

Ερωτήσεις πολλαπλής επιλογής - Κεφάλαιο 6

- 1.** Ο προγραμματισμός αφορά την διατύπωση ενός αλγορίθμου σε κατανοητή από τον υπολογιστή μορφή
- 2.** Βασικό στοιχείο του προγράμματος είναι τα δεδομένα οι δομές δεδομένων πέραν της κωδικοποίησης
- 3.** Οι γλώσσες προγραμματισμού αναπτύχθηκαν με σκοπό την επικοινωνία ανθρώπου - μηχανής
- 4.** Ένα πρόγραμμα γραμμένο σε γλώσσα μηχανής είναι εντολές προς τον επεξεργαστή για στοιχειώδεις λειτουργίες
- 5.** Ένα πρόγραμμα σε γλώσσα μηχανής χρειάζεται μετατροπή σε ακολουθία δυαδικών ψηφίων ώστε να εκτελεστεί από τον υπολογιστή
- 6.** Μια ακολουθία εντολών 0, 1 είναι κατανοητή από τον επεξεργαστή και μπορεί να εκτελεστεί χωρίς μετατροπή
- 7.** Ο προγραμματισμός σε γλώσσα μηχανής ήταν μια εξαιρετικά δύσκολη δουλειά που ελάχιστοι μπορούσαν να πραγματοποιήσουν
- 8.** Ένα πρόγραμμα σε συμβολική γλώσσα ή γλώσσα χαμηλού επιπέδου τελικά μετατρέπεται σε γλώσσα μηχανής
- 9.** Η συμβολική γλώσσα είναι μια ακολουθία 0 και 1
- 10.** Μια εντολή ενός προγράμματος γλώσσας χαμηλού επιπέδου μεταφράζεται σε γλώσσα μηχανής
- 11.** Με τις γλώσσες χαμηλού επιπέδου το κέντρο βάρους μετακινήθηκε από τον υπολογιστή προς τον άνθρωπο
- 12.** Οι εντολές σε συμβολική γλώσσα αποτελούνται από συμβολικά ονόματα που αντιστοιχούν σε εντολές της γλώσσας μηχανής
- 13.** Οι συμβολικές γλώσσες έφεραν την ανεξαρτησία από την αρχιτεκτονική κάθε υπολογιστή

- 14.** Τα προγράμματα σε γλώσσες χαμηλού επιπέδου είναι εύκολο να γραφτούν και να συντηρηθούν
- 15.** Τα προγράμματα σε γλώσσες υψηλού επιπέδου είναι ανεξάρτητα του υπολογιστή που αναπτύχθηκαν
- 16.** Οι γλώσσες υψηλού επιπέδου ήρθαν να επιλύσουν τις αδυναμίες των συμβολικών γλωσσών για καλύτερη επικοινωνία ανθρώπου - μηχανής
- 17.** Η Fortran είναι γλώσσα χαμηλού επιπέδου
- 18.** Η γλώσσα προγραμματισμού Fortran είναι κατάλληλη για την επίλυση όλων των ειδικών προβλημάτων
- 19.** Η Fortran είναι κατάλληλη για την επίλυση μαθηματικών και επιστημονικών προβλημάτων
- 20.** Ένα πρόγραμμα γλώσσας υψηλού επιπέδου μπορεί να εκτελεστεί σε οποιονδήποτε υπολογιστή ανεξάρτητα της αρχιτεκτονικής
- 21.** Δεν έχει αναπτυχθεί γλώσσα υψηλού επιπέδου που να επιλύει όλα τα είδη προβλημάτων
- 22.** Η COBOL είναι γλώσσα προσανατολισμένη στην ανάπτυξη εμπορικών εφαρμογών
- 23.** Η COBOL δεν μπορεί να επιλύσει μαθηματικά προβλήματα
- 24.** Η Algol είναι μια γλώσσα γενικού σκοπού αλλά με ελάχιστη πρακτική εφαρμογή
- 25.** Η Algol αναπτύχθηκε για την επίλυση μαθηματικών προβλημάτων
- 26.** Η PL/1 είναι γλώσσα υψηλού επιπέδου γενικής χρήσης με ελάχιστη επιτυχία
- 27.** Η PL/1 προσπάθησε να συνθέσει Fortran και COBOL
- 28.** Η Lisp και η Prolog είναι γλώσσες προγραμματισμού που χρησιμοποιούνται στον τομέα της Τεχνητής Νοημοσύνης
- 29.** Η Basic είναι γλώσσα γενικού σκοπού με έμφαση στην εκπαίδευση αρχαρίων στον προγραμματισμό

- 30.** Η Basic κατόρθωσε με συνεχείς ανανεώσεις να καταστεί από τις πλέον δημοφιλείς γλώσσες προγραμματισμού
- 31.** Η γλώσσα Pascal είναι γλώσσα γενικού σκοπού και κατάλληλη τόσο για την εκπαίδευση όσο και για τη δημιουργία ισχυρών προγραμμάτων
- 32.** Η Pascal αποτέλεσε τη βάση για την παρουσίαση γλωσσών όπως η ADA και η Modula-2
- 33.** Η C χρησιμοποιήθηκε για την ανάπτυξη συστημάτων και έχει πολλές δυνατότητες χαμηλού επιπέδου
- 34.** Η C++ είναι γλώσσα τέταρτης γενιάς
- 35.** Η C++ και οι Java είναι αντικειμενοστραφείς γλώσσες
- 36.** Η Java είναι κατάλληλη για την ανάπτυξη εφαρμογών που εκτελούνται δικτυακά
- 37.** Ο οδηγούμενος από τα γεγονότα προγραμματισμός και οι γλώσσες οπτικού προγραμματισμού αξιοποιούν τα γραφικά χαρακτηριστικά του υπολογιστή (ποντίκι, μενού κ.λ.π.)
- 38.** Στον οπτικό προγραμματισμό ο προγραμματιστής δεν γράφει κώδικα αλλά σχεδιάζει τα οπτικά αντικείμενα της εφαρμογής του
- 39.** Οι πιο διαδεδομένες οπτικές γλώσσες είναι η Visual Basic, η Visual C++ και η Java
- 40.** Ο δομημένος προγραμματισμός επιτρέπει την άμεση μεταφορά των αλγορίθμων σε πρόγραμμα
- 41.** Χάρη στο δομημένο προγραμματισμό δημιουργούνται προγράμματα απλούστερα, κατανοήσιμα και εύκολα στη διόρθωση
- 42.** Στο δομημένο προγραμματισμό ακολουθούνται οι αρχές του ιεραρχικού και του τμηματικού προγραμματισμού
- 43.** Στην ιεραρχική σχεδίαση, η ανάλυση του αλγορίθμου πραγματοποιείται με την τεχνική «από πάνω προς τα κάτω»
- 44.** Ένα πλεονέκτημα των γλωσσών προγραμματισμού χαμηλού επιπέδου είναι η μεταφερσιμότητα των προγραμμάτων

- 45.** Παρά τη μεταφερσιμότητά τους, τα προγράμματα υψηλού επιπέδου είναι δυσκολότερο να διορθωθούν και να συντηρηθούν
- 46.** Στα πλεονεκτήματα των γλωσσών προγραμματισμού υψηλού επιπέδου σε σχέση με τις συμβολικές γλώσσες είναι η ανεξαρτησία από τον τύπο του υπολογιστή
- 47.** Οι γλώσσες υψηλού επιπέδου προτιμούνται για τον κοντύτερα στον άνθρωπο τρόπο έκφρασης
- 48.** Είναι ευκολότερη η εκμάθηση μιας γλώσσας χαμηλού επιπέδου
- 49.** Ο παράλληλος προγραμματισμός προϋποθέτει την ύπαρξη περισσότερων από έναν επεξεργαστών
- 50.** Οι γλώσσες 4ης γενιάς χρησιμοποιούνται σε εφαρμογές που χρησιμοποιούν βάσεις δεδομένων
- 51.** Η καλύτερη γλώσσα προγραμματισμού είναι η Pascal
- 52.** Η επιλογή της καλύτερης γλώσσας προγραμματισμού εξαρτάται από το είδος της εφαρμογής
- 53.** Μια γλώσσα προγραμματισμού προσδιορίζεται από: το αλφάβητό της, το λεξιλόγιό της, τη γραμματική της και τη σημασιολογία της
- 54.** Λεξιλόγιο μιας γλώσσας είναι όλες οι ακολουθίες που δημιουργούνται από τα στοιχεία του αλφαβήτου της γλώσσας, τις λέξεις
- 55.** Η γραμματική είναι το συντακτικό μιας γλώσσας
- 56.** Δεν υπάρχουν διαφορές μεταξύ φυσικών και τεχνητών γλωσσών
- 57.** Η ιεραρχική σχεδίαση διασπά ένα πρόβλημα σε υποπροβλήματα και τα επιλύει, λύνοντας το αρχικό πρόβλημα
- 58.** Ο τμηματικός προγραμματισμός υλοποιεί την φιλοσοφία της ιεραρχικής σχεδίασης
- 59.** Κάθε υποπρόβλημα της ιεραρχικής σχεδίασης αποτελεί μια ανεξάρτητη ενότητα στον τμηματικό προγραμματισμό
- 60.** Ο δομημένος προγραμματισμός περιέχει την ιεραρχική σχεδίαση και τον τμηματικό προγραμματισμό

- 61.** Ο δομημένος προγραμματισμός προϋποθέτει τη χρήση εξειδικευμένων αλγοριθμικών δομών
- 62.** Η αντικειμενοστραφής προσέγγιση εκλαμβάνει ως πρωτεύοντα δομικά στοιχεία ενός προγράμματος τις τρεις αλγοριθμικές δομές: δομή ακολουθίας, δομή επιλογής και δομή επανάληψης
- 63.** Για την ανάπτυξη προγραμμάτων με τη φιλοσοφία του παράλληλου προγραμματισμού απαιτείται η χρήση εξειδικευμένων γλωσσών προγραμματισμού
- 64.** Ο διερμηνευτής (interpreter) μετατρέπει το εκτελέσιμο πρόγραμμα (executable) σε πρόγραμμα αντικείμενο (object)
- 65.** Ο μεταγλωττιστής μας επιτρέπει να συντάσσουμε ένα πρόγραμμα
- 66.** Ο μεταγλωττιστής αναλαμβάνει το ρόλο της μετάφρασης από το πηγαίο πρόγραμμα στο εκτελέσιμο πρόγραμμα
- 67.** Ο διερμηνευτής ελέγχει και μετατρέπει μια-μια τις εντολές του πηγαίου προγράμματος σε γλώσσα μηχανής
- 68.** Ο συνδότης μετατρέπει το πηγαίο πρόγραμμα σε βιβλιοθήκη
- 69.** Το αντικείμενο πρόγραμμα είναι ουσιαστικά γλώσσα μηχανής
- 70.** Ο συνδότης είναι ένα πρόγραμμα ελέγχου των συντακτικών λαθών του πηγαίου προγράμματος
- 71.** Τα λογικά λάθη εντοπίζονται από έναν μεταγλωττιστή αλλά όχι από έναν διερμηνευτή
- 72.** Τα συντακτικά λάθη επιδιορθώνονται γρήγορα και εύκολα αλλά για τα λογικά λάθη η επιδιόρθωση είναι μια επίπονη διαδικασία
- 73.** Ο μεταγλωττιστής έχει το μειονέκτημα ότι ελέγχει όλο το πρόγραμμα και πραγματοποιεί και την διαδικασία της σύνδεσης πολλές φορές μέχρι να επιδιορθωθούν όλα τα λάθη
- 74.** Ο διερμηνευτής έχει το πλεονέκτημα ότι το πρόγραμμα εκτελείται γρηγορότερα

- 75.** Τα σύγχρονα προγραμματιστικά περιβάλλοντα χρησιμοποιούν μικτές υλοποιήσεις διερμηνευτή και μεταγλωττιστή
- 76.** Σ' ένα σύγχρονο προγραμματιστικό περιβάλλον απαιτείται η παρουσία συντάκτη για την συγγραφή του κειμένου των εντολών
- 77.** Για την επιδιόρθωση των λογικών λαθών πολλές φορές ο προγραμματιστής καλείται να εκτελέσει το πρόγραμμά του επανειλημμένα
- 78.** Ο μεταγλωττιστής σ' ένα σύγχρονο προγραμματιστικό περιβάλλον καθιστά την ύπαρξη του συνδέτη προαιρετική
- 79.** Ο μεταγλωττιστής διορθώνει όλα τα συντακτικά λάθη με τη χρήση βιβλιοθηκών
- 80.** Η παράλειψη μιας εντολής Τέλος_αν είναι λογικό λάθος
- 81.** Η χρήση της εντολής $MO \leftarrow a + \beta + \gamma / 3$ αντί της $MO \leftarrow (a + \beta + \gamma) / 3$ είναι λογικό λάθος
- 82.** Τα λογικά λάθη ενός προγράμματος εμφανίζονται κατά τη μεταγλώττιση
- 83.** Ο εντοπισμός των συντακτικών λαθών σε ένα πρόγραμμα γίνεται από τον μεταγλωττιστή (compiler)
- 84.** Το έργο της μετάφρασης εντολών γλώσσας χαμηλού επιπέδου σε ακολουθία 0, 1 αναλαμβάνει ο _____
- 85.** Η δυσκολία προγραμματισμού σε γλώσσα μηχανής οδήγησε στην δημιουργία _____ γλωσσών ή γλωσσών _____ επιπέδου
- 86.** Βασική τεχνική σχεδίασης προγραμμάτων είναι η τεχνική του _____ προγραμματισμού
- 87.** Κάθε πρόγραμμα γλώσσας υψηλού επιπέδου μεταφράζεται σε γλώσσα μηχανής από ένα ειδικό πρόγραμμα που ονομάζεται _____
- 88.** Η ανεξαρτησία των γλωσσών προγραμματισμού από την αρχιτεκτονική των υπολογιστών ονομάζεται _____
- 89.** Ο δομημένος προγραμματισμός στηρίζεται στη χρήση τριών στοιχειωδών δομών: της _____, της _____ και της _____

- 90.** _____ είναι το σύνολο των κανόνων που ορίζει τις μορφές που μια λέξη είναι αποδεκτή
- 91.** _____ είναι το σύνολο των κανόνων που καθορίζει τη διάταξη και τη σύνδεση των λέξεων
- 92.** Η λανθασμένη γραφή των δεσμευμένων λέξεων της γλώσσας προγραμματισμού είναι _____ λάθος
- 93.** _____ είναι το σύνολο των κανόνων που καθορίζει το νόημα των λέξεων και κατ' επέκταση των εκφράσεων και προτάσεων που χρησιμοποιούνται σε μια γλώσσα
- 94.** Κάθε προγραμματιστικό περιβάλλον διαθέτει τον μεταγλωττιστή του
- 95.** Ο μεταγλωττιστής δέχεται στην είσοδό του έναν πρόγραμμα σε γλώσσα _____ δημιουργώντας ισοδύναμο σε γλώσσα _____
- 96.** Η μεταγλώττιση ενός προγράμματος γίνεται από τους _____ ή τους _____
- 97.** Οι εντολές ενός προγράμματος μετατρέπονται σε ακολουθίες που αποτελούνται από 0 και 1, δηλαδή σε _____
- 98.** Τα _____ λάθη δεν είναι δυνατόν να εντοπίζονται από το προγραμματιστικό περιβάλλον
- 99.** 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
- 100.** 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»

101. Χαρακτηριστικό του οπτικού προγραμματισμού είναι:

- A. Επιτρέπει τη γραφική δημιουργία του περιβάλλοντος
- B. Επιτρέπει την ανάπτυξη του προγράμματος σε τμήματα
- Γ. Είναι ταχύτερος στην εκτέλεση των προγραμμάτων
- Δ. Επιτρέπει την διαγραμματική παράσταση της σχεδίασης

102. Η Basic είναι

- A. Κατάλληλη για εφαρμογές τεχνητής νοημοσύνης
- B. Υποστηρίζει την ανάπτυξη παράλληλου προγραμματισμού
- Γ. Μία γλώσσα γενικής χρήσης
- Δ. Κατάλληλη μόνο για εκπαίδευση

103. Από τον συντάκτη παράγεται:

- α) Τα αντικείμενα
- β) ο εκτελέσιμος κώδικας
- γ) ο πηγαίος κώδικας
- δ) τίποτα από τα παραπάνω

104. 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

105. Να αντιστοιχίσετε τα στοιχεία της στήλης Α με τα στοιχεία της στήλης Β:

Στήλη Α	Στήλη Β
Σκοπιές που μελετά η πληροφορική 1. τους αλγορίθμους 2. τα δεδομένα	I. Υλικού II. Γλωσσών προγραμματισμού III. Δομών δεδομένων IV. Θεωρητική V. Ανάλυσης δεδομένων VI. Αναλυτική

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

Στήλη Α	Στήλη Β
1. Δομημένος προγραμματισμός 2. Τμηματικός προγραμματισμός 3. Αντικειμενοστραφής προγραμματισμός 4 . Παράλληλος προγραμματισμός	A. Διάρθρωση του προβλήματος σε τμήματα που εκτελούνται παράλληλα B. Ένα πρόβλημα περιγράφει "ενέργειες" που εφαρμόζονται στα δεδομένα Γ. Κάθε υποπρόβλημα αποτελεί ξεχωριστή ενότητα Δ. Χρήση βασικών αλγοριθμικών δομών

107. Να αντιστοιχίσετε τα στοιχεία της στήλης Α και της στήλης Β

A	B
1. Δομή επιλογής	A) Επαναληπτική εκτέλεση εντολών

2. Δομή επανάληψης	Β) Χρήση εντολή GoTo
3. Μη δομημένος προγραμματισμός	Γ) Σειριακή εκτέλεση εντολών
4. Δομή ακολουθίας	Δ) Επιλεκτική εκτέλεση εντολών
5. Δομημένος προγραμματισμός	Ε) Αποφυγή χρήσης Εντολής GoTo

108. 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

109. Το εκτελέσιμο είναι το τελικό πρόγραμμα που εκτελείται από τον υπολογιστή

110. Στις γλώσσες υψηλού επιπέδου δεν υπάρχει η ικανότητα της μεταφερσιμότητας

111. Υπάρχουν δυο μεγάλες κατηγορίες μεταφραστικών προγραμμάτων: οι μεταφραστές και οι μεταγλωττιστές

112. Ο μεταγλωττιστής εξάγει το πηγαίο πρόγραμμα

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

Στήλη Α	Στήλη Β

<p>A. Διερμηνευτής</p> <p>B. Μεταγλωττιστής</p> <p>Γ. Δομημένος προγραμματισμός</p> <p>Δ. Συντάκτης</p>	<p>1. Πρόγραμμα που δέχεται ένα πρόγραμμα γραμμένο σε γλώσσα υψηλού επιπέδου και παράγει ένα ισοδύναμο πρόγραμμα σε γλώσσα μηχανής</p> <p>2. Μεθοδολογία σύνταξης προγραμμάτων που βοηθάει στην ανάπτυξη σύνθετων προγραμμάτων και στη διαχείρισή τους</p> <p>3. Πρόγραμμα που χρησιμοποιείται για τη σύνταξη και τη διόρθωση προγραμμάτων</p> <p>4. Πρόγραμμα που διαβάζει μια προς μια τις εντολές αρχικού προγράμματος και για κάθε μια εκτελεί αμέσως μια ισοδύναμη ακολουθία εντολών μηχανής</p>
---	---

114. Τα βήματα για τη δημιουργία ενός προγράμματος είναι:

- α) συντάκτης
- β) συνδέτης
- γ) πηγαίο πρόγραμμα
- δ) μεταγλωττιστής
- ε) εκτελέσιμο πρόγραμμα
- στ) αντικείμενο πρόγραμμα

Λύσεις

1: Σωστό	2: Σωστό	3: Σωστό	4: Σωστό	5: Λάθος
6: Σωστό	7: Σωστό	8: Σωστό	9: Λάθος	10: Σωστό

11: Σωστό	12: Σωστό	13: Λάθος	14: Λάθος	15: Σωστό
16: Σωστό	17: Λάθος	18: Λάθος	19: Σωστό	20: Σωστό
21: Σωστό	22: Σωστό	23: Λάθος	24: Σωστό	25: Λάθος
26: Σωστό	27: Σωστό	28: Σωστό	29: Σωστό	30: Σωστό
31: Σωστό	32: Σωστό	33: Σωστό	34: Λάθος	35: Σωστό
36: Σωστό	37: Σωστό	38: Σωστό	39: Σωστό	40: Σωστό
41: Σωστό	42: Σωστό	43: Σωστό	44: Λάθος	45: Λάθος
46: Σωστό	47: Σωστό	48: Σωστό	49: Σωστό	50: Σωστό
51: Λάθος	52: Σωστό	53: Σωστό	54: Λάθος	55: Λάθος
56: Λάθος	57: Σωστό	58: Σωστό	59: Σωστό	60: Σωστό
61: Λάθος	62: Λάθος	63: Σωστό	64: Λάθος	65: Λάθος
66: Λάθος	67: Σωστό	68: Λάθος	69: Σωστό	70: Λάθος
71: Λάθος	72: Σωστό	73: Λάθος	74: Λάθος	75: Σωστό
76: Σωστό	77: Σωστό	78: Λάθος	79: Λάθος	80: Λάθος

81: Σωστό	82: Λάθος	83: Σωστό	84: συμβολομεταφραστής	85: συμβολικών, χαμηλού
86: δομημένου	87: μεταγλωττιστής	88: μεταφερσιμότητα	89: ακολουθίας, επιλογής και επανάληψης	90: τυπικό
91: συντακτικό	92: συντακτικό	93: σημασιολογία	94: Λάθος	95: υψηλού επιπέδου, μηχανής
96: μεταγλωττιστές, διερμηνευτές	97: γλώσσα μηχανής	98: λογικά	99: β	100: β
101: α	102: γ	103: γ	104: 1.Ι.Δ 2.ΙΙ.Δ 3.VI.Δ 4.VII.Γ 5.VII.A 6.VI.Δ 7.VI.Δ 8.IV.Δ 9.IV.B 10.V.B 11.III.Δ	105: 1. I, II, IV, VI 2. I, II, III, V
106: 1Δ, 2Γ, 3Β, 4Α	107: 1Δ, 2Α, 3Β, 4Γ, 5Ε	108: Α5, Β3, Γ2, Δ4, Ε1	109: Σωστό	110: Λάθος
111: Λάθος	112: Λάθος	113: Α4, Β1, Γ2, Δ3	114: α, γ, δ, στ, β, ε	