

## Examen práctico de Fundamentos de Computadores – Gijón. Curso 2008-2009

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: **Fernández Fernández José – 10999888J**
- Descarga el esqueleto de la práctica de <http://www.atc.uniovi.es/teleco/2fc/docs/exams/Practico-Septiembre-2009.zip>  
Contraseña del ZIP: septiembre2009
- 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 **Add to archive...** para comprimirla. Comprueba en la ventana que sale que en **Archive format** pone **Zip** y, si no es así, cámbialo.
- Dale a **OK**, con lo que se generará un fichero **.zip** que deberás entregar.
- Borra el directorio temporal antes de cerrar la sesión.

### Ejercicio

- Completar el esqueleto del programa incluido en el fichero **eval.ens**, de tal forma que la realice la función que se indica en el enunciado. Lee atentamente los comentarios incluidos en el fichero para comprender el funcionamiento global del programa, el cual incluye la implementación de una rutina de interrupción asociada al periférico **TECLADO** cuya dirección base es **B100h**.

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

Dirección de mapeo: **B278h**

- 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: **1**

Módulos de 16K: **1**

Módulos de 8K: **1**

Módulos de 4K: **1**

- ¿Cuál es el código ascii y el código de scan de la tecla correspondiente a la 'D' (Responder en hexadecimal).

Código Ascii: **44h**

Código Scan: **16h**

- Una vez hayas obtenido el fichero ejecutable, carga el programa en la CPU elemental, configura y mapea los periféricos necesarios de forma adecuada. 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 de Fundamentos de Computadores – Gijón. Curso 2008-2009

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

Fecha: \_\_\_\_\_

```
; Este programa utiliza 2 periféricos: Teclado
; y Pantalla. El programa principal
; lleva a cabo las siguientes acciones:
;
; 1- Instala la rutina de interrupción de
; teclado en memoria y habilita el
; tratamiento de interrupciones.
; 2- Ejecuta un bucle infinito, el cual se
; encargara de escribir el carácter indicado
; por la variable "carácter" en la columna
; de la pantalla indicada por la variable
; "columna".
;
; Inicialmente el carácter a escribir será
; la 'A', y la escritura se realizará sobre
; la columna 0.
;
; Cuando se genere una interrupción desde el
; periférico Teclado, la rutina de tratamiento
; asociada al periférico deberá:
; a) Si la tecla pulsada es un número se cambia
; el valor de la variable "columna" con el
; valor del número pulsado.
; b) Si la tecla pulsada es una letra MINÚSCULA
; se cambia el valor de la variable
; "carácter" con el código ASCII del carácter
; de la tecla pulsada en MAYUSCULAS.
; c) Para el resto de teclas no se realiza
; ninguna función.
;

ORIGEN 2600h
INICIO main
.PILA 20h
.DATOS
character VALOR 'A'
columna VALOR 0

.CODIGO

PROCEDIMIENTO rutina_teclado

FINP

main:
    MOVH R0, 0
    MOVL R0, 5
    MOVH R1, BYTEALTO DIRECCION rutina_teclado
    MOVL R1, BYTEBAJO DIRECCION rutina_teclado
    MOV [R0], R1
    STI

bucle:
    ; Cargamos en R0 la dirección base de la
    ; pantalla
    MOVH R0, 0B2h
    MOVL R0, 0h

    ; Cargamos en R1 el número de la columna sobre
    ; la que escribir
    MOVH R1, BYTEALTO DIRECCION columna
    MOVL R1, BYTEBAJO DIRECCION columna
    MOV R1, [R1]
    ADD R0, R0, R1

    ; Cargamos en R1 el carácter a sacar por
    ; pantalla
    MOVH R1, BYTEALTO DIRECCION caracter
    MOVL R1, BYTEBAJO DIRECCION caracter
    MOV R1, [R1]
    MOVH R1, 7

    ; Cargamos en R2 el número de filas de la
    ; pantalla
    MOVH R2, 0
    MOVL R2, 8

    ; Cargamos en R3 el numero de columnas de la
    ; pantalla
    MOVH R3, 0
    MOVL R3, 15

    escritura:
    ; Escribir en la columna el carácter indicado
    MOV [R0], R1
    ADD R0, R0, R3
    DEC R2
    BRNZ escritura

    JMP bucle

FIN
```