

ΑΛΓΟΡΙΘΜΟΙ

Στο σημείωμα αυτό αρχικά εξηγείται η έννοια αλγόριθμος και παραθέτονται τα σπουδαιότερα κριτήρια που πρέπει να πληρεί κάθε αλγόριθμος. Στη συνέχεια, η σπουδαιότητα των αλγορίθμων συνδυάζεται με την εξέλιξη της επιστήμης της Πληροφορικής. Ακολούθως γίνεται μία αξιολόγηση των αλγορίθμων και τελευταία γίνεται μια εισαγωγή στις σπουδαιότερες δομές δεδομένων και τις αντίστοιχες πράξεις που μπορούμε να κάνουμε με αυτές.

Τι είναι αλγόριθμος

Ο όρος αλγόριθμος χρησιμοποιείται για να δηλώσει μεθόδους που εφαρμόζονται για την επίλυση προβλημάτων. Ωστόσο υπάρχει ένας πιο αυστηρός ορισμός της έννοιας αυτής που είναι ο εξής:

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

Κάθε αλγόριθμος απαραίτητα ικανοποιεί τα επόμενα κριτήρια:

- **Είσοδος (input).** Καμία, μία ή και 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 των συναρτήσεων παραγωγής τυχαίων αριθμών, ή με την 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 τη 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 αλγορίθμου να είναι απλή. Αυτό σημαίνει ότι μια εντολή δεν αρκεί να έχει ορισθεί, αλλά πρέπει να είναι και εκτελέσιμη.

Σπουδαιότητα αλγορίθμων

Η πληροφορική μπορεί να ορισθεί ως η επιστήμη που μελετά τους αλγόριθμους από τις ακόλουθες σκοπιές:

- **Υλικό (hardware).** Η ταχύτητα εκτέλεσης ενός αλγορίθμου επηρεάζεται από τις διάφορες τεχνολογίες υλικού, δηλαδή από τον τρόπο που είναι δομημένα σε μία ενιαία αρχιτεκτονική τα διάφορα συστατικά του υπολογιστή.
- **Γλώσσες Προγραμματισμού(programming languages).** Το είδος της γλώσσας προγραμματισμού που χρησιμοποιείται (χαμηλότερου ή υψηλότερου επιπέδου) αλλάζει τη δομή και τον αριθμό των εντολών ενός αλγορίθμου.
- **Θεωρητική(theoretical).** Το ερώτημα που συχνά τίθεται είναι, αν πραγματικά υπάρχει ή όχι κάποιος αποδοτικός αλγόριθμος για την επίλυση ενός προβλήματος. Η προσέγγιση αυτή είναι ιδιαίτερα σημαντική γιατί προσδιορίζει τα όρια της λύσης που θα βρεθεί σε σχέση με ένα συγκεκριμένο πρόβλημα.
- **Αναλυτική(analytical).** Μελετώνται οι υπολογιστικοί πόροι (**computer resources**) που απαιτούνται από ένα αλγόριθμο όπως για παράδειγμα το μέγεθος της κύριας μνήμης και της δευτερεύουσας μνήμης, ο χρόνος για λειτουργίες της CPU και για τις λειτουργίες εισόδου/εξόδου.

Σύμφωνα με όσα αναφέρθηκαν ο αλγόριθμος αποσκοπεί στην επίλυση ενός προβλήματος. Έτσι είναι απαραίτητο να γίνεται μια καλή ανάλυση του κάθε προβλήματος και να προτείνεται συγκεκριμένη μεθοδολογία και ακολουθία βημάτων. Βασικός στόχος μας είναι η πρόταση έξυπνων και αποδοτικών λύσεων.

Η ανάλυση ενός προβλήματος σε ένα σύγχρονο υπολογιστικό περιβάλλον περιλαμβάνει:

- την καταγραφή της υπάρχουσας πληροφορίας για το πρόβλημα,
- την αναγνώριση των ιδιοτήτων του προβλήματος,
- την αποτύπωση των συνθηκών και προϋποθέσεων υλοποίησης του,
- τη πρόταση επίλυσης του με χρήση κάποιας μεθόδου, και
- την τελική επίλυση με χρήση υπολογιστικών συστημάτων.

Έτσι κατά την ανάλυση ενός προβλήματος θα πρέπει να δοθεί απάντηση σε κάθε μια από τις επόμενες ερωτήσεις:

1. Ποια είναι τα δεδομένα του προβλήματος,
2. Ποιες είναι οι συνθήκες που πρέπει να πληρούνται για την επίλυση του προβλήματος,
3. Ποια είναι η πλέον αποδοτική μέθοδος επίλυσης τους (σχεδίαση αλγορίθμου)
4. Πώς θα καταγραφεί η λύση σε ένα πρόβλημα (π.χ σε ψευδογλώσσα), και
5. Ποιος είναι ο τρόπος υλοποίησης στο συγκεκριμένο υπολογιστικό σύστημα (π.χ επιλογή γλώσσας προγραμματισμού)

Περιγραφή και αναπαράσταση αλγορίθμων

Υπάρχουν διάφοροι τρόποι για την αναπαράσταση ενός αλγορίθμου:

- με **ελεύθερο κείμενο (free text)**, που αποτελεί τον πιο ανεπεξέργαστο και αδόμητο τρόπο παρουσίασης αλγορίθμου. Έτσι μπορεί εύκολα να οδηγήσει σε μη εκτελέσιμη παρουσίαση παραβιάζοντας το τελευταίο χαρακτηριστικό των αλγορίθμων, δηλαδή την αποτελεσματικότητα.
- με **διαγραμματικές τεχνικές (diagramming techniques)**, που συνιστούν ένα γραφικό τρόπο παρουσίασης αλγορίθμου. Η πιο γνωστή από αυτές 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 στη βιβλιογραφία και πράξη.
- με **φυσική γλώσσα (natural language)** κατά βήματα. Στην περίπτωση αυτή χρειάζεται προσοχή, γιατί μπορεί να παραβιασθεί το τρίτο βασικό χαρακτηριστικό ενός αλγορίθμου, όπως προσδιορίστηκε προηγουμένως δηλαδή το κριτήριο του καθορισμού.
- με **κωδικοποίηση (coding)**, δηλαδή με ένα πρόγραμμα που όταν εκτελεσθεί θα δώσει τα ίδια αποτελέσματα με τον αλγόριθμο.

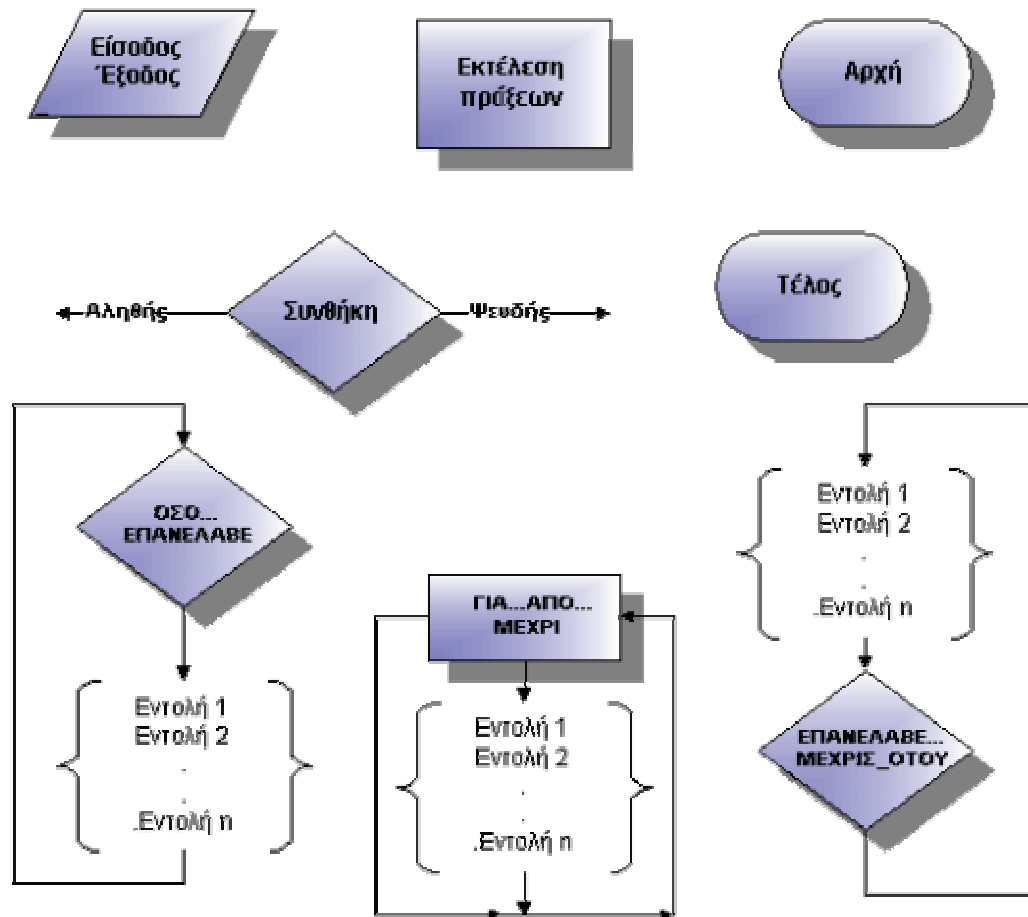
Όλοι οι αλγόριθμοι που παρουσιάζονται στη μελέτη αυτή είναι κωδικοποιημένοι σε μία υποθετική δομημένη ψευδογλώσσα, ωστόσο οι περισσότεροι από αυτούς μπορούν να προγραμματιστούν σε οποιαδήποτε γλώσσα προγραμματισμού.

Σύμβολα ροής διαγράμματος

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

- **έλλειψη**, που δηλώνει την αρχή και το τέλος του κάθε αλγορίθμου
- **ρόμβο**, που δηλώνει μία ερώτηση με δυο ή περισσότερες εξόδους για απάντηση
- **ορθογώνιο**, που δηλώνει την εκτέλεση μίας ή περισσότερων πράξεων, και
- **πλάγιο παραλληλόγραμμο**, που δηλώνει είσοδο ή έξοδο στοιχείων .
- συγκεκριμένα σχέδια για τις επαναληπτικές δομές **ΓΙΑ...ΑΠΟ...ΜΕΧΡΙ**, **ΟΣΟ...ΕΠΑΝΕΛΑΒΕ** και **ΕΠΑΝΕΛΑΒΕ...ΜΕΧΡΙΣ_ΟΤΟΥ**.

Το επόμενο σχήμα αποτυπώνει όλα αυτά τα σύμβολα:



Μαζί με τα διαγράμματα ροής χρησιμοποιείται και η περιγραφή του αλγορίθμου υπό μορφή ψευδογλώσσας, που θα περιγραφεί λεπτομερειακά σε επόμενο κεφάλαιο.

Αξιολόγηση αλγορίθμων

Οι αλγόριθμοι μπορούν να αξιολογηθούν κατά πόσο είναι καλοί ή όχι. Εάν δεν ικανοποιούν έστω ένα από τα πιο πάνω κριτήρια τότε ο αλγόριθμος κρίνεται ως «κακός αλγόριθμος».

Πιο κάτω υπάρχουν μερικά ερωτήματα για την κατανόηση της επίδοσης ενός αλγορίθμου. Αυτά είναι:

- πώς υπολογίζεται ο χρόνος εκτέλεσης ενός αλγορίθμου;
- πώς μπορούν να συγκριθούν μεταξύ τους οι διάφοροι αλγόριθμοι;
- πώς μπορεί να γνωρίζει κανείς, αν ένας αλγόριθμος είναι «βέλτιστος»;

Οι πληροφορίες αυτές αφορούν κυρίως στην αναγνώριση της χειρότερης περίπτωσης του αλγορίθμου και στην αποτύπωση του μεγέθους του προβλήματος, με βάση το πλήθος των δεδομένων.

Η χειρότερη περίπτωση ενός αλγορίθμου αφορά το μέγιστο κόστος εκτέλεσης του αλγορίθμου, κόστος που μετράται σε υπολογιστικούς πόρους. Το κόστος αυτό πολλές φορές κρίνει την επιλογή και το σχεδιασμό ενός αλγορίθμου. Για να εκφρασθεί αυτή η χειρότερη περίπτωση χρειάζεται κάποιο μέγεθος σύγκρισης και αναφοράς που να χαρακτηρίζει τον αλγόριθμο. Η πλέον συνηθισμένη πρακτική μέτρηση είναι η μέτρηση του αριθμού των βασικών πράξεων που θα πρέπει να εκτελέσει ο αλγόριθμος στη χειρότερη περίπτωση. Για παράδειγμα μία βασική πράξη μπορεί να είναι:

- ανάθεση τιμής
- σύγκριση μεταξύ δυο μεταβλητών, ή
- οποιαδήποτε αριθμητική πράξη μεταξύ δυο μεταβλητών

Η χειρότερη περίπτωση αντιπροσωπεύει τις τιμές εκείνες, που όταν δίνονται ως είσοδος στον αλγόριθμο, οδηγούν στην εκτέλεση μέγιστου αριθμού πράξεων.

Δομές Δεδομένων

Δεδομένα: Είναι η αφαιρετική αναπαράσταση της πραγματικότητας και συνεπώς μιας απλοποιημένης όψης της. Είναι ακατέργαστα γεγονότα και κάθε φορά η επιλογή τους εξαρτάται από τον τύπο του προβλήματος. Η συλλογή των ακατέργαστων δεδομένων και ο συσχετισμός τους δίνει ως αποτέλεσμα την πληροφορία (**information**).

Η Πληροφορική θεωρείται η επιστήμη που μελετά τα δεδομένα από τις ακόλουθες σκοπιές :

- **Υλικό.** Το υλικό (**hardware**) δηλαδή η μηχανή, επιτρέπει στα δεδομένα ενός προγράμματος να αποθηκεύονται στη κύρια μνήμη και στις περιφερειακές συσκευές του υπολογιστή με διάφορες αναπαραστάσεις (**representations**). Τέτοιες μορφές είναι η δυαδική, με τη βοήθεια του κώδικα ASCII, το συμπλήρωμα του 1 ή του 2 κ.τ.λ
- **Γλώσσες προγραμματισμού.** Οι γλώσσες προγραμματισμού υψηλού επιπέδου (**high level programming languages**) επιτρέπουν τη χρήση διάφορων τύπων (**types**) μεταβλητών (**variables**) για να περιγράψουν ένα δεδομένο. Ο μεταφραστής κάθε γλώσσας φροντίζει για την αποδοτικότερη μορφή αποθήκευσης, από πλευράς υλικού, κάθε μεταβλητής στον υπολογιστή.
- **Δομές Δεδομένων.** Δομή Δεδομένων (**data structure**) είναι ένα σύνολο δεδομένων μαζί με ένα σύνολο επιτρεπτών λειτουργιών επί αυτών. Για παράδειγμα, μία τέτοια δομή είναι η εγγραφή (**record**), που μπορεί να περιγράψει ένα είδος, ένα πρόσωπο κ.τ.λ. Η εγγραφή αποτελείται από τα πεδία (**fields**) που αποθηκεύουν χαρακτηριστικά (**attributes**) διαφορετικού τύπου, όπως για παράδειγμα ο κωδικός, η περιγραφή κλπ. Άλλη μορφή δομής δεδομένων είναι το αρχείο που αποτελείται από ένα σύνολο εγγραφών. Μία επιτρεπτή λειτουργία σε ένα αρχείο είναι η σειριακή προσπέλαση όλων των εγγραφών του.
- **Ανάλυση Δεδομένων.** Τρόποι καταγραφής και αλληλοσυσχέτισης των δεδομένων μελετώνται έτσι ώστε να αναπαρασταθεί η γνώση για πραγματικά γεγονότα. Οι τεχνολογίες των Βάσεων Δεδομένων (**Databases**), της Μοντελοποίησης Δεδομένων (**Data Modelling**) και της Αναπαράστασης Γνώσης (**Knowledge Representation**) ανήκουν σε αυτή την περιοχή μελέτης των δεδομένων.

Τα δεδομένα ενός προβλήματος αποθηκεύονται στον υπολογιστή, είτε στη κύρια μνήμη του είτε στη δευτερεύουσα. Η αποθήκευση αυτή δεν γίνεται κατά ένα τυχαίο τρόπο αλλά συστηματικά δηλαδή χρησιμοποιώντας μία δομή. Η έννοια της δομής δεδομένων (**data structure**) είναι σημαντική για την Πληροφορική και ορίζεται με τον ακόλουθο ορισμό:

Ορισμός: Δομή Δεδομένων είναι ένα σύνολο αποθηκευμένων δεδομένων που υφίστανται επεξεργασία από ένα σύνολο λειτουργιών.

Κάθε μορφή δομής δεδομένων αποτελείται από ένα σύνολο κόμβων (**nodes**). Οι βασικές λειτουργίες (ή αλλιώς πράξεις) επί των δομών δεδομένων είναι οι ακόλουθες:

- **Προσπέλαση (access)**, πρόσβαση σε ένα κόμβο με σκοπό να εξετασθεί ή να τροποποιηθεί το περιεχόμενό του.
- **Εισαγωγή (insertion)**, δηλαδή η προσθήκη νέων κόμβων σε μία υπάρχουσα δομή.
- **Διαγραφή (deletion)**, που αποτελεί το αντίστροφο της εισαγωγής, δηλαδή ένας κόμβος αφαιρείται από μία δομή.
- **Αναζήτηση (searching)**, κατά την οποία προσπελαύνονται οι κόμβοι μιας δομής, προκειμένου να εντοπιστούν ένας ή περισσότεροι που έχουν μια δεδομένη ιδιότητα.
- **Ταξινόμηση (sorting)**, όπου οι κόμβοι μιας δομής διατάσσονται κατά αύξουσα ή φθίνουσα σειρά.
- **Αντιγραφή (copying)**, κατά την οποία όλοι οι κόμβοι ή μερικοί από τους κόμβους μιας δομής αντιγράφονται σε άλλη δομή.
- **Συγχώνευση (merging)**, κατά την οποία δύο ή περισσότερες δομές συνενώνονται σε μία ενιαία δομή.
- **Διαχωρισμός (separation)**, που αποτελεί την αντίστροφη πράξη της συγχώνευσης.

Υπάρχει μεγάλη εξάρτηση μεταξύ της δομής δεδομένων και του αλγόριθμου και αυτό δικαιολογεί την εξίσωση που διατυπώθηκε το 1976 από τον Wirth (που σχεδίασε και υλοποίησε τη γλώσσα Pascal)

Αλγόριθμοι + Δομές Δεδομένων = Προγράμματα