

ΣΥΜΒΟΛΑΙΟ ΤΑΞΗΣ

Από την αρχή της φετινής χρονιάς με καθηγητές τον κ. Γκοντζή και τον κ Ζυγούρα στο μάθημα του project συζητήσαμε ότι θα ήταν καλή ιδέα να φτιάξουμε ένα συμβόλαιο για την τάξη. Το συμβόλαιο είναι ένας τρόπος με τον οποίο μπορούμε να θέσουμε κάποιους κανόνες όπου θα τηρούμε όλοι οι μαθητές μέσα στην τάξη. Μετά από κάποιες συζητήσεις συμφωνήσαμε για την δημιουργία αυτού του συμβολαίου για τις δομές του και για την τήρηση του από όλους τους μαθητές στην ακόλουθη σχολική χρονιά. Με τον τρόπο του συμβολαίου θα μπορούμε να κάνουμε το μάθημα μας διασκεδαστικό ενδιαφέρον με ηρεμία, χωρίς εντάσεις, διαμάχες ,αντιζηλιές κ.τ.λ. έτσι ώστε όσο οι μαθητές τόσο και οι καθηγητές να περνούν καλά και όχι βαρετά την ώρα του μαθήματος. Στην εργασία αυτή οι καθηγητές μας βοήθησαν αρκετά ώστε να βγει κάτι ωραίο και σωστό. Μας πρότειναν διάφορα πράγματα που θα μπορούσαμε εύκολα να περιγράψουμε τις ιδέες μας. Αρχικά ο κάθε μαθητής έγραψε σε ένα χαρτάκι πώς θα ήθελε να είναι η τάξη και ποια κατά την γνώμη του η κατάλληλη συμπεριφορά των παιδιών. Με αυτόν τον τρόπο καταγράψαμε τις απαιτήσεις και αυτά τα οποία θα έκαναν έναν μαθητή να νιώθει καλά στην ώρα του μαθήματος. Αφού συνοψίσαμε τις γνώμες μας με απλές και συγκεκριμένες λέξεις καταλήξαμε στο πώς θα φτιάχναμε το συμβόλαιο μας. Αποφασίσαμε λοιπόν και κρατήσαμε μερικές λέξεις που θα χαρακτήριζαν τους όρους που είχαμε επιλέξει. Καθώς ολοκληρώσαμε το περιεχόμενο περάσαμε στο εξωτερικό κομμάτι. Σκεφτήκαμε ότι θα ήταν ωραία ιδέα το φόντο να είναι κάτι σχετικό με την πληροφορική. Τελικά αποφασίσαμε να ζωγραφίσουμε έναν υπολογιστή για φόντο και στην οθόνη του να γράψουμε τις λέξεις του συμβολαίου. Μείναμε όλοι ευχαριστημένοι και η χρονιά κύλησε όμορφα και διασκεδαστικά.

ΕΝΝΟΙΑ ΠΡΟΒΛΗΜΑ

Τη λέξη πρόβλημα την έχετε συναντήσει πολλές φορές από τις πρώτες τάξεις του σχολείου. Έχετε λύσει, για παράδειγμα, προβλήματα στα Μαθηματικά και τη Φυσική. Προβλήματα, όμως, αντιμετωπίζουμε και καθημερινά, όπως: ποιος είναι ο πιο σύντομος δρόμος, για να πάμε στο σχολείο μας, πώς να οργανώσουμε μία εκδρομή, πώς να τακτοποιήσουμε τα βιβλία στη βιβλιοθήκη, ώστε να τα βρίσκουμε ευκολότερα. Τα προβλήματα που μόλις αναφέραμε είναι σχετικά απλά και σύντομα βρίσκουμε τη λύση τους. Πολλά προβλήματα, όμως, είναι πιο πολύπλοκα και η επίλυσή τους μας δυσκολεύει ιδιαίτερα. Για παράδειγμα, η ρύπανση της ατμόσφαιρας, η εξοικονόμηση ενέργειας, η θεραπεία ορισμένων ασθενειών, η εξερεύνηση του διαστήματος και η κατασκευή μιας γέφυρας μεγάλου μήκους, είναι ιδιαίτερα σύνθετα προβλήματα. Υπάρχουν επίσης και άλλες κατηγορίες προβλημάτων που: είτε δεν μπορούμε να τα επιλύσουμε με τις μέχρι τώρα γνώσεις μας, όπως η ακριβής πρόβλεψη των σεισμών, η γήρανση του ανθρώπου, η ανακάλυψη εξωγήινων πολιτισμών και η επικοινωνία μαζί τους, είτε έχει αποδειχθεί ότι δεν μπορούμε να τα επιλύσουμε, όπως: ο τετραγωνισμός του κύκλου με κανόνα και διαβήτη ή το ταξίδι στο παρελθόν.

Τα προβλήματα που καλούμαστε να επιλύσουμε στο σχολείο είναι συνήθως υπολογιστικά και απαιτούν μια σειρά από λογικές σκέψεις και μαθηματικές πράξεις. Για παράδειγμα: «ποιο είναι το εμβαδόν ενός τετραγώνου με πλευρά μήκους 10 εκατοστών;», «σε πόσο χρόνο θα πέσει ένα αντικείμενο που εκτελεί ελεύθερη πτώση από ύψος 10 μέτρων;» Παρόμοια υπολογιστικά προβλήματα συχνά καλούμαστε να επιλύσουμε και στην καθημερινή μας ζωή, όπως: «ποιος είναι ο μέσος όρος της βαθμολογίας μου;», «τι διαστάσεις πρέπει να έχει το γραφείο που θα αγοράσω, για να χωράει στο δωμάτιο μου;», «πόσα χρήματα χρειαζόμαστε, για να αγοράσουμε τον αγαπημένο μας δίσκο μουσικής, όταν η αρχική του τιμή είναι 15 € και έχει έκπτωση 20%;».

Γενικότερα, ως πρόβλημα θεωρούμε κάθε ζήτημα που τίθεται προς επίλυση, κάθε κατάσταση που μας απασχολεί και πρέπει να αντιμετωπιστεί. Η λύση ενός προβλήματος δεν μας είναι γνωστή, ούτε προφανής.

Η πρώτη μας ενέργεια για να λύσουμε πιο εύκολα ένα πρόβλημα, είναι η καταγραφή των δεδομένων. Δεδομένα προβλήματος είναι τα στοιχεία που μας είναι γνωστά και μπορούν να μας βοηθήσουν στη λύση του προβλήματος. Σε κάθε πρόβλημα ψάχνουμε να βρούμε την απάντηση σε μια ερώτηση. Αυτό που ψάχνουμε είναι το ζητούμενο. Για παράδειγμα, το ζητούμενο σε μια κατασκήνωση μπορεί να είναι το στήσιμο της σκηνής ή ο καταμερισμός των εργασιών. Σε μια παρτίδα σκάκι ζητούμενο είναι οι κατάλληλες κινήσεις που θα μας οδηγήσουν σε «ματ» του αντίπαλου βασιλιά. Σε ένα γεωμετρικό πρόβλημα ζητούμενο μπορεί να είναι το μήκος ενός ευθυγράμμου τμήματος. Η διαδικασία μέσω της οποίας βρίσκουμε το ζητούμενο και επιτυγχάνουμε τον επιθυμητό στόχο ονομάζεται επίλυση προβλήματος. Υπάρχουν προβλήματα, των οποίων τη λύση μπορούμε να περιγράψουμε με ακρίβεια (π.χ.: ο υπολογισμός της υποτεινουσας ορθογωνίου τριγώνου) και προβλήματα που δεν έχουν ακριβή λύση (π.χ.: η αξιοποίηση του ελεύθερου χρόνου μας). Ακόμα πολλές φορές πρέπει να ελέγχουμε, αν τα δεδομένα του προβλήματος που έχουμε είναι επαρκή, ώστε να μπορούμε να σχεδιάσουμε την επίλυσή του.

Πολλές φορές η λύση ενός προβλήματος χρειάζεται περισσότερη διερεύνηση. Για παράδειγμα στο επόμενο πρόβλημα:

Ένας εργάτης χτίζει 1 μέτρο τοίχο σε 2 ώρες. Σε πόσο χρόνο θα έχει ολοκληρώσει το χτίσιμο 11 μέτρων, αν δουλέψει μόνος του;

Η απάντηση: σε 22 ώρες φαίνεται λογική, αλλά ξεχνάμε ότι ένας εργάτης δεν μπορεί να δουλέψει 22 ώρες συνεχόμενες!

Έτσι, για να επιλύσουμε ένα πρόβλημα πρέπει αρχικά να το κατανοήσουμε. Πρέπει δηλαδή να καταλάβουμε καλά το περιεχόμενο του, να διακρίνουμε τα δεδομένα που έχουμε στη διάθεσή μας και τα ζητούμενα του. Είναι σημαντικό, όμως, να προσδιορίσουμε και το «περιβάλλον» ή το πλαίσιο μέσα στο οποίο εντάσσεται το πρόβλημα (χώρος του προβλήματος).

Για παράδειγμα, στο σύνολο των φυσικών αριθμών η αφαίρεση 3 - 9 είναι αδύνατη, ενώ στο σύνολο των ακεραίων αριθμών η ίδια αφαίρεση έχει αποτέλεσμα $3 - 9 = -6$. Στο παράδειγμα της οργάνωσης μιας εκδρομής το «περιβάλλον» του προβλήματος είναι το σχολικό περιβάλλον. Η οργάνωση μιας εκπαιδευτικής εκδρομής έχει αρκετά διαφορετικά στοιχεία από την οργάνωση μιας εκδρομής με φίλους. Μια εκπαιδευτική εκδρομή πρέπει να πραγματοποιηθεί μέσα στα πλαίσια των κανόνων που καθορίζονται από το σχολικό περιβάλλον, ενώ μια εκδρομή με φίλους ακολουθεί διαφορετικούς κανόνες.

Στην πραγματικότητα, τα περισσότερα προβλήματα είναι σύνθετα και δε μας έρχεται στο νου η λύση τους με την πρώτη ματιά. Χρειάζεται πολλές φορές να τα μελετήσουμε σε βάθος και να εξερευνήσουμε διαφορετικούς πιθανούς τρόπους επίλυσής τους. Όσο περισσότερο μελετάμε ένα πρόβλημα, τόσο πιο πιθανό είναι να το επιλύσουμε. Συχνά μάλιστα η λύση του μας έρχεται σαν αναλαμπή, σε άσχετη φαινομενικά στιγμή. Αρκεί να θυμηθούμε το πρόβλημα του Αρχιμήδη που βασάνιζε για καιρό το μυαλό του –πώς θα μπορέσει να αποδείξει ότι το στέμμα του βασιλιά αποτελείται μόνο από χρυσάφι ή από πρόσμικξη και άλλων μετάλλων ίδιου βάρους– και όταν ξαφνικά βρήκε τη λύση την ώρα που έκανε μπάνιο, πήδησε έξω ενθουσιασμένος φωνάζοντας «Εύρηκα!».

Για να μπορέσουμε να επιλύσουμε ένα σύνθετο πρόβλημα, είναι αναγκαίο να το αναλύσουμε σε απλούστερα προβλήματα. Για παράδειγμα, η οργάνωση μίας σχολικής εκδρομής (Σχήμα 1.1), αν και φαίνεται απλή, είναι ένα σύνθετο πρόβλημα. Για την καλύτερη επίλυσή του μπορούμε να το χωρίσουμε σε μια σειρά από απλούστερα προβλήματα.

ΔΙΕΡΕΥΝΗΣΗ ΠΡΟΒΛΗΜΑΤΟΣ

Η οποιαδήποτε προσπάθεια αντιμετώπισης ενός προβλήματος είναι καταδικασμένη σε αποτυχία αν προηγουμένως δεν έχει γίνει απόλυτα κατανοητό το πρόβλημα που τίθεται. Η κατανόηση ενός προβλήματος αποτελεί συνάρτηση δύο παραγόντων, της σωστής διατύπωσης εκ μέρους του δημιουργού του και της αντίστοιχα σωστής ερμηνείας από τη μεριά εκείνου που καλείται να το αντιμετωπίσει.

Η μορφή με την οποία παρουσιάζεται ένα πρόβλημα μπορεί να είναι οποιαδήποτε αρκεί να μπορεί να γίνει αντιληπτή από μία από τις πέντε ανθρώπινες αισθήσεις. Το πρόβλημα της ρύπανσης της ατμόσφαιρας της πρωτεύουσας μπορεί να το αντιληφθεί ο καθένας κοιτάζοντας τον αττικό ουρανό ή αναπνέοντας με δυσκολία ανηφορίζοντας κάποιο καλοκαιρινό μεσημέρι την οδό Ακαδημίας. Τα προβλήματα και τα δεινά που ταλαιπωρούν και σκοτώνουν χιλιάδες συνανθρώπους μας εξ αιτίας των πολεμικών συγκρούσεων στα διάφορα μέρη του κόσμου, μας γίνονται γνωστά είτε διαβάζοντας εφημερίδες, είτε ακούγοντας το ραδιόφωνο, είτε βλέποντας τηλεόραση.

Τα προβλήματα που μπορεί να κληθούμε να αντιμετωπίσουμε κατά τη διάρκεια της ζωής μας μπορούν να αναφέρονται σε οποιοδήποτε τομέα, μπορεί να αφορούν στα μαθηματικά, στη φυσική, στη λογική, στην καθημερινή ζωή ή οτιδήποτε άλλο θα μπορούσε κάποιος να σκεφτεί. Μπορεί να απαιτούνται γνώσεις συγκεκριμένων επιστημών ή μπορεί οι βιωματικές μας καταστάσεις και εμπειρίες να επαρκούν για την αντιμετώπισή τους. Μπορεί να είναι πολύπλοκα ή σχετικά απλά, πρωτόγνωρα ή συνηθισμένα. Σε κάθε όμως περίπτωση θα πρέπει να γίνουν απολύτως κατανοητά πριν γίνει κάθε προσπάθεια αντιμετώπισής τους.

Η κατανόηση ενός προβλήματος εξαρτάται σε μεγάλο βαθμό από την διατύπωσή του. Οποιοδήποτε μέσο μπορεί να χρησιμοποιηθεί για να αποδοθεί η διατύπωση ενός προβλήματος. Συνηθέστερο από όλα είναι ο λόγος, είτε ο προφορικός, είτε ο γραπτός.

Σαφήνεια διατύπωσης

Ο λόγος σαν μέσο επικοινωνίας και συνεννόησης πρέπει να χαρακτηρίζεται από σαφήνεια. Άστοχη χρήση ορολογίας, λανθασμένη σύνταξη, είναι δύο στοιχεία που μπορούν να προκαλέσουν παρερμηνείες και παραπλανήσεις. Η ικανότητα εκφοράς σωστού προφορικού και γραπτού λόγου αποτελεί μεγάλο προτέρημα για κάθε άτομο. Η παρερμηνεία είναι δυνατή ακόμα και σε περιπτώσεις όπου όλοι οι λεξικολογικοί και συντακτικοί κανόνες κρατούνται με ευλάβεια. Στάδια αντιμετώπισης του προβλήματος
Η σωστή επίλυση ενός προβλήματος προϋποθέτει τον επακριβή προσδιορισμό των δεδομένων που παρέχει το πρόβλημα. Απαιτεί επίσης την λεπτομερειακή καταγραφή των ζητούμενων που αναμένονται ως αποτελέσματα της επίλυσης του προβλήματος. Θα πρέπει να δοθεί μεγάλη προσοχή στον προσδιορισμό των δεδομένων του προβλήματος. Το ίδιο προσεκτικά θα πρέπει να αποσαφηνιστούν και τα ζητούμενα του προβλήματος

Τα τρία στάδια της αντιμετώπισης ενός προβλήματος είναι:

κατανόηση, όπου απαιτείται η σωστή και πλήρης αποσαφήνιση των δεδομένων και των ζητούμενων του προβλήματος

ανάλυση, όπου το αρχικό πρόβλημα διασπάται σε άλλα επί μέρους απλούστερα προβλήματα

επίλυση, όπου υλοποιείται η λύση του προβλήματος, μέσω της λύσης των επιμέρους προβλημάτων.

παράδειγμα

ΚΑΤΗΓΟΡΙΕΣ ΠΡΟΒΛΗΜΑΤΩΝ

Με κριτήριο τη δυνατότητα επίλυσης ενός προβλήματος διακρίνουμε τρεις κατηγορίες προβλημάτων:

Επιλύσιμα, είναι εκείνα τα προβλήματα για τα οποία η λύση τους είναι ήδη γνωστή και έχει διατυπωθεί

Ανοικτά, ονομάζονται εκείνα τα προβλήματα για τα οποία η λύση τους δεν έχει μεν ακόμα βρεθεί, αλλά παράλληλα δεν έχει αποδειχθεί, ότι δεν επιδέχονται λύση.

Άλυτα χαρακτηρίζονται εκείνα τα προβλήματα για τα οποία έχουμε φτάσει στη παραδοχή, ότι δεν επιδέχονται λύση.

Με κριτήριο το βαθμό δόμησης των λύσεων τους, τα επιλύσιμα προβλήματα μπορούν να διακριθούν σε τρεις επίσημες κατηγορίες:

Δομημένα χαρακτηρίζονται εκείνα τα προβλήματα των οποίων η επίλυση προέρχεται από μια αυτοματοποιημένη διαδικασία.

Ημιδομημένα ονομάζονται τα προβλήματα εκείνων των οποίων η λύση επιδιώκεται στα πλαίσια ενός εύρους πιθανών λύσεων αφήνοντας στον ανθρώπινο παράγοντα τα περιθώρια επιλογής.

Αδόμητα χαρακτηρίζονται τα προβλήματα εκείνα των οποίων οι λύσεις δεν μπορούν να δομηθούν ή δεν έχει διερευνηθεί σε βάθος η δυνατότητα δόμησης τους.

Με κριτήριο το είδος επίλυσης που επιζητούν τα προβλήματα διακρίνονται σε τρεις κατηγορίες:

Απόφασης, όπου η απόφαση που πρόκειται να ληφθεί ως λύση του προβλήματος που τίθεται απαντά σε ένα ερώτημα και πιθανόν αυτή η απάντηση να είναι ένα "Ναι" ή ένα "Όχι". Αυτό που θέλουμε να διαπιστώσουμε σε ένα πρόβλημα απόφασης είναι αν υπάρχει απάντηση που ικανοποιεί τα δεδομένα που θέτονται από το πρόβλημα.

Υπολογιστικά, όπου το πρόβλημα που τίθεται απαιτεί τη διενέργεια υπολογισμών, για να μπορεί να δοθεί μια απάντηση στο πρόβλημα. Σε ένα υπολογιστικό πρόβλημα ζητάμε να βρούμε τη τιμή της απάντησης που ικανοποιεί τα δεδομένα του προβλήματος.

ΕΠΙΛΥΣΗ ΕΝΟΣ ΠΡΟΒΛΗΜΑΤΟΣ

Για να μπορέσουμε να βρούμε την λύση ενός προβλήματος θα πρέπει να γνωρίζουμε τη δημιουργία αυτό το πρόβλημα, έπειτα πρέπει να μείνουμε ψύχραιμοι, επίσης το πρόβλημα θα πρέπει να λυθεί με ομιλία και όχι με βίαιους τρόπους(αναλόγως το κάθε πρόβλημα) ΛΥΣΗ -Ποια είναι τα στάδια επίλυσης ενός προβλήματος: -Ποιες είναι οι κατηγορίες των προβλημάτων: 1)Εντοπίζουμε το πρόβλημα. 2)Συζητάμε για το πρόβλημα αφότου το βρούμε(για να το καταλάβουμε καλύτερα) 3)Διαλέγουμε τον καλύτερο τρόπο επίλυσης για να λύσουμε το πρόβλημα. ΚΑΤΑΝΟΗΣΗ ΑΝΑΛΥΣΗ ΕΠΙΛΥΣΗ

Αλγόριθμος

Ως αλγόριθμος ορίζεται μια πεπερασμένη σειρά ενεργειών, αυστηρά καθορισμένων και εκτελέσιμων σε πεπερασμένο χρόνο, που στοχεύουν στην επίλυση ενός προβλήματος. Πιο απλά αλγόριθμο ονομάζουμε μία σειρά από εντολές που έχουν αρχή και τέλος, είναι σαφείς και εκτελέσιμες που σκοπό έχουν την

επίλυση κάποιου προβλήματος.

Η λέξη αλγόριθμος προέρχεται από μία μελέτη του Πέρση μαθηματικού του 8ου αιώνα μ.Χ. Χουαρίζμι, η οποία περιείχε συστηματικές τυποποιημένες λύσεις αλγεβρικών προβλημάτων και αποτελεί ίσως την πρώτη πλήρη πραγματεία άλγεβρας. Πέντε αιώνες αργότερα η μελέτη μεταφράστηκε στα Λατινικά και άρχισε με τη φράση "Algoritmus dixit" (ο Αλγόριθμος είπε...). Έτσι η λέξη αλγόριθμος καθιερώθηκε αργά τα επόμενα χίλια χρόνια με την έννοια «συστηματική διαδικασία αριθμητικών χειρισμών». Τη σημερινή της σημασία την οφείλει στη γρήγορη ανάπτυξη των ηλεκτρονικών υπολογιστών στα μέσα του 20ου αιώνα.

Η έννοια του αλγορίθμου γίνεται ευκολότερα αντιληπτή με το παρακάτω παράδειγμα. Αν κάποιος επιθυμεί να γευματίσει θα πρέπει να εκτελέσει κάποια συγκεκριμένα βήματα: να συγκεντρώσει τα υλικά, να προετοιμάσει τα σκεύη μαγειρικής, να παρασκευάσει το φαγητό, να στρώσει το τραπέζι, να ετοιμάσει τη σαλάτα, να γευματίσει, να καθαρίσει το τραπέζι και να πλύνει τα πιάτα. Προφανώς, η προηγούμενη αλληλουχία οδηγεί στο επιθυμητό αποτέλεσμα. Δεν είναι όμως η μοναδική για την επίτευξη του σκοπού, αφού μπορεί να αλλάξει η σειρά των βημάτων (π.χ. πρώτα να ετοιμάσει τη σαλάτα και μετά να στρώσει το τραπέζι). Ωστόσο το νόημα είναι πως η κατάτμηση μιας σύνθετης εργασίας σε διακριτά βήματα που εκτελούνται διαδοχικά, είναι ο πιο πρακτικός τρόπος επίλυσης πολλών προβλημάτων.

Δημιουργία αλγορίθμου

Τα βήματα δημιουργίας αλγορίθμου είναι:

- Διατύπωση του προβλήματος
- Κατανόηση του προβλήματος
- Λύση του προβλήματος
- Διατύπωση του αλγορίθμου
- Έλεγχος της λύσης

Κριτήρια αλγορίθμου

Οι αλγόριθμοι θα πρέπει να πληρούν κάποια πρότυπα και να διατυπώνονται με συγκεκριμένο τρόπο.

Έτσι ένας αλγόριθμος πρέπει να ικανοποιεί τα επόμενα κριτήρια:

- Καθοριστικότητα
- Περαιότητα

- Αποτελεσματικότητα
- Επεκτασιμότητα
- Να έχει είσοδο δεδομένων, επεξεργασία και έξοδο αποτελεσμάτων

Καθοριστικότητα - Definiteness

Κάθε κανόνας του ορίζεται επακριβώς και η αντίστοιχη διεργασία είναι συγκεκριμένη. Κάθε εντολή πρέπει να καθορίζεται χωρίς καμία αμφιβολία για τον τρόπο εκτέλεσής της. Π.χ. Σε μία διαίρεση να λαμβάνεται υπόψη και η περίπτωση όπου ο διαιρετέος λαμβάνει μηδενική τιμή. Τυπικές περιπτώσεις η διαίρεση με το μηδέν, υπόριζος ποσότητα αρνητική, κλπ. Προβλήματα καθοριστικότητας αντιμετωπίζονται συχνά με τη λογική της επιλογής, δηλ. Αν $a > 0$ τότε αλλιώς

Περατότητα - Finiteness

Κάθε εκτέλεση είναι πεπερασμένη, δηλαδή τελειώνει ύστερα από έναν πεπερασμένο αριθμό διεργασιών ή βημάτων. Μία διαδικασία που δεν τελειώνει μετά από συγκεκριμένο/πεπερασμένο αριθμό βημάτων λέγεται απλά υπολογιστική διαδικασία.

Αποτελεσματικότητα - Effectiveness

Είναι μηχανιστικά αποτελεσματικός, δηλαδή όλες οι διαδικασίες που περιλαμβάνει μπορούν να πραγματοποιηθούν με ακρίβεια και σε πεπερασμένο χρόνο "με μολύβι και χαρτί". Κάθε μεμονωμένη εντολή του αλγορίθμου να είναι απλή (και όχι σύνθετη). Δηλαδή μία εντολή δεν αρκεί να έχει ορισθεί αλλά πρέπει να είναι και εκτελέσιμη.

Είσοδος δεδομένων - Input

Κατά την εκκίνηση εκτέλεσης του αλγορίθμου καμία, μία ή περισσότερες τιμές δεδομένων πρέπει να δίνονται ως είσοδοι στον αλγόριθμο. Η περίπτωση που δε 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 αποτελεσμάτων - Output

Δίδει τουλάχιστον ένα μέγεθος ως αποτέλεσμα που εξαρτάται κατά κάποιο τρόπο από τις αρχικές εισόδους. Ο αλγόριθμος πρέπει να δημιουργεί τουλάχιστον μία τιμή (δεδομένων) ως αποτέλεσμα προς το χρήστη ή προς ένα άλλο αλγόριθμο.

Περιγραφή και αναπαράσταση

Τέσσερις είναι οι βασικοί τρόποι αναπαράστασης ενός αλγορίθμου:

1. *Ελεύθερο κείμενο*, που αποτελεί τον πιο αδόμητο τρόπο παρουσίασης αλγορίθμου. Ελλοχεύει η

δημιουργία μιας μη εκτελέσιμης κατάστασης παραβιάζοντας έτσι το κριτήριο της αποτελεσματικότητας.

2. *Διάγραμμα ροής*, που συνιστά έναν πιο γραφικό τρόπο παρουσίασης του αλγορίθμου. Η χρήση διαγραμμάτων ροής δεν είναι η πλέον ενδεδειγμένη λύση για ένα πρόβλημα και για αυτό χρησιμοποιούνται σπάνια στη βιβλιογραφία.
3. *Φυσική γλώσσα* που εκτελείται κατά βήματα. Σε αυτή την περίπτωση μπορεί να παραβιαστεί το κριτήριο του καθορισμού μεταξύ των βημάτων.
4. *Κωδικοποίηση* του αλγορίθμου σε ψευδογλώσσα ή γλώσσα προγραμματισμού. Έτσι ο αλγόριθμος παρουσιάζεται πιο συνοπτικός, συμπαγής ενώ πληρεί και τις προϋποθέσεις του Δομημένου προγραμματισμού.

Βασικές αλγοριθμικές δομές

Δομή ακολουθίας

Η δόμηση των διαδικασιών σε τέτοια μορφή, έτσι ώστε οι διαδικασίες να εκτελούνται με τη σειρά από τον υπολογιστή. Με άλλα λόγια δομή ακολουθίας είναι μια σειρά βημάτων που ακολουθούμε για να εκτελέσουμε κάτι.

Δομή επιλογής

Η προγραμματιστική δομή που περικλείει τον έλεγχο μιας συνθήκης και μία ή δύο ομάδες εντολών. Από τις ομάδες των εντολών εκτελείται η πρώτη, αν ισχύει η συνθήκη, ή αν υπάρχει και δεύτερη ομάδα εντολών εκτελείται η δεύτερη αν δεν ισχύει η συνθήκη. Με τον όρο συνθήκη εννοούμε δυο όρους ίδιου τύπου και ανάμεσα τους ένας τελεστής σύγκρισης. Με τον όρο τελεστή σύγκρισης εννοούμε ένα από τα σύμβολα $<$, $>$, $<>$, $<=$, $>=$, $=$. Το αποτέλεσμα της σύγκρισης των όρων (νοείται εφόσον οι όροι έχουν κάποια τιμή, δηλαδή δεν περιέχουν την τιμή null) είναι η αλγεβρική τιμή Αληθής (True) ή Ψευδής (False). Οι όροι μπορεί να είναι μεταβλητές ή σταθερές. Με λίγα λόγια δομή επιλογής είναι η επιλογή της καλύτερης ενέργειας μεταξύ κάποιων που έχουμε.

- Αν συνθήκη τότε... αλλιώς

Δομή επανάληψης

Η προγραμματιστική δομή που περικλείει τον συνεχή έλεγχο μίας συνθήκης και μία ομάδα εντολών. Οι

εντολές εκτελούνται ανάλογα με την δομή της επανάληψης. Υπάρχουν τριών ειδών επαναλήψεις. Με λίγα λόγια δομή επανάληψης είναι η συνεχόμενη εκτέλεση βημάτων μέχρι να ικανοποιηθεί μια συνθήκη.

- **Όσο συνθήκη επανέλαβε.** Σε αυτή την δομή επανάληψης ελέγχεται πρώτα η συνθήκη και εφόσον ισχύει (δίνει τιμή αληθή) εκτελείται η ομάδα εντολών.
- **Επανάλαβε μέχρις ότου συνθήκη.** Σε αυτή την δομή επανάληψης εκτελείται η ομάδα εντολών , στην συνέχεια ελέγχεται αν ισχύει η συνθήκη και εφόσον **ΔΕΝ ισχύει** (δίνει τιμή ψευδής) εκτελείται ξανά η ομάδα εντολών.
- **Για N φορές επανάλαβε.** Σε αυτή την δομή επανάληψης εκτελείται η ομάδα εντολών N φορές όπου N είναι αριθμός θετικός ακέραιος.

Τυποποιημένοι αλγόριθμοι

Οι αλγόριθμοι είναι σημαντικοί γιατί σχετίζονται άμεσα με τον τρόπο με τον οποίο οι υπολογιστές επεξεργάζονται δεδομένα και παράγουν πληροφορίες. Ένα πρόγραμμα υπολογιστών είναι ουσιαστικά ένας αλγόριθμος που λέει στον υπολογιστή ποια συγκεκριμένα βήματα να εκτελέσει (σε ποια συγκεκριμένη σειρά) προκειμένου να επιτευχθεί ένας συγκεκριμένος στόχος, όπως π.χ. ο υπολογισμός των μισθών των υπαλλήλων ή η εκτύπωση των έλεγχων των μαθητών. Κατά συνέπεια, ένας αλγόριθμος μπορεί να θεωρηθεί οποιαδήποτε ακολουθία εντολών που μπορεί να εκτελεσθεί από μια υπολογιστική μηχανή (turing).

Χαρακτηριστικά, όταν ένας αλγόριθμος συνδέεται με την επεξεργασία πληροφοριών, τα δεδομένα διαβάζονται από μια συσκευή εισόδου, γράφονται σε μια συσκευή εξόδου, και / ή αποθηκεύονται για την περαιτέρω χρήση. Τα αποθηκευμένα στοιχεία θεωρούνται ως τμήμα της εσωτερικής κατάστασης του συστήματος που εκτελεί τον αλγόριθμο.

Για οποιαδήποτε τέτοια υπολογιστική διαδικασία, ο αλγόριθμος πρέπει να οριστεί αυστηρά: να είναι ορισμένος για όλες τις πιθανές περιστάσεις που θα μπορούσαν να προκύψουν. Δηλαδή οποιαδήποτε υπό όρους βήματα πρέπει να εξεταστούν συστηματικά, και σε κάθε περίπτωση τα κριτήρια πρέπει να είναι σαφή (και υπολογίσιμα).

Επειδή ένας αλγόριθμος είναι ένας ακριβής κατάλογος βημάτων ακριβείας, η σειρά του υπολογισμού θα είναι σχεδόν πάντα κρίσιμη για τη λειτουργία του αλγόριθμου. Οι εντολές συνήθως απαριθμούνται ρητά, και περιγράφονται σαν να ξεκινούν "από την κορυφή" και πηγαίνοντας "προς στο κατώτατο σημείο", μια ιδέα που περιγράφεται τυπικά με τον όρο της "ροής ελέγχου".

Μέχρι τώρα, σε αυτήν η συζήτηση για την τυποποίηση του αλγορίθμου, έχουμε δεχθεί ως βάση τον διαδικαστικό προγραμματισμό. Αυτή είναι και η πιο κοινή αντίληψη, η οποία προσπαθεί να περιγράψει ένα έργο με διακεκριμένα, "μηχανικά" μέσα. Μοναδικός σε αυτήν την αντίληψη των αλγορίθμων είναι ο ρόλος της λειτουργίας ανάθεσης (ο καθορισμός της τιμής μιας μεταβλητής) ο οποίος προέρχεται από τη ιδέα "της μνήμης" σαν πρόχειρο τετράδιο. Δείτε ακόμα το λειτουργικό προγραμματισμό και τον λογικό προγραμματισμό για εναλλακτικές αντιλήψεις για το τι αποτελεί έναν αλγόριθμο.

Εφαρμογή των αλγορίθμων

Οι αλγόριθμοι μπορούν να υλοποιηθούν από προγράμματα ηλεκτρονικών υπολογιστών, μολονότι συχνά σε περιορισμένες μορφές. Ένα λάθος στον σχεδιασμό ενός αλγόριθμου για την λύση ενός προβλήματος μπορεί να οδηγήσει σε αποτυχίες/βλάβες στο εφαρμοσμένο πρόγραμμα.

Οι αλγόριθμοι δεν υλοποιούνται μόνο ως προγράμματα υπολογιστών, αλλά συχνά επίσης και με άλλα μέσα, όπως π.χ. σε ένα βιολογικό νευρικό δίκτυο, ή σε ένα ηλεκτρονικό κύκλωμα, ή σε μια μηχανική συσκευή.

Η ανάλυση και η μελέτη των αλγορίθμων είναι ένας τομέας τής επιστήμης της πληροφορικής, και ασκείται συχνά αφαιρετικά (χωρίς τη χρήση μιας συγκεκριμένης γλώσσας προγραμματισμού ή άλλη εφαρμογή). Από αυτή την άποψη, μοιάζει με άλλους μαθηματικούς τομείς, συγκεκριμένα στο ότι η εστίαση της ανάλυσης είναι πάνω στις βασικές αρχές του αλγορίθμου, και όχι σε οποιαδήποτε ιδιαίτερη εφαρμογή του. Ένας τρόπος απεικόνισης ένας αλγόριθμου είναι το γράψιμο του ψευδοκώδικα. Άλλοι τρόποι είναι: με ελεύθερο κείμενο , με φυσική γλώσσα περιγράφοντας τα βήματα και με λογικό διάγραμμα .

Αλγόριθμος του Ευκλείδη

Μέθοδος του Ευκλείδη για την εύρεση του μέγιστου κοινού διαιρέτη (ΜΚΔ) των δύο αρχικών μηκών BA και DC, και για τα δύο ορίζεται να είναι τα πολλαπλάσια μιας κοινής «μονάδας» μήκους. Το γεγονός ότι μικραίνει το μήκος DC, χρησιμοποιείται για να "μετρήσει" το BA, αλλά μόνο μια φορά, επειδή το υπόλοιπο EA είναι μικρότερο από το CD. Το EA μετρά πλέον (δύο φορές), το μικρότερο μήκος DC, με την υπόλοιπη ομάδα FC μικρότερη από EA. Στη συνέχεια το FC είναι (τρεις φορές) το μήκος EA. Επειδή δεν υπάρχει υπόλοιπο, η διαδικασία τελειώνει με το μήκος FC να είναι ο ΜΚΔ. Δεξιά Νικάμαχος παράδειγμα με τους αριθμούς 49 και 21, με αποτέλεσμα ΜΚΔ τους να είναι το 7.

Στα μαθηματικά, ο αλγόριθμος του Ευκλείδη ή Ευκλείδειος αλγόριθμος, είναι μια αποτελεσματική μέθοδος για τον υπολογισμό του μέγιστου κοινού διαιρέτη (ΜΚΔ) δύο ακέραιων αριθμών, είναι επίσης γνωστός ως ο μεγαλύτερος κοινός παράγοντας ή υψηλότερος κοινός παρονομαστής. Το όνομά του προέρχεται από τον Έλληνα μαθηματικό Ευκλείδη, ο οποίος τον περιγράφει στα βιβλία VII και X του βιβλίου του Στοιχεία.

Στην απλούστερη μορφή του, ο αλγόριθμος του Ευκλείδη ξεκινά με ένα ζεύγος θετικών ακεραίων και σχηματίζει ένα νέο ζευγάρι που αποτελείται από το μικρότερο αριθμό και τη διαφορά μεταξύ των μεγαλύτερων και των μικρότερων αριθμών. Η διαδικασία επαναλαμβάνεται έως ότου οι αριθμοί είναι ίσοι. Αυτός ο αριθμός είναι τότε ο μέγιστος κοινός διαιρέτης του αρχικού ζεύγους.

Η παλαιότερη περιγραφή που υπάρχει για τον Ευκλείδειο αλγόριθμο είναι στα Στοιχεία του Ευκλείδη (περ. 300 π.Χ.), καθιστώντας τον ένα από τους παλαιότερους αριθμητικούς αλγορίθμους ακόμα σε κοινή χρήση. Ο αρχικός αλγόριθμος αφορούσε μόνο φυσικούς αριθμούς και γεωμετρικά μήκη (πραγματικοί αριθμοί), αλλά ο αλγόριθμος γενικεύτηκε τον 19ο αιώνα και σε άλλους τύπους αριθμών, όπως Gaussian ακέραιοι και πολυώνυμα μιας μεταβλητής. Αυτό οδήγησε στις σύγχρονες αφηρημένες αλγεβρικές

έννοιες, όπως οι Ευκλείδειες δομές. Ο Ευκλείδειος αλγόριθμος έχει γενικευτεί περαιτέρω και σε άλλες μαθηματικές δομές, όπως κόμβους κύκλων pots και ολυώνυμα πολλών μεταβλητών .

Ο αλγόριθμος έχει πολλές θεωρητικές και πρακτικές εφαρμογές. Μπορεί να χρησιμοποιηθεί για να δημιουργήσει σχεδόν όλους τους πιο σημαντικούς παραδοσιακούς μουσικούς ρυθμούς που χρησιμοποιούνται σε διαφορετικούς πολιτισμούς σε όλο τον κόσμο. Πρόκειται για ένα βασικό στοιχείο του RSA αλγορίθμου , μια μέθοδο κρυπτογράφησης δημόσιου κλειδιού που χρησιμοποιείται ευρέως στο ηλεκτρονικό εμπόριο. Χρησιμοποιείται για την επίλυση Διοφαντικών εξισώσεων, όπως η εξεύρεση αριθμών που ικανοποιούν πολλαπλά σχήματα με ίδιο ύψος και μέγεθος (Κινέζικο θεώρημα υπολοίπων) ή πολλαπλασιαστικά αντίστροφα ενός πεπερασμένου πεδίου. Μπορεί επίσης να χρησιμοποιηθεί για την κατασκευή συνεχών κλασμάτων , στην αλυσίδα Sturm, μέθοδο για την εύρεση πραγματικών ριζών ενός πολυωνύμου, και σε πολλές σύγχρονες ακέραιες παραγοντοποιήσεις αλγορίθμων. Τέλος, είναι ένα βασικό εργαλείο για την απόδειξη θεωρημάτων στη σύγχρονη θεωρία αριθμών, όπως το θεώρημα των τεσσάρων τετραγώνων του Lagrange και το θεμελιώδες θεώρημα της αριθμητικής (μοναδική παραγοντοποίηση).

Εάν υλοποιηθεί με τη χρήση υπολοίπων της Ευκλείδειας διαίρεσης και όχι με αφαιρέσεις, ο αλγόριθμος του Ευκλείδη υπολογίζει το ΜΚΔ μεγάλων αριθμών αποτελεσματικά: Ποτέ δεν απαιτεί περισσότερα βήματα διαίρεσης από πέντε φορές τον αριθμό των ψηφίων (με βάση το 10) από τον μικρότερο ακέραιο. Αυτό αποδείχθηκε από τον Gabriel Lamé το 1844 και σηματοδοτεί την έναρξη της υπολογιστικής θεωρίας πολυπλοκότητας . Μέθοδοι για τη βελτίωση της απόδοσης του αλγορίθμου αναπτύχθηκαν τον 20ο αιώνα.

Ο ΜΚΔ των δύο αριθμών είναι ο μεγαλύτερος αριθμός που διαιρεί τους δύο χωρίς να αφήνει υπόλοιπο . Ο Ευκλείδειος αλγόριθμος βασίζεται στην αρχή ότι ο μέγιστος κοινός διαιρέτης των δύο αριθμών δεν αλλάζει εάν ο μικρότερος αριθμός αφαιρεθεί από το μεγαλύτερο αριθμό. Αν k , m και n είναι ακέραιοι, και το k είναι ένας κοινός παράγοντας των δύο ακεραίων αριθμών A και B , τότε $A = Nk$ και $B = mk$ συνεπάγεται $A - B = (n - m)k$, συνεπώς, k είναι επίσης ένας κοινός παράγοντας της διαφοράς. Αυτό το k μπορεί επίσης να αντιπροσωπεύει τον μέγιστο κοινό διαιρέτη όπως αποδεικνύεται παρακάτω. Για παράδειγμα, το 21 είναι ο ΜΚΔ των 105 ($252 = 12 \times 21$; $105 = 5 \times 21$). Από το $252 - 105 = (12 - 5) \times 21 = 147$, ο ΜΚΔ των 147 και 105 είναι επίσης 21.

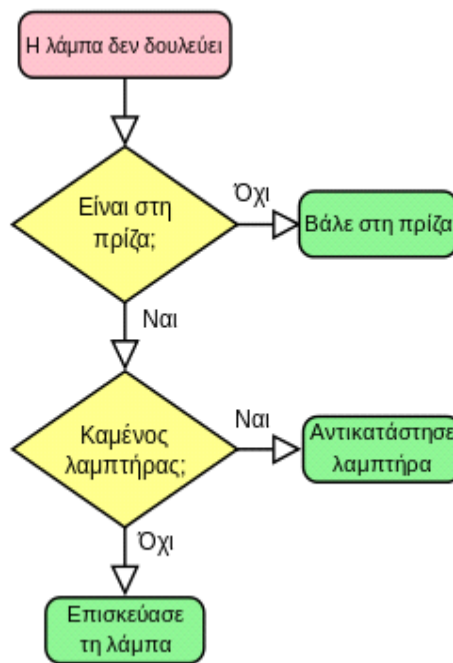
Δεδομένου ότι ο μεγαλύτερος από τους δύο αριθμούς μειώνεται, επαναλαμβάνοντας τη διαδικασία, αυτή θα δίνει διαδοχικά μικρότερους αριθμούς μέχρι ένας από αυτούς να γίνει μηδέν. Όταν αυτό συμβεί, ο ΜΚΔ είναι ο μη μηδενικός αριθμός που απομένει. Αντιστρέφοντας τα βήματα του αλγορίθμου του Ευκλείδη , ο ΜΚΔ μπορεί να εκφραστεί ως το άθροισμα/γραμμικός συνδιασμός των δύο αρχικών αριθμών καθώς πολλαπλασιάζεται με ένα θετικό ή αρνητικό ακέραιο, π.χ. $21 = [5 \times 105] + [(-2) \times 252]$. Αυτή η σημαντική ιδιότητα είναι γνωστή ως αυτότητα του Βézout.

Περιγραφή

Η απλούστερη μορφή του αλγορίθμου του Ευκλείδη ξεκινά με ένα ζεύγος θετικών ακεραίων και σχηματίζει ένα νέο ζευγάρι που αποτελείται από το μικρότερο αριθμό και τη διαφορά μεταξύ των

μεγαλύτερων και των μικρότερων αριθμών. Η διαδικασία επαναλαμβάνεται έως ότου οι αριθμοί είναι ίσοι. Στη συνέχεια ο αριθμός αυτός είναι ο μέγιστος κοινός διαιρέτης του αρχικού ζεύγους. Ωστόσο, εάν ένας αριθμός είναι πολύ μικρότερος από τον άλλο, πολλά βήματα αφαίρεσης θα είναι αναγκαία πριν ο μεγαλύτερος αριθμός μειωθεί σε μία τιμή μικρότερη ή ίση από τον άλλον αριθμό στο ζεύγος. Αφαιρώντας ένα μικρό θετικό αριθμό από ένα μεγάλο αριθμό αρκετές φορές μέχρι αυτό που απομένει είναι μικρότερο από τον αρχικό ελάχιστο αριθμό, μπορεί να αντικατασταθεί με την εύρεση του υπόλοιπο σε μεγάλη διαίρεση. Έτσι, η μορφή διαίρεσης του αλγορίθμου του Ευκλείδη ξεκινά με ένα ζεύγος θετικών ακεραίων και σχηματίζει ένα νέο ζεύγος που αποτελείται από το μικρότερο αριθμό και το υπόλοιπο που λαμβάνεται διαιρώντας τον μεγαλύτερο αριθμό από το μικρότερο αριθμό. Η διαδικασία επαναλαμβάνεται έως ότου ένας αριθμός είναι μηδέν. Ο άλλος αριθμός είναι τότε ο μέγιστος κοινός διαιρέτης του αρχικού ζεύγους.

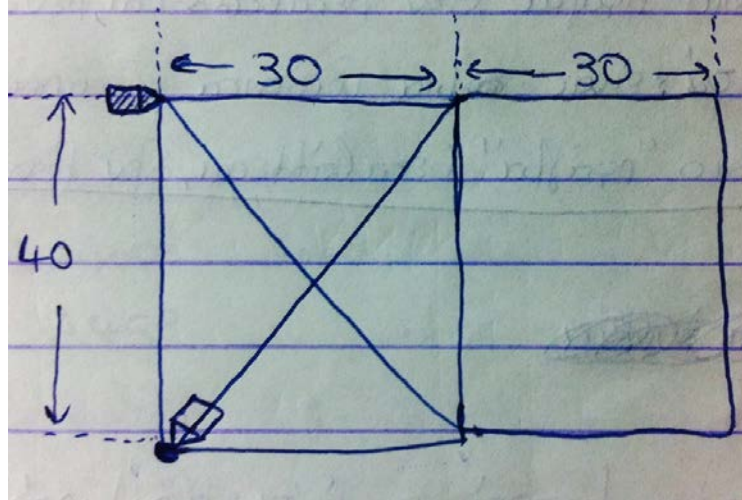
παραδείγματα αλγορίθμου:



Αλγόριθμος καθοδήγηση πένας

Να καθοδηγήσετε την πένα έτσι ώστε να σχεδιάσει το συγκεκριμένο σχήμα χωρίς να περάσει από την ίδια γραμμή.

Σχήμα:



Λύση:

1. στροφή δεξιά 90 μοίρες
2. επανέλαβε βήμα 5 εκατοστά 8 φορές
3. στροφή αριστερά 90 μοίρες
4. επανέλαβε βήμα 5 εκατοστά 12 φορές
5. στροφή αριστερά 90 μοίρες
6. επανέλαβε βήμα 5 εκατοστά 8 φορές
7. στροφή αριστερά 90 μοίρες
8. επανέλαβε βήμα 5 εκατοστά 6 φορές
9. στροφή αριστερά 90 μοίρες
10. επανέλαβε βήμα 5 εκατοστά 8 φορές
11. στροφή διαγώνια 45 μοίρες δεξιά
12. επανέλαβε βήμα 5 εκατοστά 10 φορές
13. στροφή διαγώνια δεξιά 45 μοίρες
14. επανέλαβε βήμα 5 εκατοστά 6 φορές
15. στροφή διαγώνια δεξιά 45 μοίρες
16. επανέλαβε βήμα 5 εκατοστά 10 φορές

Αλγόριθμος καθοδήγηση ρομπότ

Να καθοδηγήσετε το ρομπότ από την θέση που βρίσκετε να βρεί την έξοδο

Λύση:

1. σήκω πάνω
2. πήγαινε ένα βήμα μπροστά
3. στρίψε αριστερά 90 μοίρες
4. πήγαινε 4 βήματα μπροστά
5. στρίψε αριστερά 90 μοίρες
6. πήγαινε 20 βήματα μπροστά
7. στρίψε δεξιά 90 μοίρες
8. πήγαινε 10 βήματα μπροστά
9. σήκωσε τα χέρια σου, πιάσε το χερούλι
10. κατέβασε το χερούλι προς τα κάτω
11. κάνε δύο βήματα μπροστά

Πύργοι Hanoi

Οι πύργοι του Hanoi είναι ένα μαθηματικό puzzle. Αποτελείται από τρεις κάθετες ράβδους και κάποιους στρογγυλούς δίσκους, διαφορετικού μεγέθους, οι οποίοι μπορούν να μπουν σε οποιαδήποτε ράβδο. Ξεκινώντας το puzzle, οι δίσκοι βρίσκονται τακτοποιημένοι στην αριστερή ράβδο με φθίνουσα σειρά, ως προς το μέγεθος των δίσκων, σχηματίζοντας ένα κωνικό σχήμα.

Το ζητούμενο σ' αυτό το puzzle είναι να μετακινηθεί ολόκληρη η στοίβα από την αριστερή ράβδο στην τελευταία δεξιά, με τους εξής κανόνες :

- κάθε φορά μπορείς να μετακινήσεις μόνο έναν δίσκο,
- σε κάθε κίνηση επιτρέπεται να πάρεις μόνο τον πάνω πάνω δίσκο από μια ράβδο και πρέπει να τον τοποθετήσεις στην υψηλότερη θέση στη ράβδο που θα επιλέξεις, αν υπάρχουν ήδη εκεί άλλοι δίσκοι,
- και τέλος, κανένας δίσκος δε μπορεί να τοποθετηθεί πάνω από έναν, μικρότερου μεγέθους.

Δεδομένο:

Είναι ότι μπορεί να γίνει αντιληπτό από 2 ή περισσότερους παρατηρητές με 1 από τις 5 αισθήσεις. Η επεξεργασία δεδομένων είναι μία από τις πρώτες έννοιες της Πληροφορικής. Με τον όρο αυτό εννοούμε την καταγραφή, τη διαχείριση και την ανάκτηση δεδομένων και πληροφοριών με τους υπολογιστές.

Τα δεδομένα (data), είναι γεγονότα, μηνύματα, κωδικοποιημένα ή όχι και αποτελούν ακατέργαστο πληροφοριακό υλικό. Σήμερα οι υπολογιστές μπορούν να διαχειρίζονται και επεξεργάζονται όχι μόνο δεδομένα αλλά και πλήρεις λογικές ενότητες τους, τις πληροφορίες. Για το λόγο αυτό ο όρος Επεξεργασία Δεδομένων αντικαθίσταται από τον όρο Επεξεργασία Πληροφοριών (Information Processing). Παράλληλα με τον όρο Επεξεργασία Πληροφοριών χρησιμοποιείται ευρύτατα και ο όρος Επεξεργασία και Επικοινωνία Πληροφοριών (Information and Communication Processing - ICP). Ο όρος αυτός επικρατεί γιατί εκφράζει πιστότερα τη σημερινή πραγματικότητα προσυνοδεύει την τεχνολογία επεξεργασίας των πληροφοριών με τη δυνατότητα διαθέσεώς τους μέσω των δικτύων υπολογιστών.

Πληροφορία:

Είναι το αποτέλεσμα της επεξεργασίας των δεδομένων. Η **πληροφορία (information)** είναι το αποτέλεσμα που προκύπτει από κάποια κατάλληλη επεξεργασία δεδομένων. Η πληροφορία αυξάνει τη γνώση και η διάθεσή της συμβάλει στη λήψη απόφασης. Οι πληροφορίες που παίρνουμε από την επεξεργασία δεδομένων μπορεί να χρησιμοποιηθούν ως **νέα δεδομένα** για περαιτέρω επεξεργασία και να δώσουν ως αποτελέσματα νέες πληροφορίες. Έτσι, πολλές φορές, ο όρος **επεξεργασία πληροφοριών** χρησιμοποιείται ως ταυτόσημος της επεξεργασίας δεδομένων. Μάλιστα, από τη δεκαετία του '80 ο όρος επεξεργασία πληροφοριών επικράτησε γιατί οι υπολογιστές μπορούν πλέον να επεξεργάζονται πληροφορίες αντί των στοιχειωδών δεδομένων.

Το αντικείμενο της Πληροφορικής είναι η φύλαξη, η επεξεργασία, η παραγωγή και η χρησιμοποίηση της πληροφορίας. Η πληροφορία παράγεται από τα δεδομένα και χρησιμοποιείται για τη λήψη αποφάσεων.

Ένας **υπολογιστής** είναι μία μηχανή η οποία, με τον έλεγχο ενός αποθηκευμένου σε αυτήν προγράμματος, δέχεται αυτόματα και επεξεργάζεται δεδομένα και πληροφορίες και παρέχει άλλες πληροφορίες ως αποτελέσματα αυτής της επεξεργασίας.

Σταθερά:

Κάτι που αλλάζει τιμή. Όπως οι μεταβλητές έτσι και οι σταθερές καταλαμβάνουν κάποιο χώρο στην μνήμη. Οι σταθερές διαφέρουν από τις μεταβλητές στο ότι τα περιεχόμενά τους δεν αλλάζουν κατά την εκτέλεση του προγράμματος. Οι σταθερές έχουν όνομα όπως και οι μεταβλητές και οι περιορισμοί στην ονοματολογία τους είναι ίδια με αυτή των μεταβλητών όπως αναφέραμε παραπάνω. Οι σταθερές είναι χρήσιμες όταν γράφουμε προγράμματα που έχουν κάποια βασική παράμετρο που δεν αλλάζει. Π.χ. ο αριθμός $\pi=3.14...$

Μεταβλητή:

Κάτι που δεν αλλάζει τιμή. Οι μεταβλητές χρησιμοποιούνται για να αποθηκεύουμε δεδομένα του προγράμματος και να τα χρησιμοποιήσουμε σε κάποιο άλλο σημείο τους. Οι μεταβλητές στην πραγματικότητα δείχνουν σε κάποια θέση μνήμης στον υπολογιστή που είναι αποθηκευμένα τα δεδομένα που αντιστοιχούν στην μεταβλητή. Τα ονόματα των μεταβλητών ξεκινάνε πάντα με τουλάχιστον ένα πεζό αγγλικό γράμμα. Τα κεφαλαία τα χρησιμοποιούμε αποκλειστικά να γράφουμε τις εντολές της Xbasic. Είναι απαραίτητο να γίνει αυτό γιατί εντολή που έχει έστω και ένα πεζό γράμμα δεν αναγνωρίζεται. Στα ονόματα των μεταβλητών η γλώσσα δεν μετατρέπει σε κεφαλαία ή πεζά ότι μοιάζει

και έτσι οι μεταβλητές numA και numB είναι διαφορετικές. Τα ονόματα μεταβλητών δεν μπορούν να περιέχουν σύμβολα εκτός από το την υπογράμμιση (Underscore _).

Αντικειμενοστραφής προγραμματισμός:

Είναι μία τεχνική προγραμματισμού που υπάρχει η έννοια του αντικειμένου το οποίο είναι ένα σύνολο από χαρακτηριστικά και διαδικασίες που χρειάζονται αυτά τα χαρακτηριστικά. Π.χ. αντικείμενο : κύκλος - ιδιότητες : ακτίνα, σταθερά $\pi=3,14$ - διαδικασίες : α) εύρεση περιμέτρου $\pi=2 \cdot P \cdot R$ - β) εύρεση εμβαδού : $E=P \cdot R^2$. Ποιο συγκεκριμένα Ο Αντικειμενοστραφής Προγραμματισμός (Object-Oriented Programming ή OOP για συντομία), όπως λέει και το όνομά του συνδέεται άμεσα με τα αντικείμενα και κατά συνέπεια, όπως θα δούμε και στη συνέχεια, και με τις κλάσεις. Τι είναι όμως ο Αντικειμενοστραφής Προγραμματισμός;

Κατ' ουσίαν, ο OOP είναι ένας τρόπος οργάνωσης των προγραμμάτων που γράφουμε. Ο τρόπος αυτός οργάνωσης, δεν είναι φυσικά ούτε μοναδικός, ούτε καν ο βέλτιστος. Άλλοι τρόποι οργάνωσης είναι ο *διαδικαστικός* (procedural ή imperative) και ο *συναρτησιακός* (functional). Συνηθίζεται, αυτοί οι τρόποι οργάνωσης ή τεχνικές οργάνωσης των προγραμμάτων να ονομάζονται *προγραμματιστικά παραδείγματα* (programming paradigms).

Η επιλογή του προγραμματιστικού παραδείγματος το οποίο θα χρησιμοποιήσουμε εξαρτάται από το πρόβλημα το οποίο καλούμαστε να επιλύσουμε, αλλά και από τη γλώσσα προγραμματισμού την οποία χρησιμοποιούμε. Δεν επιτρέπουν όλες οι γλώσσες προγραμματισμού τη χρήση όλων των προγραμματιστικών παραδειγμάτων. Π.χ. υπάρχουν γλώσσες που επιτρέπουν τη δημιουργία μόνο διαδικαστικών προγραμμάτων όπως η Fortran. Υπάρχουν άλλες που επιτρέπουν τη δημιουργία μόνο (ή κυρίως) συναρτησιακών προγραμμάτων, όπως η Lisp και οι πολλές παραλλαγές της. Ενώ υπάρχουν και γλώσσες που επιτρέπουν, με περισσότερη ή λιγότερη ευκολία, να εφαρμόσεις περισσότερα από ένα προγραμματιστικά παραδείγματα. Η Python ανήκει σε αυτήν την κατηγορία καθώς σου επιτρέπει να γράψεις κώδικα που χρησιμοποιεί και τα τρία προγραμματιστικά παραδείγματα.

Τμηματικός προγραμματισμός:

Είναι η τεχνική κατά την οποία το αρχικό πρόγραμμα διασπάται σε απλούστερα τμήματα έτσι ώστε η λύση κάθε τμήματος να είναι πιο απλή.

Πλεονεκτήματα τμηματικού προγραμματισμού

1. Διευκόλυνση στην ανάπτυξη του προγράμματος
2. Διευκόλυνση στην κατανόηση και διόρθωση του προγράμματος
3. Εξοικονόμηση χρόνου και προσπάθειας κατά στη συγγραφή του προγράμματος
4. Επεκτείνει τις δυνατότητες των γλωσσών προγραμματισμού.

Ιεραρχικός προγραμματισμός:

Είναι η τεχνική κατά την οποία το πρόγραμμα σχεδιάζεται και υλοποιείται από πάνω προς τα κάτω. Κατά την διαδικασία αυτή δημιουργούνται απλούστερα τμήματα. Η σχεδίαση προχωρά από πάνω προς τα κάτω σε επίπεδα. Ξεκινάμε από μια σύντομη δήλωση του τι κάνει το πρόγραμμα στο ψηλότερο επίπεδο. Σε κάθε επόμενο επίπεδο αναλύουμε την κάθε επεξεργασία σε μικρότερες και πιο λεπτομερειακές. Τμηματικός προγραμματισμός. Οι διάφορες επεξεργασίες είναι τμήματα προγράμματος:

- Λογικά
- Πλήρη
- Ανεξάρτητα μεταξύ τους
- Όσο το δυνατόν μικρά
- Μια είσοδο και μια έξοδο

Δομημένος προγραμματισμός:

Είναι μια τεχνική προγραμματισμού που στηρίζεται στις 3 δομές ακολουθία, επιλογή, και επανάληψη και χρησιμοποιεί τις αρχές του ιεραρχικού και τμηματικού προγραμματισμού. Διευκόλυνση στην ανάπτυξη του αλγόριθμου, έτσι ώστε να αποτελείται από ανεξάρτητα τμήματα. Ευκολία και ταχύτητα στην κωδικοποίηση. Καλύτερη ποιότητα των προγραμμάτων. Ευκολία στις διορθώσεις και τη συντήρηση. Τεκμηρίωση σχεδόν εξ' ολοκλήρου στο ίδιο το πρόγραμμα.

Πλεονεκτήματα δομημένου προγραμματισμού

1. Δημιουργία απλούστερων προγραμμάτων
2. Άμεση μεταφορά των αλγορίθμων σε προγράμματα
3. Διευκόλυνση της ανάλυσης του προγράμματος κατά τμήματα
4. Διευκόλυνση στην ανάγνωση του προγράμματος από τρίτους
5. Περιορισμός των λαθών κατά την ανάπτυξη του προγράμματος
6. Ευκολότερη διόρθωση και συντήρηση του προγράμματος.

Διαφορές προγραμματισμών

{Ιεραρχικός-Δομημένος-Τμηματικός-Αντικειμενοστραφής}

Ο ιεραρχικός προγραμματισμός διασπά το πρόβλημα σε απλούστερα όπως επίσης και ο δομημένος διασπά το πρόβλημα σε ακόμα πιο απλούστερα και τα αναλύει όσο πιο πολύ μπορεί για να επιλυθεί το πρόβλημα. Επίσης ο δομημένος προγραμματισμός βασίζεται στην αρχή του “διαίρει και βασίλευει”, σαν τον τμηματικό προγραμματισμός όπου αποτελείται από ένα σύνολο τμημάτων τα οποία επιλύουν τα επιμέρους πρόβλήματα του αρχικού προβλήματος Ένω ο αντικειμενοστραφής προγραμματισμός είναι ένας τρόπος οργάνωσης των προγραμμάτων που κάνουμε και διασπά την έννοια σε αντικείμενα.