

COGNOMS:

NOM:

SIGNATURA:

1) Escriu un programa en *assembler* de SISA-F que es sincronitzi per enquesta i que compti el nombre de bits que valen 1 en un nombre de 16 bits emmagatzemat a memòria a través de la variable *w* (per exemple, $w=0xFOE0$). Mostreu el resultat de la compta per pantalla en la fila i columna que doni el resultat (per exemple, si el resultat es 5, en la fila i columna 5).

(2,5 puntos)

```
.include "macros.s"
.include "crt0.s"
```

```
.data
    w: .word 0xFOE0
.text
main:
```

2) Escriu un programa en *assembler* de SISA-F que es sincronitzi per interrupcions i que, a l'igual que el problema 1), compti el número de bits que son "1" de *w* (per exemple, $w=0xFOE0$), i els mostri en pantalla després de 10 s. A més, quan es premi una tecla, envieu a la Impressora el nombre de bits que valen "0". Escriu el "main" i les dues RSI. No cal fer una RSI per a la impressora. RSI del relox amb nom **clock**. RSI del teclat amb nom **teclat**.

(2,5 puntos)

```
.include "macros.s"
.include "crt0.s"
```

```
.data
    w: .word 0xFOE0
    ticks: .word 0 ; variable global para acumulador de ticks
    final: .byte 0 ; variable global indica que se alcanzó los 10 s
           .balign 2
    tecla: .byte 0 ; variable global que guarda la tecla pulsada
           .balign 2
.text
main:
```

3) Donat un processador d'arquitectura de 32 bits en un sistema amb memòria principal de 16 MB, amb una memòria cache directa de 16 KB, i de 4 words per bloc.

a) Donada una **direcció a memòria** originada des del processador ¿quants bits seran de byte offset, word offset, index i tag?

(1 punto)

b) Calcular el nombre total de bits emmagatzemats en aquesta memòria cache directa.

(1 punto)

c) Donada la següent seqüència de **@ de word** que genera el processador: 8208, 8206, 12303, 8211, 4098, 32769.

Omple les següents taules.

(3 puntos)

@ word	@bloque MP	Index	Hit/Miss
8208			
8206			
12303			
8211			
4098			
32769			
Tasa de Hits			

Memoria Cache:

INDEX	V	TAG	Word3	Word2	Word1	Word0
00.....000						
00.....001						
00.....010						
00.....011						
00.....100						
11.....111						