



**UNIVERSIDAD AUTÓNOMA DEL ESTADO DE BAJA
CALIFORNIA
FACULTAD DE INGENIERÍA, ARQUITECTURA Y DISEÑO**

ASIGNATURA	PROGRAMACIÓN Y MÉTODOS NUMÉRICOS	CLAVE	33534
NOMBRE DE LA PRÁCTICA	Introducción al entorno y variables de programación en lenguaje C	PRÁCTICA NÚMERO	1
PROGRAMA EDUCATIVO	TRONCO COMÚN INGENIERÍA	PLAN DE ESTUDIOS	2020-2

1. INTRODUCCION

Gracias al Desarrollo de la actual era digital es imprescindible para un ingeniero el estudiar algún lenguaje de programación. Esto, además de desarrollar el pensamiento lógico, es una herramienta actual para la solución de diversos problemas en el ámbito de cualquier rama de la ingeniería. Implementar algoritmos y desarrollar programas computacionales es algo no ajeno a áreas distintas a la computación.

2. OBJETIVOS

- Identificar el entorno de desarrollo y la estructura de un programa en el lenguaje de programación, mediante la declaración de variables, constantes y funciones de entrada y salida, para la familiarización de su herramienta de trabajo, con actitud honesta y responsable.
- Identificar los pasos para creación de nuevos programas con la estructura del lenguaje de programación, aplicando los tipos de datos para declarar variables y constantes utilizando elementos de entrada y salida de datos.
- Comenzar a identificar el lenguaje de programación C y su sintaxis



UNIVERSIDAD AUTÓNOMA DEL ESTADO DE BAJA CALIFORNIA

FACULTAD DE INGENIERÍA, ARQUITECTURA Y DISEÑO

3. TEORÍA

Un algoritmo es una secuencia lógica de pasos para lograr un fin determinado. Parece esta definición ser muy técnica, sin embargo, nosotros conocemos el concepto de algoritmo de manera intrínseca ya que lo aplicamos de manera cotidiana. El simple hecho de levantarse y alistarse para salir a la escuela se da en una secuencia de diferentes pasos con ese propósito. Un programa computacional consiste en traducir una secuencia de instrucciones a algún lenguaje de programación de alto nivel. Se trata de una serie de líneas de código con diferentes instrucciones para que la computadora realice alguna tarea, la cual, a diferencia del ser humano, puede realizar de manera repetitiva, exhaustiva y más rápida.

Durante este laboratorio se manejará lenguaje de programación C ya que es muy didáctico. En efecto, al aprender este lenguaje el estudiante podrá aprender otros lenguajes con gran facilidad. Es portátil y posee un gran número de plataformas y sistemas operativos. Se permite en él la agrupación de instrucciones, la separación de módulos y es de alto nivel.

```
#include <stdio.h>

int main()
{
    printf( ";Hola mundo!");
    return 0;
}
```

Encabezado del programa. Se le dice al compilador que busque e interprete todos los elementos definidos en el archivo

Todo programa tiene una función principal (main). De esta depende la ejecución del programa.

Aquí va el cuerpo del programa. En este caso "printf" es una función para desplegar un valor o texto en pantalla. El texto que se desea imprimir se deberá resaltar entre comillas dobles.

La función principal regresa un entero. En este caso es cero indica que no se regresará nada y el programa finaliza.

Toda instrucción termina en punto y coma. Nos marcará error al no incluirlo.



UNIVERSIDAD AUTÓNOMA DEL ESTADO DE BAJA CALIFORNIA

FACULTAD DE INGENIERÍA, ARQUITECTURA Y DISEÑO

En algunas ocasiones, las instrucciones de un programa computacional requieren llevar a cabo cálculos. Para ello es necesario almacenar valores que podrán ser usados después (ejemplo: multiplicaciones o sumas de valores repetidos, imprimir en pantalla algún mensaje de error, etc.). En el caso de C se usan las **variables** para almacenar estos valores.

Antes de ejecutar las instrucciones del programa estos valores deberán **declararse**, esto es, definir a qué tipo pertenecen: entero, decimal, carácter, etc. Los tipos de datos especifican cómo la información será ingresada a nuestro programa en lenguaje C. Básicamente se conocen 2 tipos de datos. De manera general, podemos clasificar nuestros tipos de datos como:

- **Tipos de datos primarios:** Son los tipos de variables fundamentales e imprescindibles. Básicamente, enteros, punto flotante, caracteres y tipo void.
- **Tipos de datos derivados:** No son primarios, se derivan de grupos de variables de tipo primario. Por ejemplo: arreglos (arrays), estructuras, uniones, apuntadores.

El tipo de datos es determinado por su tipo en el encabezado del programa. Por ejemplo, si se declara a la variable **x** como **int** significa que la variable “x” sólo almacenará valores tipo enteros. Es necesario en C que se declaren todas las variables que se usarán en el programa.

Tipos de datos primarios.

Tipo de variable	Indicador de lectura/escritura	Ejemplos	Tamaño (bytes)	Intervalo
entero con signo	int	int deuda	2	-32,768 a 32767
entero sin signo	unsigned int	unsigned int edad	2	0-65535
Flotante (float)	float	float temperatura	4	3.4×10^{-38} 3.4×10^{38}
Caracter	double	double pi	8	1.7×10^{-308} 1.7×10^{308}



UNIVERSIDAD AUTÓNOMA DEL ESTADO DE BAJA CALIFORNIA

FACULTAD DE INGENIERÍA, ARQUITECTURA Y DISEÑO

Definición de constantes.

Una constante es un valor que no puede ser alterado o modificado durante la ejecución de un programa, únicamente se usa para ser leído. Se reserva en una parte de la memoria con valores fijos. Por ejemplo, puede definirse la constante $\pi=3.1415926$.

```
#define PI 3.1415926
```

Al definir una variable se le asigna un nombre por medio del signo “=”. Este nombre se sugiere que se elija de acuerdo con una descripción: peso_atomo, const_gravedad, area_edificio, etc. En C++ existen las siguientes restricciones:

- El nombre debe empezar con una letra o un guion bajo “_”. El resto de los caracteres pueden ser otras letras, números o guiones bajos: (alpha2, volumen_2, total_N).
- No se pueden usar los símbolos “\$” ó “%”. No se permiten los espacios, se sugiere para estos usar el guion bajo o tipo camel-case: centroMasa.
- Los nombres de las variables son muy sensibles a cualquier cambio, por ejemplo: centro_masa es una variable diferente a centro_Masa.
- No usar palabras reservadas: int, double, return, ya que éstas definen otras funciones en el código y puede estropearse.

Lectura e impresión de variables.

La función usada en lenguaje C para leer variables está dada por **scanf**. En ésta se deben indicar especificadores de formato para conocer el tipo de variable que se va a leer. Por ejemplo, el siguiente programa lee la edad de una persona y la imprime en la pantalla:

```
#include<stdio.h>

int edad; //Definimos la variable edad como tipo entero

main() {

    printf("¿Cuál es tu edad?");

    scanf("%d",&edad); //Leemos y guardamos la edad que ingresa el
    usuario en la variable tipo entero "edad".
```



**UNIVERSIDAD AUTÓNOMA DEL ESTADO DE BAJA
CALIFORNIA
FACULTAD DE INGENIERÍA, ARQUITECTURA Y DISEÑO**

```
printf("Tu edad es de %d años", edad);  
  
}
```

Observe que el indicador de variable “%d” nos indica que se leerá e imprimirá una variable de formato entero. Observe en la siguiente tabla los indicadores para los distintos tipos de variables.

Tipo de variable	Indicador
Entero	%d
Caracter	%c
Cadena de caracteres	%s
Flotante	%f
Double	%lf

4. DESARROLLO DE LA PRÁCTICA

Teniendo estas funciones y la manera de declarar variables realizar los siguientes programas:

- El usuario ingresa el radio de una esfera y se calculará su volumen $V = \frac{4}{3}\pi r^3$. Imprimir un mensaje con el volumen calculado en pantalla. Definir el volumen como una constante.
- Calcular la fuerza de atracción del peso de una persona $W = mg$ dada su masa m y la constante gravitatoria $g = 9.8 \frac{m}{s^2}$.
- El usuario ingresa su nombre, edad y peso (kg). El programa imprime un mensaje con su nombre, edad y peso y calcula su índice de masa corporal (IMC):

$$IMC = \frac{\text{peso}}{\text{estatura}^2}$$

4.3.- Errores

¿Por qué los siguientes programas presentan errores y no pueden ser compilados? Correr los programas en el simulador, analizarlos y reportar la causa del error.



**UNIVERSIDAD AUTÓNOMA DEL ESTADO DE BAJA
CALIFORNIA
FACULTAD DE INGENIERÍA, ARQUITECTURA Y DISEÑO**

A)

```
#include <stdio.h>
main()
{
    printf("El resultado de 2+5 es 7")
}
```

B)

```
#include <stdio.h>
#define PI 3.1416
float radio, area;
main()
{
    printf( "Ingrese valor para el radio");
    scanf("%d", &radio);
    Area = pi*r*r;
    printf("El area del circulo es %f", area);
}
```

C)

```
#include <stdio.h>
using namespace std;
char base, altura;
float area;
main()
{
    base = 4;
    altura = 6;
    area = base*altura;
    printf("El area del rectangulo es");
}
```



**UNIVERSIDAD AUTÓNOMA DEL ESTADO DE BAJA
CALIFORNIA
FACULTAD DE INGENIERÍA, ARQUITECTURA Y DISEÑO**

D)

```
#include <stdio.h>
double volumen;
double 5precio;

int main()
{
    volumen = 15;
    precio =volumen*34.5;
    printf("El precio final es");
    return 0;
}
```

Subir un reporte con los resultados obtenidos (incluir código de la parte 4.2) en la carpeta compartida en Blackboard. Se tienen 2 semanas como fecha límite. Reportar en las conclusiones los nuevos conocimientos adquiridos.