

Οργάνωση και Σχεδίαση Υπολογιστών (ECE 219)

Χειμερινό Εξάμηνο 2020-2021

Εργαστήριο 3

8% του συνολικού εργαστηριακού βαθμού

Άσκηση 1- Αναδρομικός Αλγόριθμος Εύρεσης στοιχείου (5 μονάδες)

Να γράψετε ένα αναδρομικό πρόγραμμα MIPS assembly το οποίο θα βρίσκει και θα επιστρέφει την θέση idx ενός τοπικού μέγιστου σε έναν πίνακα ακεραίων A . Το τοπικό μέγιστο $A[idx]$ έχει τις εξής ιδιότητες:

$$\begin{aligned} A[idx - 1] &\leq A[idx] \geq A[idx + 1], & \text{εάν } 0 \leq idx \leq n - 1 \\ A[idx] &\geq A[idx + 1], & \text{εάν } idx = 0 \\ A[idx] &\geq A[idx - 1], & \text{εάν } idx = n - 1 \end{aligned}$$

Για παράδειγμα, εάν
η είσοδος είναι $A = [-3 \ 5 \ 7 \ 10 \ 8 \ 9]$
τότε το Output θα είναι 3.

Εάν είσοδος είναι $A = [0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 0]$
τότε το Output θα είναι 7.

```
/* Συνάρτηση εύρεσης τοπικού μέγιστου στον πίνακα A. Αρχικά η συνάρτηση καλείται από την  
main ως findElement (A, 0, n-1, n); */  
int findElement (int A[], int low, int high, int n)
```

Σε μία λίστα είναι πιθανόν να υπάρχουν πολλά στοιχεία με αυτές τις ιδιότητες. Μας ενδιαφέρει να βρούμε οποιοδήποτε από αυτά τα στοιχεία.
Επίσης, προσέξτε ότι η λίστα δεν είναι υποχρεωτικό να είναι ταξινομημένη.

Πριν γράψετε το πρόγραμμα σε assembly καλό είναι να αναπτύξετε την αναδρομική σχέση σε κάποια ψευδογλώσσα (ή ακόμα καλύτερα σε C ώστε να τρέξετε και τον κώδικα σας). Οποιαδήποτε μη αναδρομική λύση δεν θα γίνεται δεκτή.

Άσκηση 2- Ευθυγράμμιση (3 μονάδες)

Θεωρείστε την παρακάτω δομή δεδομένων σε C:

```
struct foo {  
    char a;  
    bool b;    // Implement in an effective way  
    int c;  
    double d;
```

```

short e;
float f;
double g;
char *cptr;
float *fptr;
int x;
}

```

Ποιο είναι το μέγεθος της struct foo για έναν επεξεργαστή 32-bit; Ποιο είναι το ελάχιστο μέγεθος της struct foo υποθέτοντας ότι μπορούμε να αλλάξουμε την σειρά των πεδίων της struct foo με όποιον τρόπο θέλουμε; Επαναλάβετε τις ίδιες ερωτήσεις για έναν επεξεργαστή 64-bit.

Σημειώστε ότι εκτός από τις απαιτήσεις για ευθυγραμμισμένα πεδία μέσα σε μία struct, ή ίδια η struct πρέπει να είναι επίσης ευθυγραμμισμένη. Εξηγείστε ποιος είναι αυτός ο τελευταίος περιορισμός, και πρέπει να τον βάζουμε. Σημ. σκεφτείτε τι ακριβώς γίνεται όταν έχουμε για παράδειγμα έναν πίνακα από τέτοιες structs (πχ. struct foo[100]).

Άσκηση 3- Αριθμητική Κινητής Υποδιαστολής (2 μονάδες)

Πράξεις με αριθμούς κινητής υποδιαστολής. Οι απαντήσεις (και μόνο αυτές) σε κάθε ερώτηση θα πρέπει να δοθούν αποκλειστικά μέσα στα κουτιά.

Έστω οι αριθμοί κινητής υποδιαστολής $A = -0,865625 \times 10^1$ και $B = 1,5546875 \times 4^2$. Θεωρείστε το σύστημα αριθμητικής κινητής υποδιαστολής FP_PETIT με 4 bits στο πεδίο του σημαντικού (significand) και 5 bits στο πεδίο του πολωμένου εκθέτη (exponent). Το bit προσήμου είναι το ίδιο με το IEEE FP Standard 754.

1	5	4
Πρόσημο	Εκθέτης	Σημαντικό

- Ποια είναι η πόλωση του συστήματος FP_PETIT έτσι ώστε ο εκθέτης των 5 bits να είναι πάντα θετικός αριθμός;
- Να γραφούν οι αριθμοί A και B σε κανονικοποιημένη δυαδική μορφή και μετά οι 10-bit αναπαραστάσεις των αριθμών αυτών σύμφωνα με το σύστημα FP_PETIT. Προσοχή με τυχόν στρογγυλοποιήσεις.
- Να γραφεί το άθροισμα $\text{Sum} = A + B$ σε κανονικοποιημένη δυαδική μορφή και μετά η 10-bit αναπαράσταση του Sum σύμφωνα με το σύστημα FP_PETIT.
- Να γραφεί το γινόμενο $\text{Prod} = A * B$ σε κανονικοποιημένη δυαδική μορφή και μετά η 10-bit αναπαράσταση του Prod σύμφωνα με το σύστημα FP_PETIT.
- Να γραφεί το πηλίκο $\text{Div} = A / B$ σε κανονικοποιημένη δυαδική μορφή και μετά η 10-bit αναπαράσταση του Div σύμφωνα με το σύστημα FP_PETIT.