



**TECNOLOGICO DE ESTUDIOS
SUPERIORES DE ECATEPEC**



**DIVISIÓN DE INGENIERÍA ELECTRÓNICA Y
TELEMÁTICA**

PRÁCTICAS DE LABORATORIO

PROGRAMACION I

REALIZÓ:

MIGUEL ANGEL ALVARADO CRUZ

SEPTIEMBRE 2009.

PRESENTACIÓN

El presente manual de prácticas fue realizado, para la asignatura de Programación 1, el cual, intenta proporcionar a los docentes y estudiantes un material de apoyo que facilite el proceso enseñanza-aprendizaje, a través del trabajo en el laboratorio, reforzando de esta manera, la teoría mostrada en el salón de clases.

Las prácticas de este manual, son presentadas para que el estudiante logre un aprendizaje significativo, debido a que están diseñadas de forma que el docente actúe como guía y el docente participe activamente, haciendo experimentos y al mismo tiempo aprendiendo por descubrimiento.

Dicho lo anterior, se justifica el brindar a los alumnos un manual que los encamine a la aplicación de los conceptos teóricos, permitiendo profundizar más en los casos prácticos.

ÍNDICE

	Pagina
PRÁCTICA 1. Comandos básicos.	4
PRÁCTICA 2. If-else	7
PRÁCTICA 3. If-else anidado.	10
PRÁCTICA 4. Switch-case.	14
PRÁCTICA 5. For definido e indefinido	18
PRÁCTICA 6. Arreglo unidimensional y bidimensional.	24

PRACTICA No 1

COMANDOS BASICOS.

I.OBJETIVO

Al terminar de la práctica el alumno reconocerá la estructura de un programa en C, y sus diferentes partes: cabeceras o programas de librería, declaraciones de funciones y procedimientos, declaraciones de variables globales o locales, programa principal, agrupación mediante las llaves {}. Además conocerá las funciones básicas printf y scanf, para imprimir, para leer, y su sintaxis.

II.MATERIAL Y EQUIPO:

1 EQUIPO DE CÓMPUTO

CPU Marca_____Mod_____N.S._____

Monitor Marca_____Mod_____N.S._____

Teclado Marca_____Mod_____N.S._____

1 JUEGO DE DISCOS DE LENGUAJE C

III.INTRODUCCION TEÓRICA

Todo programa en C consta de una o más funciones, una de las cuales se llama main .el programa siempre comenzara por la ejecución de la función main, cada función deba contener:

- 1) Una cabecera de la función que consta del nombre de la función, seguido de una lista adicional de argumentos encerrados en el paréntesis.
- 2) Una lista de declaración de argumentos, si se incluyen en la cabecera.
- 3) Una sentencia compuesta que contiene el resto de la función. Cada sentencia compuesta se encierra con un par de llaves {}, y las llaves pueden contener combinaciones de sentencia elementales (denominadas sentencias de expresión) y otras sentencias compuestas. Cada sentencia de expresión debe terminar con un punto y coma (;)
- 4) Los comentarios pueden ser aparecer en cualquier lugar del programa siempre y cuando estén situados entre los delimitadores /* y*/.
Cabe hacer notar que el programa en si debe escribirse con minúsculas sin acentuar, excepto lo que este marcado como comentario, dentro de las llaves se recomienda el uso de tabuladores y espacios para lograr una mejor comprensión del programa.

MATERIAL Y EQUIPO

1 Equipo de cómputo

	No MAQUINA	FUNCIONA		OBSERVACIONES
		SI	NO	
CPU				
MONITOR				
TECLADO				
MAUSE				

1 juego de disco del lenguaje C

IV. DESARROLLO

EL ALUMNPO SEGUIRA LA SECUENCIA INDICADA, DESCRIBIENDO EN CADA INCISO LOS PASOS QUE SIGA:

- 1.-Trabaje con la computadora busque las ayudas relacionadas a printf y scanf.
- 2.-Teclee el siguiente programa:

```
include<stdio.h>
#include<conio.h>
Char c,cadena[200];

main()
{
clrscr();
gotoxy(15,5);
printf("Practica #2:");
printf("\n\n Presione<Enter>,para continuar...\n");
scanf("%c",&c);
printf("\aTeclee una cadena de caracteres:");
scanf("%s",&cadena);
printf("\n\n\n Los caracteres leídos son:%s",cadena ,"\nFin...\n");
printf("\n\n\n\n");
}
```

V.CUESTIONARIO (TRABAJO COMPLEMENTARIO)

1. Dado el programa anterior, además de editarlo, debe guardarlo, copiarlo y ejecutarlo. Describa:

a) los errores que obtuvo (en caso de haberlos)

b) el desplegado obtenido

2. Una vez que haya corrido quite las líneas `#include` del programa y vuelva a correrlo, anote lo que observa.

3. Investigue los diferentes tipos de variables que se manejan en lenguaje C.

4. De la definición de sintaxis de forma general y de sintaxis dentro de los lenguajes de programación.

5. Investigue la sintaxis de:

a) `printf`

b) `scanf`

VI.CONCLUSIONES Y OBSERVACIONES

PRACTICA No 2

IF ELSE.

I.OBJETIVO

Al término de la práctica el alumno reconocerá los diferentes tipos de variables y sabrá su manejo. Podrá distinguir entre diferentes formas de obtener datos y de presentarlos en pantalla. Además aprenderá sobre los comandos para limpiar pantalla (**clrscr**) y direccionar en diferentes posiciones el despliegue (**gotoxy**).

Y aprenderá sobre el manejo de **include**.

II.MATERIAL Y EQUIPO:

1 EQUIPO DE CÓMPUTO

CPU Marca_____Mod_____N.S._____

Monitor Marca_____Mod_____N.S._____

Teclado Marca_____Mod_____N.S._____

1 JUEGO DE DISCOS DE LENGUAJE C

III.INTRODUCCION TEÓRICA

Ya hemos visto que el lenguaje C va acompañado de una serie de funciones de biblioteca que incluye un cierto número de funciones de entrada/salida. Algunas de ellas son: **getchar, putchar, gets, puts, scanf y printf** (estas dos últimas son las que se utilizan más comúnmente). Estas funciones permiten la transferencia de información entre la computadora y los dispositivos de entrada/salida estándar.

a) Las funciones **getchar** y **putchar** permiten la transferencia hacia dentro y hacia afuera de caracteres sueltos.

b) Las funciones **printf** y **scanf** permiten la transferencia de caracteres sueltos, valores numéricos y cadenas de caracteres.

c) Las funciones **gets** y **puts** permiten la entrada y salida de cadenas de caracteres.

Dentro del lenguaje C podemos modificar el lugar donde queremos presentar el desplegado de pantalla (con **gotoxy**), limpiar la pantalla **clrscr** y presentar gráficos en pantalla ayudados de la librería **graphics**.

MATERIAL Y EQUIPO

1 Equipo de cómputo

	No MAQUINA	FUNCIONA		OBSERVACIONES
		SI	NO	
CPU				
MONITOR				
TECLADO				
MAUSE				

1 juego de disco del lenguaje C

IV.DESARROLLO

```
include<stdio.h>
#include<conio.h>

main();
{
int edad;
char sexo,nombre[60],pausa basura;
float estatura;

Clrscr();
gotoxy(1,1); printf(Practica #3:");
gotoxy(1,5); printf("Teclea tu nombre(%d caracteres):",60):gets(nombre);
gotoxy(10,7); printf("Teclea tu sexo");scanf("%c",&sexo);
gotoxy(10,9); printf("Teclea tu edad :");scanf("%d%c",&edad,basura);
gotoxy(10,11); printf("Teclea tu estatura");

gotoxy(20,20); printf("Presione<Enter>,para continuar...");
scanf("%c",&pausa);

Clrscr();
gotoxy(1,13);
scanf("%f%c",&pausa);
printf("\aTu nombre es:%s\nTu sexo es:%c\n",nombre,sexo);
```

```
printf("Tu edad es:%dTu estatura es:%4.2f\n".edad,estaturasexo);  
}
```

V.CUESTIONARIO (TRABAJO COMPLEMENTARIO)

1. Dado el programa anterior, además de editarlo, debe guardarlo, copiarlo y ejecutarlo. Describa:
 - a) los errores que obtuvo (en caso de haberlos)
 - b) el desplegado obtenido
2. Una vez que haya corrido quite las líneas #include del programa y vuelva a correrlo, anote lo que observa.
3. Investigue la sintaxis de:
 - a) gets
 - b) puts

VI.CONCLUSIONES Y OBSERVACIONES

PRACTICA No 3

IF ELSE ANIDADO.

I.OBJETIVO

Al término de la práctica el alumno sabrá utilizar diferentes tipos de variables en un programa y sobre su uso en **expresiones simples** (que no requieren reglas jerárquicas). Conocerá los términos operadores y sus categorías (operadores monarios, aritméticos, relacionales y lógicos, de asignación y el operador condicional).

II.MATERIAL Y EQUIPO:

1 EQUIPO DE CÓMPUTO

CPU Marca _____ Mod _____ N.S. _____

Monitor Marca _____ Mod _____ N.S. _____

Teclado Marca _____ Mod _____ N.S. _____

1 JUEGO DE DISCOS DE LENGUAJE C

III.INTRODUCCION TEÓRICA

En el lenguaje C todo programa consta de una o más funciones y recordemos que main es una función (además de ser el programa principal). Estas funciones a su vez, junto con variables, constantes, elementos de arreglos (como el caso de cadena [200]), se pueden unir con varios operadores para formar expresiones .

En este lenguaje, existe un gran número de operadores que se pueden agrupar en diferentes categorías

Categoría de operador	Operadores	Asociatividad
Operadores monarios	-+!!!sizeof (tipo)	D I
Multiplicación, división y resto aritmético	*/%	I D
Suma y sustracción	+ -	I D
Operadores	<<=>=	I D
Operadores de	=!=	I D
Y lógica	&&	I D
O lógica		I D
Operador condicional	?:	D I
Operador de asignación	=+=-*=/=%=	D I

Dentro de las sentencias compuestas se pueden declarar nombres simbólicos para representar cierto tipo de información dentro de esa parte del programa, a esto le conoceremos como variables locales, y por otro lado hablaremos de variables globales cuando hacemos referencia a esos nombres simbólicos en diferentes funciones y representen lo mismo.

Existen dos sentencias de expresión importantes, ya que una de ellas nos permite tener los datos que vamos a manejar en el programa (**scanf**) y la otra para pedir los datos y/o desplegarlos y desplegar resultados (**printf**)

IV. DESARROLLO

EL ALUMNO SEGUIRA LA SECUENCIA INDICADA, DESCUBRIENDO EN CADA INCISO LOS PASOS QUE SIGA:

1. Trabaje con la computadora busque las ayudas relaciones a float e int.
2. Teclee el siguiente programa:

```
include<stdio.h>
#include<conio.h>
main();
{
Int a,b,c;
char enter;
float suma,multiplicacion;

Clrscr();
gotoxy(1,1); printf("Practica #5:(Suma multiplicación de 3 numeros)" );
gotoxy(10,4); printf("tecleee dato uno:"); scanf("%d%c",&a,enter);
gotoxy(10,4); printf("tecleee dato uno:"); scanf("%d%c",&a,enter);
gotoxy(10,6); printf("tecleee dato tres:"); scanf("%d%c",&c,enter);

suma =a+b+c;
multiplicación =a*b*c;
gotoxy(20,8); printf("Presione<Enter>,para continuar...");
scanf("%c",&enter);

gotoxy(1,10);
printf("La suma es:%5.0f\nLa multiplicación
es:8.2f\n",suma,multiplicación);
}
```

MATERIAL Y EQUIPO

1 Equipo de cómputo

	No MAQUINA	FUNCIONA		OBSERVACIONES
		SI	NO	
CPU				
MONITOR				
TECLADO				
MAUSE				

1 juego de disco del lenguaje C

V. CUESTIONARIO (TRABAJO COMPLEMENTARIO)

1. Dado el programa anterior, además de editarlo, debe guardarlo, copiarlo y ejecutarlo. Desc riba:

a) los errores que obtuvo (en caso de haberlos)

b) el desplegado obtenido

2. ¿Cual sería el valor resultante de la siguiente expresión?

$W=2*((i\%5)*(4+(j-3)/(k+2)));$ donde $i=8, j=15$ y $k=4$

3. Dadas las siguientes expresiones evalúelas y diga el valor resultante:

a) $i+f$

donde $i=7$

b) $i+c$

$f=5.5$

c) $i+c''$

c variable de tipo carácter= w

d) $(i+c)-(2*f/5)$

Nota: La w se codifica como 11i9n (en decimal) y el "(cero) como 48 en **ASCII**

Dado que se mezclan en las operaciones diferentes tipos de variables, el valor resultante se expresara en el tipo de variable que sea más consistente con los tipos de datos de los operandos y con la mayor precisión posible.

Así el tipo de variable será, para a) de doble precisión, b) entero, c) entero y d) doble precisión.

VI. CONCLUSIONES Y OBSERVACIONES

PRACTICA No 4

SWITCH CASE.

I.OBJETIVO

Al término de la práctica el alumno sabrá utilizar diferentes tipos de variables en **expresiones complejas**, respetando el orden de precedencia o jerárquicas.

II.MATERIAL Y EQUIPO:

1 EQUIPO DE CÓMPUTO

CPU Marca _____ Mod _____ N.S. _____
Monitor Marca _____ Mod _____ N.S. _____
Teclado Marca _____ Mod _____ N.S. _____

1 JUEGO DE DISCOS DE LENGUAJE C

III.INTRODUCCION TEÓRICA

En la práctica pasada vimos los diferentes operadores, su orden de precedencia y asociatividad; en la práctica veremos un ejemplo de ese orden jerárquico, recordemos la tabla utilizada:

n±

el lenguaje C todo programa consta de una o más funciones y recordemos que main es una función (además de ser el programa principal).Estas funciones a su vez, junto con variables, constantes, elementos de arreglos(como el caso de cadena [200]),se pueden unir con varios operadores para formar expresiones .

En este lenguaje, existe un gran número de operadores que se pueden agrupar en diferentes categorías

Categoría de operador	Operadores	Asociatividad
Operadores monarios	-+!!!sizeof (tipo)	D I
Multiplicación, división y resto aritmético	*/%	I D
Suma y sustracción	+ -	I D
Operadores	<<=>=	I D
Operadores de	=!=	I D
Y lógica	&&	I D
O lógica		I D
Operador condicional	?:	D I
Operador de asignación	=+=-*=/=%=	D I

Dentro de las sentencias compuestas se pueden declarar nombres simbólicos para representar cierto tipo de información dentro de esa parte del programa , a esto le conoceremos como variables locales, y por otro lado hablaremos de variables globales cuando hacemos referencia a esos nombres simbólicos en diferentes funciones y representen lo mismo. Existen dos sentencias de expresión importantes, ya que una de ellas nos permite tener los datos que vamos a manejar en el programa (**scanf**) y la otra para pedir los datos y/o desplegarlos y desplegar resultados (**printf**).

IV.DESARROLLO

EL ALUMNO SEGUIRA LA SECUENCIA INDICADA, DESCUBRIENDO EN CADA INCISO LOS PASOS QUE SIGA:

1. Trabaje con la computadora busque las ayudas relaciones a float e int.
2. Teclee el siguiente programa:

```
include<stdio.h>
#include<conio.h>
main();
{
Int a,b,c;
char enter;
float suma,multiplicacion;

Clrscr();
gotoxy(1,1); printf("Practica #5:(Suma multiplicación de 3 numeros)" );
gotoxy(10,4); printf("teclea dato uno:"); scanf("%d%c",&a,enter);
gotoxy(10,4); printf("teclea dato dos:"); scanf("%d%c",&a,enter);
gotoxy(10,6); printf("teclea dato tres:"); scanf("%d%c",&c,enter);

suma =a+b+c;
multiplicación =a*b*c;
gotoxy(20,8); printf("Presione<Enter>,para continuar...");
scanf("%c",&enter);

gotoxy(1,10);
printf("La suma es:%5.0f\nLa multiplicación
es:8.2f\n",suma,multiplicación);
}
```

MATERIAL Y EQUIPO

1 Equipo de cómputo

	No MAQUINA	FUNCIONA		OBSERVACIONES
		SI	NO	
CPU				
MONITOR				
TECLADO				
MAUSE				

1 juego de disco del lenguaje C

V. CUESTIONARION (TRABAJO COMPLEMENTARIO)

1. Dado el programa anterior, además de editarlo, debe guardarlo, copiarlo y ejecutarlo. Describa:

a) los errores que obtuvo (en caso de haberlos)

b) el desplegado obtenido

2. ¿Cual sería el valor resultante de la siguiente expresión?

$W=2*((i\%5)*(4+(j-3)/(k+2)));$ donde $i=8, j=15$ y $k=4$

3. Dadas las siguientes expresiones evalúelas y diga el valor resultante:

a) $i+f$

donde $i=7$

b) $i+c$

$f=5.5$

c) $i+c''$

c variable de tipo carácter= w

d)(i+c)-(2*f/5)

Nota: La w se codifica como 11i9n (en decimal) y el " (cero) como 48 en **ASCII**

Dado que se mezclan en las operaciones diferentes tipos de variables, el valor resultante se expresara en el tipo de variable que sea más consistente con los tipos de datos de los operandos y con la mayor precisión posible.

Así el tipo de variable será, para a) de doble precisión, b) entero, c) entero y d) doble precisión.

VI.CONCLUSIONES Y OBSERVACIONES

PRACTICA 5

FOR DEFINIDO E INDEFINIDO

I.OBJETIVO

Al termino de la practica el alumno conocerá una estructura anidada, la estructura de control if_else y el uso de comparaciones compuestas.

II.MATERIAL Y EQUIPO

a) Equipo de computo

CPU	Marca_____	Mod_____	N.S._____
Monitor	Marca_____	Mod_____	N.S_____
Teclado	Marca_____	Mod_____	N.S_____

III. INTRODUCCION TEORICA

En los programas que se han realizado hasta ahora, cada instrucción que aparecía en el programa se ejecutaba solo una vez, pero muchos programas requieren que algún grupo de instrucciones se ejecuten repetidamente, hasta que se satisfaga una condición lógica.

A estas estructuras se les denomina como bucles. A veces no se sabe con exactitud, por adelantado, el número de veces que se deben de repetir las instrucciones, si no que se repite la ejecución hasta que alguna condición lógica se hace cierta. Otras veces un grupo de instrucciones consecutivas se repite un cierto número especificado de veces, y esto es otra forma de bucle.

Muchos programas requieren que se realice una comprobación lógica (o test) en algún punto concreto dentro de ellos. Se realizara a continuación alguna acción que dependerá del resultado de la comprobación lógica. Esto se conoce como ejecución condicional. Y finalmente existe una clase especial de ejecución condicional en la que un grupo de sentencias es seleccionado entre varios grupos disponibles. A esto se le llama a veces selección.

Todas estas operaciones se pueden realizar utilizando diversas sentencias de control incluidas en C y que veremos en las siguientes prácticas.

Antes de hablar de estas sentencias de control, revisemos algunos conceptos necesarios.

Primero vamos a formar expresiones lógicas que bien pueden ser ciertas o falsas, para lo cual utilizaremos los cuatro operadores relacionales y los dos operadores de igualdad.

(Anótalos en las siguientes líneas con nombre y símbolo)

a) _____

b) _____

c) _____

d) _____

e) _____

f) _____

Además de seis operadores, C tiene dos conectivas lógicas(también conocidos como operadores lógicos), &&(Y) y ||(O), y el operador monario de negación(!). Las conectivas lógicas se usan para combinar expresiones lógicas, formándose así expresiones complejas. El operador de negación se utiliza para invertir el valor de una expresión lógica (por ejemplo de verdadero a falso).

Otro operador que también se utiliza es el operador condicional (?:)que también hace uso de una expresión cierta o falsa, este operador es equivalente a la estructura simple if-Then else que veremos más adelante.

Es conveniente recordar que en C existen tres tipos de sentencias:

- a) De expresión.- consiste en una expresión seguida de punto y coma.
- b) Sentencias compuestas.- consiste en una secuencia de dos o mas sentencias consecutivas encerradas entre llaves { y }, las sentencias dentro de las llaves pueden ser sentencias de expresión, otras sentencias compuestas o sentencias de control.
- c) Sentencias de control.- que como ya dijimos incluyen una prueba o comprobación para seguir alguno de los caminos, la mayoría de las sentencias de control contienen sentencias de expresión o sentencias compuestas (o ambas).

IV. DESARROLLO

EL ALUMNO SEGUIRA LA SECUENCIA INDICADA, DESCRIBIENDO EN CADA INCISO LOS PASOS QUE SIGA:

- 1.- Trabaje con la computadora. Vea las ayudas para if_else.
- 2.- Teclee el siguiente programa

```
#include <stdio.h>
```

```
#include <math.h>
```

```
#include <conio.h>
```

```
float funcionf (float *fx); /*Despues de declarar las librerias, se  
declaranlas funciones junto
```

con el argumento de cada función, que en este caso van hacer los
apuntadores

hacia

donde se dirige el contenido de las variables*/

```
float funciong (float *gx);
float funcionz (float *zx);
void main (void)
{

float k,fun1,fun2,fun3; /*declaramos variables globales*/

printf("\tLos datos obtenidos son los siguientes:\t\n");
printf("\t f(x)          g(x)          z(x)\t\n");

for(k=0; k<3.5;k= k + 0.2)
{

fun1 = funcionf(&k); /*En esta parte del programa se estan llamando a
las funciones en el cual el argumento de cada una de ellas
es el contenido de la dirección
de la variable donde estamos introduciendo la información,
el retorno de los datos de cada función se almacenara en
la variable correspondiente*/

fun2 = funciong(&k);

fun3 = funcionz(&k);
```

```
printf("fun1(%.3f)= %.3f \t",k,fun1); /*En esta parte se imprimen los resultados finales con tres decimales*/
```

```
printf("fun2(%.3f)= %.3f \t",k,fun2);
```

```
printf("fun3(%.3f)= %.3f \t",k,fun3);
```

```
printf("\n");
```

```
}
```

```
getch ();
```

```
}
```

```
/*A partir de este momento se empiezan a crear cada una de las funciones antes declaradas*/
```

```
float funcionf (float *fx)
```

```
{
```

```
float a;
```

```
a=3*(exp(*fx))- 2 * *fx;
```

```
return a; /*aquí regresaremos el resultado de la ecuación*/
```

```
}
```

```
float funciong (float *gx)
```

```
{
```

```
float b;
```

```
b= -*gx * sin (*gx) + 1.5;
```

```
return b; /*aquí regresaremos el resultado de la ecuación*/
```

```
}
```

```
float funcionz (float *zx)
{
float c;
c>(*zx)*(*zx)*(*zx) -2 * (*zx) + 1;
return c; /*aquí regresaremos el resultado de la ecuación*/
}

/*fin del programa*/
```

MATERIAL Y EQUIPO

Equipo de computo-Observaciones.

CPU:

MONITOR:

TECLADO:

V. CUESTIONARIO (TRABAJO COMPLEMENTARIO)

1. Dado el programa anterior, además de editarlo, debe guardarlo, compilarlo y ejecutarlo. Describa:

- a) los errores que obtuvo (en caso de haberlos).
- b) el desplegado obtenido.

2. Marque en el programa donde comienza y donde terminan las estructuras de control utilizadas.

3. ¿Cuál es la funcionalidad de la estructura if_else?

VI. CONCLUSIONES Y OBSERVACIONES

PRACTICA 6

ARREGLO UNIDIMENSIONAL Y BIDIMENSIONAL

I.OBJETIVO

Al término de la práctica el alumno conocerá las estructuras de control if else y switch. Además aprenderá el manejo de los comandos de pantalla para cambiar el color del fondo (textbackground) y el color del texto (textcolor).

II.MATERIAL Y EQUIPO

a) Equipo de computo

CPU	Marca_____	Mod_____	N.S._____
Monitor	Marca_____	Mod_____	N.S._____
Teclado	Marca_____	Mod_____	N.S._____

III. INTRODUCCION TEORICA

En la presente práctica hablaremos específicamente de las sentencias de control switch. Esta sentencia se utiliza para seleccionar un grupo de sentencias entre varios grupos disponibles. La selección se basa en el valor de una expresión que se incluye en la sentencia switch.

La forma general de la sentencia switch es:

switch(expresión)sentencia

En donde expresión debe ir entre paréntesis, como se indica y devuelve el número entero, expresión también puede ser de tipo carácter (de tipo char), ya que los caracteres individuales tienen valores enteros.

La sentencia incluida es generalmente una sentencia compuesta que especifica opciones posibles a seguir. Cada opción se expresa como un grupo de una o más sentencias individuales dentro de la sentencia global incluida.

Para cada opción, la primera sentencia dentro del grupo debe ser precedida por una o más etiquetas case.

Las etiquetas case identifican los diferentes grupos de sentencias (las distintas opciones) y distinguen unas de otras. Las etiquetas case deben ser, por tanto, únicas dentro de una sentencia switch dada.

Cada grupo de sentencias se escribe de forma general:

Case expresión:

Sentencia1

Sentencia2

.

.

Sentencia n

O en el caso de varias etiquetas case:

case expresión 1:

case expresión 2:

.

.

Case expresión m;

Sentencia 1

Sentencia2

.

.

Sentencia n

En donde expresión 1, expresión 2,.....expresión m representan expresiones constantes de valores enteros. Normalmente, cada una de estas expresiones se escribirá como constante entera o como constante de carácter. Cada sentencia individual que sigue a las etiquetas case puede ser simple o compuesta.

IV. DESARROLLO

EL ALUMNO SEGUIRA LA SECUENCIA INDICADA, DESCRIBIENDO EN CADA INCISO LOS PASOS QUE SIGA:

1.- Trabaje con la computadora. Vea las ayudas para if_else.

2.- Teclee el siguiente programa

```

#include <conio.h>

int factorial (int dato);

main ()
{
    int a,valor;

    clrscr ();

    printf("\t\tPROGRAMA QUE CALCULA EL FACTORIAL DE UN NUMERO\t\t");
    printf("\n\n");
    printf("Introduce el numero al que le deseas calcular el factorial\n\n");
    scanf("%d",&valor);
    a=factorial (valor);

    printf("\n\nEl valor factorial de %d es: %d",valor,a);
    getch ();
    return 0;
}

int factorial (int dato)
{
    int i,f;
    printf ("\nVariable acumulativa dato: %d\n",dato);
    if (dato == 0)
    {
        return 1;
    }
    else

```

```
{  
    f=dato;  
    for (i=dato-1 ; i>=2 ; i--)  
    {  
        f=f*i;  
        printf ("Variable acumulativa dato: %d\n",f);  
    }  
    return f;  
}  
}
```

MATERIAL Y EQUIPO

Equipo de computo-Observaciones.

CPU:

MONITOR:

TECLADO:

V. CUESTIONARIO (TRABAJO COMPLEMENTARIO)

1. Dado el programa anterior, además de editarlo, debe guardarlo, compilarlo y ejecutarlo. Describa:
 - a) los errores que obtuvo (en caso de haberlos).
 - b) el desplegado obtenido.
2. Marque en el programa donde comienza y donde termina el switch.
3. ¿Cuál es la funcionalidad de la estructura switch?

VI. CONCLUSIONES Y OBSERVACIONES