



TECNOLÓGICO  
NACIONAL DE MÉXICO



Tecnológico nacional de México  
Instituto tecnológico de Tijuana

Subdirección Académica

Departamento de sistemas y computación

SEMESTRE:  
Febrero-Junio 2021.

CARRERA:  
**Ingeniería en Sistemas Computacionales**

## **Modularización**

### **Unidad III**

MATERIA Y SERIE:  
Lenguajes de interfaz

YEPIZ LUVIANO JONATHAN 18212285

NOMBRE DEL MAESTRO (A):  
Luis Alfonso Gaxiola Vegaz

### 3 Modularización

Básicamente es crear varias rutinas (funciones o procedimientos) y así segmentar el código y hacerlo más entendible para el programador ya sea para actualizaciones o depuraciones

Este enfoque de segmentación o modularización es útil en dos casos :

Cuando existe un grupo de instrucciones o una tarea específica que deba ejecutarse en más de una ocasión.

Cuando un problema es complejo o extenso, la solución se “divide” o “segmenta” en módulos que ejecutan “partes” o tareas específicas. Dicha solución se organiza de forma similar a como lo hacen las empresas cuando se estructuran con base en las funciones para realizar sus actividades; en otras palabras, el trabajo se divide en partes que sean fácilmente manejables y que , lógicamente, puedan ser separadas ; así, cada una de estas partes se dedica a ejecutar una determinada tarea, lo que redundara en una mayor concentración , entendimiento y capacidad de solución a la hora de diseñar la lógica de cada una de estas. Dichas partes son módulos o segmentos del algoritmo, algunos de ellos son los módulos directivos o de control, que son los que se encargaran de distribuir el trabajo de los demás módulos. De esta manera se puede diseñar un organigrama que indique la estructura general de un algoritmo.

#### 3.1 Procedimientos

Es una colección de instrucciones que realizan una tarea específica. Dependiendo de su extensión y complejidad, un programa puede contener uno, algunos o inclusive cientos de procedimientos. Para emplear un procedimiento en un programa se requiere definirlo y llamarlo. Al definir un procedimiento escribimos las instrucciones que contiene. Al llamar al procedimiento transferimos el flujo al procedimiento para que sus instrucciones se ejecuten.

#### 3.2 MACROS

Una macro es un grupo de instrucciones repetitivas en un programa que se codifican solo una vez y pueden utilizarse cuantas veces sea necesario.

La principal diferencia entre una macro y un procedimiento es que en la macro se hace posible el paso de parámetros y en el procedimiento no.

Podemos decir entonces que un procedimiento es una extensión de un determinado programa, mientras que la macro es un módulo con funciones específicas que puede ser utilizado por diferentes programas.

### Sintaxis de una macro

Las partes que componen a una macro son:

- \* Declaración de la macro
- \* código de la macro
- \* Directiva de terminación de la macro

La declaración de la macro se lleva a cabo de la siguiente forma:

Nombre Macro MACRO [parametro1, parametro2...]

Aunque se tiene la funcionalidad de los parámetros es posible crear una macro que no los necesite.

La directiva de terminación de la macro es: ENDM

Un ejemplo de macro, para colocar el cursor en alguna posición determinada de la pantalla es:

Posición MACRO Fila, Columna

PUSH AX

PUSH BX

PUSH DX

MOV AH, 02H

MOV DH, Fila MOV DL, Columna

MOV BH, 0

INT 10H

POP DX

POP BX

POP AX

ENDM

Para utilizar una macro solo es necesario llamarla por su nombre, como si fuera una instrucción más del ensamblador, ya no son necesarias las directivas como en el caso de los procedimientos.

