



# ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ  
ΕΡΓΑΣΤΗΡΙΟ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ  
[www.cslab.ece.ntua.gr](http://www.cslab.ece.ntua.gr)

## 1η ΑΣΚΗΣΗ ΣΤΗΝ ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΥΠΟΛΟΓΙΣΤΩΝ

Ακ. έτος 2021-2022, 5ο Εξάμηνο, Σχολή ΗΜ&ΜΥ

Τελική Ημερομηνία Παράδοσης: **28/11/2021**

### ΜΕΡΟΣ Α

Δίνεται το παρακάτω πρόγραμμα σε C καθώς και μια μετάφραση του σε assembly MIPS. Θεωρήστε πως οι μεταβλητές  $x$  και  $y$  είναι αποθηκευμένες στους καταχωρητές  $\$s2$  και  $\$s3$  αντίστοιχα, ενώ οι καταχωρητές  $\$s0$  και  $\$s1$  περιέχουν τη διεύθυνση του πρώτου στοιχείου των πινάκων `array1` και `array2` αντίστοιχα. Συμπληρώστε τα κενά. Σας υπενθυμίζουμε ότι ο καταχωρητής  $\$0$  (ή  $\$zero$ ) είναι πάντα μηδέν.

```
int array1[30], array2[30], count;  
int *x, *y;
```

```
count = 0;  
x = &array1[5];  
y = &array2[10];
```

```
for (count = 0; count < 10; count++) {  
    if (*x > *y)  
        *x += *y;  
    else  
        *y -= *x;  
    x++;  
    y++;  
}
```

```
add    ____, $zero, $zero  
addi   ____, $s0,  ____,  
addi   ____, $s1,  ____,  
LOOP:  ____, $t1, 0($s2)  
lw     $t2, 0($s3)  
slt    $t3, ____, ____,  
____, $t3, ____, ____,  
____, ____, ____, ____,  
sw     ____, ____,  
jmp     ____  
else:  ____, ____, ____, ____,  
sw     ____, ____,  
next:  addi   $s2, $s2, ____,  
        addi   $s3, $s3, ____,  
        addi   $t0, $t0, 1  
        subi   $t5, ____, 10  
bne    ____, ____, ____,
```

**ΜΕΡΟΣ Β**

Οι παρακάτω ρουτίνες σε C υλοποιούν τους αλγορίθμους δυαδικής (binary) και εκθετικής (exponential) αναζήτησης καθώς και την αναζήτηση με παρεμβολή (interpolation search) σε έναν ταξινομημένο πίνακα ακεραίων με N στοιχεία. Οι ρουτίνες επιστρέφουν τη θέση μέσα στον πίνακα που βρίσκεται ο ακέραιος που αναζητούμε ή -1 σε περίπτωση που ο ακέραιος δεν βρεθεί. Υλοποιήστε τις ρουτίνες σε assembly του MIPS.

```
int binary_search(int *A, int N, int key) {
    return binary_search_rec(A, 0, N-1, key);
}

int binary_search_rec(int *A, int left, int right, int key) {
    int mid;

    if (right < left) return -1;

    mid = left + (right - left) / 2;
    if (A[mid] == key) return mid;

    else if (A[mid] > key)
        return binary_search_rec(A, left, mid-1, key);
    else
        return binary_search_rec(A, mid+1, right, key);
}

int exponential_search(int *A, int N, int key) {
    int bound = 1;

    while(bound < N && A[bound] < key) {
        bound *=2;
    }

    if (bound < N-1)
        return binary_search_rec(A, bound/2, bound, key);
    else
        return binary_search_rec(A, bound/2, N-1, key);
}

int interpolation_search(int *A, int N, int key) {
    int low = 0, up = N-1, pos;

    while (low <= up) {
        if ((key < A[low]) || (key > A[up]))
            return -1;

        pos = low + (up-low) * (key-A[low]) / (A[up]-A[low]);

        if (A[pos] == key)
            return pos;
        else if (A[pos] > key)
            up = pos - 1;
        else
            low = pos+1;
    }
    return -1;
}
```

**ΜΕΡΟΣ Γ**

Υλοποιήστε ξανά την αναζήτηση με παραμεβολή (interpolation search) χρησιμοποιώντας αυτή τη φορά αναδρομή.

Για την υλοποίηση της άσκησης μπορείτε να χρησιμοποιήσετε τον **MILE**, ένα MIPS emulator που αναπτύχθηκε από συμφοιτητές σας και διατίθεται από το Εργαστήριο Υπολογιστικών Συστημάτων (<http://www.cslab.ece.ntua.gr/courses/comparch/assign.go>). Στον emulator αυτό, μπορείτε να γράφετε MIPS assembly και να την εκτελείτε παρακολουθώντας τα περιεχόμενα των καταχωρητών και της μνήμης καθιστώντας έτσι ευκολότερη την παραγωγή και τον έλεγχο του απαιτούμενου κώδικα.

Παραδοτέο της άσκησης θα είναι ένα ηλεκτρονικό κείμενο (**pdf, docx ή odt**) που θα περιέχει τους κώδικες assembly και των 3 μερών της άσκησης. Ο κώδικας θα πρέπει να περιέχει αναλυτικά σχόλια για την κατανόηση της λύσης σας από τους διδάσκοντες.

Στο ηλεκτρονικό κείμενο να αναφέρετε στην αρχή τα στοιχεία σας (Όνομα, Επώνυμο, ΑΜ).

Η άσκηση θα παραδοθεί ηλεκτρονικά στο moodle του μαθήματος:

<https://helios.ntua.gr/course/view.php?id=1037>

*Δουλέψτε ατομικά. Έχει ιδιαίτερη αξία για την κατανόηση του μαθήματος να κάνετε μόνοι σας την εργασία. Μην προσπαθήσετε να την αντιγράψετε από άλλους συμφοιτητές σας.*