ΟΝΟΜΑ ΑΕΜ

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ – ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ, ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΚΑΙ ΔΙΚΤΥΩΝ

ΠΡΟΟΔΟΣ ΣΤΗΝ «ΕΙΣΑΓΩΓΗ ΣΤΟΥΣ Η/Υ (ΗΥ134)»

Τετάρτη, 27 Μαρτίου 2013

ΔΙΑΡΚΕΙΑ ΔΙΑΓΩΝΙΣΜΑΤΟΣ 2 ΩΡΕΣ

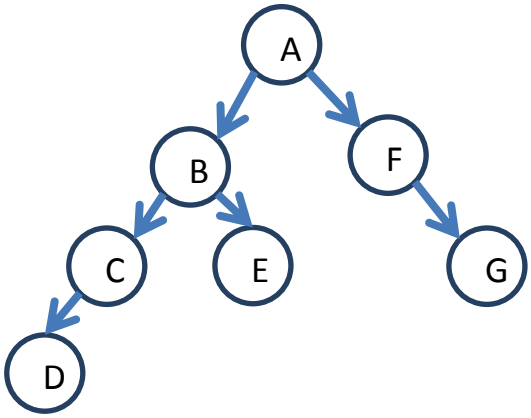
ΚΑΛΗ ΕΠΙΤΥΧΙΑ!

ΘΕΜΑ 1 (35 μονάδες, 15+10+10)

Ένα *δυαδικό δέντρο* είναι μία δομή δεδομένων που αποτελείται από ένα σύνολο κόμβων που συνδέονται από ακμές. Ορίζεται αναδρομικά ως εξής:

- **Ένα δυαδικό δένδρο είναι είτε κενό είτε**
- **αποτελείται από μια ρίζα (root), δηλαδή ένα κόμβο στον οποίο δεν καταλήγουν αλλά μόνο ξεκινούν ακμές, και μέχρι 2 υποδέντρα T_1, T_2 το καθένα ξεχωριστό από τα άλλα και από τη ρίζα. Υπάρχει μια ακμή από τη ρίζα στις ρίζες των T_1, T_2.**

Ο βαθμός (degree) ενός κόμβου είναι ο αριθμός των παιδιών του και είναι 0, 1, ή 2. Η άσκηση σας ζητάει να απαντήσετε στα παρακάτω ερωτήματα:



a) Να γράψετε την αναδρομική συνάρτηση *count_nodes* που υπολογίζει τον αριθμό των κόμβων ενός δυαδικού δέντρου:

$$count\_nodes(root\ *R) = \begin{cases} 0, \text{if R = NULL} \\ 1 + count\_nodes(R->left) + count\_nodes(R->right), \text{if R <> NULL} \end{cases}$$

ΟΝΟΜΑ ΑΕΜ

b) Υποθέστε ότι υλοποιούμε την συνάρτηση *count_nodes* σε MIPS assembly χρησιμοποιώντας αναδρομή. Ποιοί είναι οι καταχωρητές που θα πρέπει οπωσδήποτε να σώζονται στην στοίβα σε κάθε κλήση της συνάρτησης *count_nodes*; Ζητάμε τους ελάχιστους καταχωρητές που θα πρέπει να σώζονται ώστε η αναδρομή να εκτελεσθεί σωστά.

Οι καταχωρητές $a0 και $ra πρέπει να σώζονται οπωσδήποτε στην στοίβα σε κάθε κλήση της συνάρτησης *count_nodes.* Ο καταχωρητής $a0 θα περιέχει το όρισμα εισόδου *R->right* και το όρισμα εισόδου *R->left*.

c) Ποιό είναι το μέγιστο μέγεθος της στοίβας σε bytes που απαιτείται για την εκτέλεση της αναδρομής *count_nodes(root *R)* στην γενική περίπτωση ενός δυαδικού δέντρου. Να βρεθεί το μέγιστο μέγεθος της στοίβας για το συγκεκριμένο δέντρο που σας δίδεται στην εικόνα.

Μέγιστο μέγεθος στοίβας = 2*4*height of tree.

Στο συγκεκριμένο παράδειγμα Μέγιστο μέγεθος στοίβας = 2*4*4 = 32 bytes.

ΟΝΟΜΑ ΑΕΜ

ΘΕΜΑ 2 (35 μονάδες)

(64-bit unsigned SUB). Η αρχιτεκτονική εντολών του MIPS περιέχει πράξεις μόνο μεταξύ 32-bit ακεραίων αριθμών. Είναι πολλές φορές αναγκαίο σε ένα πρόγραμμα να εκτελέσουμε πράξεις μεταξύ ορισμάτων διαφορετικής ακρίβειας όπως, για παράδειγμα, 8-bits ή 64-bits. Η άσκηση αυτή σας ζητάει να γράψετε κώδικα σε MIPS assembly που να εκτελεί σωστά ΑΦΑΙΡΕΣΗ δύο ακεραίων 64-bit αριθμών που είναι αποθηκευμένοι στα ζεύγη καταχωρητών ($t0, $t1) και ($t2, $t3). Το αποτέλεσμα να αποθηκεύεται στο ζεύγος καταχωρητών ($t0, $t1).

Σημείωση:
Για την σωστή αφαίρεση θα πρέπει να δοθεί προσοχή στο κρατούμενο εξόδου (carry out) που παράγεται από την αφαίρεση δύο ακεραίων 32-bit αριθμών, εάν ο $t1 < $t3.
Η άσκηση μπορεί να λυθεί το πολύ με 5 εντολές assembly. Χρήση παραπάνω εντολών θα επιφέρει μείωση βαθμολογίας.

	($t0 $t1)	Όρισμα 1
-	($t2 $t3)	Όρισμα 2
	($t0 $t1)	Διαφορά

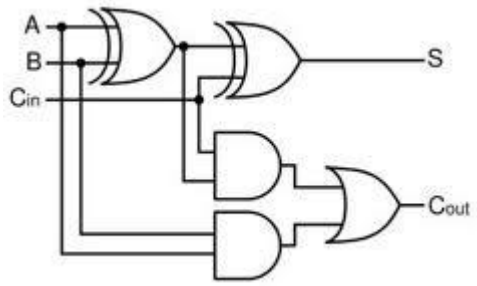
```
subu $t1, $t1, $t3  # sub $t3 from $t1
# If there is a carry out (i.e. $t1 < $t3),
# the result in $t1 is larger in than both $t1 and $t3.
sgtu $t4, $t1, $t3  # set carry-in bit
addu $t2, $t2, $t4  # add in $t4 (0 or 1) to $t2
subu $t0, $t0, $t2  # add in second most significant word
```

ΘΕΜΑ 3 (30 μονάδες, 15+15)

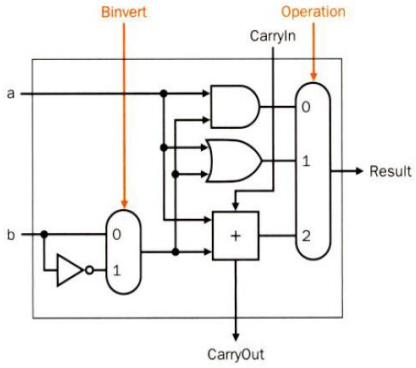
Να δοθεί ο πίνακας αλήθειας (truth table) ενός πλήρους αθροιστή (full adder) του 1-bit, και να σχεδιάσετε το λογικό διάγραμμα του αθροιστή αυτού.

A	B	Cin	S	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

ΟΝΟΜΑ ΑΕΜ



b) Να δοθεί το βασικό κυκλωματικό διάγραμμα μιας ALU του 1-bit η οποία υλοποιεί τις εντολές *add*, *sub*, *and*, και or του MIPS. Ποιες θα είναι οι τιμές των εισόδων ελέγχου του κυκλώματος της ALU για εκτέλεση κάθε μιας από τις εντολές; Τα διαθέσιμα εξαρτήματα είναι βασικές πύλες, πολυπλέκτες N-σε-1, και ο πλήρης αθροιστής του 1-bit.



	Binvert	Operation
ADD	0	10
SUB	1	10
AND	0	00
OR	0	01