

6ο Κεφάλαιο

ΕΡΩΤΗΣΕΙΣ ΑΝΑΠΤΥΞΗΣ

1. Ποια είναι τα στάδια επίλυσης ενός προβλήματος με υπολογιστή;
2. Τι είναι μια γλώσσα προγραμματισμού;
3. Τι είναι οι γλώσσες μηχανής;
4. Τι είναι γλώσσα χαμηλού επιπέδου ; Πώς ονομάζεται το πρόγραμμα μετατροπής μίας γλώσσας χαμηλού επιπέδου σε γλώσσα μηχανής ;
5. Ποια είναι τα μειονεκτήματα των συμβολικών γλωσσών ή γλωσσών χαμηλού επιπέδου;
6. Τι είναι γλώσσα υψηλού επιπέδου ; Ποια τα πλεονεκτήματά τους ; Αναφέρατε τρεις γλώσσες υψηλού επιπέδου
7. Για τι είναι κατάλληλη η FORTRAN;
8. Σε τι προσανατολίζεται η COBOL;
9. Η ALGOL είναι γλώσσα γενικής φύσης ή συνδέεται με συγκεκριμένες εφαρμογές;
10. Η PL/1 είναι γλώσσα γενικής φύσης ή επιτρέπει παράλληλο προγραμματισμό;
11. Τι γνωρίζετε για τις γλώσσες προγραμματισμού LISP, PROLOG;
12. Η LISP είναι αντικειμενοστραφής γλώσσα ή προσανατολίζεται σε χειρισμό λιστών από σύμβολα;
13. Η PROLOG προσανατολίζεται σε εμπορικές εφαρμογές ή σε προβλήματα τεχνητής νοημοσύνης;
14. Τι γνωρίζετε για τη γλώσσα προγραμματισμού BASIC;
15. Τι γνωρίζετε για τη γλώσσα προγραμματισμού PASCAL;
16. Για τι είναι κατάλληλη η PASCAL;
17. Η C είναι γλώσσα προγραμματισμού συστημάτων ή χειρισμού βάσεων δεδομένων;
18. Τι ονομάζουμε οπτικό προγραμματισμό
19. Τι ονομάζουμε προγραμματισμό οδηγούμενο από το γεγονός;
20. Ποια είναι τα πλεονεκτήματα της χρήσης γλωσσών υψηλού επιπέδου;
21. Τι είναι η μεταφερισιμότητα;
22. Τι γνωρίζετε για τις γλώσσες 4ης γενιάς;
23. Πώς προσδιορίζεται μία φυσική ή τεχνητή γλώσσα; Δώστε ένα παράδειγμα.
24. Να δοθεί ο ορισμός του αλφαβήτου, του λεξιλογίου, της γραμματικής και της σημασιολογίας μιας γλώσσας.
25. Ποιες είναι οι κυριότερες διαφορές των φυσικών και των τεχνητών γλωσσών;
26. Τι ονομάζεται ιεραρχική σχεδίαση προγράμματος;
27. Η εντολή GOTO χρειάζεται στο δομημένο προγραμματισμό ή είναι περιττή;
28. Δώστε τον ορισμό του δομημένου προγραμματισμού. Ποια είναι τα πλεονεκτήματά του;
29. Ποια είναι τα είδη λαθών που εμφανίζονται σ' ένα πρόγραμμα κατά τη διαδικασία ανάπτυξής του;
30. Τι είναι τα συντακτικά και τι τα λογικά λάθη;
31. Ποιες είναι οι διαφορές μεταγλωττιστή και διερμηνευτή;
32. Ποια προγράμματα και εργαλεία περιέχει ένα προγραμματιστικό περιβάλλον;
33. Τι είναι ένας μεταγλωττιστής;
34. Περιγράψτε τα στάδια μεταγλώττισης και σύνδεσης ενός προγράμματος από το αρχικό πρόγραμμα ως το εκτελέσιμο
35. Τι είναι ένας διερμηνευτής;
36. Ποια η διαφορά του μεταγλωττιστή και του διερμηνευτή;
37. Πότε έχουμε πηγαίο, αντικείμενο και εκτελέσιμο πρόγραμμα ;
38. Περιγράψτε τη διαδικασία μετάφρασης και εκτέλεσης ενός προγράμματος σε έναν διερμηνευτή
39. Πως λέγεται το ειδικό πρόγραμμα που χρησιμοποιείται για τη συγγραφή του προγράμματος από τον προγραμματιστή
40. Τι είναι ο συντάκτης και τι ο συνδέτης;

ΕΡΩΤΗΣΕΙΣ ΚΡΙΣΕΩΣ

1. Οι γλώσσες προγραμματισμού αναπτύχθηκαν με σκοπό την επικοινωνία ανθρώπου - μηχανής
2. Ένα πρόγραμμα γραμμένο σε γλώσσα μηχανής είναι εντολές προς τον επεξεργαστή για στοιχειώδεις λειτουργίες

3. Ένα πρόγραμμα σε γλώσσα μηχανής χρειάζεται μετατροπή σε ακολουθία δυαδικών ψηφίων ώστε να εκτελεστεί από τον υπολογιστή
4. Μια ακολουθία εντολών 0, 1 είναι κατανοητή από τον επεξεργαστή και μπορεί να εκτελεστεί χωρίς μετατροπή
5. Ο προγραμματισμός σε γλώσσα μηχανής ήταν μια εξαιρετικά δύσκολη δουλειά που ελάχιστοι μπορούσαν να πραγματοποιήσουν
6. Ένα πρόγραμμα σε συμβολική γλώσσα ή γλώσσα χαμηλού επιπέδου τελικά μετατρέπεται σε γλώσσα μηχανής
7. Η συμβολική γλώσσα είναι μια ακολουθία 0 και 1
8. Οι εντολές σε συμβολική γλώσσα αποτελούνται από συμβολικά ονόματα που αντιστοιχούν σε εντολές της γλώσσας μηχανής
9. Οι συμβολικές γλώσσες έφεραν την ανεξαρτησία από την αρχιτεκτονική κάθε υπολογιστή
10. Τα προγράμματα σε γλώσσες χαμηλού επιπέδου είναι εύκολο να γραφτούν και να συντηρηθούν
11. Τα προγράμματα σε γλώσσες υψηλού επιπέδου είναι ανεξάρτητα του υπολογιστή που αναπτύχθηκαν
12. Οι γλώσσες υψηλού επιπέδου ήρθαν να επιλύσουν τις αδυναμίες των συμβολικών γλωσσών για καλύτερη επικοινωνία ανθρώπου - μηχανής
13. Η Fortran είναι γλώσσα χαμηλού επιπέδου
14. Η Fortran είναι κατάλληλη για την επίλυση μαθηματικών και επιστημονικών προβλημάτων
15. Ένα πρόγραμμα γλώσσας υψηλού επιπέδου μπορεί να εκτελεστεί σε οποιονδήποτε υπολογιστή ανεξάρτητα της αρχιτεκτονικής
16. Δεν έχει αναπτυχθεί γλώσσα υψηλού επιπέδου που να επιλύει όλα τα είδη προβλημάτων
17. Η COBOL είναι γλώσσα προσανατολισμένη στην ανάπτυξη εμπορικών εφαρμογών
18. Η COBOL δεν μπορεί να επιλύσει μαθηματικά προβλήματα
19. Η Algol είναι μια γλώσσα γενικού σκοπού αλλά με ελάχιστη πρακτική εφαρμογή
20. Η Algol αναπτύχθηκε για την επίλυση μαθηματικών προβλημάτων
21. Η PL/1 είναι γλώσσα υψηλού επιπέδου γενικής χρήσης με ελάχιστη επιτυχία
22. Η PL/1 προσπάθησε να συνθέσει Fortran και COBOL
23. Η Lisp και η Prolog είναι γλώσσες προγραμματισμού που χρησιμοποιούνται στον τομέα της Τεχνητής Νοημοσύνης
24. Η Basic είναι γλώσσα γενικού σκοπού με έμφαση στην εκπαίδευση αρχαρίων στον προγραμματισμό
25. Η γλώσσα Pascal είναι γλώσσα γενικού σκοπού και κατάλληλη τόσο για την εκπαίδευση όσο και για τη δημιουργία ισχυρών προγραμμάτων
26. Η C χρησιμοποιήθηκε για την ανάπτυξη συστημάτων και έχει πολλές δυνατότητες χαμηλού επιπέδου
27. Η C++ είναι γλώσσα τέταρτης γενιάς
28. Η C++ και οι Java είναι αντικειμενοστραφείς γλώσσες
29. Η Java είναι κατάλληλη για την ανάπτυξη εφαρμογών που εκτελούνται δικτυακά
30. Ο οδηγούμενος από τα γεγονότα προγραμματισμός και οι γλώσσες οπτικού προγραμματισμού αξιοποιούν τα γραφικά χαρακτηριστικά του υπολογιστή (ποντίκι, μενού κ.λ.π.)
31. Χάρη στο δομημένο προγραμματισμό δημιουργούνται προγράμματα απλούστερα, κατανοήσιμα και εύκολα στη διόρθωση
32. Στο δομημένο προγραμματισμό ακολουθούνται οι αρχές του ιεραρχικού και του τμηματικού προγραμματισμού
33. Στην ιεραρχική σχεδίαση, η ανάλυση του αλγορίθμου πραγματοποιείται με την τεχνική «από πάνω προς τα κάτω»
34. Η μεταφοριστικότητα είναι χαρακτηριστικό των γλωσσών χαμηλού επιπέδου.
35. Η ανεξαρτησία των γλωσσών προγραμματισμού υψηλού επιπέδου από τον τύπο του υπολογιστή αποτελεί ένα από τα πλεονεκτήματά τους.
36. Οι εντολές των γλωσσών υψηλού επιπέδου είναι κοντύτερα στον ανθρώπινο τρόπο έκφρασης
37. Η εκμάθηση μιας γλώσσας χαμηλού επιπέδου είναι εύκολη
38. Οι γλώσσες 4ης γενιάς χρησιμοποιούνται σε βάσεις δεδομένων
39. Η καλύτερη γλώσσα προγραμματισμού είναι η C++
40. Η επιλογή της καλύτερης γλώσσας προγραμματισμού εξαρτάται από το είδος της εφαρμογής
41. Η επιλογή της καλύτερης γλώσσας προγραμματισμού εξαρτάται μόνο από το είδος της εφαρμογής

42. Μια γλώσσα προγραμματισμού προσδιορίζεται από: το αλφάβητό της, το λεξιλόγιό της, τη γραμματική της και τη σημασιολογία της
43. Δεν υπάρχουν διαφορές μεταξύ φυσικών και τεχνητών γλωσσών
44. Ο τμηματικός προγραμματισμός υλοποιεί την φιλοσοφία της ιεραρχικής σχεδίασης
45. Στο δομημένο προγραμματισμό η επίλυση ενός προβλήματος σχεδιάζεται ιεραρχικά και το πρόγραμμα υλοποιείται τμηματικά.
46. Ο δομημένος προγραμματισμός προϋποθέτει μόνο τη χρήση των δομών ακολουθίας, επιλογής, επανάληψης.
47. Ο διερμηνευτής (interpreter) μετατρέπει το εκτελέσιμο πρόγραμμα (executable) σε πρόγραμμα αντικείμενο (object)
48. Ο μεταγλωττιστής μας επιτρέπει να συντάσσουμε ένα πρόγραμμα
49. Ο μεταγλωττιστής αναλαμβάνει το ρόλο της μετάφρασης από το πηγαίο πρόγραμμα στο εκτελέσιμο πρόγραμμα
50. Ο διερμηνευτής ελέγχει και μετατρέπει μια-μια τις εντολές του πηγαίου προγράμματος σε γλώσσα μηχανής
51. Ο συνδέτης μετατρέπει το πηγαίο πρόγραμμα σε πρόγραμμα βιβλιοθήκης
52. Το αντικείμενο πρόγραμμα είναι γλώσσα μηχανής
53. Ο συνδέτης είναι ένα πρόγραμμα ελέγχου των συντακτικών λαθών του πηγαίου προγράμματος
54. Τα λογικά λάθη εντοπίζονται από έναν μεταγλωττιστή αλλά όχι από έναν διερμηνευτή
55. Τα συντακτικά λάθη επιδιορθώνονται γρήγορα και εύκολα αλλά για τα λογικά λάθη η επιδιόρθωση είναι μια επίπονη διαδικασία
56. Πλεονέκτημα του διερμηνευτή είναι ότι το πρόγραμμα εκτελείται γρηγορότερα
57. Τα σύγχρονα προγραμματιστικά περιβάλλοντα χρησιμοποιούν μικτές υλοποιήσεις διερμηνευτή και μεταγλωττιστή
58. Σ' ένα σύγχρονο προγραμματιστικό περιβάλλον απαιτείται η παρουσία συντάκτη για την συγγραφή του κειμένου των εντολών
59. Για την επιδιόρθωση των λογικών λαθών πολλές φορές ο προγραμματιστής καλείται να εκτελέσει το πρόγραμμά του επανειλημμένα
60. Σ' ένα σύγχρονο προγραμματιστικό περιβάλλον με μεταγλωττιστή η ύπαρξη του συνδέτη είναι προαιρετική
61. Οι βιβλιοθήκες χρησιμεύουν στη διόρθωση των συντακτικών λαθών.
62. Η παράλειψη μιας εντολής Τέλος_Επανάληψης είναι λογικό λάθος
63. Η χρήση της εντολής $MO \leftarrow \alpha + \beta + \gamma / 3$ αντί της $MO \leftarrow (\alpha + \beta + \gamma) / 3$ είναι συντακτικό λάθος
64. Τα λογικά λάθη ενός προγράμματος εμφανίζονται κατά τη μεταγλώττιση
65. Ο εντοπισμός των συντακτικών λαθών σε ένα πρόγραμμα γίνεται από τον μεταγλωττιστή (compiler)
66. Στον δομημένο προγραμματισμό χρησιμοποιούμε:
 - α) την εντολή goto
 - β) μόνο τις βασικές αλγοριθμικές δομές
 - γ) τις δομές δεδομένων
 - δ) τις βασικές αλγοριθμικές δομές και σπανίως την εντολή goto
67. Κατά την ανάλυση, ο κατακερματισμός ενός αλγορίθμου σε απλούστερους αποτελεί χαρακτηριστικό:
 - α) του τμηματικού προγραμματισμού
 - β) του ιεραρχικού προγραμματισμού
 - γ) της κλασσικής ανάπτυξης
 - δ) της υλοποίησης με την μέθοδο «από κάτω προς τα πάνω»
68. Χαρακτηριστικό του οπτικού προγραμματισμού είναι:

Α. Επιτρέπει τη γραφική δημιουργία του περιβάλλοντος προγράμματος σε τμήματα	Β. Επιτρέπει την ανάπτυξη του προγράμματος σε τμήματα
Γ. Είναι ταχύτερος στην εκτέλεση των προγραμμάτων	Δ. Επιτρέπει την διαγραμματική παράσταση της σχεδίασης
69. Η Basic είναι

Α. Κατάλληλη για εφαρμογές τεχνητής νοημοσύνης	Β. Υποστηρίζει την ανάπτυξη παράλληλου προγραμματισμού
Γ. Μία γλώσσα γενικής χρήσης	Δ. Κατάλληλη μόνο για εκπαίδευση
70. Από τον συντάκτη παράγεται:
 - α) Τα αντικείμενα
 - β) ο εκτελέσιμος κώδικας
 - γ) ο πηγαίος κώδικας
 - δ) τίποτα από τα παραπάνω
71. Να συνδέσετε τα στοιχεία της στήλης Α με τα στοιχεία των στηλών Β και Γ

ΟΝΟΜΑ ΓΛΩΣΣΑΣ	ΤΟΜΕΑΣ ΕΦΑΡΜΟΓΩΝ	ΕΙΔΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ
1. Fortran 2. Cobol 3. Algol 4. Prolog 5. Lisp 6. Pascal 7. Basic 8. C 9. C++ 10. Java 11. PL/1	I. Επιστημονικός II. Εμπορικός III. Επιστημονικός και Εμπορικός IV. Προγραμματισμός συστημάτων V. Προγραμματισμός στο διαδίκτυο VI. Γενικής χρήσης VII. Τεχνητής νοημοσύνης	A. Συναρτησιακός B. Αντικειμενοστραφής C. Μη διαδικασιακός D. Διαδικασιακός

72. Να αντιστοιχίσετε τα στοιχεία της στήλης Α με τα στοιχεία της στήλης Β

Στήλη Α	Στήλη Β
Α. Γλώσσα μηχανής Β. Μεταγλωττιστής Γ. Συμβολικές γλώσσες Δ. Συνδέτης - Φορτωτής Ε. Γλώσσες υψηλού επιπέδου	1. Basic, Pascal, C 2. Συμβολομεταφραστής 3. Αντικείμενο πρόγραμμα 4. Βιβλιοθήκες 5. Ακολουθίες από 0 και 1

73. Τα βήματα για τη δημιουργία ενός προγράμματος είναι: (βάλτε στη σωστή σειρά)
 α) συντάκτης β) συνδέτης γ) πηγαίο πρόγραμμα δ) μεταγλωττιστής ε) εκτελέσιμο πρόγραμμα στ) αντικείμενο πρόγραμμα