



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: Marco Antonio Martinez Quintana

Asignatura: Fundamentos de Programación

Grupo: 3

No de Práctica(s): 6

Integrante(s): Lopez Martinez Diana

*No. de Equipo de
cómputo empleado:*

No. de Lista o Brigada: 28

Semestre: 1

Fecha de entrega: 27/11/2020

Observaciones:

CALIFICACIÓN: _____

Estructuras de Selección

Objetivo:

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.

Introducción:

Las estructuras de control de flujo en un lenguaje especifican el orden en que se realiza el procesamiento de datos.

Las estructuras de selección (o condicionales) permiten realizar una u otra acción con base en una expresión lógica. Las acciones posibles a realizar son mutuamente excluyentes, es decir, solo se puede ejecutar una a la vez dentro de toda la estructura.

Lenguaje C posee 3 estructuras de selección: la estructura if-else, la estructura switch y la estructura condicional o ternaria.

Estructura de control selectiva if

Su sintaxis es la siguiente:

```
if (expresión_lógica) {  
    // bloque de código a ejecutar  
}
```

En esta estructura 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.

NOTA 1: Si el bloque de código a ejecutar consta de una sola línea de código no es necesario el uso de las llaves

NOTA 2: La expresión lógica evaluada regresará como resultado un número entero. Dentro de las estructuras de control 0 indica que la expresión lógica es falsa y cualquier número diferente de 0 indica que la expresión lógica es verdadera.

Código (estructura de control selectiva if)

```
estructuraSelectivaif: Bloc de notas  
Archivo Edición Formato Ver Ayuda  
#include<stdio.h>  
  
/*  
Este programa valida si el número a es mayor al número b.  
*/  
int main (){  
    int a, b;  
    a = 8;  
  
    b = 5;  
  
    if (a > b) {  
  
        printf("\ta (%d) es mayor a b (%d).\n",a,b);  
    }  
  
    printf("\t\nEl programa sigue su flujo.\n");  
  
    return 0;  
}
```

```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>gcc estructuraSelectivaif.c -o estructuraSelectivaif.exe  
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>estructuraSelectivaif.exe  
a (8) es mayor a b (5).  
El programa sigue su flujo.
```

estructuraSelectivaif: Bloc de notas
Archivo Edición Formato Ver Ayuda
#include<stdio.h>

```
/*
Este programa comprueba que las condiciones son numéricas
0 -> falso
≠ 0 -> Verdadero
*/

int main(){
    if (0){

        printf("Esta instrucción nunca se ejecuta\n");
        printf("porque la condición siempre es falsa (0).\n");

    }
    char au=162;
    if (-68)

        // El bloque de código de esta estructura if
        // solo consta de una línea porque los comentarios
        // no son tomados en cuenta por el compilador.
        // La condición siempre es verdadera (diferente de 0)
        printf("Esta instrucci%cn siempre se ejecuta.\n",au);

    return 0;
}
```

```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>gcc estructuraSelectivaif.c -o estructuraSelectivaif.exe
```

```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>estructuraSelectivaif.exe
Esta instrucción siempre se ejecuta.
```

Estructura de control selectiva if-else

La sintaxis de la estructura de control de flujo if-else es la siguiente:

```
if (expresión_lógica) {  
    // bloque de código a ejecutar  
    // si la condición es verdadera  
} else {  
    // bloque de código a ejecutar  
    // si la condición es falsa  
}
```

Esta estructura 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.

Es posible anidar varias estructuras if-else, es decir, dentro de una estructura if-else tener una o varias estructuras if-else.

Código (estructura de control selectiva if-else)

```
*estructuraSelectivaie: Bloc de notas  
Archivo Edición Formato Ver Ayuda  
#include <stdio.h>  
  
/*  
Este programa permite validar si un número es par o impar.  
El número se lee desde la entrada estándar (el teclado).  
*/  
int main(){  
    char au=163;  
    int num;  
  
    printf("Ingrese un número:\n",au);  
    scanf("%d",&num);  
  
    if ( num%2 == 0 )  
  
        printf("El número %d es par.\n",au,num);  
    else  
        printf("El número %d es impar.\n",au,num);  
  
    return 0;  
}
```

```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>gcc estructuraSelectivaie.c -o estructuraSelectivaie.exe
```


```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>estructuraSelectivaie.exe
```

```
Ingrese un número:
```

```
4
```

```
El número 4 es par.
```

Código (estructura de control selectiva if-else anidada)

 *estructuraSelectivaiea: Bloc de notas

Archivo Edición Formato Ver Ayuda

```
#include <stdio.h>
/*
    Este programa ordena en forma descendente tres valores enteros dados.
    Los valores se leen desde la entrada estándar (el teclado).
*/

int main(){
    char au=163;
    int num_1, num_2, num_3;

    printf ("Ingrese 3 nmeros separados por espacios:\n",au);
    scanf ("%d %d %d", &num_1, &num_2, &num_3);

    if (num_1 > num_2){
        if (num_2 > num_3){
            printf("%d es mayor a %d que es mayor a %d\n", num_1, num_2, num_3);
        }else {
            if (num_1 > num_3) {
                printf("%d es mayor a %d que es mayor a %d\n", num_1, num_3, num_2);
            } else {
                printf("%d es mayor a %d que es mayor a %d\n", num_3, num_1, num_2);
            }
        }
    } else {
        if (num_2 > num_3){
            if (num_3 > num_1) {
                printf("%d es mayor a %d que es mayor a %d\n", num_2, num_3, num_1);
            } else {
                printf("%d es mayor a %d que es mayor a %d\n", num_2, num_1, num_3);
            }
        } else {
            printf("%d es mayor a %d que es mayor a %d\n", num_3, num_2, num_1);
        }
    }

    return 0;
}
```

```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>gcc estructuraSelectivaiea.c -o estructuraSelectivaiea.exe
```

```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>estructuraSelectivaiea.exe
```

```
Ingrese 3 números separados por espacios:
```

```
5 8 2
```

```
8 es mayor a 5 que es mayor a 2
```

Estructura de control selectiva switch-case

La sintaxis de la estructura switch-case es la siguiente:

```
switch (opcion_a_evaluar){  
    case valor1:  
        /* Código a ejecutar*/  
        break;  
    case valor2:  
        /* Código a ejecutar*/  
        break;  
    ...  
    case valorN:  
        /* Código a ejecutar*/  
        break;  
    default:  
        /* Código a ejecutar*/  
}
```

La estructura switch-case 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). Los tipos de datos que puede evaluar esta estructura son enteros, caracteres y enumeraciones. 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.

Si la opción a evaluar no coincide dentro de algún caso, entonces se ejecuta el bloque por defecto (default). El bloque por defecto normalmente se escribe al final de la estructura, pero se puede escribir en cualquier otra parte. Si se escribe en alguna otra parte el bloque debe terminar con la palabra reservada break.

Código (estructura de control selectiva switch-case)

```
#include <stdio.h>

/*
    Este programa permite elegir una opción del menú a partir del carácter
    ingresado. La opción se lee desde la entrada estándar (el teclado).
*/

int main(){
    char au = 163;
    char ao = 162;
    char ad = 160;
    char ae = 130;
    char op = '\0';

    printf("\tMen%c\n\n",au);
    printf("Elegir la opci%cn deseada\n",ao);
    printf("a) Ingresar\n");
    printf("b) Registrarse\n");
    printf("c) Salir\n");
    scanf("%c",&op);

    switch(op) {
        default:
            printf("Opci%cn no v%clida, int%cntalo de nuevo.\n",ao,ad,ae);
            break;
        case 'a':
            printf("Se seleccion%c 'Ingresar'.\n",ao);
            break;
        case 'b':
            printf("Se seleccion%c 'Registrarse'.\n",ao);
            break;
        case 'c':
            printf("Se seleccion%c 'Salir'.\n",ao);
            break;
    }
    return 0;
}
```

```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>gcc estructuraSelectivaswc.c -o estructuraSelectivaswc.exe
```

```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>estructuraSelectivaswc.exe
```

```
Menú
```

```
Elegir la opción deseada
```

```
a) Ingresar
```

```
b) Registrarse
```

```
c) Salir
```

```
a
```

```
Se seleccionó 'Ingresar'.
```



```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>gcc estructuraSelectivaswc.c -o estructuraSelectivaswc.exe
```

```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>estructuraSelectivaswc.exe
```

Menú

Elegir la opción deseada


a) Ingresar

b) Registrarse

c) Salir

m

Opción no válida, inténtalo de nuevo.

 estructuraSelectivaswcb: Bloc de notas

Archivo Edición Formato Ver Ayuda

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

```
/*
```

```
Este programa permite elegir una opción del menú a partir del entero
ingresado. La opción se lee desde la entrada estándar (el teclado).
```

```
*/
```

```
int main(){
```

```
    char au = 163;
```

```
    char ao = 162;
```

```
    char ad = 160;
```

```
    char ae = 130;
```

```
    int op = 0;
```

```
    printf("\tMen%c\n\n",au);
```

```
    printf("Elegir la opci%cn deseada\n",ao);
```

```
    printf("1) Ingresar\n");
```

```
    printf("2) Registrarse\n");
```

```
    printf("3) Salir\n");
```

```
    scanf("%d",&op);
```

```
    switch(op) {
```

```
        case 1:
```

```
            printf("Se seleccion%c 'Ingresar'\n",ao);
```

```
        break;
```

```
        case 2:
```

```
            printf("Se seleccion%c 'Registrarse'\n",ao);
```

```
        break;
```

```
        case 3:
```

```
            printf("Se seleccion%o 'Salir'\n",ao);
```

```
            break;
```

```
        default:
```

```
            printf("Opci%cn no v%clida, int%cntalo de nuevo\n",ao,ad,ae);
```

```
    }
```

```
    return 0;
```

```
}
```

```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>gcc estructuraSelectivaswcb.c -o estructuraSelectivaswcb.exe
```

```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>estructuraSelectivaswcb.exe
```

```
Menú
```

```
Elegir la opción deseada
```

```
1) Ingresar
```

```
2) Registrarse
```

```
3) Salir
```

```
2
```

```
Se seleccionó 'Registrarse'
```

Enumeración

Existe otro tipo de dato constante conocido como enumeración. 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).

Ejemplo

```
enum boolean {FALSE, TRUE};
```

La enumeración se llama 'boolean' y contiene dos elementos, el primero (FALSE) posee el valor 0 y el siguiente (TRUE) posee el valor 1. Si hubiese más elementos en la enumeración, la numeración correría de manera ascendente.

Es posible cambiar el valor de un elemento, para ello solo se le asigna el valor deseado:
`enum diasSemana {LUNES, MARTES, MIERCOLES=5, JUEVES, VIERNES};`

Código (variables tipo enumeración)

```
#include <stdio.h>
/*
    Este programa crea diversas variables tipo enum (enumerador) y
    permite visualizar la manera en la que se maneja el tipo de dato.
*/
int main(){
    // declaración de la enumeración
    enum boolean {NO, YES};

    // declaración de una variable tipo enumeración
    enum boolean valorBooleano;
    valorBooleano = YES;

    // Se comprueba que el valor de una enumeración es entero
    printf("%d\n", valorBooleano);

    // Se comprueba que el valor de una enumeración se puede reasignar
    enum mesAño {ENERO, FEBRERO, MARZO=9, ABRIL, MAYO};
    printf("\n%d", ENERO);
    printf("\n%i", FEBRERO);
    printf("\n%d", MARZO);
    printf("\n%i", ABRIL);
    printf("\n%d\n", MAYO);

    return 0;
}
```

```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>gcc estructuraSelectivaen.c -o estructuraSelectivaen.exe
```

```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>estructuraSelectivaen.exe
```

```
1
0
1
9
10
11
```

```
#include <stdio.h>
/*
    Este programa permite elegir una opción del menú a partir del entero
    ingresado. La opción se lee desde la entrada estándar (el teclado).
*/
int main(){
    char an=164;
    // Los valores de una enumeración son enteros y constantes
    enum mesAño {ENERO, FEBRERO, MAYO, SEPTIEMBRE, OCTUBRE, NOVIEMBRE, DICIEMBRE};
    int op;
    printf("Ingrese el mes del a%co.\n",an);
    printf("1) ENERO\n");
    printf("2) FEBRERO\n");
    printf("3) MAYO\n");
    printf("4) SEPTIEMBRE\n");
    printf("5) OCTUBRE\n");
    printf("6) NOVIEMBRE\n");
    printf("7) DICIEMBRE\n");
    scanf("%d", &op);

    switch(op-1){
        case ENERO:
        case FEBRERO:
            printf("Inicio de a%co.\n",an);
            break;
        case MAYO:
            printf("Mitad de a%co.\n",an);
            break;
        case SEPTIEMBRE:
            printf("¡Casi inicia el fin de a%co!\n",an);
            break;
        case OCTUBRE:
        case NOVIEMBRE:
            printf("¡Fin de a%co!\n",an);
            break;
        case DICIEMBRE:
            printf("Ultimo mes del a%co!!!.\n",an);
            break;
        // No se necesita default
    }

    return 0; // Valor entero en hexadecimal
}
```

```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>gcc enumeracionb.c -o enumeracionb.exe
```

```
C:\Users\paust\Downloads\Lenguaje c\Ejemplos>enumeracionb.exe
```

```
Ingrese el mes del año.
```

- 1) ENERO
- 2) FEBRERO
- 3) MAYO
- 4) SEPTIEMBRE
- 5) OCTUBRE
- 6) NOVIEMBRE
- 7) DICIEMBRE

```
4
```

```
¡Casi inicia el fin de año!
```

Estructura de control selectiva condicional

La estructura condicional (también llamado operador ternario) permite realizar una comparación rápida. Su sintaxis es la siguiente:

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 ':'.
.

Código (Estructura de control selectiva condicional o ternaria)



*condicional: Bloc de notas

Archivo Edición Formato Ver Ayuda

```
#include <stdio.h>
/*
Este programa permite calcular el error matemático a partir de dos
valores (a y b) ingresados desde la entrada estándar (el teclado), a partir
de la fórmula:
 $E = |a - b|$ 
Donde a es el valor real y b es el valor aproximado o viceversa.
*/
int main(){
    char ad=160;
    double a, b, res;

    printf("Calcular el error matemático E = |a - b|\n\n",ad);
    printf("Por favor ingrese el valor de a:\n");
    scanf("%lf",&a);
    printf("Por favor ingrese el valor de b:\n");
    scanf("%lf",&b);

    res = a < b ? b-a : a-b;

    printf("El error matemático de\n",ad);
    printf("| %lf - %lf | es %lf\n", a, b, res);

    return 0;
}
```

Conclusion

Para crear códigos en programación las estructuras de control son un pilar fundamental ya que con ellas se ejecuta un programa y su manejo permitirá realizarlo correctamente.

Bibliografía

Solano Gálvez, García Cano, Sandoval Montaña, Nakayama Cervantes, Arteaga Ricci, Castañeda Perdomo, I., 2020. Laboratorio Salas A Y B. [online] Lcp02.fi-b.unam.mx. Available at: <<http://lcp02.fi-b.unam.mx/>> [Accessed 6 April 2020].