



Αρχιτεκτονική Υπολογιστών Ι
Θέματα Ιουνίου 2026
ΟΜΑΔΑ Β



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ
Πανεπιστήμιο Πελοποννήσου

Τμήμα	Πληροφορικής και Τηλεπικοινωνιών, Πανεπιστήμιο Πελοποννήσου
Εξεταστική	Ιούνιος 2026
Διδάσκων	Γρηγόρης Δημητρουλάκος
Μάθημα	Αρχιτεκτονική Υπολογιστών Ι
Ονοματεπώνυμο	
ΑΜ	

Κυκλώστε τη σωστή απάντηση. Για κάθε λανθασμένη απάντηση υπάρχει αρνητική βαθμολόγηση 25%. Για κάθε ερώτημα που δεν απαντάται, η βαθμολογία είναι 0. Τα ερωτήματα 1-24 και 30-63 βαθμολογούνται με 2 μονάδες το καθένα. Για κάθε λανθασμένη απάντηση στα ερωτήματα αυτά αφαιρούνται 0,5 μονάδες. Τα ερωτήματα 25-29 βαθμολογούνται με 4 μονάδες το καθένα. Για κάθε λανθασμένη απάντηση στα ερωτήματα αυτά αφαιρείται 1 μονάδα. Η συνολική βαθμολογία του διαγωνίσματος είναι 136 μονάδες. Καλή επιτυχία!!!

ΣΚΕΛΟΣ ΑΡΧΙΤΕΚΤΟΝΙΚΗΣ: 1) Σε δύο μοντέλα της ίδιας οικογένειας υπολογιστών εκτελείται το ίδιο δυαδικό πρόγραμμα. Το νεότερο μοντέλο αλλάζει την οργάνωση της L1 κρυφής μνήμης από άμεση σε 4-way set associative, χωρίς να αλλάζει το σύνολο εντολών. Πού ταξινομείται κυρίως η αλλαγή;

- A) Στην Αρχιτεκτονική Υπολογιστή, αν θα απαιτούσε διαφορετικές εντολές φόρτωσης/αποθήκευσης.
- B) Στην Αρχιτεκτονική Υπολογιστή, επειδή τότε θα έπρεπε να αλλάξει η ορατή μορφή των εντολών.
- C) Και στα δύο εξίσου, μόνο αν ο μεταγλωττιστής αλλάζει το πρόγραμμα για να αξιοποιήσει την cache.
- D) Στην Οργάνωση μόνο αν αλλάξει και η πολιτική αντικατάστασης μαζί με την αντιστοίχιση.
- E) Στην Οργάνωση Υπολογιστή, αφού αλλάζει εσωτερική υλοποίηση μνήμης χωρίς νέα ορατή εντολή.

2) Ποια από τις παρακάτω αλλαγές ανήκει περισσότερο στην Οργάνωση Υπολογιστή και όχι στην ορατή Αρχιτεκτονική Συνόλου Εντολών;

- A) Προσθήκη νέου ορατού καταχωρητή στο σύνολο καταχωρητών που μπορεί να χρησιμοποιήσει ο assembler.
- B) Προσθήκη νέας εντολής διακλάδωσης με νέο opcode και ορατή αλλαγή στη ροή εκτέλεσης.
- C) Προσθήκη νέου τρόπου διευθυνσιοδότησης σε υπάρχουσες εντολές που γράφεται στον κώδικα μηχανής.
- D) Αλλαγή της μονάδας ελέγχου σε μικροπρογραμματισμένη, με ίδια ορατή συμπεριφορά εντολών.
- E) Αλλαγή της δυαδικής μορφής εντολής που πρέπει να παραχθεί από assembler ή μεταγλωττιστή.

3) Ποια συνέπεια συνδέεται πιο άμεσα με τη αύξηση πυκνότητας τρανζίστορ που περιγράφει ο νόμος του Moore;

- A) Περισσότερες λειτουργίες στο ίδιο chip και χαμηλότερο κόστος ανά λειτουργία.
- B) Η συχνότητα ρολογιού αυξάνεται χωρίς ουσιαστικό όριο, ανεξάρτητα από θερμότητα και διασυνδέσεις.
- C) Το πλήθος εντολών κάθε προγράμματος παραμένει αμετάβλητο σε κάθε νέα γενιά επεξεργαστή.
- D) Η κύρια μνήμη ακολουθεί πάντα την ταχύτητα του επεξεργαστή, οπότε δεν δημιουργείται χάσμα.
- E) Η ιεραρχία μνήμης παύει να χρειάζεται, επειδή η μεγαλύτερη πυκνότητα μηδενίζει τις καθυστερήσεις.

4) Ποια περιγραφή ταιριάζει καλύτερα σε ενσωματωμένο σύστημα που ελέγχει ένα φυσικό περιβάλλον, όπως αισθητήρες και ενεργοποιητές;

- A) Ελεύθερη εγκατάσταση εφαρμογών γενικού σκοπού, όπως σε προσωπικό υπολογιστή.
- B) Προβλέψιμη απόκριση σε γεγονότα και λειτουργία με περιορισμένους πόρους.
- C) Αποφυγή ειδικού λογισμικού, επειδή η εξειδίκευση θα αναιρούσε τον υπολογιστικό χαρακτήρα.
- D) Κύρια μνήμη μεγαλύτερη από προσωπικό υπολογιστή, ώστε να αποθηκεύονται όλα τα σήματα αισθητήρων.
- E) Χρήση αποκλειστικά εξωτερικής μνήμης, επειδή οι αισθητήρες δεν χρειάζονται τοπική επεξεργασία.

5) Τι συμβαίνει συνήθως όταν μια πρόβλεψη διακλάδωσης αποδειχθεί λανθασμένη σε διασωληνωμένο επεξεργαστή;

- A) Οι εντολές του λάθους μονοπατιού ολοκληρώνονται κανονικά και τα αποτελέσματα δεσμεύονται στο πρόγραμμα.
- B) Ο επεξεργαστής κρατά και τα δύο μονοπάτια ως οριστικά, ώστε να αποφύγει επανεκκίνηση της ανάκλησης.
- C) Η πρόβλεψη διορθώνεται μόνο στον επόμενο βρόχο, χωρίς ακύρωση εντολών που έχουν ήδη εισαχθεί.
- D) Το forwarding καλύπτει την ποινή ελέγχου, άρα η λάθος πρόβλεψη δεν προκαλεί flush.
- E) Ακυρώνονται οι εντολές λάθους μονοπατιού και γίνεται ανάκληση από τη σωστή διεύθυνση.

6) Ποια φυσική αιτία εξηγεί γιατί οι διασυνδέσεις μπορούν να περιορίσουν τη συχνότητα σε πυκνά ολοκληρωμένα κυκλώματα;

- A) Μικρότερη διατομή μειώνει ταυτόχρονα R και C, άρα το όριο μεταφέρεται μόνο στη μνήμη.
- B) Μικρότερη διατομή αυξάνει R και κοντινές γραμμές μπορούν να αυξήσουν C.
- C) Κοντινότερες γραμμές καταργούν τη χωρητική σύζευξη, άρα η καθυστέρηση είναι μόνο λογική.
- D) Η καθυστέρηση διασύνδεσης καθορίζεται κυρίως από τον αριθμό εντολών της αρχιτεκτονικής.
- E) Η υψηλότερη πυκνότητα κάνει όλα τα σήματα να φτάνουν ταυτόχρονα χωρίς χρονισμό.

7) Δύο επεξεργαστές εκτελούν το ίδιο πρόγραμμα με ίδιο πλήθος εκτελούμενων εντολών. Ο Α έχει CPI=1.0 στα 2.4 GHz και ο Β έχει CPI=1.3 στα 3.6 GHz. Ποια σύγκριση είναι σωστή; Διακαιολογίστε τις απαντήσεις σας

- A) Ο Α είναι περίπου 1.15 φορές ταχύτερος, επειδή έχει μικρότερο CPI.
- B) Οι δύο έχουν ίδια επίδοση, επειδή η καλύτερη συχνότητα του Β αντισταθμίζεται πάντα από το μεγαλύτερο CPI.
- C) Ο Β είναι 1.50 φορές ταχύτερος, επειδή αρκεί ο λόγος συχνοτήτων 3.6/2.4.
- D) Δεν μπορεί να γίνει σύγκριση όταν το πλήθος εντολών είναι ίδιο, επειδή λείπει η γλώσσα προγραμματισμού.
- E) Ο Β είναι περίπου 1.15 φορές ταχύτερος, επειδή συγκρίνονται οι λόγοι CPI/συχνότητα.

8) Ένα πρόγραμμα έχει αρχικά $p=1.2$ κύκλους ανά εντολή για αποκωδικοποίηση/εκτέλεση, $m=0.08$ αναφορές μνήμης ανά εντολή και $k=12$ κύκλους αναμονής ανά αναφορά. Μετά από αλλαγή κώδικα χωρίς να αλλάζει ο αριθμός εντολών προγράμματος, $p=1.3$ και $m=0.05$, ενώ k ίδιο. Ποιο συμπέρασμα είναι σωστό; Διακαιολογίστε τις απαντήσεις σας

- A) Υπάρχει επιβράδυνση, επειδή το p αυξάνεται και το m δεν επηρεάζει τον χρόνο εκτέλεσης.
- B) Η επίδοση μένει ίδια, επειδή η αύξηση του p και η μείωση του m αλληλοαναιρούνται ακριβώς.
- C) Δεν μπορεί να υπολογιστεί CPI, επειδή ο τύπος $p+k \cdot m$ ισχύει μόνο για εντολές διακλάδωσης.
- D) Υπάρχει επιτάχυνση 60%, επειδή το m μειώνεται από 0.08 σε 0.05.
- E) Υπάρχει επιτάχυνση περίπου 1.14, επειδή το CPI πέφτει από 2.16 σε 1.90.

9) Σε σύγκριση συστημάτων με κανονικοποιημένους λόγους επίδοσης από πολλά benchmarks, ποια χρήση μέσου είναι πιο κατάλληλη;

- A) Ο γεωμετρικός μέσος, επειδή χειρίζεται κανονικοποιημένους λόγους επίδοσης.
- B) Ο αριθμητικός μέσος των απόλυτων χρόνων, επειδή δεν επηρεάζεται από την κλίμακα κάθε benchmark.
- C) Ο μέγιστος χρόνος εκτέλεσης, επειδή αντιπροσωπεύει πλήρως όλα τα προγράμματα του συνόλου.
- D) Ο ελάχιστος λόγος επίδοσης, επειδή αφαιρεί benchmarks που δεν ευνοούν την αρχιτεκτονική.
- E) Ο μέσος των συχνοτήτων ρολογιού, επειδή αρκεί για σύγκριση διαφορετικών συστημάτων.

10) Σε διανυσματική διακοπή, ποια πληροφορία βοηθά τον επεξεργαστή να μεταβεί στη σωστή ρουτίνα εξυπηρέτησης;

- A) Η αποθηκευμένη τιμή του μετρητή προγράμματος, επειδή αυτή δείχνει πάντα τη ρουτίνα κάθε συσκευής.
- B) Η μάσκα διακοπών, επειδή περιέχει απευθείας τις διευθύνσεις όλων των ρουτινών.
- C) Ένας κώδικας/διάνυσμα της πηγής διακοπής που οδηγεί στη διεύθυνση της ρουτίνας.
- D) Το σήμα αναγνώρισης διακοπής, επειδή από μόνο του κωδικοποιεί τη διεύθυνση εξυπηρέτησης.
- E) Η γραμμή προτεραιότητας, επειδή η προτεραιότητα είναι ισοδύναμη με τη διεύθυνση ρουτίνας.

11) Ποια πρόταση για τις διακοπές είναι ΨΕΥΔΗΣ;

- A) Μια διακοπή μπορεί να επιτρέπει στον επεξεργαστή να συνεχίζει άλλη εργασία ώσπου να είναι έτοιμη η συσκευή.
- B) Πριν από την επιστροφή στο πρόγραμμα πρέπει να έχει ανακτηθεί η απαραίτητη κατάσταση εκτέλεσης.
- C) Σε σύστημα προτεραιοτήτων, διακοπή υψηλότερου επιπέδου μπορεί να διακόψει ρουτίνα χαμηλότερου επιπέδου.
- D) Η ρουτίνα εξυπηρέτησης εκτελείται από τον ελεγκτή της συσκευής, όχι από τον επεξεργαστή.
- E) Αν οι διακοπές έχουν προσωρινά απενεργοποιηθεί, νέο αίτημα μπορεί να μείνει σε αναμονή.

12) Ένας μικρός βρόχος εκτελείται 10.000 φορές και οι ίδιες εντολές του βρόχου ανακαλούνται ξανά και ξανά. Ποια τοπικότητα περιγράφεται καλύτερα;

- A) Χρονική τοπικότητα εντολών, επειδή ίδιες διευθύνσεις εντολών ανακαλούνται ξανά σύντομα.
- B) Χωρική τοπικότητα εντολών, επειδή κάθε επανάληψη χρησιμοποιεί διαφορετικές γειτονικές ρουτίνες.
- C) Καμία τοπικότητα, επειδή οι επαναλήψεις του ίδιου βρόχου θεωρούνται ανεξάρτητα προγράμματα.
- D) Χρονική τοπικότητα δεδομένων, επειδή η ανάκληση εντολών ισοδυναμεί πάντα με επανάληψη δεδομένων.
- E) Χωρική τοπικότητα δεδομένων, μόνο αν οι εντολές του βρόχου γράφουν σε συνεχόμενες θέσεις.

13) Ποια πρόταση για τη χωρική τοπικότητα είναι ΨΕΥΔΗΣ;

- A) Εμφανίζεται όταν διαδοχικές προσπελάσεις βρίσκονται σε κοντινές διευθύνσεις μνήμης.
- B) Αξιοποιείται όταν ένα block φέρνει περισσότερες λέξεις από τη ζητούμενη στην κρυφή μνήμη.
- C) Απαιτεί επαναπροσπέλαση της ίδιας διεύθυνσης, όχι γειτονικών.
- D) Συχνά είναι ισχυρή στις σειριακές προσπελάσεις στοιχείων πίνακα σε row-major διάταξη.
- E) Μπορεί να εξαρτάται από το ποιος δείκτης μεταβάλλεται στον εσωτερικό βρόχο.

14) Ποια πρόταση για τον ρυθμό εκατομμυρίων εντολών ανά δευτερόλεπτο είναι ΨΕΥΔΗΣ;

- A) Είναι πάντα ασφαλής για σύγκριση διαφορετικών συνόλων εντολών και προγραμμάτων.
- B) Εξαρτάται από τη συχνότητα ρολογιού και το μέσο CPI για συγκεκριμένο πρόγραμμα.
- C) Μπορεί να αυξηθεί αν μειωθεί το CPI ενώ η συχνότητα παραμένει σταθερή.
- D) Μπορεί να παραπλανήσει όταν διαφορετικά σύνολα εντολών απαιτούν άλλο πλήθος εντολών για το ίδιο έργο.
- E) Δεν αρκεί μόνος του για τον συνολικό χρόνο χωρίς να ληφθεί υπόψη το πρόγραμμα.

15) Ποιος είναι ο ρόλος της ανάλυσης ροής δεδομένων σε σύγχρονο επεξεργαστή;

- A) Να μετατρέπει κώδικα μηχανής σε πηγαίο κώδικα ώστε να μειωθεί το πλήθος εντολών του προγράμματος.
- B) Να εντοπίζει εξαρτήσεις εντολών ώστε να μειώνεται η αναμονή όπου επιτρέπεται.
- C) Να θεωρεί ανεξάρτητες όλες τις εντολές που χρησιμοποιούν διαφορετικούς καταχωρητές.
- D) Να επιτρέπει αναδιάταξη εντολών ακόμη και όταν υπάρχει πραγματική εξάρτηση δεδομένων.
- E) Να χρονοπρογραμματίζει εντολές χωρίς έλεγχο διαθεσιμότητας των τελεστών.

16) Ποια διατύπωση για Double Data Rate Synchronous Dynamic Random Access Memory είναι σωστή;

- A) Μεταφέρει δεδομένα μόνο στην ανερχόμενη ακμή και κερδίζει χρόνο καταργώντας το refresh.
- B) Χρησιμοποιεί στατικά κελιά όπως η SRAM, άρα δεν απαιτεί δυναμική ανανέωση.
- C) Μεταφέρει δεδομένα και στις δύο ακμές του ρολογιού.
- D) Διπλασιάζει τον ρυθμό μεταφοράς επειδή διπλασιάζει το πλάτος του διαύλου δεδομένων.
- E) Αυξάνει τον ρυθμό επειδή κάθε προσπέλαση ολοκληρώνεται χωρίς ανανέωση δυναμικών κελιών.

Κοινά δεδομένα για τα ερωτήματα 17-20: τοπικότητα σε πίνακες αποθηκευμένους κατά γραμμές.

17) Δίνεται πίνακας $\text{int } A[N][N]$ σε row-major διάταξη και ο κώδικας:

```
1. for (int j=0; j<N; j++)
2.     for (int i=0; i<N; i++)
3.         sum += A[i][j];
4.
```

Τι συμβαίνει στον εσωτερικό βρόχο i;

- A) Ισχυρή χωρική τοπικότητα, επειδή κάθε αύξηση του i οδηγεί σε γειτονικό στοιχείο της ίδιας γραμμής.
- B) Ασθενής χωρική τοπικότητα, επειδή οι προσπελάσεις απέχουν κατά γραμμή πίνακα.
- C) Ισχυρή χρονική τοπικότητα, επειδή κάθε $A[i][j]$ επαναχρησιμοποιείται άμεσα στην επόμενη επανάληψη.
- D) Η χωρική τοπικότητα είναι ισχυρή, επειδή οι στήλες αποθηκεύονται διαδοχικά στη row-major διάταξη.
- E) Ο εσωτερικός βρόχος προσπελαίνει την ίδια ακριβώς διεύθυνση σε όλες τις επαναλήψεις.

18) Αν ο προηγούμενος κώδικας αλλάξει σε

```
1. for (int i=0; i<N; i++)
2.     for (int j=0; j<N; j++)
3.         sum += A[i][j];
4.
```

τι βελτιώνεται κυρίως;

- A) Η χρονική τοπικότητα, επειδή κάθε στοιχείο $A[i][j]$ χρησιμοποιείται περισσότερες φορές από πριν.
- B) Το πλήθος αριθμητικών πράξεων, επειδή οι δύο βρόχοι εκτελούν λιγότερες επαναλήψεις.
- C) Η χωρική τοπικότητα, επειδή ο j κινείται σε γειτονικά στοιχεία της ίδιας γραμμής.
- D) Η χωρική τοπικότητα μειώνεται, επειδή η αύξηση του j αλλάζει το offset μέσα στη γραμμή.
- E) Δεν αλλάζει η χωρική τοπικότητα, επειδή ο πίνακας παραμένει αποθηκευμένος κατά γραμμές.

19) Γιατί η προσπέλαση $A[i][j]$, $A[i][j+1]$, $A[i][j+2]$ είναι φιλικότερη προς την κρυφή μνήμη σε row-major διάταξη από την $A[i][j]$, $A[i+1][j]$, $A[i+2][j]$;

- A) Επειδή η πρώτη ακολουθία αλλάζει φυσική διεύθυνση κατά ολόκληρη γραμμή κρυφής μνήμης σε κάθε βήμα.
- B) Επειδή η δεύτερη ακολουθία επαναχρησιμοποιεί πάντα την ίδια διεύθυνση, άρα προκαλεί χρονική τοπικότητα.
- C) Επειδή η row-major διάταξη αποθηκεύει πρώτα τις στήλες και κατόπιν τις γραμμές κάθε πίνακα.
- D) Επειδή το πλήθος των εντολών μηχανής μηδενίζεται όταν αυξάνεται ο δείκτης j στον εσωτερικό βρόχο.
- E) Η πρώτη ακολουθία κινείται σε συνεχόμενες θέσεις της ίδιας γραμμής και αξιοποιεί το block.

20) Ποιο μοτίβο προσπέλασης αξιοποιεί καλύτερα τη χωρική τοπικότητα σε πίνακα row-major;

- A) $A[0][j]$, $A[1][j]$, $A[2][j]$, $A[3][j]$
- B) $A[i][0]$, $A[i+2][0]$, $A[i+4][0]$, $A[i+6][0]$
- C) $A[0][0]$, $A[N-1][N-1]$, $A[0][N-1]$, $A[N-1][0]$
- D) $A[i][0]$, $A[i][1]$, $A[i][2]$, $A[i][3]$
- E) Μόνο προσπελάσεις σε καταχωρητές, όχι σε μνήμη

21) Ένα πρόγραμμα έχει παραλληλοποιήσιμο τμήμα 60% και εκτελείται σε 4 ισοδύναμους επεξεργαστές. Ποια είναι η επιτάχυνση κατά Amdahl; Διακαιολογίστε τις απαντήσεις σας

- A) Περίπου 1.82
- B) Περίπου 2.50
- C) Περίπου 4.00
- D) Περίπου 1.43
- E) Περίπου 0.55

Κοινά δεδομένα για τα ερωτήματα 22-24: διάσπαση διεύθυνσης κρυφής μνήμης Διακαιολογίστε τις απαντήσεις σας

22) Ένα σύστημα έχει κύρια μνήμη 128 MB με διευθυνσιοδότηση ανά byte και κρυφή μνήμη 256 KB, 8-way set associative, με block 64 bytes. Πόσα bits έχει το offset; Διακαιολογίστε τις απαντήσεις σας

- A) 5 bits
- B) 7 bits
- C) 8 bits
- D) 6 bits
- E) 9 bits

23) Στο ίδιο σύστημα, πόσα bits έχει το πεδίο set; Διακαιολογίστε τις απαντήσεις σας

- A) 8 bits
- B) 10 bits
- C) 11 bits
- D) 12 bits
- E) 9 bits

24) Στο ίδιο σύστημα, πόσα bits έχει το πεδίο tag; Διακαιολογίστε τις απαντήσεις σας

- A) 10 bits
- B) 11 bits
- C) 12 bits
- D) 13 bits
- E) 14 bits

Κοινά δεδομένα για τα ερωτήματα 25-29: AMAT, CPI, πραγματικό CPI και MIPS

Ένας επεξεργαστής με διασωλήνωση 5 σταδίων λειτουργεί σε συχνότητα 3.0 GHz. Ένα πρόγραμμα αναφοράς που εκτελείται σε αυτόν αποτελείται από 1.4×10^9 εντολές.

Το μίγμα εντολών του προγράμματος είναι:

- Αριθμητικές / Λογικές / Μεταφοράς: 50% του συνόλου, με ιδανικό CPI = 1.
- Προσπέλασης μνήμης: 30% του συνόλου, με ιδανικό CPI = 3.0, όταν όλες οι προσπελάσεις είναι hit στο L1.
- Διακλάδωσης: 20% του συνόλου, με ιδανικό CPI = 2.

Η κρυφή μνήμη L1 έχει χρόνο ελέγχου/επιτυχίας 1.0 ns, ποσοστό αστοχίας 5% και ποινή αστοχίας 30 κύκλους μετά τον έλεγχο στο L1. Σε περίπτωση αστοχίας θεωρείται ότι πρώτα έχει γίνει προσπέλαση στο L1 και στη συνέχεια προστίθεται η ποινή αστοχίας.

Επιπλέον:

- Το 15% του συνόλου των εντολών εξαρτάται από το αποτέλεσμα της αμέσως προηγούμενης εντολής. Ο μηχανισμός προώθησης επιλύει το 60% αυτών των περιπτώσεων. Οι υπόλοιπες περιπτώσεις απαιτούν 2 κύκλους αναμονής.
- Η πρόβλεψη διακλαδώσεων έχει ποσοστό επιτυχίας 90%. Κάθε λανθασμένη πρόβλεψη διακλάδωσης προκαλεί ποινή 3 κύκλων.

25) Ποιος είναι ο μέσος αριθμός κύκλων ανά προσπέλαση μνήμης;

- A) 3.5 κύκλοι
- B) 4.5 κύκλοι
- C) 5.0 κύκλοι
- D) 4.0 κύκλοι
- E) 6.0 κύκλοι

26) Ποιο είναι το ιδανικό CPI του προγράμματος με 100% hit στο L1 και χωρίς πρόσθετες καθυστερήσεις από hazards;

- A) 1.50
- B) 1.80
- C) 1.65
- D) 2.10
- E) 1.95

27) Αν ληφθούν υπόψη μόνο οι αστοχίες μνήμης, ποιο είναι το νέο CPI του προγράμματος;

Στο ερώτημα αυτό οι καθυστερήσεις από κινδύνους δεδομένων και κινδύνους ελέγχου δεν λαμβάνονται υπόψη.

- A) 1.95
- B) 2.00
- C) 2.25
- D) 2.10
- E) 2.40

28) Αν ληφθούν υπόψη οι αστοχίες μνήμης, οι κίνδυνοι δεδομένων και οι κίνδυνοι ελέγχου, ποιο είναι το πραγματικό CPI του προγράμματος;

- A) 2.16
- B) 2.20
- C) 2.43
- D) 2.10
- E) 2.28

29) Με βάση το πραγματικό CPI του ερωτήματος 28, ποια είναι η πραγματική επίδοση του επεξεργαστή σε MIPS;

- A) περίπου 1235 MIPS
- B) περίπου 1429 MIPS
- C) περίπου 1200 MIPS
- D) περίπου 1500 MIPS
- E) περίπου 1091 MIPS

ΣΚΕΛΟΣ ASSEMBLY : ΘΕΜΑ 1: ΑΣΚΗΣΗ ΠΟΛΛΑΠΛΩΝ ΕΡΩΤΗΜΑΤΩΝ (Ερωτήματα 30-50, 35,7 μονάδες)

Δίνονται οι κάτωθι δηλώσεις μεταβλητών (extra set 12):

var60 DWORD 16h,3h,16h,11h

var61 SBYTE 11h,-11h,02Eh,15h

var62 SWORD 29h,-09Ah,-09Fh

var63 SDWORD 0E3h,-01AAh,0EAh,01Bh

var64 WORD 153h,44h,01B0h,189h,01B7h

Να βρεθεί για τα ακόλουθα υποερωτήματα το περιεχόμενο του καταχωρητή ή της θέσης μνήμης προορισμού, θεωρώντας ότι η αποθήκευση γίνεται με little-endian byte order και ξεκινά από τη σχετική διεύθυνση 00000000h.

Οι αποδεκτές απαντήσεις με σειρά προτεραιότητας είναι οι ακόλουθες:

i) Δεκαεξαδική τιμή με το πλήθος των ψηφίων που αντιστοιχεί στον καταχωρητή προορισμού.

ii) Η λέξη ADDRESS ακολουθούμενη από έναν ακέραιο δεκαδικό που δείχνει τη σχετική διεύθυνση.

iii) Η φράση SYNTAX ERROR με επεξήγηση του λάθους.

iv) Η φράση LOGICAL/RUNTIME ERROR με επεξήγηση του λάθους.

v) Η φράση UNDETERMINED VALUE.

vi) Για την επιλογή «Κανένα από τα παραπάνω», δίνεται η δική σας απάντηση.

Ισοδύναμα αρνητικών αριθμών σε συμπλήρωμα ως προς 2:

• -11h = EFh (SBYTE) • -09Ah = FF66h (SWORD) • -09Fh = FF61h (SWORD) • -01AAh = FFFFFFFE56h (SDWORD)

Συμπληρώστε τον παρακάτω πίνακα για να σας βοηθήσει:

Variable (Όνομα)	Relative Address (Δεκαδικό)	Value (Byte)
	0	
	1	
	2	
	3	
	4	
	5	
	6	
	7	
	8	
	9	
	10	
	11	
	12	
	13	
	14	
	15	
	16	
	17	
	18	
	19	
	20	
	21	
	22	
	23	
	24	
	25	
	26	
	27	
	28	
	29	
	30	
	31	
	32	
	33	
	34	
	35	
	36	
	37	
	38	
	39	
	40	
	41	
	42	
	43	
	44	

	45	
	46	
	47	
	48	
	49	
	50	
	51	

30) mov ebx, OFFSET var61

- A) ADDRESS(15)
- B) ADDRESS(20)
- C) ADDRESS(17)
- D) ADDRESS(26)
- E) ADDRESS(16)

31) mov esi, [ebx]

- A) 11EF2E15h
- B) 00000011h
- C) 152EEF11h
- D) EF110000h
- E) 00002EEFh

32) mov ebx, var61[10]

- A) SYNTAX ERROR
- B) 000000E3h
- C) FFFFFFFE3h
- D) 00000056h
- E) LOGICAL/RUNTIME ERROR

33) mov ecx, TYPE var62

- A) 00000001h
- B) 00000002h
- C) 00000003h
- D) 00000006h
- E) 00000004h

34) mov eax, OFFSET var60

- A) ADDRESS(1)
- B) ADDRESS(16)
- C) ADDRESS(42)
- D) ADDRESS(0)
- E) ADDRESS(20)

35) inc eax

- A) ADDRESS(0)
- B) ADDRESS(2)
- C) ADDRESS(16)
- D) ADDRESS(42)
- E) ADDRESS(1)

36) mov ch, [eax]

- A) 00h
- B) 16h
- C) 03h
- D) EFh
- E) 11h

37) inc [eax]

- A) 01h
- B) LOGICAL/RUNTIME ERROR
- C) SYNTAX ERROR
- D) 17h
- E) 00h

38) mov esi, 5C132297h

- A) 9722135Ch
- B) 5C132297h
- C) 5C132200h
- D) 0000005Ch
- E) 5C130000h

39) mov ax, [esi]

- A) UNDETERMINED VALUE
- B) 2297h
- C) 5C13h
- D) LOGICAL/RUNTIME ERROR
- E) SYNTAX ERROR

40) movsx ecx, -23h

- A) FFFFFFFDDh
- B) FFFFFFF23h
- C) 000000DDh
- D) 00000023h
- E) SYNTAX ERROR

41) mov eax, var63

- A) 000000E3h
- B) E3000000h
- C) FFFFFFFE56h
- D) 00000056h
- E) 000000EAh

42) mov ebx, LENGTHOF var60

- A) 00000010h
- B) 00000001h
- C) 00000004h
- D) 00000002h
- E) 00000008h

43) mov esi, var60[10]

- A) 00000011h
- B) 00110000h
- C) 11000000h
- D) 00001100h
- E) 16000011h

44) mov bl, -01Ah

- A) 1Ah
- B) D6h
- C) F6h
- D) E6h
- E) 26h

45) mov eax, OFFSET var64+1

- A) ADDRESS(42)
- B) ADDRESS(44)
- C) ADDRESS(41)
- D) ADDRESS(46)
- E) ADDRESS(43)

46) lea ecx, [eax-1]

- A) ADDRESS(43)
- B) ADDRESS(41)
- C) ADDRESS(44)
- D) ADDRESS(46)
- E) ADDRESS(42)

47) mov cx, [ecx-2]

- A) 0000h
- B) 001Bh

- C) 0153h
- D) 5301h
- E) 0001h

48) mov cx, SIZEOF var62

- A) 0003h
- B) 0006h
- C) 0002h
- D) 0004h
- E) 000Ch

49) mov cx, var61[34]

- A) 01B7h
- B) B701h
- C) SYNTAX ERROR
- D) 00B7h
- E) B700h

50) mov bx, var64[4]

- A) 0189h
- B) 0153h
- C) B001h
- D) 01B0h
- E) 0044h

ΘΕΜΑ 2 - Μετατροπή C σε Assembly / Έλεγχος ορθότητας (Ερωτήματα 51-56, 10,2 μονάδες)

Δίνεται ο παρακάτω κώδικας C:

```
short int a, b;
```

```
if (a < b) {
```

```
    a = 5;
```

```
}
```

```
b = b - 1;
```

Δίνονται οι παρακάτω υλοποιήσεις σε Assembly (Intel/MASM σύνταξη):

A)	B)	C)	D)	E)
1. .data	1. .data	1. .data	1. .data	1. .data
2. a DWORD ?	2. a SWORD ?	2. a SWORD ?	2. a SWORD ?	2. a SWORD ?
3. b DWORD ?	3. b SWORD ?	3. b SWORD ?	3. b SWORD ?	3. b SWORD ?
4. ...	4. ...	4. ...	4. ...	4. ...
5. mov ax, a	5. mov ax, a	5. mov ax, a	5. mov ax, a	5. mov ax, a
6. cmp ax, b	6. cmp a, b	6. cmp ax, b	6. cmp ax, b	6. cmp ax, b
7. jge L_EXIT	7. jge L_EXIT	7. jge L_EXIT	7. jl L_EXIT	7. jge L_EXIT
8. mov a, 5	8. mov a, 5	8. mov a, 5	8. mov a, 5	8. mov b, 5
9. L_EXIT:	9. L_EXIT:	9. L_EXIT:	9. L_EXIT:	9. L_EXIT:
10. dec b	10. dec b	10. dec b	10. dec b	10. dec b

51) Ποια από τις υλοποιήσεις A-E είναι σωστή και αποδοτική;

- A) A
- B) B
- C) C
- D) D
- E) E

52) Εξετάζοντας ΜΟΝΟ τον κώδικα A, ποια επιλογή αποτελεί την πιο ακριβή διάγνωση ως προς την ορθότητά του;

- A) Το jge πρέπει να αντικατασταθεί από jae, επειδή οι μεταβλητές είναι 16-bit.
- B) Η εντολή dec b βρίσκεται μέσα στο THEN και δεν εκτελείται πάντα.
- C) Το mov a,5 ενημερώνει λάθος μεταβλητή σε σχέση με τον κώδικα C.
- D) Οι a,b δηλώνονται ως DWORD, ενώ ζητείται 16-bit signed short int.
- E) Δεν υπάρχει λάθος.

53) Εξετάζοντας ΜΟΝΟ τον κώδικα Β, ποια επιλογή περιγράφει σωστά το σφάλμα;

- A) Οι μεταβλητές έχουν δηλωθεί με λάθος μέγεθος για short int.
- B) Η cmp έχει δύο τελεστέους μνήμης, κάτι που δεν επιτρέπεται.
- C) Το jge είναι unsigned άλμα και δεν ταιριάζει σε signed short int.
- D) Το THEN παραλείπεται μόνο στην περίπτωση $a > b$ και όχι στην ισότητα.
- E) Δεν υπάρχει λάθος.

54) Εξετάζοντας ΜΟΝΟ τον κώδικα C, ποια επιλογή είναι η σωστή διάγνωση;

- A) Οι δηλώσεις πρέπει να είναι DWORD, επειδή ο επεξεργαστής είναι 32-bit.
- B) Η σύγκριση έχει δύο τελεστές μνήμης.
- C) Το jge είναι unsigned άλμα.
- D) Το THEN γράφει στη b αντί στη a.
- E) Δεν υπάρχει λάθος.

55) Εξετάζοντας ΜΟΝΟ τον κώδικα D, ποια επιλογή δίνει την πιο κρίσιμη παρατήρηση;

- A) Το jl οδηγεί σε έξοδο όταν $a < b$, δηλαδή παραλείπει το THEN όταν η συνθήκη είναι αληθής.
- B) Η δήλωση των μεταβλητών έχει λάθος μέγεθος για short int.
- C) Η cmp χρησιμοποιεί δύο τελεστές μνήμης.
- D) Το jl είναι unsigned άλμα και πρέπει να αντικατασταθεί από jae.
- E) Δεν υπάρχει λάθος.

56) Εξετάζοντας ΜΟΝΟ τον κώδικα E, ποια επιλογή είναι η σωστή διάγνωση;

- A) Η δήλωση των μεταβλητών έχει λάθος μέγεθος.
- B) Η cmp χρησιμοποιεί δύο τελεστές μνήμης.
- C) Το σώμα του THEN ενημερώνει τη b αντί για τη a.
- D) Το jge παραλείπει λανθασμένα την περίπτωση $a == b$.
- E) Δεν υπάρχει λάθος.

ΘΕΜΑ 3: Ανάλυση Πλαισίου Στοιβάς (Stack Frame) (Ερωτήματα 57-63, 11,9 μονάδες)

Για το παρακάτω τμήμα κώδικα C, να απαντηθούν οι ερωτήσεις που αφορούν το περιεχόμενο θέσεων μνήμης στο πλαίσιο ενεργοποίησης (activation record) της καλούμενης υπορουτίνας Callee. Οι διευθύνσεις είναι εκφρασμένες ως προς τον καταχωρητή EBP της Callee.

1. int i=8h;
2. unsigned char v[3]={0D1h,99h,08Dh};
- 3.
4. void Caller() {
5. char l=39h;
6. unsigned short int w[5]={04Ch,0Eh,0C9h,02Eh,79h};
7. Callee(i,w,l,v,0090h);
8. }
- 9.
10. void Callee(int ix,unsigned short int *wx, char lx,unsigned char *vx, unsigned short int immx) {
11. int t=-0E0h;
12. char s[5]={-57h,-0A3h,36h,-0A9h,07Ah};
13. ...
14. }

Οι αποδεκτές απαντήσεις είναι:

1. Αριθμητική τιμή
2. Η λέξη ADDRESS
3. Η φράση RETURN ADDRESS
4. Η λέξη EBP
5. Η λέξη REGISTERS

57) Τι περιέχει η θέση μνήμης στη διεύθυνση EBP + 4;

- A) EBP
- B) RETURN ADDRESS
- C) ADDRESS
- D) 8h
- E) 0090h

58) Τι περιέχει η θέση μνήμης στη διεύθυνση EBP + 12;

- A) 39h
- B) RETURN ADDRESS
- C) ADDRESS
- D) 07Ah
- E) -0E0h

59) Τι περιέχει η θέση μνήμης στη διεύθυνση EBP - 8;

- A) -0A9h
- B) 07Ah
- C) -0E0h
- D) REGISTERS
- E) ADDRESS

60) Τι περιέχει η θέση μνήμης στη διεύθυνση EBP + 24;

- A) 39h
- B) ADDRESS
- C) RETURN ADDRESS
- D) 0090h
- E) 8h

61) Τι περιέχει η θέση μνήμης στη διεύθυνση EBP - 16;

- A) ADDRESS
- B) -0A9h
- C) 36h
- D) REGISTERS
- E) EBP

62) Τι περιέχει η θέση μνήμης στη διεύθυνση EBP - 12;

- A) -0E0h
- B) -0A9h
- C) 07Ah
- D) REGISTERS
- E) 0090h

63) Τι περιέχει η θέση μνήμης στη διεύθυνση EBP + 8;

- A) ADDRESS
- B) 39h
- C) RETURN ADDRESS
- D) 0090h
- E) 8h