

Laboratorio de Sistemas Basados en Microprocesadores

Práctica 1: Modos de direccionamiento. Directivas y operadores.

La práctica consiste en la realización de programas simples que realicen unas tareas muy concretas con la finalidad de utilizar y comprender el uso de diferentes modos de direccionamiento utilizados por microprocesadores de la familia 8086, así como algunas directivas de reserva de memoria.

Ejercicio 1a. pract1a.asm (3ptos)

Escribir un programa que realice las siguientes operaciones:

- Cargar 9BH en AX
- Cargar B2H en BL
- Cargar 3412H en CX
- Cargar 01001001b en DS
- Cargar el contenido de CX en DX
- Cargar en BH el contenido de la posición de memoria 55235H y en BL en contenido de la posición 55236H
- Cargar en la posición de memoria 70008H el contenido de CH
- Cargar en AX el contenido de la dirección de memoria apuntada por SI
- Cargar en BX el contenido de la dirección de memoria que está 10 bytes por encima de la dirección apuntada por BP

Ejercicio 1b. practica1b.pdf (3 ptos)

Suponiendo que

DS= 03XXH, donde XX son las 2 primeras cifras de tu DNI

ES=0300H, BX=0212H y DI=1010H, determinar las direcciones de memoria a las cuales acceden cada una de las siguientes instrucciones:

a) MOV AL,DS:[0211H]

b) MOV AX,[BX]

c) MOV [DI],AL

Añade y completa el código de las siguientes instrucciones de forma que accedan a los mismos valores en las mismas posiciones que las instrucciones anteriores de cada punto respectivamente:

a1) MOV AL,ES:[????H]

b1) MOV SI, ???H
MOV AX,ES:[SI]

c1) MOV ES:[????H],AL

Indicar en todos los casos, la dirección real prevista (20 bits).

Entregar los resultados en una memoria: **practica1b.pdf**

Ejercicio 1c. pract1c.asm (4 ptos)

Escribir un programa en el que se definan las siguientes estructuras de datos:

- Reservar memoria para una variable, MARCADOR, de un byte de tamaño.
- Reservar memoria para una variable, BEBE, de dos bytes de tamaño, e inicializarla con el valor CAFEH
- Reservar 200 bytes para una tabla llamada TABLA200
- Guardar en memoria la cadena de texto “Este programa se cuelga siempre.”, de nombre ERRORTOTAL1, para agilizar la salida de mensajes en un programa de corrección automática de prácticas.

Con las estructuras de datos definidas anteriormente, realizar las siguientes acciones:

- Copiar el tercer carácter de la cadena ERRORTOTAL1 en la posición 63H de TABLA200
- Copiar el contenido de la variable BEBE a partir de la posición 23H de TABLA200
- Copiar el byte más significativo de BEBE a la variable MARCADOR

ENTREGA DE LA PRÁCTICA: Fecha y contenido

Ejercicios a y b : se deberá subir a Moodle un zip con los ficheros fuente pract1a.asm y la memoria practica1b.pdf incluyendo en comentarios una breve explicación de las modificaciones realizadas y los resultados obtenidos/verificados. Además, se deberá incluir un fichero makefile de compilación para el fichero fuente.

Ejercicio c: Se deberá subir a Moodle un fichero comprimido zip que contenga el fichero fuente comentado y el Makefile. El fichero fuente, deberá incluir el nombre del alumno y grupo en la cabecera.

El límite de fecha de subida de los ficheros para cada grupo es:

Ejercicios 1 y 2:

Grupos del Jueves: 11 de Marzo a las 20:15

Grupos del Viernes: 12 de Marzo a las 19:15h

Ejercicio 3:

Grupos del Jueves: 17 de Marzo a las 23:55

Grupos del Viernes: 18 de Marzo a las 23:55h

Para la resolución de los problemas a y c se propone utilizar la plantilla o estructura de programa indicada en la página siguiente, a falta de avanzar más en el temario y poder comprender todo lo que está escrito.

```
;*****
; SBM 2020. ESTRUCTURA BÁSICA DE UN PROGRAMA EN ENSAMBLADOR
;*****
; DEFINICION DEL SEGMENTO DE DATOS
DATOS SEGMENT
    ;-- rellenar con los datos solicitados
DATOS ENDS
;*****
; DEFINICION DEL SEGMENTO DE PILA
PILA SEGMENT STACK "STACK"
    DB 40H DUP (0) ;ejemplo de inicialización, 64 bytes inicializados a 0
PILA ENDS
;*****
; DEFINICION DEL SEGMENTO EXTRA
EXTRA SEGMENT
    RESULT DW 0,0 ;ejemplo de inicialización. 2 PALABRAS (4 BYTES)
EXTRA ENDS
;*****
; DEFINICION DEL SEGMENTO DE CODIGO
CODE SEGMENT
ASSUME CS: CODE, DS: DATOS, ES: EXTRA, SS: PILA
; COMIENZO DEL PROCEDIMIENTO PRINCIPAL
INICIO PROC
    ; INICIALIZA LOS REGISTROS DE SEGMENTO CON SU VALOR
    MOV AX, DATOS
    MOV DS, AX
    MOV AX, PILA
    MOV SS, AX
    MOV AX, EXTRA
    MOV ES, AX
    MOV SP, 64 ; CARGA EL PUNTERO DE PILA CON EL VALOR MAS ALTO
    ; FIN DE LAS INICIALIZACIONES
    ; COMIENZO DEL PROGRAMA
    ; -- rellenar con las instrucciones solicitadas
    ; FIN DEL PROGRAMA
    MOV AX, 4C00H
    INT 21H
INICIO ENDP
; FIN DEL SEGMENTO DE CODIGO
CODE ENDS
; FIN DEL PROGRAMA INDICANDO DONDE COMIENZA LA EJECUCION
END INICIO
```