



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: M.I. Marco Antonio Martínez Quintana

Asignatura: Fundamentos de Programación

Grupo: 3

No de Práctica(s): #08

Integrante(s): Cuevas Antunez Samantha

*No. de Equipo de
cómputo empleado:* No aplica

No. de Lista o Brigada: 12

Semestre: Primer semestre

Fecha de entrega: 23/11/2020

Observaciones:

CALIFICACIÓN: _____



Estructuras de selección



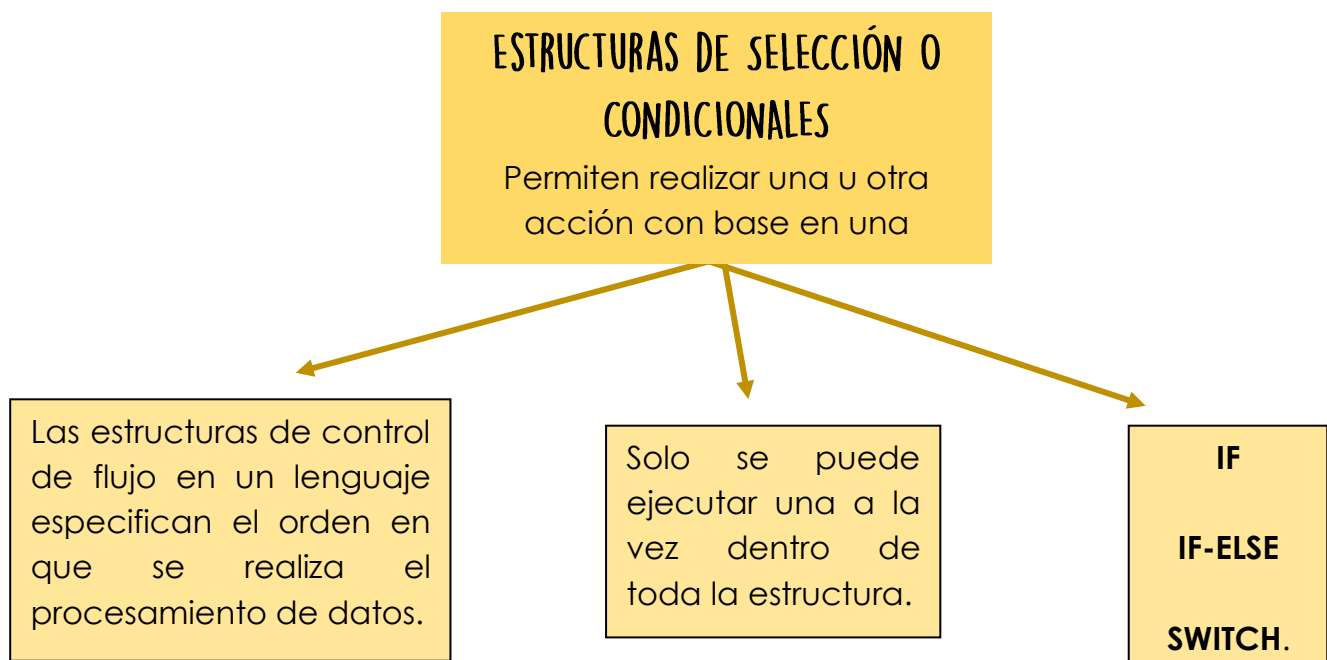
OBJETIVOS

Elaborar programas en lenguaje C que incluyan las estructuras de selección if, if-else, switch y ternaria (o condicional) para la resolución de problemas básicos.

ACTIVIDADES

- Elaborar expresiones lógicas/condicionales utilizadas en las estructuras de selección y realizar su evaluación.
- Elaborar un programa en lenguaje C para cada estructura de selección.

INTRODUCCIÓN



ESCTRUC-TURA	SINTAXIS	
IF	<pre>if (expresión lógica) { // bloque de código a ejecutar }</pre>	<p>Se evalúa la expresión lógica y, si se cumple (si la condición es verdadera), se ejecutan las instrucciones del bloque que se encuentra entre las llaves de la estructura. Si no se cumple la condición, se continúa con el flujo normal del programa.</p>
IF-ELSE	<pre>If (condición) { //sentencias else { //sentencias }</pre>	<p>Evalúa la expresión lógica y si la condición es verdadera se ejecutan las instrucciones del bloque que se encuentra entre las primeras llaves, si la condición es falsa se ejecuta el bloque de código que está entre las llaves después de la palabra reservada 'else'. Al final de que se ejecute uno u otro código, se continúa con el flujo normal del programa.</p>
SWITCH	<pre>Swicth (variable) { case valor_1: //sentencias break; case valor_2: //sentencias break; default: //sentencias }</pre>	<p>Evalúa la variable que se encuentra entre paréntesis después de la palabra reservada switch y la compara con los valores constantes que posee cada caso (case).</p> <p>Los tipos de datos que puede evaluar esta estructura son enteros, caracteres y <u>enumeraciones</u>. Al final de cada caso se ejecuta la instrucción break, si se omite esta palabra reservada se ejecutaría el siguiente caso, es decir, se utiliza para indicar que el bloque de código a ejecutar ya terminó y poder así salir de la estructura.</p>

		Si la opción a evaluar no coincide dentro de algún caso, entonces se ejecuta el bloque por defecto (default)
--	--	--

Por lo tanto:

En la estructura If-else si la condición es verdadera se ejecutan las sentencias if y si la condición no es cierta, las de else.

La estructura switch toma múltiples decisiones en torno a una variable y es el más utilizado para hacer menús.

Enumeraciones.

Una variable enumerador se puede crear de la siguiente manera:

```
enum identificador {VALOR1, VALOR2, ... , VALORN};
```

Para crear una enumeración se utiliza la palabra reservada enum, seguida de un identificador (nombre) y, entre llaves se ingresan los nombres de los valores que puede tomar dicha enumeración, separando los valores por coma. Los valores son elementos enteros y constantes (por lo tanto, se escriben con mayúsculas).

Estructura de control selectiva condicional

La estructura condicional permite realizar una comparación rápida. Su sintaxis:

Condición ? SiSeCumple : SiNoSeCumple

Consta de tres partes, una condición y dos acciones a seguir con base en la expresión condicional. Si la condición se cumple (es verdadera) se ejecuta la instrucción que se encuentra después del símbolo '?'; si la condición no se cumple (es falsa) se ejecuta la instrucción que se encuentra después del símbolo ':'.
‘.’.

RESULTADOS

Ejemplos. IF

```
#include<stdio.h>

/*
Este programa valida si el número a es mayor al número b.
*/

int main ()
{
    int a, b;
    a = 3;
    b = 2;

    if (a > b)
    {
        printf("\ta (%d) es mayor a b (%d).\n",a,b);
    }

    printf("\n\tEl programa sigue su flujo.\n");

    return 0;
}
```

```
Simbolo del sistema
C:\Users\La familia\Desktop\Lenguaje C\practicass>gcc if8.c -o if8.exe
C:\Users\La familia\Desktop\Lenguaje C\practicass>if8.exe
a (3) es mayor a b (2).
El programa sigue su flujo.
C:\Users\La familia\Desktop\Lenguaje C\practicass>
```

Imagen 1

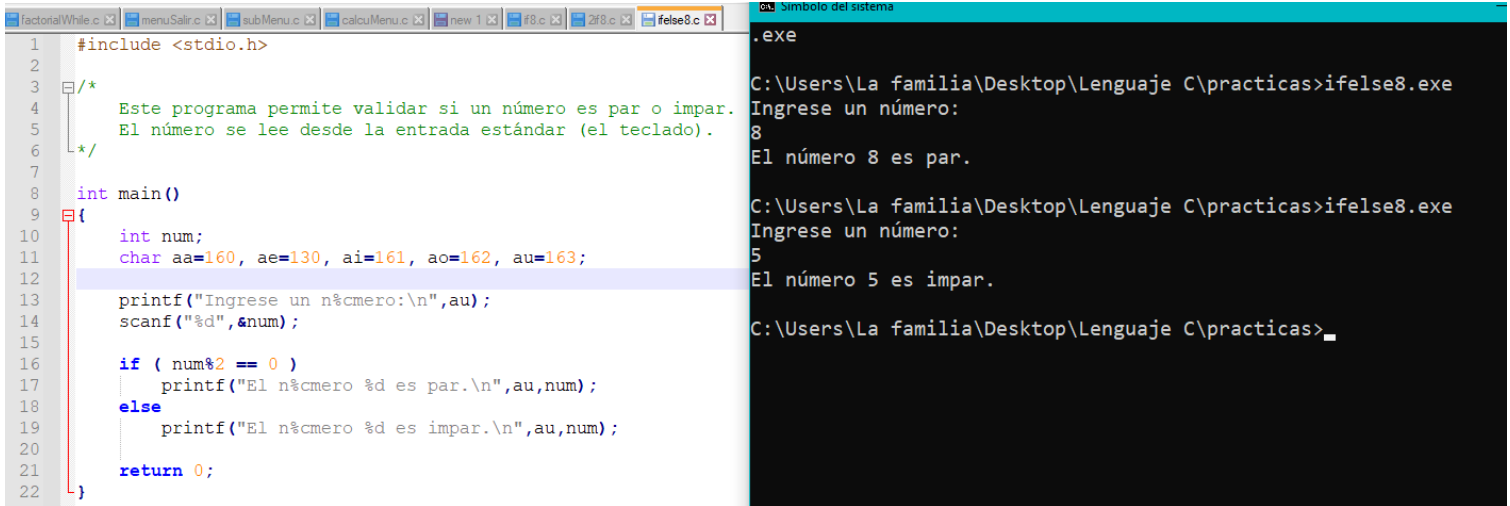
```
1 #include<stdio.h>
2
3 /*
4 Este programa valida si el número a es mayor al número b.
5 */
6
7 int main ()
8 {
9     int a, b;
10    a = 8;
11    b = 15;
12
13    if (a > b)
14    {
15        printf("\ta (%d) es mayor a b (%d).\n",a,b);
16    }
17
18    printf("\n\tEl programa sigue su flujo.\n");
19
20    return 0;
21 }
```

```
Simbolo del sistema
C:\Users\La familia\Desktop\Lenguaje C\practicass>gcc if8.c -o if8.exe
C:\Users\La familia\Desktop\Lenguaje C\practicass>if8.exe
El programa sigue su flujo.
C:\Users\La familia\Desktop\Lenguaje C\practicass>
```

Imagen 2

Como se observan en las imágenes la estructura if ejecuta ciertas acciones si la condición es verdadera y en caso de que sea falsa sigue corriendo el programa. Por ejemplo, en la imagen 1 "a" al ser mayor que "b" se cumple la condición y nos arroja en la pantalla que a es mayor a b, pero en la imagen 2 a "a" se le dio un valor menor a "b", por lo cual ya no se arrojó nada la pantalla, en el segundo caso el programa seguirá corriendo.

Ejemplos. IF-ELSE



The screenshot shows a C program in a text editor and its execution in a command prompt. The program, named `ifelse8.c`, includes `<stdio.h>` and has a comment: "Este programa permite validar si un número es par o impar. El número se lee desde la entrada estándar (el teclado)." The `main` function declares `int num;` and `char aa=160, ae=130, ai=161, ao=162, au=163;`. It prompts the user to enter a number and uses `scanf` to read it. An `if-else` statement checks if the number is even (`num%2 == 0`). If true, it prints "El número %d es par.", otherwise, it prints "El número %d es impar." The program returns 0.

```
1 #include <stdio.h>
2
3 /*
4  Este programa permite validar si un número es par o impar.
5  El número se lee desde la entrada estándar (el teclado).
6  */
7
8 int main()
9 {
10     int num;
11     char aa=160, ae=130, ai=161, ao=162, au=163;
12
13     printf("Ingrese un número:\n", au);
14     scanf("%d", &num);
15
16     if ( num%2 == 0 )
17         printf("El número %d es par.\n", au, num);
18     else
19         printf("El número %d es impar.\n", au, num);
20
21     return 0;
22 }
```

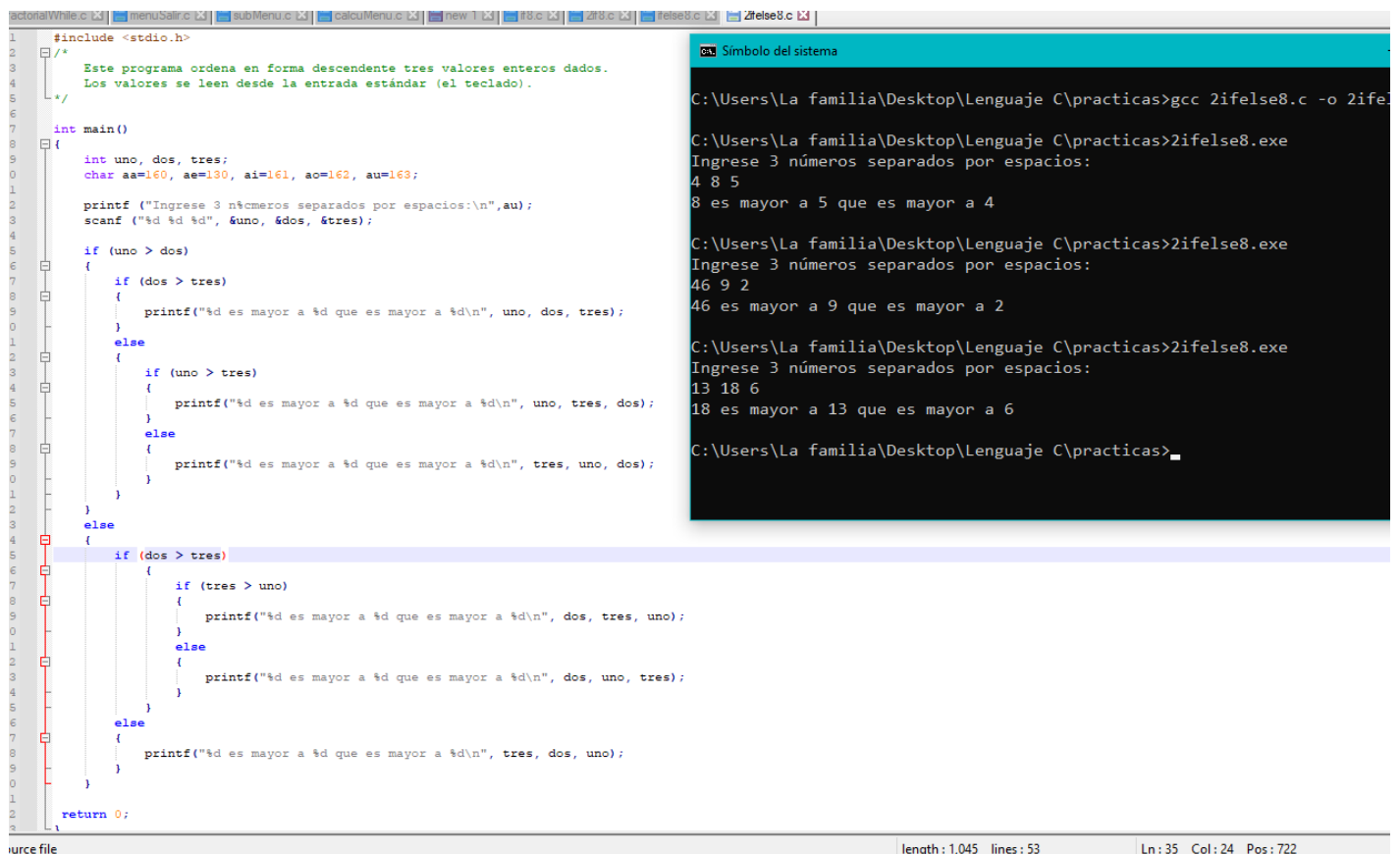
Simbolo del sistema

```
C:\Users\La familia\Desktop\Lenguaje C\practicas>ifelse8.exe
Ingrese un número:
8
El número 8 es par.

C:\Users\La familia\Desktop\Lenguaje C\practicas>ifelse8.exe
Ingrese un número:
5
El número 5 es impar.

C:\Users\La familia\Desktop\Lenguaje C\practicas>
```

Imagen 3



The screenshot shows a C program in a text editor and its execution in a command prompt. The program, named `2ifelse8.c`, includes `<stdio.h>` and has a comment: "Este programa ordena en forma descendente tres valores enteros dados. Los valores se leen desde la entrada estándar (el teclado)." The `main` function declares `int uno, dos, tres;` and `char aa=160, ae=130, ai=161, ao=162, au=163;`. It prompts the user to enter three numbers separated by spaces and uses `scanf` to read them. A series of nested `if-else` statements compare the numbers to determine the largest one. The program prints the result and returns 0.

```
1 #include <stdio.h>
2
3 /*
4  Este programa ordena en forma descendente tres valores enteros dados.
5  Los valores se leen desde la entrada estándar (el teclado).
6  */
7
8 int main()
9 {
10     int uno, dos, tres;
11     char aa=160, ae=130, ai=161, ao=162, au=163;
12
13     printf ("Ingrese 3 números separados por espacios:\n", au);
14     scanf ("%d %d %d", &uno, &dos, &tres);
15
16     if (uno > dos)
17     {
18         if (dos > tres)
19         {
20             printf("%d es mayor a %d que es mayor a %d\n", uno, dos, tres);
21         }
22         else
23         {
24             if (uno > tres)
25             {
26                 printf("%d es mayor a %d que es mayor a %d\n", uno, tres, dos);
27             }
28             else
29             {
30                 printf("%d es mayor a %d que es mayor a %d\n", tres, uno, dos);
31             }
32         }
33     }
34     else
35     {
36         if (dos > tres)
37         {
38             if (tres > uno)
39             {
40                 printf("%d es mayor a %d que es mayor a %d\n", dos, tres, uno);
41             }
42             else
43             {
44                 printf("%d es mayor a %d que es mayor a %d\n", dos, uno, tres);
45             }
46         }
47         else
48         {
49             printf("%d es mayor a %d que es mayor a %d\n", tres, dos, uno);
50         }
51     }
52
53     return 0;
54 }
```

Simbolo del sistema

```
C:\Users\La familia\Desktop\Lenguaje C\practicas>gcc 2ifelse8.c -o 2ifelse8.exe
C:\Users\La familia\Desktop\Lenguaje C\practicas>2ifelse8.exe
Ingrese 3 números separados por espacios:
4 8 5
8 es mayor a 5 que es mayor a 4

C:\Users\La familia\Desktop\Lenguaje C\practicas>2ifelse8.exe
Ingrese 3 números separados por espacios:
46 9 2
46 es mayor a 9 que es mayor a 2

C:\Users\La familia\Desktop\Lenguaje C\practicas>2ifelse8.exe
Ingrese 3 números separados por espacios:
13 18 6
18 es mayor a 13 que es mayor a 6

C:\Users\La familia\Desktop\Lenguaje C\practicas>
```

length: 1,045 lines: 53 Ln: 35 Col: 24 Pos: 722

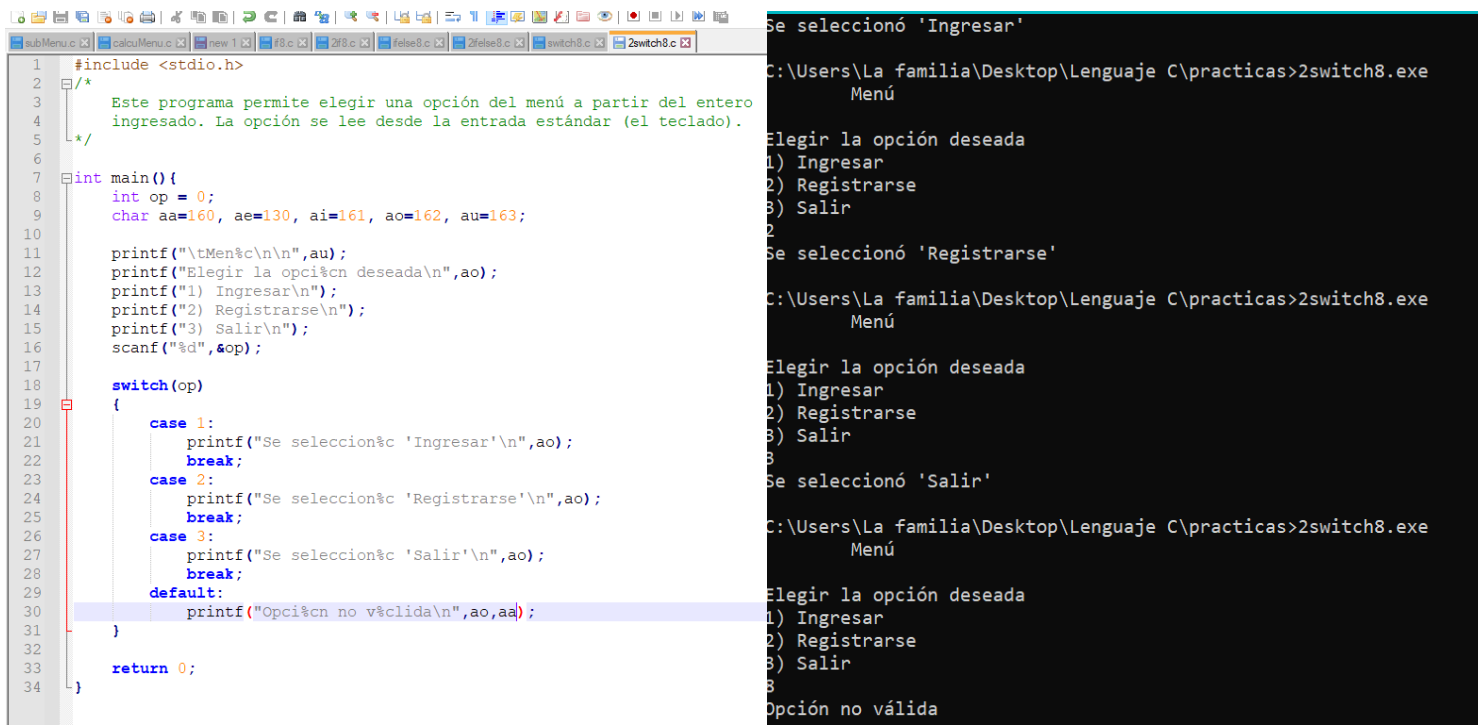
Imagen 4

La gran diferencia entre la estructura `if` e `if-else` es que en la segunda estructura

cuando la condición es falsa se ejecutan las acciones del else. En la imagen 3 se muestra como en el primer caso del if la condición es que el módulo del número entre 2 sea igual a 0 y si esto se cumple quiere decir que se trata de un número par, pero sí el resultado es diferente de 0 (la condición es falsa), se ejecutan las sentencias del else que en este caso nos arroja que el número es impar.

En la imagen 4, se nos muestra otro ejemplo en el que se puede utilizar la estructura if-else, lo destacable de dicho ejemplo es que dentro de una estructura se implementan otras estructuras if-else.

Ejemplos SWITCH



The image shows a C program in a code editor and its execution output in a terminal window. The program uses a switch statement to handle menu options.

```
1 #include <stdio.h>
2
3 /*
4  Este programa permite elegir una opción del menú a partir del entero
5  ingresado. La opción se lee desde la entrada estándar (el teclado).
6 */
7
8 int main() {
9     int op = 0;
10    char aa=160, ae=130, ai=161, ao=162, au=163;
11
12    printf("\tMenú\n", au);
13    printf("Elegir la opción deseada\n", ao);
14    printf("1) Ingresar\n");
15    printf("2) Registrarse\n");
16    printf("3) Salir\n");
17    scanf("%d", &op);
18
19    switch(op)
20    {
21        case 1:
22            printf("Se seleccionó 'Ingresar'\n", ao);
23            break;
24        case 2:
25            printf("Se seleccionó 'Registrarse'\n", ao);
26            break;
27        case 3:
28            printf("Se seleccionó 'Salir'\n", ao);
29            break;
30        default:
31            printf("Opción no válida\n", ao, aa);
32    }
33
34    return 0;
35 }
```

The terminal output shows the program's execution for three different inputs:

```
C:\Users\La familia\Desktop\Lenguaje C\practicas>2switch8.exe
Menú
Elegir la opción deseada
1) Ingresar
2) Registrarse
3) Salir
2
Se seleccionó 'Registrarse'

C:\Users\La familia\Desktop\Lenguaje C\practicas>2switch8.exe
Menú
Elegir la opción deseada
1) Ingresar
2) Registrarse
3) Salir
3
Se seleccionó 'Salir'

C:\Users\La familia\Desktop\Lenguaje C\practicas>2switch8.exe
Menú
Elegir la opción deseada
1) Ingresar
2) Registrarse
3) Salir
8
Opción no válida
```

Imagen 5

La estructura switch nos permite tomar múltiples decisiones en torno a una variable, en la imagen 5 se observa que dependiendo de lo que elijamos se ejecutarán distintas sentencias, por lo que esta estructura es ideal para los menús.

CONCLUSIONES

Las estructuras de selección como su nombre lo dicen, toma distintas decisiones dependiendo las condiciones y variable.

En el caso del if solo se realizan las sentencias cuando la condición es verdadera. Por otro lado, en el if-else e en el igual de igual forma se ejecutan las sentencias cuando la condición es verdadera y sino (cuando es falsa la condición) se ejecutan otras sentencias, que se encuentran abordados después del else.

En la estructura switch se dan distintas sentencias a los diversos casos que se dan, además de los casos también se puede dar el caso en que el resultado no coincida con ningún caso establecido por lo cual se ejecutarán las sentencias establecidas en el default.

Todas las estructuras vistas tienen sus ventajas, pero en general, nos ayudan a ejecutar distintas sentencias dependiendo el caso, por lo que nos ayudan a agilizar los programas y realizarlos más eficazmente.

REFERENCIAS

Facultad de Ingeniería. (2018, 6 abril). Manual de prácticas de fundamentos de programación MAD0-17_FP. Laboratorio de Computación Salas A y B. <http://lcp02.fi-b.unam.mx/>