

## Problemas AC

### 1.12

$$a) \frac{1}{125.000} + \frac{1}{1.000.000} + \frac{1}{200.000} + \frac{4}{1.000.000} + \frac{1}{500.000} + \frac{8}{100.000} = \frac{1}{10.000}$$

Por tanto 10.000 horas

$$b) \text{MTTB} = \text{MTTR} + \text{MTTF} = 10020 \text{ horas.}$$

$$c) \text{Availability} = \frac{\text{MTTF}}{\text{MTTF} + \text{MTTR}} = \frac{10000}{10020} = 0.998$$

### 2.1

Expresión	valor binario	valor hex	Expresión	valor binario	valor hex
$x \ \& \ y$	0000 0010	0x02	$x \ \&\& \ y$	1	0x01
$x \   \ y$	1111 0111	0xF7	$x \    \ y$	1	0x01
$\sim x \   \ \sim y$	1111 1101	0xFD	$!x \    \ !y$	1	0x01
$x \ \& \ !y$	0110 0100	0x64	$x \ \&\& \ \sim y$	1	0x01

### 2.2

x		x << 4		x >> 3 (lógico)		x >> 3 (aritmético)	
hex	binario	hex	binario	hex	binario	hex	binario
0xF0	1111 0000	0x00	0000 0000	0x1E	0001 1110	0xFE	1111 1110
0x0F	0000 1111	0xF0	1111 0000	0x01	0000 0001	0x01	0000 0001
0xCC	1100 1100	0xC0	1100 0000	0x19	0001 1001	0xF9	1111 1001
0x55	0101 0101	0x50	0101 0000	0x0A	0000 1010	0x0A	0000 1010
0x80	1000 0000	0x00	0000 0000	0x10	0001 0000	0xF0	1111 0000
0x02	0000 0010	0x20	0010 0000	0x00	0000 0000	0x00	0000 0000

## 2.5

Dada la siguiente definición de variables globales:

```
char A[256];  
char tabla[256];
```

Traducid a ensamblador de IA32 el siguiente fragmento de código:

```
for (i=0; i<256; i++)  
    A[i] = tabla[A[i]];
```

```
movl $A, %eax  
movl $tabla %ebx  
movl $0 %ecx
```

```
for:  cmpl $256, %ecx  
      jge final  
      movb (%eax, %ecx), %edx    A[i]  
      movb (%ebx, %edx), %esi    tabla[A[i]]  
  
      movb %esi, (%eax,%ecx)    A[i] = tabla[A[i]]  
      incl %ecx                 ++i  
      jmp for
```

```
final:
```

## 2.6

Dado el siguiente código escrito en C:

```
int *sorpresa (int i, int *x)
{
    if (i>-10 && i<10)
        *x = i;
    else
        x = &i;
    return x;
}
```

Teniendo en cuenta que los parámetros de la función son accesibles en las siguientes direcciones: `i` en `8(%ebp)`, dirección de `x` en `12(%ebp)`, traducid a ensamblador del IA32 el cuerpo de la subrutina.

sorpresa;  
    pushl %ebp

else:

fin:  
    popl %ebp  
    return