1.12: FIABILITAT, DISPONIBILITAT:

A)

$$\frac{1}{MTTF} = 125 \times 10^{+3^{-1}} + 1 \times 10^{6^{-1}} + 200 \times 10^{+3^{-1}} + \frac{4}{1 \times 10^{6}} + 300 \times 10^{-2^{-1}} + \frac{8}{100 \times 10^{-2}} = \frac{1}{10000} = 100000\,h$$

B) $MTBF = MTTF + MTTR = 10000 + 20 = 10020\,h$

C) $DISP = \dfrac{10000}{10020} = 0'998$

2.1: Operadors lògics: $x = 0x66$, $y = 0x93$

EXPRESSIÓ	BINARI	HEXA	EXPRESSIÓ	BINARI	HEXA
x & y	0000 0010	0x02	x && y	0000 0001	0x01
x | y	1111 0111	0xF7	x || y	0000 0001	0x01
¬x | ¬y	1111 1101	0xFD	!x || !y	0000 0000	0x00
x & !y	0000 0000	0x00	x && ~y	0000 0001	0x01

2.2: DESPLAÇAMENTS:

x HEXA	x BINARI	x << 4 HEXA	x << 4 BINARI	(LÒGIC) x >> 3 HEXA	(LÒGIC) x >> 3 BINARI	(ARITMÈTRIC) x >> 3 HEXA	(ARITMÈTRIC) x >> 3 BINARI
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: TRADUCCIÓ:

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

=>

```
        movl  $A, %esx
        movl  $tabla , %ecx
        movl  $0 , %ebx
for :   cmpl  $256 , %ebx
        jge   Fifor
        movsbl (%esx, %ebx), %edx
        ...
```

```
        ...movb (%ecx , %edx), %dl
        movb  %dl , (%esx, %ebx)
        incl  %ebx
        jmp   for
Fifor: ...
```

2.6: TRADUCCIÓ:

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

=>

```
sorpesA : pushl %ebp
          movl  %esp, %ebp
          movl  8(%ebp), %ebx
          movl  12 (%ebp), %eBx
          cmpl  $-10 , %ebx // i>-10?
          jle   else
          cmpl  $10, %ebx // i<10?
          jbe   else
          movl  %ebx , (%ecx)
          ...
```

```
          ... jmp Fi
    else : leal -8(%ebp), %ebx
           movl %ebx, 12 (%ebp)
    Fi:  movl 12 (%ebp), %eax
         popl %ebp
         ret
```