
Punë Laboratori 7

Ushtrim 1:

Çfarë bën programi i mëposhtëm? Shpjegoni përgjigjen tuaj.

```
.data
array: .word 6, 2, -33, 10, 4, 12, 4
length: .word 7 .text
.globl main
main:
    la $a0, array lw
    $a1, length jal
    max add $a0, $v0,
    $zero li $v0, 1
    syscall li $v0, 10
    syscall
max:
    add $t0, $zero, $zero
    add $v0, $zero, $zero
    add $v1, $zero, $zero
cikel:
    sltu $t2, $t0, $a1
    beq $t2, $zero,
    fund lw $t1, 0($a0)
    sltu $t2, $t1, $v0
    bne $t2, $zero,
    kalo add $v0, $t1,
    $zero add $v1, $t0,
    $zero
kalo:
    addi $t0, $t0, 1
    addi $a0, $a0, 4
    j cikel
fund:
    jr $ra
```

Për secilin instruksion vendosni komentën korrespondues.

Ku vendoset në memorje e dhëna **array**? Po e dhëna **length**? Shpjegoni përgjigjen tuaj.

Tregoni përmbajtjen e regjistrave në fund të ekzekutimit të programit. Shpjegoni përmbajtjen e secilit prej tyre.

Ushtrim 2:

Ndërtoni një program që shfaq në ekran elementët e një vektori (me 8 elementë) që janë shumfisha të numrit pesë. (Vektori të jetë variabël global). Programi të përmbajë procedurën **main** edhe një procedurë tjetër. Numri i elementëve që do të kontrollohen të merret si input nga përdoruesi.

Të supozohet që inputi i dhënë nga përdoruesi nuk tejkalon përmasën e vektorit.

(p.sh. nëse vektori ka elementët: 7 15 14 25 5 115 16 16, edhe përdoruesi jep si input vlerën 4, atëherë në ekran do të shfaqen vetëm numrat 15, 25).

Ushtrimi 1

```
.data
array: .word 6, 2, -33, 10, 4, 12, 4
length: .word 7
.text
.globl main

# Funkcioni max gjen vlerën maksimale dhe pozicionin e saj në vektor

main:
    # Vendos adresën e fillimit të vektorit në $a0
    la $a0, array
    # Ngarko gjatësinë e vektorit në $a1
    lw $a1, length
    # Thirr funksionin max për të gjetur vlerën maksimale dhe pozicionin e saj
    jal max

    # Shfaq vlerën maksimale në konsolë
    li $v0, 1
    syscall

    # Dalje nga programi
    li $v0, 10
    syscall

max:
    # Inicializo variablat $v0 dhe $v1 me vlerat fillestare
    add $v0, $zero, $zero    # $v0 = 0 (vlera maksimale fillestare)
    add $v1, $zero, $zero    # $v1 = 0 (pozicioni fillestar i vlerës maksimale)

    # Loop për të gjetur vlerën maksimale dhe pozicionin e saj
    # i: iteratori për vektorin
cikel:
    sltu $t2, $t0, $a1        # Kontrollon nëse iteratori është më i vogël se gjatësia
    # e vektorit
    beq $t2, $zero, fund      # Nëse nuk, shko në fund të funksionit

    # Ngarko vlerën në indeksin aktual të vektorit
    lw $t1, 0($a0)

    # Kontrolllo nëse vlera aktuale është më e vogël se vlera maksimale
    sltu $t2, $t1, $v0
    bne $t2, $zero, kalo      # Nëse po, kalo në iteracionin tjetër

    # Nëse vlera aktuale është më e madhe, ruaj vlerën dhe pozicionin e saj
    add $v0, $t1, $zero        # Vlera maksimale aktualizohet
    add $v1, $t0, $zero        # Pozicioni i vlerës maksimale aktualizohet

    # Kalo në iteracionin tjetër
    addi $t0, $t0, 1           # Rrit iteratorin për të kaluar në elementin tjetër
    addi $a0, $a0, 4           # Leviz në adresën e elementit tjetër të vektorit
    j cikel                   # Kthehu në fillim të ciklit

fund:
    jr $ra                   # Kthehu në thirrësin e shkëputur
```

Vlera maksimale në fund do të jetë më e madhja në vektor në vlerë absolute.

\$a0: adresa e fillimit të vektorit

\$a1: gjatësia e vektorit

\$v0: vlera maksimale

\$v1: pozicioni i vlerës maksimale në vektor

```
PC      = 400094
EPC     = 0
Cause   = 0
BadVAddr = 0
Status  = 3000ff10
```

```
HI      = 0
LO      = 0
```

```
R0 [r0] = 0
R1 [at] = 10010000
R2 [v0] = a
R3 [v1] = 0
R4 [a0] = 1001005c
R5 [a1] = 5
R6 [a2] = 7ffffee64
R7 [a3] = 0
R8 [t0] = 0
R9 [t1] = 6
R10 [t2] = 73
R11 [t3] = 5
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 10010018
R17 [s1] = 6
R18 [s2] = 0
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 10008000
R29 [sp] = 7ffffee58
R30 [s8] = 0
R31 [ra] = 400064
```

Ushtrimi 2

```
# arrayCount
.data
arrayA: .word :7,15,14,25, 5, 115,16,16
prompt: .ascii "Vendos numrin e elementëve që do të kontrollohen në vektor: "
space: .ascii " "
.text

main:

    la $s0, arrayA
    li $v0, 4
    la $a0, prompt
    syscall
    li $v0, 5
    syscall
    move $s1, $v0
    li $t1, 0          # Inicializo numrin për iteracionet
    li $t3, 5          # Inicializo numrin për shumefishat


LOOP:
    slt $t0, $t1, $s1
    beq $t0, $zero, FUND
    lw $t2, 0($s0)

    move $a0,$t2
    move $a1,$t3
    jal modulus

    bne $v0, $zero, SKIP

    move $a0,$t2
    li $v0, 1
    syscall
    la $a0, space
    li $v0, 4
    syscall

SKIP:
    addi $s0, $s0, 4
    addi $t1, $t1, 1
    j LOOP
FUND:
    li $v0, 10
    syscall
modulus:
mod_loop:
    sub $a0, $a0, $a1
    bgez $a0, mod_loop
    add $a0,$a0,$a1
    move $v0, $a0
    jr $ra
```

 Console

Vendos numrin e elementeve qe do te kontrollohen ne vektor: 6
15 25 5 115 |