



## Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

# Laboratorios de computación salas A y B

*Profesor:* Alejandro Esteban Pimentel Alarcon

*Asignatura:* Fundamentos de Programación

*Grupo:* 3

*No. de práctica:* 7

*Integrantes:* Lucia Nicole Rosette Hernández

*No. de Equipo de cómputo empleado:*

*No. de lista o Brigada:* 420052768

*Semestre:* 1

*Fecha de entrega:* octubre 3, 2019

*Observaciones:* Muy bien

**CALIFICACIÓN: 10**

**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 variables y expresiones.

**Introducción:**

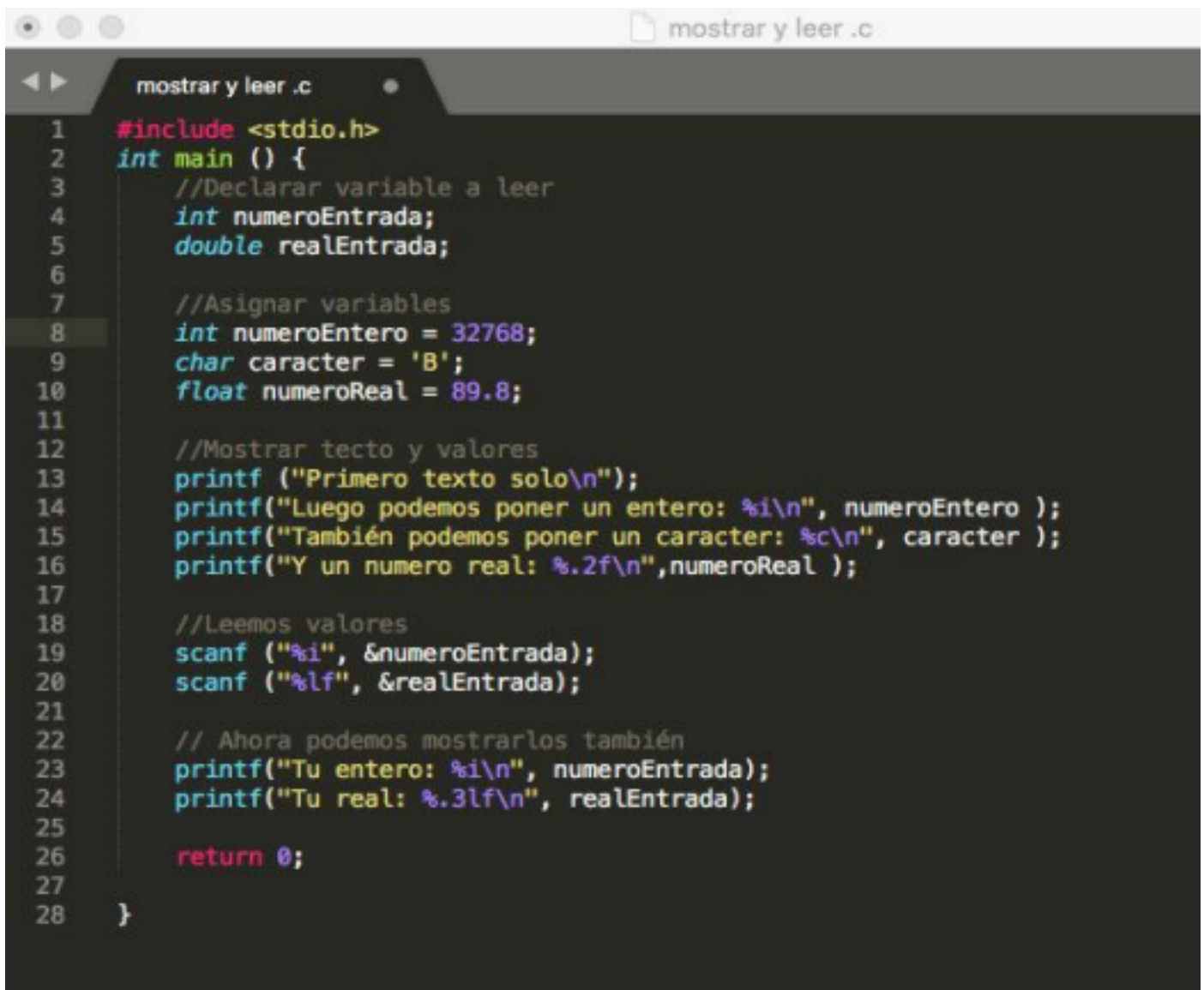
Las instrucciones de control son aquellas que nos permiten variar o alterar la secuencia normal de ejecución de un programa. En estas se encuentran las instrucciones condicionales o alternativas, Instrucciones de salto e Instrucciones repetitivas.

**Tipos de Variable**

A screenshot of a code editor window with a dark background. The title bar at the top shows three window control buttons on the left and a file icon followed by the text 'tipo de variable .c' on the right. The editor area displays C code with line numbers 1 through 15 on the left margin. The code defines a 'main' function that declares several variables: three integer types (short, signed int, unsigned long) and one character type, followed by two floating-point types (float, double), and ends with a return statement.

```
1  int main () {  
2      // variables enteras  
3      short numeroentero1;  
4      signed int numeroentero2;  
5      unsigned long numeroentero3;  
6  
7      //caracter  
8      char caracter;  
9  
10     //variables reales  
11     float puntoFlotante1;  
12     double puntoFlotante2;  
13  
14     return 0;  
15 }
```

## Mostrar y Leer

A screenshot of a code editor window titled 'mostrar y leer .c'. The editor shows a C program that demonstrates basic input and output operations. The code includes the standard input/output header, defines the main function, declares variables for an integer, a character, and a double, assigns initial values to some of them, and uses printf and scanf to display and read user input. The code is line-numbered from 1 to 28.

```
1  #include <stdio.h>
2  int main () {
3      //Declarar variable a leer
4      int numeroEntrada;
5      double realEntrada;
6
7      //Asignar variables
8      int numeroEntero = 32768;
9      char caracter = 'B';
10     float numeroReal = 89.8;
11
12     //Mostrar tecto y valores
13     printf ("Primero texto solo\n");
14     printf("Luego podemos poner un entero: %i\n", numeroEntero );
15     printf("También podemos poner un caracter: %c\n", caracter );
16     printf("Y un numero real: %.2f\n",numeroReal );
17
18     //Leemos valores
19     scanf ("%i", &numeroEntrada);
20     scanf ("%lf", &realEntrada);
21
22     // Ahora podemos mostrarlos también
23     printf("Tu entero: %i\n", numeroEntrada);
24     printf("Tu real: %.3lf\n", realEntrada);
25
26     return 0;
27
28 }
```

```
mostrar y leer — -bash — 101x38

Last login: Wed Oct  2 12:05:46 on ttys000
MacBook-Pro-de-Lucia:~ rafaelrosettetorres$ cd /Users/rafaelrosettetorres/Desktop/mostrar\ y\ leer
MacBook-Pro-de-Lucia:mostrar y leer rafaelrosettetorres$ gcc mostrar\ y\ leer\ .c -o main
MacBook-Pro-de-Lucia:mostrar y leer rafaelrosettetorres$ ./main
Primero texto solo
Luego podemos poner un entero: 32768
También podemos poner un caracter: B
Y un numero real: 89.80
3
6
Tu entero: 3
Tu real: 6.000
MacBook-Pro-de-Lucia:mostrar y leer rafaelrosettetorres$
```

### Leer y mostrar

```
operadores.c

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

```
operadores — -bash — 70×36
Last login: Wed Oct  2 12:26:00 on ttys000
MacBook-Pro-de-Lucia:~ rafaelrosettetorres$ cd /Users/rafaelrosettetorres/Desktop/operadores
MacBook-Pro-de-Lucia:operadores rafaelrosettetorres$ gcc operadores.c -o main
MacBook-Pro-de-Lucia:operadores rafaelrosettetorres$ ./main
5 / 2 = 2.0
5 / 2 = 2.5
MacBook-Pro-de-Lucia:operadores rafaelrosettetorres$
```

## Operadores Lógicos

```
operadores lógicos.c
1  #include <stdio.h>
2  int main () {
3      int num1, num2, res;
4      char c1, c2;
5
6      num1 = 7;
7      num2 = 15;
8      c1 = 'h';
9      c2 = 'H';
10
11     printf("%d num1 es menor a num2 ? -> %d\n", num1<num2);
12     printf("%d c1 es igual a c2 ? -> %d\n", c1==c2 );
13     printf("%d c1 es diferente a c2 ? -> %d\n", c1!=c2);
14
15     res = num1 < num2 && c1 == 'h';
16     printf("%d num1 < num2 Y c1 es igual a 'h' ? -> %d\n", res);
17
18     res = c1 == 's' || c2 == 'H';
19     printf("%d c1 es igual a 's' o c2 a 'H' ? -> %d\n", res );
20
21     return 0;
22 }
```



```
operadoreslógicos — -bash — 79x28
Last login: Wed Oct  2 13:05:01 on ttys000
MacBook-Pro-de-Lucia:~ rafaelrosettetorres$ cd /Users/rafaelrosettetorres/Desktop/operadoreslógicos
MacBook-Pro-de-Lucia:operadoreslógicos rafaelrosettetorres$ gcc operadores\ lógicos.c -o main
MacBook-Pro-de-Lucia:operadoreslógicos rafaelrosettetorres$ ./main
¿ num1 es menor a num2 ? ->      1
¿ c1 es igual a c2 ? ->        0
¿ c1 es diferente a c2 ? ->    1
¿ num1 < num2 Y c1 es igual a 'h' ? ->  1
¿ c1 es igual a 's' o c2 a 'H' ? ->    1
MacBook-Pro-de-Lucia:operadoreslógicos rafaelrosettetorres$
```

### **Conclusión:**

Para poder elaborar programas es necesario aprenderse los diferentes tipos de comando. Como declarar variables y los diferentes tipos de estas o como mostrar el resultado que deseamos.