

**Pregunta 1:**

**En què simplificaria molt el codi del programa un dels modes d'adreçament del simulador Ripes?**

L'inici del programa serà el següent: .

```
.define
    num 5
.data 00h
    mat1: db 1,2,3,4,5
    mat2: db 6,7,8,9,0
    mat3: db 0,0,0,0,0
.org 100h
```

Calculeu les mides del codi del vostre programa i el nombre de cicles per a la seva execució.

**Pregunta 2:**

**Quants cicles de rellotge triga en executar-se una instrucció aritmètic – lògica qualsevol? Feu servir el fitxer adjunt on especifica el ISA del 8085. Indica quina és la mida mitjana de les teves instruccions. Calcula els cicles per instrucció mitjà per aquests codis**

**Aritmètica:**

**ADD reg tarda 4 Cicles.**

**Lògica:**

**ANA reg tarda 4 Cicles.**

**La mitja de aquetes instruccions es de 4 a 7 cicles.**

**Pregunta/Tasca 3:**

**Pugeu el vostre codi i marqueu quina és la instrucció del vostre programa que triga més cicles en executar-se**

```
.define
    num 4
clau 55h
.data 00h
    mat1: db 1,2,3,4,5
    mat2: db 6,7,8,9,0
.org
    LXI H, mat1 - > 3 10
    MVI C, mat2
    MVI B, 0
    MVI D, num
loop:
    MOV E, M
```

```

        LDAX B
        ADD E
        STAX B
        INX H
        INX B
        DCR D
        J loop
    hlt
    -----

.define
    num 5
.data 00h
    mat1: db 1,2,3,4,5
    mat2: db 6,7,8,9,0
    mat3: db 0,0,0,0,0
.org 100h
    LXI H, mat1 -> 3 10
    LXI B, mat3
    MVI D, num 1 7
carga:
    MOV A,M 1 7
    STAX B 1 7
    INX H 1 6
    INX B
    DCR D 1 4
    JP carga 3 -> 7/10
    LXI H, mat2
    LXI B, mat3
    MVI D,num
loop:
    MOV E, M
    LDAX B 1 7
    ADD E 1 4
    STAX B
    INX H
    INX B
    DCR D
    JP loop
    hlt

```

Les que més cicles triguen a executar-se son JP i LXI ja que JP es un condicional i s'executa sin que deixi de ser positiu. Triguen 10 CIC.

#### **Pregunta/Tasca 4:**

**Traduiu el codi per fer-lo servir amb el simulador Ripes. Quants cicles triga en executar-se? Compareu els resultats (mida de codi, accessos a memòria i clicles promig per instrucció) amb els valors obtinguts per l'i8085.**

```

.data
mat1: .word 1,2,3,4,5
mat2: .word 6,7,8,9,0
.text

```

```

main:
    la a0, mat1
    addi a3, zero, 5
loop:
    beqz a3, end
    lw a1, 0(a0)
    lw a2, 20(a0)
    add a2, a2, a1
    addi a3, a3, -1
    sw a2, 20(a0)
    addi a0, a0, 4
    j loop
end:
    nop

```

aquest primer codi triga 66 cicles en completar-se.

```

-----

.data
mat1: .word 1,2,3,4,5
mat2: .word 6,7,8,9,0
mat3: .word 0,0,0,0,0
.text
main:
    la a0, mat1
    addi a4, zero, 5
loop:
    beqz a4, end
    lw a1, 0(a0)
    lw a2, 20(a0)
    add a2, a2, a1
    addi a4, a4, -1
    sw a2, 40(a0)
    addi a0, a0, 4
    j loop
end:
    nop

```

## Exercici 2: SUBROUTINES

### Pregunta 4:

**Quina instrucció fem servir en tots dos casos per assignar la posició inicial al registre SP?**

La instrucció es `.org` més un numero hexadecimal a la primera posició (inici del programa).

### Pregunta 5:

**Quina es la instrucció utilitzada per guardar el PC en la pila quan treballem amb subrutines? I per recuperar de nou el valor del PC?**

La instrucció que ens permet guardar un valor en pila es PUSH, i per recuperar de nou el valor de PC seria POP.

## **Tasca 2**

Pujeu el codi creat amb les subrutines

//mirar-mho després.

```
.define
    num 5
    clau 55h
.data 00h
    mat1: db 1,2,3,4,5
    mat2: db 6,7,8,9,0
    mat3: db 0,0,0,0,0
.org 100h
    LXI H, mat1
    LXI B, mat3
    MVI D, num
carga:
    MOV A,M
    STAX B
    INX H
    INX B
    DCR D
    JP carga
    LXI H, mat2
    LXI B, mat3
    MVI D,num
loop:
    MOV E, M
    LDAX B
    ADD E
    STAX B
    INX H
    INX B
    DCR D
    JP loop
    call sub_codificador

hlt

.org 150h
sub_codificador:
    PUSH H
    PUSH B
    PUSH D
    PUSH PSW
    MVI B, clau
    LXI H, mat3
```

```
        MVI D, num
codifica:
        MOV A, M
        XRI clau
        MOV M, A
        INX H
        DCR D
        JP codifica
        POP PSW
        POP D
        POP B
        POP H
RET
```

### **Conclusions:**

En aquesta pràctica hem après a treballar amb el simulador i8085, amb les seves instruccions, també hem continuat treballant amb el simulador riscV.