Δίκτυα καλωδιακής τηλεόρασης

Με τα δίκτυα καλωδιακής τηλεόρασης γίνεται η μεταφορά σημάτων τηλεοπτικών καναλιών και χρησιμοποιούνται σε πολλές χώρες της Ευρώπης. Ακόμη, με τα δίκτυα αυτά, έχουμε τη δυνατότητα επικοινωνίας διπλής κατεύθυνσης.

Πως συνδέονται οι Η/Υ ενός δικτύου.

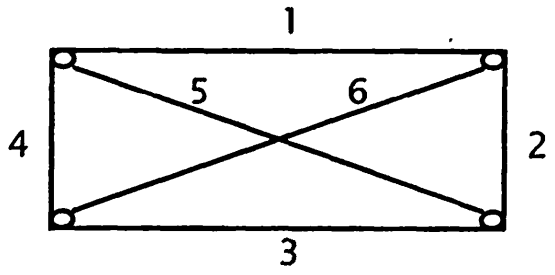
Το 1969 το Υπουργείο Αμυνας των ΗΠΑ, χρησιμοποίησε το ARPANET (Advanced Research Projects Agency), για να συνδέσει τους Η/Υ που είχαν μεγάλη απόσταση μεταξύ τους. Η επικοινωνία των δεδομένων αυτών είναι αντίστοιχη με το δίκτυο τηλεπικοινωνιών και λέγεται Subnet ή δίκτυο επικοινωνίας δεδομένων και παρεμβάλλεται στους Hosts του δικτύου.
Τα δεδομένα του Host A που στέλνονται στον Host E περνούν μέσα από κόμβους του Subnet και από εκεί είτε στέλνονται όλα μαζί (store and forward) μια φορά στον E, είτε στέλνονται τμήματα-τμήματα και στον τελευταίο κόμβο, πριν τον Host E, ταξινομούνται και μπαίνουν στη σειρά τους (διάγραμμα δ1). Ο δεύτερος αυτός τρόπος μεταγωγής μηνυμάτων λέγεται PSN (Packet Switched Network). Το Packet Switching επιτρέπει πολλές και όχι μόνο 2 επικοινωνίες μεταξύ Η/Υ να οδεύουν μέσα από τον ίδιο κόμβο βάσει ενός ειδικά οργανωμένου συστήματος. Το όνομα Packet

(πακέτο) δηλώνει ότι οι πληροφορίες που μεταφέρονται, είναι χρήσιμες κατά τμήματα, καθένα από τα οποία και το χειριζόμαστε διαφορετικά.

Επικοινωνίες μεταξύ υπολογιστών

Ο αριθμός επικοινωνιών μεταξύ N Η/Υ είναι N (N-1)/2.
Παράδειγμα: Για 4 Η/Υ έχουμε τις παρακάτω 6 επικοινωνίες (N=4, N-1=3 και συνεπώς 4*3/2=6).

Πιο αναλυτικά παρουσιάζονται στο διάγραμμα 14.2 που ακολουθεί.



Διάγραμμα 14.2. Αριθμός επικοινωνιών μεταξύ 4 Η/Υ.

Με τον τρόπο αυτό μεγάλοι οργανισμοί, όπως τράπεζες, πολυεθνικές εταιρείες, συνδέουν τους Host Η/Υ. Οι μικροί χρήστες που δεν μπορούν να αναλάβουν κάποιο τέτοιο κόστος σύνδεσης, συνδέονται μέσω ΟΤΕ που έχει αντίστοιχο δίκτυο και ένα από αυτά είναι το HELLASPAC.

Διασύνδεση δικτύων υπολογιστών

Στη διασύνδεση των δικτύων υπολογιστών χρησιμοποιούνται οι παρακάτω συσκευές:

- Δρομολογητής (router) ονομάζεται μια συσκευή διασύνδεσης που συνδέει δύο ή περισσότερα δίκτυα .στο επίπεδο του δικτύου, προωθώντας πακέτα μεταξύ όμοιων ή και ανόμοιων δικτύων. Οι δρομολογητές για δίκτυα T.C.P./I.P ονομάζονται Πύλες (gateways) ενώ για OSI δίκτυα, ενδιάμεσα συστήματα (intermediate systems IS).

- Πύλες (gateways) ή μετατροπείς πρωτοκόλλων (protocol converters) λέγονται οι συσκευές διασύνδεσης που επιτυγχάνουν την διασύνδεση δύο η περισσοτέρων δικτύων σε υψηλότερα του επιπέδου δικτύου επίπεδα.
- επαναλήπτης (repeater) ονομάζεται μια συσκευή διασύνδεσης που συνδέει δύο ή περισσότερα δίκτυα στο φυσικό επίπεδο, προωθώντας bits από το ένα δίκτυο στο άλλο.
- Γέφυρα (bridge) ονομάζεται μια συσκευή διασύνδεσης που συνδέει δύο ή περισσότερα δίκτυα στο επίπεδο σύνδεσης δεδομένων, προωθώντας πλαίσια από ένα δίκτυο στα άλλα

Το hardware και το software ενός δικτύου επικοινωνίας

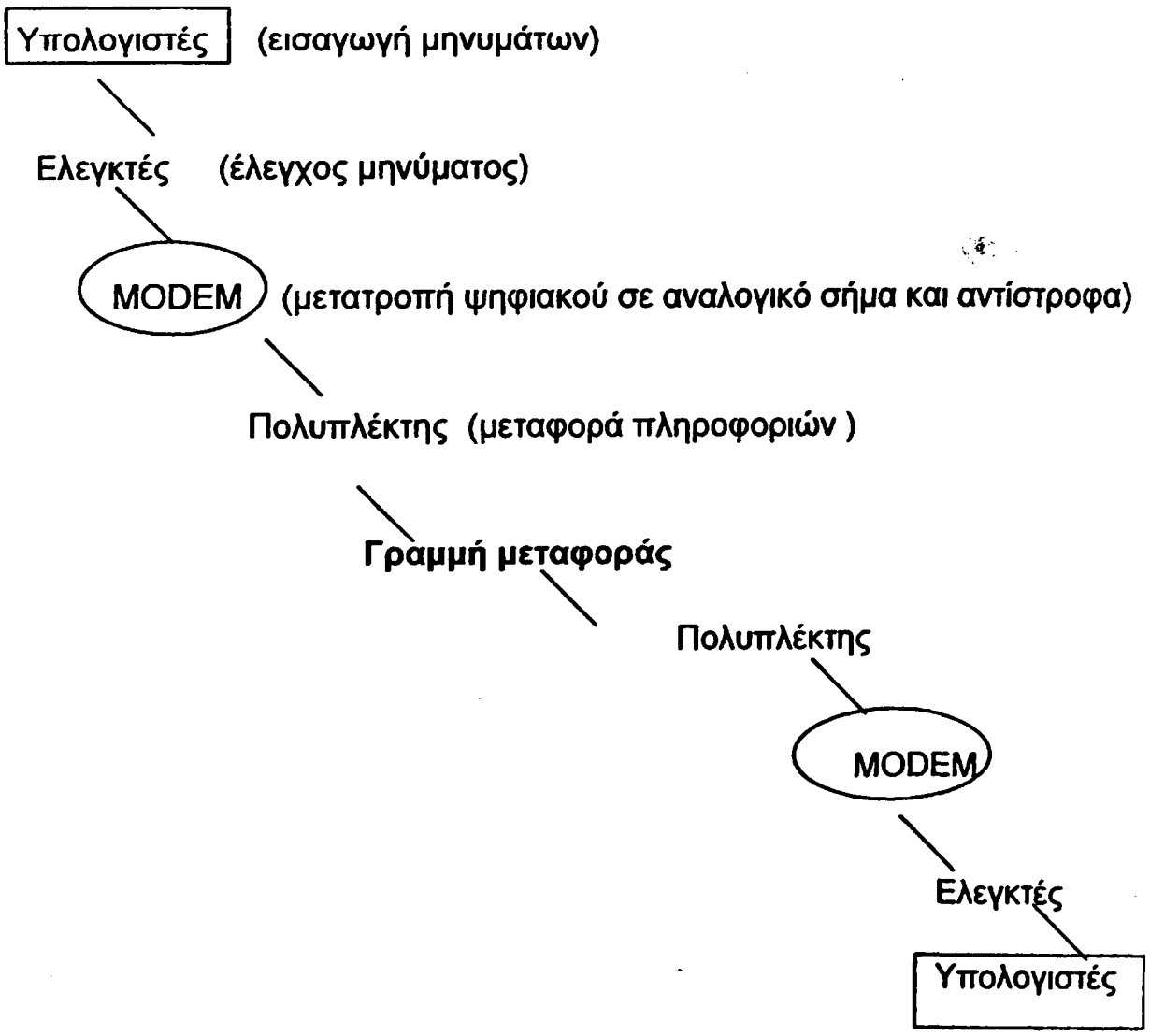
Το hardware ενός δικτύου επικοινωνίας αποτελείται κυρίως από :

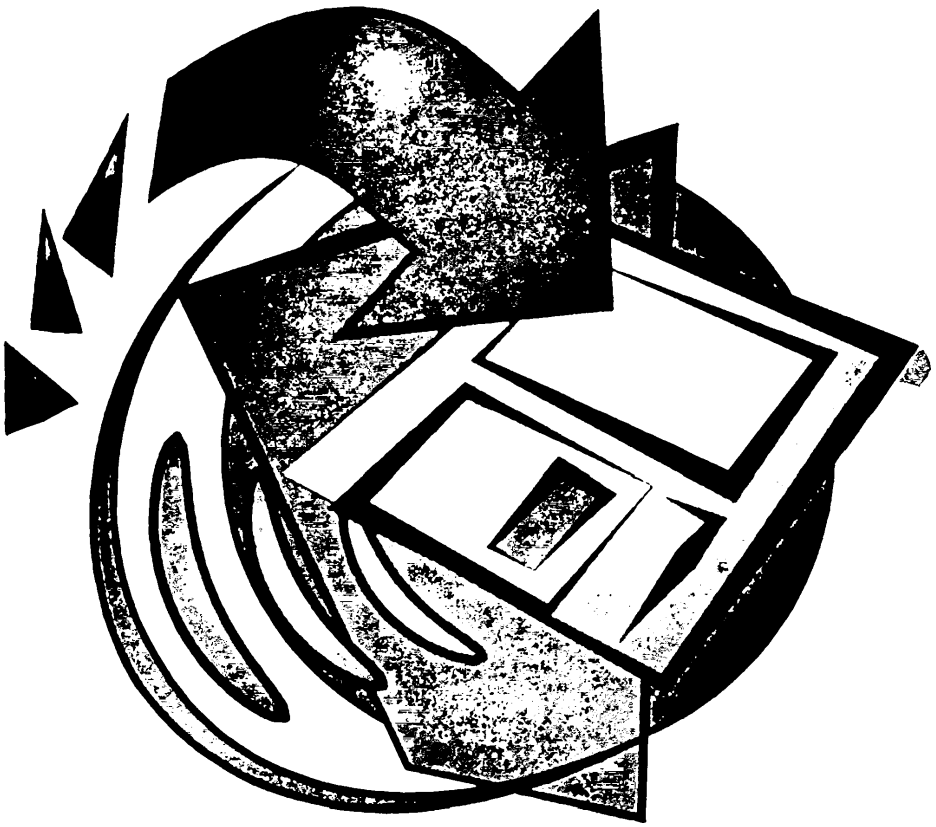
1. Η/Υ, τερματικά (terminals) τα οποία επιτρέπουν στους χρήστες του δικτύου να λάβουν ή να δώσουν πληροφορίες.
2. Τον κεντρικό υπολογιστή (main computer), ο οποίος αποτελεί το host όλου του δικτύου.
3. Τις μονάδες ελέγχου τερματικών (terminal controllers ή T.C.) οι οποίες αποτελούνται από ειδικά κυκλώματα τα οποία ελέγχουν και συντονίζουν τις λειτουργίες των τερματικών.
4. Τα κανάλια επικοινωνίας (communications channels) που χρησιμοποιούνται για την μεταφορά των πληροφοριών και την σύνδεση του δικτύου μπορεί να είναι καλώδια, τηλεφωνικές γραμμές, τηλεγραφικές γραμμές, τηλεπικοινωνιακοί δορυφόροι.
5. Τις μονάδες διανομής γραμμών (multiplexors) οι οποίες τοποθετούνται στην αρχή και στο τέλος μιας γραμμής επικοινωνίας μετά τα modems με σκοπό τη μεταφορά συγχρόνως πολλών πληροφοριών δια μέσου της ίδιας γραμμής.
6. Τα MODEMS (modulators-demodulators). Τα modems μετατρέπουν τα ψηφιακά σήματα των Η/Υ σε αναλογικά σήματα και αντιστρόφως. Τα modems τοποθετούνται στην αρχή και στο τέλος μιας γραμμής επικοινωνίας Η/Υ.
7. Τη μονάδα ελέγχου επικοινωνίας (transmission control unit ή T.C.U.). Η T.C.U. που είναι μια περιφερειακή μονάδα τοποθετημένη μεταξύ του κεντρικού υπολογιστή και του modem με σκοπό τον έλεγχο της μεταβίβασης των πληροφοριών. Μερικές από τις λειτουργίες της είναι: η επιλογή γραμμής επικοινωνίας, ο έλεγχος της γραμμής και η αποθήκευση αυτών των πληροφοριών στη μνήμη.

Το software του δικτύου αποτελείται από :

1. Το λειτουργικό σύστημα (operating system), το οποίο είναι εφοδιασμένο με ειδικά προγράμματα ελέγχου και λειτουργικές μεθόδους μεταβίβασης-προσπέλασης των πληροφοριών.
2. Τα πρωτόκολλα επικοινωνίας (communications protocols), που προσδιορίζουν τον τρόπο και τις τεχνικές επικοινωνίας μεταξύ των σημείων του δικτύου.

Πως συνδέονται όμως οι μονάδες του hardware στα δίκτυα;
Η απλή σύνδεση Η/Υ σε δίκτυο φαίνεται στο παρακάτω σχήμα.





INTERNET

Τι είναι το Internet.

Το Internet είναι ένα δίκτυο υπολογιστών. Αποκαλείται και ως "το δίκτυο των δικτύων", διότι συνδέει χιλιάδες τοπικά δίκτυα. Τον Ιούλιο του 1994 συνέδεε 35.000 δίκτυα και οι χρήστες του ανέρχονταν σε 22 εκατομμύρια σε 83 χώρες του κόσμου. Το Internet αποτελεί ουσιαστικά την λεωφόρο των πληροφοριών και ένα 80% περίπου των επιστημόνων που υπάρχουν σήμερα είναι συνδεδεμένοι με αυτό. Αναλυτικότερα είναι ένα μέσο που παρέχει τη δυνατότητα να διασυνδεθούν μεταξύ τους πολλά δίκτυα σε διάφορα σημεία του πλανήτη μας. Το Internet παρέχει τις διαδικασίες και τη δυνατότητα να διασυνδεθούν όλα τα δίκτυα μεταξύ τους επιτρέποντας στους χρήστες τους να έχουν πρόσβαση, όχι μόνο στο τοπικό δίκτυο του οργανισμού τους αλλά και σε παγκόσμιο επίπεδο. Με το Internet είναι συνδεδεμένα όλα σχεδόν τα μεγάλα ερευνητικά κέντρα του κόσμου και περιλαμβάνει : πληροφορίες από βιβλιογραφία, περιοδικά, χρήσιμες πληροφορίες για μουσική, κινηματογραφικές ταινίες, σύνδεση με πολλές από τις μεγαλύτερες βιβλιοθήκες του κόσμου και σύνδεση με Πανεπιστήμια και Εκπαιδευτικούς Οργανισμούς. Ακόμα είναι δυνατή και εύκολη η επικοινωνία των χρηστών μέσω του ηλεκτρονικού ταχυδρομείου, κάτι που όλοι οι χρήστες του Internet το χρησιμοποιούν πάρα πολύ συχνά.

Η ιστορία του Internet

Πριν από αρκετά χρόνια ξεκίνησε μια προσπάθεια στις Ηνωμένες Πολιτείες της Αμερικής, προκειμένου να κατασκευαστεί ένα δίκτυο, το οποίο θα μπορούσε να εξυπηρετήσει βασικούς σκοπούς σε περίπτωση εκτάκτου ανάγκης, όπως για παράδειγμα σε περίπτωση μιας πολεμικής σύρραξης. Αρχικά, τα τεχνικά προβλήματα τα οποία παρουσιάστηκαν ήταν αδύνατον να ξεπεραστούν με τη μορφή που του είχε δοθεί . Για παράδειγμα, σε περίπτωση καταστροφής κάποιου κύριου σημείου του δικτύου, όλο το σύστημα θα διακόπτονταν λόγω κλειστής οδού από το κατεστραμμένο σημείο. Τότε ένα νέο δίκτυο κατασκευάστηκε αποτελούμενο από ισοδύναμα σημεία τα οποία και ονομάζονται "κόμβοι". Οι κόμβοι αυτοί θα μπορούσαν να λειτουργούν είτε συνολικά (όλοι μαζί) είτε ανεξάρτητα, δηλαδή και απώλεια μερικών από αυτούς. Η κατασκευή επέτρεπε ελαστικότητα στην επικοινωνία μεταξύ κόμβων, δηλαδή έδινε τη δυνατότητα να μπορεί να ακολουθηθεί διαφορετικός δρόμος σύνδεσης, εάν κάτι απρόοπτο συνέβαινε σε ένα κόμβο. Το δίκτυο αυτό ονομάστηκε στην αρχή **ARPANET**, από την εταιρεία κατασκευής του. Αρχικά ήταν συνδεδεμένοι μεταξύ τους πολύ ισχυροί υπολογιστές, που επικοινωνούσαν μέσω τηλεφωνικού δικτύου, χρησιμοποιώντας κάποιες "ειδικές γραμμές" καλύτερης ποιότητας, για την απρόσκοπτη μετάδοση των δεδομένων. Στο δίκτυο αυτό ήταν ενσωματωμένα δυο κομμάτια. Ένα στρατιωτικό, για την κάλυψη των αναγκών που προαναφέραμε και ένα που αφορούσε Οργανισμούς, Υπηρεσίες, Τεχνολογικά Ιδρύματα, Πανεπιστήμια και απλούς πολίτες. Με την πάροδο του χρόνου, ο αριθμός χρηστών αυξήθηκε και το ARPANET είχε μεγαλώσει τόσο, ώστε να είναι αναγκαία η επέκταση του, για την κάλυψη των όλο και αυξανόμενων αναγκών. Για την καλύτερη εξυπηρέτηση των χρηστών, αποφασίστηκε ο χωρισμός του ARPANET σε δύο κομμάτια :

- Το **MILENET** , που κάλυπτε το στρατιωτικό μόνο μέρος και
- Το **INTERNET** , που κάλυπτε όλο το υπόλοιπο κομμάτι.

Στις αρχές του Internet πρόσβαση είχαν στο δίκτυο μόνο οι ερευνητές σε θέματα ηλεκτρονικών υπολογιστών, οι υπάλληλοι του κράτους και οι κυβερνητικοί συντονιστές. Στη συνέχεια όμως απέκτησαν τη δυνατότητα να συνδέονται και με τα μεγάλα Πανεπιστήμια (*με χρηματοδότηση του Internet)* υπό την προϋπόθεση ότι θα προωθούσαν την εξάπλωση του δικτύου. Έτσι ο κάθε πολίτης ο οποίος φοιτούσε σε κάποιο πανεπιστήμιο θα μπορούσε να γίνει χρήστης του Internet.

Σήμερα οι πολλές χιλιάδες υπερυπολογιστών που αποτελούν τους κόμβους του δικτύου προσφέρουν σε όλους τους χρήστες του ένα τεράστιο όγκο πληροφοριών, προγραμμάτων και εφαρμογών. Όλα αυτά μπορεί εύκολα να τα αντλήσει κανείς, μέσα από το Internet, χρησιμοποιώντας μια σειρά από απλές εντολές.

Πώς λειτουργεί

Η σύνδεση στο Internet δύο διαφορετικών κόμβων και δυο διαφορετικών υπολογιστών αντίστοιχα για να μπορέσουν να επικοινωνήσουν μεταξύ τους ασύμβατοι υπολογιστές, επιτυγχάνεται μέσω της χρήσης κάποιου πρωτοκόλλου επικοινωνίας, (ενός πρότυπου κοινών τρόπων λειτουργίας), που επιτρέπει σε δυο ενδεχομένως πολύ διαφορετικά συστήματα να επικοινωνήσουν μεταξύ τους. Αρχικά χρησιμοποιήθηκε το πρωτόκολλο **NCP**, που αργότερα αντικαταστάθηκε από το **TCP/IP** (Transmit Control Protocol / Internet Protocol).

Τα TCP/IP καθιερώθηκε ως πρωτόκολλο επικοινωνίας μεταξύ υπολογιστών στο Internet. Το **πρωτόκολλο επικοινωνίας** είναι ίσως και το βασικότερο στοιχείο του δικτύου, γιατί επιτυγχάνεται η επικοινωνία ακόμα και των ασύμβατων Ηλεκτρονικών Υπολογιστών. Η δουλειά του είναι να παραλαμβάνει τις πληροφορίες από τον υπολογιστή-αποστολέα, να τις ξεχωρίζει σε πακέτα, να τις διοχετεύει στην τηλεφωνική γραμμή και να τις ανασυντάσσει μετά στον υπολογιστή-παραλήπτη.

Τι ακριβώς όμως είναι το πρωτόκολλο **TCP/IP;** Το TCP/IP είναι το μέσο το οποίο αναλαμβάνει να στείλει κάποιες σελίδες και στην συνέχεια να ταξινομηθούν στον παραλήπτη. Έτσι ένα μέρος των πληροφοριών διοχετεύεται στον φάκελο **tcp**. Στη συνέχεια ο φάκελος αυτός τοποθετείται στον φάκελο **ip** και περνάει στο δίκτυο για την περαιτέρω μεταφορά του. Στο σημείο που παραλαμβάνεται, ένα πρόγραμμα **tcp** αναλαμβάνει τη συλλογή των φακέλων, και τη σωστή ταξινόμηση τους. Αν κάποια δεδομένα λείπουν, ζητάει από τον αποστολέα να τα αναμεταδώσει. Όταν όλα τα στοιχεία βρίσκονται στη σωστή σειρά και είναι πλήρη, περνάει τις πληροφορίες στο οποιοδήποτε πρόγραμμα χρησιμοποιείται από τον παραλήπτη.

Μπαίνοντας στο Internet θα πρέπει να **προσέξουμε** την τυχόν παραβίαση του password μας. Δεν θα θέλαμε να ανακαλύψουμε ότι λείπουν όλα τα αρχεία μας ή ότι έχουμε εξαντλήσει το χρόνο πρόσβασης σ΄ αυτό. Χρησιμοποιώντας μη προβλέψιμους κωδικούς που θα τους αλλάζουμε ανά τακτά χρονικά διαστήματα, είναι δύσκολο για κάποιο τρίτο χρήστη να ανακαλύψει τον κωδικό μας.

Βασικές υπηρεσίες του Internet

1. Μεταφορά αρχείων στο Internet - FTP

Μετά το ηλεκτρονικό ταχυδρομείο η πιο διαδεδομένη εφαρμογή του INTERNET ανάμεσα στους χρήστες του είναι το **ftp (File Transfer Protocol** - Πρωτόκολλο Μεταφοράς Αρχείων). Το ftp αποκαθιστά μια σύνδεση μεταξύ δύο υπολογιστών με σκοπό την ανταλλαγή αρχείων. Στους υπολογιστές που είναι συνδεδεμένοι στο INTERNET, υπάρχουν χιλιάδες αρχεία με εικόνες, κείμενα και εκτελέσιμα ακόμα προγράμματα, που μπορεί κάθε χρήστης να τα μεταφέρει στον τοπικό του υπολογιστή και να τα χρησιμοποιήσει δωρεάν. Το ftp αναλαμβάνει τη σύνδεση με τον απομακρυσμένο υπολογιστή και τη μεταφορά των δεδομένων χωρίς απώλειες και λάθη. Το μόνο που χρειάζεται να του δώσουμε είναι το όνομα της περιοχής ή τη διεύθυνση στο Internet του απομακρυσμένου υπολογιστή. Έτσι μέσα από το δικό του λειτουργικό περιβάλλον (που αναγνωρίζεται από το αρχικό εντολών **ftp>**) με εντολές : **ftp** *όνομα_περιοχής* στη γραμμή εντολών του UNIX επιτυγχάνεται η σύνδεση των απομακρυσμένων υπολογιστών. Απαραίτητος είναι κάποιος κωδικός εισόδου για να γίνει η σύνδεση. Επειδή κανένας χρήστης δεν είναι δυνατό να έχει κωδικό εισόδου σε όλους τους υπολογιστές από τους οποίους θα ήθελε να πάρει ή να δώσει αρχεία, έγινε το **ανώνυμο ftp**. Ο χρήστης δίνει σαν κωδικό εισόδου τη λέξη ***anonymous*** και σαν password τη διεύθυνση ηλεκτρονικού ταχυδρομείου του. Όταν συνδεθεί ο χρήστης, θα βρεθεί σε έναν κατάλογο απομακρυσμένου υπολογιστή, από όπου μπορεί να πάρει ή να δώσει αρχεία ή και να μεταφερθεί ακόμα και σε άλλους καταλόγους χρησιμοποιώντας εντολές του ftp.

Οι κυριότερες εντολές του ftp είναι οι εξής:

open , **(o)** *όνομα_περιοχής*. Αποκαθιστά μια σύνδεση ftp ανάμεσα στον τοπικό υπολογιστή και στον απομακρυσμένο υπολογιστή.

Close Τερματίζει τη σύνδεση ftp, χωρίς να τερματίζει το πρόγραμμα ftp.

Dir Εμφανίζει τα περιεχόμενα του τρέχοντος καταλόγου στον απομακρυσμένο υπολογιστή και περιλαμβάνει τα ονόματα αρχείων, τον κάτοχο και το μέγεθός τους.

cd *όνομα_καταλόγου.* Αλλάζει τον τρέχοντα κατάλογο στον απομακρυσμένο υπολογιστή.

get *όνομα_αρχείου.* Μεταφέρει το καθοριζόμενο αρχείο στον τοπικό υπολογιστή του χρήστη. (Εάν το επιθυμεί ο χρήστης, το αρχείο μπορεί να μεταφερθεί στον τοπικό υπολογιστή με διαφορετικό όνομα)

mget *όνομα-αρχείου_1 όνομα_αρχείου_2* ...Μεταφέρει ομάδα αρχείων στον τοπικό υπολογιστή του χρήστη.

put *όνομα_αρχείου.* Είναι η αντίστροφη εντολή της get και μεταφέρει το καθοριζόμενο αρχείο από τον τοπικό υπολογιστή στον απομακρυσμένο.

mput *όνομα-αρχείου_1 όνομα_αρχείου_2* ... Μεταφέρει ομάδα αρχείων από τον τοπικό υπολογιστή του χρήστη στον απομακρυσμένο.

Binary Δηλώνει ότι τα μεταφερόμενα αρχεία θα είναι δυαδικής μορφής. Με αυτόν τον τρόπο το λειτουργικό σύστημα δεν θα επέμβει κατά τη μεταφορά για να διορθώσει κάποιους χαρακτήρες που περιέχονται στο αρχείο, που θα τους θεωρούσε σαν χαρακτήρες ελέγχου.

Ascii Δηλώνει ότι τα μεταφερόμενα αρχεία είναι κείμενα. Το μειονέκτημα για τους Έλληνες είναι ότι λόγω της κωδικοποίησης ASCII 7 bit που εφαρμόζει το UNIX, δεν μεταφέρονται οι ελληνικοί χαρακτήρες, παρά μόνο οι λατινικοί. Για το λόγο αυτό, όταν ο χρήστης θέλει να μεταφέρει ένα ελληνικό κείμενο, χρησιμοποιεί τον binary τρόπο μεταφοράς αρχείων.

Status Εμφανίζει την τρέχουσα κατάσταση όλων των ρυθμίσεων του ftp.

help *όνομα_εντολής_ftp* Δίδει άμεση βοήθεια για τις εντολές του ftp.

Hash Εκτυπώνει χαρακτήρες # ανά τακτά διαστήματα στο τερματικό για τον έλεγχο λειτουργίας του τερματικού κατά τη μεταφορά δεδομένων.

Bye Τερματίζει το πρόγραμμα ftp και επιστρέφει σε εντολές UNIX.

Στο περιβάλλον των **windows 95** το ftp λειτουργεί με μορφή απλών παραθύρων και όλες οι εντολές δίνονται από το ποντίκι.

Αξίζει να προσέξουμε εδώ ότι σε κάθε μεταφορά αρχείου binary που κάνουμε από ένα anonymous ftp site στον υπολογιστή μας, υπάρχει κίνδυνος να μεταφέρουμε μαζί και κάποιο **virus (ιό)** που θα μας δημιουργήσει αργότερα προβλήματα στα αρχεία μας. Γι αυτό θα πρέπει να έχουμε πάντα εγκατεστημένα στον υπολογιστή μας **antiviruses.**

Παράδειγμα χρήσης ftp στο unix :

ftp>o

(to) 195.242.35.10

user: anonymous

password: user1@cc.uoi.gr

ftp>cd pub

ftp>bi

ftp>get όνομα αρχείου

ftp> bye

2. Ηλεκτρονικό ταχυδρομείο e-mail

Το ηλεκτρονικό ταχυδρομείο παρέχει τη δυνατότητα στους χρήστες να στέλνουν και να δέχονται μηνύματα μέσω ηλεκτρονικού υπολογιστή. Είναι η πιο δημοφιλής εφαρμογή του INTERNET και έχει γίνει είδος πρώτης ανάγκης στις σύγχρονες επιχειρήσεις. Το ηλεκτρονικό ταχυδρομείο λειτουργεί περίπου όπως το κανονικό ταχυδρομείο. Όταν ένας χρήστης στείλει ένα μήνυμα, ένα αρχείο που περιέχει αυτό το μήνυμα μετακινείται από τον έναν υπολογιστή του INTERNET στον άλλο μέχρι να φτάσει στον υπολογιστή προορισμού, όπου και αποθηκεύεται στο ηλεκτρονικό γραμματοκιβώτιο, έως ότου διαβαστεί από τον παραλήπτη. Το ηλεκτρονικό ταχυδρομείο δεν είναι μια καινούρια εφαρμογή. Τα συστήματα που υποστηρίζουν πολλούς χρήστες (UNIX, VAX/VMS κτλ) διαθέτουν ηλεκτρονικό ταχυδρομείο ως βασικό μέρος του συστήματος για επικοινωνία

μεταξύ των χρηστών τους. Με την ανάπτυξη όμως του INTERNET, το e-mail έγινε ένα ισχυρό εργαλείο που επιτρέπει την ηλεκτρονική επικοινωνία μεταξύ χρηστών όλου του κόσμου.

Χρήση του e-mail

Το ηλεκτρονικό ταχυδρομείο αποτελεί μια ανεξάρτητη εφαρμογή στον τοπικό υπολογιστή. Η εφαρμογή αυτή λέγεται **mail** και υπάρχει σε όλα τα μηχανήματα UNIX. Η δημιουργία νέων μηνυμάτων είναι μια διαδικασία τριών βημάτων.

Ο χρήστης γράφει:

1. το όνομα του παραλήπτη,
2. το μήνυμα και
3. το στέλνει.

Το όνομα ενός χρήστη στο INTERNET έχει τη μορφή

όνομα χρήστη@όνομα περιοχής.

Εάν λοιπόν θέλουμε να στείλουμε e-mail στο χρήστη ***maria*** που βρίσκεται στον υπολογιστή του Παν/μιου Ιωαννίνων (όνομα περιοχής) ***cc.uoi.gr,*** πληκτρολογούμε: **mail** ***maria@cc.uoi.gr.***

Η εφαρμογή mail ζητάει πρώτα το θέμα (subject) του μηνύματος και καλό είναι ο χρήστης να γράφει το θέμα του κάθε μηνύματος, ώστε να μπορεί ο παραλήπτης με μια ματιά να καταλάβει περί τίνος πρόκειται. Μετά από το θέμα, ο χρήστης γράφει το κυρίως μήνυμα πατώντας κάθε φορά [ENTER] για να αλλάξει γραμμή. Χρησιμοποιώντας το **[CTRL+C]** ακυρώνεται η σύνταξη του μηνύματος, αν για κάποιο λόγο αλλάξουμε γνώμη και δεν θελήσουμε να στείλουμε το μήνυμα στον παραλήπτη.

Όταν ολοκληρωθεί το μήνυμα, ο χρήστης για να το στείλει πατάει

[CTRL+D] ή

στην αρχή της γραμμής του κειμένου.

Η εφαρμογή mail έχει δικό της περιβάλλον εργασίας που ξεχωρίζει από το UNIX από το & και στο οποίο μπαίνει ο χρήστης πληκτρολογώντας **mail**. Το περιβάλλον αυτό επιτρέπει στο χρήστη να διαχειρίζεται τα μηνύματά του, να στέλνει δηλαδή, να διαβάζει να διαγράφει και να απαντάει σε αυτά.

Οι βασικές εντολές του mail είναι:

h

Εμφανίζει μια λίστα με εντολές και με τα μηνύματα που δεν έχει διαβάσει ακόμα ο χρήστης, κατά χρονολογική σειρά, καθώς και τον αποστολέα και το θέμα του κάθε μηνύματος. Το τρέχον μήνυμα είναι αυτό που έχει το σύμβολο > πριν τον αύξοντα αριθμό του. Για να διαβάσει ο χρήστης ένα μήνυμα, πληκτρολογεί απλώς τον αύξοντα αριθμό του.

r ή R

Απαντάει στο τρέχον μήνυμα. Η εφαρμογή mail βάζει αυτόματα στο όνομα του παραλήπτη, το όνομα του αποστολέα, και σαν θέμα βάζει το Re: *αρχικό θέμα*. Στη συνέχεια, ο χρήστης συντάσσει το μήνυμά του.

s *α/α όνομα_αρχείου*

Αν ο χρήστης επιθυμεί να σώσει το μήνυμα, τότε το αποθηκεύει με το όνομα που θα του δώσουμε δηλαδή *όνομα_αρχείου*.

d *α/α*

Διαγράφει το μήνυμα με τον καθοριζόμενο αύξοντα αριθμό.

t *α/α*

Τυπώνει το μήνυμα στην οθόνη.

q

Τερματίζει το πρόγραμμα mail αποθηκεύοντας τα μηνύματα που διάβασε ο χρήστης στο αρχείο *mbox* του χρήστη. Τα μηνύματα που δεν διαβάστηκαν, παραμένουν στο γραμματοκιβώτιο του συστήματος.

x

Τερματίζει το πρόγραμμα mail, χωρίς να αποθηκεύσει τα μηνύματα που διάβασε ο χρήστης. Τα μηνύματα παραμένουν στο γραμματοκιβώτιο του συστήματος.

Εάν ένα μήνυμα απευθύνεται σε μια ομάδα χρηστών τότε το όνομα του παραλήπτη δε θα είναι της μορφής ***όνομα_χρήστη@όνομα_περιοχής***, αλλά μόνο ***όνομα_λίστας*** και το μήνυμα θα το λάβουν όλα τα μέλη της λίστας.

Ανάγνωση του e-mail μας από τρίτους

Όπως και στις τηλεφωνικές γραμμές, έτσι και στα συστήματα όπως το Internet μπορεί να υπάρχει κάποιου είδους παρακολούθηση. Φυσικά, δεν είναι δυνατόν να διαβάζονται όλα μας τα μηνύματα από κάποιον τρίτο, αλλά κάποια από αυτά. Είναι λοιπόν καλό να μη στέλνουμε πληροφορίες υψηλής ασφάλειας μέσω του δικτύου ή αν είναι αυτό απαραίτητο να χρησιμοποιούμε την τακτική της κωδικοποίησης σε τέτοια έγγραφα. Η κωδικοποίηση αυτή, γνωστή και ως **encryption,** δίνει τη δυνατότητα στον αποστολέα να χρησιμοποιεί κάποιο κλειδί για να κλειδώνει ό,τι στέλνει, χωρίς το οποίο δεν μπορεί να το αποκωδικοποιήσει και να το διαβάσει κάποιος τρίτος, εκτός από τον παραλήπτη που θα πρέπει να είναι ενημερωμένος για το κλειδί που θα χρησιμοποιούμε όχι μέσω E - mail, αλλά με κάποιον άλλο τρόπο εκτός δικτύου.

3. Telnet

Με το πρόγραμμα telnet έρχονται σε επαφή οι χρήστες του δικτύου, με οποιουδήποτε υπολογιστή συνδεδεμένο στο INTERNET.

Πως δουλεύει το telnet

Η εντολή telnet χρειάζεται μόνο το όνομα του απομακρυσμένου υπολογιστή ή μια διεύθυνση **IP (Internet Protocol)** που προσδιορίζει ακριβώς τον υπολογιστή με τον οποίο θέλουμε να συνδεθούμε. Ο κάθε ένας από τις χιλιάδες συνδεδεμένους υπολογιστές έχει μία μοναδική διεύθυνση, η οποία καθορίζεται όταν ο υπολογιστής καταχωρηθεί στο Κέντρο Πληροφοριών Δικτύου INTERNET (INTERNET Network Information Center). Η διεύθυνση αυτή είναι ένας αριθμός των 32 bit και για να γίνει ευκολότερα κατανοητός από τους χρήστες είναι συνήθως γραμμένος σαν 4 αριθμοί χωρισμένοι με τελείες : π.χ. 193.92.4.7 (Πανεπιστήμιο Ιωαννίνων). Μια διεύθυνση IP προσδιορίζει την ταυτότητα ενός υπολογιστή

με μοναδικό τρόπο, που με τη μορφή τεσσάρων αριθμών χωρισμένων με τελείες είναι δύσκολο να απομνημονευθεί από τους ανθρώπους. Γι' αυτό το λόγο έχει αναπτυχθεί παράλληλα και μια υπηρεσία καταλόγου για την αναζήτηση ονομάτων. Η υπηρεσία αυτή και οι κανόνες που τη διέπουν είναι γνωστά σαν Σύστημα Ονομασίας Περιοχών (Domain Name System - **DNS**), όπου η περιοχή (domain) είναι μια επώνυμη ομάδα υπολογιστών.

Τα ονόματα των περιοχών αποτελούνται από μια σειρά ονομάτων που χωρίζονται με τελείες. Το πεδίο που βρίσκεται στο αριστερό μέρος του ονόματος είναι το όνομα υπολογιστή υπηρεσίας. Μετά το όνομα του υπολογιστή υπηρεσίας ακολουθούν τα ονόματα των περιοχών στις οποίες ανήκει με σειρά αύξουσας γενικότητας. Παραδείγματος χάρη, ο υπολογιστής με διεύθυνση orpheas.cc.uoi.gr έχει όνομα orpheas, ανήκει στην ομάδα του υπολογιστικού κέντρου (cc - Computer Center), που με τη σειρά του ανήκει στο Πανεπιστήμιο Ιωαννίνων (uoi - Univercity of Ioannina), το οποίο ανήκει στην Ελλάδα (gr - Greece).

Επομένως, για να χρησιμοποιήσουμε το telnet πρέπει να πληκτρολογήσουμε

telnet *διεύθυνση_IP* ή

telnet *όνομα_περιοχής*

π.χ. Telnet solon.cc.uoi.gr

Μόλις αποκατασταθεί μια σύνδεση telnet πρέπει ο χρήστης να δώσει ένα όνομα σύνδεσης (**login name**) και ένα **password** που να είναι έγκυρα για τον απομακρυσμένο υπολογιστή. Από εκεί και πέρα, ο χρήστης μπορεί να χρησιμοποιήσει κανονικά τα στοιχεία του απομακρυσμένου υπολογιστή.

Για να διακοπεί η σύνδεση, ο χρήστης δίνει την εντολή **logout** , **exit** και επιστρέφει στη γραμμή εντολών του τοπικού του υπολογιστή.

Οι κύριες εντολές του telnet είναι οι παρακάτω:

open *όνομα-περιοχής*

Η εντολή ανοίγει μια και μόνο σύνδεση με τον απομακρυσμένο υπολογιστή.

close

Διακόπτει την τρέχουσα σύνδεση και επιστρέφει στην κατάσταση εντολών του telnet.

quit

Διακόπτει την τρέχουσα σύνδεση και τερματίζει την εκτέλεση του telnet.

status

Εμφανίζει το όνομα του υπολογιστή με τον οποίο είναι συνδεδεμένος ο χρήστης, καθώς και την τρέχουσα κατάσταση λειτουργίας του telnet.

set escape *χαρακτήρας*

Αλλάζει το χαρακτήρα διαφυγής του telnet.

? *εντολή*

Δίδει άμεση βοήθεια για την εντολή αυτή του telnet.

Παράδειγμα:

Telnet>o

(to) solon

Login: maria

password: maria@cc.uoi.gr

Usenet

Το USENET (INTERNET NEWS) είναι ο χώρος που οι χρήστες του INTERNET μπορούν να ανταλλάξουν μηνύματα, απόψεις ακόμη και να κάνουν επαφές πάνω σε θέματα κοινού ενδιαφέροντος. Είναι τόσα πολλά τα θέματα που καλύπτονται, ώστε ο κάθε χρήστης θα βρει εύκολα κάτι που να τον ενδιαφέρει. Newsgroup λέγεται κάθε ομάδα μηνυμάτων με κοινό θέμα και μπορεί να έχει και υποομάδες που αναφέρονται σε ειδικότερα θέματα. Τα μηνύματα αυτά μεταφέρονται από υπολογιστή σε υπολογιστή του INTERNET αυτόματα και αυτό έχει σαν αποτέλεσμα τη μετακίνηση τεραστίου μεγέθους αρχείων, για αυτό το λόγο και οι υπεύθυνοι πολλών υπολογιστών είτε δεν αποθηκεύουν τα μηνύματα του USENET καθημερινά, αλλά αποθηκεύουν μόνο ένα μέρος των newsgroups. Οι επίσημες ομάδες μηνυμάτων - newsgroups - του USENET είναι επτά και περιέχουν πολλές υποομάδες. Οι ομάδες αυτές είναι:

News Θέματα σχετικά με το USENET.

Sci Ομάδες μηνυμάτων που αναφέρονται σε επιστημονικά θέματα.

Soc Ομάδες μηνυμάτων που ασχολούνται με κοινωνικά θέματα

Rec Ομάδες μηνυμάτων που ασχολούνται με θέματα ψυχαγωγικού περιεχομένου

Comp Ομάδες μηνυμάτων που αναφέρονται σε θέματα σχετικά με τους υπολογιστές

Talk Ανοιχτές συζητήσεις για οποιοδήποτε θέμα.

Misc Διάφορα θέματα

Σύνδεση μέσω Internet

Εκτός από εταιρείες και οργανισμούς που χρησιμοποιούν το Internet για την ανταλλαγή πληροφοριών και προγραμμάτων, οι μαθητές και οι εκπαιδευτικοί χρησιμοποιούν το Internet για πρόσβαση σε εκπαιδευτικές πηγές (π.χ. on-line εγκυκλοπαίδειες). Έτσι, οι μαθητές με τη χρήση του Internet αποκτούν περισσότερο ενδιαφέρον για γνώση και δουλειά. Απαραίτητη προϋπόθεση για τη σύνδεση με το δίκτυο είναι να έχουμε έναν υπολογιστή, σειριακή κάρτα, modem (Κωδικοποιητή - Αποκωδικοποιητή), τηλεφωνική γραμμή, Software επικοινωνίας, Account (λογαριασμό στον υπολογιστή κόμβο - host π.χ. Πανεπιστήμιο). Αξίζει να σημειωθεί ότι :

1. Μπορεί να χρησιμοποιηθεί οποιοσδήποτε ηλεκτρονικός υπολογιστής για τη σύνδεσή μας με το Internet.
2. Το modem. Βασικό χαρακτηριστικό του modem είναι η ταχύτητά του. Υπάρχουν ταχύτητες από 1200 έως 19200 bps και ακόμη 64000 bps που χρησιμοποιούνται σε επαγγελματικές συνδέσεις ('leased lines'). Άλλα χαρακτηριστικά τους είναι: τα πρωτόκολλα επικοινωνίας που χρησιμοποιούν, η διόρθωση των λαθών που κάνουν, ακόμα και η διόρθωση της ποιότητας της γραμμής μέχρι και 95%.

Η σύνδεση στο Internet γίνεται με κάποια από τις παρακάτω κατηγορίες:

1.Άμεσες μόνιμες συνδέσεις. Οι συνδέσεις αυτές γίνονται απευθείας σε κάποιο δίκτυο TCP/IP που με τη σειρά του συνδέεται απευθείας στο Internet π.χ. Πανεπιστήμια και μεγάλες εταιρίες.

2.Άμεσες προσωρινές συνδέσεις. Με λογισμικό που να υποστηρίζει Point - to - Point Protocol, (ή PPP) ή και το παλαιότερο πρωτόκολλο Serial Line IP (SLIRP) μέσω modem μπορεί να συνδέεται κανείς άμεσα στο Internet.

3.Συνδέσεις από τερματικά μέσω τηλεφώνου.

4.Συνδέσεις μόνο για e - mail.

World Wide Web (WWW)

Είναι ένα δικτυακό σύστημα server-client, γνωστό και ως **W3** ή **Web**. Θεωρείται σαν ένα μέσο ή **project** που φιλοδοξεί να προσφέρει ένα **standard interface** για τη διαχείριση του τεράστιου όγκου πληροφοριών στο Internet. Το εργαλείο αυτό είναι πρακτικά ένα πρωτόκολλο **hypermedia**, που σχεδιάστηκε και αναπτύχθηκε στο εργαστήριο σωματιδιακής φυσικής στο **CERN** της Ελβετίας. Το WWW υλοποιήθηκε στο πλαίσιο μιας προσπάθειας ενοποίησης της διαθέσιμης πληροφορίας και της κατασκευής ενός εργαλείου που ελαχιστοποιούσε την ανάγκη χρήσης διαφορετικών προγραμμάτων για την αξιοποίηση των πηγών πληροφορίας του Internet.

Η ιστορία του WWW άρχισε το **Μάρτιο του 1989** και η αρχική πρόταση γράφτηκε από τον **Tim Berners Lee**, με κύριο στόχο την ανταλλαγή ιδεών μεταξύ των μελών της επιστημονικής κοινότητας *(σχετικά με τη φυσική υψηλών ενεργειών*. Η υλοποίηση του project ξεκίνησε το **Νοέμβριο του 1990** με την κατασκευή του πρώτου πρωτοτύπου σε περιβάλλον **NeXT**. Μέχρι τα Χριστούγεννα του ίδιου έτους ήταν έτοιμοι δύο clients, μέσω των οποίων έγινε η επίδειξη πρόσβασης σε κείμενα **hypertext** . Ένα χρόνο αργότερα, μέσω ενός **newsletter** του CERN, έγινε γνωστή η ύπαρξη του WWW. Η **NCSA** *(National Center for Supercomputing Applications)*, ξεκίνησε την κατασκευή ενός **interface** για το WWW, δηλαδή το **Mosaic**, που θα κάλυπτε όλες τις υπολογιστικές πλατφόρμες και το οποίο ονομάστηκε. Οι αρχικές εκδόσεις του Mosaic αναπτύχθηκαν σε X-Windows (UNIX), Microsoft Windows (PC 's) και Macintosh, ενώ η ελεύθερη διάθεση του μέσω του Internet συνέβαλε σημαντικά στην εξάπλωση του WWW. Ακολούθησαν εκδόσεις για NeXT και Amiga, καλύπτοντας έτσι την πλειοψηφία των λειτουργικών συστημάτων που χρησιμοποιούνται σήμερα. Τον **Ιανουάριο του 1993** υπήρχαν πάνω από 50 WWW servers σε όλο το Internet και η διακίνηση πληροφοριών μέσω τηλεπικοινωνιακών γραμμών *(traffic)*, διπλασιαζόταν κάθε 4 μήνες. Σήμερα υπάρχουν πάνω από 10.000 WWW servers σε όλο το Inetrnet, ενώ το ποσοστό των MBytes που διακινούνται μέσω αυτών αυξάνεται συνεχώς.

Ο τρόπος λειτουργίας του WWW

Το WWW είναι ένα εργαλείο αναζήτησης και παρουσίασης πληροφοριών του Internet, που στηρίζεται στην απλή αρχή του **client / server**. Τί είναι όμως οι **client / server** ;

Server είναι ένα πρόγραμμα που διευκολύνει την επικοινωνία ενός χρήστη με άλλους υπολογιστές στο δίκτυο (π.χ. Telnet) και χρησιμοποιεί συγκεκριμένη γλώσσα. Για κάθε πρόγραμμα **server** υπάρχει ένα αντίστοιχο **client** πρόγραμμα. Με τα προγράμματα αυτά οι χρήστες φτάχνουν τις ερωτήσεις τους προς τον server σε αναγνώσιμη μορφή (π.χ. netscape). Γνωστοί multimedia clients είναι τα Mosaic και Netscape. **Οι clients στο web λέγονται browsers** και δουλεύουν σε οποιονδήποτε υπολογιστή. Server ή δαίμονας(daemon) είναι ένα πρόγραμμα που σκοπό έχει να κάνει προσιτές τις διάφορες πηγές πληροφοριών σε άλλους υπολογιστές. Οι περισσότεροι servers τρέχουν σε μηχανές Unix αλλά μπορούν να τρέχουν και σε PCs και Macs. Οι servers (daemons) επικοινωνούν με το hypertext transmission protocol **(http)** και περιέχουν κυρίως **text files** με **links** στον ίδιο ή άλλο server. Συνοπτικά η λειτουργία του www είναι η ακόλουθη:

1. ο χρήστης επιλέγει κάποιο κομμάτι hypertext που είναι σε κάποια διεύθυνση στο Internet π.χ." Ιστορία της Ευρώπης". Το κείμενο αυτό μπορεί να είναι συνδεδεμένο και με κάποιο άλλο κείμενο π.χ. Ιστορία της Ελλάδας.
2. ο web client συνδέεται στο συγκεκριμένο server που έχει τη διεύθυνση που έδωσε ο χρήστης στο συγκεκριμένο δίκτυο και ζητάει αυτό το κείμενο.
3. ο server στέλνει το κείμενο (π.χ."Ιστορία της Ευρώπης") και ό,τι άλλο εμπεριέχεται στο κείμενο αυτό (π.χ. Ιστορία της Ελλάδας), στον client.
4. ανάλογα με τη μορφή του αρχείου που λαμβάνει ο client επιλέγει και το ανάλογο πρόγραμμα για να το παρουσιάσει π.χ. ghostview.

Η γλώσσα στο www

Η πιο διαδεδομένη γλώσσα που χρησιμοποιεί το web είναι τέτοια ώστε να μπορεί να δημιουργεί και να αναγνωρίζει κείμενα σε hypermedia και ονομάζεται HTML (Hypertext Markup Language). Τα αρχεία στο HTML έχουν στο τέλος το διακριτικό .html ή .htm.

URL(Uniform Resource Locators)

Σχεδόν κάθε αρχείο ή υπηρεσία στο Internet αντιπροσωπεύεται από ένα url. Μια url είναι μία παράσταση της μορφής:

Μέθοδος πρόσβασης :// διεύθυνση του υπολογιστή που έχει τα δεδομένα. όνομα αρχείου

Παραδείγματα:

1. http://info.cern.ch/hypertext/
2. file://localhost/usr/local/html/bioss.html
3. ftp://ftp.bioss.sari.ac.uk
4. gopher://pollux.ucs.ed.ac.uk
5. http://www.yahoo.com
6. http://www.altavista.com
7. http://www.uoi.gr

Μερικές χρήσιμες διευθύνσεις στο Internet

1. http://www.forthnet.gr
2. http://www.microsoft.com
3. http://www.yahoo.com
4. http://www.enet.gr
5. http://info.cern.ch/hypertext

15 ΕΠΙΚΟΙΝΩΝΙΕΣ

Ο όρος επικοινωνία χρησιμοποιείται για να εκφράσει τη σχέση, τη συνάφεια, τη συνεννόηση, την ανταλλαγή ή τη μετάδοση μιας πληροφορίας μεταξύ ατόμων, ομάδων, ή λαών. Η επικοινωνία κατ' αρχάς γινόταν με λόγο ή με κίνηση και αργότερα έγινε μέσα από πιο εξελιγμένες μορφές όπως η γραφή και η τηλεφωνία.

Στην αρχαιότητα χρησιμοποιούνταν πρώτα οι αγγελιοφόροι, οι φωτιές και οι δάδες για τη διάδοση των πληροφοριών και αργότερα άρχισε η γραπτή μετάδοσή τους. Τέλος, στο 19° αιώνα που εφευρέθηκαν οι ηλεκτρικές μέθοδοι, άρχισαν και οι ενσύρματες επικοινωνίες. Συγκεκριμένα, το 1837 ο Morse κατασκεύασε την ηλεκτρομαγνητική τηλεγραφική συσκευή, το 1876 ο Bell εφεύρε το τηλέφωνο και το 1895 ο Poppof έκανε τον ασύρματο.

Στα μέσα του 20ου αιώνα παρατηρούμε ότι τα περισσότερα μέσα επικοινωνιών ανήκουν κυρίως στο κράτος, προσφέροντας υπηρεσίες ταχυδρομείου και τηλεφωνικών επικοινωνιών.

Σήμερα, βλέπουμε ότι πολλές ιδιωτικές εταιρείες έχουν μεγάλο μερίδιο στο χώρο των επικοινωνιών προσφέροντας τόσο υπηρεσίες ταχυδρομείου και τηλεφώνου όσο και υπηρεσίες μεταβίβασης δεδομένων.

Υπάρχουν πολλά είδη επικοινωνιών όπως:

- Διμερής επικοινωνία (η μετάδοση και η λήψη γίνεται σε δύο κατευθύνσεις)
- Κυκλική επικοινωνία (επικοινωνία πολλών διευθύνσεων)
- Επικοινωνία μέσω διαύλου

Μέσα επικοινωνιών

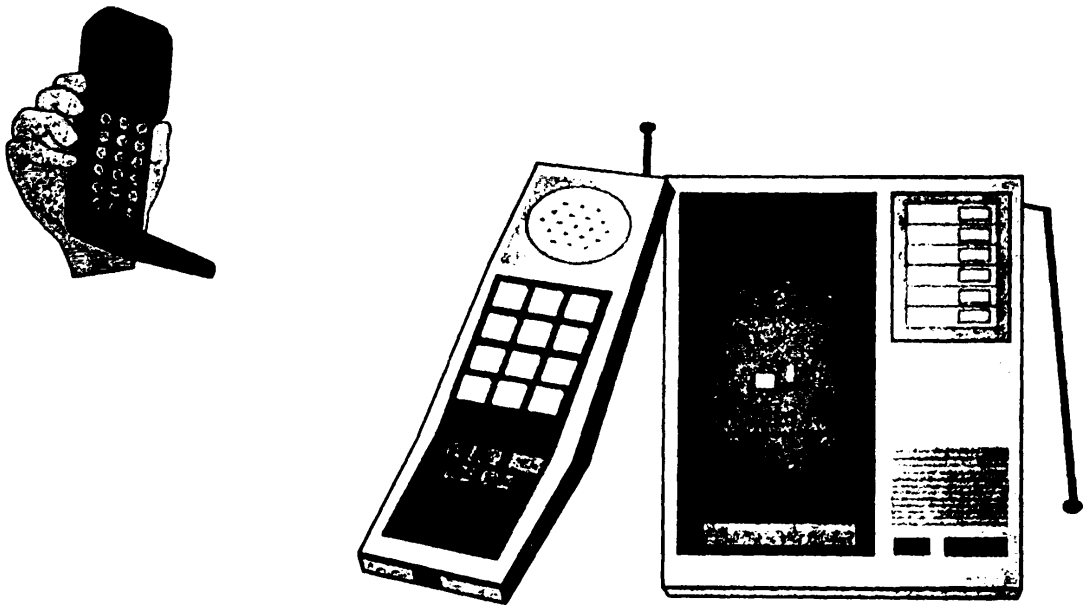
Στις επικοινωνίες χρησιμοποιούμε τα ακόλουθα μέσα:

Τηλέτυπος: Είναι μία τηλεγραφική συσκευή εκτύπωσης κειμένων με μορφή διάτρητης ταινίας που διακρίνεται από τη μη συνεχή λειτουργία πομπού και δέκτη. Η επικοινωνία με τηλέτυπο (Telex) πρωτοεμφανίστηκε στη δεκαετία του 1930 και την εφάρμοσαν οι Αγγλία, Γερμανία, Αμερική (ΗΠΑ). Στην λειτουργία της εφαρμόστηκαν δύο τρόποι: αυτός με ακουστικές συχνότητες συνεχούς και αυτός με συχνότητες εναλλασσομένου ρεύματος. Στην πρώτη περίπτωση συνδέονται οι συσκευές μεταξύ τους με μια ξεχωριστή δισύρματη γραμμή και στη δεύτερη χρησιμοποιείται το τηλεφωνικό δίκτυο. Η διάδοσή της οφείλεται κύρια στην απλή λειτουργία της χωρίς την απαραίτητη επίβλεψή της από το χειριστή.

Τηλέφωνο: Είναι μια απλή και σε όλους γνωστή συσκευή που χρησιμεύει στη μετάδοση του λόγου. Αποτελείται από ένα μικρόφωνο και ένα τηλέφωνο που και τα δύο μαζί συνενώνονται στο ακουστικό και το σύστημα πλήκτρων. Τον τελευταίο καιρό της έχει προστεθεί και η δυνατότητα της παράλληλης προβολής εικόνας μαζί με τη συνομιλία (εικονοτηλέφωνο). Η επικοινωνία με το τηλέφωνο γίνεται με ηλεκτρικά σήματα και σε οποιαδήποτε απόσταση. Το τηλεφωνικό δίκτυο συνδέει διάφορα τηλέφωνα μεταξύ τους και ψάχνει να βρει με ειδικά σήματα τη συντομότερη διαδρομή ανάμεσα στον καλούντα και στον καλούμενο. Θεωρείται ότι στις μέρες μας είναι από τα πιο μαζικά και λειτουργικά μέσα επικοινωνίας. Οι Η/Υ προσφέρουν τις υπηρεσίες τους στην τηλεφωνία με το να βοηθούν να σταλεί το κάθε μήνυμα στον προορισμό του ταχύτερα και ασφαλέστερα.

Στα επόμενα χρόνια θεωρείται ότι το τηλεφωνικό δίκτυο θα εξελιχτεί σε ένα ενοποιημένο επικοινωνιακό δίκτυο και πιθανότατα θα είναι της μορφής ISDN ή BISDN. (Εικόνα 15.1)

Κινητή Τηλεφωνία (Cellular phones): Το 1980 άρχισε να εμφανίζεται το τηλέφωνο στο αυτοκίνητο. Ο οδηγός του αυτοκινήτου με τον τρόπο αυτό, μπορούσε να επικοινωνήσει τηλεφωνικά με όποιο άτομο ήθελε τηρώντας πάντα τους κανόνες ασφαλείας. Σήμερα η χρήση του είναι ευρέως διαδεδομένη και η μορφή του τηλεφώνου συνεχώς βελτιώνεται καθώς και οι προσφερόμενες υπηρεσίες του. Τελευταία υπάρχει και η δυνατότητα αποστολής μηνυμάτων σε κινητό τηλέφωνο μέσω του διαδικτύου καθώς και η σύνδεσή του μέσω του διαδικτύου.



Εικόνα 15.1 Τηλέφωνα

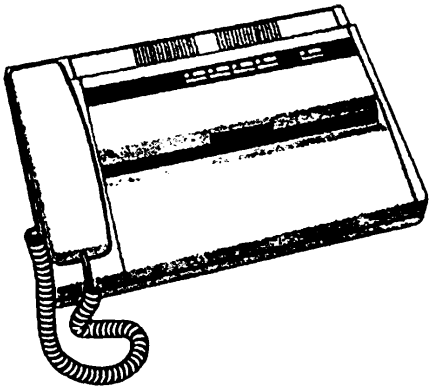
Modems: Η ονομασία τους προέρχεται από τις λέξεις modulation - demodulation που σημαίνει γενικότερα ότι σ' ένα κύμα ή σήμα του αλλάζω τη μορφή. Πρωτοχρησιμοποιήθηκαν για να μεταβιβάζουν δεδομένα από Η/Υ σε τηλεφωνικές γραμμές και αντίστροφα. Για το λόγο αυτό, όταν συνδέουμε έναν υπολογιστή στο διαδίκτυο τότε ανάμεσα από τον υπολογιστή και τη γραμμή του ΟΤΕ παρεμβάλλεται ένα modem που σκοπό έχει να μετατρέπει το δυαδικό σήμα του Η/Υ σε αναλογικό του ΟΤΕ και αντίστροφα. Η ευρεία χρήση τους άρχισε το 1960 περίπου και συνεχίζεται στις μέρες μας. Η ταχύτητα μεταβίβασης των δεδομένων συνεχώς μεγαλώνει και γι αυτό παρατηρούμε ότι Modems που ήταν φτιαγμένα πριν δύο χρόνια να θεωρούνται σήμερα παλιά. Το σημαντικότερο με τα Modems είναι ότι μπορούν να συνδεθούν με οποιοδήποτε Η/Υ (από απλό-φτηνό έως ακριβό Η/Υ).

Multiplexing: Επιτρέπει σε πολλαπλά πακέτα ηλεκτρονικών πληροφοριών να μεταβιβάζονται μέσω της ίδιας σύνθεσης. Αναλυτικότερα θα το βλέπαμε ως εξής: πολλοί ποδηλάτες να τρέχουν σ' ένα στενό δρόμο και να αντιμετωπίζουν σοβαρά προβλήματα στην ελεύθερη κίνησή τους, με συνέπεια να δημιουργείται ένα κυκλοφοριακό χάος. Αν όμως οι ποδηλάτες μπουν σ' ένα φορτηγό που κινείται στον ίδιο δρόμο και μεταφερθούν κατά ομάδες τότε κινούνται όλοι μαζί πιο γρήγορα. Εδώ τη θέση του φορτηγού την έχει το Multiplexing.

Videotex: Απευθύνεται στο πλατύ κοινό και για να λειτουργήσει χρειάζεται: - μία συσκευή τηλεφώνου
- μία συσκευή τηλεόρασης
- ένας adaptor που παρεμβάλλεται μεταξύ του τηλεφώνου και της τηλεόρασης. Το κόστος εγκατάστασης είναι μεγάλο, διότι πέρα από τα όσα προαναφέραμε, απαιτείται να δημιουργηθεί και ένα κέντρο για τη μεταβίβαση των πληροφοριών, που θα το αποτελεί ένας mainframe ή ένας mini υπολογιστής στον οποίο θα υπάρχει μια βάση δεδομένων. Σύντομα ο ΟΤΕ θα είναι σε θέση να έχει ένα κεντρικό σύστημα videotex. Τα συστήματα αυτά μπορούν να είναι είτε παθητικά είτε διαλογικά. Στα παθητικά (Non interactive videotex), οι χρήστες δεν θα έχουν τη δυνατότητα επιλογής των σελίδων. Οι σελίδες στην περίπτωση αυτή θα είναι σαν αυτές της τηλεόρασης. Στα διαλογικά (interactive) οι χρήστες μπορούν να επιλέξουν μέσα από ένα κατάλογο τις πληροφορίες που θέλουν να δουν. Οι συνδρομητές θα βλέπουν στην οθόνη με την οποία θα είναι συνδεδεμένοι πληροφορίες καταλόγου ΟΤΕ, καθώς και τηλεπαιγνίδια, τηλεαγορές και γενικές άλλες πληροφορίες.

Teletext: Εχει τη δυνατότητα μετάδοσης σειράς από εικόνες σε κείμενα ή και γραφικά χωρίς τη δυνατότητα αλληλεπίδρασης. Το σύστημα αυτό υπάρχει σε αρκετούς σταθμούς τηλεόρασης.

Telefax: (Fax), (Τηλεομοιοτυπία) Είναι μία συσκευή που δίνει τη δυνατότητα ανταλλαγής γραπτών κειμένων, σχεδίων, εικόνων σε ελάχιστο χρόνο μεταξύ των συνδρομητών που διαθέτουν τις αντίστοιχες συσκευές τηλεομοιοτυπίας. Συνδέεται όπως μια απλή συσκευή του ΟΤΕ πάνω στο δίκτυο του ΟΤΕ (Εικόνα 15.2).



Εικόνα 15.2 Fax.

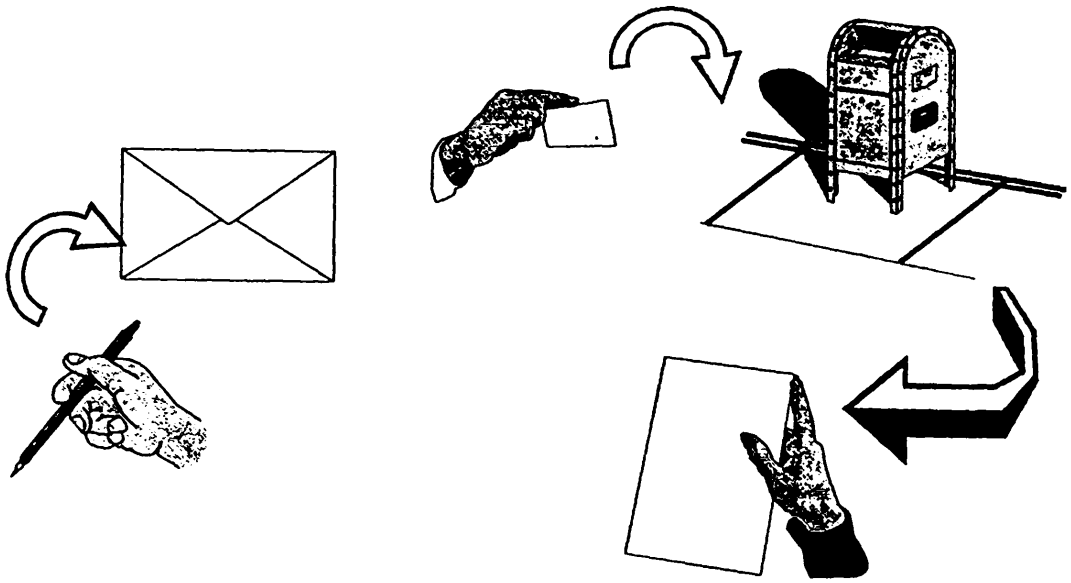
Τηλεπικοινωνίες: Οι τηλεπικοινωνίες λειτουργούν στη βάση απλών αρχών, δηλαδή χρησιμοποιούν το ηλεκτρικό ρεύμα,τα ραδιοκύματα, τις φωτεινές ακτίνες και μεταφέρουν τα μηνύματα. Οι Η/Υ όμως που έχουν δυαδικά στοιχεία 0-και-1, υπερέχουν των αναλογικών μέσων μεταφοράς, επειδή υπάρχει πάντα μικρότερη πιθανότητα λάθους κατά τη μεταφορά των δεδομένων.

Teletel

Στα μέσα της δεκαετίας του 1980, στο Παρίσι ειδικότερα, οι συνδρομητές του τηλεφώνου της Γαλλίας συνδέθηκαν "on line" με Η/Υ, τα λεγόμενα Minitel και κατ' αυτόν τον τρόπο είχαν την ευκαιρία να απολαμβάνουν εξυπηρετήσεων μέσω Η/Υ όπως για παράδειγμα να ελέγχουν τους λογαριασμούς τους στις τράπεζες, να αγοράζουν ρούχα, να ενημερώνονται από τα νέα των εφημερίδων, να αφήνουν μηνύματα σε φίλους που είχαν και αυτοί ηλεκτρονικό γραμματοκιβώτιο (electronic mailbox) .
Η πιο δημοφιλής απ' όλες τις παραπάνω χρήσεις του Minitel ήταν ο λεγόμενος διάλογος "Dialog", βάσει του οποίου οι συνδρομητές μπορούσαν να στέλνουν και να παίρνουν μηνύματα. Αυτό το σύστημα επικοινωνιών ονομάστηκε Teletel.

Δίκτυα και υπηρεσίες της Τηλεπληροφορικής

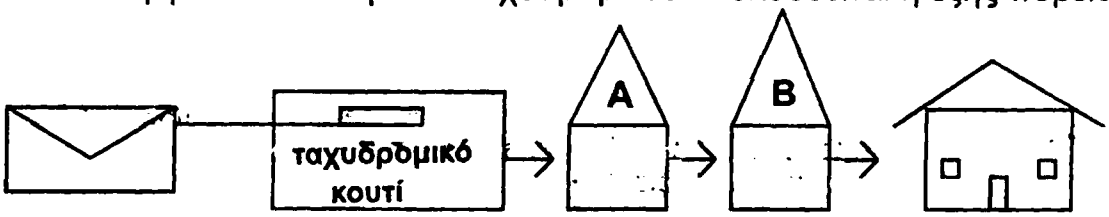
Ο άνθρωπος επικοινωνεί με το περιβάλλον του με διάφορους τρόπους. Αν προσπαθήσουμε να τοποθετήσουμε με κάποια χρονική σειρά τους τρόπους επικοινωνίας των ανθρώπων, θα δούμε ότι αρχικά αυτό γινόταν με τα νεύματα και το λόγο, αργότερα με την ανταλλαγή επιστολών (Εικόνα 15.3), στη συνέχεια με την τηλεγραφία-τηλεφωνία και πιο σύγχρονα με το Ηλεκτρονικό ταχυδρομείο (MHS) και την Ηλεκτρονική μηνυματοδοσία.



Εικόνα 15.3. Ανταλλαγή επιστολών.

Message Handling System (MHS)

Για να γίνει πιο κατανοητό το πως λειτουργεί το MHS θα γίνει η αντίστοιχη συσχέτιση με το γνωστό σε όλους ταχυδρομικό σύστημα. Για την ανταλλαγή επιστολών μέσω ταχυδρομείου ακολουθείται η εξής πορεία:



Διάγραμμα 1. Διάγραμμα πορείας ταχυδρομικών επιστολών

δηλαδή παίρνει κάποιος μια επιστολή, την τοποθετεί σε ένα ταχυδρομικό κουτί, μετά αυτή πηγαίνει στο κέντρο διαλογής Α διευκρινίζεται η πόλη του παραλήπτη, κατόπιν πηγαίνει στο κέντρο διαλογής Β της πόλης που μένει ο παραλήπτης και τέλος ο ταχυδρομικός διανομέας την πηγαίνει στο σπίτι του παραλήπτη. Με αυτό τον τρόπο λειτουργεί σε γενικές γραμμές το ταχυδρομείο. Κατ' αντιστοιχία στο MHS έχουμε την ακόλουθη πορεία για τη διαβίβαση του μηνύματος από τον ένα χρήστη στον άλλο:

Ο UA 1 διαβιβάζει μήνυμα μέσω των®MTA 1, και MTA 2,® στον αποδέκτη UA 2. όπου:
UA= user agent software και
MTA= message transfer agent.
Με τον τρόπο αυτό παρατηρούμε ότι για τη μεταφορά ενός μηνύματος από τον χρήστη 1 στο χρήστη 2 είναι απαραίτητος ο MTA. Η σημερινή τεχνολογία έχει στρέψει το ενδιαφέρον της στην τελειοποίηση όσο το δυνατό των MTA και UA.

Ηλεκτρονικό Ταχυδρομείο

Το Ηλεκτρονικό ταχυδρομείο αποστέλλει, λαμβάνει και τυποποιεί τα μηνύματα. Το πως λειτουργεί το Ηλεκτρονικό ταχυδρομείο βασίζεται στη λειτουργία του απλού ταχυδρομείου. Με τον τρόπο αυτό ο κάθε χρήστης έχει έναν User account και ένα User Mail-box η δε διαδικασία μετάδοσης του μηνύματος, μοιάζει με εκείνη των απλών επιστολών που διακινούνται μέσω ταχυδρομείου, δηλαδή:
από ..... προς ..... (κείμενο .....).
Οι προϋποθέσεις μιας σωστής επικοινωνίας εστιάζονται στα εξής σημεία:
α) Να υπάρχουν οι ίδιες εξυπηρετήσεις και στα δύο άκρα (δηλαδή να υπάρχει Η/Υ αποστολέα - Η/Υ παραλήπτη) και
β) Τα διαφορετικά πρότυπα που υπήρχαν παλαιότερα σε κάθε κατασκευαστή προϊόντων Ηλεκτρονικού Ταχυδρομείου (MHS - Message Handling System), να γίνονται όσο το δυνατόν κοινώς αποδεκτά (x400, x500).
Τα πλεονεκτήματα της χρήσης του είναι μεταξύ των άλλων:
η σημαντική οικονομία χρόνου,
το μικρό κόστος,
η αξιοπιστία και
η αμεσότητα στη επικοινωνία.
Με το ηλεκτρονικό ταχυδρομείο ο χρήστης μπορεί να παραλάβει τα μηνύματά του από τον υπολογιστή του εάν και όποτε αυτός το θελήσει. Με τον τρόπο αυτό ο χρήστης εξασφαλίζει μια ασύγχρονη επικοινωνία με κάποιον άλλο χρήστη.
Στα συστήματα αυτά της διαχείρισης των μηνυμάτων (MHS - Message Handling System), ενσωματώνονται τώρα και τα συστήματα τηλεομοιοτυπίας, videotex, κ.α.

Ηλεκτρονική μηνυματοδοσία

Η ηλεκτρονική μηνυματοδοσία είναι ένα σύνολο εξυπηρετήσεων που παρέχονται στους χρήστες για την ανταλλαγή μεταξύ τους:

Διαπροσωπικών μηνυμάτων
Μηνυμάτων EDI (Electronic Data Interchange)
Μηνυμάτων Τηλετυπίας
Μηνυμάτων Τηλεεικονογραφίας.

Οι χρήστες για να επικοινωνήσουν μεταξύ τους συνδέονται μέσω: Τηλεφωνικού δικτύου, HELLASPAC, TELEX ή ISDN.

Δίκτυα επικοινωνιών - δεδομένων HELLASPAC

Στον τομέα της τηλεπληροφορικής τα δίκτυα επικοινωνιών έχουν ανοίξει μια νέα εποχή. Ενα δημόσιο δίκτυο μετάδοσης και μεταγωγής πακέτων δεδομένων, που επιτρέπει τις επικοινωνίες δεδομένων μεταξύ Η/Υ και των τερματικών διατάξεών τους στην Ελλάδα και στο Εξωτερικό, είναι το HELLASPAC.

Η σύνδεση του χρήστη με το κέντρο του Hellaspac γίνεται είτε από μόνιμη σύνδεση, είτε μέσω του τηλεφωνικού δικτύου και απευθύνεται κυρίως σε επιχειρήσεις, δημόσιες υπηρεσίες, τράπεζες, χρηματιστήριο, εκπαιδευτικά και ερευνητικά ιδρύματα .

ISDN

Το ISDN (Integrated Services Digital Network) είναι η υψηλής ταχύτητας και ποιότητας επικοινωνία που επιτρέπει να χρησιμοποιούνται ταυτόχρονα ήχος, εικόνα, κείμενο και δεδομένα (data), (Διάγραμμα 2). Το ISDN προσφέρει ενοποίηση της επικοινωνίας, ταχύτητα, αξιοπιστία επικοινωνίας, ευελιξία (λόγω του ότι μπορεί να υποστηρίζει διαφορετικούς τύπους επικοινωνίας με μια μόνο σύνδεση), οικονομία, διασυνεργασία με άλλα δίκτυα και εναρμόνιση σύμφωνα με τα διεθνή πρότυπα (Euro-ISDN). Βασικό του χαρακτηριστικό είναι η ταχύτητα καθώς και η ευκολία με την οποία μπορούν οι χρήστες να το προσπελάσουν.

Περιγραφή του Euro-ISDN:

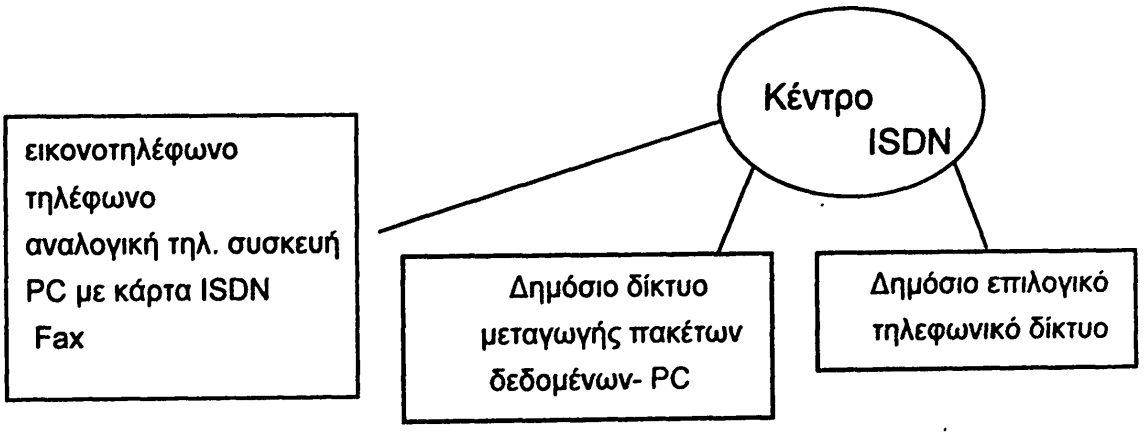
1. Τρόποι σύνδεσης:

- Βασική πρόσβαση (Basic Rate Access-BRA) Δύο κανάλια Β των 64kb/s και ένα κανάλι 1D των 16kb/s (2B+D).
- Πρωτεύουσα πρόσβαση (Primary Rate Access-PRA) Με 30 κανάλια 64kb/s και ένα κανάλι 1D των 64kb/s (30B+D).

1. Τι χρεάζεται για την σύνδεση;

Για τη βασική πρόσβαση θα πρέπει να έχουμε μόνο τις απαραίτητες τερματικές συσκευές ενώ,
για την πρωτεύουσα πρόσβαση εκτός από τα τερματικά, θα πρέπει να αναβαθμιστεί το συνδρομιτικό κέντρο σε ISDN.

2. Υπηρεσίες του ISDN
   - τηλεφωνία 3.1KHz και 7KHz
   - τηλεομοιοτυπία (telefax)
   - τηλεκειμενογραφία (videotex)
   - εικονοτηλεφωνία (video telephony)
   - μεταφορά αρχείου (eurofile)



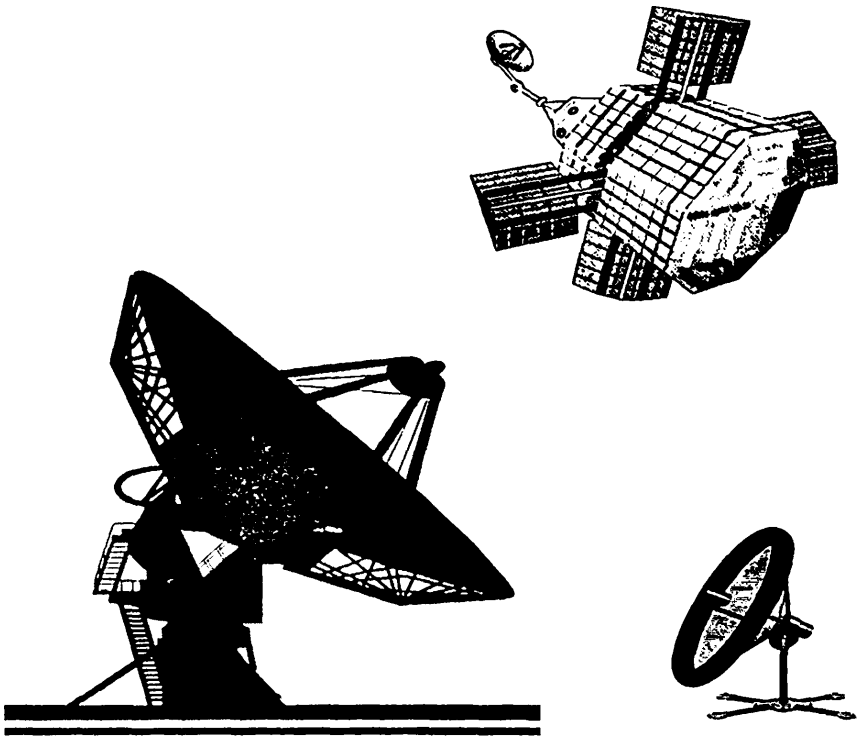
Διάγραμμα 2. Διάγραμμα σύνδεσης τερματικών με βασική πρόσβαση στο δίκτυο ISDN

Δορυφορικές επικοινωνίες

Οι δορυφορικές επικοινωνίες επειδή δεν εξαρτώνται από χερσαία ή υποβρύχια καλώδια και οπτικές ίνες, διαθέτουν μοναδικές ιδιότητες και έχουν σημαντικές προοπτικές χρησιμοποίησής τους μέσα στην προσεχή δεκαετία.

- Η πρώτη ιδέα για τις δορυφορικές επικοινωνίες προήλθε από τον συγγραφέα επιστημονικής φαντασίας, Arthur Clarke το 1945, σαν ένα παγκόσμιο σύστημα τηλεπικοινωνιών μέσω δορυφόρων.
- Μετά από 12 χρόνια, το **1957**, εκτοξεύτηκε ο πρώτος τεχνητός δορυφόρος, ο Sputnik-1, από την τότε USSR (Σοβιετική Ενωση) για να μελετηθούν τα ραδιοσήματα που εξέπεμπε. Ο Sputnik-1 ήταν μια σφαίρα από ειδικό κράμα αλουμινίου με διάμετρο 58 εκ. και βάρος 83,6 κιλών.
- Το **1965** τίθεται σε τροχιά από τις ΗΠΑ ο πρώτος εμπορικός γεωστατικός δορυφόρος που ονομάζεται Early Bird.
- Το **1969** στις 21 Ιουλίου οι πρώτοι άνθρωποι, Νιλ Αρμστρονγκ και Εντουιν Αλντριν, περπατούν στη σελήνη. Εκτοτε έχουν σταλεί επιτυχώς περισσότερες από 10 αποστολές στο διάστημα, με προορισμό διάφορους πλανήτες ή δορυφόρους.
- Το **1997** ο 'Ιχνηλάτης του Αρη προσεδαφίζεται στον Αρη και συλλέγει διάφορα στοιχεία από τον πλανήτη.

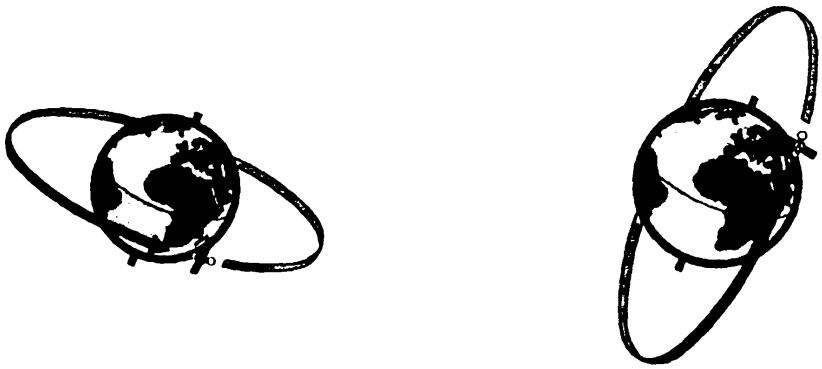
Στις δορυφορικές τηλεπικοινωνίες υπάρχουν δύο τμήματα: Το διαστημικό τμήμα και το επίγειο τμήμα.
Οι σταθμοί εδάφους ανήκουν στους τηλεπικοινωνιακούς οργανισμούς των διαφόρων χωρών, ενώ οι δορυφορικοί ανήκουν σε διεθνείς ή εθνικούς δορυφορικούς οργανισμούς (Εικόνα 15.4).



Εικόνα 15.4. Δορυφορικές επικοινωνίες.

Οι τροχιές των δορυφόρων είναι τρεις:
1. Η πολική (Από τον έναν πόλο στον άλλο)
2. Η Ισημερινή (Γύρω από τον ισημερινό)
3. Η κεκλιμένη (Πλάγια κλίση)

Το σχήμα της τροχιάς των δορυφόρων είναι κυκλικό ή ελλειπτικό (Εικόνα 15.5).



Εικόνα 15.4. Τροχιές δορυφόρων.

Οι δορυφορικές επικοινωνίες ανάλογα με τα σημεία που επικοινωνούν, αν είναι δηλαδή σταθερά ή κινητά, διακρίνονται σε: δορυφορικές επικοινωνίες σταθερής υπηρεσίας και δορυφορικές επικοινωνίες κινητής και ναυτιλιακής υπηρεσίας.

Τα σημαντικότερα δορυφορικά συστήματα που υπάρχουν σήμερα είναι:
-Το INTELSAT που ιδρύθηκε το 1964 και έχει έδρα την Ουάσιγκτον.
-Το INMARSAT που ιδρύθηκε το 1974 για ναυτιλιακές και κινητές δορυφορικές επικοινωνίες και έχει έδρα το Λονδίνο και
-Το *Eutelsat,* που είναι ο κυριότερος ευρωπαϊκός δορυφορικός οργανισμός, ιδρύθηκε το 1985 και έχει έδρα στο Παρίσι.

Πέρα από τα παραπάνω δορυφορικά συστήματα, από τις αρχές της δεκαετίας του 1980 άρχισαν να εμφανίζονται και άλλα δορυφορικά συστήματα που έδιναν έμφαση στις εθνικές επικοινωνίες και την τηλεόραση.

Η Ελλάδα είναι από τις πρώτες χώρες που χρησιμοποίησαν τις διεθνείς δορυφορικές επικοινωνίες με 3 σταθμούς εδάφους του συστήματος Intelsat, με 1 σταθμό εδάφους του συστήματος Eutelsat, και με 1 κεραία της υπηρεσίας Euteltracs εγκατεστημένη από το 1970 στην περιοχή Θερμοπυλών, στο Κέντρο Δορυφορικών Επικοινωνιών "Θερμοπύλαι". Πέρα όμως από το Κέντρο Δορυφορικών Επικοινωνιών "Θερμοπύλαι" υπάρχει και Κέντρο Δορυφορικών Επικοινωνιών "Νεμέα" που έχει ανάλογους σταθμούς εδάφους.

Το Ηλεκτρονικό Εμπόριο

Τι είναι το ηλεκτρονικό εμπόριο;

Ως ηλεκτρονικό εμπόριο θα μπορούσαμε να πούμε ότι είναι η οποιαδήποτε μορφή εμπορικής συναλλαγής κατά την οποία οι ενδιαφερόμενοι έρχονται σε επαφή με ηλεκτρονικά μέσα αντί για άμεσους τρόπους συναλλαγής ή απευθείας επαφή με τον προμηθευτή. Έτσι το Internet έχει οδηγήσει τις εταιρίες στον πειραματισμό νέων πρωτοποριακών μεθόδων συναλλαγής και τους καταναλωτές σε ένα περιβάλλον όπου επικρατεί ο «ΥΠΟΛΟΓΙΣΤΗΣ». Ωστόσο το πνεύμα του ηλεκτρονικού εμπορίου φαίνεται πολύ καλύτερα στην πράξη αν το δούμε σαν ένα συγκερασμό ανταλλαγής αγαθών και τεχνολογίας, στη λειτουργία μιας σύγχρονης επιχείρησης σε ένα χώρο που οι απαιτήσεις των καταναλωτών συνεχώς αυξάνονται.
Ο όρος ηλεκτρονικό εμπόριο είναι σε όλους μας γνωστός μια και οι περισσότεροι από εμάς κάνουμε ανάληψη χρημάτων από αυτόματα μηχανήματα και μπορούμε να ελέγχουμε το χρηματικό μας υπόλοιπο από το τηλέφωνο. Οι επιχειρήσεις και οι οργανισμοί λειτουργούν ανάλογα και πολλές από αυτές ενσωμάτωσαν το ηλεκτρονικό εμπόριο στα πλαίσια της γενικής επιχειρησιακής στρατηγικής τους με σκοπό να κατοχυρώσουν την παρουσία τους στη νέα παγκόσμια αγορά.
Η ανάπτυξη του Internet και του w.w.w. έχει οδηγήσει στην καθιέρωση της παρουσίας κάποιων καταναλωτών και εταιρειών σε μια διεθνή "on line" αγορά. Αυτές οι εξελίξεις του Internet το καθιερώνουν ως μια νέα αγορά. Έτσι το ηλεκτρονικό εμπόριο θα αναπτύσσεται με ραγδαίους ρυθμούς στο διαδίκτυο. Με τον τρόπο αυτό μέσα στην πρώτη δεκαετία του 2000 εκατομμύρια ανθρώπων και εταιρειών θα διαφημίζουν, θα πουλάνε, θα αγοράζουν, και θα συνεργάζονται σε καθημερινή βάση καθώς το Internet συγχωνεύεται με άλλους κλάδους της τεχνολογικής πορείας με συνέπεια να μειώνεται αισθητά το κόστος της λειτουργίας μιας προμηθευόμενης βάσης. Το Internet όμως της σημερινής εποχής έχει ακόμη κάποιους περιορισμούς στις δυνατότητές του όπως για παράδειγμα τους κανόνες ασφαλείας του και τις προϋποθέσεις ελέγχου πρόσβασης χρηστών στο διαδίκτυο. Τα προβλήματα αυτά με την ανάπτυξη της τεχνολογίας μπορούν σύντομα να εξαλειφθούν.
Με το ηλεκτρονικό εμπόριο ωφελούνται οι εταιρίες και έτσι μπορούν να είναι περισσότερο συνεπείς και ελαστικές στις εσωτερικές τους λειτουργίες, να δουλεύουν πιο στενά με τους προμηθευτές τους και να είναι πιο κοντά στις ανάγκες και τις απαιτήσεις των πελατών τους. Με τον τρόπο αυτό δίνεται η δυνατότητα στις εταιρίες να επιλέγουν τους καλύτερους προμηθευτές ανεξάρτητα από την γεωγραφική τους θέση και να πουλάνε σε μία παγκόσμια αγορά. Ακόμη, ο προμηθευτής έχει τη δυνατότητα να

προσφέρει πέρα από τα αγαθά και υπηρεσίες ή πληροφορίες στον πελάτη, με ή χωρίς αντάλλαγμα την πληρωμή. Μία ειδική περίπτωση ηλεκτρονικής ανταλλαγής είναι το ηλεκτρονικό retailing όπου ο πελάτης είναι ένας συνηθισμένος καταναλωτής και όχι μια άλλη εταιρεία. Οι εταιρείες που επιλέγουν να αλλάξουν τις δομές και την διαδικασία παραγωγής ώστε να εκμεταλλευτούν όσο το δυνατόν περισσότερο τις δυνατότητες που τους δίνει το ηλεκτρονικό εμπόριο μπορούν να κερδίσουν πολλά. Οι επιχειρήσεις οι οποίες επιθυμούν να χρησιμοποιήσουν το ηλεκτρονικό εμπόριο θα πρέπει συνεχώς να λαμβάνουν υπ' όψιν τους τόσο τους κινδύνους όσο και τις ευκαιρίες που δημιουργούνται δια μέσου του ηλεκτρονικού εμπορίου όπως για παράδειγμα την εξέλιξη της τεχνολογίας από τη μια μεριά και την έλλειψη διεθνών κανονισμών κατά τις συναλλαγές του ηλεκτρονικού εμπορίου από την άλλη.

Οι διάφοροι επαγγελματικοί οργανισμοί μπορούν να χρησιμοποιήσουν τις ηλεκτρονικές υπηρεσίες (ηλεκτρονικό εμπόριο, e-mail) για διαφορετικούς σκοπούς. Οι επιχειρήσεις με τον τρόπο αυτό και μέσα από τη χρήση του διαδικτύου μπορούν να εφαρμόσουν άλλου είδους πωλήσεις και κανάλια προώθησης των προϊόντων τους. Ακόμη μπορούν να αντικαταστήσουν τους καταλόγους των προϊόντων τους, τις λίστες των τιμών και άλλα φυλλάδια και έτσι οι επιχειρήσεις να πουλάνε περισσότερο και περισσότερα προϊόντα απευθείας στους πελάτες τους. Αρκετές εταιρίες υψηλής τεχνολογίας δίνουν την δυνατότητα τεχνικής υποστήριξης σε ανοιχτή σύνδεση (on line) επιτρέποντας στους πελάτες τους να βρουν εύκολα απαντήσεις στα προβλήματα τους όλη την ημέρα όλη την εβδομάδα, διαφυλάσσοντας έτσι τα χρήματα της εταιρίας από έξοδα στήριξης προσωπικού και ταχυδρομείου. Η τεχνολογία του ηλεκτρονικού εμπορίου μπορεί ακόμη να χρησιμοποιηθεί σε οποιοδήποτε περιβάλλον μέσα στο οποίο ανταλλάσσονται έγγραφα ανάμεσα σε οργανισμούς και προμηθευτές, οικονομικές υπηρεσίες και υπηρεσίες ανταλλαγής και μεταφοράς, υγείας, δικαιοσύνης, εφορίας. Το ηλεκτρονικό εμπόριο μειώνει τον χρόνο διακίνησης εγγράφων, μειώνει την γραφειοκρατία και την διοικητική διαχείριση στις υπηρεσίες και τους οργανισμούς και επιτρέπει στα πρακτορεία διακίνησης προϊόντων να υιοθετήσουν πιο κατάλληλες πρακτικές όπως just in time, direct store delivery. ο ηλεκτρονικό εμπόριο είναι πολύ καλά εδραιωμένο και πετυχημένο σε ανεπτυγμένες χώρες.Τον δρόμο στο ηλεκτρονικό εμπόριο οδηγούν οι ΗΠΑ, η Ιαπωνία και η Ευρώπη Ο αντίκτυπος του ηλεκτρονικού εμπορίου θα είναι σύντομα ορατός και στις αναπτυσσόμενες εταιρίες και στο σύνολο της κοινωνίας μας γενικότερα. Για τις εταιρίες είνες ου εκμεταλλεύονται ολοκληωτικά τις δυνατότητές του, το ηλεκτρονικό εμπόριο είναι σε θέση να προσφέρε τη δυνατότητα μεγάλων αλλαγών που θα αλλάξουν τις προσδοκίες των καταναλωτών, θα ορίσουν εκ νέου την αγορά ή ακόμη θα δημιουργήσουν ολοκληρωτικά νέες αγορές και οι περιορισμοί της γεωγραφικής θέσεως και του χρόνου γενικότερα θα εξουδετεωθούν.

Κατηγορίες ηλεκτρονικού Εμπορίου

Οι κατηγορίες του ηλεκτρονικού εμπορίου είναι οι εξής :

1) ΕΤΑΙΡΙΑ - ΚΑΤΑΝΑΛΩΤΗΣ
2) ΕΤΑΙΡΙΑ - ΕΤΑΙΡΙΑ
3) ΕΤΑΙΡΙΑ - ΔΙΟΙΚΗΣΗ
4) ΚΑΤΑΝΑΛΩΤΗΣ –ΔΙΟΙΚΗΣΗ

Η **κατηγορία εταιρία-καταναλωτή** αναφέρεται περισσότερο στην ηλεκρονική λιανική πώληση. Αυτή η κατηγορία έχει επεκταθεί αρκετά με την καθιέρωση του w.w.w.. Δια μέσου του Internet προσφέρονται στον αγοραστή διάφορα υλικά αγαθά π.χ. υπολογιστές, βιβλία, αυτοκίνητα.
Ένα παράδειγμα της **δεύτερη κατηγορίας** (εταιρία- εταιρία) είναι η χρήση του διαδικτύου από μια εταιρία για να εκτελεί τις παραγγελίες από τους προμηθευτές της, να παίρνει τιμολόγια και να κάνει πληρωμές. Αυτή η κατηγορία του ηλεκτρονικού εμπορίου χρησιμοποιώντας ειδικότερα το Ηλεκτρονικό Σύστημα Ανταλλαγής (E.D.I.) σε ιδιωτικά ή value –added δίκτυα, έχει εδραιωθεί αρκετά καλά στην παγκόσμια αγορά εδώ και πολλά χρόνια.
Η **τρίτη κατηγορία** (εταιρία-διοίκηση) καλύπτει όλες τις συνδιαλλαγές μεταξύ εταιριών και κυβερνητικών οργανισμών. Για παράδειγμα, οι λεπτομέρειες για τους προτιθέμενους διαγωνισμούς γνωστοποιούνται στο Internet και οι εταιρίες μπορούν να ανταποκριθούν ηλεκτρονικά. Επιπέον εκτός από τις προκηρύξεις των διαγωνισμών, οι δημόσιες υπηρεσίες και ειδικά οι εφορίες μπορούν να προσφέρουν την εναλακτική της ηλεκτρονικής συνδιαλλαγής σε ότι αφορά τις διευκούνσεις στην επιστροφή χρημάτων, στην πληρωμή κοινών-δημοσίων φόρων και αλλού.
Η **τέταρτη κατηγορία** (καταναλωτή-διοίκηση) δεν έχει ακόμα ανατυχθεί. Στόχος της κατηγορίας αυτής είναι να καλύψει τομείς κοινωνικής ευημερίας, με κύριο έργο πληρωμές πολιτών, επιστροφή φόρων και άλλα.

Το ηλεκτρονικό εμπόριο σήμερα

Το ηλεκτρονικό εμπόριο μπορεί ακόμη να κάνει μεγάλη αυτοματοποίηση πληροφοριών (πληροφοριακών συστημάτων) ειδικά στην προώθηση προηγμένων υστημάτων και στην ηλεκτρονική κατανομή συνδιαλλαγών, όπως στα συστήματα αεροπορικών κρατήσεων, στα εμπορικά αποθέατα, στις ασφάλειες, κλπ. Ειδικότερα:

Στα συστήματα αεροπορικών κρατήσεων: οι υπηρεσίες που αφορούν τη διακίνηση των αεροπορικών εισιτηρίων και είναι διασκορπισμένες μέσα σε μία χώρα ή ακόμα και σε όλο τον κόσμο θα πρέπει να χρησιμοποιούν μια συγκεκριμένη βάση δεδομένων και να έχουν συμφωνήσει να χρησιμοποιούν όλες οι εταιρείες την ίδια βάση.

Στα εμπορικά αποθέματα και στους προμηθευτές εργοστασιακών προϊόντων: επιτρέπεται η γρήγορη επαφή μεταξύ μακρινών κατατημάτων κάποιας εταιρίας διασκορπισμένων σε όλο τον κόσμο για άμεση ενημέρωση του υπολοίπου του προϊόντος που υπάρχει στα διάφορα καταστήματά τους και άμεση πληροφόρηση των προμηθευτών.

Στις ασφάλειες: γίνεται ευκολότερη συλλογή δεδομένων.

Ωστόσο, το ηλεκτρονικό εμπόριο πέρα από την αυτοματοποίηση των πληροφοριών μπορεί να επεκταθεί και σε άλλους τομείς της καθημερινής μας ζωής.

Τέλος, στην ερώτηση: «Τι μπορούμε να κάνουμε με τη χρήση του ηλεκτρονικού εμπορίου σήμερα;», η απάντηση είναι κυρίως μία από τις ακόλουθες:

I. Η αγορά προϊόντων θα μπορεί να γίνεται από το σπίτι με άμεσο επακόλουθο οι τιμές τους να εμφανίζονται χαμηλότερες εξαιτίας της μείωσης του ενοικίου των καταστημάτων.

II. Η παγκόσμια αγορά θα μπορεί να γίνει προσιτή σε μικρότερους πωλητές και όχι μόνο σε μεγάλες εταιρίες.

III. Κάποια άτομα ή εταιρίες θα μπορούν να δημιουργήσουν έναν συνεταιρισμό, ο οποίος θα στηρίζεται αποκλειστικά και μόνο στην ηλεκτρονική διεκπεραίωση.

IV. Η απευθείας παράδοση προϊόντων θα γίνεται συντομότερα προς όφελος όλων μας.

ΜΕΡΟΣ IV

ΕΙΔΙΚΑ ΘΕΜΑΤΑ

16 Ο ΥΠΟΛΟΓΙΣΤΗΣ ΣΤΟ ΝΗΠΙΑΓΩΓΕΙΟ

Το πρώτο Ελληνικό Νηπιαγωγείο ιδρύθηκε στην Αθήνα στα τέλη του 19ου αιώνα από την Αικ. Λασκαρίδου και η μέθοδος διδασκαλίας που εφαρμόστηκε βασίστηκε στις θεωρίες του Φραίμπελ. Αργότερα, το 1929, αναγνωρίστηκε ο θεσμός του νηπιαγωγείου από το επίσημο Ελληνικό κράτος προτάσσοντας ως βασικό σκοπό την καλλιέργεια του προφορικού λόγου και τη διευκόλυνση της κοινωνικής ένταξης των παιδιών. Σήμερα, στα νηπιαγωγεία της χώρας μας αν και εφαρμόζονται σύγχρονες παιδαγωγικές μέθοδοι διδασκαλίας, η εισαγωγή νέων τεχνολογικών μέσων δεν αποτελεί τον κανόνα, αλλά τη μεμονωμένη περίπτωση.

Ωστόσο, θεωρείται δεδομένο ότι η εκπαίδευση του παιδιού και η επαφή του με την τεχνολογία αρχίζει από πολύ νεαρή ηλικία και συγκεκριμένα από το νηπιαγωγείο. Η τεχνολογία είναι διασκεδαστική και συγχρόνως εκπαιδευτική για τα παιδιά του Νηπιαγωγείου. Τα παιδιά θεωρούν την τεχνολογία σαν ένα μέρος του φυσικού τους περιβάλλοντος. Ένα παιδί δηλαδή 4-6 ετών είναι δυνατό να γίνει χρήστης Η/Υ, ενώ αντίθετα αρκετοί ενήλικες αντιμετωπίζουν με δυσπιστία και φόβο τη σκέψη και μόνο να χειριστούν Η/Υ. Αυτό συμβαίνει γιατί τα παιδιά, σε αντίθεση με τους μεγάλους, δεν έχουν προκατάληψη με τον υπολογιστή. Τα σημερινά παιδιά απλά έτυχε να γεννηθούν αυτή την εποχή που αποκαλείται "Εποχή της Ηλεκτρονικής Επανάστασης". Έτσι, αντικρίζουν τον υπολογιστή σαν ένα από τα εποπτικά μέσα που έχουν στη διάθεσή τους για παιγνίδι και μάθηση. Επιπλέον ο Η/Υ δεν έχει σκοπό να αντικαταστήσει κανένα παραδοσιακό εποπτικό μέσο την τάξη, αλλά αντίθετα, σαν ένα νέο εποπτικό μέσο, να συμβάλλει και αυτός με το δικό του τρόπο στην όλη διαδικασία της διδασκαλίας και μάθησης. Οι Υπολογιστές λοιπόν, η τηλεόραση και το video μπορούν να χρησιμοποιηθούν για τη διδασκαλία πολλών εννοιών, σχημάτων, χρωμάτων, καθώς και για τη διδασκαλία της

γλώσσας και των αριθμών στο Νηπιαγωγείο. Γενικότερα η διδασκαλία των μαθημάτων σε παιδιά Νηπιαγωγείου, πέρα από το έντυπο υλικό μπορεί ακόμη να γίνεται είτε με τα οπτικοακουστικά μέσα είτε με τους Η/Υ. Για να προετοιμάσουμε τα παιδιά στη χρήση των Η/Υ είναι απαραίτητο να διαρρυθμιστεί κατάλληλα ένας χώρος μέσα στην τάξη που θα ονομαστεί "Η γωνιά του Υπολογιστή". Ο υπολογιστής πρέπει να τοποθετηθεί μέσα στην τάξη και όχι σε ξεχωριστή αίθουσα, γιατί για τα παιδιά ο υπολογιστής θα είναι σημαντικός μόνο αν αποτελεί μέρος της συνολικής εκπαιδευτικής δραστηριότητας, που αναπτύσσεται μέσα στο χώρο αυτό. Ακόμη, ο χώρος που θα τοποθετηθεί ο υπολογιστής πρέπει να επιτρέπει σε όλα τα παιδιά να παρακολουθούν τι συμβαίνει, γιατί η χρήση του υπολογιστή είναι συλλογική εργασία. Επειδή τα παιδιά θα πηγαίνουν κατά ομάδες των τριών ή τεσσάρων ατόμων και θα δουλεύουν μαζί, θα βοηθά το ένα το άλλο, θα πρέπει να έχουν όλα οπτική επαφή με την οθόνη και πρόσβαση στο πληκτρολόγιο. Ένας ιδιαίτερος λόγος απαιτεί να τοποθετηθεί ο υπολογιστής μπροστά από ένα τοίχο, γιατί υπάρχει κίνδυνος να παρασυρθεί από τα παιδιά.

Η χρήση ηλεκτρονικών εκπαιδευτικών προγραμμάτων από παιδιά προσχολικής ηλικίας πρέπει να ενταχθεί μέσα στη διδακτέα ύλη που καθορίζεται στις αρχές κάθε νέας σχολικής χρονιάς. Με άλλα λόγια η εξοικείωση με τους Η/Υ πρέπει να γίνεται προγραμματισμένα, όπως προγραμματισμένες πρέπει να είναι όλες οι δραστηριότητες στο νηπιαγωγείο. Τα εκπαιδευτικά προγράμματα που θα χρησιμοποιηθούν στο νηπιαγωγείο, πρέπει να έχουν ορισμένα χαρακτηριστικά, που να ανταποκρίνονται στις δυνατότητες και τις επιθυμίες των παιδιών. Τα χαρακτηριστικά αυτά κυρίως είναι τα ακόλουθα:

- Απλότητα στο χειρισμό
- Μαθήματα διάφορων επιπέδων δυσκολίας
- Δημιουργία ενός φιλικού περιβάλλοντος, έτσι που τα παιδιά να ανταποκρίνονται στα μηνύματα του Η/Υ με το πάτημα ενός κουμπιού.
- Εκπαιδευτικές ενότητες όπως χρώματα, σχήματα, μέγεθος, βάρος αντικειμένων, συσχετισμός/συνειρμός αντικειμένων, ζώα, αρίθμηση, ανθρώπινο σώμα.
- Πλούτος γραφικών παραστάσεων, χρωμάτων και ήχου.

Γενικότερα όλη η δομή της διδασκαλίας και η εξοικείωση με τον Η/Υ θα πρέπει να έχει τη μορφή παιγνιδιού. Η μάθηση μέσα από το παιγνίδι θεωρούνταν πάντοτε πολύ αποτελεσματική, τόσο από τους νεότερους όσο και από τους παλαιότερους παιδαγωγούς. Γιατί, όπως αναφέρει στην 'Πολιτεία' ο Πλάτωνας, πριν από δύο χιλιάδες χρόνια, 'καμιά υποχρεωτική μάθηση δε μένει στην ψυχή του ανθρώπου'. Όταν διδάσκεις παιδιά, κάνε το με τη μορφή του παιχνιδιού '.

Βασικός παράγοντας στην επιτυχία της εκπαίδευσης των νηπίων με Η/Υ είναι η σωστή κατάρτιση των νηπιαγωγών. Προκειμένου να μυήσουν τα παιδιά στην ορθή χρήση του υπολογιστή είναι απαραίτητο οι νηπιαγωγοί ν' αποκτήσουν μια ορισμένη κατάρτιση. Κι αυτό γιατί η επιτυχία οποιουδήποτε προγράμματος πληροφορικής βασίζεται στη συνεργασία και την ενεργό συμμετοχή εκείνου που το διδάσκει. Είναι γνωστό άλλωστε το πόσες ερωτήσεις κάνουν τα παιδιά σ' αυτή την ηλικία. Κι ο εκπαιδευτικός πρέπει να είναι έτοιμος να δώσει ικανοποιητικές απαντήσεις τουλάχιστο στις περισσότερες ερωτήσεις. Μελέτες που διεξήγαγε το 1986 ο Robert Hess, καθηγητής των Παιδαγωγικών στο Πανεπιστήμιο του Stanford των ΗΠΑ, για τις επιδράσεις που έχει ο υπολογιστής στα παιδιά προσχολικής ηλικίας, έδειξαν ότι παιδιά που χρησιμοποιούσαν τον υπολογιστή τουλάχιστο μια φόρα τη βδομάδα βελτίωσαν την ικανότητα ανάγνωσης σε σύγκριση με παιδιά που δε χρησιμοποιούσαν καθόλου τον υπολογιστή. Τα αποτελέσματα ήταν ακόμα πιο εντυπωσιακά στις περιπτώσεις παιδιών τα οποία χρησιμοποιούσαν υπολογιστή και στο σπίτι. Η Sherry Turkley, στο βιβλίο της Computers and the Second Self, αναφέρει ότι για παιδιά που είναι κάπως κοινωνικά απομονωμένα, η επαφή με τους Η/Υ παρέχει μια ασφαλή και επιτυχή εμπειρία, η οποία μπορεί να καταστεί το πρώτο βήμα για μεγαλύτερη επικοινωνία και συνεργασία με τους δασκάλους τους. Πρόσφατες έρευνες στην Αμερική έχουν δείξει ότι ο υπολογιστής αυξάνει την ποιότητα της κοινωνικής συναναστροφής, ενθαρρύνει τη μάθηση κατά ομάδες και αυξάνει την αυτοπεποίθηση και την ικανότητα για λύση προβλημάτων. Αυτά τα συμπεράσματα υπογραμμίζουν το γεγονός ότι η εξοικείωση με τους Η/Υ από την προσχολική ηλικία υποβοηθάει σε μεγάλο βαθμό την ψυχοκοινωνική ανάπτυξη του παιδιού.

Βέβαια στις μέρες μας, στο νηπιαγωγείο, μπορούμε να χρησιμοποιήσουμε είτε έντυπο υλικό, είτε να κάνουμε χρήση οπτικοακουστικών μέσων, είτε να έχουμε Η/Υ και μέσα εκπαιδευτικής τεχνολογίας, είτε ένα συνδυασμό από τα παραπάνω.

Όλες οι παραπάνω μέθοδοι διδασκαλίας έχουν πλεονεκτήματα και μειονεκτήματα. Αναλυτικότερα,

1. Με τη χρήση του έντυπου υλικού ελέγχεται η ροή της παρουσίασης από το μαθητή και ανατρέχει κανείς ευκολότερα σε παλιότερες σημειώσεις και βιβλία. Βασικό όμως μειονέκτημα αυτού του τρόπου διδασκαλίας είναι ότι τα γραφικά είναι στατικά και τα παιδιά δεν εντυπωσιάζονται από την παρουσίαση.

2. Τα οπτικοακουστικά μέσα πλεονεκτούν στα γραφικά τους που είναι ρεαλιστικά και ενδιαφέροντα, μια και μπορούν να τα συνδυάσουν με κίνηση και ήχο δίνοντας έτσι ένα ξεχωριστό τύπο παρουσίασης. Το βασικό τους μειονέκτημα όμως είναι ότι δεν γίνεται εύκολα επανάληψη της πληροφορίας και δεν ελέγχεται ο ρυθμός της παρουσίασης στα παιδιά.

3. Τέλος οι Η/Υ για τη διδασκαλία των διαφόρων μαθημάτων, υπερέχουν των παραπάνω μεθόδων, έχοντας ως μόνο σοβαρό μειονέκτημα το υψηλό κόστος αγοράς τους. Τα κυριότερα πλεονεκτήματά τους είναι ότι:
α) ελέγχεται η ροή της παρουσίασης του μαθήματος
β) μπορούν να επαναλάβουν προηγούμενες πληροφορίες
γ) έχουν ενδιαφέροντα γραφικά που συνδυάζονται με ήχο και κίνηση
δ) απαιτούν την ενεργό συμμετοχή του μαθητή
και ε) μπορούν να καταγράφουν τις επιδόσεις των μαθητών στα διάφορα μαθήματα ή test.

Οι υπολογιστές όμως βοηθούν τα παιδιά της προσχολικής ηλικίας να αναπτύξουν δραστηριότητες και να βελτιώσουν τις ικανότητές τους στη λύση των διαφόρων προβλημάτων. Έτσι ο υπολογιστής γίνεται:

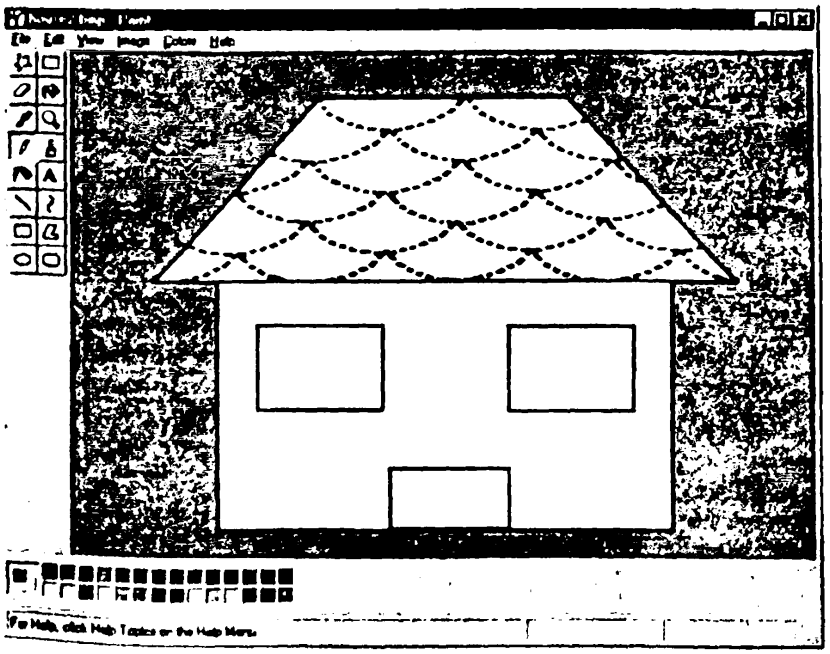
1. *Προσωπικός εκπαιδευτής*. Ο μαθητής καθοδηγείται από τον υπολογιστή για την εκτέλεση συγκεκριμένου προγράμματος, το οποίο έχει προγραμματιστεί από το δάσκαλο. Πολλά από τα προγράμματα αυτά χρησιμοποιούν σαν βάση τον διάλογο ανάμεσα στον μαθητή και τον εκπαιδευτικό ή σε ομάδα μαθητών.

2. *Εργαλείο μάθησης*. Με τη χρήση αυτή του υπολογιστή οι μαθητές μπορούν να μάθουν να ζωγραφίζουν (όπως φαίνεται στις παρακάτω εικόνες 16.1, 16.2) ή να χρησιμοποιούν άλλα προγράμματα για την απόκτηση γνώσεων. Ορισμένα παιγνίδια δίνουν στους παίκτες ευκαιρίες να δημιουργήσουν τις δικές τους εικόνες και ιστορίες ή τη δική τους μουσική. Τα παιγνίδια αυτά μπορούν να χρησιμοποιηθούν από ένα παιδί ή μια ομάδα παιδιών που συνεργάζονται μεταξύ τους. Με τον τρόπο αυτό οι υπολογιστές αναπτύσσουν βασικές ικανότητες που οδηγούν είτε στη γνώση είτε στην ανάπτυξη συναισθηματικών και κοινωνικών διαθέσεων. Μερικά σχετικά προγράμματα είναι τα εξής:

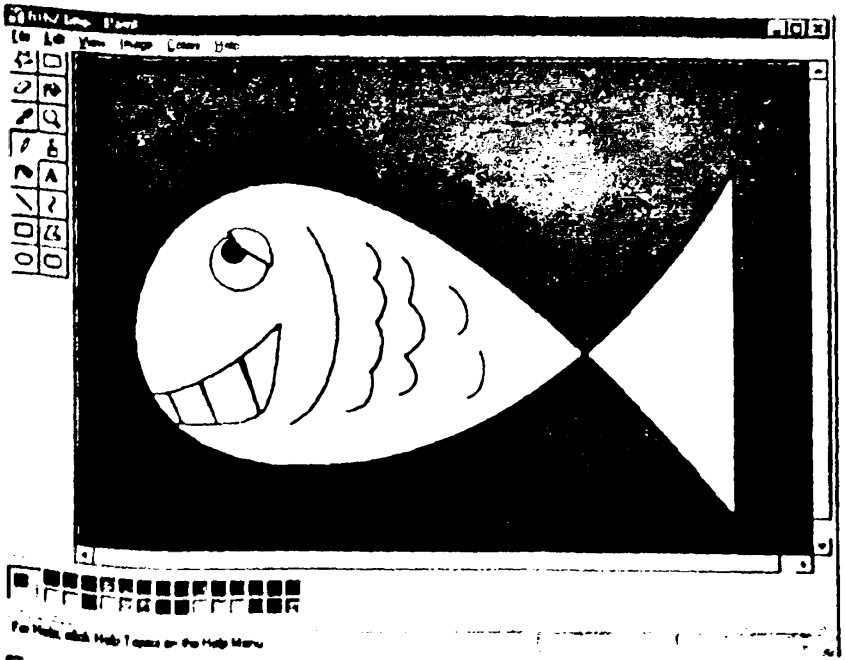
- Mix and Match (Τα παιδιά μπορούν να δημιουργούν δικές τους εικόνες παίρνοντας έτοιμα στοιχεία)
- Face-it (Τα παιδιά φτιάχνουν ένα πρόσωπο)
- Micro-movie (Τα παιδιά σχεδιάζουν κινούμενα σχέδια)
- Making Decisions (Τα παιδιά πρέπει να αποφασίσουν τι θα πάρουν σε μια καταναλωτική κοινωνία)
- Air Flight Simulator (Το πρόγραμμα αυτό αναφέρεται σε μεγάλα παιδιά και δείχνει τον τρόπο με τον οποίο πετάει ένα αεροπλάνο)
- Barbie (Τα περισσότερα προγράμματα από αυτά έχουν παιγνίδια που επιτρέπουν στο παιδί να δημιουργήσει δικά του σχέδια)

- Candy Land (Το πρόγραμμα δείχνει εικόνες με γράμματα, με σχέδια και σχήματα, με μαθηματικές έννοιες, έχει ομαδικά παιγνίδια, κ.ά.)
- Broadway (Τα παιδιά συμμετέχουν στις χαρές και στις λύπες στο ανέβασμα ενός θεατρικού έργου) κ.ά.

3. *Μαθητής.* Τη θέση εδώ του μαθητή την παίρνει ο υπολογιστής. Με τον τρόπο αυτό ο μαθητής εκπαιδεύει (προγραμματίζει) τον υπολογιστή για να μπορεί να λύνει διάφορα προβλήματα ή ακόμη να εκτελεί διάφορα παιγνίδια.



Εικόνα 16.1



Εικόνα 16.2.

Σελίδες στο INTERNET για παιδιά προσχολικής ηλικίας

Μέσα στο διαδίκτυο υπάρχουν πολλές σελίδες που αφορούν παιδιά προσχολικής ηλικίας. Οι σελίδες αυτές επιτρέπουν στα παιδιά μεταξύ άλλων να ζωγραφίσουν, να δουν εικόνες από παραμύθια, να τυπώσουν κάποιες εικόνες από τα παραμύθια και μετά να τις ζωγραφίσουν κ.ά. Επιπλέον υπάρχουν σελίδες που αναφέρονται στους νηπιαγωγούς και στους γονείς.

Ειδικότερα, οι σελίδες που αφορούν τους νηπιαγωγούς περιλαμβάνουν κυρίως προγράμματα για τη διδασκαλία μαθημάτων και δραστηριοτήτων παιδιών της προσχολικής ηλικίας, ενώ οι σελίδες που απευθύνονται στους γονείς, πέρα από τον συμβουλευτικό χαρακτήρα που έχουν, απαντούν και σε διάφορες ερωτήσεις-θέματα σχετικά με τα παιδιά προσχολικής ηλικίας.

Παρακάτω παρέχονται ορισμένες διευθύνσεις σχολείων, μεθόδων διδασκαλίας, προγραμμάτων προσχολικής αγωγής που υπάρχουν στο διαδίκτυο για τη διευκόλυνση του αναγνώστη:

http://www.bellmontessori.org/

http://www.teachingstrategies.com/

http://www.ncrel.org/skrs/areas/ea0cont.htm

http://www.netzone.com/~pvand/educate.htm

http://www.edpro.com

Επιπλέον είναι εύκολο να βρούμε διευθύνσεις στο διαδίκτυο για παιδιά προσχολικής ηλικίας αν στις διάφορες μηχανές (π.χ. altavista, yahoo, κλπ) γράψουμε στην εντολή ανεύρεσης (SEARCH) τη λέξη προσχολική ηλικία (early childhood education).

17 ΥΠΟΛΟΓΙΣΤΗΣ ΚΑΙ ΕΚΠΑΙΔΕΥΣΗ

• *Άτομα με Ειδικές Ανάγκες & Η/Υ*

• *Τηλεμάθηση*

Εισαγωγή

Η τεχνολογία είναι ένας από τους σημαντικότερους παράγοντες που διαμορφώνουν τη ζωή του ανθρώπου.

Σήμερα η τεχνολογία παρουσιάζει αλματώδη ανάπτυξη σε όλους τους τομείς. Ένας από αυτούς είναι και η ηλεκτρονική που περιλαμβάνει τα ηλεκτρονικά και τους computers. Πολλοί υποστηρίζουν ότι βρισκόμαστε σήμερα στα πρόθυρα της Ηλεκτρονικής ή Πληροφορικής Επανάστασης στην οποία τον κύριο ρόλο παίζουν οι Ηλεκτρονικοί Υπολογιστές (Η/Υ).

Οι Η/Υ που μοιάζουν με μια μηχανή που μπορεί να αποθηκεύσει τεράστιες ποσότητες πληροφοριών και να τις επεξεργασθεί με μεγάλη ταχύτητα και καταπληκτική ακρίβεια καθιστούν αναγκαία την παρουσία τους και στο χώρο της εκπαίδευσης. Γενικά, οι σημαντικότεροι τρόποι χρήσης των Η/Υ στην εκπαίδευση είναι δύο:

α.) Διδασκαλία Βασισμένη σε Υπολογιστές(Computer Based Instruction)
β.) Διδασκαλία με τη βοήθεια Υπολογιστών(Computer Aided Instruction ή Computer Assisted Instruction).

Στην πρώτη περίπτωση έχουμε να κάνουμε με ένα μάθημα, το οποίο αναφέρεται στη λειτουργία και στην τεχνολογική γνώση των Η/Υ. Βέβαια, όπως εύλογα μπορεί κανείς να αντιληφθεί, ένα τέτοιο μάθημα δεν μπορεί να γίνει αντικείμενο διδασκαλίας σε όλες τις βαθμίδες της εκπαίδευσης. Αντίθετα, μάλιστα, επειδή απαιτούνται αυξημένες νοητικές ικανότητες κι ένα υψηλό επίπεδο κατανόησης και η διδασκαλία του είναι εφικτή μόνο στην ανώτερη βαθμίδα της εκπαίδευσης, δηλαδή είτε σε πανεπιστημιακά ιδρύματα, είτε σε τεχνολογικά εκπαιδευτικά ιδρύματα, είτε ακόμη και σε ιδιωτικές σχολές πληροφορικής.

Στη δεύτερη περίπτωση, ο Η/Υ χρησιμοποιείται σα βοήθημα, σαν εργαλείο, που με τον τρόπο του διευκολύνει ή βελτιώνει τη διδασκαλία του μαθήματος. Με τη βοήθεια του Η/Υ μπορούν να γίνουν παρουσιάσεις μαθημάτων, εικόνων, να γίνουν πειράματα, να φτιάξει κανείς γραφικά ή να επεξεργαστεί κείμενα ή εικόνες. Με λίγα λόγια, ο Η/Υ στην περίπτωση αυτή μπορεί να χρησιμοποιηθεί σε όλες τις βαθμίδες της εκπαίδευσης, από το νηπιαγωγείο ως το πανεπιστήμιο.

Σε τι ωφελεί η παρουσία του Η/Υ στο σχολείο;

α) Ελαττώνει το χρόνο απασχόλησης του διδακτικού προσωπικού σε διοικητικά καθήκοντα,
β) Παρέχει δυνατότητες επεξεργασίας στοιχείων των μαθητών,
γ) Ενισχύει τους αδύνατους μαθητές και προωθεί τους καλούς με ειδικά προγράμματα,
δ) Κάνει πιο αποδοτική τη διδασκαλία,
ε) Μπορεί να γίνει η εκπαίδευση από απόσταση, κ.λ.π.

Βέβαια υπάρχει και ο αντίλογος που υποστηρίζει ότι η χρήση των Η/Υ είναι μηχανιστική, αντιανθρώπινη, οδηγεί στην κατάργηση της σκέψης και των συναισθημάτων, προωθεί την ατομική δράση και δεν βοηθά στην ανάπτυξη της ομαδικής δράσης (συνεργασία).

Σήμερα όλοι αναγνωρίζουν ότι η νέα τεχνολογία αποτελεί επανάσταση στην ανθρώπινη επινόηση και σαν τέτοια δεν μπορεί να αγνοηθεί. Θα πρέπει όμως η χρησιμοποίησή της να γίνεται με την αναγκαία περίσκεψη. Επιβάλλεται να μελετηθούν τρόποι πλήρους αξιοποίησης της υπολογιστικής μηχανής ως οργάνου για την εκμάθηση δεξιοτήτων υψηλότερου επιπέδου. Μια τέτοια προσπάθεια έχει γίνει στην ειδική εκπαίδευση, όπου εκεί πράγματι μπορούμε να μιλάμε για ανώτερη μορφή ή και πλήρη αξιοποίηση της υπολογιστικής μηχανής και στην τηλεμάθηση (εκπαίδευση από απόσταση). Είναι φυσικό ότι η εκπαίδευση από απόσταση δεν καταργεί την ιδρυματική εκπαίδευση (το σχολείο), αλλά τη συμπληρώνει και προσφέρει δυνατότητες που δεν έχει το παραδοσιακό σχολείο. Επιπλέον η εκπαίδευση από απόσταση μπορεί να λειτουργήσει και πέρα από τις συγκεκριμένες ηλικίες στις οποίες απευθύνεται το σχολείο. Το σημαντικότερο στοιχείο της εκπαίδευσης από απόσταση είναι η μεγάλη ευελιξία της.

Ειδική αγωγή και μικροϋπολογιστές

Η Ρόζα Ιμβριώτη, ορίζει την ειδική αγωγή ως την επιστήμη που φροντίζει για τη μόρφωση, διδασκαλία και πρόνοια όλων των παιδιών που η σωματική και ψυχική του εξέλιξη εμποδίζεται αδιάκοπα από παράγοντες ατομικούς και κοινωνικούς, όπως:

- αδυναμίες στις αισθήσεις (τυφλά, κουφά παιδιά ή παιδιά με βλάβες στην όραση ή στην ακοή,
- σταμάτημα στην εξέλιξη του κεντρικού νευρικού συστήματος (πολύ-μέτρια-λίγο καθυστερημένα παιδιά),
- σωματικές αρρώστιες, αναπηρία, ελαττώματα που οφείλονται στο περιβάλλον (δύσκολα, απειθάρχητα παιδιά).

Τα προβλήματα των προσώπων με ειδικές ανάγκες συνοψίζονται στην έλλειψη επαφής με το περιβάλλον και στη δυσκολία λήψης ή δυσκολία σωστής λήψης πληροφορίας από το περιβάλλον αυτό. Ο εφοδιασμός λοιπόν των ατόμων αυτών με Η/Υ και η εκπαίδευσή τους στη χρήση των συσκευών αυτών, συμβάλλουν σημαντικά στην αυτοεξυπηρέτησή τους καθώς και στην επανένταξή τους στην κοινωνία και στην παραγωγική διαδικασία. Οι Η/Υ στην ειδική αγωγή, εφοδιασμένοι με τα κατάλληλα προγράμματα, δε βοηθούν μόνο στη διαδικασία ξεπεράσματος των μαθησιακών δυσκολιών που δημιουργεί η ύπαρξη της συγκεκριμένης αναπηρίας ή μειονεκτήματος, αλλά επιπλέον μπορούν να αποκαλύψουν πτυχές και δυνατότητες που παρέμειναν αναξιοποίητες λόγω της κύριας αναπηρίας ή μειονεξίας του παιδιού.

Εφαρμογές της νέας τεχνολογίας στην Ειδική Αγωγή.

Τα τελευταία χρόνια έχουν γίνει πολλές έρευνες για τη χρήση των υπολογιστών στην εκπαίδευση παιδιών με ειδικές ανάγκες. Το θέμα αυτό είναι μεγάλο και πολύπλοκο και αφορά, από το ένα μέρος παιδιά που έχουν σοβαρές μαθησιακές δυσκολίες και από το άλλο παιδιά με διάφορα οργανικά προβλήματα.

Ο συνδυασμός διαφόρων συσκευών και προσθετικών εξαρτημάτων της μικροηλεκτρονικής τεχνολογίας και η σωστή χρήση τους στα σχολεία που εφαρμόζουν ειδικά εκπαιδευτικά προγράμματα πετυχαίνει, σε πολλές περιπτώσεις, να βοηθάει τα παιδιά με ειδικές ανάγκες να ξεπεράσουν

ορισμένες δυσκολίες, κυρίως αισθητηριακές, κινητικές και ψυχολογικές, που τα εμποδίζουν στην επικοινωνία, τη σχολική και κοινωνική ενσωμάτωση και την εργασιακή αποκατάσταση και αλληλαποδοχή από τους συμπολίτες τους.

Όταν άρχισαν να χρησιμοποιούνται οι μικροϋπολογιστές ως εκπαιδευτικά εργαλεία, οι εκπαιδευτικοί της ειδικής αγωγής και οι ειδικοί μελετητές των εφαρμογών της νέας τεχνολογίας στην εκπαίδευση ατόμων με ειδικές ανάγκες, προσπάθησαν να αναπτύξουν προγράμματα και να επινοήσουν κατάλληλα περιφερειακά εξαρτήματα σύνδεσης με τον μικροϋπολογιστή, για μια καλύτερη και αποδοτικότερη εκπαίδευση των αναπήρων. Σ' αυτό το σημείο πρέπει να αναφερθεί το σύστημα Braille και το σύστημα Bliss. Το Braille είναι σύστημα επικοινωνίας για τυφλά παιδιά με δυνατότητα προσαρμογής του σε μικροϋπολογιστή, ενώ το σύστημα Bliss παρέχει ένα μέσο επικοινωνίας σ' εκείνα τα άτομα που δεν μπορούν να μιλήσουν και έχουν ανεπαρκή κινητικό έλεγχο, όπως τα σπαστικά παιδιά. Το σύστημα αυτό αποτελείται από σύμβολα που έχουν μία λογική σύνδεση και έχει αποδειχθεί πολύ αποτελεσματικό σε άτομα που υποφέρουν από εγκεφαλική παράλυση.

Έτσι σήμερα οι Η/Υ χρησιμοποιούνται:

Α) Από τυφλούς για να διαβάζουν για χάρη τους, μόνοι τους, με τη χαρακτηριστική μεταλλική φωνή τους οποιοδήποτε βιβλίο, ενώ υπάρχει και δυνατότητα μεταγραφής του βιβλίου στη γραφή Braille, την οποία διαβάζουν οι τυφλοί με την αφή και παράλληλα μπορούν να γράφουν με την ειδική γραφομηχανή που διαθέτει το σύστημα για τυφλούς και μη.

Υπάρχουν ειδικά πληκτρολόγια τα οποία χρησιμοποιούνται με ένα απλό άγγιγμα ενός τετράγωνου τμήματος που περιέχει σύμβολα που το καθένα αντιπροσωπεύει μια ειδική λέξη ή εντολή. Ο Υπολογιστής παίρνει το μήνυμα και μ' αυτό τον τρόπο το παιδί κατορθώνει να γράψει προτάσεις που εμφανίζονται στην οθόνη του Η/Υ, ή τυπώνονται στο συνδεδεμένο με τον υπολογιστή εκτυπωτή, ή ακόμα ακούγονται από το συνθέτη φωνής με την προσθήκη στον Η/Υ μικροκυκλωμάτων (chips).

Β) Από σωματικώς ανάπηρους. Ο Jorgen Hansen (1985), αναφέρεται σε ένα κορίτσι 11 ετών με ολική παραλυσία και με μόνη δυνατότητα σωματικής κίνησης το ανοιγοκλείσιμο των βλεφάρων, που φοιτά σε ειδικό σχολείο στην Αγγλία. Ουσιαστικά αυτό το κορίτσι δεν έχει

δυνατότητα επικοινωνίας, αλλά με τη χρήση της νέας τεχνολογίας την αναπτύσσει. Δηλαδή με το βλέμμα της ενεργοποιεί έναν ειδικό διακόπτη ο οποίος θέτει σε λειτουργία σχετικό πρόγραμμα ενός υπολογιστή με τελικό προϊόν τη δημιουργία ενός κειμένου, που αφενός φαίνεται τμηματικά στην οθόνη, αφετέρου γράφεται στον εκτυπωτή του μικροϋπολογιστή ως πλήρες κείμενο. Το κορίτσι αυτό, καθώς και άλλοι άνθρωποι με παρόμοιο είδος αναπηρίας, κατορθώνουν τώρα να επικοινωνούν.

Γ) Από ανάπηρους ως προς την ακοή: Στο Παρίσι, στο Γαλλικό Ινστιτούτο, στο τμήμα για κουφά παιδιά, οι μαθητές μαθαίνουν την προφορική γλώσσα παίζοντας με τον Υπολογιστή. Ο Υπολογιστής παρουσιάζει μια οπτική απεικόνιση της φωνής τους στην οθόνη του. Για παράδειγμα, μιλώντας ο μαθητής σε ένα μικρόφωνο συνδεδεμένο με τον υπολογιστή του, βλέπει στην οθόνη μια καμήλα να προχωράει ανάμεσα σε φοινικόδενδρα, η δε ποιότητα της προφοράς του κάθε παιδιού είναι αυτή που θα καθορίσει, αν η καμήλα θα φτάσει στον προορισμό της ή θα συγκρουσθεί στην πορεία της με κάποιο από τα δένδρα εμπόδια και θα σταματήσει.

Δ) Από ανάπηρους ως προς τη γλωσσική επικοινωνία. Αυτή η κατηγορία αναπήρων περιλαμβάνει άτομα που καταλαβαίνουν τη γλώσσα του περιβάλλοντός τους, αλλά δεν μπορούν να μιλήσουν, έτσι ώστε να γίνονται κατανοητοί από τους άλλους. Γι΄ αυτά τα άτομα το σύστημα Bliss είναι ένα από τα συχνότερα χρησιμοποιούμενα συστήματα συμβολικής επικοινωνίας. Αποτελείται περίπου από 1400 σύμβολα, που το καθένα καλύπτει μια ειδική έννοια. Τα σύμβολα μπορούν να συνδυασθούν σε ειδικά φύλλα και έτσι το ανάπηρο παιδί κατορθώνει να επικοινωνήσει.

Ακόμη οι Η/Υ χρησιμοποιούνται από:

α) άτομα με νοητική καθυστέρηση,
β) άτομα με προβληματική συμπεριφορά,
γ) άτομα με πολλαπλές διαταραχές,
δ) άτομα με δυσλεξία κ.λ.π.

Στη χώρα μας, δυστυχώς δεν υπάρχει ακόμη κανένα Κέντρο/φορέας για θέματα εφαρμογών-χρήσης της Πληροφορικής στην Ειδική Αγωγή για αυτό και θα πρέπει να επιταχυνθεί η διαδικασία για την αξιοποίηση και εφαρμογή της Πληροφορικής σε όλη την έκταση της Ειδικής Αγωγής. Οι δυνατότητες που προσφέρει η εξέλιξη της Πληροφορικής και παράλληλα της Ιατρικής για

τη βελτίωση της ποιότητας ζωής και εκπαίδευσης των αναπήρων και μειονεκτούντων ατόμων είναι σημαντικές και εξελισσόμενες.

Η πρόοδος όμως της νέας τεχνολογίας αυξάνει την ευθύνη μας όσον αφορά τη χρήση της στην εκπαίδευση των αναπήρων ατόμων, γιατί αφενός θα μεγαλώνει η εξάρτηση των αναπήρων από την εξελισσόμενη τεχνολογία και τους διδάσκοντες και αφετέρου η χρήση του Η/Υ και των λοιπών ηλεκτρονικών εξαρτημάτων στην εκπαίδευση εμπεριέχει τον κίνδυνο, για τον ανάπηρο, μείωσης των ευκαιριών ανάπτυξης των διαπροσωπικών σχέσεων, που αποτελούν παράγοντα ζωτικής σημασίας για τα μειονεκτούντα άτομα.

Η συνεχώς εξελισσόμενη τεχνολογία της Πληροφορικής, μας υπόσχεται ότι θα χαρίσει το κλειδί για να ελευθερώσει τα παιδιά με Ειδικές Ανάγκες από τη φυλακή της αναπηρίας τους, ώστε να μπορέσουμε να τα εντάξουμε ισότιμα στην κοινωνία που ζούμε.

Τηλεμάθηση (μάθηση από απόσταση)

Η μάθηση από απόσταση αναφέρεται κυρίως στη εκπαίδευση μαθητών που βρίσκονται μακριά από το χώρο εκπαίδευσης κυρίως για τους εξής λόγους:

1. Έλλειψη χρόνου
2. Μεγάλη απόσταση μεταξύ κατοικίας και σχολείου
3. Φυσικές αναπηρίες μαθητών
4. Μεγάλα έξοδα μετακίνησης και πολλές φορές ανυπαρξία μεταφορικού μέσου για τη μετακίνηση των μαθητών
5. Εργαζόμενοι μαθητές

Πέρα όμως από τις παραπάνω κατηγορίες μπορούν και οι μαθητές, φοιτητές να έχουν τη δυνατότητα της τηλεμάθησης, παρόλο ότι παρακολουθούν κάποιο σχολείο, καθώς και οι εργαζόμενοι που θέλουν να αυξήσουν τις γνώσεις τους, τα άτομα που διαμένουν σε αγροτικές περιοχές μακριά από κέντρα εκπαίδευσης, οι μεσήλικες, τα άτομα με χαμηλά εισοδήματα, τα άτομα που τους αρέσει να σπουδάζουν συνεχώς και ακόμη οι φυσικά ανίκανοι (ανεξαρτήτως ηλικίας) μπορούν, χωρίς να μετακινηθούν από τα σπίτια τους να επωφεληθούν των προγραμμάτων αυτών.

Αρχικά, στη μάθηση από απόσταση χρησιμοποιήθηκε το mail. Αργότερα όμως εισήχθησαν οι κασέτες, το βίντεο, η τηλεόραση, το τηλέφωνο. Το βασικό πρόβλημα αυτού του είδους εκπαίδευσης είναι η απομόνωση των μαθητών στον ιδιωτικό τους χώρο, απ' όπου δεν μπορούν να έρθουν σε επαφή με εργαστήρια, ασκήσεις, συζητήσεις, σπουδαστές με τα ίδια ενδιαφέροντα, το εκπαιδευτικό περιβάλλον γενικότερα και επιπλέον η δυσκολία να έρχονται σε επαφή με την γραμματειακή υποστήριξη του Πανεπιστημίου. Το φαινόμενο αυτό περιγράφεται από τον Thiagarajan (1978), σαν "μοναξιά της εκπ/σης από απόσταση".

Στα παραπάνω αναφερόμενα προβλήματα γίνεται μια προσπάθεια ελαχιστοποίησής τους προς όφελος της εκπαίδευσης από απόσταση, που θα δίνει καλύτερη εκπαίδευση με χαμηλότερο κόστος σ' ένα πληθυσμό που θα διαμένει σ' οποιαδήποτε γεωγραφική περιοχή.
Καθώς όμως οι μαθητές στην τηλεμάθηση είναι κυρίως ενήλικες, μπορούν ευκολότερα να κατανοήσουν τις ανάγκες τους, τις απαιτήσεις των μαθημά-

των και τις δυνατότητες που τους προσφέρονται από αυτού του είδους την εκπαίδευση.

Η διδασκαλία από απόσταση μοιάζει με την κατά πρόσωπο διδασκαλία. Οι εμπειρίες όμως από τη διδασκαλία από απόσταση είναι διαφορετικές, διότι οι συμμετέχοντες βρίσκονται σε ξεχωριστούς χώρους, χωρίς να έχουν τη δυνατότητα να βλέπουν ο ένας τον άλλο και η επικοινωνία μεταξύ τους γίνεται με ηλεκτρονικά μέσα. Συνεπώς οι απαντήσεις δε δίνονται σε πραγματικό χρόνο. Αυτές δε οι καταστάσεις αλλάζουν το ρόλο του δασκάλου. Ο δάσκαλος δεν έρχεται σε άμεση επαφή με τον μαθητή και οι ερωτήσεις που κάνει καθώς και οι απαντήσεις που παίρνει από τον μαθητή περνούν μέσα από τα ηλεκτρονικά μέσα και έτσι η κλασική θέση του δασκάλου διαφοροποιείται.

Τι θα μπορούσε όμως να γίνει με τα παιδιά της προσχολικής ηλικίας, όπου η θέση του δασκάλου είναι σημαντική ;

Στην ηλικία των 2 ετών τα παιδιά μπορούν με ασφάλεια να χειρίζονται μερικές από τις οικιακές συσκευές. Είναι γνωστό όμως ότι τόσο μικρά παιδιά δεν είναι ικανά να χειριστούν από μόνα τους τέτοια μηχανήματα, αλλά είναι ακροατές και θεατές της λειτουργίας τους, υπό την επιτήρηση πάντα των ενηλίκων. Τα παιδιά της προσχολικής ηλικίας έχουν πρόσβαση στο ραδιόφωνο, την τηλεόραση και όλες τις άλλες τεχνολογικές εφαρμογές που υπάρχουν στο σπίτι. Παίζοντας μ' αυτές τις συσκευές και τα συστήματα, προτιμούν να εξερευνούν τη λειτουργία τους ή να ακούν το ραδιόφωνο, ή να παρακολουθούν κάποιο πρόγραμμα στην τηλεόραση, από το να εξάγουν συμπεράσματα για τη λειτουργία τους, αφού συμβουλευτούν τις οδηγίες χρήσεώς τους. Με αυτό τον τρόπο καλλιεργούν ποικίλες νοητικές λειτουργίες, όπως τη μνήμη, την κρίση, την παρατήρηση κ.λ.π. Η νέα Πληροφορική Επιστήμη αυξάνει έτσι σίγουρα την ποικιλία και την ποσότητα της ανεπίσημης γνώσης, που μπορούν οι γονείς να προσφέρουν στα παιδιά τους σ' αυτή την ηλικία. Θα είναι ωστόσο λάθος να πούμε ότι η Πληροφορική έχει τη μεγαλύτερη επίδραση σ' αυτή την ηλικία στην προσφορά γνώσης.

Αξίζει επίσης να αναφέρουμε ότι έχει παρατηρηθεί ένας συσχετισμός ανάμεσα στις υψηλές σχολικές επιδόσεις και την ποσότητα των ανεπίσημων γνώσεων που προσλαμβάνουν τα παιδιά σ' αυτή την ηλικία. Τα μεγέθη είναι ανάλογα και το φαινόμενο θυμίζει την αναλογία που υπάρχει ανάμεσα στις υψηλές σχολικές επιδόσεις και σ' ένα πλούσιο μορφωτικό οικογενειακό περιβάλλον.

Μέσα απ' αυτό το παιγνίδι θα μπορούσαν τα παιδιά προσχολικής ηλικίας να ωφεληθούν από τα προγράμματα τηλεμάθησης, όταν αυτά δημιουργηθούν, έτσι ώστε να είναι προσαρμοσμένα στις δικές τους ανάγκες.

Ο Riddle (1994) παρατήρησε ότι πολλοί εκπαιδευτές μαθητών από απόσταση πιστεύουν ότι ο τύπος του μαθητή που πρόκειται να είναι επιτυχημένος στη μάθηση από απόσταση είναι ενήλικες, που δε χρειάζονται τόσο πολύ την προσωπική επαφή με τον εκπαιδευτή-δάσκαλο και κατά συνέπεια είναι εκείνοι οι μαθητές που έχουν υψηλό δείκτη αυτοπεποίθησης, που πιστεύουν στις δυνάμεις τους και έχουν υψηλό βαθμό αυτοσυγκράτησης στα εκάστοτε τεκταινόμενα. Γενικά είναι αυτοί που δε χρειάζονται τη συστηματική βοήθεια που παρέχει το κλασικό σύστημα διδασκαλίας.

Καθώς όμως τα εκπαιδευτικά προγράμματα για μάθηση από απόσταση πληθαίνουν και ο εκπαιδευτικός πληθυσμός στον οποίο αναφέρονται συνεχώς μεγαλώνει, το βασικό, προς εξέταση, θέμα που μένει είναι πώς θα γίνονται με επιτυχία αυτά τα προγράμματα, ώστε να προσφέρουν το καλύτερο δυνατό.

18 ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ

Από το τέλος του 1940 όλοι οι επιστήμονες είχαν στρέψει την προσοχή τους στο πώς θα κατασκευάσουν Η/Υ που θα λειτουργούν κατά τέτοιο τρόπο, έτσι ώστε τα αποτελέσματα των εργασιών που κάνουν να μην διαφέρουν από εκείνα που θα έκανε η ανθρώπινη σκέψη.
Σίγουρα το να πραγματοποιηθεί αυτή η σκέψη είναι ένα επίτευγμα. Με την πάροδο όμως του χρόνου βλέπουμε ότι, αν -για παράδειγμα- ένας Η/Υ έχει αρκετά δεδομένα, μπορεί να απαντήσει σε πολλές ερωτήσεις, ταχύτερα απ' ότι ένας ανθρώπινος νους.
Εάν ο άνθρωπος σκέφτεται να κατασκευάσει Η/Υ με τους οποίους θα μιμηθεί τον ανθρώπινο εγκέφαλο, τότε θα πρέπει να βρει τον τρόπο με τον οποίο όλα τα νεύρα του εγκεφάλου λειτουργούν, πράγμα το οποίο θεωρείται από τα δυσκολότερα στην επιστήμη.

Διάφορες προσπάθειες έγιναν από την εποχή ακόμα των αρχαίων Αιγυπτίων και Ρωμαίων, όταν εκείνοι κατασκεύαζαν αγάλματα τα οποία έδιναν την αίσθηση ότι μπορούν να μιλούν και να προφητεύουν, μέχρι τις ημέρες μας, που επιχειρείται η κατασκευή της ηλεκτρονικής σκέψης.

Ο φιλόσοφος Hobbes (1588-1679) θεωρείται από πολλούς σαν ο πρόδρομος της τεχνητής νοημοσύνης για τις απόψεις του πάνω στη δομημένη σκέψη. Ο ίδιος υποστήριζε ότι:

- η σκέψη συγκροτείται από συμβολικές πράξεις όπως αυτές που κάνουμε με χαρτί και μολύβι.
- η σκέψη είναι διαυγέστερη και πιο ορθολογική όταν ακολουθεί μεθοδικούς κανόνες.

Ο John McCarthy (1927-) επινόησε τον όρο τεχνητή νοημοσύνη και το 1958 ίδρυσε στο M.I.T. - ΗΠΑ μαζί με τον Marvin Minsky το πρώτο

εργαστήριο τεχνητής νοημοσύνης, ενώ το 1963 στο Stanford University-ΗΠΑ, έκανε ένα νέο υπολογιστή τον LISP.

Το 1952 οι υπολογιστές εντυπωσίασαν τους Αμερικάνους τηλεθεατές, προβλέποντας τα αποτελέσματα των εκλογών αρκετές ώρες πριν την τελική ανακοίνωσή τους.

Προς το τέλος της δεκαετίας του 1950 άρχισε να αναπτύσσεται σταδιακά η τεχνητή νοημοσύνη. Η πρώτη φάση ασχολείται με την καθαρή επιστήμη, (χρησιμοποιεί τους Η/Υ σαν πειραματικά εργαλεία για να ερευνήσει θεωρίες σχετικά με το πώς σκέφτονται οι άνθρωποι). Η δε δεύτερη ενδιαφέρεται για το μηχανικό, τεχνικό μέρος, των Η/Υ έτσι ώστε να μπορούν να γίνουν πιο εύχρηστοι από όλους όσους ασχολούνται με αυτούς. Στην τεχνητή νοημοσύνη προγραμματίζεται μια δέσμη γενικών πληροφοριών και αρχών.

Πολύ απλό παράδειγμα τεχνητής νοημοσύνης στις ημέρες μας είναι τα γνωστά παιχνίδια σκακιού στους Η/ Υ.

Αναγνώριση αντικειμένων μέσω Η/Υ

Ερευνητές πάνω στην τεχνητή νοημοσύνη έχουν προσπαθήσει να δημιουργήσουν μηχανές που θα μπορούσαν να αναγνωρίσουν αντικείμενα σχεδόν με τον ίδιο τρόπο όπως και η ανθρώπινη όραση. Η ανθρώπινη όραση δεν μπορεί εύκολα να ερευνηθεί σ' ένα εργαστήριο και γι' αυτό είναι εξαιρετικά δύσκολο να χρησιμοποιηθεί σαν βάση για τη δημιουργία ενός συστήματος σε Η/Υ.

Κάθε σύστημα "μηχανικής" όρασης πρέπει να είναι ικανό να φτιάξει, να αναλύσει και να παρουσιάσει εικόνες. Η μηχανή απομονώνει χαρακτηριστικά στις εικόνες, όπως χρώμα, σχήμα, υφή και προσπαθεί κατόπιν να συνδέσει αυτά τα χαρακτηριστικά με πληροφορίες που είναι στη μνήμη της. Η σύνδεση αυτή των χαρακτηριστικών των εικόνων είναι από τις βασικότερες λειτουργίες πάνω στην τεχνητή νοημοσύνη ενός Η/Υ.

Αναλυτικότερα, για τη δημιουργία μιας εικόνας ακολουθούνται τα παρακάτω βήματα:

- Ο ορισμός των περιφερειακών γραμμών που καθορίζουν το σχήμα ενός αντικειμένου, γίνεται μέσα από μία κάμερα η οποία καθορίζει τα όρια της περιφέρειας και αναλύει τις διαβαθμίσεις φωτεινότητας της εικόνας.
- Το βάθος του αντικειμένου (δηλ. να οριστεί αυτό τρισδιάστατα), ορίζεται από συγκεκριμένα σημεία του αντικειμένου που δίνουν τις διαστάσεις του.
- Επίσης με διάφορες μεθόδους αναλύεται η υφή κατ το χρώμα του αντικειμένου.

Αφού γίνει αυτή η ανάλυση, τότε ο Η/Υ προσπαθεί να το παρομοιάσει με κάποιο άλλο αντικείμενο, του οποίου τα χαρακτηριστικά είναι αποθηκευμένα στη μνήμη του και με το οποίο ταιριάζει.

Το 1956 ο Allen Newell και Herbert Sinon έκαναν ένα πρωτοποριακό βήμα στην τεχνητή νοημοσύνη. Το πρόγραμμα που έκαναν για Η/Υ, ονομαζόμενο Logic Theorisit, μπορούσε να αποδείξει θεωρήματα συμβολικής λογικής.
Η στρατηγική της επίλυσης των προβλημάτων βασίζεται στην κατάτμηση ενός προβλήματος σε επιμέρους προβλήματα με σκοπό τη λύση του επιμέρους προβλήματος πρώτα και μετά του συνολικού.

Expert Systems

Τα Expert Systems (έμπειρα συστήματα) είναι προγράμματα Η/Υ φτιαγμένα για να βοηθήσουν τόσο στο να γνωρίσουμε σε βάθος τα διάφορα θέματα, όσο στο να εξετάσουν τη σχέση αιτίου-αποτελέσματος σε έναν ορισμένο τομέα. Έχουν χρησιμοποιηθεί σε διάφορους τομείς όπως π.χ. στην ανακάλυψη του υπεδάφους, την κατασκευή ηλεκτρονικών υπολογιστών, την ψυχολογία.

Ενα τυπικό Expert System αποτελείται από δύο μέρη, δηλ. το ένα μέρος της γνώσης και το άλλο της κατασκευής ενός προγράμματος που κυρίως αποτελείται από προτάσεις IF...THEN.

Συμπερασματικά θα λέγαμε ότι οι μηχανές θα μπορούσαν να γίνουν έξυπνες και να μοιάσουν στον ανθρώπινο εγκέφαλο τότε μόνον, όταν θα μπορούν να βαρεθούν να διαβάζουν κάτι παλιό ή να εντυπωσιαστούν από κάτι καινούργιο υποστηρίζει ο K. Schank από το Παν/μιο του Yale.

Έχοντας, και όχι μόνο, τη δυσκολία στην αναγνώριση του αντικειμένου, οι Η/Υ οποιουδήποτε τύπου, δεν μπορούν ακόμη να ασχοληθούν και να αντιληφθούν πράγματα όπως ο χρόνος, αίτιο και αποτέλεσμα, σύνθετες σκέψεις κλπ. Δεδομένης της πολυπλοκότητας των προβλημάτων οι περισσότεροι ερευνητές στις αρχές του 1980 κατέληξαν στο συμπέρασμα ότι χρειάζονται νέοι τύποι υπολογιστών. Η πέμπτη γενιά Η/Υ που κυκλοφόρησε από το 1980 και μετά, βασίστηκε στην ιδέα της παράλληλης επεξεργασίας. Προφανώς το κύριο πλεονέκτημα αυτής της επεξεργασίας είναι το ότι μιμείται την ίδια τη φύση.

Η μεγαλύτερη ίσως εφαρμογή της AI (τεχνητής νοημοσύνης) είναι στο χώρο του διαστήματος, που η αρτιότερη κατανόησή του εκ μέρους μας υπόσχεται πολλά στις επόμενες γενιές.

Τελικά όμως οι εκλεπτισμένες ικανότητες της ανθρώπινης νοημοσύνης, όπως π.χ. η ικανότητα της μάθησης από την εμπειρία και η αναπροσαρμογή της νοοτροπίας ανάλογα, θα παραμένουν πιθανώς τα βασικά θέματα με τα οποία θα ασχοληθεί η τεχνητή νοημοσύνη στο μέλλον.

Στον ερευνητικό τομέα της τεχνητής νοημοσύνης (AI) είναι εμφανείς τρεις διαφορετικές προσεγγίσεις δηλ. οι κάτωθι:

1. Γίνεται ανάλυση της υπάρχουσας πραγματικότητας και πώς αυτή εφαρμόζεται και αναπτύσσεται σε μηχανές, με τους ίδιους κανόνες και αρχές όπως και στους ανθρώπους.

2. Η ανθρώπινη ευφυΐα εμφανίζεται με διαφορετικές μορφές, τις οποίες προσπαθεί να μιμηθεί μία μηχανή στα μέτρα του δυνατού.
Αυτή η προσέγγιση έχει εφαρμογές και στην Ψυχολογία.

3. Ο Η/Υ μπορεί να χρησιμοποιηθεί σαν ένας καλός συνεταίρος, αν είναι εφοδιασμένος με το κατάλληλο λογισμικό.

ΠΑΡΑΡΤΗΜΑΤΑ

Ο ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΤΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Α ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΚΑΙ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ

Πως προγραμματίζουμε

Οι Γλώσσες του Υπολογιστή

Βασικό εργαλείο για την μετάδοση γνώσεων μεταξύ των όντων και την ανάπτυξη της νοημοσύνης τους είναι η γλώσσα. Κάθε κατηγορία όντων ανάλογα με το επίπεδο της νοημοσύνης που έχει, προσδιορίζει (και προσδιορίζεται) από τις δυνατότητες της γλώσσας που χρησιμοποιεί (πλούτος εκφράσεων, ευελιξία κλπ). Από τη διαπίστωση αυτή προκύπτει ότι ο άνθρωπος το κατά τεκμήριο περισσότερο νοήμον πλάσμα στον πλανήτη μας θα έχει και το τελειότερο μέσον επικοινωνίας, την πιό πλούσια γλώσσα. Η γλώσσα αυτή που αποτελεί και το όριο, το ιδανικό για τις γλώσσες όλων των άλλων κατηγοριών όντων λέγεται φυσική ανθρώπινη γλώσσα.

Φυσική Γλώσσα Μηχανής και Ξένες Γλώσσες

Σαν "νοήμον" όν (έστω και τεχνητά), ο Υ έχει κι αυτός τη δική του γλώσσα-μέσον επικοινωνίας. Η γλώσσα αυτή είναι χαμηλού επιπέδου μια και η νοημοσύνη της μηχανής είναι χαμηλή και λέγεται γλώσσα μηχανής. Η γλώσσα μηχανής λαμβάνει υπόψη τους περιορισμούς της μηχανής να αναγνωρίζει μόνο δυαδικούς αριθμούς οπότε και η ίδια η γλώσσα θα πρέπει τελικά να εκφράζεται στο δυαδικό σύστημα.

Στη γλώσσα αυτή θα πρέπει κανονικά ο άνθρωπος να γράφει τις οδηγίες - προγράμματα για να τις καταλαβαίνει και να τις εκτελεί ο Υ. Πράγματι στα πρώτα χρόνια χρήσης του Υ (δεκαετία του 50) όταν ο Υ χρησιμοποιούνταν κυρίως σε ερευνητικά εργαστήρια οι επιστήμονες που το χρησιμοποιούσαν

έγραφαν τα προγράμματά τους σε γλώσσα μηχανής. Η δουλειά δεν ήταν καθόλου εύκολη και απαιτούσε σημαντική προσπάθεια από τους χρήστες και ειδικές γνώσεις εσωτερικής λειτουργίας της μηχανής, που οι πρώτοι χρήστες-επιστήμονες βέβαια διέθεταν.

Ηταν φανερό όμως πως για να γίνει ευρύτερη η χρήση του Υ έπρεπε η επικοινωνία με τον άνθρωπο να απλοποιηθεί. Να μπορεί δηλ ο άνθρωπος να γράφει τις οδηγίες του σε μιά πιό απλή γλώσσα πιό κοντινή στη δική του φυσική γλώσσα, την οποία όμως θα καταλάβαινε και ο Υ. Η γλώσσα αυτή θα ήταν ξένη στον Υ που φυσικά καταλαβαίνει μόνο τη δική του γλώσσα μηχανής.

Επρεπε λοιπόν ο Υ να "μάθει" ξένες γλώσσες. Οπως ακριβώς ο έλληνας ή ο κινέζος που έχουν δύσκολες γλώσσες πρέπει να μάθουν ξένη γλώσσα (πχ αγγλικά) για να μπορούν να επικοινωνούν με πιό πολύ κόσμο έτσι και η μηχανή αν "ήθελε" να έχει πιό πολλούς χρήστες - προγραμματιστές έπρεπε να μάθει την γλώσσα τους. Η συνειδητοποίηση αυτή ενεργοποίησε την έρευνα και την προσπάθεια του ανθρωπου να "διδάξει" στον Υ γλώσσες υψηλότερου επιπέδου από την δική του γλώσσα μηχανής που να μοιάζουν με την ανθρώπινη φυσική γλώσσα με όριο και ιδανικό τη γλώσσα αυτή έτσι ώστε κάποτε η επικοινωνία ανθρωπου-μηχανής να γίνει άμεση και φυσική. Για διάφορους λόγους η φυσική ανθρώπινη γλώσσα που προσπαθούν να προσεγγίσουν οι γλώσσες προγραμματισμού είναι η αγγλική (διεθνοποίηση της αγγλικής, τα κυριότερα κέντρα έρευνας και ανάπτυξης Υ είναι στην Αμερική κλπ).

Στο πρώτο στάδιο της προσπάθειας αυτής αναπτύχθηκαν οι λεγόμενες συμβολικές ή **Assembly** γλώσσες. Οι γλώσσες αυτές διατηρούσαν τις βασικές αρχές της γλώσσας μηχανής με την διαφορά πως αντί να γράφουμε τις εντολές στη αρχική δυαδική τους μορφή τις γράφουμε με μνημονικούς κώδικες (στα αγγλικά) ενδεικτικούς του τί κάνει η εντολή. Για παράδειγμα όταν θέλουμε να χρησιμοποιήσουμε την εντολή πρόσθεσης δύο αριθμών αντί να γράψουμε κατευθείαν τον δυαδικό της κώδικα έστω 0100 (γλώσσα μηχανής) γράφουμε τον μνημονικό-συμβολικό έστω ADD.

Με την πάροδο του χρόνου αναπτύχθηκαν γλώσσες όλο και πιό κοντά στη φυσική γλώσσα του ανθρώπου (αγγλικά). Τέτοιες γλώσες που λέγονται και γλώσσες υψηλού επιπέδου (ΥΕ), υπάρχουν σήμερα πολλές οι πιό γνωστές από τις οποίες είναι οι COBOL, BASIC, FORTRAN, PASCAL κλπ. Οπως συμβαίνει όμως και με τον άνθρωπο έτσι και ο Υ για να "μιλήσει" την ξένη

γλώσσα πρέπει πρώτα να την μάθει. Τί σημαίνει όμως "γνωρίζω" μιά ξένη γλώσσα. Σημαίνει ότι έχω τη γνώση της γλώσσας στο μυαλό μου ώστε να πετυχαίνω την μετάφραση κάθε ξένης φράσης στη γλώσσα μου (για να την καταλάβω) και αντιστρόφως. Αυτή η γνώση δίνεται στον Υ υπό μορφή προγράμματος που λέγεται μεταφραστής και το οποίο πρέπει να βρίσκεται μέσα στη μνήμη του Υ όποτε χρειάζεται να καταλαβαίνει τα προγράμματα - κείμενα που του γράφουν οι χρήστες σ'αυτές τις "ξένες" γλώσσες.

Οι μεταφραστές για γλώσσες assembly λέγονται συνήθως **assemblers** ενώ για γλώσσες υψηλού επιπέδου λέγονται **compilers**. Υπάρχουν compilers για BASIC, COBOL κλπ. Οι compilers αυτοί (δηλ τα μεταφραστικά προγράμματα) γράφονται και διατίθενται από τις εταιρίες υπολογιστών μαζί με τις μηχανές έτσι ώστε οι χρήστες δεν έχουν να κάνουν τίποτα άλλο από να τα φορτώνουν στη μνήμη των Υ δίνοντάς τους έτσι ακαριαία τη γνώση της αντίστοιχης γλώσσας.
Ο compiler μεταφράζει ολόκληρο το πρόγραμμα υψηλού επιπέδου του χρήστη και δημιουργεί το αντίστοιχο πρόγραμμα σε γλώσσα μηχανής το οποίο στη συνέχεια φυλάσσεται και εκτελείται κάθε φορά που το χρειαζόμαστε χωρίς να απαιτείται νέα μετάφραση.

Εκτός από τον compiler ένα άλλο είδος μεταφραστικού προγράμματος είναι ο λεγόμενος **interpreter**. Η διαφορά έγκειται στον τρόπο με τον οποίο γίνεται η μετάφραση. Αντίθετα με τη διαδικασία της μετάφρασης με compiler που είδαμε παραπάνω ο interpreter μεταφράζει και εκτελεί ξεχωριστά κάθε μιά εντολή του προγράμματος του χρήστη κάθε φορά που γίνεται εκτέλεση του προγράμματος. Δηλαδή η διαδικασία της μετάφρασης είναι συνυφασμένη με την εκτέλεση και επαναλαμβάνεται κάθε φορά που χρειάζεται εκτέλεση του προγράμματος του χρήστη. Η μέθοδος αυτή της μετάφρασης είναι καλή για τον προγραμματιστή (τον βοηθάει να επισημαίνει και να διορθώνει τα λάθη του προγράμματός του) αλλά κακή για το πρόγραμμα (μεγαλώνει τον χρόνο εκτέλεσης). Πολλές γλώσσες (πχ. BASIC) διαθέτουν και compiler και interpreter, οπότε μας συμφέρει να χρησιμοποιούμε τον interpreter για τις πρώτες ροές του προγράμματος (δοκιμές) και όταν βεβαιωθούμε πως αυτό είναι σωστό να το μεταφράσουμε με τον compiler και στη μορφή αυτή να το φυλάξουμε και να το τρέχουμε στις κανονικές ροές του.

Συμπερασματικά μπορούμε να πούμε πως οι γλώσσες Υψηλού Επιπέδου, εύχρηστες και εύκολες στην εκμάθηση και χωρίς απαιτήσεις για ειδικές γνώσεις μηχανής, επέτρεψαν σε ένα ευρύ φάσμα χρηστών να μάθουν να

χρησιμοποιούν τον Υ στο περιβάλλον της δουλειάς τους. Στη συνέχεια η χρήση πακέτων και ο ερχομός των micros αποτέλεσαν τους ουσιαστικούς καταλύτες για την σημερινή τάση γενίκευσης της χρήσης των Υπολογιστών.

Γλώσσες Προγραμματισμού- Γενιές Software

Τις γλώσσες προγραμματισμού που όπως είδαμε και παραπάνω αποκτούν χαρακτηριστικά όλο και πιο υψηλού επιπέδου συχνά τις χαρακτηρίζουμε πως ανήκουν σε γενιές.

Οι χαμηλού επιπέδου γλώσσες (μηχανής και assembly) χαρακτηρίζονται συχνά σαν γλώσσες **πρώτης και δεύτερης γενιάς.**

Οι γλώσσες υψηλού επιπέδου που εμφανίστηκαν στη συνέχεια (BASIC, COBOL, FORTRAN) χαρακτηρίζονται σαν γλώσσες **τρίτης γενιάς**. Κύριο χαρακτηριστικό των γλωσσών αυτών είναι πως η δομή τους ευνοεί την απόδοση αλγορίθμων και διαδικασιών. Στα προγράμματα δηλ που γράφονται με τη βοήθεια τέτοιων γλωσσών περιγράφεται το πρόβλημα καθώς και η διαδικασία επίλυσής του (αλγόριθμος) βήμα προς βήμα. Γιαυτό και οι γλώσσες αυτές συχνά αποδίδονται σαν **αλγοριθμικές** ή διαδικαστικές (procedural).

Η επόμενη γενιά γλωσσών έχει χαρακτηριστικά που αποστασιοποιούν ακόμα περισσότερο τον χρήστη από τη μηχανή. Πραγματικά οι γλώσσες **τέταρτης γενιάς** ή **4GL** (4th Generation Languages) έχουν εντολές που κάνουν τον προγραμματισμό ακόμα πιό εύκολο για τον προγραμματιστή ενώ παρέχουν παράλληλα και ένα ολόκληρο περιβάλλον ανάπτυξης εφαρμογών που επιτρέπει ακόμα και σε απλούς χρήστες να προγραμματίζουν και να αναπτύσσουν εφαρμογές. Παραδείγματα τέτοιων γλωσσών είναι η MAPPER, η CLIPPER (για τους προσωπικούς Υ), η POWERHOUSE.

Οι γλώσσες **πέμπτης γενιάς** είναι οι γλώσσες της εποχής της **Τεχνητής Νοημοσύνης**. Εδώ υπάρχει μιά ουσιαστική διαφορά αντιμετώπισης σε σχέση με τις κλασικές αλγοριθμικές γλώσσες τρίτης γενιάς. Ο χρήστης με τη βοήθεια μιάς τέτοιας γλώσσας θέτει απλά προς τον Υ το προς επίλυση πρόβλημα και του δίνει και τις απαραίτητες βασικές γνώσεις για να το λύσει. Στη συνέχεια αφήνει τον Υ να βρεί μόνος του με τις λεγόμενες ευριστικές μεθόδους - heuristics, την πιό κατάλληλη λύση. Είναι δηλ. μή αλγοριθμικές

γλώσσες που επιτρέπουν στον Υ να καταπιαστεί με προβλήματα και εφαρμογές που παρουσιάζουν ένα βαθμό αβεβαιότητας και δε μπορούν να αναλυθούν σε μαθηματικούς τύπους και να αποδοθούν με ένα αλγόριθμο. Τέτοιες γλώσσες είναι η PROLOG, η LISP, η SMALLTALK κ.ά

Ανάλογα με τα κοινά τους χαρακτηριστικά οι σύγχρονες γλώσσες προγραμματισμού κατατάσονται στις παρακάτω κατηγορίες:

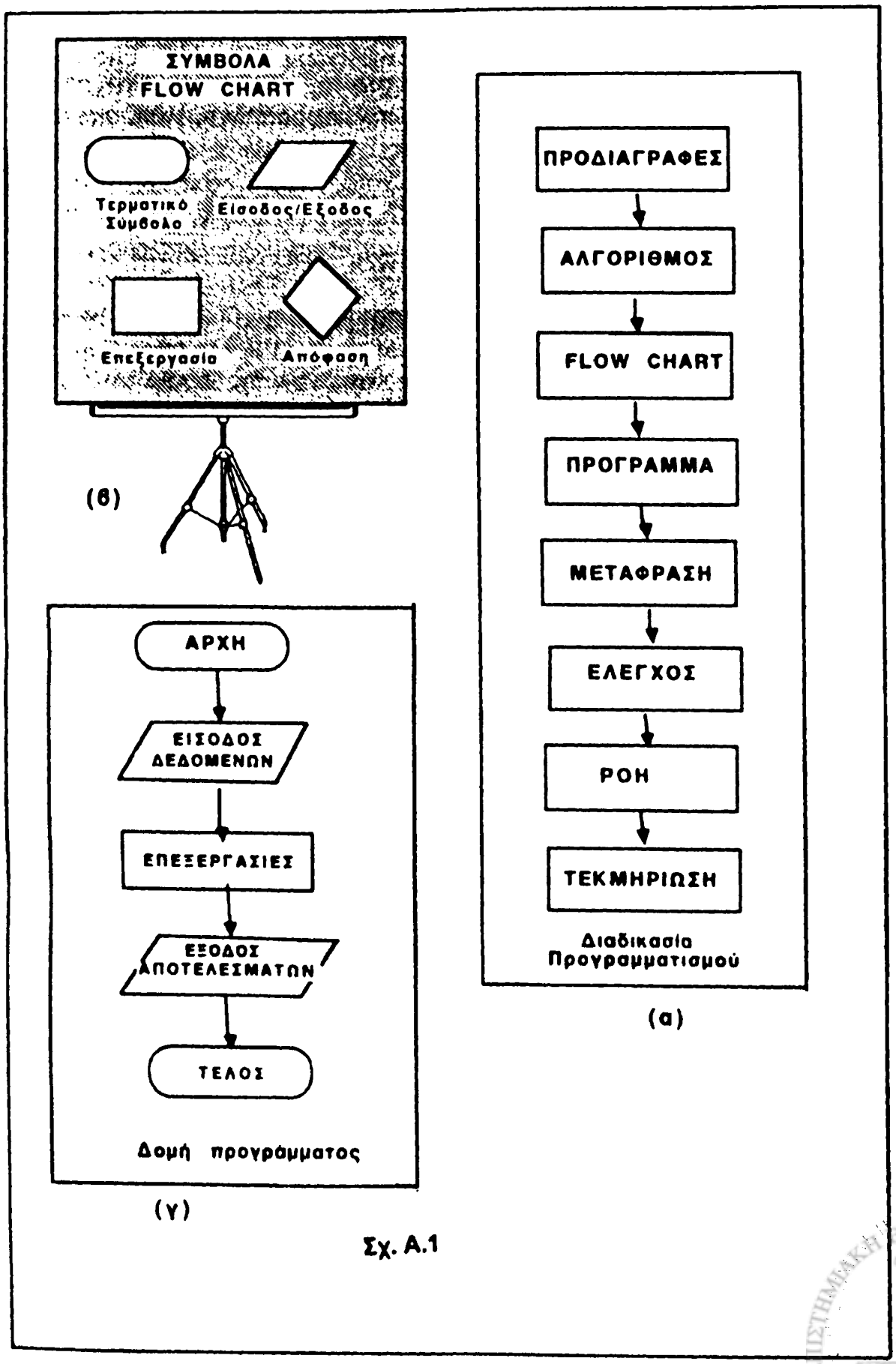
1. **Προστακτικές** (Imperative). Εδώ ανήκουν οι πιο κλασσικές γλώσσες όπως Pascal, C, Fortran. Στις γλώσσες αυτές το συντακτικό τους σχετίζεται με την αρχιτεκτονική του υπολογιστή.
2. **Συναρτησιακές** (Functional). Οι γλώσσες αυτές βασίζονται στην έννοια της συνάρτησης (στο λεγόμενο λ-λογισμικό). Με τις γλώσσες αυτές ελέγχεται η ορθότητα των προγραμμάτων και γίνονται εύκολα μαθηματικοί μετασχηματισμοί προγραμμάτων σε άλλα ισοδύναμα. Στην κατηγορία αυτή ανήκουν οι Lisp, ML, Miranda, Gofer, Haskell, κ.α..
3. **Λογικές** (Logic). Βασικός εκπρόσωπός τους είναι η Prolog. Ολες οι γλώσσες αυτές στηρίζονται στη μαθηματική λογική.
4. **Περιορισμών** (Constraint). Κύριος σκοπός των γλωσσών αυτών είναι να εκφράσουν με φυσικό τρόπο τους διάφορους περιορισμούς που υπάρχουν στο προς επίλυση πρόβλημα. Η πιο γνωστή γλώσσα της κατηγορίας αυτής είναι η GLP.
5. **Αντικειμενοστραφείς** (Object-Oriented). Οι πιο γνωστές γλώσσες της κατηγορίας αυτής είναι οι C++, Smalltalk και Java. Στις γλώσσες ο υπολογισμός γίνεται με τη μορφή αντικειμένων τα οποία ανταλλάσουν μηνύματα μεταξύ τους.
6. **Ροής δεδομένων** (Dataflow). Οι πιο γνωστές γλώσσες της κατηγορίας αυτής είναι οι Lucid, Val, Id, Sisal. Βρήσκουν εφαρμογές στο πεδίο του επιστημονικού υπολογισμού και εκτελούνται πιο αποτελεσματικά στις λεγόμενες αρχιτεκτονικές ροής δεδομένων. Εδώ τα προγράμματα μοιάζουν κύριως με δίκτυα που αποτελούνται από κόμβους και κανάλια.
7. **Παράλληλες** (Parallel). Δημιουργήθηκαν όταν αναπτύχθηκαν οι παράλληλοι υπολογιστές. Οι πιο γνωστές είναι η CCS και η CSP.

Πρόγραμμα και Προγραμματισμός

Το πρόγραμμα και ο προγραμματισμός είναι η πεμπτουσία της χρήσης του Υπολογιστή. Εκφράζει και κατά κάποιο τρόπο ενσαρκώνει την κυριαρχία του ανθρώπου πάνω στην μηχανή.

Ενα πρόγραμμα δεν είναι τίποτα άλλο από ένα κείμενο γραμμένο σε μιά γλώσσα προγραμματισμού. Οπως ένα κείμενο σε οποιαδήποτε γλώσσα έτσι και το πρόγραμμα αποτελείται από προτάσεις, λέξεις, σημεία στίξεως και άλλα σύμβολα αποδεκτά από την υπόψη γλώσσα και κατάλληλα διατυπωμένα ώστε να γίνονται κατανοητά από εκείνους που πρόκειται να τα διαβάσουν δηλ τους υπολογιστές.

Πώς γράφουμε ένα πρόγραμμα σε μιά γλώσσα προγραμματισμού; Μα όπως γράφουμε ένα κείμενο σε μια οποιαδήποτε γλώσσα! Στην αρχή προετοιμαζόμαστε για το τί θέλουμε να γράψουμε δηλ οργανώνουμε τις ιδέες που θέλουμε να αναπτύξουμε. Στη συνέχεια τις διατυπώνουμε όσο μπορούμε καλύτερα χρησιμοποιώντας τις κατάλληλες λέξεις οργανωμένες σε προτάσεις που να έχουν νόημα και να είναι συντακτικά σωστές. Ιδιαίτερη σημασία δίνεται στην εμφάνιση του κειμένου, τα διαστήματα, τα κενά, τον χωρισμό των παραγράφων κλπ. Αν δεν ξέρουμε άριστα τη γλώσσα που χρησιμοποιούμε για να γράψουμε το κείμενο (πχ είναι μιά ξένη γλώσσα) συνήθως δίνουμε το κείμενο να το διαβάσει κάποιος που ξέρει τη γλώσσα καλά, κανουμε τις διορθώσεις που μας υποδεικνύει και τότε μόνο λέμε πως τελειώσαμε και το κείμενο μπορεί να σταλεί στους αποδέκτες του.



Σχ. Α.1

Διαδικασία Προγραμματισμού

Παρόμοια σε γενικές γραμμές είναι η διαδικασία που ακολουθούμε για να γράψουμε και ένα πρόγραμμα (βλ. Σχ. Α.1α) :

Προδιαγραφές. Οι προδιαγραφές του προγράμματος δεν είναι τίποτα άλλο από ένα κατατοπιστικό κείμενο όπου διατυπώνονται σαφώς οι εργασίες που πρέπει να κάνει το πρόγραμμα. Οι προδιαγραφές προκύπτουν από την ανάλυση του συστήματος που γίνεται από τον αναλυτή συστημάτων. Οι προδιαγραφές περιλαμβάνουν και μία περιγραφή της μορφής που έχουν τα δεδομένα εισόδου (πχ. απογραφικά, δελτία μεταβολών) και της μορφής που θέλουμε να έχουν τα αποτελέσματα (εκτυπωτικές καταστάσεις κλπ).

Αλγόριθμος. Στη συνέχεια γράφουμε (στη γλώσσα που μιλάμε) τα απλά αναλυτικά λογικά βήματα που πρέπει να ακολουθήσει ο Υ για να ικανοποιήσει τις παραπάνω προδιαγραφές. Δηλαδή δεν περιοριζόμαστε να πούμε στον Υ τί θα κάνει αλλά του λέμε και πώς θα το κάνει. Δεν του λέμε δηλ. μόνο το πρόβλημα αλλά και τη λύση του. Αυτή η περιγραφή της λύσης ενός προβλήματος ή μιας διαδικασίας με τη μορφή απλών λογικών βημάτων λέγεται αλγόριθμος λύσης.

Λογικό Διάγραμμα (Flow-chart). Πολλές φορές για να είναι η παραπάνω περιγραφή πιό παραστατική, κατανοητή και αυστηρή (να μην επιδέχεται παρερμηνείες) χρησιμοποιούμε ένα παραστατικό τρόπο περιγραφής με τη βοήθεια ειδικών τυποποιημένων συμβόλων (βλ σχ Α.1β). Τί είναι τα σύμβολα αυτά και πώς επιλέχτηκαν. Απλά διαπιστώσανε γρήγορα οι ειδικοί επιστήμονες πως οι οδηγίες που περιλαμβάνονται σε ένα αλγόριθμο ανήκουν σε τέσσερις βασικές κατηγορίες (Σχ. Α.1γ):

μιά ένδειξη αρχής και τέλους του αλγόριθμου
οδηγίες εισόδου δεδομένων και εξόδου αποτελεσμάτων
οδηγίες επεξεργασίας
οδηγίες συσχετίσεων και αποφάσεων

Για κάθε μιά από τις παραπάνω κατηγορίες οδηγιών έδωσαν ένα σχήμα-σύμβολο που θα μπορεί να χρησιμοποιεί ο χρήστης κατάλληλα για να γράψει τον αλγόριθμο χωρίς να χρειάζεται πολλά λόγια. Ενας τέτοιος περιγραφικός αλγόριθμος λέγεται Λογικό διάγραμμα ή διάγραμμα ροής ή Flow Chart

Πρόγραμμα. Απόδοση του αλγόριθμου στη γλώσσα προγραμματισμού που θέλουμε να χρησιμοποιήσουμε. Στο σημείο αυτό έχουμε το κείμενο-πρόγραμμα το οποίο στη συνέχεια θα πρέπει να εισαχθεί στον Υ για έλεγχο ορθότητας και ροή. Το πρόγραμμα γραμμένο στη γλώσσα ΥΕ λέγεται και αρχικό πρόγραμμα ή πηγαίο πρόγραμμα ή source πρόγραμμα

Μετάφραση (compilation) του προγράμματος. Σε οποιαδήποτε γλώσσα ΥΕ και αν είναι γραμμένο το source πρόγραμμά μας, για να "τρέξει" στον Υ πρέπει να μεταφραστεί (από τον αντίστοιχο compiler ή interpreter). Με τον compiler το πρόγραμμα μεταφράζεται στο ισοδύναμό του σε γλώσσα μηχανής πρόγραμμα. Το πρόγραμμα αυτό λέγεται δυαδικό πρόγραμμα ή object πρόγραμμα.

Για να γίνει όμως αυτή η μετάφραση θα πρέπει το πρόγραμμα που έχουμε γράψει (το source) να είναι συντακτικά σωστό. Ο έλεγχος αυτός γίνεται κατά την μετάφραση του προγράμματος από τον μεταφραστή - compiler της γλώσσας, που έχει οπως είπαμε στο προηγούμενο κεφάλαιο όλη τη γνώση της υπόψη γλώσσας. Ο compiler μας υποδεικνύει τα συντακτικά μας λάθη, τα διορθώνουμε και επαναλαμβάνουμε τη compilation μέχρις ότου δεν υπάρχουν λάθη. Τότε λέμε πως το πρόγραμμά μας είναι συντακτικά σωστό και διαθέτουμε το object πρόγραμμα έτοιμο να τρέξει.

Ειδικά στους μικρο-υπολογιστές το object πρόγραμμα δεν εκτελείται έτσι όπως είναι αλλά πρέπει να περάσει από μιά προετοιμασία που λέγεται linking για να δημιουργηθεί η εκτελέσιμη μορφή του που στη συνέχεια εκτελείται. Η προετοιμασία αυτή γίνεται από ένα άλλο πρόγραμμα εσωτερικό της μηχανής (όπως και ο compiler) που λέγεται linker. Στους πιό μεγάλους Υ η διαδικασία αυτή είναι ενσωμετωμένη στους compilers.

Λογικός έλεγχος. Προτού όμως δώσουμε το πρόγραμμα σε κείνους που μας το παράγγειλαν για κανονική ροή και εκμετάλλευση θα πρέπει να το ελέγξουμε και λογικά δηλ να διαπιστώσουμε αν έχουμε γράψει τις σωστές οδηγίες στον Υ ώστε να μας κάνει αυτά που θέλουμε. Για τον λογικό έλεγχο εκτελούμε το πρόγραμμα χρησιμοποιώντας επιλεγμένα δεδομένα (test data) που να καλύπτουν όλες τις δυνατές περιπτώσεις και για τα οποία έχουμε εμείς ήδη με το χέρι από πρίν υπολογίσει τα αποτελέσματα. Στη συνέχεια συγκρίνουμε τα αποτελέσματα που μας δίνει ο Υ με τα δικά μας αποτελέσματα (dry test). Αν συμφωνούν λέμε πως το πρόγραμμά μας είναι σωστό.

Η διαδικασία ελέγχου και διόρθωσης λαθών (συντακτικών και λογικών) λέγεται και debugging (ντεμπάγκινγκ) ή ξεκαθάρισμα λαθών μιά και ιστορικά καθιερώθηκε να αποδίδεται το λάθος με τον όρο bug (ζωύφιο)

Ροή προγράμματος. Αφού εξασφαλίσουμε οτι το πρόγραμμά μας είναι συντακτικά και λογικά σωστό λέμε πως το πρόγραμμα "τρέχει" και το δίνουμε για εκμετάλλευση δηλ για να χρησιμοποιηθεί για την επεξεργασία οποιωνδήποτε δεδομένων στο πλαίσιο της εφαρμογής για την οποία γράφτηκε.

Τεκμηρίωση προγράμματος. Η εργασία του προγραμματιστή δεν τελειώνει με το γράψιμο ενός σωστού προγράμματος. Για να ολοκληρωθεί χρειάζεται ο προγραμματιστής να γράψει και την πλήρη τεκμηρίωση του προγράμματος - program documentation. Αυτή συνήθως περιλαμβάνει μιά πλήρη παρουσίαση όλων των σταδίων ανάπτυξης του προγράμματος όπως είναι οι προδιαγραφές, το flow chart, η περιγραφή της μορφής των δεδομένων εισόδου που δέχεται το πρόγραμμα και των αποτελεσμάτων εξόδου (πχ εκτυπωτικών καταστάσεων), το πρόγραμμα (listing) κλπ. Τα λεπτομερή αυτά στοιχεία θα επιτρέπουν αργότερα στον ίδιο ή σε άλλον προγραμματιστή να μπορεί να καταλάβει το πρόγραμμα και να κάνει αλλαγές που τυχόν θα χρειαστούν.

Δομή Προγράμματος

Είναι γεγονός οτι η γλώσσα προγραμματισμού που χρησιμοποιείται επηρεάζει τον τρόπο που ο προγραμματιστής γράφει το πρόγραμμά του. Μπορούμε όμως να πούμε πως σε γενικές γραμμές κάθε πρόγραμμα σε όποια γλώσσα και αν γραφεται αποτελείται από εντολές που με τη σειρά (βλ Σχ Α.1γ) :

- εισάγουν τα δεδομένα στη μνήμη
- εκτελούν τις επεξεργασίες
- τυπώνουν τα αποτελέσματα

Προδιαγραφές

Θέλουμε να καταρτίζουμε μηχανογραφικά την μηνιαία κατάσταση μισθοδοσίας του προσωπικού μιάς επιχείρησης.

Για κάθε υπάλληλο θα δίδονται ο βασικός του μισθός, τα επιδόματα και οι κρατήσεις του

Το σύστημα θα υπολογίζει και θα τυπώνει το καθαρό πληρωτέο.

Αλγόριθμος

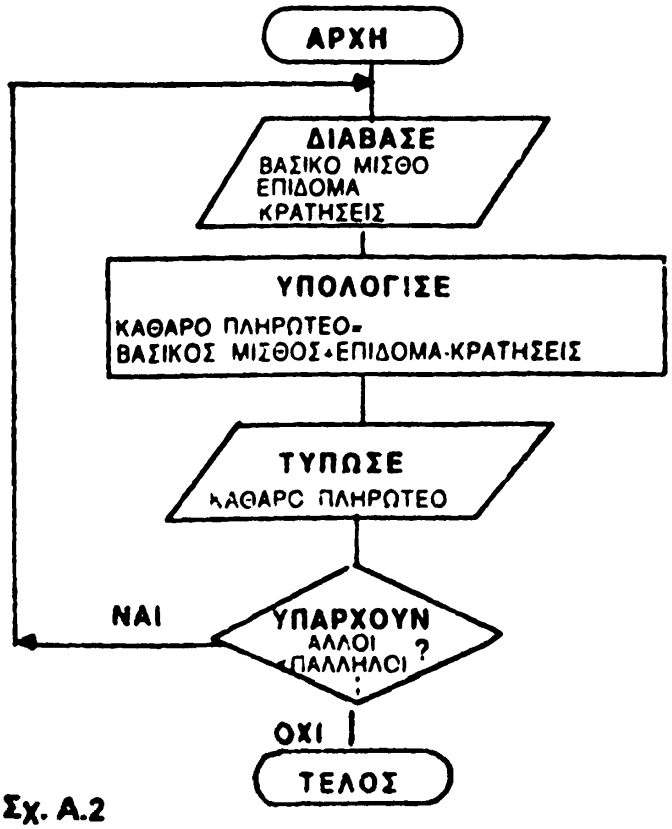
Για κάθε υπάληλο το πρόγραμμα πρέπει να ακολουθεί τα παρακάτω βήματα:

1. ΔΙΑΒΑΣΕ ΒΑΣΙΚΟ ΜΙΣΘΟ, ΕΠΙΔΟΜΑ,ΚΡΑΤΗΣΕΙΣ
2. ΥΠΟΛΟΓΙΣΕ ΚΑΘΑΡΟ ΠΛΗΡΩΤΕΟ=ΒΑΣΙΚΟΣ ΜΙΣΘΟΣ+ΕΠΙΔΟΜΑ-ΚΡΑΤΗΣΕΙΣ
3. ΤΥΠΩΣΕ ΚΑΘΑΡΟ ΠΛΗΡΩΤΕΟ

Στη συνέχεια ελέγχουμε αν υπάρχουν και άλλοι υπάλληλοι

4. ΥΠΑΡΧΟΥΝ ΑΛΛΟΙ ΥΠΑΛΛΗΛΟΙ ;
Αν ΝΑΙ
5. ΠΗΓΑΙΝΕ πίσω στο βήμα 1 και επανάλαβε τη διαδικασία
Αν ΟΧΙ
6. ΤΕΛΟΣ

Flow chart



Σχ. Α.2

Ενα Παράδειγμα

Στο Σχ. Α.2 υπάρχει μιά περιγραφή των πρώτων φάσεων της διαδικασίας του προγραμματισμού που παρουσιάσαμε παραπάνω εφαρμοσμένη στο παράδειγμα της απλής μισθοδοσίας με την οποία οι αναγνώστες μας είναι ήδη εξοικειωμένοι από την αρχή του βιβλίου αυτού.

Συγκεκριμένα παρουσιάζονται οι φάσεις που αναφέρονται στις προδιαγραφές, τον αλγόριθμο, και το flow chart. Η επόμενη φάση της κατάρτισης του προγράμματος θα καλυφθεί στο επόμενο κεφάλαιο με τη βοήθεια της γλώσσας BASIC τις βασικές εντολές της οποίας θα παρουσιάσουμε.

B ΣΤΟΙΧΕΙΑ ΤΗΣ ΓΛΩΣΣΑΣ BASIC

Μια από τις πρώτες και πιό εύχρηστες γλώσσες προγραμματισμού ήταν η BASIC που παρόλα τα χρόνια της και τις σημαντικές εξελίξεις στο χώρο των υπολογιστών εξακολουθεί να είναι πολύ δημοφιλής.

Στην παρουσίαση που ακολουθεί θα δούμε μερικές από τις πιό βασικές εντολές της γλώσσας ώστε να μπορέσουμε να συντάξουμε το πρόγραμμα για την εφαρμογή της απλής μισθοδοσίας του παραδείγματός μας και να καταδείξουμε μέσα από το παράδειγμα αυτό τον τρόπο προγραμματισμού

Γενικά

Οι γλώσσες προγραμματισμού Υψηλού Επιπέδου (ΥΕ), όπως είπαμε και στο προηγούμενο κεφάλαιο, τείνουν να προσεγγίσουν όσο γίνεται περισσότερο τις φυσικές γλώσσες του ανθρώπου. Στόχος ιδιαίτερα δύσκολος (και για πολλούς επιστήμονες ανέφικτος) γιατί θέτει το γενικότερο φιλοσοφικό θέμα των ορίων της τεχνητής νοημοσύνης.

Οι γλώσσες ΥΕ και οι φυσικές γλώσσες έχουν πολλά κοινά σημεία αλλά και μία βασική διαφορά. Τα κοινά τους σημεία είναι οτι οι γλώσσες ΥΕ όπως ακριβώς και οι φυσικές γλώσσες χρησιμοποιούνται για να γράφονται κείμενα. Τα κείμενα αυτά στην περίπτωση των γλωσσών Υ είναι τα προγράμματα. Τα προγράμματα αυτά όπως όλα τα κείμενα αποτελούνται από προτάσεις, οι προτάσεις από λέξεις και οι λέξεις από χαρακτήρες. Αυτό σημαίνει πως οι γλώσσες ΥΕ θα πρέπει να έχουν (όπως και οι φυσικές) το αλφάβητό τους (σύνολο αποδεκτών χαρακτήρων αλφαβητικών, αριθμητικών, σημείων στίξης κλπ.), τη γραμματική τους (σύνολο κανόνων

σχηματισμού σωστών λέξεων) και το συντακτικό τους (σύνολο κανόνων σχηματισμού σωστών προτάσεων).

Η διαφορά τους είναι ότι οι φυσικές γλώσσες χαρακτηρίζονται από την ελευθερία της έκφρασης τον πλούτο των λέξεων και των κανόνων τους, των παραλλαγών και των εξαιρέσεων τους. Η γραμματολογική και συντακτική ορθότης είναι ικανή αλλά όχι και αναγκαία προυπόθεση για την κατανόηση ενός κειμένου από ένα άνθρωπο. Ο άνθρωπος δηλαδή λόγω της ευελιξίας της ανθρώπινης νοημοσύνης μπορεί να καταλάβει ένα κείμενο και άν ακόμα δεν είναι απόλυτα συντακτικά ή γραμματικά σωστό. Αντίθετα στις γλώσσες ΥΕ οι κανόνες είναι σαφείς και αυστηροί και η τήρησή τους στη σύνταξη ενός κειμένου (προγράμματος) απαραίτητη προυπόθεση για να γίνει κατανοητό από ένα Υ λόγω του (ακόμα) περιοριστικού χαρακτήρα της τεχνητής νοημοσύνης. Γλώσσες με τα χαρακτηριστικά αυτά λέγονται και φορμαλιστικές γλώσσες λόγω της σημασίας που δίνουν στη φόρμα και την τήρηση των κανόνων.

Βασικά Χαρακτηριστικά μιάς γλώσσας Υψηλού Επιπέδου
Η Γλώσσα BASIC

Σύμφωνα με το πνεύμα της παραπάνω παραγράφου μπορούμε να πούμε πως μία γλώσσα ΥΕ (όπως και μία φυσική γλώσσα) αποτελείται από χαρακτήρες, λέξεις και προτάσεις και προσδιορίζεται από το αλφάβητο, τη γραμματική και το συντακτικό της. Στις παραγράφους που ακολουθούν θα παρουσιάσουμε τα βασικά αυτά χαρακτηριστικά των γλωσσών προγραμματισμού ενώ παράλληλα θα τα δούμε να εφαρμόζονται στην πολύ γνωστή γλώσσα BASIC.

Με τη βοήθεια της BASIC στη συνέχεια, θα προγραμματίσουμε την απλή εφαρμογή της μισθοδοσίας που προδιαγράψαμε στο Παραρτ. Α.

Αλφάβητο της Γλώσσας - Χαρακτήρες

Αλφάβητο είναι το σύνολο των αποδεκτών στη γλώσσα αυτή χαρακτήρων. Αποτελείται από τους χαρακτήρες που αναγνωρίζει η γλώσσα και με την βοήθεια των οποίων σχηματίζει τις λέξεις της. Τέτοιοι χαρακτήρες είναι :

- αλφαβητικοί χαρακτήρες συνήθως τα κεφαλαία κα πεζά γράμματα του λατινικού αλφαβήτου
- αριθμητικοί χαρακτήρες τα δέκα αριθμητικά ψηφία 0-9

- ειδικοί χαρακτήρες όπως:
  σημεία πράξεων + - * /
  σημεία στίξης . ,
  ειδικά σύμβολα = ( ) $ κλπ

Για την παράσταση κάθε χαρακτήρα χρησιμοποιείται ένα byte, πράγμα που σημαίνει πως με τη βοήθεια του byte μπορούμε να παραστήσουμε ένα σύνολο 256 χαρακτήτων, ένα αρκετά δηλ μεγάλο αλφάβητο που μπορεί να καλύψει άνετα και τις πιό απαιτητικές ανάγκες. Ο τρόπος παράστασης των χαρακτήρων αυτών υπακούει σε διάφορα στάνταρ κωδικοποίησης.

Γραμματική της Γλώσσας - Λέξεις

Γραμματική είναι το σύνολο των κανόνων σχηματισμού σωστών λέξεων. Μιά λέξη σε μιά γλώσσα προγραμματισμού δεν είναι τίποτα άλλο παρά ένα σύνολο ενός ή περισσοτέρων χαρακτήρων που ανήκουν στο αλφάβητο της γλώσσας. Ο τρόπος συνδυασμού των χαρακτήρων αυτών για να αποτελέσουν μιά λέξη που να έχει νόημα στην υπόψη γλώσσα προσδιορίζεται από την γραμματική της γλώσσας. Σαν λέξεις σε μια γλώσσα Υψηλού Επιπέδου μπορεί να θεωρηθούν οι λεγόμενες σταθερές, οι μεταβλητές και οι αριθμητικές εκφράσεις. Οι δύο πρώτες μπορούν να χαρακτηρισθούν σαν απλές λέξεις ενώ οι τελευταίες σαν σύνθετες λέξεις.

Σταθερές (Constants)

(α) Είναι οι αριθμοί που εμφανίζονται αυτούσιοι στο πρόγραμμα πχ. 15, 175.50 κλπ. Διακρίνονται σε ακέραιες (INTEGER) και δεκαδικές (με υποδιαστολή, REAL).
(β) Εκτός από αριθμούς μπορούμε να χρησιμοποιήσουμε και αλφαβητικούς χαρακτήρες αυτούσιους μέσα στο πρόγραμμα. Στην περίπτωση αυτή για να μην μπερδεύεται ο Υ τοποθετούμε τους χαρακτήρες αυτούς μεταξύ εισαγωγικών πχ. *ΑΒ*, *ΠΑΝΟΣ* κλπ

Μεταβλητές (Variables)

Είναι συμβολικά ονόματα που δίνουμε σε θέσεις μνήμης τις οποίες χρησιμοποιούμε για να τοποθετήσουμε συγκεκριμμένα δεδομένα (πχ. τον βασικό μισθό ενός υπάλληλου σε ένα πρόγραμμα μισθοδοσίας). Με τον

τρόπο αυτό μπορούμε να αναφερόμαστε στις θέσεις αυτές με το συμβολικό τους όνομα χωρίς να χρειάζεται να γνωρίζουμε την πραγματική τους διεύθυνση μέσα στη μνήμη.

Οι κανόνες δημιουργίας των ονομάτων αυτών ποικίλλει από γλώσσα σε γλώσσα. Στη FORTRAN για παράδειγμα ένα τέτοιο όνομα αποτελείται από 1-6 αλφαριθμητικούς (αλφαβητικούς ή αριθμητικούς) χαρακτήρες με τον περιορισμό ο πρώτος χαρακτήρας να είναι πάντα αλφαβητικός πχ. A, OLIKO, H2SO4, PANOS είναι ονόματα αποδεκτά στη FORTRAN.

Στη BASIC επιφυλάσσεται ειδική μεταχείριση για τα ονόματα θέσεων μνήμης που προορίζονται να περιέχουν αλφαβητικούς χαρακτήρες (πχ. ονοματεπώνυμα). Συγκεκριμένα απαιτεί να τοποθετείται ένα σημάδι (το σύμβολο $) μετά το κανονικό όνομα για να ξεχωρίζουν από τα ονόματα των θέσεων μνήμης που περιέχουν μόνο αριθμούς.
Για παράδειγμα οι μεταβλητές A$, ONOMA$ δείχνουν αμέσως πως οι αντίστοιχες θέσεις μνήμης τους θα περιέχουν και αλφαβητικούς χαρακτήρες. Αντίθετα οι μεταβλητές A, B, PANOS, OLIKO κλπ θα περιέχουν μόνο αριθμούς.

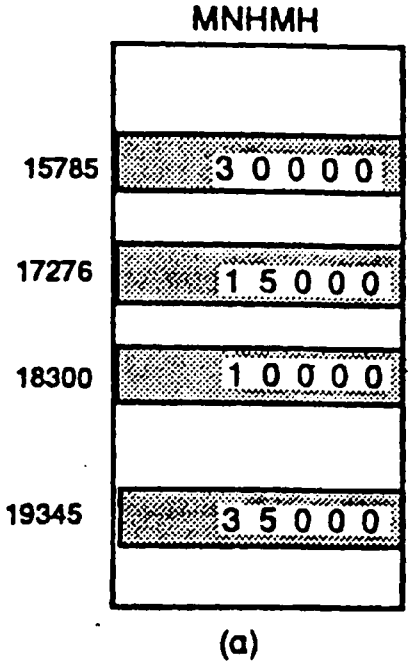
Η δυνατότητα να δίνουμε ονόματα στις θέσεις της μνήμης είναι ένα ιδιαίτερης σημασίας κοινό χαρακτηριστικό των γλωσσών Υψηλού Επιπέδου. Επειδή θεωρούμε πως η κατανόηση της δυνατότητας αυτής έχει μεγάλη σημασία για τον προγραμματισμό σε οποιαδήποτε γλώσσα, θα αναφερθούμε στην έννοια της μεταβλητής με κάπως μεγαλύτερη λεπτομέρεια.

Η Ονοματοποίηση Θέσεων Μνήμης

Ας δούμε τώρα από κάπως πιό κοντά τη σχέση μεταξύ ονόματος και θέσης μνήμης. Οπως είπαμε και στο πρώτο μέρος του βιβλίου κάθε λέξη-θέση μνήμης αναγνωρίζεται από τη διεύθυνσή της μέσα στη Μνήμη πχ λέμε η λέξη (ή το byte) 15785 της μνήμης. Προκειμένου τώρα να τοποθετήσουμε ένα στοιχείο πχ. τον βασικό μισθό στο παράδειγμα της μισθοδοσίας μας (έστω 30000) στη μνήμη του Υ θα πρέπει να ψάξουμε στη μνήμη και όταν βρούμε μιά θέση κενή (έστω τη 15785) το τοποθετούμε. Στη συνέχεια όταν θέλουμε να χρησιμοποιήσουμε το στοιχείο αυτό για υπολογισμούς (στο πρόγραμμά μας) θα πρέπει να θυμόμαστε πως το έχουμε βάλει στη θέση 15785 για να το αναζητήσουμε. Κάτι τέτοιο βέβαια θα δυσχέρανε πολύ όπως είναι φυσικό τη διαδικασία του προγραμματισμού. Ηλθαν λοιπόν οι γλώσσες προγραμματισμού να απλοποιήσουν τα πράγματα.

Σε μια γλώσσα προγραμματισμού λοιπόν μπορούμε να πούμε το εξής: θέλω να βάλω ένα στοιχείο (εστω το 30000) στη μνήμη. Λέω στον Υ: βρές μιά θέση μνήμης κενή (έστω πως βρίσκει την 15785) τοποθέτησε το 30000 μέσα και βάφτισε τη θέση (τη 15785) με ένα όνομα (έστω BASMISQOS). Από κεί και πέρα όταν θέλω να αναφερθώ στο 30000 δεν χρειάζεται να αναζητήσω τη διεύθυνση 15785 (που άλλωστε σαν προγραμματιστής δεν γνωρίζω) αλλά να χρησιμοποιήσω το όνομα της (που το έχω δώσει εγώ) δηλ το BASMISQOS (βλ Σχ. Β.1(α)).

Εστω τώρα οτι εκτός από το 30000 που έχω τοποθετήσει ήδη στη μνήμη σε μιά θέση με το όνομα BASMISQOS τοποθετώ και ένα άλλον αριθμό πχ τον 15000 σε μιά άλλη κενή θέση της μνήμης στην οποία δίνω ένα άλλο όνομα πχ EPIDOMA. Με τον ίδιο τρόπο μπορώ να τοποθετήσω και ένα τρίτο αριθμό πχ, τον 10000 σε μιά τρίτη θέση μνήμης στην οποία δίνω ένα άλλο όνομα έστω KRATHSEIS. Αν θέλω να βρώ τώρα το άθροισμα των δύο πρώτων αριθμών και μετά από το άθροισμα να αφαιρέσω τον τρίτο δεν έχω παρά να χρησιμοποιήσω τα ονόματά τους BASMISQOS + EPIDOMA - KRATHSEIS. Με την παράσταση αυτή ο Υ καταλαβαίνει πως για να κάνει τις ενδεικνυόμενες πράξεις πρέπει να πάρει τα περιεχόμενα των αντίστοιχων θέσεων μνήμης. Βλέπουμε δηλ πως με τις γλώσσες Υψηλού Επιπέδου αποδεσμευόμαστε από τις ενοχλητικές λεπτομέρειες της μηχανής, θέσεις μνήμης κλπ και δουλεύουμε με τα συμβολικά ονόματα που εμείς οι ίδιοι έχουμε δώσει στις θέσεις μνήμης.

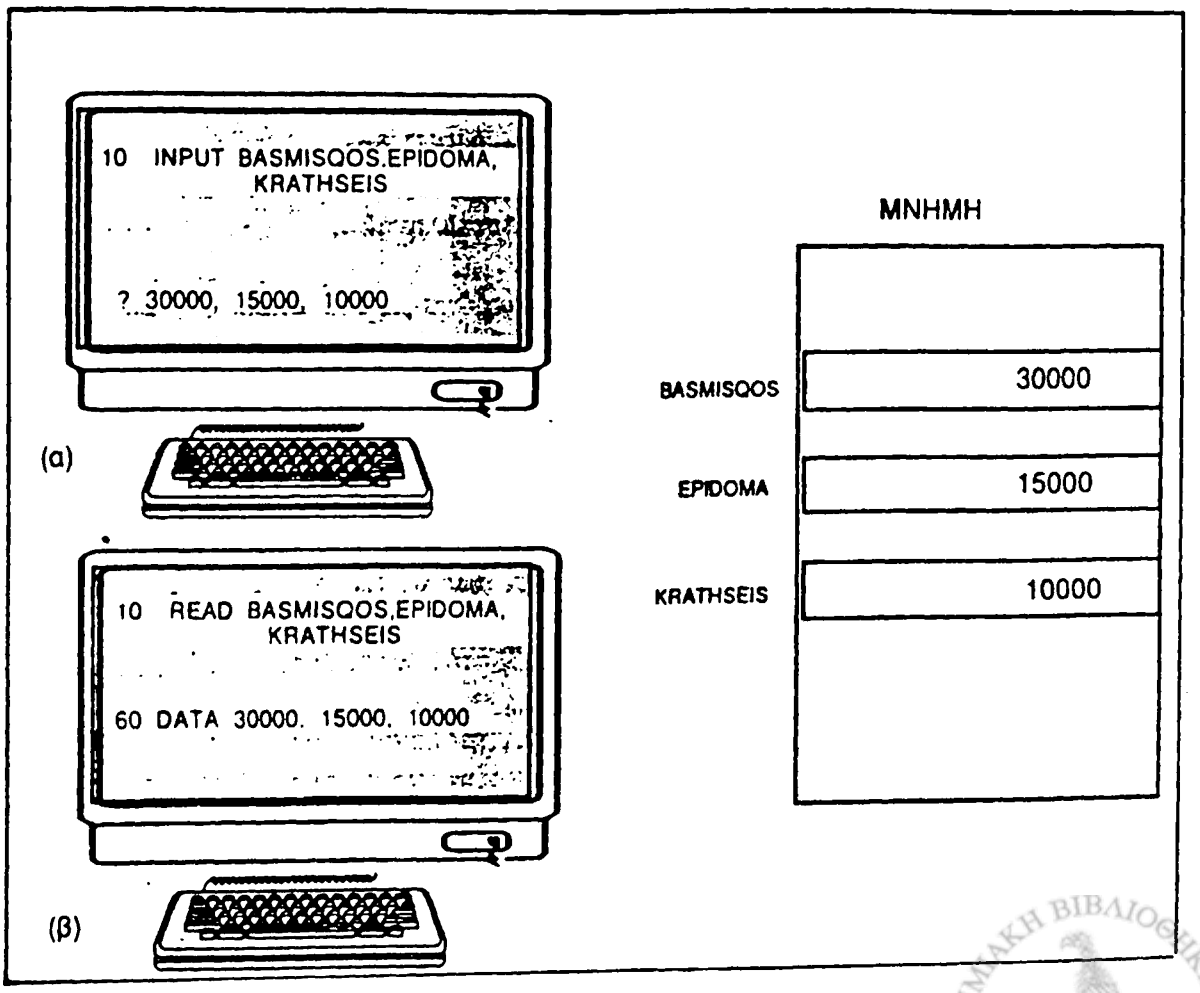


ΠΙΝΑΚΑΣ ΟΝΟΜΑΤΟΠΟΙΗΣΗΣ

ΟΝΟΜΑ ΜΕΤΑΒΛΗΤΗΣ	ΘΕΣΗ ΜΝΗΜΗΣ
BASMISQOS	15785
EPIDOMA	17276
KRATHSEIS	18300
PLHRWTEO	19345
...	...

(β)

Σχ. Β.1



Σχ. Β.2

Πώς τώρα γίνεται στην πράξη αυτή η αντιστοίχηση ώστε να ξέρει ο Υ τα ονόματα που χρησιμοποιούμε και να μας φέρνει τα σωστά στοιχεία από τις σωστές διευθύνσεις; Ο τρόπος που δουλεύει η ονοματοθεσία σε γενικές γραμμές είναι η εξής: οταν ζητάμε να τοποθετηθεί ένα στοιχείο στη μνήμη με το όνομα έστω BASMISQOS το σύστημα ψάχνει τη μνήμη βρίσκει μιά θέση κενή έστω 15785 τοποθετεί το στοιχείο και μετά γράφει σε πίνακα (που κρατάει πάλι μέσα στη μνήμη) οτι η διέυθυνση 15785 έχει ονομαστεί BASMISQOS (βλ. Σχ.Β.1(β)). Το ίδιο κάνει και για τα άλλα στοιχεία που είδαμε παραπάνω. Οταν τώρα ο Υ δεί στο πρόγραμμά μας πως πρέπει να κάνει τον υπολογισμό BASMISQOS + EPIDOMA - KRATHSEIS παίρνει πρώτα το BASMISQOS, βρίσκει από τον πίνακα σε ποιά διεύθυνση αντιστοιχεί αυτό το όνομα (15785) και από εκεί παίρνει το περιεχόμενο (30000). Το ίδιο κάνει και για το EPIDOMA και τις KRATHSEIS έτσι υπολογίζει το 30000 + 15000 - 10000 και βρίσκει 35000.

Τα δεδομένα που βάλαμε παραπάνω στις θέσεις μνήμης με τα ονόματα BASMISQOS, EPIDOMA, KRATHSEIS ανήκουν στον βασικό μισθό, επίδομα και κρατήσεις ενός υπαλλήλου του οποίου έστω ζητούμε το καθαρό πληρωτέο. Οταν έλθει η σειρά άλλου υπαλλήλου μπορούμε να βάλουμε τα νέα στοιχεία του υπαλλήλου αυτού στις ίδιες θέσεις μνήμης. Βλέπουμε λοιπόν πως τα περιεχόμενα των θέσεων αυτών στη διάρκεια μιάς επεξεργασίας αλλάζουν, για το λόγο αυτό οι θέσεις αυτες και κατεπέκταση τα ονόματά τους λέγονται μεταβλητές.

Αριθμητικές εκφράσεις (Arithmetic Expressions)

Είναι σύνθετες λέξεις που αποτελούνται από μία ή περισσότερες μεταβλητές και σταθερές συνδεδεμένες μεταξύ τους με τα σημεία των πράξεων + ,- ,* , / και άλλους αριθμητικούς τελεστές. Είναι το αντίστοιχο μιάς αλγεβρικής παράστασης που υποδηλώνει μιά σειρά αριθμητικών υπολογισμών σε αριθμητικά στοιχεία που είτε έχουν τοποθετηθεί στη μνήμη (μεταβλητές) είτε είναι σταθερές. Το αποτέλεσμα των πράξεων αυτών (τιμή της αριθμητικής έκφρασης) είναι ένας αριθμός που εξαρτάται από τα εκάστοτε περιεχόμενα των μεταβλητών που υπεισέρχονται στην αριθμητική έκφραση. Για παράδειγμα η έκφραση BASMISQOS + EPIDOMA - KRATHSEIS που είδαμε παραπάνω είναι μια τέτοια σύνθετη λέξη και η τιμή της είναι 35000 όταν τα περιεχόμενα των μεταβλητών της είναι 30000, 15000 και 10000 αντίστοιχα.

Ιδιωτικό Λεξιλόγιο της Γλώσσας

Εκτός από τα παραπάνω είδη λέξεων των οποίων τα ονόματα καθορίζονται από τον ίδιο τον προγραμματιστή μέσα στο πρόγραμμά του, κάθε γλώσσα ΥΕ έχει ένα λεξιλόγιο-ρεπερτόριο συγκεκριμένων λέξεων (reserved words) που χρησιμοποιούνται για τις ανάγκες της ίδιας της γλώσσας. Τέτοιες λέξεις μπορεί να είναι ρήματα πχ READ (ΔΙΑΒΑΣΕ), PRINT(ΤΥΠΩΣΕ), DO (ΕΚΤΕΛΕΣΕ), COMPUTE (ΥΠΟΛΟΓΙΣΕ) κλπ. Ακόμα υποθετικά μόρια πχ IF (ΕΑΝ), WHEN (ΟΤΑΝ) κλπ. Οι λέξεις αυτές αποτελούν συστατικά στοιχεία των προτάσεων-εντολών της γλώσσας που θα δούμε παρακάτω. Τις λέξεις αυτές απαγορεύεται να τις χρησιμοποιεί ο χρήστης σαν ονόματα μεταβλητών.

Συντακτικό της Γλώσσας - Προτάσεις, Εντολές

Μια αλληλουχία λέξεων σε μιά οποιαδήποτε γλώσσα μας δίνει προτάσεις στη γλώσσα αυτή. Ο σχηματισμός σωστών προτάσεων καθορίζεται από ορισμένους βασικούς κανόνες που απαρτίζουν το συντακτικό της γλώσσας. Στις γλώσσες ΥΕ το συντακτικό είναι ιδιαίτερα περιοριστικό και οι κανόνες αυστηροί. Οι προτάσεις στις γλώσσες ΥΕ που λέγονται συνήθως εντολές της γλώσσας ή statements πρέπει να συμμορφώνονται απόλυτα με τους συντακτικούς κανόνες διαφορετικά δεν αναγνωρίζονται.
Τα είδη των προτάσεων-εντολών στις γλώσσες ΥΕ ποικίλλουν από γλώσσα σε γλώσσα ανάλογα με τον πλούτο-νοημοσύνη της κάθε γλώσσας και των υπηρεσιών που θέλει να παρέχει στους χρήστες της. Ετσι υπάρχουν γλώσσες με εντολές που διευκολύνουν κυρίως τις επιστημονικές επεξεργασίες και υπολογισμούς (FORTRAN, BASIC), τις εμπορικές εφαρμογές και τις επεξεργασίες με αρχεία (COBOL) κλπ.
Κάποια είδη όμως εντολών είναι γενικής εφαρμογής και απαντώνται σε όλες σχεδόν τις γλώσσες ΥΕ. Οι εντολές αυτές ανήκουν στις ακόλουθες βασικές κατηγορίες :

- **εντολές Εισόδου/ Εξόδου**, για την εισαγωγή δεδομένων στη μνήμη και την παρουσίαση αποτελεσμάτων στα μηχανήματα εξόδου

- **εντολές Ροής Εκτέλεσης** προγράμματος, που καθορίζουν οποιαδήποτε αλλαγή από τη φυσιολογική σειριακή ροή του προγράμματος

- **εντολές Επεξεργασίας**, που μπορεί να είναι
    - *εντολές Υπολογισμών*,
    - *εντολές Αποφάσεων* με τη βοήθεια των οποίων ο Υ παίρνει "αποφάσεις".
    - *εντολές Επανάληψης* με τη βοήθεια των οποίων ο προγραμματιστής ζητάει από τον Υ να εκτελέσει μία συνεχή σειρά εντολών πολλές φορές. Η σειρά αυτή των εντολών των οποίων η εκτέλεση πρέπει να επαναληφθεί λέγεται loop.

    Πιο κάτω αναφέρονται οι κυριώτερες εντολές της Quick Basic.

Λίστα Εντολών

ABS Συνάρτηση
ASC Συνάρτηση
BEEP Εντολή
CALL Εντολή
CHAIN Εντολή
CHR$ Συνάρτηση
CLOSE Εντολή
CLS Εντολή
COLOR Γραφική εντολή.
DATE$ Συνάρτηση
DECLARE Εντολή
DIM Εντολή
DO ... LOOP Εντολή
END Εντολή
EOF Συνάρτηση
ERASE Εντολή
EXIT Εντολή
EXP Συνάρτηση
FIELD Εντολή
FIX Συνάρτηση
FOR NEXT Εντολή
FUNCTION Εντολή
GET Εντολή διαχείρισης αρχείων
GOSUB...RETURN Εντολές
GOTO Εντολή
IF..THEN..ELSE Εντολή
INPUT Εντολή
INPUT# Εντολή
INPUT$ Συνάρτηση
INSTR Συνάρτηση
INT Συνάρτηση
KILL Εντολή
LEFT$ Συνάρτηση
LEN Συνάρτηση
LINE Εντολή γραφικών
LINE INPUT Εντολή
LINE INPUT# Εντολή
LOCATE Εντολή
LPRINT και LPRINT USING Εντολές
MID Συνάρτηση και Εντολή
MKDIR Εντολή
NAME Εντολή
OPEN Εντολή
PRINT Εντολή
PRINT USING Εντολή
PRINT# και PRINT# USING Εντολές
PUT Εντολή διαχείρισης αρχείων
RANDOMIZE Εντολή
REDIM Εντολή
RESET Εντολή
RESUME Εντολή
RETURN Εντολή
RIGHT$ Συνάρτη
RND Εντολή
RUN Εντολή
SEEK Εντολή
SEEK Συνάρτηση
SELECT CASE E
SHELL Εντολή
SLEEP Εντολή
SOUND Εντολή
SPACE$ Συνάρτη
SPC Συνάρτηση
SQR Συνάρτηση
STOP Εντολή
SWAP Εντολή
SYSTEM Εντολή
TAB Συνάρτηση
TIME$ Συνάρτησ
TYPE Δηλωτική ε
VAL Συνάρτηση
WAIT Εντολή
WHILE..WEND E
WRITE Εντολή

Βασικές Εντολές της BASIC

Α. Εντολές Εισόδου-Εξόδου (I/O Statements)

Εισαγωγή Στοιχείων στη Μνήμη του Υπολογιστή

(α) Εντολή INPUT

Πώς ζητάμε μέσω του προγράμματος από το σύστημα να εισαγάγει δεδομένα στη μνήμη και πως καθορίζουμε πού θα πρέπει να τα τοποθετήσει μέσα στη μνήμη;
Η πιό εύχρηστη εντολή για την εισαγωγή στοιχείων στη μνήμη είναι η εντολή

INPUT Μεταβλητή1, Μεταβλητή2,...

Η εντολή αυτή έχει σαν αποτέλεσμα να περιμένει ο Υ να του δώσουμε κάποια στοιχεία (πχ με πληκτρολόγηση) τα οποία τοποθετεί με τη σειρά που του τα δίνουμε αντίστοιχα στις θέσεις μνήμης με τα ονόματα Μεταβλητή1, Μεταβλητή2,..

Για να το καταλάβουμε ας χρησιμοποιήσουμε ένα παράδειγμα:
Εστω πως θέλουμε να βάλουμε στη μνήμη τα δεδομένα της μισθοδοσίας ενός υπαλλήλου δηλ τον βασικό μισθό, το επίδομα και τις κρατήσεις. Και ας πούμε πως το πρώτο θέλουμε να το τοποθετήσουμε σε μία θέση μνήμης με το όνομα BASMISQOS, το δεύτερο στη θέση με το όνομα EPIDOMA και το τρίτο στη θέση με το όνομα KRATHSEIS. Τότε γράφουμε την εντολή:

INPUT BASMISQOS, EPIDOMA, KRATHSEIS

Τί γίνεται τότε στον Υπολογιστή;
Οταν βάλουμε το πρόγραμμα στη μνήμη και ο Υ αρχίζει να εκτελεί τις εντολές μία-μία όταν φτάσει στην εντολή αυτή σταματάει και εμφανίζει ενα ερωτηματικό ή άλλο ειδικό σύμβολο στην οθόνη για να μας ειδοποιήσει πως περιμένει κάποια δεδομένα ώστε να μπορέσει να προχωρήσει στην εκτέλεση των υπόλοιπων εντολών του προγράμματος. Τότε εμείς πληκτρολογούμε με τη σειρά τα δεδομένα (δηλ τους αριθμούς) βασικό μισθό, το επίδομα και τις κρατήσεις ακριβώς δηλ με τη σειρά που εμφανίζονται στην εντολή INPUT. Τα δεδομένα ξεχωρίζουν το ένα από το άλλο με κόμματα. Αν για παράδειγμα για ένα υπάλληλο έχουμε 30000 βασικό μισθό, 15000 επίδομα και 10000 κρατήσεις γράφουμε στην οθόνη

μετά το ερωτηματικό: 30000, 15000, 10000. Τότε ο Υ παίρνει τον πρώτο αριθμό (30000) και τον τοποθετεί σε μιά κενή θέση της μνήμης που την ονομάζει BASMISQOS, στη συνέχεια παίρνει τον δεύτερο (15000) και τον τοποθετεί σε μιά άλλη θέση με το όνομα EPIDOMA και τον τρίτο (10000) και τον τοποθετεί σε μιά άλλη θέση με το όνομα KRATHSEIS (βλ Σχ.Β.2α). Αφού ολοκληρωθεί αυτή η διαδικασία το πρόγραμμα προχωρεί στην εκτέλεση της αμέσως επόμενης εντολής.

Οπως είδαμε παραπάνω η εντολή INPUT απαιτεί από τον χρήστη να δώσει τα δεδομένα κατά τη διάρκεια της εκτέλεσης του προγράμματος και όταν του τα ζητήσει ο Υ. Αυτό σημαίνει πως ο Υπολογιστής θα περιμένει τον χρήστη να πληκτρολογήσει τα δεδομένα για να προχωρήσει.

(β) Εντολές READ/DATA

Για να αποφεύγονται τέτοιου είδους καθυστερήσεις μπορεί ο χρήστης να έχει έτοιμα τα δεδομένα από πρίν έτσι ώστε να μπορεί ο Υ να τα πάρει όταν τα χρειάζεται από εκεί που τα έχει τοποθετήσει ο χρήστης. Πού μπορεί να τα τοποθετήσει τώρα ο χρήστης. Προφανώς σε ένα αρχείο δεδομένων που θα υπάρχει στον δίσκο. Ενημερώνει δε σχετικά το πρόγραμμα δίνοντας το όνομα του αρχείου αυτού για να γνωρίζει ο Υ όταν εκτελεί το πρόγραμμα πού να το βρεί.
Οταν τα δεδομένα είναι λίγα ο χρήστης μπορεί να τα τοποθετήσει στο ίδιο το πρόγραμμα ειδοποιώντας παράλληλα τον Υ πού θα τα βρεί όταν τα χρειαστεί. Για την περίπτωση αυτή η BASIC διαθέτει το ζεύγος των εντολών
READ
DATA
Η εντολή READ έχει την ίδια ακριβώς μορφή με την INPUT δηλ

READ Μεταβλητή1,Μεταβλητή2,..

και με την ίδια σημασία. Η διαφορά έγκειται στον τρόπο με τον οποίο ο Υ βρίσκει τα δεδομένα για να τα βάλει στις προκαθορισμένες θέσεις μνήμης Μεταβλητή1, Μεταβλητή2,..
Η εντολή DATA χρησιμοποιείται για να γράψουμε τα δεδομένα μας και έχει τη μορφή

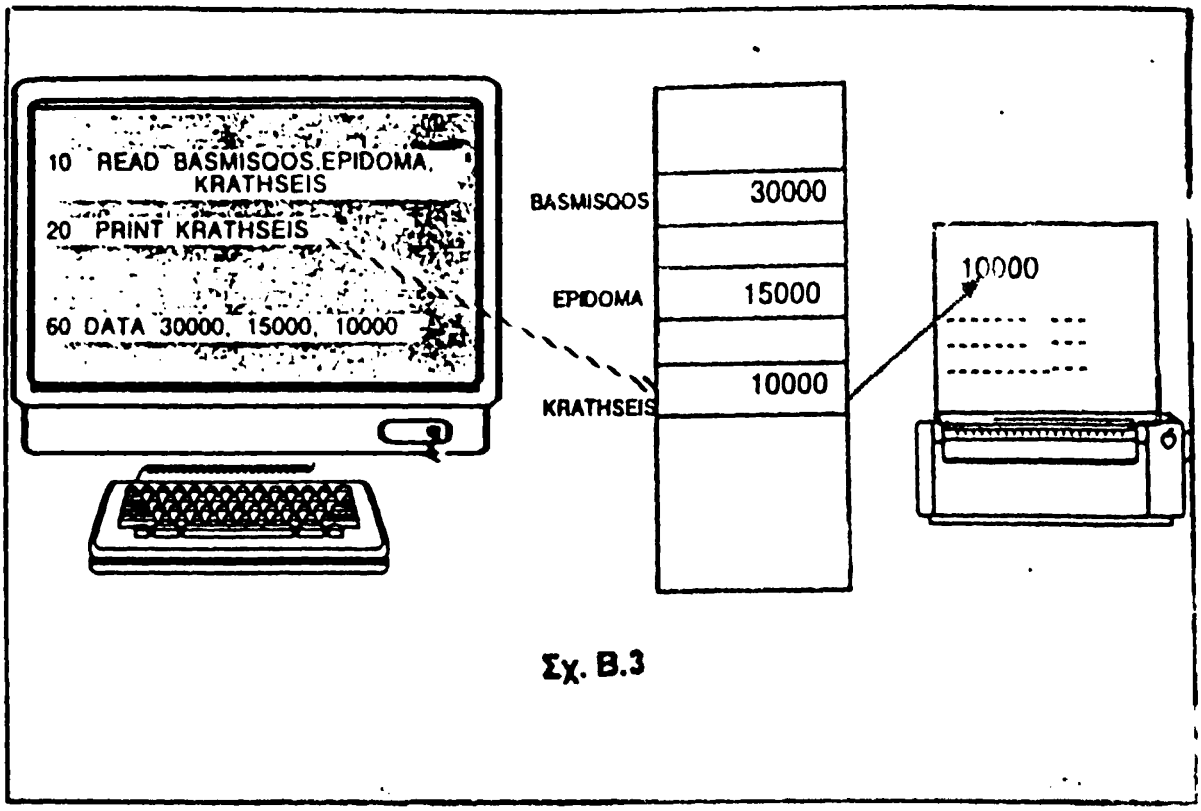
DATA Δεδομένο1,Δεδομένο2,...

Ας δούμε τώρα πώς λειτουργεί το ζεύγος αυτό των εντολών.

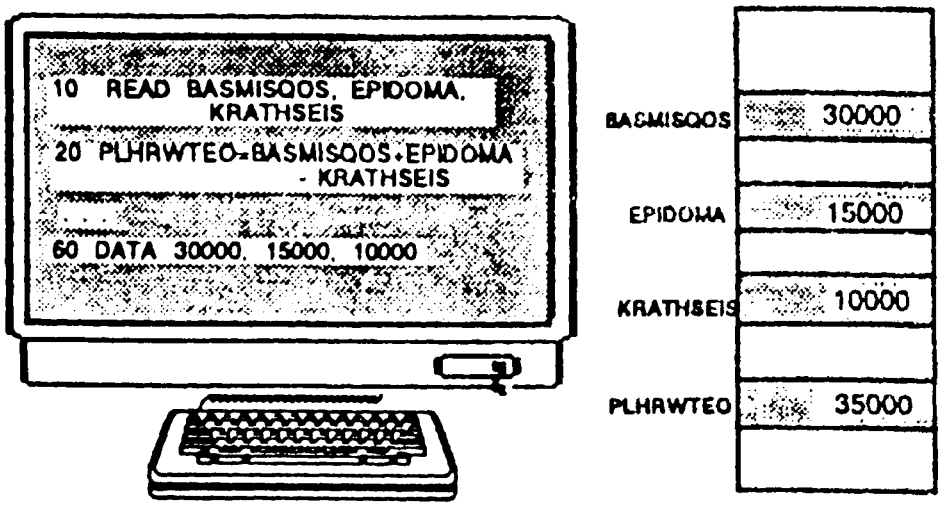
Οταν ο Υ εκτελεί την εντολή READ καταλαβαίνει πως θα πρέπει να ψάξει μέσα στο πρόγραμμα για την αντίστοιχη εντολή DATA όπου και θα βρεί τα δεδομένα για να τα τοποθετήσει κατά σειρά στις θέσεις μνήμης Μεταβλητή1, Μεταβλητή2,...(δηλ το Δεδομένο1 στη Μεταβλητή1 κλπ)
Εδώ λοιπόν δεν περιμένει ο Υ τον χρήστη να πληκτρολογήσει τα δεδομένα αλλά αυτόματα ψάχνει και τα βρίσκει στην εντολή DATA του προγράμματος. Θα πρέπει να σημειώσουμε εδώ πως η εντολή DATA είναι μή εκτελεστή εντολή δηλ δεν εκτελείται όταν έλθει η σειρά της όπως οι άλλες εντολές από τον Υ, είναι απλά βοηθητική εντολή που ενημερώνει τον Υ πού υπάρχουν τα δεδομένα που έχει ζητήσει να διαβάσει με την εντολή READ. Αυτό σημαίνει πως μπορεί να βρίσκεται οπουδήποτε μέσα στο πρόγραμμα χωρίς να χαλάει τον *ειρμό* δηλ τη συνέχεια του προγράμματος. Με άλλα λόγια, όταν ο Υ συναντάει την εντολή DATA μέσα στο πρόγραμμα κατά την σειριακή εκτέλεση των εντολών, την προσπερνάει χωρίς να την εκτελέσει. Η εντολή δηλαδή αυτή υπάρχει πάντα σε συνδυασμό με τη READ και απλά αναζητείται όταν πρόκειται να εκτελεσθεί μία εντολή READ. Συνήθως οι εντολές DATA με τα δεδομένα τοποθετούνται είτε στην αρχή είτε στο τέλος του προγράμματος.

Στο παράδειγμά μας δηλ θα είχαμε (βλ. Σχ. Β.2β) :

```
READ BASMISQOS, EPIDOMA, KRATHSEIS
.
.
DATA 30000,15000,10000
```


10 READ BASMISOOS,EPIDOMA,
KRATHSEIS
20 PRINT KRATHSEIS
60 DATA 30000, 15000, 10000
BASMISOOS
30000
EPIDOMA
15000
KRATHSEIS
10000
10000

Σχ. Β.3


10 READ BASMISOOS, EPIDOMA,
KRATHSEIS
20 PLHRWTEO=BASMISOOS+EPIDOMA
- KRATHSEIS
60 DATA 30000, 15000, 10000
BASMISOOS
30000
EPIDOMA
15000
KRATHSEIS
10000
PLHRWTEO
35000

Σχ. Β.4

Εξοδος Στοιχείων

Η πιό συνηθισμένη εντολή εξόδου είναι η εντολή PRINT με τη μορφή

PRINT Μεταβλητή1, Μεταβλητή2,..

Με την εντολή αυτή ο χρήστης ζητάει από τον Υ να εμφανίσει στην οθόνη με τη σειρά το περιεχόμενο των θέσεων μνήμης με τα ονόματα Μεταβλητή1,Μεταβλητή2,..
Για παράδειγμα η εντολή PRINT KRATHSEIS εκτελούμενη μετά τις παραπάνω εντολές εισόδου θα είχε σαν αποτέλεσμα την εμφάνιση στην οθόνη του περιεχομένου της θέσης μνήμης KRATHSEIS δηλ. το 10000 (βλ. Σχ. Β.3).
Αν τώρα θέλουμε να τυπώσουμε ένα μήνυμα (κείμενο) αυτούσιο τοποθετούμε το μήνυμα μέσα σε εισαγωγικά μετά το PRINT πχ. η εντολή PRINT *ΜΙΣΘΟΔΟΤΙΚΗ ΚΑΤΑΣΤΑΣΗ* προκαλεί την εκτύπωση του μηνύματος ΜΙΣΘΟΔΟΤΙΚΗ ΚΑΤΑΣΤΑΣΗ στην οθόνη όταν εκτελεσθεί από τον Υ.

Β. Εντολές Για Επεξεργασία

Εντολή Υπολογισμού ή Τοποθέτησης- Assignment Statement

Βάλαμε λοιπό τα δεδομένα μας μέσα στη μνήμη στις θέσεις που προσδιορίσαμε. Τώρα θέλουμε βέβαια να κάνουμε υπολογισμούς με τα δεδομένα αυτά. Κάθε υπολογισμός εκφράζεται με την κατάλληλη αριθμητική έκφραση με την βοήθεια των ονομάτων των αντίστοιχων μεταβλητών.
Στην περίπτωση του παραδείγματός μας θέλουμε να υπολογίσουμε τη σχέση: BASMISQOS+EPIDOMA-KRATHSEIS. Ο Υ για να εκτελέσει τον υπολογισμό αυτό θα ανατρέξει στις αντίστοιχες θέσεις μνήμης και θα πάρει τα περιεχόμενά τους με τα οποία θα κάνει τις σημειούμενες πράξεις και θα βγάλει ένα αποτέλεσμα. Το αποτέλεσμα αυτό θα πρέπει φυσικά να το τοποθετήσει σε μιά άλλη θέση μνήμης για να μπορέσει να το ξαναχρησιμοποιήσει όταν το χρειαστεί. Η τοποθέτηση αυτή του αποτελέσματος ενός υπολογισμού σε μία θέση μνήμης γίνεται με τη βοήθεια της εντολής της τοποθέτησης ή assignment statement. Το βασικό σύμβολο της εντολής είναι το = και η μορφή της είναι:

Μεταβλητή = Αριθμητική Εκφραση

Ο Υ υπολογίζει την Αριθμητική Εκφραση που βρίσκεται στα δεξιά του = και τοποθετεί το αποτέλεσμα σε μία νέα θέση μνήμης με το όνομα που προσδιορίζει ο χρήστης στα αριστερά του =.
Στο παράδειγμά μας η εντολή

PLHRWTEO= BASMISQOS+EPIDOMA-KRATHSEIS

έχει σαν αποτέλεσμα να κάνει ο Υ τον υπολογισμό 30000+15000-10000 και να τοποθετήσει το αποτέλεσμα 35000 στη θέση μνήμης με το όνομα PLHRWTEO (βλ Σχ.Β.4).

Εντολή Τέλους Προγράμματος - END

Το τέλος του προγράμματος δηλώνεται με την εντολή END. Η εκτέλεση της εντολής αυτής προκαλεί τερματισμό της σχετικής επεξεργασίας.

Ενα Πρόγραμμα στην BASIC - Το πρώτο απλό μας πρόγραμμα

Με βάση λοιπόν τα παραπάνω και τις εντολές που μάθαμε μέχρι τώρα μπορούμε να γράψουμε το πρώτο απλό μας πρόγραμμα, που θα υλοποιεί το παράδειγμα μισθοδοσίας που αναλύσαμε στο Παράρτημα Α. Στην αρχή για απλότητα θα θεωρήσουμε πως έχουμε ένα μόνο υπάλληλο.
Εδώ θα πρέπει να σημειώσουμε πως ο προγραμματιστής προκειμένου να γράψει ένα πρόγραμμα θα πρέπει να έχει ήδη υπόψη του τις προδιαγραφές και τον αλγόριθμο που είδαμε στο σχ.Α.2 του Παραρτ. Α. Από το σημείο αυτό θα προχωρήσει ώς εξής: μελετώντας τον αλγόριθμο θα διαπιστώσει πως χρειάζεται να κρατήσει θέσεις μνήμης για να τοποθετηθούν τα στοιχεία που είδαμε παραπάνω δηλ ο βασικός μισθός, τα επιδόματα και οι κρατήσεις (δεδομένα εισόδου) όπως και το καθαρό πληρωτέο (αποτέλεσμα). Για πληρότητα θα προσθέσουμε άλλο ένα στοιχείο στα δεδομένα εισόδου, το ονοματεπώνυμο του υπάλληλου.
Για να κρατηθούν λοιπόν οι κατάλληλες θέσεις μνήμης για τα παραπάνω στοιχεία κάθε υπάλληλου θα πρέπει ο προγραμματιστής να δώσει ονόματα στις μεταβλητές αυτές. Θεωρητικά ο προγραμματιστής μπορεί να δώσει όποια ονόματα θέλει αρκεί να είναι σύμφωνα με τη γραμματική της γλώσσας, στην πράξη όμως προσέχει να διαλέγει ονόματα τέτοια που να θυμίζουν τα περιεχόμενα των αντίστοιχων θέσεων μνήμης. Μιά επιλογή ονομάτων (όχι κατανάγκη η καλύτερη) μπορεί να είναι η ακόλουθη:
- ΟΝΟΜΑ για τη θέση μνήμης που θα τοποθετείται το ονοματεπώνυμο του υπάλληλου. Η θέση αυτή της μνήμης θα περιέχει πάντα αλφαβητικούς χαρακτήρες και αυτό πρέπει να το δηλώσουμε κατάλληλα στον Υ, βάζοντας το σημάδι $ στο τέλος του ονόματος της μεταβλητής (ΟΝΟΜΑ$)
- BASMISQOS για το βασικό μισθό
- EPIDOMA για το επίδομα
- KRATHSEIS για τις κρατήσεις
- PLHRWTEO για το καθαρό πληρωτέο

Στη συνέχεια ο προγραμματιστής απλά μεταφράζει κάθε βήμα του αλγόριθμου στην αντίστοιχη εντολή της γλώσσας BASIC. Αυτή η αντιστοίχηση είναι απλή και εμφανής στο απλό μας παράδειγμα, σε πιό σύνθετα όμως προγράμματα τα πράγματα δεν είναι τόσο απλά (χωρίς βέβαια να είναι πάντα και ιδιαίτερα δύσκολα).

Μορφή του προγράμματος BASIC

Κάθε εντολή του προγράμματος BASIC πρέπει να έχει μία αρίθμηση (κάτι σαν αύξοντα αριθμό) που να δείχνει και τη σειρά εκτέλεσής της από τον Υ πχ.
1 εντολή
2 εντολή
3 εντολή

Δεν είναι απαραίτητο η αρίθμηση να είναι συνεχόμενη όπως η παραπάνω αλλά πρέπει να είναι αύξουσα. Συνήθως επιλέγεται από τους προγραμματιστές η αρίθμηση 10, 20, 30 κλπ
10 εντολή
20 εντολή
30 εντολή
.
έτσι ώστε να υπάρχει περιθώριο ενδιάμεσων αριθμών όταν χρειάζεται να παρεμβληθούν μερικές νέες εντολές ανάμεσα σε υπάρχουσες.

Με τα δεδομένα αυτά έχουμε το απλό πρόγραμμα που φαίνεται στο Πρ. Β.1

```
10 INPUT ONOMA$, BASMISQOS, EPIDOMA, KRATHSEIS
20 PLHRWTEO=BASMISQOS + EPIDOMA - KRATHSEIS
30 PRINT ONOMA$, PLHRWTEO
40 END
```

ΠΡ.Β.1

Αλλες Εντολές

Εντολές Προσδιορισμού Ροής Εκτέλεσης Προγράμματος
(Αλλαγής της κατά σειρά εκτέλεσης των εντολών).

Εντολή GOTO

Μέχρι τώρα είδαμε πως οι εντολές εκτελούνται από τον Υ με τη σειρά η μία μετά την άλλη. Τί γίνεται όμως όταν χρειάζεται να υπερπηδηθούν κάποιες εντολές και να συνεχισθεί η εκτέλεση από μιαν άλλη εντολή (πάνω ή κάτω) από την τρέχουσα.
Στην περίπτωση αυτή χρησιμοποιούμε την εντολή GOTO με την μορφή:

GOTO αριθμός εντολής

με την εντολή αυτή ο χρήστης παραγγέλλει στον Υ να μην εκτελέσει την αμέσως επόμενη εντολή αλλά την εντολή που θέλει εκείνος και την προσδιορίζει με τον αριθμό της. Ας πάρουμε το πρόγραμμα του προηγούμενου παραδείγματός μας το οποίο υπολογίζει το πληρωτέο ποσό για ένα μόνο υπάλληλο. Αν θέλουμε να επαναληφθεί η ίδια διαδικασία υπολογισμού πολλές φορές (για πολλούς υπαλλήλους) δεν χρειάζεται βέβαια να γράψουμε τις ίδιες εντολές πολλές φορές, απλά υποδεικνύουμε στον Υ να τις εκτελέσει από την αρχή με την εντολή GOTO 10, όπως φαίνεται στο πρόγραμμα Πρ. Β.2. Εξυπακούεται πως αφού ο Υ πάει στην υποδεικνυόμενη εντολή από κεί και πέρα συνεχίζει την σειριακή εκτέλεση των εντολών μέχρι να του υποδειχθεί να αλλάξει τακτική με ειδική εντολή.

```
10 READ ONOMA$,BASMISQOS,EPIDOMA,KRATHSEIS
20 PLHRWTEO=BASMISQOS+EPIDOMA-KRATHSEIS
30 PRINT ONOMA$,PLHRWTEO
40 GOTO 10
50 END
60 DATA ΚΟΡΡΕ Α,30000,15000,10000
70 DATA ΜΑΚΡΗΣ Π,40000,20000,15000
```

ΠΡ. Β.2

Στο πρόγραμμα αυτό χρησιμοποιούμε τις εντολές READ/DATA για την εισαγωγή των στοιχείων και έτσι τοποθετούμε τα δεδομένα για όλους τους υπαλλήλους της εταιρίας μας μέσα στο πρόγραμμα ώστε να μήν χρειάζεται να τα πληκτρολογούμε κάθε φορά κατά τη διάρκεια της ροής του προγράμματος. Για τα δεδομένα κάθε υπάλληλου χρησιμοποιούμε ξεχωριστή DATA. Κάθε φορά που εκτελείται η εντολή READ (για κάθε GOTO 10) ο Υ παίρνει τα δεδομένα του επόμενου υπάλληλου από την επόμενη εντολή DATA.

Εντολές Επανάληψης - Εντολή FOR ...NEXT

Με την εντολή GOTO 10 που προαναφέραμε προκαλούμε μιά αέναη ανακύκλωση των εντολών του προγράμματος το οποίο δεν φτάνει ποτέ στην εντολή END για να τελειώσει. Οταν θέλουμε να επαναλάβουμε ένα σύνολο εντολών (ή loop εντολών όπως λέγεται) για συγκεκριμένο αριθμό επεναλήψεων, χρησιμοποιούμε το ζεύγος των εντολών επανάληψης FOR / NEXT.

Η εντολή FOR τοποθετείται πρίν από την πρώτη εντολή του Loop και η NEXT μετά την τελευταία ώστε να προδιαγραφούν τα όρια των εντολών των οποίων θα επαναληφθεί η εκτέλεση. Το πόσες φορές θα γίνει η εκτέλεση προσδιορίζεται από ένα μετρητή του οποίου καθορίζεται η πρώτη και η τελευταία τιμή ώστε να αντιστοιχεί στον επιθυμητό αριθμό των επαναλήψεων.

FOR <μετρητής>=1 TO <αριθμός επαναλήψεων>
εντολή-1
εντολή-2

εντολή-ν
NEXT <μετρητής>

Κάθε φορά που φτάνουμε στο τέλος του loop και εκτελείται η εντολή NEXT, ο μετρητής αυξάνεται αυτόματα κατά ένα και η εκτέλεση επιστρέφει στην αρχή του loop, στην εντολή FOR όπου συγκρίνεται η τρέχουσα τιμή του μετρητή με τον μέγιστο αριθμό των επαναλήψεων. Αν η τιμή είναι μικρότερη

ή ίση επαναλαμβάνεται η εκτέλεση των εντολών του loop, αν όχι σημαίνει πως έγιναν οι επιθυμητές επαναλήψεις, οπότε δεν γίνεται άλλη επανάληψη και ο Υ παρακάμπτει τις εντολές του loop και συνεχίζει την εκτέλεση του προγράμματος από την επόμενη του NEXT εντολή.

Αν στο παράδειγμά μας έχουμε 3 υπάλληλους στη εταιρία μας θα πρέπει να χρησιμοποιήσουμε τις εντολές
FOR I=1 TO 3 και NEXT I για να περιβάλουμε τις εντολές επεξεργασίας μας όπως φαίνεται και στο πρόγραμμα Πρ. Β.3.

```
 5 FOR I = 1 TO 3
10 READ ONOMA$,BASMISQOS,EPIDOMA,KRATHSEIS
20 PLHRWTEO=BASMISQOS+EPIDOMA-KRATHSEIS
30 PRINT ONOMA$,PLHRWTEO
40 NEXT I
50 END
60 DATA KOPPE A,30000,15000,10000
70 DATA ΜΑΚΡΗΣ Π,40000,20000,15000
80 DATA ΠΑΡΧΑ Κ,50000,30000,20000
```

ΠΡ. Β.3

Εντολές Απόφασης-Εκτέλεση εντολών

Εντολή IF

ΑΝ ισχύει μία συνθήκη

Τί σημαίνει αλήθεια πως ο Υ αποφασίζει; Σημαίνει πως έχει τη δυνατότητα να επιλέγει μία ενέργεια από σύνολο δυνατών ενεργειών ανάλογα με το άν ισχύει ή όχι μία συγκεκριμένη συνθήκη. Τη δυνατότητα όμως αυτή δεν την έχει από μόνος του την αποκτά εκτελώντας τις κατάλληλες εντολές που έχει γράψει ο προγραμματιστής στο πρόγραμμά του. Είναι λοιπόν σημαντικό μία γλώσσα προγραμματισμού να διαθέτει αυτές τις κατάλληλες εντολές. Η BASIC έχει την λεγόμενη υποθετική εντολή που αρχίζει με το υποθετικό μόριο IF και έχει την μορφή:

IF συνθήκη THEN εντολή

Με την εντολή αυτή ο προγραμματιστής ζητάει από τον Υ να εκτελέσει την εντολή που υπάρχει μετά το THEN μόνο άν ισχύει η συνθήκη που υπάρχει μετά το IF αλλιώς να συνεχίσει με την εκτέλεση της επόμενης εντολής.
Μιά συνθήκη ουσιαστικά προκύπτει από τη σύγκριση μιάς μεταβλητής (ή αριθμητικής έκφρασης) με μία σταθερά ή με το περιεχόμενο άλλης μεταβλητής (ή αριθμητικής έκφρασης).
Για να καταλάβουμε τη λειτουργία της εντολής αυτής ας κάνουμε μιά μικρή παρατήρηση στο προηγούμενο παράδειγμά μας. Οταν γνωρίζουμε τον ακριβή αριθμό υπαλλήλων χρησιμοποιούμε τις εντολές επανάληψης που είδαμε παραπάνω. Ο αριθμός όμως αυτός μπορεί να αλλάζει από καιρού εις καιρό οπότε θα πρέπει αντίστοιχα να αλλάζουμε και στο πρόγραμμα τον αριθμό των επαναλήψεων (στην εντολή FOR). Για να το αποφύγουμε αυτό σκεφτόμαστε ώς εξής: τοποθετούμε τα στοιχεία των υπαλλήλων (πχ. χρησιμοποιώντας την εντολή DATA) και στο τέλος βάζουμε ένα σημάδι για να δείξουμε πως τελειώσαμε. Ας πούμε για παράδειγμα πως το σημάδι (ένδειξη τέλους) είναι ένας υποθετικός υπάλληλος που έχει σαν ονοματεπώνυμο ένα αστερίσκο (*) και μηδενικά όλα τα άλλα στοιχεία. Για να καταλάβει λοιπόν ο Υ πότε ετελείωσαν οι προς επεξεργασία υπάλληλοι και να τελειώσει την επεξεργασία (να *πάει* στο END), θα πρέπει ο προγραμματιστής να του ζητάει με την κατάλληλη εντολή στο πρόγραμμα να ελέγχει κάθε υπάλληλο και άν είναι ο υποθετικός-ένδειξη τέλους (ΟΝΟΜΑ=*) να τελειώνει αν όχι θα σημαίνει πως είναι κανονικός υπάλληλος και θα προχωρεί στον υπολογισμό του μισθού του. Η κατάλληλη εντολή για το σκοπό αυτό είναι η εντολή IF ... THEN όπως φαίνεται και στο πρόγραμμα Πρ. Β.4.

```
10 READ ONOMA$,BASMISQOS,EPIDOMA,KRATHSEIS
20 IF ONOMA$ = *** THEN  GOTO  60
30 PLHRWTEO=BASMISQOS + EPIDOMA - KRATHSEIS
40 PRINT ONOMA$,PLHRWTEO
50 GOTO 10
60 END
70 DATA ΚΟΡΡΕ Α,30000,15000,10000
80 DATA ΜΑΚΡΗΣ Π,40000,20000,15000
80 DATA ΠΑΡΧΑ Κ,50000,30000,20000
```

ΠΡ. Β.4

Με το παραπάνω παράδειγμα παρουσιάσαμε απλά μία περίπτωση εφαρμογής της εντολής IF. Οι χρήσεις της εντολής αυτής είναι απεριόριστες, πράγμα που την κάνει μιά από τις πιό χρήσιμες εντολές που έχει στη διάθεσή του ένας προγραμματιστής.

Και Αλλες Εντολές

Τελειώνοντας την συνοπτική παρουσίαση των βασικών εντολών της BASIC θα πρέπει να προειδοποιήσουμε τους αναγνώστες μας πως και αν ακόμα κατάλαβαν τα παραπάνω παραδείγματα δεν σημαίνει πως έμαθαν και την BASIC. Η BASIC είναι μία ιδιαίτερα πλούσια γλώσσα με πολλές δυνατότητες η γνώση της οποίας απαιτεί την μελέτη των αντίστοιχων εγχειριδίων ή βοηθητικών βιβλίων που αφθονούν. Ο σκοπός μας σαυτό το σύντομο κεφάλαιο ήταν να δώσουμε στους αναγνώστες μας μιά πρώτη γεύση προγραμματισμού και να προετοιμάσουμε όσους ενδιαφέρονται για πιό βαθειά μελέτη.-

Γ ΣΤΟΙΧΕΙΑ ΤΗΣ ΓΛΩΣΣΑΣ PASCAL

Εισαγωγή

Η Pascal μπορεί να θεωρηθεί σαν μία από τις καλύτερες γλώσσες προγραμματισμού, ευρείας χρήσης και ιδανική για να μάθει κανείς ή να διδάξει τις αρχές ενός καλού προγραμματισμού. Πήρε το όνομά της από τον Γάλλο μαθηματικό και φιλόσοφο Blaise Pascal (1623-1662).

Το 1968 παρουσιάζεται από τον Niklaus Wirth του Πολυτεχνείου της Ζυρίχης η πρώτη έκδοση της Pascal (όπου δηλαδή οι γνώσεις και οι εμπειρίες στον προγραμματισμό είχαν ήδη προχωρήσει πολύ).

Το πρώτο βιβλίο της Pascal εκδόθηκε το 1974 από τους K. Jensen-N. Wirth με τίτλο Pascal User Manual and Report.

Στη Ζυρίχη δημιουργήθηκε ένας Pascal compiler γραμμένος σε Pascal και ήταν έτσι κατασκευασμένος, ώστε να μπορεί να τρέξει σχεδόν σε οποιοδήποτε υπολογιστή, είτε αυτός είναι μεγάλος υπολογιστής, είτε μικρός προσωπικός υπολογιστής.

Τα βασικότερα πλεονεκτήματα της Pascal είναι τα ακόλουθα:

α) Τα προγράμματα διαβάζονται εύκολα και γίνονται εύκολα κατανοητά.

β) Το μέγεθος των προγραμμάτων που γράφουμε είναι ως επί το πλείστον μικρό και ως εκ τούτου τρέχουν γρήγορα

και

γ) Η μορφή και η δομή των προγραμμάτων είναι τέτοια που μας βοηθάει να μπορούμε εύκολα να βρούμε τα λάθη μας μέσα σε ένα πρόγραμμα.

Το 1983 η εταιρεία Borland εμφάνισε ένα νέο προϊόν, την Turbo Pascal που περιελάμβανε editor και compiler και παρήγαγε καθαρό εκτελέσιμο κώδικα μηχανής, σε αντίθεση με ό,τι συνέβαινε την εποχή εκείνη. Τα πλεονεκτήματα που προσέφερε ήταν αφενός η μεγάλη ταχύτητα του compiler και αφετέρου η πολύ χαμηλή τιμή της.

Αρχιτεκτονική Προγράμματος

Ένα πρόγραμμα στην Pascal αποτελείται από μια λίστα εντολών, που πρέπει να εκτελεστούν με κάποια σειρά και έχει γενικά την ακόλουθη μορφή :

```
PROGRAM name (output) ;
var ......
BEGIN
      ....
     begin
      .......
     end;
    {εντολές} ;
      ....
END .
```

Αναλυτικώτερα:

Κατά την συγγραφή του προγράμματος χρησιμοποιούμε τόσο τα μικρά όσο και τα κεφαλαία γράμματα.
Στην αρχή του προγράμματος γράφουμε PROGRAM και δίνουμε ένα όνομα - name - που χρησιμεύει στο να προσδιορίζει ονομαστικά το πρόγραμμα που θα φτιάξουμε στη συνέχεια. Με τη βοήθεια παραμέτρων εξασφαλίζουμε την επικοινωνία του προγράμματος με τον έξω κόσμο. Τα δε ονόματα που χρησιμοποιούνται σε ένα πρόγραμμα δηλώνονται πριν χρησιμοποιηθούν ακόμα και οι υπορουτίνες. Τα τυποποιημένα ονόματα εξαιρούνται.

Το κύριο μέρος των εντολών του προγράμματος αρχίζει με την εντολή BEGIN.
Οι εντολές χωρίζονται μεταξύ τους με το ελληνικό ερωτηματικό (;).
Στο τέλος του προγράμματος γράφουμε END και βάζουμε τελεία.
Παραδείγματα:

1.
```
PROGRAM hello (output) ;

  begin
    writeln ( ' hello ') ;
    writeln ( ' happy new year' )
  end.
```

2.
```
PROGRAM  test (output) ;
  begin
    writeln ('to athrisma ton 32 kai 21 einai',  32+21);
    writeln ('to athrisma ton 67 kai 21 einai',  67+21)
  end.
```

Αλφάβητο της Γλώσσας

Το αλφάβητο της Pascal, αποτελείται από αλφαβητικούς, αριθμητικούς και ειδικούς χαρακτήρες.

Οι αλφαβητικοί χαρακτήρες είναι τα γνωστά 26 γράμματα του Αγγλικού αλφαβήτου. Το Ελληνικό αλφάβητο μπορούμε να το χρησιμοποιήσουμε στα σχόλια που κάνουμε σ΄ένα πρόγραμμα.

Οι αριθμητικοί χαρακτήρες είναι τα ψηφία από 0-9.

Ειδικοί χαρακτήρες είναι τα σύμβολα +, _, /, *.

Για τις αριθμητικές εκφράσεις χρησιμοποιούμε τα σύμβολα:

+ πρόσθεση
- αφαίρεση
* πολλαπλασιασμός
/ διαίρεση

Το 1983 η εταιρεία Borland εμφάνισε ένα νέο προϊόν, την Turbo Pascal που περιελάμβανε editor και compiler και παρήγαγε καθαρό εκτελέσιμο κώδικα μηχανής, σε αντίθεση με ό,τι συνέβαινε την εποχή εκείνη. Τα πλεονεκτήματα που προσέφερε ήταν αφενός η μεγάλη ταχύτητα του compiler και αφετέρου η πολύ χαμηλή τιμή της.

Αρχιτεκτονική Προγράμματος

Ένα πρόγραμμα στην Pascal αποτελείται από μια λίστα εντολών, που πρέπει να εκτελεστούν με κάποια σειρά και έχει γενικά την ακόλουθη μορφή :

```
PROGRAM name (output) ;
var ......
BEGIN
      ....
     begin
      .......
     end;
   {εντολές} ;
      ....
END .
```

Αναλυτικώτερα:

Κατά την συγγραφή του προγράμματος χρησιμοποιούμε τόσο τα μικρά όσο και τα κεφαλαία γράμματα.
Στην αρχή του προγράμματος γράφουμε PROGRAM και δίνουμε ένα όνομα - name - που χρησιμεύει στο να προσδιορίζει ονομαστικά το πρόγραμμα που θα φτιάξουμε στη συνέχεια. Με τη βοήθεια παραμέτρων εξασφαλίζουμε την επικοινωνία του προγράμματος με τον έξω κόσμο. Τα δε ονόματα που χρησιμοποιούνται σε ένα πρόγραμμα δηλώνονται πριν χρησιμοποιηθούν ακόμα και οι υπορουτίνες. Τα τυποποιημένα ονόματα εξαιρούνται.

Το κύριο μέρος των εντολών του προγράμματος αρχίζει με την εντολή BEGIN.
Οι εντολές χωρίζονται μεταξύ τους με το ελληνικό ερωτηματικό (;).
Στο τέλος του προγράμματος γράφουμε END και βάζουμε τελεία.
Παραδείγματα:

```
1. PROGRAM hello (output) ;

   begin
      writeln ( ' hello ') ;
      writeln ( ' happy new year' )
   end.

2. PROGRAM  test (output) ;
   begin
      writeln ('to athrisma ton 32 kai 21 einai',  32+21);
      writeln ('to athrisma ton 67 kai 21 einai',  67+21)
   end.
```

Αλφάβητο της Γλώσσας

Το αλφάβητο της Pascal, αποτελείται από αλφαβητικούς, αριθμητικούς και ειδικούς χαρακτήρες.

Οι αλφαβητικοί χαρακτήρες είναι τα γνωστά 26 γράμματα του Αγγλικού αλφαβήτου. Το Ελληνικό αλφάβητο μπορούμε να το χρησιμοποιήσουμε στα σχόλια που κάνουμε σ΄ένα πρόγραμμα.

Οι αριθμητικοί χαρακτήρες είναι τα ψηφία από 0-9.

Ειδικοί χαρακτήρες είναι τα σύμβολα +, _, /, *.

Για τις αριθμητικές εκφράσεις χρησιμοποιούμε τα σύμβολα:

+ πρόσθεση
- αφαίρεση
* πολλαπλασιασμός
/ διαίρεση

Τα σύμβολα *, / προηγούνται των συμβόλων + και - .

Παραδείγματα:

α) 6+7* 2= 20 (πρώτα γίνεται η πράξη 7*2=14 και μετά προσθέτουμε το 6 δηλαδή 14+6=20) (ακέραιος) (integer)

β) 12/2 -2=4 (πραγματικός) (real)

Σε μεγάλες αλγεβρικές παραστάσεις χρησιμοποιούμε τις παρενθέσεις για να δώσουμε προτεραιότητα σε ορισμένες πράξεις.
Τις παρενθέσεις τις βάζουμε μόνο όταν αυτές είναι αναγκαίες για τη σωστή εκτέλεση μιας αλγεβρικής παράστασης.

Σταθερές

Η λέξη CONST στην Pascal χρησιμοποιείται για να δηλώσουμε κάποια σταθερά τιμή που δεν θα αλλάξει κατά την πορεία του προγράμματος. Σταθερά μπορεί να είναι οποιοσδήποτε πραγματικός αριθμός. Στις σταθερές μπορούμε να βάλουμε και πρόσημο. Η δήλωση γίνεται στην αρχή των εντολών του προγράμματος, όπως φαίνεται στο ακόλουθο παράδειγμα:

Παράδειγμα:

```
Program test (output);

CONST
  vasi= 12 ; ipsos=10;

BEGIN

  writeln (' to trigono exei vasi : ', vasi ,' kai ipsos: ' , ipsos) ;
  writeln (' to embado toy trigonou einai :', vasi*ipsos);
  writeln

END.
```

Ασκήσεις

1. Πως θα γράψουμε τις παρακάτω σταθερές;

ύψος=1.80μ

βάρος= 10 κιλά

διάμετρος= 5εκ

αριθμός μαθητών = 100

2. Δηλώστε σε ένα πρόγραμμα τις σταθερές α=185 και β=156 και τυπώστε τες.

3. Δηλώστε σε ένα πρόγραμμα Pascal τις παρακάτω σταθερές X=35, α=214.

Μεταβλητές

Μεταβλητές είναι δείκτες που μπορεί να παίρνουν διάφορες τιμές κατά τη διάρκεια εκτέλεσης ενός προγράμματος. Οι μεταβλητές μπορεί να είναι της μορφής: κ, n, m,...z, ή λέξεις όπως age, time κλπ .

Η δήλωσή τους μέσα στο πρόγραμμα γίνεται με την εντολή VAR. Οι μεταβλητές VAR γράφονται μετά την εντολή CONST και πριν την εντολή BEGIN.

Η δήλωσή τους γίνεται με το σύμβολο :=

Τα ονόματά τους μπορεί να είναι των εξής τύπων:

integer
real
boolean
char

Παραδείγματα:

1. var
   pets: integer
   ch : char
   p,x : real

2. var num, x : integer;

   begin
   num:= 45;
   x:= num+20;

3. var place : char;

   begin
   place:= 'a';

4. var x, y : real;

   begin
   x:= 34.9 * y;

Αναλυτικότερα:

Οι μεταβλητές τύπου integer δεν περιέχουν δεκαδικό μέρος

δηλαδή: οι αριθμοί 34, -358, 0, 564 είναι ακέραιοι.

Οι μεταβλητές τύπου real περιέχουν δεκαδικό μέρος

δηλαδή: οι αριθμοί 34.55, -358.9 , 0.0 , 564.1E+11 είναι πραγματικοί

Οι μεταβλητές τύπου char είναι γράμματα, χαρακτήρες, ειδικά σύμβολα κλπ

δηλαδή τα ακόλουθα: ( ) * & % + = { } [ ]
XYZ HI_ANN

Ασκήσεις

1. Τι είναι οι παρακάτω μεταβλητές;

χ, 87, -56.8

2. Ποιες από τις παρακάτω μεταβλητές είναι λάθος και γιατί;

A:= 89
κ:=5%
MM:= .789

TT:= ATN

K:= MM

H:= M

Η εντολή INPUT

Με την εντολή αυτή εισάγουμε μία νέα μεταβλητή που δεν έχει αρχική τιμή αλλά παίρνει κάποια τιμή κατά την διάρκεια του προγράμματος. Εάν σε κάποιο πρόγραμμα η μεταβλητή διαβάζεται μόνο από το τερματικό τότε βάζουμε την λέξη input στην αρχή του προγράμματος, δηλαδή στην εντολή PROGRAM.

Παράδειγμα:

PROGRAM test (input,output)

Η εντολή READ

Με την εντολή αυτή δίνουμε δεδομένα στο πρόγραμμα μας.
Δηλαδή: read (x,z); ή

readln (a,b,c) που εξετάζει όλες τις τιμές που είναι στην ίδια γραμμή

Η εντολή read μπαίνει μετά την εντολή BEGIN.

Το <enter > διαβάζεται σαν κενό στη read και δεν αξιολογείται από τη readln

Παράδειγμα:

```
PROGRAM  test ( input ,output );
var x:  integer;
BEGIN
     wtiteln ('dose ena arithmo'); read (x);
      writeln;
     writeln (' to diplasio tou x einai 2x=' ,2*x)
END.
```

Ασκήσεις

1. Τι διαβάζει το παρακάτω πρόγραμμα

```
program ena (input,output);
 var  x,y: integer;

begin
       read (x);
       read  (y);
end.
```

Δεδομένα 98 και 23

2. Τι διαβάζει το παρακάτω πρόγραμμα;

```
program dio (input,output);
 var  x,y: integer;

begin
        readln (x);
        readln  (y);
end.
```

Δεδομένα 34 και 29

3. Τι διαβάζει το παρακάτω πρόγραμμα;

```
program tria  (input,output);
 var  x,y,z,k,d, n: char;

begin
        readln (x);
        readln  (y);
        readln (z,k,d);
        readln  (x, n);
end.
```

Δεδομένα ΑΒΟΡΥΙ

4. Τι διαβάζει το παρακάτω πρόγραμμα;

```
program tria  (input,output);
 var  x,y,z,k,d, n: char;

begin
        read(x); read(y);
        read(z,k,d);
        read(x, n);
end.
```

Δεδομένα ΑΒΟΡΥΙ

Η εντολή WRITE

Με την εντολή αυτή τυπώνουμε μια παράσταση ή ένα αποτέλεσμα.

Οτιδήποτε βάλουμε σε ' ' εισαγωγικά τυπώνεται όπως είναι δηλαδή χωρίς αλλαγές.

Παράδειγμα:

```
PROGRAM  ena  ( input ,output );

BEGIN
    wtiteln (' oi arithmoi  1,3,5  einai peritoi');
     writeln;
    writeln (' to triplasio tou 5 einai 3*5=' ,3*5)
END.
```

Άσκηση

Τι τυπώνει το παρακάτω πρόγραμμα;

```
program tria  (input,output);
 var  x,y,z,k,d, n: char;

begin
        read(x); read(y);
        read(z,k,d);
        read(x, n);
        writeln (x,  y, 'z=', z,  'k=',k);
end.
```

Δεδομένα A,T, M, O, Z,B.

Άλλες εντολές στην PASCAL

1. Εντολές ανάθεσης

Για να κάνουμε κάποια ανάθεση τιμής σε κάποια μεταβλητή χρησιμοποιούμε τις εντολές ανάθεσης (assignment statements), πού έχουν την μορφή:

μεταβλητή := έκφραση παράδειγμα: κ:= 3/χ-9

όπου στη θέση έκφραση μπορεί να γράψουμε οποιαδήποτε παράσταση.

Σύμφωνα με αυτές τις αναθέσεις, η μεταβλητή παίρνει τη τιμή της έκφρασης μέχρις ότου βρεθεί κάποια άλλη εντολή ανάθεσης και αλλάξει την τιμή της μεταβλητής.

Παραδείγματα:

```
const pi=3.14

var  x,y : real;
     n,m,k: integer;
```

2. Η εντολή IF

Η εντολή IF έχει την μορφή :

IF συνθήκη THEN *πρόταση 1* ELSE *πρόταση 2* .

Εάν η συνθήκη ισχύει τότε εκτελείται η πρόταση 1, αλλιώς εκτελείται η πρόταση 2.

Ας σημειωθεί εδώ ότι ανάμεσα στις προτάσεις 1 και 2 δεν βάζουμε το ; .
Στη συνθήκη μπορούμε να χρησιμοποιήσουμε ένα από τα παρακάτω σύμβολα:

>	μεγαλύτερο
<	μικρότερο
>=	μεγαλύτερο ή ίσο
<=	μικρότερο ή ίσο
=	ίσο
<>	διάφορο (μη ίσο)

Παραδείγματα:

```
1]  if a> k then
      writeln ( a,k+1)
    else
      writeln ( k-1, a)

2]  if y>= 0 then
      a := y
    else
      a := -y

3]  if y1>y2 then writeln ('in order') ELSE
      if y1<y2 then writeln ('not in order') else
      writeln ('telos')
```

Ασκήσεις

1. Να γραφεί ένα πρόγραμμα που να τυπώνει το χ όταν αυτό είναι μεγαλύτερο του 0.

2. Να γράφεί ένα πρόγραμμα που να τυπώνει το μέσο όρο 6 βαθμών αν αυτός είναι μεγαλύτερος από το 5.

Οι προτάσεις AND, OR, NOT

Οι προτάσεις αυτές εκφράζουν σύζευξη, διάζευξη και άρνηση μεταξύ δύο προτάσεων.

Παραδείγματα

```
if (A=1) and (B=3) then writeln ('Hello');

if (A=1) or (B=3) then writeln ('Hello');

if NOT ((A=1) and (B=3)) then writeln ('Hello');

if (A>1) and (B=3) then writeln ('Hello');
if (A<>1) and (B<3) then writeln ('Hello');

if (A=1) or (B>3) then writeln ('Hello');

if (A=1) or (B<>3) then writeln ('Hello');
```

3. Η εντολή FOR .... TO... DO...

Οταν θέλουμε να κάνουμε μια διαδικασία πολλές φορές τότε χρησιμοποιούμε εντολές επανάληψης ή ***loops***.

Η γενική μορφή της εντολής είναι :

for n:=a **to** b **do** k
όπου n είναι μεταβλητή
a,b είναι εκφράσεις που δείχνουν κάποια αρχή και κάποιο τέλος και
κ είναι μια οποιαδήποτε πρόταση

Παραδείγματα:

1.
```
Program test (output);

 begin
   for n:= 1 to 10  do
     writeln ('*****');
     writeln ('.....');
      writeln
end.
```

2.
```
Program test (input,output);
   var
     y, n:integer;
 begin
   readln (y,n);
   for n:= 1 to n do
       y:=y*3;
    writeln (y)
end.
```

4. Η εντολή WHILE ..... DO ...

Την εντολή αυτή την χρησιμοποιούμε όταν δεν γνωρίζουμε τον ακριβή αριθμό των επαναλήψεων που πρέπει να κάνουμε σε ένα επαναληπτικό σχήμα (loop). Οι επαναλήψεις τελειώνουν όταν δεν επαληθεύεται πλέον η προς εξέταση σχέση.

Αν η τιμή της παράστασης από την αρχή δεν επαληθεύεται η εντολή μετά το **do** δεν εκτελείται ποτέ.

Η γενική μορφή της εντολής είναι:

while test **do** statement όπου test είναι μία σχέση που εξετάζουμε και statement είναι μία πρόταση

Παραδείγματα:

```
1].  while  not eof (oldmaster) do
      begin
        writeln
              ......
           end;
        .........
2].  while  n<> 0 do
       k:=k+1
```

5. **Η εντολή REPEAT .... UNTIL....**

Μια άλλη εντολή επανάληψης είναι η Repeat όπου κάνουμε την επανάληψη της πρότασης τόσες φορές έως ότου επαληθευτεί το κριτήριό μας. Η γενική μορφή της εντολής είναι:

repeat statement... until test όπου statement είναι οι πρόταση/σεις προς επανάληψη και test είναι το κριτήριο που εξετάζουμε

Παράδειγμα:

```
Program kletter  (input, output);

const
  space =' ' ;
var
   letter : char ;
begin
   write ( 'write your word' ) ;

repeat
   read (letter)
until    input^ = space ;

writeln;
writeln (letter)

end.
```

Συναρτήσεις

Οι πιο συχνές συναρτήσεις είναι οι παρακάτω:

- ABS= απόλυτη τιμή

Με τη συνάρτηση αυτή παίρνουμε την απόλυτη τιμή του αριθμού που δίνουμε

Παραδείγματα

ABS (-43) = 43
ABS (-55.9) = 55.900000E+00

- COS= συνημίτονο

Δίνει την τιμή του συνημιτόνου

Παραδείγματα

COS (0) = 1

- EXP=εκθετική συνάρτηση

Δίνει την τιμή της εκθετικής συνάρτησης

Παραδείγματα

EXP (10) e στη δύναμη του 10

- LN=λογαριθμική συνάρτηση

Παραδείγματα

Δίνει την τιμή της λογαριθμικής συνάρτησης

- DIN & Mod

Δίνει το αέραιο πηλίκο ή υπόλοιπο της διαίρεσης δύο αριθμών.

- ODD=άρτιος

Δίνει τιμή αληθινή όταν ο αριθμός είναι άρτιος

- Random

Επιτρέπει έναν τυχαίο πραγματικό αριθμό από το 0 έως το 09999

- ROUND = δίνει τον πλησιέστερο ακέραιο

- SIN=ημίτονο

Δίνει την τιμή του ημιτόνου

Παραδείγματα

SIN (PI/2) = 1.0
SIN (0) =0

- TRUNC

Δίνει την τιμή του αριθμού χωρίς δεκαδικά

Παραδείγματα

TRUNC (49.87) = 49

Πιό κάτω στα σχήματα ΠΡ. Γ. 1,2 αναφέρονται παραδείγματα χρήσης των συναρτήσεων αυτών.

```
Program Round Number (Input, Output)
Var
Number: Real;
Round Number: Integer;
Begin
  Writeln ("Δώσε μου ένα αριθμό για να τον στρογγυλοποιή σω");
  Readln (Number);
  Round Number = Round (Number);
  Writeln ("Ο Αριθμός", Number, "στρογγυλοποιημένος είναι",
  Round Number);
End.
```

ΠΡ. Γ. 1

```
Program Round Number (Input, Output)
Var
Random Number: Real;
Number: Integer;
Begin
  Writeln ("Δώσε μου ένα θετικό ακέραιο αριθμό);
  Readln (Number);
  Random Number: = (Random* Number);
  Writeln (Θα σου δώσω ένα τυχαίο αριθμό από το 0 έως το,
  Number,);
  Writeln (Random Number);
End.
```

ΠΡ. Γ. 2

```
Program Hotel Rooms; (Input, Output)
Var
Club Members, Double Rooms, Single Room: Integer;
Begin
  Writeln (Πόσα άτομα της ομάδας θα μείνουν στο ξενοδοχείο;)
  Readln (Club Members);
  Double Rooms: = (Club Members Div 2);
  Single Room: = (Club Members Mod 2);
  Writeln (Θα χρειαστείτε να κλείσετε, Double Rooms, δίκλινα
       δωμάτια και, Single Room, μονόκλινο δωμάτιο);
End.
```

ΠΡ. Γ. 3

Συντομογραφικοί Οροι

ACC Accumulator (καταχωρητής του Επεξεργαστή)
AI Artificial Intelligence- Τεχνητή Νοημοσύνη
AM Αριθμητική Μονάδα
ASCII American Standard Code for Information Interchange. Κωδικοποίηση χαρακτήρων

BIOS Basic Input Output System
BPI Bytes Per Inch. Μονάδα πυκνότητας εγγραφής μαγνητικών ταινιών
BPS Bits per Second μονάδα μέτρησης μεταφοράς στοιχείων
BSC Binary Synchronous Communications. Σύγχρονο πρωτόκολλο

CAD/CAM Computer Aided Design/Manufacture. Εφαρμογές Γραφικών
CD-ROM Compact Disk-Read Only memory
CIM Computer Input Microfiche. Μηχάνημα Εισόδου
CGA Color Graphics Adaptor, Κάρτα Γραφικών 640x240 pixel
COM Computer Output Microfiche. Μηχάνημα Εξόδου
CP/M Control Program for Microprocessors
CPS Characters Per Second. Μονάδα ταχύτητας εκτύπωσης
CPU Central Processing Unit. Κεντρική Μονάδα Επεξεργασίας
CRT Cathode Ray Tube. Οθόνη

DBMS Data Base Management System. Πρόγραμμα για επεξεργασία Βάσεων Δεδομένων
DCE Data Communications Equipment. Βοηθητική Μονάδα Επικοινωνίας (modem)
DES Data Entry System. Συγκεντρωτική εισαγωγή στοιχείων με πληκτρολόγηση
DMA Direct Memory Access. Απευθεία πρόσβαση των περιφερειακών στη μνήμη
DPI Dots Per Inch. Μονάδα μέτρησης διακριτικής ικανότητας οθόνης
DTE Data Terminal Equipment.. Τερματική Μονάδα Επικοινωνίας (Υ ή τερματικό)
DTP Desk Top Publishing. Συστήματα Επιτραπέζιων Εκδόσεων

EBCDIC Extended Binary Coded Decimal Interchange Code. Πρότυπο Κωδ/σης IBM
ECL Emitter Coupled Logic. Τεχνολογία κατασκευής λογικών κυκλωμάτων υψηλών προδιαγραφών
EDI Electronic Data Interchange
EDP Electronic Data Processing. Μηχανογραφικές Εφαρμογές
EGA Enhanced Graphics Adaptor. Κάρτα γραφικών 770x350 pixel
EISA Extended ISA (βλ.λ). Εναλλακτικό πρότυπο bus 32-μπιτων PC
ΕΛΟΤ Ελληνικός Οργανισμός Τυποποίησης
EPROM Erasable PROM
ES Expert Systems. Συστήματα Εμπειρογνώμονες

FEP Front End Processor. Μικρός Υ συνδεδεμένος με mainframe για την εξυπηρέτηση της επικοινωνίας με μεγάλο αριθμό τερματικών.

GB Giga Bytes. Μονάδα χωρητικότητας ΜΜΚ (κυρίως)
GFLOPS Giga Fgoating Point Operations Per Second. Δισεκατομμύρια δεκαδικές πράξεις το δευτερόλεπτο. Μονάδα ισχύος επεξεργαστών υπερυπολογιστών

H/W Hardware.

IBG Inter Block Gap. Απόσταση μεταξύ διαδοχικών blocks ταινίας
IC Instruction Counter. Καταχωρητής της ΚΜΕ
IPS Inches Per Second. Μονάδα ταχύτητας ταινιών
IR Instruction Register
ISA Industry Standard architecture. Πρότυπο bus 16-μπιτων PC
ISDN Integrated Services Data Network. Ενιαίο Δίκτυο Ψηφιακής Μετάδοσης Φωνής - Εικόνας - Δεδομένων

K Kay= 210=1024 (περίπου 100)
KB Kay Bytes μονάδα μέτρησης σε χιλιάδες χαρακτήρες
KBS KBytes Per Second. Μονάδα ταχύτητας μεταφοράς χαρακτήρων μεταξύ μνήμης και ΜΜΚ
ΚΜΕ Κεντρική Μονάδα Επεξεργασίας

LAN Local Area Network. Τοπικό δίκτυο υπολογιστών
LP Line Printer. Εκτυπωτής Γραμμών
LPM Lines Per Minute. Μονάδα ταχύτητας εκτύπωσης
LSI Large Scale Integration. Ολοκληρωμένα κυκλώματα υψηλής πυκνότητας στοιχείων

M Mega=1000K
MAR Memory Address Register. Καταχωρητής της ΚΜΕ
MB Mega Byte (εκατομμύρια χαρακτήρες)
MCA Micro Channel Architecture. Πρότυπο bus για τη σειρά PS/2 των IBM PC
MDR Memory Dta Register
ΜΕ Μονάδα Ελέγχου
MHz Mega Hertz. Μονάδα ταχύτητας ρολογιού
MICR Magnetic Ink Character Recognition. Μηχανήματα εισαγωγής μαγνητισμένων χαρακτήρων
MIPS Mega Instructions Per Second. Μονάδα ισχύος επεξεργαστών
MOS Metal Oxide Semiconductor
MS-DOS Microsoft Disk Operating System

OCR Optical Character Recognition. Μέθοδοι και μηχανήματα εισαγωγής χαρακτήρων καθορισμένου σχήματος
OMR Optical Mark Recognition
OS/2 Operating System για τα συστήματα PS/2 της IBM
OSI Open System Interconnection. Πρότυπο Αρχιτεκτονικής Δικτύων, 7 επιπέδων.

PC Personal Computer - πρώτη γενιά προσωπικών υπολογιστών IBM
PS/2 Personal System/2 - δεύτερη γενιά προσωπικών Y IBM
PROM Programmable ROM (μνήμη)

RAM Random Access Memory. Τσίπς κύριας μνήμης Y
RISC Reduced Instruction Set Computer. Αρχιτεκτονική κατασκευής υπολογιστών.
ROM Read Only Memory
RS-232 Recommended Standard-232 (για σειριακές επικοινωνίες)

SPOOL Simultaneous Peripheral Operations On Line. Μεθοδος εκτύπωσης μέσω ταινίας ή δίσκου
SCSI Small Computer System Interface
SDLC Synchronous Data Link Control. Σύγχρονο πρωτόκολλο
SNA System Network Architecture. Αρχιτεκτονική Δικτύων της IBM
TPI Tracks Per Inch. Μονάδα πυκνότητας tracks δίσκου.

VAN Value Added Network. Υπηρεσίες μέσω δημοσίων δικτύων
VDU Visual Display Unit. Οθόνη
VGA Video Graphics Array
VLSI Very Large Scale Integration. Τσίπς πολύ υψηλής πυκνότητας

WAN Wide Area Network. Δίκτυο υπολογιστών μεγάλων αποστάσεων
WP Word Processing. Εφαρμογές Επεξεργασίας Κειμένου

Ευρετήριο Ορων

Βιβλιογραφία

Α. Για τα Μέρη Ι και ΙΙ

Tanenbaum, A.S. *Structured Computer Organization*, Prentice-Hall International Editions, 1990

Ιδιαίτερη αναφορά πρέπει να γίνει στο περιοδικό:
BYTE - International Edition της McGraw-Hill

Επίσης στα ελληνικά περιοδικά
- Computer για Ολους
- Computer and Software
- Open Newsletter
- RAM

Β. Για το Μέρος ΙΙΙ

- Για το κεφάλαιο 13

1. Akl S. (1985) Parallel Sorting Algorithms. Academic Press, New York.
2. Hockney R. W. and Jesshope C. R. (1988)Parallel Computers 2. Adam Hilger, Bristol.

Β. Για το Μέρος IV

- Για το κεφάλαιο 17

- **Ανθούλιας Τάσος:** ®Πληροφορική και εκπαίδευση, εκδ. GUTENBERG, Αθήνα 1989.

- **Ινστιτούτο Παιδαγωγικών ερευνών:** Οι Η/Υ στην εκπαίδευση και ειδικότερα στην πρωτοβάθμια και δευτεροβάθμια εκπαίδευση, Αθήνα 1988.

- **Καΐλα Μ.-Πολεμικάς Ν.-Φιλίππου Γ.:** ®Ατομα με ειδικές ανάγκες, Β' τόμος, Διεπιστημονικό Ευρωπαϊκό Συμπόσιο, Ρόδος, Μάιος 1992, εκδ. Ελληνικά Γράμματα, Αθήνα 1994.

- **Μεϊμάρης Μ.:** Νέες τεχνολογίες και πρόσωπα με ειδικές ανάγκες στο ®Ατομα με ειδικές ανάγκες, Β' τόμος, Διεπιστημονικό Ευρωπαϊκό Συμπόσιο, Ρόδος, Μάιος 1992, εκδ. Ελληνικά Γράμματα, Αθήνα 1994.

- **Νικόδημος Σ.:** ®Ατομα με ειδικές ανάγκες, Β' τόμος, Διεπιστημονικό Ευρωπαϊκό Συμπόσιο, Ρόδος, Μάιος 1992, εκδ. Ελληνικά Γράμματα, Αθήνα 1994.

- **Παπάς Γρ. Γιώργος:** ®Η πληροφορική στο σχολείο, εκδ. Συμεών, Αθήνα 1989.

- **Σιδέρη-Ζώνιου Αθηνά:** ®Οι ανάπηροι και η εκπαίδευσή τους. Μια ψυχοπαιδαγωγική προσέγγιση της ένταξης. Εκδ. Βιβλιογονία, Αθήνα 1991.

Ξένη βιβλιογραφία

1. **Au. K. H., & Kawikami, A. J.** (1986). The influence of the social organization of instruction on children's text comprehension ability: A Vygotskian perspective. In *The contexts of school based literacy* (pp. 63-78). T. Raphael (Ed.), New York: Random House.
2. **Aveling, N., Smith, S., & Wilson, C.** (1992). Meeting the needs of isolated students: is a technological fix the answer? Parer, Ed. *Academia Under Pressure: Theory and Practice for the 21st Century, Research and Development in Higher Education, 15,* Churchiil Vic: HERDSA
3. **Bakhtin,** M. (1981): *The Dialogic Imagination,* trans. C. Emerson and M. Holquist. Austin: University of Texas Press.
4. **Banathy, B. H.** (1991). Systems design of education: A journey to create the Cliffs, NJ: Educational Technology Publications.
5. **Beaudoin, M.** (1990). Instructor's changihg role in distance education. American Journal of Distance Education, 4(2), 21-29.
6. **Bignall,** A. R., Bond, D. K., Rice, J. C., & Windley, P. J. (1994). Uses of Mosaic in a university setting. *The Second International WWW Conference 94: Mosaic and thw Web Advance Proceedings,* Vol. 1, October
7. **Bogdan,** R. C. & Biklen, S. K. (1982). *Qualitative research for education: An introduction to theory and methods.* New York: Allyn and Bacon
8. **Brown,** A. L., Bransford, J. D., Ferrara, R. A., & Campione, J.C. (1983). Learning, remembering, and understanding. *Handbook of Child Psychlogy: Cognitive Development,* Vol. 3, Wiley, pp. 77, 166.
9. **Burge, E., & Howard, J.L.** (1990). Audio-conferencing in graduate education: A case study. American Journal of Distance Education, 4(2), 3-13.
10. **Campbell,** J. K., Hurley, S., Jones, S. B., & Stephens, N.M. (1995). Constructing educational courseware using NCSA Mosaic and the World Wide Web. *Electronic Proceedings of the Third WWW Conference* 95, April.
11. **Carrison, R. D.** (1989). Distance education. In S. Merriam & P. M. Cunningham (Eds.), Handbook of adult and continuing education. San Francisco: Jossey-Bass.

12. **Cazden,** C. (1988). *Classroom discourse: the language of teaching and learning.* Portsmouth: Heinemann.
13. **Cuban,** L. (1984). *How teachers taught: Constancy and change in American classrooms, 1890-1980.* New York: Longman.
14. **Delgado-Gaitan,** C., 1989. Classroom literacy activity for Spanish-speaking students. *Linguistics and Education: An International Research Journal* 1. Pp. 285-298.
15. **Dwyer,** D. C., Ringstaff, C., & Sandholtz, J. H. (1991). Changes in teachers' beliefs and practices in technology-rich classrooms. *Educational Leadership.* May 45-52.
16. **Eisner, E. W.** (1991). What reallly counts in scools. Educational Leadership, 48(3), 10-17.
17. **Farnham-Diggory,** S. (1990). *"Schoooling", The Developing Child.* Harvard University Press. MA, Cambridge
18. **Glasser,** B. G., & Strauss, A.L. (1967). *The discovery of grounded theory: Strategies for qualitative research.* Chicago: Aldine
19. **Gonzalez,** N., Moll, L., Floyd-Tenery, M., Rivera, A., Rendon, P., Gonzales, R. & Amanti, C. (1993). *Teacher Research on Funds of Knowledge: Learning from households.* (OERI Contract No R117G10022) Tucson, AZ: National Center for Research on Cultural Diversity and Second Language Learning
20. **Hammersley,** M., & Atkinson, P. (1983). *Ethnography principles and practice.* New York: Tavistock.
21. **Haughey, M.** (1983). Teaching and learning via satellite. Victoria, BC: University of Victoria. (ERIC Document Reproduction Service No. ED 235 791).
22. **Heath,** S. B. (1983). *Ways with words: Language, life, and work in communities and classrooms.* Cambridge University Press.
23. **Johnson,** D. W., Johnson, R. T., & Smith, K.A. (1991). Cooperative learning: increasing college faculty instructional productivity. *ASHE-ERIC Higher Education.* Report No. 4, George Washington University
24. **Jones,** R. P., Ruehr, F., & Salter, R. (1996). Web-based laboratories in the introductory curriculum enhance formal methods. CISCSE Bull. 28 (1), ACM Press, pp. 160-164.
25. **Kember, D.** (1995). Open learning courses for adults: A model of student progress. Englewood Cliffs, NJ: Educatinal Technology Publications.
26. Kentucky Department of Education (1991). **Kentucky.** Educational Reform Act assessment materials. Frankofort: Kentucky Department of Education

27. **Lloyd P. Rieber.** Using Computer-Based Microworfels with children with Pervasive Developmental Disorders: An Informal Case study. Jl of Educat. Multim. and Hypermedia (1995) 4 (1) 75-94.
28. **Lynch,** P. J., 1995 Entry-level multimedia authoring tools for education. *Syllabus* 18 8, pp, 10-18
29. **Mehan,** H. (1985). *Computers in the classrooms: A quasi-experiment in guided change* (NIE Grant 6-83-0027). Washington DC: National Institute for Education
30. **Murray,** J. (1986). Work Processing in Elementary Schools: Seven Case Studies. *Education and Technology Services* (Report No. Ed. 416 926-47-7) Toronto: The Ontario Institute for Studies in Education
31. **Neorman,** D. A., & Spohrer, J. C. (1996). Learner-centered education. *Communications of the ACM, 39 (4).*
32. **Neumann, D. R.** (1986). Teleconferencing as a public speaking platform. In L. A. Parker & C. H. Olgren (Eds.), Teleconferencing and electronic communication V. Madison, WI: University of Wisconsin-Extension.
33. **Philips,** S.U. (1972). Participant Structures and Communicative Competence: Warm Springs Children in Community and Classroom. In *Functions of Language in the Classroom,* C. B. Cazden, V.P. John and D. Hymes (Eds), New York: Teachers College
34. **Reilly,** B. (1992). The negotiations of group authorship among second graders using multimedia composing software (Apple Classrooms of Tomorrow Report 14). Cupertino, CA: Apple Computer Publications.
35. **Riddle J., F.** (1994). Factors contributing to achievement and course satisfaction of distance education students.
36. **Riddle, J. F.** (1994). Factors constributing to achievement and course satisfaction of distance education students. Greeley, CO: University of Northern Colorado. Unpublished doctoral dissertation.
37. **Ringstaff,** C., Sandholtz, J. H., & Dwyer, D. C. (1993). *Trading places: When teachers utilize student expertise in technology-intensive classrooms.* (Apple Classrooms of Tomorrow Publishers Report 15), Cupertino, CA: Apple Computer Publications
38. **Sponder, B. M.** (1990). Distance education in rual Alaska: An overview of teaching and learning practices in audioconference courses. (Monograph Series in Distance Education No 1). Fairbanks: University of Alaska Center for Cross-Cultural Studies.
39. **Thiagarajan, S.** (1978). Loneliness of the long-distance learner. Audiovisual Instruction, 23(1), 22+.
40. **Vygotsky,** L. (1978). *Mind in society: The development of higher psychological processes.* Cambridge, MA: Harvard University Press.

41. **Wagner, E. D.** (1990). Instructional design: Contingency management for distance education. In M. G. Moore (Ed.), North American issues in distance education. llondon: Pergamon Press.
42. **Wellens, R. A.** (1986). Use of a psychological distancing model to assess differences in telecommunications media. In L. A. Parker 7 C. H. Olgren (Eds.), Teleconferencing and electronic communication V. Madison, WI: University of Wisconsin-Extension.
43. **Willis, B.** (Ed.) (1994). Distance education: Strategies and tools. Englewood Cliffs, NJ: Educational Technology Publications. Greeley, CO: University of Northern. In M. G. Moore (Ed), London Pergamon Press.
44. **Zeuli,** J.S., & Floden, R.E. (1987). *Cultural incongruities and inequities of schooling: implications for practice from ethnographic research?* (Occasional Paper No 117). East Lansing: Michigan State University, Institute for Research on Teaching.