

ej 2

(a)

$(1010\ 1010)_2 - (1100\ 1100)_2 = (1101\ 1110)_2$

(b)

Un char es de 8 bits y con signo.

-34

(c)

Un unsigned char es de 8 bits, sin signo.

$(222)_{10}$

$(de)_{16}$

$(336)_8$

(d)

La instrucción "cbw" extiende con signo los 8 bits de al a 16 bits en ax

(e)

$(1111\ 1111\ 1101\ 1110)_2$

ej 4

(b)

.global main

main:

```
    movq $a, %rsi
    movl (%rsi), %eax    # eax = 0x646000
    movb 4(%rsi), %ah    # ah = 0x89
    movw 8(%rsi), %ax    # ax = 0xdeff
    movl 2(%rsi), %eax    # eax = 0x02890064
    movq 12(%rsi), %rax   # rax = 0x1122334455667788
    movl 14(%rsi), %eax   # eax = 0x33445566
    movl 16(%rsi), %eax   # eax = 0x11223344
    retq
```

ej 5

(b) Al ejecutar el código en C con el comando:

./a.out hola mundo

se imprimirán por pantalla los valores correspondientes a argc (3 en este caso) y a argv (puntero, dirección de memoria).

Para mirar el retorno de la ejecución, lo hacemos con el comando "echo \$?" y vemos que el programa retorna 3.

ej 6

1 megabyte son 1000000 bytes, es decir 2^{20} bytes.

(a) En el peor de los casos, una página de 1 mega es ocupada por un proceso (o lo restante de un proceso guardado en otra página) de 1 byte y se pierden $(2^{20}) - 1$ bytes.

En el mejor de los casos el proceso es de 1 mega, ocupa toda la página y no habría perdida.

Por lo tanto, en promedio, se va a perder media página por proceso, es decir 2^{10} bytes en este caso.

(b) Si se utilizan páginas de 1 kb, es decir 2^{10} bytes, también se va a perder en promedio media página por proceso, pero en este caso serían 2^{10} bytes.

(c) $(256 = 2^8)$

Al utilizar páginas de 256 bytes, es más factible que los procesos ocupen más de una página, al proveer poco espacio. Además, se utilizarían solo 8 bits para el offset, de los 64 que tiene una dirección de memoria, dejando 56 restantes para referenciar la página, lo que implica que las tablas de paginación serán más grandes y así ocuparían más espacio.

(d) Utilizar segmentación elimina la fragmentación interna, pero introduce la fragmentación externa, que es más costosa de resolver. Si ya tengo la memoria fragmentada por utilizar segmentación y quiero ejecutar 100 procesos largos, va a costar poder ubicar dichos procesos en memoria. Sin embargo, facilitaría la conversión de dirección virtual a dirección física.