ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ ΑΛΓΟΡΙΘΜΩΝ

Αλγόριθμος είναι μία πεπερασμένη σειρά ενεργειών, αυστηρά καθορισμένων και εκτελέσιμων σε πεπερασμένο χρόνο, που στοχεύουν στην επίλυση ενός συγκεκριμένου προβλήματος.

Κάθε αλγόριθμος απαραίτητα ικανοποιεί τα επόμενα κριτήρια :

- **Είσοδος (input)** . Καμία, μία ή περισσότερες τιμές δεδομένων πρέπει να δίνονται ως είσοδοι στον αλγόριθμο. Η περίπτωση που δεν δίνονται τιμές δεδομένων εμφανίζεται, όταν ο αλγόριθμος δημιουργεί και επεξεργάζεται κάποιες πρωτογενείς τιμές με τη βοήθεια συναρτήσεων παραγωγής τυχαίων αριθμών ή με τη βοήθεια άλλων απλών εντολών.
- **Έξοδος (output)** . Ο αλγόριθμος πρέπει να δημιουργεί τουλάχιστον μία τιμή δεδομένων ως αποτέλεσμα προς το χρήστη ή προς έναν άλλο αλγόριθμο.
- **Καθοριστικότητα (definiteness)**. Κάθε εντολή πρέπει να καθορίζεται χωρίς καμία αμφιβολία για τον τρόπο εκτέλεσής της. Λόγου χάριν, μία εντολή διαίρεσης πρέπει να θεωρεί και την περίπτωση, όπου ο διαιρέτης λαμβάνει μηδενική τιμή.
- **Περατότητα (finiteness)**. Ο αλγόριθμος να τελειώνει μετά από πεπερασμένα βήματα εκτέλεσης των εντολών του. Μία διαδικασία που δεν τελειώνει μετά από ένα συγκεκριμένο αριθμό βημάτων δεν αποτελεί αλγόριθμο, αλλά λέγεται απλά υπολογιστική διαδικασία (computational procedure).
- **Αποτελεσματικότητα (effectiveness)** . Κάθε μεμονωμένη εντολή του αλγόριθμου να είναι απλή. Αυτό σημαίνει ότι μία εντολή δεν αρκεί να έχει οριστεί αλλά πρέπει να είναι και εκτελέσιμη.

Η έννοια του αλγορίθμου δε συνδέεται αποκλειστικά και μόνο με προβλήματα πληροφορικής.

ΣΠΟΥΔΑΙΟΤΗΤΑ ΑΛΓΟΡΙΘΜΩΝ

Οι σκοπιές από τις οποίες μελετά η πληροφορική τους αλγορίθμους είναι οι ακόλουθες :

- ✓ **Υλικού:** το κατά πόσο δηλαδή επηρεάζεται η ταχύτητα εκτέλεσης ενός αλγορίθμου, από τη δομή και τις δυνατότητες που έχει και υποστηρίζει το υλικό του υπολογιστή.
- ✓ **Γλωσσών προγραμματισμού :**η δομή και το πλήθος των εντολών σε έναν αλγόριθμο, εξαρτάται άμεσα από το είδος της γλώσσας στην οποία θα υλοποιηθεί ο αλγόριθμος.
- ✓ **Θεωρητική :**το αν μπορεί ή όχι να βρεθεί ένας αποδοτικός αλγόριθμος για την επίλυση κάθε προβλήματος.
- ✓ **Αναλυτική :**η μελέτη για τους υπολογιστικούς πόρους, που χρειάζεται ένας αλγόριθμος.

ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΑΝΑΠΑΡΑΣΤΑΣΗ ΑΛΓΟΡΙΘΜΩΝ

Στη βιβλιογραφία συναντώνται διάφοροι τρόποι αναπαράστασης ενός αλγόριθμου :

- **Με ελεύθερο κείμενο (free text) ,**που αποτελεί τον πιο ανεπεξέργαστο και αδόμητο τρόπο παρουσίασης αλγορίθμου. Έτσι εγκυμονεί τον κίνδυνο ότι μπορεί εύκολα να οδηγήσει σε μη εκτελέσιμη παρουσίαση παραβιάζοντας το τελευταίο χαρακτηριστικό των αλγορίθμων, δηλαδή την αποτελεσματικότητα.
- **Φυσική γλώσσα (natural language) κατά βήματα.** Στην περίπτωση αυτή χρειάζεται προσοχή, γιατί μπορεί να παραβιαστεί το τρίτο βασικό χαρακτηριστικό ενός αλγορίθμου, όπως προσδιορίστηκε προηγουμένως, δηλαδή το κριτήριο της καθοριστικότητας.
- **Διαγραμματικές τεχνικές (diagramming techniques) ,** που συνιστούν ένα γραφικό τρόπο παρουσίασης του αλγορίθμου. Από τις διάφορες διαγραμματικές τεχνικές που έχουν επινοηθεί, η πιο παλιά και η πιο γνωστή ίσως, είναι το **διάγραμμα ροής (flow chart)**. Ωστόσο η χρήση διαγραμμάτων ροής για την παρουσίαση αλγορίθμων δεν αποτελεί την καλύτερη λύση, γι' αυτό και εμφανίζονται όλο και σπανιότερα στην πράξη.

Σύμβολα διαγράμματος ροής

Ένα διάγραμμα ροής αποτελείται από ένα σύνολο γεωμετρικών σχημάτων, που το καθένα δηλώνει μία συγκεκριμένη ενέργεια ή λειτουργία. Τα γεωμετρικά αυτά σχήματα ενώνονται μεταξύ τους με βέλη, που δηλώνουν τη σειρά εκτέλεσης των ενεργειών αυτών. Τα κυριότερα χρησιμοποιούμενα σχήματα είναι τα εξής :

Αρχή Τέλος	**Έλλειψη** : παριστάνει την αρχή και το τέλος κάθε αλγορίθμου

Εκτέλεση πράξεων	**Ορθογώνιο:** Δηλώνει επεξεργασία ή εκτέλεση πράξεων
Είσοδος Έξοδος	**Πλάγιο παραλληλόγραμμο** : Δηλώνει την είσοδο των δεδομένων στον αλγόριθμο ή την έξοδο αποτελεσμάτων του.
Συνθήκη	**Ρόμβος** : Ο υπολογισμός της συνθήκης οδηγεί σε δύο διαφορετικές εξόδους ανάλογα με το αν η συνθήκη είναι αληθής ή ψευδής

- **Κωδικοποίηση (coding) ,** δηλαδή με ένα πρόγραμμα που όταν εκτελεστεί θα δώσει τα ίδια αποτελέσματα με τον αλγόριθμο.

ΠΑΡΑΔΕΙΓΜΑ

Έστω ότι μας ζητείται να βρούμε τον μέγιστο μεταξύ τριών αριθμών (έστω Α, Β, Γ). Το παράδειγμα θα επιλυθεί και με τους τέσσερις τρόπους αναπαράστασης αλγορίθμων.

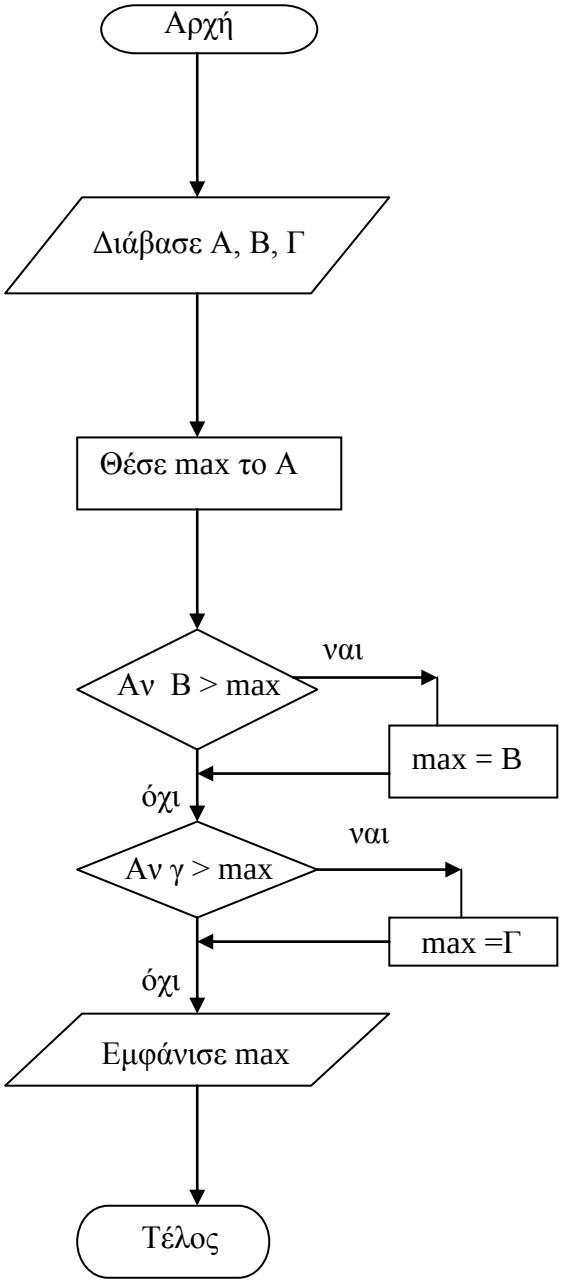
- **Ελεύθερο κείμενο**

Θεώρησε τον πρώτο αριθμό σαν μέγιστο. Στη συνέχεια σύγκρινε τους υπόλοιπους , έναν-έναν, με το μέγιστο. Κάθε φορά που βρίσκεις κάποιον μεγαλύτερο από τον μέγιστο , θέσε μέγιστο αυτόν τον αριθμό. Όταν τελειώσουν οι αριθμοί θα έχεις βρει το μέγιστο.

❖ **Φυσική γλώσσα με βήματα**

1. Θέσε σαν μέγιστο τον Α.
2. Σύγκρινε τον μέγιστο με τον επόμενο αριθμό Β.
3. Αν ο Β είναι μεγαλύτερος, βάλε στο μέγιστο το Β.
4. Σύγκρινε τον Γ με το μέγιστο.
5. Αν ο Γ είναι μεγαλύτερος τότε βάλε στο μέγιστο το Γ.
6. Εμφάνισε το μέγιστο.

❖ **Διαγραμματική τεχνική (Διάγραμμα ροής) *****



**** Το παραπάνω παράδειγμα δεν είναι γραμμένο σύμφωνα με τους κανόνες του ψευδοκώδικα που θα χρησιμοποιηθεί στη συνέχεια, αλλά χρησιμοποιεί λέξεις και σύμβολα που αργότερα θα αντικατασταθούν*

- **Κωδικοποίηση (Ψευδογλώσσα ή Ψευδοκώδικας)**

Αλγόριθμος Ευρεση_Μεγίστου ***
Αρχή
Διάβασε τους Α, Β. Γ
Θέσε max το Α
Αν Β>max τότε θέσε max το Β
Αν Γ>max τότε θέσε max το Γ
Εμφάνισε το max
Τέλος

**** Το παραπάνω παράδειγμα δεν είναι γραμμένο σύμφωνα με τους κανόνες του ψευδοκώδικα που θα χρησιμοποιηθεί στη συνέχεια, αλλά χρησιμοποιεί λέξεις και σύμβολα που αργότερα θα αντικατασταθούν.*

ΔΕΣΜΕΥΜΕΝΕΣ ΛΕΞΕΙΣ

Δεσμευμένες λέξεις ονομάζουμε τις λέξεις που έχουν αυστηρά προκαθορισμένη σημασία και χρησιμεύουν είτε στο να κάνουν πιο ομοιόμορφη τη γλώσσα του κώδικα είτε στο να περιγράψουν τις εντολές της ψευδογλώσσας.

ΣΤΑΘΕΡΕΣ – ΜΕΤΑΒΛΗΤΕΣ

Σταθερές (constants) είναι προκαθορισμένες τιμές που δεν μεταβάλλονται κατά τη διάρκεια της εκτέλεσης του προγράμματος ή αλγορίθμου.
Μεταβλητές (variables) ονομάζονται εκείνα τα μεγέθη που η τιμή τους είναι δυνατόν να αλλάξει κατά τη διάρκεια της εκτέλεσης ενός προγράμματος ή αλγορίθμου.
Σε σχέση με τα ονόματα που μπορούμε να δώσουμε σε μία μεταβλητή ή σταθερά υπάρχουν κάποιοι κανόνες που πρέπει πάντα να εφαρμόζονται. Συγκεκριμένα μια μεταβλητή ή σταθερά παίρνει οποιοδήποτε όνομα (που είναι δυνατόν να περιέχει συνδυασμούς από γράμματα και αριθμούς) αρκεί να εφαρμόζονται τα παρακάτω :
Το όνομα μιας μεταβλητής ή σταθεράς

- ✓ ξεκινάει πάντα με γράμμα (λατινικού ή ελληνικού αλφαβήτου)
- ✓ δεν πρέπει να είναι δεσμευμένη λέξη
- ✓ δεν πρέπει να έχει κενά (π.χ. μέσος όρος ΛΑΘΟΣ). Στην περίπτωση αυτή χρησιμοποιούμε συνήθως την κάτω παύλα _ (π.χ. μέσος_όρος)

ΤΥΠΟΙ ΔΕΔΟΜΕΝΩΝ

Οι τύποι δεδομένων που υποστηρίζει η ΓΛΩΣΣΑ είναι το σύνολο των τιμών που μπορεί να πάρει μία σταθερά ή μία μεταβλητή. Υπάρχουν τέσσερις διαφορετικοί τύποι που υποστηρίζονται από τη Γλώσσα:

- **Ακέραιος τύπος** : Ο τύπος αυτός περιλαμβάνει τους ακέραιους αριθμούς που είναι γνωστοί από τα μαθηματικά.
- **Πραγματικός τύπος :** Ο τύπος αυτός περιλαμβάνει τους επίσης γνωστούς από τα μαθηματικά πραγματικούς αριθμούς.
- **Χαρακτήρας (ή αλφαριθμητικός)** : Ο τύπος αυτός αναφέρεται τόσο σε έναν χαρακτήρα όσο και σε μία σειρά χαρακτήρων. Τα δεδομένα αυτού του τύπου μπορούν να περιέχουν οποιοδήποτε χαρακτήρα παράγεται από το πληκτρολόγιο. Οι χαρακτήρες σε έναν αλγόριθμο βρίσκονται υποχρεωτικά ανάμεσα σε εισαγωγικά ‘ … ’ . Για παράδειγμα μπορούμε να αναφέρουμε ‘Κ’, ‘Κώστας’, ‘Τα πολλαπλάσια του 5 είναι’.
- **Λογικός τύπος :** Αυτός ο τύπος δέχεται μόνο δύο δυνατές τιμές . Οι τιμές αυτές είναι ΑΛΗΘΗΣ και ΨΕΥΔΗΣ.

ΑΡΙΘΜΗΤΙΚΟΙ ΤΕΛΕΣΤΕΣ

Οι τελεστές που χρησιμοποιούνται είναι οι ακόλουθοι:

Αριθμητικός Τελεστής	**Πράξη**
+	Πρόσθεση
-	Αφαίρεση
*	Πολλαπλασιασμός
/	Διαίρεση
^	Ύψωση σε δύναμη
DIV	Ακέραια διαίρεση
MOD	Υπόλοιπο ακέραιας διαίρεσης

Οι παραπάνω τελεστές μας βοηθούν στην αναπαράσταση των διάφορων αριθμητικών εκφράσεων. Η προτεραιότητα των πράξεων (σε περίπτωση που δεν υπάρχουν βέβαια παρενθέσεις που πάντα προηγούνται) είναι η γνωστή από τα μαθηματικά. Δηλαδή **Δυνάμεις → Πολλαπλασιασμός \ Διαίρεση \ div \ mod → Πρόσθεση \ Αφαίρεση.**

Έτσι για παράδειγμα η μαθηματική έκφραση $\frac{3A-8B^3}{4+CD}$ σε έναν αλγόριθμο γράφεται :

(3*A-8*B^3) / (4+C*D). Σε τέτοιες περιπτώσεις χρειάζεται ιδιαίτερη προσοχή στο πως θα χρησιμοποιήσουμε τις παρενθέσεις !!! Έτσι σε περίπτωση που στον

αλγόριθμό μας γράφαμε 3*A-8*B^3 / 4+C*D στα μαθηματικά θα σήμαινε $3A - \frac{8B^3}{4} + CD$.

Η πράξη div εκφράζει το πηλίκο της ακέραιας διαίρεσης δύο αριθμών. Για παράδειγμα 12 div 5 =2 , 15div2 = 7 , ενώ η πράξη mod εκφράζει το υπόλοιπο της ακέραιας διαίρεσης δύο αριθμών, π.χ. 7 mod 3 = 1, 6 mod 3 =0.

Ιδιαίτερη προσοχή πρέπει να δοθεί στην πράξη mod με τη βοήθεια της οποίας ελέγχουμε αν ένας ακέραιος αριθμός είναι άρτιος ή περιττός (πολύ χρήσιμο στη συνέχεια !!!!!) Για να γίνει αυτό παίρνουμε το αποτέλεσμα της διαίρεσης του αριθμού με το 2. Στην περίπτωση που το υπόλοιπο είναι 0 ο αριθμός είναι άρτιος, ενώ αν το υπόλοιπο είναι 1 ο αριθμός είναι περιττός

ΔΕΣΜΕΥΜΕΝΕΣ ΣΥΝΑΡΤΗΣΕΙΣ

Πολλές γνωστές συναρτήσεις από τα μαθηματικά είναι ενσωματωμένες στη Γλώσσα. Οι συναρτήσεις αυτές είναι :

Συνάρτηση	**Σημασία**
ΗΜ(Χ)	Υπολογισμός Ημιτόνου
ΣΥΝ(Χ)	Υπολογισμός Συνημιτόνου
ΕΦ(Χ)	Υπολογισμός Εφαπτομένης
Τα_Ρ(Χ)	Υπολογισμός Τετραγωνικής ρίζας
ΛΟΓ(Χ)	Υπολογισμός Φυσικού Λογάριθμου
Ε(Χ)	Υπολογισμός του e^x
Α_Μ(Χ)	Ακέραιο μέρος του Χ
Α_Τ(Χ)	Απόλυτη Τιμή του Χ

ΕΝΤΟΛΕΣ ΕΙΣΟΔΟΥ _ ΕΞΟΔΟΥ

Τα περισσότερα προγράμματα ή αλγόριθμοι δέχονται κάποια δεδομένα με σκοπό την επεξεργασία τους και τον υπολογισμό αποτελεσμάτων. Τα δεδομένα εισάγονται κατά τη διάρκεια εκτέλεσης του αλγορίθμου από μία συσκευή εισόδου (συνήθως το πληκτρολόγιο) και τα αποτελέσματα εμφανίζονται σε μία μονάδα εξόδου του υπολογιστή (συνήθως στην οθόνη). Για την εισαγωγή δεδομένων χρησιμοποιείται η

δεσμευμένη λέξη **ΔΙΑΒΑΣΕ**. Η σύνταξη της παραπάνω εντολής στη γενική μορφή είναι η εξής :

Διάβασε ονομ_μεταβλητή_1 , ονομ_μεταβλητή_2, ……,ονομ_μεταβλητή_N

Για παράδειγμα

Διάβασε α, β

Με αυτή την εντολή διακόπτεται η ροή εκτέλεσης του αλγορίθμου, μέχρι τη στιγμή που θα ολοκληρωθεί η εισαγωγή δεδομένων. Συνήθως η εισαγωγή γίνεται από το πληκτρολόγιο. Έτσι μετά την εντολή Διάβασε οι τιμές των μεταβλητών α και β θα είναι αυτές που θα πληκτρολογήσει ο χρήστης.

Η έξοδος των αποτελεσμάτων σε έναν αλγόριθμο υλοποιείται με μία από τις εντολές **Εμφάνισε , Τύπωσε** (ή εκτύπωσε) και **Γράψε .** Η εμφάνιση των αποτελεσμάτων μπορεί να συνοδευτεί και από ανάλογο μήνυμα στο οποίο μπορούμε να γράψουμε ότι θέλουμε αλλά είναι υποχρεωτικό να περικλείεται σε εισαγωγικά ΄…΄. Στην περίπτωση που θέλουμε να εμφανίζεται μαζί με το μήνυμα και η τιμή μιας μεταβλητής ή σταθεράς πρέπει το όνομα της μεταβλητής (ή της σταθεράς) να περικλείεται σε κόμματα και να είναι έξω από τα εισαγωγικά. Η γενική σύνταξη είναι

Εμφάνισε λίστα_στοιχείων

Για παράδειγμα

Εμφάνισε ‘ Η τετραγωνική ρίζα του’, Α,’ είναι ‘, Ρ

ΕΝΤΟΛΗ ΕΚΧΩΡΗΣΗΣ ΤΙΜΗΣ

Η εντολή εκχώρησης είναι η εντολή που μας επιτρέπει να καταχωρήσουμε μία τιμή σε μία μεταβλητή. Κατά την καταχώρηση αυτή η μεταβλητή χάνει το περιεχόμενο που είχε πριν. Συμβολίζεται με το ανάποδο βέλος ← . Αριστερά του συμβόλου υπάρχει το όνομα **μίας μόνο μεταβλητής** ενώ δεξιά του μπορεί να υπάρχει αριθμός, μεταβλητή, συμβολοσειρά (χαρακτήρες) ή και μία ολοκληρωμένη μαθηματική παράσταση. Σε μία εντολή εκχώρησης είναι δυνατόν η παράσταση της οποίας η τιμή εκχωρείται σε μία μεταβλητή, να περιέχει τη μεταβλητή αυτή. Αυτό σημαίνει ότι εκχωρεί στη μεταβλητή μία τιμή που εξαρτάται από την τρέχουσα τιμή της. **Προσοχή** η εντολή εκχώρησης ← δεν έχει καμία σχέση με τη μαθηματική ισότητα. π.χ στα μαθηματικά η έκφραση Α = Α +.10 είναι αδύνατη αφού καταλήγει στην αδύνατη εξίσωση 0 = 10 , ενώ αν σε έναν αλγόριθμο έχουμε την παρακάτω εντολή Α ← Α+10 σημαίνει ότι στην προηγούμενη τιμή της μεταβλητής Α πρόσθεσε το 10 και το αποτέλεσμα «τοποθέτησέ» το ξανά στη μεταβλητή Α. ακολουθούν μερικά παραδείγματα εκχώρησης τιμών σε μεταβλητές.

Α ← 6 : Καταχώρησε το 6 στην μεταβλητή Α

Β ← Α +2 : Καταχώρησε στο Β την τιμή της μεταβλητής Α αφού πρώτα του προσθέσεις το 2

Γ ← Α + Β : Καταχώρησε στη μεταβλητή Γ το άθροισμα των τιμών των μεταβλητών Α και Β

Όνομα ← ‘ Μαρία’ : Στη μεταβλητή Όνομα (προφανώς μεταβλητή που ο τύπος της είναι χαρακτήρων) καταχώρησε τη συμβολοσειρά (λέξη) ‘Μαρία’

Α ← Α – Β : Στη μεταβλητή Α καταχώρησε την προηγούμενη τιμή της μείον την τιμή της μεταβλητής Β.

Δύο σημαντικά σημεία που πρέπει να προσέχουμε σχετικά με τις εντολές εκχώρησης είναι τα ακόλουθα :

- Όλες οι μεταβλητές του δεξιού μέρους πρέπει να έχουν τιμή.
- Τα αποτελέσματα στο τέλος του αλγορίθμου πρεπει να έχουν τιμές σαν αποτέλεσμα εκχώρησης.

Τέλος αξίζει να σημειωθεί ότι αν σε μία ομάδα εντολών υπάρχουν περισσότερες από μία εκχωρήσεις στην ίδια μεταβλητή, τότε η τελική της τιμή είναι αυτή που εκχωρείται τελευταία. Π.χ.

Α ←1289
Α ← Α-1000
Α ← 0

Η τιμή που έχει η μεταβλητή Α μετά την εκτέλεση των παραπάνω εντολών εκχώρησης είναι 0.

ΛΟΓΙΚΕΣ ΣΥΝΘΗΚΕΣ – ΤΕΛΕΣΤΕΣ ΣΥΓΚΡΙΣΗΣ – ΛΟΓΙΚΕΣ ΠΡΑΞΕΙΣ

Όπως αναφέρθηκε και παραπάνω σε πολλούς αλγορίθμους μας ζητείται η λήψη μιας απόφασης .Η απόφαση αυτή εξαρτάται από το αποτέλεσμα μιας συνθήκης (αληθής ή ψευδής). Για τον καθορισμό των συνθηκών που συχνά συναντάμε μας είναι χρήσιμοι οι τελεστές σύγκρισης, οι οποίοι είναι οι γνωστοί από τα μαθηματικά αλλά στη Γλώσσα ο συμβολισμός που χρησιμοποιείται είναι ο ακόλουθος

Τελεστής	**Έννοια**
=	Ισότητα
<	Μικρότερο
>	Μεγαλύτερο
<=	Μικρότερο ή ίσο
>=	Μεγαλύτερο ή ίσο
<>	Διάφορο

Επειδή σε πολλές περιπτώσεις είμαστε αναγκασμένοι να εκφράσουμε πιο σύνθετες συνθήκες από αυτές που είναι δυνατόν να εκφραστούν με τους παραπάνω τελεστές πολύ συχνά χρησιμοποιούμε και τις βασικές πράξεις από την άλγεβρα Boole που είναι :

- ✓ **Άρνηση** . Η λογική πράξη άρνηση υλοποιείται με τη δεσμευμένη λέξη **Όχι (not)** .Το αποτέλεσμα της είναι το αντίστροφο της αρχικής λογικής τιμής, δηλαδή αν η λογική συνθήκη είναι αληθής η άρνησή της είναι ψευδής και αντιστρόφως.
- ✓ **Σύζευξη.** Υλοποιείται με τη χρήση της δεσμευμένης λέξης **και (and**) . Το αποτέλεσμα της σύζευξης δύο συνθηκών είναι αληθές μόνο όταν και οι δύο συνθήκες είναι αληθείς.
- ✓ **Διάζευξη .** Υλοποιείται με τη βοήθεια της δεσμευμένης λέξης **ή (or)**. . Το αποτέλεσμα της διάζευξης δύο συνθηκών είναι αληθές όταν τουλάχιστον μία από τις δύο συνθήκες είναι αληθής.

Τα παραπάνω περιγράφονται συνοπτικά από τον πίνακα που ακολουθεί.

Συνθήκη Α	**Συνθήκη Β**	**Α ή Β**	**Α και Β**	**Όχι Α**
Ψευδής	Ψευδής	Ψευδής	Ψευδής	Αληθής
Ψευδής	Αληθής	Αληθής	Ψευδής	Αληθής
Αληθής	Ψευδής	Αληθής	Ψευδής	Ψευδής
Αληθής	Αληθής	Αληθής	Αληθής	Ψευδής

Στην περίπτωση που χρησιμοποιούμε παραπάνω από μία λογικές πράξεις (ή συνδυασμό λογικών πράξεων) η προτεραιότητα που ακολουθείται (αν βέβαια δεν υπάρχουν παρενθέσεις που πάντα προηγούνται) είναι **όχι → και → ή.**

ΣΧΟΛΙΑ

Τα σχόλια χρησιμοποιούνται σαν επεξήγηση για τους αναγνώστες του αλγορίθμου ή του προγράμματος. Δεν αποτελούν εντολές, συνεπώς δεν παίζουν κανένα ρόλο στην εκτέλεση των προγραμμάτων, απλά τα κάνουν πιο ευανάγνωστα. Το δεσμευμένο σύμβολο της Γλώσσας για την εισαγωγή σχολίων είναι το ! ακολουθούμενο από το κείμενο της επεξήγησης .

ΔΟΜΗ ΑΛΓΟΡΙΘΜΟΥ - ΠΡΟΓΡΑΜΜΑΤΟΣ

Η πρώτη εντολή σε έναν αλγόριθμο – πρόγραμμα είναι η δεσμευμένη λέξη **Αλγόριθμος (ή Πρόγραμμα)** ακολουθούμενη από το όνομα του αλγορίθμου. Το όνομα του αλγορίθμου μπορεί να είναι οποιοδήποτε αρκεί να ξεκίνα πάντα από γράμμα (ελληνικό ή λατινικό), από εκεί και πέρα μπορεί να περιέχει αριθμούς και το μοναδικό σύμβολο που μπορεί να περιέχει είναι η κάτω παύλα (_). Τέλος το όνομα δεν μπορεί να είναι δεσμευμένη λέξη.

Π.χ.

Αλγόριθμος όνομα_αλγορίθμου
Πρόγραμμα όνομα_προγράμματος

Κατόπιν ακολουθεί το **τμήμα δηλώσεων** των μεταβλητών και των σταθερών (αν υπάρχουν). Πρώτα είμαστε υποχρεωμένοι να δηλώσουμε τις σταθερές με τη χρήση της δεσμευμένης λέξης **σταθερές**. Αμέσως μετά τα γράφουμε τα ονόματά τους ακολουθεί το σύμβολο = και στη συνέχεια η τιμή που έχει καθεμία από αυτές. Στη συνέχεια ακολουθεί η δήλωση των μεταβλητών με τη βοήθεια της δεσμευμένη λέξης **μεταβλητές.** Πρώτα δηλώνουμε τους τύπους τους (το σύνολο μέσα από το οποίο παίρνουν τιμές) μετά το σύμβολο : και τέλος το όνομά τους.

Π.χ.

Σταθερές
Σταθερα_1 = τιμή_1
Σταθερά_2 = τιμή_2
.
.
Μεταβλητές
Τύπος_1 : ονομα_1,ονομα_2,…..,όνομα_N
Τύπος_2 : ονομα_1,ονομα_2,…..,όνομα_N
Τύπος_3 : ονομα_1,ονομα_2,…. ,όνομα_N
Τύπος_4 : ονομα_1,ονομα_2,…. ,όνομα_N

Στη συνέχεια ακολουθεί το κύριο μέρος του αλγορίθμου (προγράμματος) που περιέχει τις εκτελέσιμες εντολές. Το κύριο μέρος ξεκινά πάντα με τη δεσμευμένη λέξη **Αρχή** και τελειώνει με τη δεσμευμένη λέξη **Τέλος** ακολουθούμενη από το όνομα του αλγορίθμου (Για την περίπτωση του προγράμματος τελειώνουμε με τη δεσμευμένη λέξη **Τέλος_προγράμματος** ακολουθούμενη από το όνομα του προγράμματος)

Π.χ.
Αρχή
Ομάδα_εντολών
Τέλος όνομα_αλγορίθμου

ή

_ Αρχή
Ομάδα_εντολών
Τέλος_προγράμματος όνομα προγράμματος

Έτσι η γενική μορφή ενός αλγορίθμου (ή προγράμματος) είναι:

Αλγόριθμος όνομα_αλγορίθμου

Σταθερές
Σταθερα_1 = τιμή_1
Σταθερά_2 = τιμή_2

Μεταβλητές
Τύπος_1 : ονομα_1,ονομα_2,…..,όνομα_N
Τύπος_2 : ονομα_1,ονομα_2,…..,όνομα_N
Τύπος_3 : ονομα_1,ονομα_2,…. ,όνομα_N
.Τύπος_4 : ονομα_1,ονομα_2,…. ,όνομα_N

Αρχή
Ομάδα_εντολών
Τέλος όνομα_αλγορίθμου

Πρόγραμμα όνομα_προγράμματος

Σταθερές
Σταθερα_1 = τιμή_1
Σταθερά_2 = τιμή_2

Μεταβλητές
Τύπος_1 : ονομα_1,ονομα_2,…..,όνομα_N
Τύπος_2 : ονομα_1,ονομα_2,…..,όνομα_N
Τύπος_3 : ονομα_1,ονομα_2,…. ,όνομα_N
.Τύπος_4 : ονομα_1,ονομα_2,…. ,όνομα_N

Αρχή
Ομάδα_εντολών
Τέλος_προγράμματος όνομα προγράμματος

ΔΟΜΗ ΕΠΙΛΟΓΗΣ

Δεν είναι λίγες οι περιπτώσεις όπου για τη δημιουργία αλγορίθμου επίλυσης ενός προβλήματος μια ομάδα από εντολές πρέπει να εκτελεστεί κάτω από ορισμένες προϋποθέσεις. Αυτή τη δυνατότητα μας την παρέχει η δομή επιλογής. Στις περιπτώσεις που χρησιμοποιείται αυτή η δομή ακολουθείται η παρακάτω σειρά ενεργειών : Αρχικά εξετάζεται μία λογική συνθήκη. Στη συνέχεια, και ανάλογα με το αποτέλεσμα της συνθήκης αυτής (δύο είναι τα πιθανά αποτελέσματα αληθής ή ψευδής) εκτελείται και η αντίστοιχη ομάδα εντολών. Στη συνέχεια παρουσιάζονται και αναλύονται όλες οι πιθανές μορφές σύνταξης της δομής επιλογής.

Απλή επιλογή (Αν…τότε…..τέλος_αν)

Η σύνταξη της δομής της απλής επιλογής είναι η ακόλουθη:

Αν <συνθήκη> **τότε**
Ομάδα_εντολών
Τέλος_αν

Ο τρόπος λειτουργίας της παραπάνω δομής είναι ο ακόλουθος. Στην περίπτωση όπου η συνθήκη είναι αληθής τότε εκτελείται η ομάδα_εντολών. Σε διαφορετική περίπτωση η ομάδα_εντολών αγνοείται και η εκτέλεση του αλγορίθμου συνεχίζεται κανονικά. Το διάγραμμα ροής για τη δομή της απλής επιλογής φαίνεται στο παρακάτω σχήμα.

Παράδειγμα Ι

Να γράψετε αλγόριθμο ο οποίος να δέχεται τη βαθμολογία ενός μαθητή Λυκείου και αν αυτή είναι μεγαλύτερη ή ίση από 10 να του εμφανίζει στην οθόνη το μήνυμα 'συγχαρητήρια πέρασες ' .

Αλγόριθμος βαθμολογία
Μεταβλητές
Πραγματικές : β

Αρχή
Διάβασε β
Αν β>10 τότε
Εμφάνισε 'Συγχαρητήρια πέρασες'
Τέλος_αν
Τέλος βαθμολογία

<u>Παράδειγμα ΙΙ *</u>

Να σχεδιάσετε αλγόριθμο ο οποίος να δέχεται σαν είσοδο τρεις πραγματικούς αριθμούς και να υπολογίζει το μέγιστο από αυτούς.

Σε προβλήματα στα οποία απαιτείται ο υπολογισμός του μεγίστου ή του ελαχίστου μεταξύ κάποιων αριθμών ακολουθείται μία συγκεκριμένη στρατηγική. Αρχικά ορίζουμε μία μεταβλητή max (ή min) και θεωρούμε τον πρώτο από τους αριθμούς σαν μέγιστο (ή ελάχιστο). Στη συνέχεια παίρνουμε εναν-έναν τους υπόλοιπους αριθμούς και τους συγκρίνουμε με αυτόν που έχουμε θέσει μέγιστο (ελάχιστο). Σε περίπτωση που ο αριθμός που εξετάζεται είναι μεγαλύτερος (μικρότερος) από το μέγιστο (ελάχιστο) τότε θέτουμε αυτόν μέγιστο (ελάχιστο) και συνεχίζουμε διαδοχικά τις συγκρίσεις μέχρι να τελειώσουν οι αριθμοί. Όταν πραγματοποιηθεί και η τελευταία σύγκριση τότε το περιεχόμενο της μεταβλητής max (min) θα περιέχει τη μέγιστη (ελάχιστη) τιμή. Ας δούμε τώρα αυτή τη μεθοδολογία που περιγράψαμε εκφρασμένη αλγοριθμικά.

Αλγόριθμος Ευρεση_μεγίστου_αριθμού
Μεταβλητές
Πραγματικές : α, β, γ , max
Αρχή
Διάβασε α, β, γ
Max ← α
Αν β > max τότε
 Max ← β
Τέλος_αν
Αν γ > max τότε
 Max ← γ
Τέλος_αν
Τύπωσε ‘ Ο μέγιστος αριθμός είναι‘ , max
Τέλος Ευρεση_μεγίστου_αριθμού

**Ο αλγόριθμος είναι δυνατόν να δημιουργηθεί και με άλλο τρόπο που θα παρουσιαστεί πιο κάτω*

Σύνθετη επιλογή (αν…..τότε……αλλιώς…….τέλος_αν)
Εμφωλευμένη επιλογή

Η σύνταξη της σύνθετης δομής επιλογής φαίνεται παρακάτω :

Αν <συνθήκη> **τότε**
ομάδα εντολών 1
αλλιώς
ομάδα εντολών 2
τέλος_αν

Η δομή της σύνθετης επιλογής εκτελείται με τον εξής τρόπο λειτουργίας. Αρχικά ελέγχεται η συνθήκη. Αν η συνθήκη είναι αληθής τότε εκτελείται μόνο η ομάδα εντολών 1, ενώ στην περίπτωση που η συνθήκη είναι ψευδής τότε εκτελείται μόνο η ομάδα εντολών 2.

Σε πολλούς αλγορίθμους είναι πολύ πιθανόν η ομάδα εντολών 1 ή η ομάδα εντολών 2 να περιέχουν μέσα τους μία ή και περισσότερες δομές επιλογής. Σε τέτοιες περιπτώσεις έχουμε τη δομή της εμφωλευμένης επιλογής, η οποία ονομάζεται έτσι

γιατί μια ή περισσότερες δομές επιλογής περιέχονται μέσα σε μια άλλη δομή επιλογής. Ένας τυπικός τρόπος σύνταξης μιας εμφωλευμένης δομής επιλογής μέσα σε κάποια άλλη φαίνεται παρακάτω :

Αν <συνθήκη 1> **τότε**
Ομάδα εντολών 1
αλλιώς
    αν <συνθήκη 2> **τότε**
    Ομάδα εντολών 2
    αλλιώς
    Ομάδα εντολών 3
    τέλος_αν
Τέλος_αν

Στον παραπάνω αλγόριθμο η **ομάδα εντολών 1** θα εκτελεστεί αν η **συνθήκη 1** είναι αληθής. Αν η πρώτη συνθήκη είναι ψευδής τότε ελέγχεται η **συνθήκη 2** και αναλόγως με την τιμή της εκτελείται, είτε η **ομάδα εντολών 2** (όταν η συνθήκη 2 είναι αληθής), ή η **ομάδα εντολών 3** (όταν η συνθήκη 2 είναι ψευδής)

Εδώ πρέπει να σημειωθεί ότι η ψευδογλώσσα μας δίνει και έναν εναλλακτικό τρόπο έκφρασης της εμφωλευμένης επιλογής, με τη χρήση της δεσμευμένης λέξης **αλλιώς_αν.** Η σύνταξη για ένα παράδειγμα όπως το προηγούμενο θα ήταν :

Αν <συνθήκη 1> **τότε**
Ομάδα εντολών 1
Αλλιώς_αν <συνθήκη 2> **τότε**
Ομάδα εντολών 2
αλλιώς
Ομάδα εντολών 3
Τέλος_αν

Παρατηρούμε ότι η σύνταξη του αλγορίθμου είναι σχεδόν ίδια με προηγουμένως. Η μοναδική διαφορά είναι ότι η δομή αυτή κλείνει **μόνο με ένα τέλος_αν,** σε αντίθεση με τον πρώτο τρόπο σύνταξης όπου για κάθε μία εντολή αν που χρησιμοποιείται πρέπει να κλείνει με το αντίστοιχο τέλος_αν

<u>Παράδειγμα ΙΙΙ</u>

Να κατασκευάσετε έναν αλγόριθμο ο οποίος θα δέχεται έναν ακέραιο αριθμό ν, θα υπολογίζει και θα εμφανίζει την τιμή της παράστασης $(-1)^{ν}$

Όπως γνωρίζουμε δύο είναι τα πιθανά αποτελέσματα για τη συγκεκριμένη παράσταση (1 ή -1) το οποίο εξαρτάται από το αν ο εκθέτης ν είναι άρτιος ή περιττός.

```
Αλγόριθμος Παράσταση
Μεταβλητές
Ακέραιες : Χ, ν
Αρχή
Διάβασε ν
Αν ν mod 2= 0 τότε
Χ ← 1
Αλλιώς
Χ ← -1
Τέλος_αν
Εμφάνισε 'Η τιμή της παράστασης είναι ', Χ
Τέλος Παράσταση
```

<u>Παράδειγμα IV</u>

Να γράψετε αλγόριθμο που θα δέχεται τους συντελεστές α και β της πρωτοβάθμιας εξίσωσης $\alpha x + \beta = 0$ και να υπολογίζει τις ρίζες της.

```
Αλγόριθμος πρωτοβάθμια
Μεταβλητές
Πραγματικές : α, β,x

Αρχή
Διάβασε α ,β
Αν  α=0 τότε
   αν β=0 τότε
     εμφάνισε ' ΑΟΡΙΣΤΗ '
   αλλιώς
     εμφάνισε 'ΑΔΥΝΑΤΗ'
   τέλος_αν
αλλιώς
x ← - β / α
εμφάνισε 'Η λύση της εξίσωσης είναι ', x
τέλος_αν
τέλος πρωτοβάθμια
```

<u>Παράδειγμα V</u>

Να γράψετε αλγόριθμο ο οποίος να δέχεται σαν είσοδο το βαθμό πτυχίου ενός φοιτητή και να τυπώνει το αντίστοιχο μήνυμα όπως φαίνεται στον παρακάτω πίνακα.

Βαθμός πτυχίου	***Χαρακτηρισμός***
Από 5 έως και 6,5	ΚΑΛΩΣ
Από 6,51 έως και 8,5	ΛΙΑΝ ΚΑΛΩΣ
Από 8,51 και πάνω	ΑΡΙΣΤΑ

1ος τρόπος

```
Αλγόριθμος βαθμός_πτυχίου_1
Μεταβλητές
Πραγματικές : Β
Αρχή
Διάβασε Β
Αν Β<= 6,5 τότε
   Εμφάνισε 'ΚΑΛΩΣ'
Αλλιώς
   αν Β<= 8,5 τότε
    εμφάνισε 'ΛΙΑΝ ΚΑΛΩΣ'
   αλλιώς
     εμφάνισε 'ΑΡΙΣΤΑ'
   τέλος_αν
Τέλος_αν

Τέλος βαθμός_πτυχίου_1
```

2ος τρόπος

```
Αλγόριθμος βαθμός_πτυχίου_2
Μεταβλητές
Πραγματικές : Β
Αρχή
Διάβασε Β
Αν Β<= 6,5 τότε
   Εμφάνισε 'ΚΑΛΩΣ'
Αλλιώς_αν Β<= 8,5 τότε
   Εμφάνισε 'ΛΙΑΝ ΚΑΛΩΣ'
 αλλιώς
   Εμφάνισε 'ΑΡΙΣΤΑ'
Τέλος_αν
Τέλος βαθμός_πτυχίου_2
```

<u>Παράδειγμα VI</u>

Να γράψετε αλγόριθμο που θα επιλύει πλήρως τη δευτεροβάθμια εξίσωση όταν διαβάζει από το χρήστη τους συντελεστές της (α, β, γ)

```
Αλγόριθμος Δευτεροβάθμια_εξίσωση
Μεταβλητές
Πραγματικές : α, β, γ , Δ, Χ, Χ1, Χ2

αρχή
εμφάνισε 'δώστε τους συντελεστές της δευτεροβάθμιας'
διάβασε α, β, γ
Αν α=0 τότε
    Αν β=0 τότε
        Αν γ =0 τότε
         εμφάνισε 'ΑΟΡΙΣΤΗ'
        αλλιώς
          εμφάνισε 'ΑΔΥΝΑΤΗ'
        τέλος_αν
    αλλιώς
     Χ ← - γ/β
    τέλος_αν
αλλιώς
Δ ← β^2 – 4*α*γ
    Αν Δ>0 τότε
     Χ1← (-β + Τ_Ρ(Δ))/(2*α)
     Χ2← (-β - Τ_Ρ(Δ))/(2*α)
     εμφάνισε 'Οι ρίζες της εξίσωσης είναι : ', Χ1 , ' και ' , Χ2
    αλλιώς
          αν Δ=0  τότε
            Χ← -β/(2*α)
            Εμφάνισε ' Η εξίσωση έχει μία διπλή ρίζα την ', Χ
           αλλιώς
            εμφάνισε 'Η εξίσωση δεν έχει πραγματικές λύσεις'
           τέλος_αν
    τέλος_αν
τέλος_αν

τέλος Δευτεροβάθμια
```

<u>Παράδειγμα VII</u>

Η δημόσια εταιρία ύδρευσης με σκοπό την αποφυγή της άσκοπης κατανάλωσης νερού και την καταπολέμηση της λειψυδρίας ακολουθεί την παρακάτω πολιτική χρέωσης : Τα πρώτα 1000 κυβικά νερού κάθε καταναλωτής τα χρεώνεται με 0,5€/κυβικό, τα επόμενα 1000 (από 1001 έως 2000 κυβικά) χρεώνονται προς 0,7€/κυβικό και για μεγαλύτερες ποσότητες (από 2001 κυβικά και πάνω) κυβικά νερού η χρέωση είναι 1€/κυβικό. Έτσι για παράδειγμα αν κάποιος καταναλώσει 1362 κυβικά θα κληθεί να πληρώσει (1000*0,5) +(362*0,7). Να γραφεί αλγόριθμος που θα δέχεται την κατανάλωση νερού και αφού υπολογίζει να εμφανίζει το λογαριασμό.

Τέτοιας μορφής ασκήσεις είναι οι λεγόμενες κλιμακωτές οι οποίες έχουν έναν συγκεκριμένο τρόπο επίλυσης.

```
Αλγόριθμος λογαριασμός_νερού
Μεταβλητές
Πραγματικές : κ , Χ  ! κ συμβολίζουμε την κατανάλωση νερού σε κυβικά και Χ
                     ! το λογαριασμό

Αρχή
Διάβασε κ

Αν κ <= 1000 τότε
Χ ← κ *0,5
Αλλιώς
    Αν κ <= 2000 τότε
     Χ ← 1000*0,5 + (κ-1000) * 0,7
    Αλλιώς
     Χ ← 1000*0,5 + 1000*0.7  +(κ-2000) * 1
    Τέλος_αν
Τέλος_αν
 Εμφάνισε ‘ Ο λογαριασμός είναι ’ ,Χ, ‘€’
Τέλος λογαριασμός_νερού
```