

## PROBLEMAS AC

2.18)

Suma Elemento:

pushl %ebp

movl %esp, %ebp

movl 8(%ebp), %eax # %eax = i

movl 16(%ebp), %ecx # %ecx = j

sal \$2, %ecx  $\leftarrow \text{leal } (\%eax, 8), \%edx \# \%edx = 8i$   $\# \%ecx = 4j$

sarl %eax, %edx # %edx =  $8i - i$

leal (%eax, %eax, 4), %eax # %eax =  $i + 4i$

movl mat2(%eax, %eax, 4), %eax # %eax =  $M[\text{mat2} + 4(i + 4i) + 4] = \text{mat2}[i][j]$

addl mat1(%eax, %edx, 4), %eax # %eax =  $\text{mat2}[i][j] + M[\text{mat1} + 4j + 4(8i - i)] =$   
 $= \text{mat2}[i][j] + \text{mat1}[i][j]$

movl %ebp, %esp

popl %ebp

ret

a)  $4iN = 4(i + 4i) = 4i + 16i = 20i \Rightarrow N = \frac{20i}{4i} = \boxed{5}$

$4iM = 4(8i - i) = 32i - 4i = 28i \Rightarrow M = \frac{28i}{4i} = \boxed{7}$

b) Estaciones = 13 instrucciones

c) pimerificas = 11

d) Acceso Mem = 9

e) CPE = 0,8 ms non datos y 0,5 ms non datos ciclos que tarda =  $\boxed{9 \cdot 1,25 + 4 \cdot 2 = 19,25}$   
 $CPE = \frac{1}{0,8} = 1,25 \quad CPE = \frac{1}{0,5} = 2$   
 $\frac{19,25}{1,25 + 2} = 23$

f)  $CPE = \frac{1}{0,3} = 1,67 \quad CPE = \frac{1}{0,6} = 1,66 \Rightarrow \boxed{9 \cdot 1,65 + 4 \cdot 1,66 = 17,89}$

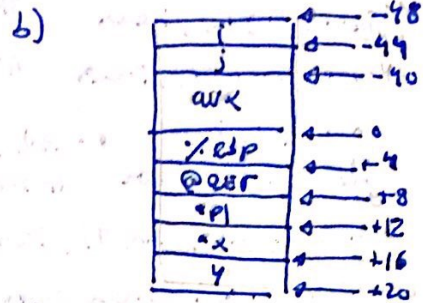
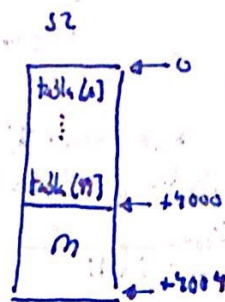
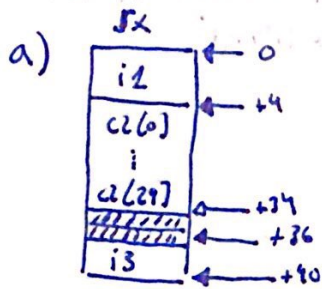
$\frac{19,25}{1,67 + 1,66} = \boxed{1,076}$

2.19)

```
typedef struct {
    int i1;
    char c2[30];
    int i3;
} s_x;
```

```
typedef struct {
    s_x tabla[100];
    int m;
} s_z;
```

```
int F(s_x *p1, int y);
int main(s_x *p1, int *x, int y) {
    int i, j;
    s_x aux;
```



c) `movl 12(%ebp), %eax # eax = x`  
`movl (%eax), %eax # eax = x`  
`addl -4(%ebp), %eax # eax = x + aux.i3`

1) `movb -12(%ebp), %al # aux(i3) = -12(%ebp)`  
`leal -40(%ebp), %ecx # ecx = &aux`  
`addl $4, %ecx # ecx = &aux + 4`  
`addl -12(%ebp), %eax # eax = &aux + 4 + i`  
`movb %al, (%ecx) # aux.c(i) = aux.c(i3)`

d) `movl 8(%ebp), %eax # ecx = j`  
`movl -44(%ebp), %ecx # ecx = j`  
`imull $40, %ecx # j = 40 * j`  
`addl %ecx, %eax # eax = j + 40`  
`movl 16(%ebp), %ecx # ecx = y`  
`pushl %ecx`  
`pushl %ecx`  
`call F`  
`movl %eax, -40(%ebp) # aux.i = ...`

e) `movl -44(%ebp), %ecx # ecx = j`  
`movl 16(%ebp), %ecx # ecx = y`  
`imull %ecx, %ecx # j = j * y`  
`movl %ecx, -48(%ebp) # i = j * y`