



# **Universidad Autónoma de Baja California**

## **Facultad de Ingeniería, Arquitectura y Diseño**



Ingeniero en Computación

**Asignatura:**

Programación Estructurada

**Actividad 4:**

**Brayan Arturo Rocha Meneses**

**Ensenada Baja California 26 de Febrero  
del 2024**

# Introducción

Las estructuras de control en C++ son fundamentales para la programación, ya que permiten que el flujo de un programa se controle de manera dinámica. Estas estructuras incluyen decisiones (if, switch), bucles (for, while, do-while) y control de flujo (break, continue, return). En esta práctica, exploraremos estas estructuras a través de varios ejercicios.

## Competencia

Al final de esta práctica, el estudiante será capaz de entender y aplicar las estructuras de control en C++ para resolver problemas de programación.

# Fundamentos

Las estructuras de control en C++ permiten que el programador especifique el flujo de control en un programa. Esto puede incluir la ejecución condicional de bloques de código y la repetición de bloques de código.

# Procedimiento

## EJERCICIOS EN C

1.- Programa en C que use un menú para realizar las 4 operaciones básicas, donde el usuario introduce 2 números enteros y realizará la operación según sea su selección del menú.

### MENÚ

- 1.- Suma
- 2.- Resta
- 3.- Multiplicación
- 4.- División

2.- Programa en C que use un menú para realizar conversiones de unidades de medida, donde el usuario selecciona una opción y realizará la opción según sea su selección del menú.

### MENÚ

- 1.- cm a pulgadas
- 2.- cm a pies
- 3.- Km a millas
- 4.- Pulgadas a cm
- 5.- pies a cm
- 6.- millas a Km

3.- Programa en C que lea 6 números, desplegar el valor del número mayor.

4.- Programa en C que sirva para calcular el salario semanal de un trabajador donde se obtiene como dato de entrada las **horas semanales** trabajadas, el **salario por hora**.

El programa deberá calcular el **salario normal**, **salario extra** y **salario total**, considerando lo siguiente:

Jornada Normal de 40 horas.

El salario normal se considera las horas trabajadas menores o igual a la jornada normal

Salario extra se considera las horas trabajadas mayores a la jornada normal y se pagan dobles las primeras 9 y triples a partir de la décima hora extra

**Nota:** Desplegar todos los datos (Salario x hora, Horas Trabajadas, Salario normal, Salario extra y Salario Total)

5.- Programa en C que sirva para desplegar el Total de una llamada telefónica donde se pide como datos de entrada los **minutos** y el **tipo de llamada**, se cobra de la siguiente manera:

### MENÚ

- 1.- **Llamada Local** \$3.00 sin límite de tiempo
  - 2.- **Llamada Nacional** \$7.00 por los 3 primeros minutos y \$2.00 minuto adicional
  - 3.- **Llamada Internacional** \$9.00 por los 2 primeros minutos y \$4.00 minuto adicional
- Desplegar, **Subtotal**, **Iva** (16%) y **Total**.

6.- Programa en C que sirva para calcular el Total a pagar por consumo de agua, donde el dato de entrada son los **M3 de agua** consumidos, Tomar en cuenta que **se cobra escalonada** de la Siguiente manera:

**Rango1:** 0 al 4 M3 \$50 x facturación sin importar cuánto consumió en este rango

**Rango2:** 5 a 15 M3 \$8.00 x M3

**Rango3:** 16 a 50 M3 \$10.00 x M3

**Rango4:** 51 M3 en adelante \$11.00 x M3

Nota: **Desplegar SubTotal, Iva(16%), y Total a pagar.**

7- En la materia programación estructurada se aplican 5 exámenes, calcular el promedio final de la materia donde la calificación menor de los exámenes se anula y el promedio se calcula en base a 4 exámenes.  
**Desplegar el promedio final.**

8.- Programa en C que sirva para el juego del CHINCHAMPU (Piedra, Papel, Tijera) para 1 jugador y la computadora, (**usar condición anidada**)

9.- Programa en C que sirva para el juego del CHINCHAMPU (Piedra, Papel, Tijera) para 1 jugador y la computadora, (**usar selección múltiple**)

## Conclusiones

Las estructuras de control en C++ son herramientas poderosas que permiten a los programadores controlar el flujo de un programa. A través de estos ejercicios, hemos visto cómo se pueden utilizar estas estructuras para resolver una variedad de problemas, desde realizar operaciones matemáticas básicas hasta realizar conversiones de unidades y calcular salarios. Con una comprensión sólida de estas estructuras, los programadores pueden comenzar a abordar problemas más complejos y escribir código más eficiente y efectivo.