

Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον § 2.1, 2.2, 2.3, 7

1. Τι είναι **αλγόριθμος**; (σ.33)
(Μια πεπερασμένη σειρά ενεργειών, αυστηρά καθορισμένων και εκτελέσιμων σε πεπερασμένο χρόνο, που στοχεύουν στην επίλυση ενός προβλήματος)
2. Ποια **κριτήρια** πρέπει να ικανοποιεί ο αλγόριθμος; (α. είσοδος, β. έξοδος, γ. καθοριστικότητα (απόλυτα καθορισμένες εντολές, όλες οι πιθανές περιπτώσεις), δ. περατότητα (πεπερασμένα βήματα, να τελειώνουν σε πεπερασμένο χρόνο), ε. αποτελεσματικότητα (κάθε εντολή απλή και εκτελέσιμη))
3. Υπό ποία πρίσματα η Πληροφορική επιστήμη μελετά τους αλγορίθμους;
 - 1) **Υλικού (hardware)** : Η ταχύτητα εκτέλεσης ενός αλγορίθμου επηρεάζεται από τις διάφορες τεχνολογίες υλικού.
 - 2) **Γλωσσών Προγραμματισμού (programming languages)** : Το είδος της γλώσσας προγραμματισμού που χρησιμοποιείται (δηλαδή, χαμηλότερου ή υψηλότερου επιπέδου) αλλάζει τη δομή και τον αριθμό των εντολών ενός αλγορίθμου.
 - 3) **Θεωρητική (theoretical)** : Υπάρχει πράγματι ή όχι κάποιος αποδοτικός αλγόριθμος για την επίλυση ενός προβλήματος; Η απάντηση απαιτεί μεγάλη θεωρητική κατάρτιση.
 - 4) **Αναλυτική (analytical)** : Μελετώνται οι υπολογιστικοί πόροι (computer resources) που απαιτούνται από έναν αλγόριθμο,
4. **Τρόποι αναπαράστασης** αλγορίθμων (σ.35)
Ελεύθερο κείμενο (πιθανό πρόβλημα αποτελεσματικότητας),
Διαγραμματικές τεχνικές (όχι πρακτικά σε μεγάλους αλγόριθμους),
Φυσική γλώσσα **κατά βήματα** (πιθανό πρόβλημα καθοριστικότητας) ,
Κωδικοποίηση (ψευδογλώσσα ή γλώσσα προγραμματισμού)
5. Τι είναι το **διάγραμμα ροής** (σ.36)
(σύνολο γεωμετρικών σχημάτων που το καθένα δηλώνει μια ενέργεια και βέλη που δηλώνουν τη σειρά εκτέλεσης)
6. Ποια **σύμβολα** χρησιμοποιούν τα διαγράμματος ροής (σ.36)
(έλλειψη για αρχή και τέλος, ρόμβος για ερωτήσεις, ορθογώνιο για εκτέλεση πράξεων, πλάγιο παραλληλόγραμμο για είσοδο ή έξοδο δεδομένων.
7. Τι ονομάζεται **εντολή** και ποια **είδη** διακρίνουμε; (σ.37)
(μια λέξη της χρησιμοποιούμενης ψευδογλώσσας, που προσδιορίζει μια σαφή ενέργεια.)
Είδη: εκτελεστές πχ: Διάβασε, δηλωτικές πχ. Αλγόριθμος.
8. Τι είναι **σταθερές**; (σ.37,129)
(είναι προκαθορισμένες αμετάβλητες τιμές που αντιστοιχούνται με ένα όνομα (π.χ $\pi=3.14$) και έπειτα να χρησιμοποιούνται με το όνομά τους. π.χ. $E_{\pi} \leftarrow \pi * R^2$ (Σταθερές ονομάζονται και οι αμετάβλητες τιμές που χρησιμοποιούνται μέσα σε μια έκφραση (π.χ. $E_{\pi} \leftarrow 3.14 * R^2$)).
9. **Τι χρησιμεύουν οι σταθερές**;
(αυτές που έχουν αντιστοιχηθεί με ονόματα) (σ.149) (πιο κατανοητό πρόγραμμα, πιο εύκολο να συντηρηθεί (γιατί αλλάζω την τιμή μιας σταθεράς εύκολα σε ένα σημείο και δεν ψάχνω παντού μέσα στο πρόγραμμα))
10. Τι είναι οι **μεταβλητές** (σ.37,129)
(ένα δεδομένο-μια ποσότητα που συμβολίζεται με ένα όνομα και περιέχει μια τιμή. Η τιμή του μπορεί να αλλάξει)
11. Τι είναι οι **δεσμευμένες** λέξεις; (σ.129)
(οι εντολές και οι άλλες λέξεις που χρησιμοποιεί μια γλώσσα προγραμματισμού) (π.χ. ΔΙΑΒΑΣΕ, ΓΡΑΨΕ, ΑΝ, ΓΙΑ...)
12. Ποιους **κανόνες** πρέπει να **πληροί ένα όνομα** (σ.129)
(Αποτελείται από γράμματα, αριθμούς και την κάτω παύλα (_), αρχίζουν με γράμμα, δεν συμπίπτουν με το όνομα μιας δεσμευμένης λέξης. Επίσης καλό το όνομα να ταιριάζει με το περιεχόμενό του)
13. Ποιοι είναι οι **τύποι** των δεδομένων-μεταβλητών; Τι τιμές παίρνουν; Δώστε παραδείγματα.
(σ37, 128) (ακέραιες, πραγματικές, (στους αλγόριθμους τις λέμε και τις δύο *αριθμητικές* χωρίς διάκριση), χαρακτήρες, λογικές)

14. Τι είναι οι **τελεστές**;
(σ37) (είναι τα σύμβολα των πράξεων)
15. Τι είναι οι **τελεσταίοι**;
(σ37) (οι σταθερές και οι μεταβλητές που μαζί με τους τελεστές δημιουργούν μια έκφραση)
16. Ποιες **κατηγορίες τελεστών** γνωρίζετε και ποιους τελεστές περιέχουν; (σ37,131,139) (α. αριθμητικοί: +, -, *, /, ^, DIV, MOD, β. λογικοί: ΚΑΙ, Η, ΟΧΙ, γ. συγκριτικοί: <, >, <=, >=, =, <> (≥, ≤, ≠ : χρήση μόνο στη ψευδογλώσσα, όχι σε Πρόγραμμα)
17. Ποια είναι η **προτεραιότητα των τελεστών**; Όταν αριθμητικοί και συγκριτικοί τελεστές συνδυάζονται σε μια έκφραση, οι αριθμητικές πράξεις εκτελούνται πρώτες. Ακόμη, οι λογικοί τελεστές έχουν χαμηλότερη ιεραρχία από τους συγκριτικούς.
18. Με ποια σειρά - **προτεραιότητα** γίνονται οι πράξεις (ή ποια η ιεραρχία των τελεστών) (σ.132)
(α. Για τους αριθμητικούς έχουμε: 1.^, 2. *, /, DIV, MOD, 3. +, - Αν έχουν ίδια προτεραιότητα γίνονται από αριστερά προς τα δεξιά. Οι πράξεις μέσα στις παρενθέσεις προηγούνται .
β. Για τους λογικούς: 1.ΟΧΙ, 2.ΚΑΙ, 3.Η)
19. Πώς διαμορφώνονται οι **εκφράσεις** και πώς παίρνουν τιμές;
(σ. 37) Οι εκφράσεις διαμορφώνονται από τους τελεστέους και από τους τελεστές. Η διεργασία αποτίμησης μιας έκφρασης συνίσταται στην απόδοση τιμών στις μεταβλητές και στην εκτέλεση των πράξεων. Η τελική τιμή μιας έκφρασης εξαρτάται από την ιεραρχία των πράξεων και τη χρήση των παρενθέσεων. Μια έκφραση μπορεί να αποτελείται από μια μόνο μεταβλητή ή σταθερά μέχρι μια πολύπλοκη μαθηματική παράσταση.
20. Τι είναι τα **σχόλια**;
(επεξηγήσεις και διευκρινήσεις που γράφονται μέσα στο πρόγραμμα για την καλύτερη κατανόηση και τεκμηρίωσή του...)
21. Ποιες είναι οι **στοιχειώδεις λογικές δομές εντολών** που χρησιμοποιούμε στο δομημένο προγραμματισμό και τι εντολές περιλαμβάνουν; (σ.30-32-35-39, 135)
(α. δομή ακολουθίας (διάβασε, γράψε, υπολόγισε),
β. δομή επιλογής-πολλαπλής επιλογής (αν...αλλιώς_αν, επίλεξε),
γ. δομή επανάληψης (για, όσο, μέχρι))
22. Τι είναι **λογικές** πράξεις; (σ.43) (οι πράξεις που γίνονται σε μια σύνθετη λογική έκφραση όπου γίνεται συνδυασμός κριτηρίων με τους λογικούς τελεστές Η (διάζευξη), ΚΑΙ (σύζευξη), ΟΧΙ (άρνηση))
23. Τι αποτέλεσμα έχουν οι λογικές πράξεις Η (διάζευξη), ΚΑΙ (σύζευξη), ΟΧΙ (άρνηση) σε όλους τους συνδυασμούς τους; (σ.43 - πίνακάκι)

A	B	A και B	A ή B	Όχι A
Αληθής	Αληθής	Αληθής	Αληθής	Ψευδής
Αληθής	Ψευδής	Ψευδής	Αληθής	Ψευδής
Ψευδής	Αληθής	Ψευδής	Αληθής	Αληθής
Ψευδής	Ψευδής	Ψευδής	Ψευδής	Αληθής

24. Ποιες γνωστές **συναρτήσεις** από τα μαθηματικά χρησιμοποιούνται και περιέχονται στη ΓΛΩΣΣΑ.

- Οι συναρτήσεις αυτές είναι:
- ΗΜ(Χ) Υπολογισμός ημιτόνου
- ΣΥΝ(Χ) Υπολογισμός συνημιτόνου
- ΕΦ(Χ) Υπολογισμός εφαπτομένης
- Τ_Ρ(Χ) Υπολογισμός τετραγωνικής ρίζας
- ΛΟΓ(Χ) Υπολογισμός φυσικού λογαρίθμου
- Ε(Χ) Υπολογισμός του e^x

- $A_M(X)$ Ακέραιο μέρος του X
- $A_T(X)$ Απόλυτη τιμή του X

- οι συναρτήσεις $HM()$, $ΣΥΝ()$ και $ΕΦ()$ δέχονται παράμετρο σε μοίρες
- το ακέραιο μέρος $A_M()$ ενός αριθμού x ορίζεται, όπως στα μαθηματικά ο ακέραιος, με την ιδιότητα $A_M(x) \leq x < A_M(x) + 1$,
- η απόλυτη τιμή $A_T()$ δέχεται αριθμητική παράμετρο, είτε ακέραιο αριθμό και επιστρέφει ακέραιο, είτε πραγματικό αριθμό και επιστρέφει πραγματικό

25. Ποια η σύνταξη και η λειτουργία της εντολής εκχώρησης

Σύνταξη ΌνομαΜεταβλητής <- έκφραση
π.χ. $A \leftarrow 132$

Υπολογίζεται η τιμή της έκφρασης στη δεξιά πλευρά και εκχωρείται η τιμή αυτή στη μεταβλητή, που αναφέρεται στην αριστερή πλευρά. Σε μια εντολή εκχώρησης η μεταβλητή και η έκφραση πρέπει να είναι του ιδίου τύπου.

26. Ποια η σύνταξη και η λειτουργία της εντολής ΔΙΑΒΑΣΕ

Σύνταξη: ΔΙΑΒΑΣΕ λίσταΜεταβλητών
π.χ. ΔΙΑΒΑΣΕ Ποσότητα, Τιμή

Η εκτέλεση της εντολής οδηγεί στην είσοδο τιμών από το πληκτρολόγιο και την εκχώρησή τους στις μεταβλητές που αναφέρονται.

Η εντολή ΔΙΑΒΑΣΕ ακολουθείται πάντοτε από ένα ή περισσότερα ονόματα μεταβλητών. Αν υπάρχουν περισσότερες από μία μεταβλητές, τότε αυτές χωρίζονται με κόμμα (,).

Κατά την εκτέλεση του προγράμματος η εντολή ΔΙΑΒΑΣΕ διακόπτει την εκτέλεσή του και το πρόγραμμα περιμένει την εισαγωγή από το πληκτρολόγιο τιμών, που θα εκχωρηθούν στις μεταβλητές. Μετά την ολοκλήρωση της εντολής η εκτέλεση του προγράμματος συνεχίζεται με την επόμενη εντολή.

27. Ποια η σύνταξη και η λειτουργία της εντολής ΓΡΑΨΕ.

Σύνταξη ΓΡΑΨΕ λίστα-στοιχείων
π.χ. ΓΡΑΨΕ 'Η τετραγωνική ρίζα του', A , είναι: 'PIZA

Χρησιμοποιείται για την εμφάνιση σταθερών τιμών καθώς και των τιμών των μεταβλητών που αναφέρονται στη λίστα. Η εντολή ΓΡΑΨΕ έχει ως αποτέλεσμα την εμφάνιση τιμών στη μονάδα εξόδου. Συσκευή εξόδου μπορεί να είναι η οθόνη του υπολογιστή, ο εκτυπωτής, βοηθητική μνήμη ή γενικά οποιαδήποτε συσκευή εξόδου έχει οριστεί στο πρόγραμμα. Η λίστα των στοιχείων μπορεί να περιέχει σταθερές τιμές και ονόματα μεταβλητών.

28. Περιγράψτε τη δομή ενός προγράμματος

Η πρώτη εντολή κάθε προγράμματος είναι υποχρεωτικά η επικεφαλίδα του προγράμματος, η οποία είναι η λέξη **ΠΡΟΓΡΑΜΜΑ** ακολουθούμενη από το όνομα του προγράμματος. Το τελευταίο πρέπει να υπακούει στους κανόνες δημιουργίας ονομάτων της ΓΛΩΣΣΑΣ. Στη συνέχεια ακολουθεί το τμήμα δήλωσης των **σταθερών** του προγράμματος, αν βέβαια το πρόγραμμά μας χρησιμοποιεί σταθερές.

Αμέσως μετά είναι το τμήμα δήλωσης **μεταβλητών**, όπου δηλώνονται υποχρεωτικά τα ονόματα όλων των μεταβλητών καθώς και ο τύπος τους.

Ακολουθεί το κύριο μέρος του προγράμματος, που περιλαμβάνει όλες τις εκτελέσιμες εντολές. Οι εντολές αυτές περιλαμβάνονται υποχρεωτικά ανάμεσα στις λέξεις **ΑΡΧΗ** και **ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ**.

Αν το πρόγραμμα χρησιμοποιεί διαδικασίες αυτές γράφονται μετά το **ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ**.

29. Ποια η λειτουργία των **χαρακτήρων !** και **&** στη σύνταξη των προγραμμάτων

Κάθε εντολή γράφεται σε ξεχωριστή γραμμή. Αν μία εντολή πρέπει να συνεχιστεί και στην επόμενη γραμμή, τότε ο πρώτος χαρακτήρας αυτής της γραμμής πρέπει να είναι ο χαρακτήρας **&**.

Αν ο πρώτος χαρακτήρας είναι το θαυμαστικό (!), σημαίνει ότι αυτή η γραμμή περιέχει σχόλια και όχι εκτελέσιμες εντολές.