

## **Sentencias decisivas, condicionales o bifurcaciones**

Una bifurcación o condicional es una estructura de control en la cual, a partir del valor de una expresión, se sigue por uno u otro camino:

**Ejemplo\_1:** construya un programa que ingrese un número entero y emita un mensaje que diga si es positivo.

/\*Estructura Selectiva simple\*/

```
#include <stdio.h>

int main(){
int num;

printf("Ingrese un número\n");
scanf("%d",&num);

if (num > 0)
    printf("Es positivo");

return 0;
}
```

(num > 0) es la condición que se evalúa, si es verdadera se ejecutan todas las sentencias que dependen de ella, si es falsa - en este caso - el programa termina sin dar otro resultado.

**Ejemplo\_2:** construya un programa que ingrese un número entero, emita un mensaje que diga si es positivo y le sume 1.

/\*Estructura Selectiva simple\*/

```
#include <stdio.h>

int main(){
int num;

printf("Ingrese un número\n");
scanf("%d",&num);

if (num > 0) {
    printf("Es positivo");
    num = num + 1;
}

return 0;
}
```

Nótese que las dos instrucciones que dependen de la condición (verdadera) se encierran entre paréntesis. Cuando es sólo una no es obligatorio pero puede utilizarse igual.

**Ejemplo\_3:** construya un programa que ingrese un número entero, emita un mensaje que diga si es positivo y le sume 1. En el caso que se negativo que también emita un mensaje.

/\*Estructura Selectiva doble\*/

```
#include <stdio.h>

int main(){
int num;

printf("Ingrese un número\n");
scanf("%d",&num);

if (num > 0) {
    printf("Es positivo");
    num = num + 1;
}
else printf("Es negativo");

return 0;
}
```

En este caso, si ingresamos un número negativo, la condición es falsa, en consecuencia se ejecuta lo que indique **else**. En el else no se escriben condiciones

**Ejemplo\_4:** construya un programa que ingrese un número entero, emita un mensaje que diga si es positivo y le sume 1. En el caso que se negativo que también emita un mensaje y le reste 2;  
/\*Estructura Selectiva doble\*/

```
#include <stdio.h>

int main(){
int num;

printf("Ingrese un número\n");
scanf("%d",&num);

if (num > 0) {
    printf("Es positivo");
    num = num + 1;
}
else {
    printf("Es negativo");
    num = num - 2;
}
return 0;
}
```

Nótese que en el else, al haber más de una condición, también debemos encerrar entre llaves todas las instrucciones que dependan de que la condición sea falsa.

**Ejemplo\_5:** construya un programa que ingrese un número entero, emita un mensaje que diga si es positivo y le sume 1. En el caso que se negativo que también emita un mensaje y le reste 2. En el caso de ser 0 (cero), que también emita un mensaje.  
/\*Estructura Selectiva múltiple\*/

```
#include <stdio.h>

int main(){
int num;

printf("Ingrese un número\n");
scanf("%d",&num);

if (num > 0) {
    printf("Es positivo");
    num = num + 1;
}
else if (num < 0){
    printf("Es negativo");
    num = num - 2;
}
else printf ("El valor ingresado es 0");
return 0;
}
```

Cuando tenemos que evaluar más de una condición podemos utilizar **else if**; else if debe llevar una o más condiciones para ser evaluadas. Un bloque de instrucciones if/else if/else puede contener muchos else if.

Programa de ejemplo:

```
#include <stdio.h>
int main()
{
    int num1; /* primer número que lee el usuario */
    int num2; /* segundo número que lee el usuario */

    printf( "Introduza dos enteros, y le dire\n" );
    printf( "las relaciones que satisfacen: " );
```

```

scanf( "%d%d", &num1, &num2 ); /* lectura de los enteros */

if ( num1 == num2 ) {
    printf( "%d es igual que %d\n", num1, num2 );
} /* fin de if */

if ( num1 != num2 ) {
    printf( "%d no es igual que %d\n", num1, num2 );
} /* fin de if */

if ( num1 < num2 ) {
    printf( "%d es menor que %d\n", num1, num2 );
} /* fin de if */

if ( num1 > num2 ) {
    printf( "%d es mayor que %d\n", num1, num2 );
} /* fin de if */

if ( num1 <= num2 ) {
    printf( "%d es menor o igual que %d\n", num1, num2 );
} /* end if */

if ( num1 >= num2 ) {
    printf( "%d es mayor o igual que %d\n", num1, num2 );
} /* fin de if */

return 0; /* indica que el programa terminó con éxito */

} /* fin de la función main */

```

Fuente: Deitel y Deitel

*Podrías hacer el mismo programa utilizando else if?*

En este tipo de estructura puede haber más de una condición, unidas por operadores lógicos.

### **Sentencia switch..case:**

La sentencia switch es una instrucción de control que controla múltiples selecciones y enumeraciones pasando el control a una de las instrucciones case de su cuerpo. Se utiliza cuando el algoritmo se divide en más de dos "ramas" a partir del valor de cierta expresión.

Programa de ejemplo)

```

#include <stdio.h>
#include <math.h>

int main() {
    int x;
    double num;
    {
        printf("\n(1) EL TRIPLE");
        printf("\n(2) EL CUADRADO");
        printf("\n(3) LOGARITMO NATURAL");
        printf("\n(4) LOGARITMO DECIMAL");
        printf("\n(5) SENO");
        printf("\n(6) COSENO");
        printf("\n\n Escribe el número de la opción que desees: ");

        scanf("%d", &x);
        switch (x) {
            case 1:
                printf("\n\nEscribe el número: ");
                scanf("%lf", &num);
                printf("\nEl triple de %lf es %lf", num, 3*num);
                break;

```

```

case 2:
    printf("\n\nEscribe el número: ");
    scanf("%lf",&num);
    printf("\nEl cuadrado de %lf es %lf",num,num*num);
    break;
case 3:
    printf("\n\nEscribe el número: ");
    scanf("%lf",&num);
    printf("\nEl logaritmo neperiano de %lf es %lf",num,log(num));
    break;
case 4:
    printf("\n\nEscribe el número: ");
    scanf("%lf",&num);
    printf("\nEl logaritmo decimal de %lf es %lf",num,log10(num));
    break;
case 5:
    printf("\n\nEscribe el número: ");
    scanf("%lf",&num);
    printf("\nEl seno de %lf es %lf",num,sin(num));
    break;
case 6:
    printf("\n\nEscribe el número: ");
    scanf("%lf",&num);
    printf("\nEl coseno de %lf es %lf",num,cos(num));
    break;
default:printf("\n\nEsto no es ninguna opción ");
    break;

    }
}

```

#### **Notas:**

- **switch(x):** switch compara el valor de la variable (en el ejemplo es x) con cada valor de case, la variable **sólo** puede ser entero o carácter. El bloque del switch se encierra entre llaves.
- **case 1: ..... break; case 2: ..... break;...** no hay límites para la cantidad de casos. Cada caso puede tener una instrucción, bloque de instrucciones o programa completo, pero deben terminar con un **break;**
- **default:** es opcional
- cuando la variable a comprar entra en el bloque sólo ejecutará una de las instrucciones, la que coincida con el valor, en el caso que no encuentre el valor, o sale o se ejecutará el default.
- switch no admite rangos de valores ni flotantes ni cadenas.

Con esta guía pueden hacer todos los ejercicios de la Práctica de condicionales.pdf y Práctica de switch.pdf