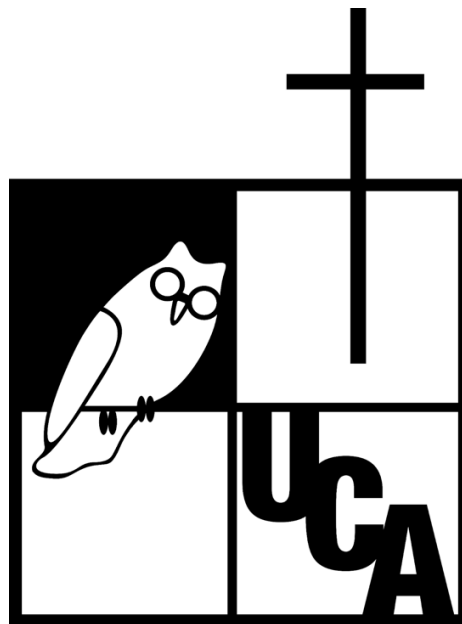


UNIVERSIDAD CENTROAMERICANA “JOSÉ SIMEÓN CAÑAS”

Departamento de Electrónica e Informática

Fundamentos de la programación

Ing. Elisa Aldana



Guía de laboratorio 2:

“Sentencias para la toma de decisiones simples y múltiples, Estructura if-else”

Fecha a realizar:

Entre el lunes 18 de septiembre al viernes 23 de septiembre del 2023

Introducción

Las estructuras de control de flujo constituyen uno de los elementos fundamentales que potencian la capacidad de los desarrolladores para concebir soluciones de mayor complejidad y sofisticación. Estas estructuras de control actúan como herramientas fundamentales que posibilitan la definición precisa de la secuencia de ejecución de un programa y su comportamiento ante situaciones particulares definidas por el desarrollador.

Dentro del ámbito de la programación, y más específicamente en el contexto de las estructuras de control, uno de los conceptos primordiales reside en el empleo de condicionales, los cuales confieren la capacidad de generar soluciones dinámicas y adaptables.

Los condicionales facultan al programa para tomar decisiones basadas en condiciones o criterios particulares que nosotros como programadores definiremos, aspecto crucial para asegurar el funcionamiento impecable de la aplicación. Entre las estructuras condicionales de mayor recurrencia en programación, figuran "if", "else" y "switch-case" (esta última la veremos en un laboratorio mas adelante), las cuales habilitan la definición de diversos cursos de acción en función de condiciones específicas.

En esta introducción al uso de las mismas emprenderemos un recorrido por los fundamentos de los condicionales en programación y su aplicación concreta en el desarrollo de aplicaciones, para ello desarrollaremos a lo largo del laboratorio el uso de "if", "else" y "else-if"

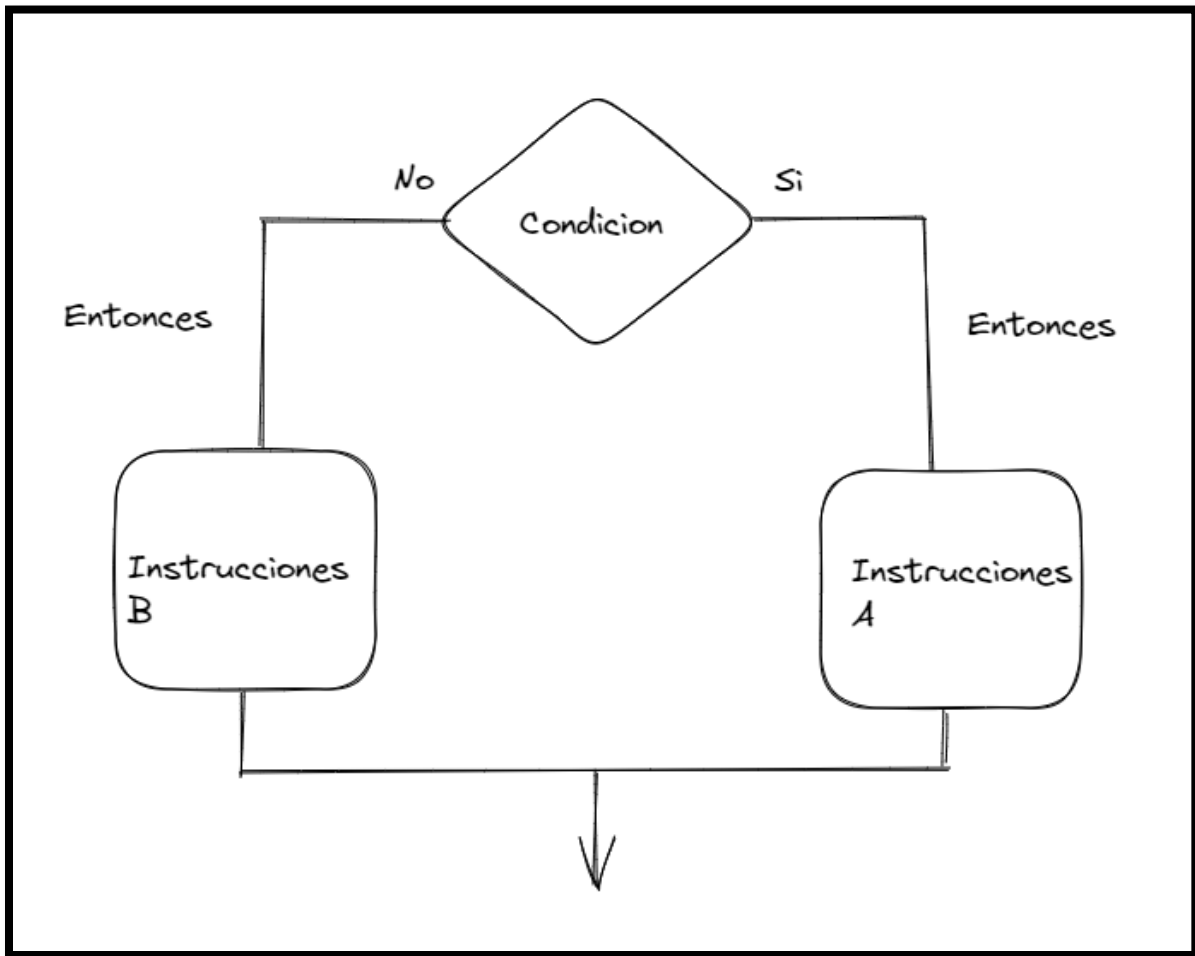
Uso de if – else

El condicional "if-else" es una estructura de control fundamental en programación que permite tomar decisiones basadas en una condición. Esta estructura permite que un programa evalúe una expresión o una condición y ejecute un bloque de código específico dependiendo de si esa condición es verdadera o falsa.

En otras palabras, el "if-else" se utiliza para crear un desvío en el flujo de ejecución de un programa. Si la condición dentro del "if" es evaluada como verdadera, el bloque de código asociado a ese "if" se ejecuta.

En caso contrario, es decir, si la condición es falsa, se ejecuta el bloque de código dentro del "else" (que debe estar presente), que ofrece una ruta alternativa de ejecución, o bien, una culminación del mismo que no necesariamente debe ser concluyente.

El "if-else" es esencial para crear programas que tomen decisiones y se adapten a diferentes situaciones, lo que es clave en la programación para construir aplicaciones más interactivas y funcionales.



La sintaxis básica del condicional “if-else” en C++ es la siguiente:

```
if (condición) {  
    // Código a ejecutar si la condición es verdadera  
}  
else {  
    // Código a ejecutar si la condición es falsa  
}
```

En esta estructura la condición es una expresión booleana que se evalúa y nos da como resultado verdadera o falsa, es decir, se cumple o no se cumple la condición preestablecida en nuestro programa. Como fue planteado con anterioridad en la guía; **sí la condición dentro del "if" es evaluada como verdadera, el bloque de código asociado a ese "if" se ejecuta. En caso contrario, es decir, si la condición es falsa, se ejecuta el bloque de código dentro del "else" (que debe estar presente), que ofrece una ruta alternativa de ejecución.**

Realizaremos un pequeño ejemplo para poder poner en práctica el uso de la condicional “if-else” en C++:

```
#include <iostream>
using namespace std;

int main() {
    int edad;
    cout << "Ingresa tu edad: ";
    cin >> edad;
    if (edad ≥ 18) {
        cout << "Eres mayor de edad" << endl;
    }
    else {
        cout << "Eres menor de edad" << endl;
    }
    return 0;
}
```

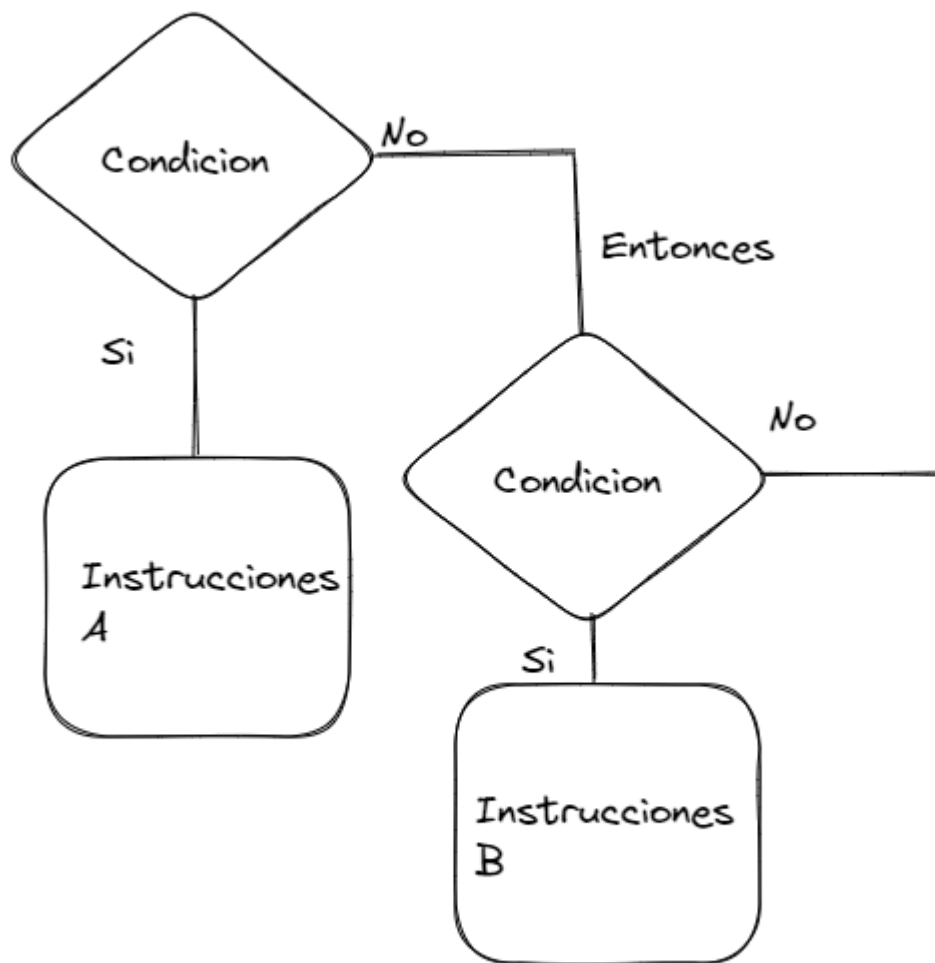
En este ejemplo, se pide al usuario que ingrese su edad y se utiliza un condicional “if-else” para determinar si es mayor o menor de edad. Si la edad es mayor o igual a 18, se imprime "Eres mayor de edad", y si es menor de 18, se imprime "Eres menor de edad".

Vamos con algo un poco más complejo y divertido que nos permitirá parametrizar de mejor manera nuestro código.

Uso de else if

C++ también admite anidamiento, lo que significa que se pueden utilizar varias condiciones “if-else” en el mismo bloque de código. Esto puede ser útil para crear soluciones más complejas y sofisticadas en programas de mayor envergadura.

Para ello conoceremos el “else-if”, esta es una estructura de control utilizada en muchos lenguajes de programación para especificar una condición adicional a una declaración “if”. Esta estructura permite evaluar varias condiciones de manera secuencial y tomar diferentes acciones dependiendo de cuál sea verdadera o falsa.



La sintaxis básica del condicional “else-if” en C++ es la siguiente:

```
if (condición1) {  
    // código a ejecutar si la condición1 es verdadera  
} else if (condición2) {  
    // código a ejecutar si la condición2 es verdadera  
} else {  
    // código a ejecutar si ninguna de las condiciones anteriores es verdadera  
}
```

Ahora bien, ¿qué significa esto?, ¿qué se hace por cada condición?

En el ejemplo de la sintaxis se evalúa la “condición1” en primera instancia, si es verdadera pues se ejecutará el bloque de código que encuentra debajo del “if”. En caso no lo sea pasaremos a nuestra “condición2”, en caso sea verdadera se ejecutará el bloque de código que se encuentre debajo del “else if”. Y en caso “ninguna” de las condiciones anteriores sea verdadera (ojo con esto, únicamente se presentan 2 condiciones pero nosotros somos capaces como programadores de agregar tantas como nosotros deseemos), pues se ejecutará el bloque de código que se encuentre debajo del “else”.

Realizaremos un ejemplo para poder poner en práctica el uso de la condicional “else if” en C++:

```
#include <iostream>  
using namespace std;  
  
int main() {  
    int calificacion = 80;  
  
    if (calificacion ≥ 90) {  
        cout << "La calificación es A\n";  
    } else if (calificacion ≥ 80) {  
        cout << "La calificación es B\n";  
    } else if (calificacion ≥ 70) {  
        cout << "La calificación es C\n";  
    } else if (calificacion ≥ 60) {  
        cout << "La calificación es D\n";  
    } else {  
        cout << "La calificación es F\n";  
    }  
  
    return 0;  
}
```

Ejercicio evaluado

1. Accede a la tarea que está en Moodle llamada cp2.
2. Busca tu nombre y acepta la tarea.
3. Clona el repositorio.
4. Realiza el ejercicio que asigne el instructor/a.
5. Puedes verificar en tu repositorio de GitHub los cambios que realizaste al código.
6. Borrar las credenciales de Windows.
7. Borrar de la computadora el o los programas creados durante la práctica.

Ejercicios a desarrollar

1. Escribe un programa que permita al usuario introducir una letra del alfabeto y muestre en pantalla si es una vocal o una consonante.
2. Escribe un programa que permita al usuario introducir el precio de un producto y la cantidad que desea comprar, y calcule el total a pagar, teniendo en cuenta que si se compran más de 10 unidades se aplica un descuento del 10%, si se compran más de 20 unidades se aplica un descuento del 20% y si se compran más de 50 unidades se aplica un descuento del 25%.
3. Crea un programa que solicite al usuario que introduzca un número del 1 al 7 que represente un día de la semana y muestre en pantalla el mismo, de igual forma, que indique cuantos días faltan para que sea el último día de la semana. Considera que la semana inicia en lunes y termina en domingo.
4. Crea un programa que solicite al usuario que introduzca tres números enteros y determine cuál es el número más grande de los tres.