**2. Sa se implementeze procedura prim(x), cu x numar intreg, care decide daca numarul x**

**este prim sau nu. (in $v0 ca .word)**

**Nota: returnez 1 => nr e prim**

**returnez 0 => nr nu e prim**

.data

.text

prim:

subu $sp, 4 #salvarea lui fp

sw $fp, 0($sp) # $sp:($fp v)(x)

addi $fp, $sp, 4 # $sp:($fpv)$fp:(x)

subu $sp, 4 # $sp:()($fpv)$fp:(x)

sw $s0, 0($sp) # $sp:($s0)($fpv)$fp:(x)

subu $sp, 4 # $sp:()($s0)($fpv)$fp:(x)

sw $s1, 0($sp) # $sp:($s1)($s0)($fpv)$fp:(x)

subu $sp, 4 # $sp:()($s1)($s0)($fpv)$fp:(x)

sw $s2, 0($sp) # $sp:($s2)($s1)($s0)($fpv)$fp:(x)

subu $sp, 4 # $sp:()($s2)($s1)($s0)($fpv)$fp:(x)

sw $s3, 0($sp) # $sp:($s3)($s2)($s1)($s0)($fpv)$fp:(x)

lw $s0, 0($fp) # s0 = x

ble $s0, 1, not\_prime # x <=1 -> nu e prim

li $s1, 2 #s1 va trece prin numerele de la 2 la [x/2]-1

div $s2, $s0, 2 # s2 = [x/2]

addi $s2, $s2, 1 # s2 = [x/2]+1

loop\_prime:

beq $s1, $s2, is\_prime #itereaza prin nr. E prim daca le parcurge pe toate

rem $s3, $s0, $s1 # t3 = x % i

beq $s3, $0, not\_prime # => restul impartirii e 0 => nu e prim

addi $s1, $s1, 1 #incrementeaza i

j loop\_prime

is\_prime:

li $v0, 1

j exit\_prim

not\_prime:

li $v0, 0

j exit\_prim

exit\_prim:

sw $s3, -20($fp) #restaurare s3, s2, s1, s0, fp

sw $s2, -16($fp)

sw $s1, -12($fp)

sw $s0, -8($fp)

sw $fp, -4($fp)

addu $sp, 20 # elibereaza stiva

jr $ra

main:

li $v0, 5 # read int

syscall

subu $sp, 4 #aloc spatiu in stiva

sw $v0, 0($sp) #salvez x pe stiva

jal prim

addu $sp, 4 #golim stiva

move $a0, $v0

li $v0, 1

syscall

li $v0, 10

syscall