



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
Ακ. έτος 2022-2023, 5ο εξάμηνο, ΣΗΜΜΥ

**ΤΜΗΜΑ 1ο (Α - ΚΑΣ)**  
**1<sup>η</sup> ΣΕΙΡΑ ΑΣΚΗΣΕΩΝ ΣΤΗΝ ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΥΠΟΛΟΓΙΣΤΩΝ**  
Ημερομηνία παράδοσης: 6/11/2022

**ΟΝΟΜΑΤΕΠΩΝΥΜΟ:** Αλεξοπούλου Γεωργία

**ΑΜ:** 03120164

**Άσκηση 1**

**A)** Ερμηνεία του κώδικα:

$\$s1 \rightarrow i$ ,  $\$s0 \rightarrow \text{arr}[]$

$\text{int } i = 0, t2 = 1000, t0 = 0$

loop: if ( $i < 1000$ )  $t0 = 1$

if ( $t0 == 0$ ) jump to done

$t0 = 4 * i$

$t0 = \&\text{arr}[i]$

$t1 = t0$

$t1 = 2^3 * t1 = 8 * t0$

$i++$

end of loop

Μετάφραση σε C:

```
for (int i = 0; i < 1000; i++) {  
    arr[i] = 8*arr[i]; }
```

**B)** Ερμηνεία κώδικα:

\$a0 → charr[]

li \$v0, 0 → 0

loop: t0 = a0

if (t0 == 0) jump to done

v0 = v0 + 1

a0 = a0 + 1

end of loop

done: return

Μετάφραση σε C:

char \*A = new char

int i = 0;

while (A[i] != '\0') {

    A[i] ++ ;

    i++; }

return i;

## Άσκηση 2

Ο κώδικας C για να ελέγξουμε αν ένα string είναι παλινδρομικό είναι ο εξής:

int palindrome (char str[]) {

    int l = 0;

    int h = strlen(str) - 1;

    while (h > l) {

        if (str[l++] != str[h--]) {

            return 0; }

        else return 1;

```
}  
}
```

Άρα, η μετάφραση σε MIPS:

```
.data
```

```
input: .asciz "ANNA"
```

```
length .word 4
```

```
.text
```

```
main:
```

```
la $a0, length
```

```
lw $a0, 0($a0)
```

```
la $a1, input
```

```
la $s0, 0
```

```
addi $s1, length, -1
```

```
jal Palindrome
```

```
syscall
```

```
Palindrome:
```

```
beq $a0, $zero, True
```

```
beq $a0, 1, True
```

```
lb $t0, $s0($a1)
```

```
lb $t1, $s1($a1)
```

```
bne $t0, $t1, False
```

```
addi $s0, $s0, 1
```

```
addi $s1, $s1, -1
```

```
addi $a0, $a0, -2
```

```
slt $t2, $s0, $s1
```

```
beq $t2, 1, Palindrome
```

False:

```
addi $v0, $zero, 0
```

```
jr $ra
```

True:

```
addi $v0, $zero, 1
```

```
jr $ra
```