

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ και ΥΠΟΛΟΓΙΣΤΩΝ

ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΕΠΙΣΤΗΜΗ ΤΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Μέρος Β' (Οργάνωση Υπολογιστών)

Υποδείξεις των λύσεων για τα θέματα της κανονικής εξέτασης Ιουλίου 2005
(καθηγητής: Νεκτάριος Κοζύρης)

Θέμα Β1 (20%):

(α) Να παρασταθεί ο δεκαδικός αριθμός 27,75

- i) σε κανονική μορφή, για έναν υπολογιστή που ακολουθεί το πρότυπο του ΕΚΥ, ο οποίος έχει λέξη των 32 bits και αφιερώνει: 1 bit για το πρόσημο, 8 bits για τον πολωμένο εκθέτη και 23 bits για το συντελεστή
- ii) σύμφωνα με το πρότυπο απλής ακρίβειας IEEE 754 (βλ. πίνακα σελ. 20 σημειώσεων)

(β) Πόσοι διαφορετικοί πραγματικοί αριθμοί μπορούν να παρασταθούν με τη μορφή α(i) και πόσοι με τη μορφή α(ii) του προηγούμενου ερωτήματος; Δικαιολογήστε την απάντησή σας.

(γ) Ποιος είναι ο ελάχιστος κατά μέτρο αριθμός και ποιος ο μέγιστος πραγματικός αριθμός που μπορεί να παρασταθεί σε κάθε μία από τις παραπάνω μορφές; Ποια είναι η απόσταση μεταξύ δύο διαδοχικών πραγματικών αριθμών που μπορούν να παρασταθούν, έχοντας εκθέτη ίσο με τον ελάχιστο και ίσο με το μέγιστο πραγματικό αριθμό που μόλις υπολογίσατε για κάθε περίπτωση;

Υπόδειξη λύσης*

(α) Ο αριθμός 27,75 μετατρέπεται σε δυαδικό, με διαδοχικές διαιρέσεις για το ακέραιο μέρος και με διαδοχικούς πολλαπλασιασμούς για το δεκαδικό μέρος. Η δυαδική αναπαράσταση που προκύπτει είναι: **11011,11**

- i) Προκειμένου να παρασταθεί σε κανονική μορφή του ΕΚΥ, ο αριθμός αυτός γράφεται ισοδύναμα: **0,1101111_{<2>} 2⁵**. Στον υπολογιστή αναπαρίσταται ως εξής:

0	10000101	110111100000000000000000
πρόσημο +	εκθέτης = 5+πώλωση 128	συντελεστής

- ii) Προκειμένου να παρασταθεί σύμφωνα με το πρότυπο IEEE, σύμφωνα με την 1^η περίπτωση του αντίστοιχου πίνακα (σελ. 20) των σημειώσεων, ο προηγούμενος αριθμός γράφεται ισοδύναμα: **1,101111_{<2>} 2⁴**. Στον υπολογιστή αναπαρίσταται ως εξής:

0	10000011	101111000000000000000000
πρόσημο +	εκθέτης = 4+πώλωση 127	συντελεστής

(β) (i) Το 1^ο bit του συντελεστή είναι πάντα ίσο με 1. Τα υπόλοιπα 31 bits μπορούν να παίρνουν οποιαδήποτε τιμή, παριστάνοντας κάθε φορά διαφορετικό πραγματικό αριθμό. Άρα, έχουμε **2³¹** διαφορετικούς πραγματικούς αριθμούς.

(ii) Από τον πίνακα της σελ. 20 των σημειώσεων προκύπτει:

πρόσημο	→	2 περιπτώσεις
εκθέτης	→	Κυμαίνεται από 0 έως 254 για τους πραγματικούς → 255 περιπτώσεις
συντελεσ.	→	Έχει 23 bits. που παίρνουν οποιαδήποτε τιμή → 2 ²³ περιπτώσεις

Συνολικά, έχουμε **255 2²⁴** διαφορετικές παραστάσεις πραγματικών αριθμών. Όμως, το μηδεν μπορεί να παρασταθεί με δύο διαφορετικούς τρόπους, ως +0 και ως -0. Άρα, μπορούν τελικά να παρασταθούν **255 2²⁴-1** διαφορετικοί πραγματικοί αριθμοί.

(γ) (i) Μέγιστος:

* Οι παρούσες απαντήσεις των θεμάτων αποτελούν απλώς υποδείξεις των λύσεων και όχι πλήρως αιτιολογημένες απαντήσεις, όπως θα έπρεπε να γραφούν σε ένα διαγώνισμα. Στα σημεία που δίδεται απλά το αποτέλεσμα των πράξεων, οι φοιτητές θα πρέπει να γράφουν και τις πράξεις που τους οδήγησαν στο αποτέλεσμα αυτό. Επίσης, τα προγράμματα πρέπει να συνοδεύονται από περιγραφή του αλγορίθμου που ακολουθείται και σχόλια για την κατανόησή τους.

μέγιστος εκθέτης 127
 μέγιστος συντελεστής 0,111 ... 1_{<2>} = $1-2^{-23}$
 Άρα έχουμε τον αριθμό $(1-2^{-23})2^{127}$.

Ελάχιστος:

ελάχιστος εκθέτης -128
 ελάχιστος συντελεστής 0,100 ... 0_{<2>} = 2^{-1}
 Άρα έχουμε τον αριθμό 2^{-129} .

Μεταξύ δύο διαδοχικών αριθμών αλλάζει μόνο το τελευταίο bit του συντελεστή. Άρα, η διαφορά τους είναι $2^{-23}2^{\text{εκθέτης}}$

Για την περίπτωση του μεγίστου έχουμε: $2^{-23}2^{127} = 2^{104}$

Για την περίπτωση του ελαχίστου έχουμε: $2^{-23}2^{-128} = 2^{-151}$

(ii) Μέγιστος:

μέγιστος εκθέτης 254-127 = 127
 μέγιστος συντελεστής 1,111 ... 1_{<2>} = $2-2^{-23}$
 Άρα έχουμε τον αριθμό $(1-2^{-24})2^{128}$.

Ελάχιστος: το μηδέν 0

Όπως και στην προηγούμενη περίπτωση, μεταξύ δύο διαδοχικών αριθμών αλλάζει μόνο το τελευταίο bit του συντελεστή. Άρα, η διαφορά τους είναι $2^{-23}2^{\text{εκθέτης}}$

Για την περίπτωση του μεγίστου έχουμε: $2^{-23}2^{127} = 2^{104}$

Για την περίπτωση του ελαχίστου έχουμε: $2^{-23}2^{-126}$ (2^η περίπτωση του πίνακα σελ. 20) = 2^{-149}

Θέμα B2 (30%):

(α) Έστω το παρακάτω τμήμα κώδικα σε γλώσσα ASSEMBLY του EKY:

ADDRESS	CON	3
RESULT	RES	1
	LDA	INSTRUCT
	ADA	ADDRESS
	STA	RESULT
	HLT	
INSTRUCT	SAL	0

Ποιο θα είναι το περιεχόμενο των θέσεων της μνήμης με labels INSTRUCT, ADDRESS, RESULT, σε δεκαεξαδική μορφή πριν και μετά την εκτέλεση του προγράμματος αυτού; Το περιεχόμενο της διεύθυνσης με label RESULT, μετά την εκτέλεση του προγράμματος, σε ποια εντολή αντιστοιχεί;

(β) Έστω ότι στις θέσεις μνήμης X και Y του EKY είναι αποθηκευμένοι δύο δυαδικοί αριθμοί, x και y, αντίστοιχα. Να φτιάξετε πρόγραμμα σε γλώσσα ASSEMBLY του EKY, το οποίο να υπολογίζει και να αποθηκεύει στη θέση RESULT το αποτέλεσμα της τέλεσης πράξης AND στα αντίστοιχα bits των αριθμών x και y (bitwise AND).

Υποδείξεις:

- 1) Στον EKY δεν υπάρχουν εντολές λογικών πράξεων.
- 2) Οι εντολές ολισθήσης SAL, SAR, δέχονται ως όρισμα έναν αριθμό, όχι μια διεύθυνση της μνήμης. Προκειμένου να καθορίσετε εντός του προγράμματος πόσες ολισθήσεις θα πρέπει να γίνουν, μπορείτε να χρησιμοποιήσετε τα συμπεράσματα του ερωτήματος (α)

Υπόδειξη λύσης

(α) Το περιεχόμενο των διευθύνσεων με labels INSTRUCT, ADDRESS, πριν και μετά την εκτέλεση του προγράμματος είναι:

	περιεχόμενο θέσης μνήμης	
	κώδικας εντολής	τμήμα διεύθυνσης
INSTRUCT	B	000
ADDRESS	0	003

Το περιεχόμενο της θέσης της μνήμης με label RESULT πριν την εκτέλεση του προγράμματος είναι απροσδιόριστο. Μετά την εκτέλεση έχει την τιμή

	περιεχόμενο θέσης μνήμης	
	κώδικας εντολής	τμήμα διεύθυνσης
RESULT	B 003	

Αντιστοιχεί, επομένως, στην εντολή SAL 3.

(β) 1^η λύση

Το παρακάτω πρόγραμμα, σε κάθε επανάληψη απομονώνει ένα bit των αριθμών x και y ως εξής:

αρχικά	X ₁₅	X ₁₄	X ₁₃	X ₁₂	X ₁₁	X ₁₀	X ₉	X ₈	X ₇	X ₆	X ₅	X ₄	X ₃	X ₂	X ₁	X ₀
ολίσθηση i θέσεις αριστερά	X ₁₁	X ₁₀	X ₉	X ₈	X ₇	X ₆	X ₅	X ₄	X ₃	X ₂	X ₁	X ₀	0	0	0	0
ολίσθηση 15 θέσεις δεξιά	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	X ₁₁

Προκειμένου να γίνονται σε κάθε επανάληψη διαφορετικός αριθμός ολισθήσεων προς τα αριστερά, θα χρησιμοποιήσουμε τη μέθοδο του προηγούμενου ερωτήματος, δηλαδή θα γράφουμε την εντολή που πρέπει να εκτελεστεί στην κατάλληλη διεύθυνση.

Το λογικό AND μεταξύ δύο αριθμών που έχουν μη μηδενικό μόνο το LSB (least significant bit) ισοδυναμεί με την πράξη του πολλαπλασιασμού. Επομένως, στη συνέχεια πολλαπλασιάζουμε τους δύο αριθμούς που προέκυψαν με τις παραπάνω ολισθήσεις. Τέλος τροποποιούμε το περιεχόμενο της θέσης RESULT ως εξής:

από προηγούμενη επανάληψη	0	0	0	0	0	0	0	0	0	0	0	0	xy ₁₅	xy ₁₄	xy ₁₃	xy ₁₂
ολίσθηση 1 θέση αριστερά	0	0	0	0	0	0	0	0	0	0	0	0	xy ₁₅	xy ₁₄	xy ₁₃	xy ₁₂
πρόσθεση γινομένου	0	0	0	0	0	0	0	0	0	0	0	0	xy ₁₅	xy ₁₄	xy ₁₃	xy ₁₂

X	CON	x	
Y	CON	y	
RESULT	RES	1	
SIXTEEN	CON	16	
ONE	CON	1	
TEMP	RES	1	
AD_SAL	RES	1	; Βοηθητική θέση μνήμης, δείχνει σε ποια επανάληψη βρισκόμαστε
	SAL	16	; Αρχικοποίηση - μηδενισμός μεταβλητών
	STA	AD_SAL	
	STA	RESULT	
LOOP	LDA	COM_SAL	; Τροποποίηση εντολών ολισθήσεως, ώστε να
	ADA	AD_SAL	; γίνονται ακριβώς όσες ολισθήσεις
	STA	EX_SAL_X	; χρειάζονται, σύμφωνα με το ερώτημα (α)
	STA	EX_SAL_Y	
EX_SAL_X	LDA	X	; Απομόνωση ενός bit του x
	SAL	*	
	SAR	15	
	STA	TEMP	
EX_SAL_Y	LDA	Y	; Απομόνωση ενός bit του y
	SAL	*	
	SAR	15	
	MLA	TEMP	; Πολλαπλασιασμός δύο bits = Λογικό AND
	STA	TEMP	

```

LDA     RESULT    ; Τοποθέτηση του αποτελέσματος στο
SAL     1         ; κατάλληλο bit του RESULT
ADA
STA     TEMP
STA     RESULT

LDA     AD_SAL    ; Αύξηση μετρητή επαναλήψεων και
ADA     ONE       ; ολισθήσεων, ώστε να προετοιμαστεί
STA     AD_SAL    ; η επόμενη επανάληψη
SBA     SIXTEEN   ; Ελέγχουμε
JAN     LOOP      ; αν πρέπει να γίνει άλλη επανάληψη
HLT

```

```

COM_SAL SAL     0

```

(β) 2^η λύση

Στη συνέχεια, δίνουμε μία λύση του ίδιου προβλήματος, η οποία δε χρησιμοποιεί τη λογική του ερωτήματος (α). Σε κάθε επανάληψη φροντίζουμε να έρθουν στο MSB (most significant bit) των αριθμών x και y τα bits πάνω στα οποία θέλουμε να εκτελεστεί η πράξη AND. Στη συνέχεια αναγνωρίζουμε αν τα bits αυτά ισούνται με 0 ή 1, ανάλογα με το αποτέλεσμα της εντολής JAN. Αν, δηλαδή, πρόκειται για bit 1, ο συνολικός αριθμός που δημιουργείται εκλαμβάνεται ως αρνητικός ακέραιος. Η τοποθέτηση του αποτελέσματος στο κατάλληλο bit του RESULT γίνεται με τον ίδιο τρόπο, όπως και στην προηγούμενη λύση.

από προηγούμενη επανάληψη	0	0	0	0	0	0	0	0	0	0	0	0	0	xy ₁₅	xy ₁₄	xy ₁₃	xy ₁₂
ολίσθηση 1 θέση αριστερά	0	0	0	0	0	0	0	0	0	0	0	0	xy ₁₅	xy ₁₄	xy ₁₃	xy ₁₂	0
πρόσθεση 1 ή τίποτα	0	0	0	0	0	0	0	0	0	0	0	0	xy ₁₅	xy ₁₄	xy ₁₃	xy ₁₂	xy₁₁

```

X      CON      x
Y      CON      y
RESULT RES      1
X_SH  RES      1
Y_SH  RES      1
N_1   CON      15
ONE   CON      1

```

```

SAL     16       ; Αρχικοποίηση - μηδενισμός μεταβλητών
STA     RESULT
LDA     X
STA     X_SH
LDA     Y
STA     Y_SH

```

```

LDI     N_1      ; Αρχικοποίηση καταχωρητή B, προκειμένου
              να γίνουν 16 επαναλήψεις, όσα και τα
              bits των θέσεων X, Y
LOOP    LDA     RESULT ; Ολίσθηση RESULT, προκειμένου να
SAL     1       ; τοποθετηθεί το αποτέλεσμα στο κατάλληλο
STA     RESULT  ; bit

```

```

LDA     X_SH
JAN     CHECK_Y ; Αν το υπό εξέταση bit του x ισούται με
              1, πρέπει να ελέγξουμε και το
              αντίστοιχο bit του y, αλλιώς το
              αποτέλεσμα είναι 0.

```

```

END     LDA     X_SH ; Προετοιμασία για την επόμενη επανάληψη
SAL     1
STA     X_SH
LDA     Y_SH
SAL     1

```

```
          STA      Y_SH
          INJ      LOOP
          HLT

CHECK_Y   LDA      Y_SH      ; Αν το υπό εξέταση bit του y είναι 1,
          JAN      FIX_RES   ; πάμε να βάλουμε 1 στο αντίστοιχο bit
          JMP      END       ; του RESULT, αλλιώς μένει ίσο με 0

FIX_RES   LDA      RESULT
          ADA      ONE
          STA      RESULT
          JMP      END
```