



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Όνοματεπώνυμο: _____

ΑΜ: _____

ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΥΠΟΛΟΓΙΣΤΩΝ

(τμήμα Λ - Ω)

Εξέταση Σεπτεμβρίου 2015

Διάρκεια 2,5 ώρες

Οι εξετάσεις θα πραγματοποιηθούν ΧΩΡΙΣ την παρουσία βιβλίων, βοηθημάτων ή άλλου είδους σημειώσεων. Το μόνο που επιτρέπεται να φέρετε είναι ένα φύλλο Α4 με προσωπικές σημειώσεις.

ΘΕΜΑ 1^ο

A) Χαρακτηρίστε κάθε μία από τις ακόλουθες προτάσεις ως Σωστή (Σ) ή Λάθος (Λ):

1. Μια εντολή απαιτεί λιγότερο χρόνο να εκτελεστεί σε ένα επεξεργαστή με σωλήνωση από ότι σε ένα αντίστοιχο χωρίς σωλήνωση (όλα τα άλλα χαρακτηριστικά είναι ίδια).
2. Η κύρια μνήμη του υπολογιστή είναι μη πτητική (non-volatile).
3. Ο χρόνος απόκρισης ισούται με τη συνολική διάρκεια εκτέλεσης μιας εργασίας.
4. Για την αποθήκευση κρίσιμων δεδομένων (critical data) ενδείκνυται η χρήση δίσκων σε συστοιχία RAID 0.
5. Η αναπαράσταση των bytes στον επεξεργαστή MIPS είναι Big Endian.
6. Το εύρος των μη προσημασμένων δυαδικών ακεραίων στον επεξεργαστή MIPS (αρχιτεκτονικής 32-bit) είναι $0 - 2^{32}$.
7. Ένας κίνδυνος ελέγχου (control hazard) οδηγεί πάντα σε καθυστέρηση (stall).
8. Υιοθέτηση μεγαλύτερου μεγέθους block οδηγεί σε μείωση του παρατηρούμενου ρυθμού αστοχίας.

B) Απαντήστε σύντομα στα κάτωθι:

1. Ένας ιδανικός επεξεργαστής σωλήνωσης θα είχε CPI ίσο με 1.0. Δώστε δύο παραδείγματα στα οποία ένας πραγματικός επεξεργαστής έχει παραπάνω.
2. Δίδονται δύο επεξεργαστές, CPU0 και CPU1. Ο πρώτος (CPU0) διαθέτει σωλήνωση 5 σταδίων με κύκλο ρολογιού διάρκειας 5ns. Ο δεύτερος (CPU1) διαθέτει σωλήνωση 7 σταδίων με κύκλο ρολογιού

διάρκειας 4ns. Εξηγείστε ποιος παρουσιάζει μικρότερη καθυστέρηση (latency) και ποιος μεγαλύτερη απόδοση (throughput). Ένα πρόγραμμα σε ποιον από τους δύο θα εκτελείται πιο γρήγορα;

Γ) Θεωρήστε τις ακόλουθες δύο ρουτίνες σε C:

```
int dummySwap (int v[], int n) {
    swap(v, n-1);
    swap(v, n-2);
}

void swap (int v[], int k) {
    int temp;
    temp = v[k];
    v[k] = v[k+1];
    v[k+1] = temp;
}
```

Έστω ότι η διεύθυνση του πρώτου στοιχείου του πίνακα v (πρώτο όρισμα της swap) βρίσκεται αποθηκευμένη στον καταχωρητή \$a0, ενώ ο ακέραιος k (δεύτερο όρισμα της swap) στον καταχωρητή \$a1.

1. Συμπληρώστε τα κενά στο μεταγλωττισμένο κώδικα της swap σε MIPS assembly:

```
swap:
    sll $t1, ____, 2
    add $t1, $t1, ____
    lw $t0, 0($t1)
    lw $t2, ____
    sw $t2, 0($t1)
    sw $t0, ____
    ____
```

2. Ποιους από τους καταχωρητές χρειάζεται ο αντίστοιχος κώδικας της dummySwap να αποθηκεύσει προτού καλέσει τη swap;

ΘΕΜΑ 2^ο

Δίδεται ο παρακάτω κώδικας σε MIPS assembly που αντιστοιχεί στη συνάρτηση strcpy της C:

<pre>//copies contents of character array y //into character array x //we're assuming that the two arrays //are the same size void strcpy(char x[], char y[]) { int i = 0; while (y[i] != 0) { x[i] = y[i]; i = i + 1; } x[i] = 0; }</pre>	<pre># assumptions: # 1) base address of x is in \$a0 # 2) base address of y is in \$a1 # 3) \$t0 will hold value of i # remember that we do not have to # multiply i by 4 in this case because # characters only take 1 byte # to store, not 1 word strcpy: add \$t0, \$zero, \$zero loop: add \$t1, \$a1, \$t0 lb \$t2, 0(\$t1) beq \$t2, \$zero, out add \$t3, \$a0, \$t0 sb \$t2, 0(\$t3) addi \$t0, \$t0, 1 j loop out: add \$t3, \$a0, \$t0 sb \$zero, 0(\$t3) jr \$ra</pre>
--	--

Η συνάρτηση strcpy δέχεται δύο ορίσματα που είναι πίνακες χαρακτήρων και επιχειρεί να αντιγράψει έναν-έναν τους χαρακτήρες του δεύτερου στον πρώτο. Το τέλος της συμβολοσειράς (string) σηματοδοτείται από τον κενό χαρακτήρα (NULL ή 0 σε κώδικα ASCII). Ο κώδικας είναι ελαφρώς διαφορετικός από αυτόν των διαφανειών και για λόγους απλότητας παραλείφθηκαν οι εντολές διαχείρισης της στοίβας.

Υποθέστε ότι έχουμε αρχιτεκτονική σωλήνωσης (pipelining) 5 σταδίων (IF, ID, EX, MEM, WB) και ότι όλες οι αναφορές στη μνήμη ικανοποιούνται από την κρυφή μνήμη σε 1 κύκλο (δεν υπάρχουν δηλαδή αστοχίες). Επίσης, η εγγραφή σε κάποιον καταχωρητή γίνεται στο πρώτο μισό ενός κύκλου, ενώ η ανάγνωση από τον ίδιο καταχωρητή στο δεύτερο μισό του ίδιου κύκλου. Υποθέστε επίσης ότι όλες οι αναφορές στη μνήμη ικανοποιούνται σε έναν κύκλο (δεν υπάρχουν δηλαδή αστοχίες), καθώς και ότι η απόφαση για μια εντολή διακλάδωσης λαμβάνεται στο στάδιο MEM.

A) Αρχικά υποθέτουμε έλλειψη σχήματος προώθησης. Συμπληρώστε το διάγραμμα χρονισμού για την πρώτη επανάληψη του βρόχου, επισημαίνοντας κάθε φορά το λόγο που προκαλείται η οποιαδήποτε καθυστέρηση. Υπολογίστε τον **συνολικό αριθμό κύκλων** που απαιτούνται για την ολοκλήρωση του προγράμματος συναρτήσει του μήκους της συμβολοσειράς προς αντιγραφή.

B) Στη συνέχεια υποθέτουμε την ύπαρξη σχήματος προώθησης. Παρουσιάστε το διάγραμμα χρονισμού της πρώτης επανάληψης του βρόχου. Υπολογίστε τον **νέο συνολικό αριθμό κύκλων** που απαιτούνται μέχρι το πέρας του προγράμματος.

Γ) Είναι δυνατόν να επιτευχθεί καλύτερη απόδοση χωρίς να αλλάξει η σημασιολογία του προγράμματος, όταν παρέχεται σχήμα **προώθησης**; Αν ναι, υποδείξτε τον τρόπο και υπολογίστε το νέο πλήθος απαιτούμενων κύκλων για την ολοκλήρωση του βελτιστοποιημένου πλέον κώδικα.

ΘΕΜΑ 3^ο

A) Έστω ότι διαθέτουμε τους τέσσερις παρακάτω δίσκους σε διάταξη RAID 5, με τα παρακάτω περιεχόμενα:

	DISK0	DISK1	DISK2	DISK3
STRIPE0	0010	1111	1010	
STRIPE1	0100	1011		0010
STRIPE2	1000		1001	0011
STRIPE3		0001	0000	1011

1. Να συμπληρωθούν τα ψηφία ισοτιμίας στον παραπάνω πίνακα (σκιασμένα κελιά).
2. Να ορισθεί πως θα γίνει η εγγραφή της νέας τιμής 0101 στο STRIPE2 του DISK0.
3. Υποθέστε ότι ο DISK3 καταστρέφεται και το σύστημα εμφανίζει την παρακάτω εικόνα:

	DISK0	DISK1	DISK2	DISK3
STRIPE0	0001	1101	0100	****
STRIPE1	0011	1000	1110	****
STRIPE2	1101	0110	1001	****
STRIPE3	0101	1001	1011	****

Να εξηγηθεί πως θα γίνει η ανάγνωση-ανάκτηση του STRIPE3 του κατεστραμμένου δίσκου.

B) Δίνεται ένα πρότυπο κινητής υποδιαστολής 24 bits αντίστοιχο με το IEEE 754, όπου όμως ο εκθέτης έχει μήκος 10 bits και το κλάσμα (μαντίσσα) 13 bits.

1. Ποια είναι η πόλωση;
2. Βρείτε τον απολύτως μεγαλύτερο αριθμό που μπορεί να παρασταθεί από το πρότυπο αυτό.

ΘΕΜΑ 4^ο

Δίνονται οι παρακάτω βρόχοι:

Loop A	Loop B
<pre>sum = 0; for (i = 0; i < 128; i++) for (j = 0; j < 64; j++) sum += A[i][j];</pre>	<pre>sum = 0; for (j = 0; j < 128; j++) for (i = 0; i < 64; i++) sum += A[i][j];</pre>

Ο πίνακας A περιέχει 32-bit ακεραίους και είναι αποθηκευμένος στη μνήμη κατά γραμμές. Δίνεται επίσης ότι μόνο οι προσβάσεις στον πίνακα A προκαλούν πρόσβαση στη μνήμη και ότι όλες οι άλλες μεταβλητές είναι αποθηκευμένες σε καταχωρητές.

A) Έστω ότι το σύστημα περιέχει μια direct-mapped L1 data cache μεγέθους 8KB με block μεγέθους 32 bytes. Αν η cache αρχικά είναι άδεια, ποιος είναι ο αριθμός των misses που προκαλεί η εκτέλεση του Loop A; Αντίστοιχα, ποιος είναι ο αριθμός των misses για το Loop B;

B) Υποθέστε ότι σας δίνεται η δυνατότητα να αντικαταστήσετε την cache με μια άλλη direct-mapped cache με μέγεθος block πάλι 32 bytes. Υπολογίστε τον ελάχιστο αριθμό cache lines που απαιτείται προκειμένου να εκτελεστεί το Loop A χωρίς κανένα άλλο miss πέρα από τα compulsory. Ποιος είναι ο αντίστοιχος αριθμός για την εκτέλεση του Loop B;

Γ) Υποθέστε ότι ο επεξεργαστής έχει τώρα μια fully-associative L1 data cache, μεγέθους 8KB με block μεγέθους 32 bytes. Αν η πολιτική αντικατάστασης είναι FIFO, υπολογίστε τον αριθμό των misses που προκαλεί η εκτέλεση του Loop A. Αντιστοίχως για το Loop B.