

6ο Κεφάλαιο

ΕΡΩΤΗΣΕΙΣ ΑΝΑΠΤΥΞΗΣ

1. Πώς προσδιορίζεται μία φυσική ή τεχνητή γλώσσα; Δώστε ένα παράδειγμα.
2. Να δοθεί ο ορισμός του αλφαβήτου, του λεξιλογίου, της γραμματικής και της σημασιολογίας μιας γλώσσας.
3. Ποιες είναι οι κυριότερες διαφορές των φυσικών και των τεχνητών γλωσσών;
4. Τι ονομάζεται ιεραρχική σχεδίαση προγράμματος;
5. Η εντολή GOTO χρειάζεται στο δομημένο προγραμματισμό ή είναι περιττή;
6. Δώστε τον ορισμό του δομημένου προγραμματισμού. Ποια είναι τα πλεονεκτήματά του;
7. Ποια είναι τα είδη λαθών που εμφανίζονται σ' ένα πρόγραμμα κατά τη διαδικασία ανάπτυξής του;
8. Τι είναι τα συντακτικά και τι τα λογικά λάθη;
9. Ποιες είναι οι διαφορές μεταγλωττιστή και διερμηνευτή;
10. Ποια προγράμματα και εργαλεία περιέχει ένα προγραμματιστικό περιβάλλον;
11. Τι είναι ένας μεταγλωττιστής;
12. Περιγράψτε τα στάδια μεταγλώττισης και σύνδεσης ενός προγράμματος από το αρχικό πρόγραμμα ως το εκτελέσιμο
13. Τι είναι ένας διερμηνευτής;
14. Ποια η διαφορά του μεταγλωττιστή και του διερμηνευτή;
15. Πότε έχουμε πηγαίο, αντικείμενο και εκτελέσιμο πρόγραμμα ;
16. Περιγράψτε τη διαδικασία μετάφρασης και εκτέλεσης ενός προγράμματος σε έναν διερμηνευτή
17. Πως λέγεται το ειδικό πρόγραμμα που χρησιμοποιείται για τη συγγραφή του προγράμματος από τον προγραμματιστή
18. Τι είναι ο συντάκτης και τι ο συνδέτης;

ΕΡΩΤΗΣΕΙΣ ΚΡΙΣΕΩΣ

1. Χάρη στο δομημένο προγραμματισμό δημιουργούνται προγράμματα απλούστερα, κατανοήσιμα και εύκολα στη διόρθωση
2. Στο δομημένο προγραμματισμό ακολουθούνται οι αρχές του ιεραρχικού και του τμηματικού προγραμματισμού
3. Στην ιεραρχική σχεδίαση, η ανάλυση του αλγορίθμου πραγματοποιείται με την τεχνική «από πάνω προς τα κάτω»
4. Μια γλώσσα προγραμματισμού προσδιορίζεται από: το αλφάβητό της, το λεξιλόγιό της, τη γραμματική της και τη σημασιολογία της
5. Δεν υπάρχουν διαφορές μεταξύ φυσικών και τεχνητών γλωσσών
6. Ο τμηματικός προγραμματισμός υλοποιεί την φιλοσοφία της ιεραρχικής σχεδίασης
7. Στο δομημένο προγραμματισμό η επίλυση ενός προβλήματος σχεδιάζεται ιεραρχικά και το πρόγραμμα υλοποιείται τμηματικά.
8. Ο δομημένος προγραμματισμός προϋποθέτει μόνο τη χρήση των δομών ακολουθίας, επιλογής, επανάληψης.
9. Ο διερμηνευτής (interpreter) μετατρέπει το εκτελέσιμο πρόγραμμα (executable) σε πρόγραμμα αντικείμενο (object)
10. Ο μεταγλωττιστής μας επιτρέπει να συντάσσουμε ένα πρόγραμμα
11. Ο μεταγλωττιστής αναλαμβάνει το ρόλο της μετάφρασης από το πηγαίο πρόγραμμα στο εκτελέσιμο πρόγραμμα
12. Ο διερμηνευτής ελέγχει και μετατρέπει μια-μια τις εντολές του πηγαίου προγράμματος σε γλώσσα μηχανής
13. Ο συνδέτης μετατρέπει το πηγαίο πρόγραμμα σε πρόγραμμα βιβλιοθήκης
14. Το αντικείμενο πρόγραμμα είναι γλώσσα μηχανής
15. Ο συνδέτης είναι ένα πρόγραμμα ελέγχου των συντακτικών λαθών του πηγαίου προγράμματος
16. Τα λογικά λάθη εντοπίζονται από έναν μεταγλωττιστή αλλά όχι από έναν διερμηνευτή
17. Τα συντακτικά λάθη επιδιορθώνονται γρήγορα και εύκολα αλλά για τα λογικά λάθη η επιδιόρθωση είναι μια επίπονη διαδικασία
18. Πλεονέκτημα του διερμηνευτή είναι ότι το πρόγραμμα εκτελείται γρηγορότερα

19. Τα σύγχρονα προγραμματιστικά περιβάλλοντα χρησιμοποιούν μικτές υλοποιήσεις διερμηνευτή και μεταγλωττιστή
20. Σ' ένα σύγχρονο προγραμματιστικό περιβάλλον απαιτείται η παρουσία συντάκτη για την συγγραφή του κειμένου των εντολών
21. Για την επιδιόρθωση των λογικών λαθών πολλές φορές ο προγραμματιστής καλείται να εκτελέσει το πρόγραμμά του επανειλημμένα
22. Σ' ένα σύγχρονο προγραμματιστικό περιβάλλον με μεταγλωττιστή η ύπαρξη του συνδέτη είναι προαιρετική
23. Οι βιβλιοθήκες χρησιμεύουν στη διόρθωση των συντακτικών λαθών.
24. Η παράλειψη μιας εντολής Τέλος_Επανάληψης είναι λογικό λάθος
25. Η χρήση της εντολής $MO \leftarrow \alpha + \beta + \gamma / 3$ αντί της $MO \leftarrow (\alpha + \beta + \gamma) / 3$ είναι συντακτικό λάθος
26. Τα λογικά λάθη ενός προγράμματος εμφανίζονται κατά τη μεταγλώττιση
27. Ο εντοπισμός των συντακτικών λαθών σε ένα πρόγραμμα γίνεται από τον μεταγλωττιστή (compiler)
28. 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
29. 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»
30. Χαρακτηριστικό του οπτικού προγραμματισμού είναι:
 - A. Επιτρέπει τη γραφική δημιουργία του περιβάλλοντος προγράμματος σε τμήματα
 - B. Επιτρέπει την ανάπτυξη του προγράμματος σε τμήματα
 - Γ. Είναι ταχύτερος στην εκτέλεση των προγραμμάτων
 - Δ. Επιτρέπει την διαγραμματική παράσταση της σχεδίασης
31. Από τον συντάκτη παράγεται:
 - α) Τα αντικείμενα
 - β) ο εκτελέσιμος κώδικας
 - γ) ο πηγαίος κώδικας
 - δ) τίποτα από τα παραπάνω
32. 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