

## Examen práctico Fundamentos de Computadores – Gijón. Curso 2009-2010

Apellidos y nombre: \_\_\_\_\_ DNI: \_\_\_\_\_ Fecha: \_\_\_\_\_

### Instrucciones generales

- Desde el explorador de Windows, comprueba que esté vacío el contenido del directorio **C:\Temp** y, si no lo está, bórralo.
- Crea una carpeta con tus apellidos, nombre y DNI. Por ejemplo: **Díaz Díaz José – 10999888J**. Guarda en esa carpeta todos los archivos que hagas.
- Tras terminar el ejercicio, y cuando creas que todo está bien, desde el Explorador de Windows pulsa con el botón derecho sobre la carpeta y escoge la opción del programa PowerArchiver **“Compress To ‘TuCarpeta.zip’”** para comprimirla.
- Dale a **OK**, con lo que se generará un fichero **.zip** que deberás subir al FTP que el profesor que indicará. Una vez que escribas en el FTP no podrás cambiar, leer ni volver a escribir el archivo, así que asegúrate de que está correcto **antes** de copiarlo.
- Borra el directorio temporal antes de cerrar la sesión.

### Ejercicio

- Completar el esqueleto del programa incluido en el fichero **eval.ens**, **rellenando los huecos marcados con interrogantes**. Los interrogantes se corresponden con un conjunto de instrucciones. Lee atentamente los comentarios incluidos en el fichero para comprender el funcionamiento global del programa.

- ¿Cuál es la dirección de mapeo del registro de control del interfaz Pantalla? (Responder en hexadecimal).

Dirección de mapeo:

- Teniendo en cuenta la configuración de mapeo de los periféricos para este ejercicio Rellena la memoria RAM del computador elemental utilizando el menor número de dispositivos de memoria. Indica cuantos dispositivos utilizas de cada tamaño.

Módulos de 32K:

Módulos de 16K:

Módulos de 8K:

Módulos de 4K:

- Una vez hayas obtenido el fichero ejecutable, carga el programa en la CPU elemental, **configura y mapea los periféricos necesarios de forma adecuada, llenando el espacio de direcciones del computador con tantos módulos de memoria como sea posible**. Ejecuta el programa y comprueba que funciona correctamente. Guarda el estado de la CPU elemental con el nombre **eval.sim** en un momento cualquiera de la ejecución del programa.
- Antes de entregar el examen comprueba que en el archivo comprimido tienes al menos los ficheros **eval.ens**, **eval.eje** y **eval.sim**.

## Examen práctico Fundamentos de Computadores – Gijón. Curso 2009-2010

Apellidos y nombre: \_\_\_\_\_ DNI: \_\_\_\_\_ Fecha: \_\_\_\_\_

```
; Este programa utiliza dos periféricos
; pantalla y luces, este último con
; capacidad para generar interrupciones.
; El registro de datos del periférico
; luces está compuesto de un registro de
; 16 bits.
; Los 8 bits más significativos codifican
; el código de color con el que se debe
; escribir en la pantalla, mientras que
; los 8 bits menos significativos nos
; indican en que fila de la pantalla
; debemos escribir (filas numeradas desde 0
; hasta 7, en el caso de que la
; codificación sea igual o mayor que 8 no
; se hace nada).

; Cada vez que se genera una interrupción
; se deben realizar las siguientes
; acciones:
; 1.- Borrar la pantalla
; 2.- Utilizando la parte baja del
; registro luces identificar la fila de
; la pantalla donde tenemos que
; empezar a escribir desde la fila 0
; a la fila 7 y calcular la posición
; de comienzo. Recordar que si la
; fila es superior a 7 no se debe
; hacer nada.
; 3.- Escribir el contenido de las
; variables texto_par o texto_impar
; en la posición calculada de la
; pantalla dependiendo de si fila
; indicada (0..7) es par o impar. A
; la hora de escribir el texto
; utilizar el código de color
; indicado por la parte alta del
; registro de datos del periférico
; luces.
```

```
ORIGEN 0500h
INICIO primera
.PILA 19
```

```
.DATOS
  dir_pantalla VALOR 0AF00h
  dir_luces VALOR 0B000h
  texto_par VALOR "FILA PAR",0
  texto_impar VALOR "FILA IMPAR",0
```

```
.CODIGO
```

```
PROCEDIMIENTO rutina_luces
```

```
??????
```

```
; Borrar la pantalla al principio por si
; acaso tuviese algo escrito
CALL Borrar_Pantalla
```

```
??????
```

```
FINP
```

```
PROCEDIMIENTO Borrar_Pantalla
```

```
PUSH R0
PUSH R1
PUSH R2
```

```
; Dirección base de la pantalla
MOVH R0, BYTEALTO DIRECCION dir_pantalla
MOVL R0, BYTEBAJO DIRECCION dir_pantalla
MOV R0, [R0]
```

```
; Dirección registro control
MOVH R1, 0
MOVL R1, 120
ADD R0, R0, R1
```

```
; Leer registro de control
MOV R1, [R0]
```

```
; Modificar registro de control
MOVH R2, 0
MOVL R2, 1
OR R1, R1, R2
```

```
; Escribir registro de control
MOV [R0], R1
```

```
POP R2
POP R1
POP R0
RET
```

```
FINP
```

```
primera:
```

```
; Instalamos la rutina del periférico
; luces en el vector 5.
MOVL R1, BYTEBAJO DIRECCION rutina_luces
MOVH R1, BYTEALTO DIRECCION rutina_luces
MOVH R0, 0
MOVL R0, 5
MOV [R0], R1
STI
JMP -1
```

```
FIN
```