



# Universidad Autónoma de Baja California

Facultad de Ciencias Químicas e Ingeniería

Organización de Computadoras y Lenguaje Ensamblador

Práctica 7: **“Estructuras de control de programa ”**

Chávez Padilla Ignacio  
1246720

Grupo: 552

Lara Camacho Evangelina

Viernes, 12 de Octubre de 2018

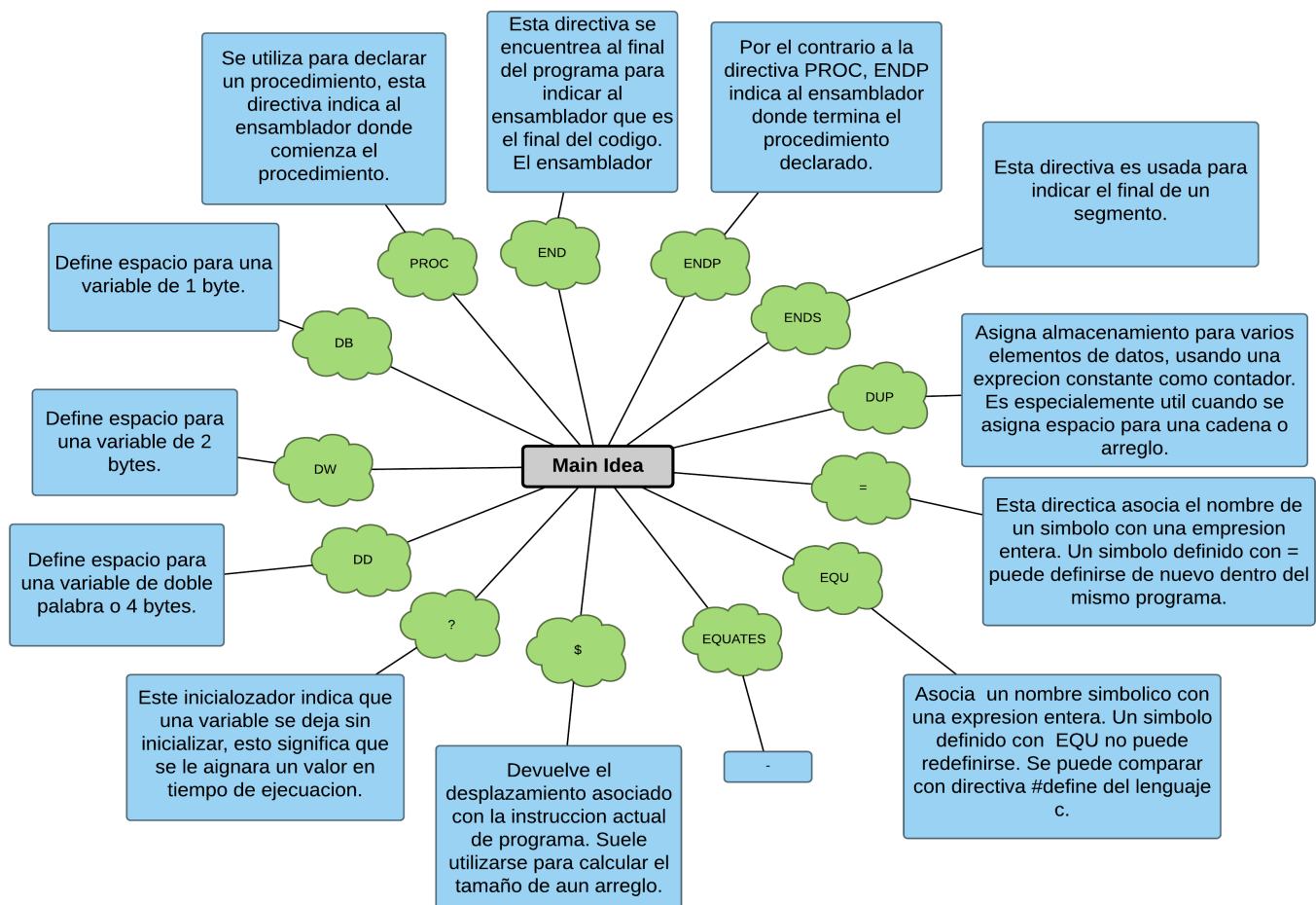
## Objetivo

El alumno se familiarizará con las instrucciones de control de programa del procesador 8088.

## Teoría

Mapa conceptual sobre las directivas:

- DB,DWyDD
- DUP, ?, \$
- Equates EQU y =
- PROC
- END, ENDP y ENDS



- Responda las preguntas:

¿Qué es un modelo de memoria? **Define la manera como un programa se almacena en la memoria del sistema.**

¿Cuáles son las diferencias entre los archivos .EXE y .COM?

**Una de las diferencias de es que los .COM no son reubicable ya que siempre inician en la dirección 0x100H mientras que los.EXE si son reubicable por que sus direcciones de memoria son relativa.**

**Otra diferencia es que el formato EXE no restringe el tamaño del ejecutable, en cambio el COM está restringido como máximo a ocupar un solo segmento de memoria. A diferencia de los archivos EXE, los COM tienen una estructura muy simple y almacenan en forma directa y lineal la imagen de memoria que será un programa. Solamente pueden almacenarse los programas que quepan en un solo segmento.**

Complete la información solicitada en la Tabla 1 sobre los modelos de memoria.

<b>Modelo de memoria</b>	<b>Descripción</b>
<b>Tiny</b>	Un solo segmento, contiene código y datos. Es utilizado por los programas con extensión .com
<b>Small</b>	Un segmento de código y otro segmento de datos. Todo el código y los datos son cercanos de manera predeterminada.
<b>Medium</b>	Varios segmentos de código y varios segmentos de datos.
<b>Compact</b>	Un segmento de código y varios segmentos de datos.

<b>Large</b>	Varios segmentos de código y de datos.
<b>Huge</b>	Es similar al modelo large, pero aquí los elementos individuales de datos pueden ser mas grandes que un solo segmento.

## Desarrollo

2. Hacer las modificaciones necesarias al archivo formato.asm para implementar en Lenguaje Ensamblador un ejemplo de cada una de las siguientes estructuras de programación pertenecientes a lenguajes de alto nivel. Realice cada ejemplo en un archivo diferente.

- a) IF-THEN
- b) IF-THEN-ELSE
- c) CASE OF
- d) FOR
- e) WHILE-DO
- f) DO-WHILE

2. Practique el uso de instrucciones condicionales y de ciclos al implementar en Lenguaje Ensamblador los siguientes programas.

a) Condicionales: Un cierto tipo de metal (ficticio) es clasificado de acuerdo a las condiciones en el Listado 1 y Tabla 2. Declare tres variables en memoria: *dureza*, *contenido\_carbon* y *maleabilidad* y asígneles valores particulares, despliegue en pantalla la clasificación del metal correspondiente a estos valores.

1. Su dureza debe ser mayor a 50.
2. Su contenido de carbón debe ser menor a 7.
3. Su maleabilidad debe ser mayor a 56.

Listado 1. Condiciones de un tipo de metal.

Clasificación	Características
Grado 10	Si las 3 condiciones se cumplen
Grado 9	Si las condiciones 1 y 2 se cumplen
Grado 8	Si las condiciones 2 y 3 se cumplen
Grado 7	Si las condiciones 1 y 3 se cumplen
Grado 6	Si solo una condición se cumple
Grado 5	Si ninguna condición se cumple

Tabla 2. Clasificación de un tipo de metal.

b) **Ciclos:** Escriba un programa que despliegue en pantalla el patrón de la Fig. 1.

```

1
22
333
4444
55555
666666
7777777
88888888
999999999

```

Figura 1. Patrón numérico.

## Conclusiones

Con esta practica se reforzaron lo temas previamente vistos en clase y se comprendió de una manera mas directa el uso de salto y ciclos dentro del lenguaje ensamblador. Tambien se aclaro la manera en la que un programa se almacena el memoria y como el modelo de memoria afecta al mismo. Por ultimo se identifico las direncias entre un programa .com y .exe.

## Dificultades en el desarrollo

No hubo dificultades mayores con el desarrollo de esta práctica.

## Referencias

TutorialsPoints

[https://www.tutorialspoint.com/assembly\\_programming/assembly\\_addressing\\_modes.html](https://www.tutorialspoint.com/assembly_programming/assembly_addressing_modes.html)