**Problema 1**

.data

x: .word 1234567890

.text

main:

lw $t0, x

addu $sp, $sp, -4

sw $t0, 0($sp)

jal cifre\_pare

addu $sp, $sp, 4

li $v0, 10

syscall

cifre\_pare:

addu $sp, $sp, -4

sw $fp, 0($sp)

addu $fp, $sp, 4

addu $sp, $sp, -4

sw $s0, 0($sp)

lw $s0, 0($fp)

addu $sp, $sp, -4

sw $s1, 0($sp)

addu $sp,$sp, -4

sw $s2, 0($sp)

#sp: (s2)(s1)(s0)(Fp)fp:(x)

for:

ble $s0, 0, intoarcere\_la\_main

rem $s1, $s0, 10

div $s0, $s0, 10

rem $s2, $s1, 2

beqz $s2, afisare

j for

afisare:

move $a0, $s1

li $v0, 1

syscall

j for

intoarcere\_la\_main:

lw $s2, -16($fp)

lw $s1, -12($fp)

lw $s0, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 16

jr $ra

**Problema 2**

.data

x: .word -4

nu: .asciiz "Numarul nu e prim"

da: .asciiz "Numarul e prim"

.text

lw $t0, x

addu $sp, $sp, -4

sw $t0, 0($sp)

ble $t0, 1, afisare\_nu

jal prim

addu $sp, $sp, 4

beqz $v0, afisare\_nu

j afisare\_da

afisare\_nu:

la $a0, nu

li $v0, 4

syscall

li $v0, 10

syscall

afisare\_da:

la $a0, da

li $v0, 4

syscall

li $v0, 10

syscall

prim:

addu $sp, $sp, -4

sw $fp, 0($sp)

addu $fp, $sp, 4

addu $sp, $sp, -4

sw $s0, 0($sp)

lw $s0, 0($fp)

addu $sp, $sp, -4

sw $s1, 0($sp) #i

addu $sp, $sp, -4

sw $s2, 0($sp)

li $s1, 2

for:

bge $s1, $s0, intoarcere\_la\_main

rem $s2, $s0, $s1

beqz $s2, nu\_e\_prim

j cont

cont:

addi $s1, $s1, 1

j for

intoarcere\_la\_main:

li $v0, 1

lw $s2, -16($fp)

lw $s1, -12($fp)

lw $s0, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 16

jr $ra

nu\_e\_prim:

li $v0, 0

lw $s2, -16($fp)

lw $s1, -12($fp)

lw $s0, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 16

jr $ra

**Problema 3**

.data

x: .word 28

nu: .asciiz "Numarul nu e perfect"

da: .asciiz "Numarul e perfect"

.text

lw $t0, x

addu $sp, $sp, -4

sw $t0, 0($sp)

ble $t0, 0, afisare\_nu

jal perfect

addu $sp, $sp, 4

beqz $v0, afisare\_nu

j afisare\_da

afisare\_nu:

la $a0, nu

li $v0, 4

syscall

li $v0, 10

syscall

afisare\_da:

la $a0, da

li $v0, 4

syscall

li $v0, 10

syscall

perfect:

addu $sp, $sp, -4

sw $fp, 0($sp)

addu $fp, $sp, 4

addu $sp, $sp, -4

sw $s0, 0($sp)

lw $s0, 0($fp)

addu $sp, $sp, -4

sw $s1, 0($sp) #i

li $s1, 2

addu $sp, $sp, -4

sw $s2, 0($sp)#suma

addu $sp, $sp, -4

sw $s3, 0($sp)

li $s2, 1

addu $sp, $sp, -4

sw $s4, 0($sp)

div $s4, $s0, 2

for:

bgt $s1, $s4, intoarcere\_la\_main

rem $s3, $s0, $s1

beqz $s3, adaugare\_la\_suma

addi $s1, $s1, 1

j for

adaugare\_la\_suma:

addu $s2, $s2, $s1

addi $s1, $s1, 1

j for

intoarcere\_la\_main:

lw $s0, 0($fp)

beq $s0, $s2, e\_perfect

j nu\_e\_perfect

e\_perfect:

li $v0, 1

lw $s3, -20($fp)

lw $s2, -16($fp)

lw $s1, -12($fp)

lw $s0, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 20

jr $ra

nu\_e\_perfect:

li $v0, 0

lw $s3, -20($fp)

lw $s2, -16($fp)

lw $s1, -12($fp)

lw $s0, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 20

jr $ra

**Problema 4**

.data

a: .word 5

be: .word 13

x: .word 3

.text

main:

lw $t0, a

lw $t1, be

lw $t2, x

addu $sp, $sp, -12

sw $t2, 8($sp)

sw $t1, 4($sp)

sw $t0, 0($sp)

jal divizibile\_in\_interval

addu $sp, $sp, 12

li $v0, 10

syscall

divizibile\_in\_interval:

addu $sp, $sp, -4

sw $fp, 0($sp)

addu $fp, $sp, 4

addu $sp, $sp, -4

sw $s0, 0($sp)

lw $s0, 8($fp) #s0=a

addu $sp, $sp, -4

sw $s1, 0($sp) #i

lw $s1, 4($fp) #s1=b

addu $sp, $sp, -4

sw $s2, 0($sp)

lw $s2, 0($fp) #s2=x

addu $sp, $sp, -4

sw $s3, 0($sp)

move $s3, $s0 #i

addu $sp, $sp, -4

sw $s4, 0($sp) #rem

li $v0, 0

for:

bgt $s3, $s1, intoarcere\_la\_main

rem $s4, $s3, $s2

beqz $s4, adaugare\_la\_nr

addu $s3, $s3, 1

j for

adaugare\_la\_nr:

move $a0, $s3

li $v0, 1

syscall

addu $s3, $s3, 1

j for

intoarcere\_la\_main:

lw $s5, -28($fp)

lw $s4, -24($fp)

lw $s3, -20($fp)

lw $s2, -16($fp)

lw $s1, -12($fp)

lw $s0, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 28

jr $ra

**Problema 5**

.data

x: .word 2

.text

main:

lw $t0, x

addu $sp, $sp, -4

sw $t0, 0($sp)

jal f

addu $sp, $sp, 4

move $a0, $v0

li $v0, 1

syscall

li $v0, 10

syscall

f:

addu $sp, $sp, -4

sw $fp, 0($sp)

addu $fp, $sp, 4

addu $sp, $sp, -4

sw $ra, 0($sp)

addu $sp, $sp, -4

sw $s0, 0($sp)

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

addu $sp, $sp, -4

sw $s0, 0($sp)

jal g

addu $sp, $sp, 4

#sp: (s0)(ra)(fp)

mul $v0, $v0, 2

lw $s0, -12($fp)

lw $ra, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 12

jr $ra

g:

addu $sp, $sp, -4

sw $fp, 0($sp)

addu $fp, $sp, 4

addu $sp, $sp, -4

sw $s0, 0($sp)

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

addi $v0, $s0, 1

lw $s0, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 8

jr $ra

**Problema 6**

.data

n: .word 3

v: .word -6 1 7

sp: .asciiz " "

.text

main:

lw $t0, n

la $t1, v

addu $sp, $sp, -8

sw $t0, 4($sp)

sw $t1, 0($sp)

jal modifica

addu $sp, $sp, 8

li $t0, 1

li $t1, 0

lw $t2, v($t1)

lw $t3, n

j afisare

li $v0, 10

syscall

modifica:

addu $sp, $sp, -4

sw $fp, 0($sp)

addu $fp, $sp, 4

addu $sp, $sp, -4

sw $s0, 0($sp)

lw $s0, 4($fp) #s0=n

addu $sp, $sp, -4

sw $s1, 0($sp)

lw $s1, 0($fp) #s1=v\*

addu $sp, $sp, -4

sw $s2, 0($sp)

li $s2, 1 #s2=i

addu $sp, $sp, -4

sw $s3, 0($sp)

lw $s3, 0($s1) #v[i]

for:

bgt $s2, $s0, intoarcere\_la\_main

addu $s3, $s3, 1

sw $s3, 0($s1)

addi $s2, $s2, 1

addu $s1, $s1, 4

lw $s3, 0($s1)

j for

intoarcere\_la\_main:

lw $s3, -20($fp)

lw $s2, -16($fp)

lw $s1, -12($fp)

lw $s0, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 20

jr $ra

afisare:

bgt $t0, $t3, exit

move $a0, $t2

li $v0, 1

syscall

la $a0, sp

li $v0, 4

syscall

j cont

cont:

addi $t0, $t0, 1

addi $t1, $t1, 4

lw $t2, v($t1)

j afisare

exit:

li $v0, 10

syscall

**Problema 7**

.data

sir: .asciiz "aici tu"

vocale: .asciiz "aeiou"

a: .byte 'a'

e: .byte 'e'

i: .byte 'i'

o: .byte 'o'

u: .byte 'u'

sp: .asciiz " "

.text

main:

la $t0, sir

addu $sp, $sp, -4

sw $t0, 0($sp)

jal afiseaza\_vocale

addu $sp, $sp, 4

li $v0, 10

syscall

afiseaza\_vocale:

addu $sp, $sp, -4

sw $fp, 0($sp)

addu $fp, $sp, 4

addu $sp, $sp, -4

sw $ra, 0($sp)

addu $sp, $sp, -4

sw $s0, 0($sp)

addu $sp, $sp, -4

sw $s1, 0($sp)

lw $s0, 0($fp)

lb $s1, 0($s0)#s1=sir[i]

#sp: (s1)(s0)(ra)(fp)

for:

beq $s1, $zero, intoarcere\_la\_main

addu $sp, $sp, -4

sb $s1, 0($sp)

jal este\_vocala

addu $sp, $sp, 4

beq $v0, 1, afisare

addu $s0, $s0, 1

lb $s1, 0($s0)

j for

intoarcere\_la\_main:

lw $s1, -16($fp)

lw $s0, -12($sp)

lw $ra, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 16

jr $ra

afisare:

move $a0, $s1

li $v0, 11

syscall

la $a0, sp

li $v0, 4

syscall

addu $s0, $s0, 1

lb $s1, 0($s0)

j for

este\_vocala:

addu $sp, $sp, -4

sw $fp, 0($sp)

addu $fp, $sp, 4

addu $sp, $sp, -4

sw $s0, 0($sp)

addu $sp, $sp, -4

sw $s1, 0($sp)

lb $s0, 0($fp)#s0=ch

lb $s1, a

beq $s1, $s0, vocala

lb $s1, e

beq $s1, $s0, vocala

lb $s1, i

beq $s1, $s0, vocala

lb $s1, o

beq $s1, $s0, vocala

lb $s1, u

beq $s1, $s0, vocala

j nu\_e\_vocala

nu\_e\_vocala:

li $v0, 0

#sp:(s2)(S1)(s0)(fp)

lw $s1, -12($fp)

lw $s0, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 12

jr $ra

vocala:

li $v0, 1

#sp:(s2)(S1)(s0)(fp)

lw $s1, -12($fp)

lw $s0, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 12

jr $ra

**Problema 8**

.data

v: .word -5 11 12 13

n: .word 4

w: .space 400

sp: .asciiz " "

line: .asciiz "\n"

.text

main:

lw $t0, n

la $t1, v

addu $sp, $sp, -4 #incarc registrii

sw $t0, 0($sp)

addu $sp, $sp, -4

sw $t1, 0($sp)

#sp: (t3)(t2)(t1)(t0)

jal numere\_prime

addu $sp, $sp, 8

li $t2, 1

li $t4, 0

lw $t3, w($t4)

j afisare

li $v0, 10

syscall

afisare:

beq $t3, $zero, exit

move $a0, $t3

li $v0, 1

syscall

la $a0, sp

li $v0, 4

syscall

addi $t2, $t2, 1

addu $t1, $t1, 4

addu $t4, $t4, 4

lw $t3, w($t4)

j afisare

exit:

li $v0, 10

syscall

numere\_prime:

addu $sp, $sp, -4

sw $fp, 0($sp)

addu $fp, $sp, 4

#sp: (fp) fp:(\*v)(\*w)(n)(p)

addu $sp, $sp, -4

sw $ra, 0($sp)

#sp: (ra)(fp) fp:(\*v)(\*w)(n)(p)

addu $sp, $sp, -4

sw $s0, 0($sp)

addu $sp, $sp, -4

sw $s1, 0($sp)

addu $sp, $sp, -4

sw $s2, 0($sp)

addu $sp, $sp, -4

sw $s3, 0($sp)

addu $sp, $sp, -4

sw $s4, 0($sp)

addu $sp, $sp, -4

sw $s5, 0($sp)

addu $sp, $sp, -4

#sp: (s3)(s2)(s1)(s0)(ra)(fp) fp:(\*v)(\*w)(n)(p)

lw $s0, 4($fp)#n

lw $s1, 0($fp)#v\*

li $s3, 0#i

lw $s4, 0($s1)#v[i]

li $s5, 0#i pentru w

j for

for:

bge $s3, $s0, intoarcere\_la\_main

addu $sp, $sp, -4

sw $s4, 0($sp)

jal prim

beq $v0, 1, modificare

addu $s3, $s3, 1#i++

addu $s1, $s1, 4

lw $s4, 0($s1)

j for

modificare:

sw $s4, w($s5)

addu $s3, $s3, 1#i++

addu $s5, $s5, 4

addu $s1, $s1, 4

lw $s4, 0($s1)

j for

intoarcere\_la\_main:

#sp: (s7)(s6)(s5)(s4)(s3)(s2)(s1)(s0)(ra)(fp) fp:(\*v)(\*w)(n)(p)

lw $s5, -32($fp)

lw $s4, -28($fp)

lw $s3, -24($fp)

lw $s2, -20($fp)

lw $s1, -16($fp)

lw $s0, -12($fp)

lw $ra, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 32

jr $ra

prim:

addu $sp, $sp, -4

sw $fp, 0($sp)

addu $fp, $sp, 4

addu $sp, $sp, -4

sw $s0, 0($sp)

addu $sp, $sp, -4

sw $s1, 0($sp)

addu $sp, $sp, -4

sw $s2, 0($sp)

addu $sp, $sp, -4

sw $s3, 0($sp)

addu $sp, $sp, -4

sw $s4, 0($sp)

#sp:(s4)(s3)(s2)(s1)(s0)(fp)

lw $s0, 0($fp)#x

li $s2, 2#i

li $s3, 0#nr de divizori

ble $s0, 1, nu\_e\_prim

loop:

bge $s2, $s0, intoarcere\_la\_procedura

rem $s4, $s0, $s2

beqz $s4, nu\_e\_prim

addi $s2, $s2, 1

j loop

intoarcere\_la\_procedura:

#sp:(s4)(s3)(s2)(s1)(s0)(fp)

li $v0, 1

lw $s4, -24($fp)

lw $s3, -20($fp)

lw $s2, -16($fp)

lw $s1, -12($fp)

lw $s0, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 24

jr $ra

nu\_e\_prim:

li $v0, 0

lw $s4, -24($fp)

lw $s3, -20($fp)

lw $s2, -16($fp)

lw $s1, -12($fp)

lw $s0, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 24

jr $ra

**Problema 9**

.data

v: .space 400

x: .word 2

n: .space 4

y: .word 3

sp: .asciiz " "

line: .asciiz "\n"

.text

main:

li $v0, 5

syscall

sw $v0, n

li $t0, 0

li $t1, 0

lw $t2, n

jal citire\_v

lw $t0, y

lw $t1, x

lw $t2, n

la $t3, v

addu $sp, $sp, -4 #incarc registrii

sw $t0, 0($sp)

addu $sp, $sp, -4

sw $t1, 0($sp)

addu $sp, $sp, -4

sw $t2, 0($sp)

addu $sp, $sp, -4

sw $t3, 0($sp)

#sp: (t3)(t2)(t1)(t0)

jal evalueaza\_expresie

addu $sp, $sp, 16

move $a0, $v1

li $v0, 1

syscall

li $v0, 10

syscall

citire\_v:

bge $t0, $t2, intoarcere

li $v0, 5

syscall

sw $v0, v($t1)

addu $t1, $t1, 4

addu $t0, $t0, 1

j citire\_v

intoarcere:

jr $ra

afisare:

bge $t0, $t2, exit

move $a0, $t3

li $v0, 1

syscall

la $a0, sp

li $v0, 4

syscall

addi $t0, $t0, 1

addu $t4, $t4, 4

lw $t3, 0($t4)

j afisare

exit:

li $v0, 10

syscall

evalueaza\_expresie:

addu $sp, $sp, -4

sw $fp, 0($sp)

addu $fp, $sp, 4

#sp: (fp) fp:(\*v)(\*w)(n)(p)

addu $sp, $sp, -4

sw $ra, 0($sp)

#sp: (ra)(fp) fp:(\*v)(\*w)(n)(p)

addu $sp, $sp, -4

sw $s0, 0($sp)

addu $sp, $sp, -4

sw $s1, 0($sp)

addu $sp, $sp, -4

sw $s2, 0($sp)

addu $sp, $sp, -4

sw $s3, 0($sp)

addu $sp, $sp, -4

sw $s4, 0($sp)

addu $sp, $sp, -4

sw $s5, 0($sp)

addu $sp, $sp, -4

sw $s6, 0($sp)

#sp: (s3)(s2)(s1)(s0)(ra)(fp) fp:(\*v)(\*w)(n)(p)

lw $s0, 12($fp)#y

lw $s1, 8($fp)#x

lw $s2, 4($fp)#n

lw $s3, 0($fp)#v\*

li $s4, 0#i

lw $s5, 0($s3)#v[i]

li $v1, 0

j for

for:

bge $s4, $s2, intoarcere\_la\_main

addu $sp, $sp, -4

sw $s5, 0($sp)

jal suma\_cif

addu $sp, $sp, 4

mul $v0, $v0, $s1

div $s6, $s4, 2

mul $s6, $s6, $s0

subu $s6, $v0, $s6

mul $s6, $s6, $s6

addu $v1, $v1, $s6

addu $s4, $s4, 1#i++

addu $s3, $s3, 4

lw $s5, 0($s3)

j for

intoarcere\_la\_main:

#sp: (s7)(s6)(s5)(s4)(s3)(s2)(s1)(s0)(ra)(fp) fp:(\*v)(\*w)(n)(p)

lw $s6, -36($fp)

lw $s5, -32($fp)

lw $s4, -28($fp)

lw $s3, -24($fp)

lw $s2, -20($fp)

lw $s1, -16($fp)

lw $s0, -12($fp)

lw $ra, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 36

jr $ra

suma\_cif:

addu $sp, $sp, -4

sw $fp, 0($sp)

addu $fp, $sp, 4

addu $sp, $sp, -4

sw $s0, 0($sp)

addu $sp, $sp, -4

sw $s1, 0($sp)

addu $sp, $sp, -4

sw $s2, 0($sp)

#sp:(s4)(s3)(s2)(s1)(s0)(fp)

lw $s0, 0($fp)#x

li $v0, 0

li $s2, 10

loop:

beq $s0, $zero, intoarcere\_la\_eval

rem $s1, $s0, $s2

addu $v0, $v0, $s1

div $s0, $s0, $s2

j loop

intoarcere\_la\_eval:

#sp:(s4)(s3)(s2)(s1)(s0)(fp)

lw $s2, -16($fp)

lw $s1, -12($fp)

lw $s0, -8($fp)

lw $fp, -4($fp)

addu $sp, $sp, 16

jr $ra