



## Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

# Laboratorios de computación salas A y B

*Profesor:* Alejandro Esteban Pimentel Alarcón

*Asignatura:* Fundamentos de Programación

*Grupo:* Grupo #3

*No de Práctica(s):* Práctica No. 7

*Integrante(s):* David Laureano González

*No. de Equipo de  
cómputo empleado:*

*No. de Lista o Brigada:* No. de Lista #25 / No. de Cuenta #9519

*Semestre:* 2020-1

*Fecha de entrega:* 3 de Octubre de 2019

*Observaciones:* Tarde entrega.  
Sigues sin incluir introducción y conclusiones.  
En tu primer captura se ve que no terminas de  
ejecutar el programa, observa los códigos, no se trata  
de que solo los copies, debes entenderlos.

**CALIFICACIÓN:** 7

## Objetivo

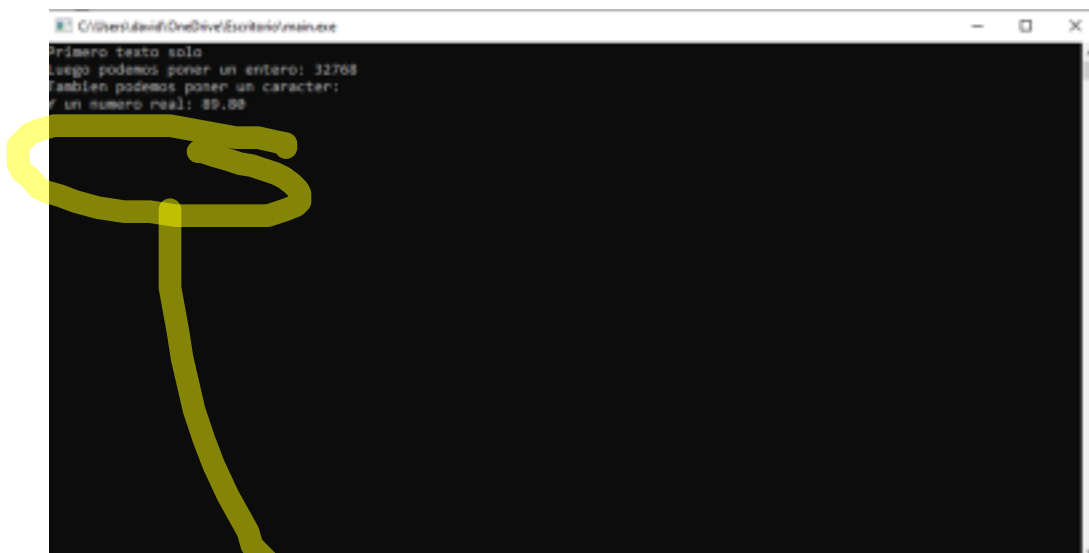
Elaborar programas en lenguaje C utilizando las instrucciones de control de tipo secuencia , para realizar la declaración de variables de diferentes tipos de datos, así como efectuar llamadas a funciones externas de entrada y salida para asignar y mostrar valores de variable y expresiones

## Actividad

Empezamos con un programa para mostrar y leer tipos de datos

```
main.c
1  #include <stdio.h>
2  int main(Datos)
3  {
4      int numeroEntrada;
5      double realEntrada;
6
7      int numeroEntero = 32768;
8      char caracter = 'B';
9      float numeroReal = 89.8;
10
11     printf("Primero texto solo\n");
12     printf("Luego podemos poner un entero: %i\n", numeroEntero);
13     printf("Tambien podemos poner un caracter: %c\n", caracter);
14     printf("Y un numero real: %.2f\n", numeroReal);
15
16     scanf("%i", &numeroReal);
17     scanf("%lf", &realEntrada);
18
19     printf("Tu entero: %i\n", numeroEntrada);
20     printf("Tu real: %.3lf\n", realEntrada);
21
22     return 0;
23 }
```

Compilamos y corremos



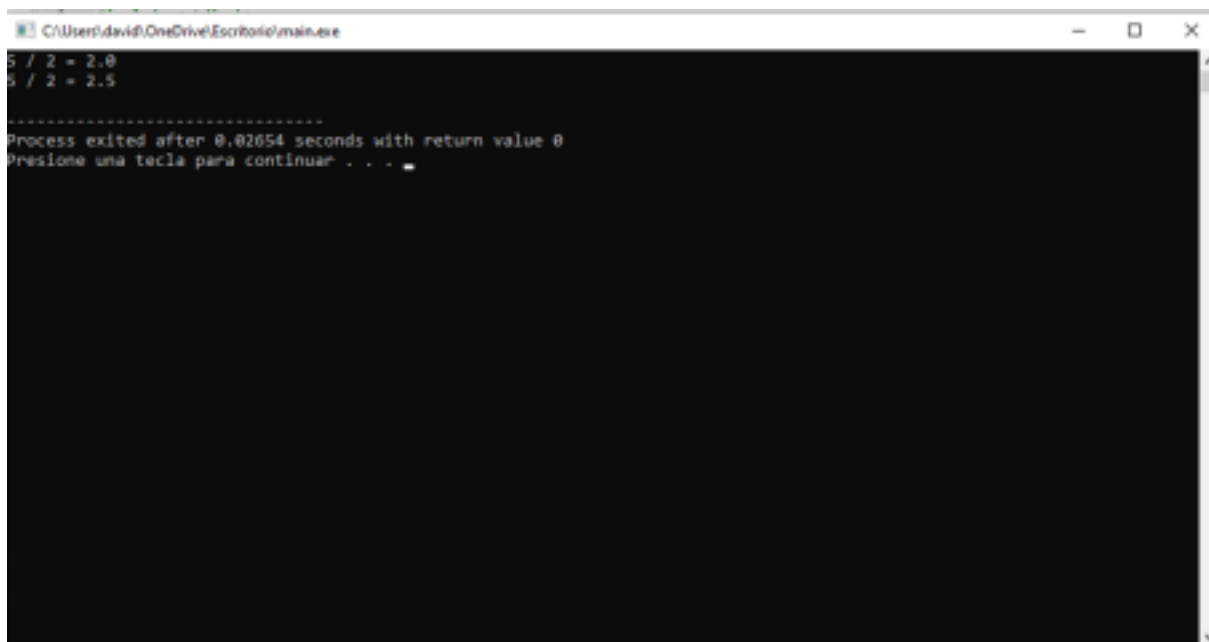
```
C:\Users\David\OneDrive\Escritorio\main.exe
Primero texto solo
Luego podemos poner un entero: 32768
Tambien podemos poner un caracter: B
Y un numero real: 89.00
```

Aquí el programa se quedó esperando valores de entreada, revisa el código.

Creamos un programa para usar operadores

```
main.c
1  #include <stdio.h>
2  int main(Operadores)
3  {
4      int dos, tres, cuatro, cinco;
5      double resultado;
6      dos = 2;
7      tres = 3;
8      cuatro = 4;
9      cinco = 5;
10     resultado = cinco/dos;
11     printf("5 / 2 = %.11f\n", resultado);
12     resultado = (double)cinco/dos;
13     printf("5 / 2 = %.11f\n", resultado);
14
15     return 0;
16
17 }
18
```

Compilamos y corremos



```
C:\User\david\OneDrive\Escritorio\main.exe
5 / 2 = 2.0
5 / 2 = 2.5

-----
Process exited after 0.02654 seconds with return value 0
Presione una tecla para continuar . . .
```

Y por ultimo creamos un programa que utilice operadores lógicos

```
main.c
1  #include <stdio.h>
2  int main(OperadoresLogicos)
3  {
4      int num1, num2, res;
5      char c1, c2;
6      num1 = 7;
7      num2 = 15;
8      c1 = "h";
9      c2 = "H";
10     printf("¿ num1 es menor a num2 ? -> %d\n", num1 < num2);
11     printf("¿ c1 es igual a c2 ? -> %d\n", c1 == c2);
12     printf("¿ c1 es diferente a c2 ? -> %d\n", c1 != c2);
13     res = num1 < num2 && c1 == "h";
14     printf("¿ num1 < num y c1 es igual a h ? -> %d\n", res);
15     res = c1 == "s" || c2 == "H";
16     printf("¿ c1 es igual a s o c2 a H ? -> %d\n", res);
17
18     return 0;
19 }
```

Compilamos y ejecutamos

```
C:\Users\dauid\OneDrive\Escritorio\main.exe
¿ num1 es menor a num2 ? -> 1
¿ c1 es igual a c2 ? -> 0
¿ c1 es diferente a c2 ? -> 1
num1 < num y c1 es igual a h ? -> 0
¿ c1 es igual a s o c2 a H ? -> 0

-----
Process exited after 0.04954 seconds with return value 0
Presione una tecla para continuar . . .
```