



## Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

# Laboratorios de computación salas A y B

*Profesor:* Alejandro Pimentel Alarcón

*Asignatura:* Fundamentos de programación

*Grupo:* Bloque 135

*No de Práctica(s):* Práctica 7

*Integrante(s):* Partida Arias Emily Rachel

*No. de Equipo de cómputo  
empleado:*

*No. de Lista o Brigada:* 41

*Semestre:* 2020-1

*Fecha de entrega:*

*Observaciones:*

**CALIFICACIÓN:** \_\_\_\_\_

## Práctica 7. Fundamentos de lenguaje C.

### 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:

Tipos de variables

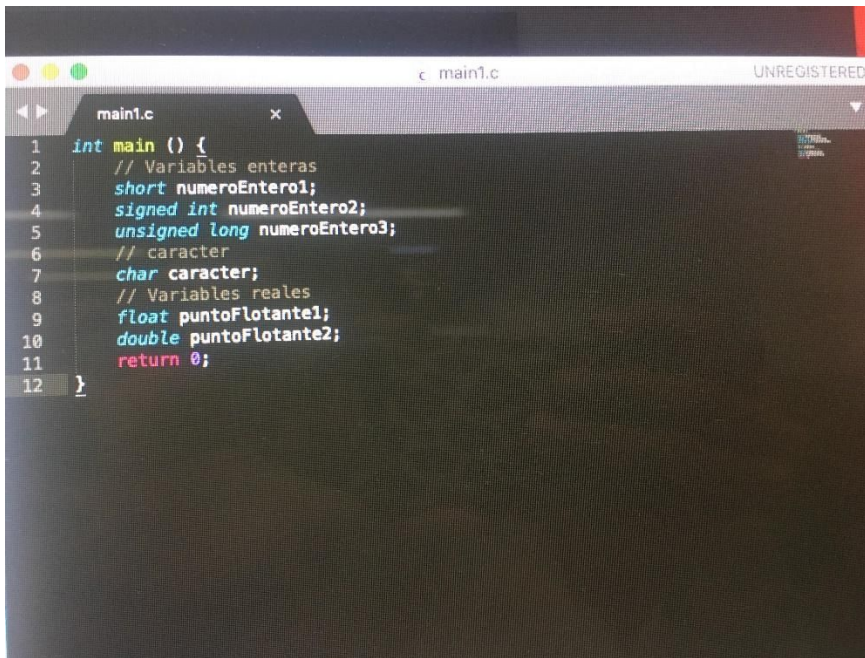
#### Tipos de datos (I)

Tipo	Bytes	Desde	Hasta
void	Es nulo (NULL) no retorna nada		
signed char	1	-128	127
unsigned char	1	0	255
signed short	2	-32768	32767
unsigned short	2	0	65535
signed int	2	-32768	32767

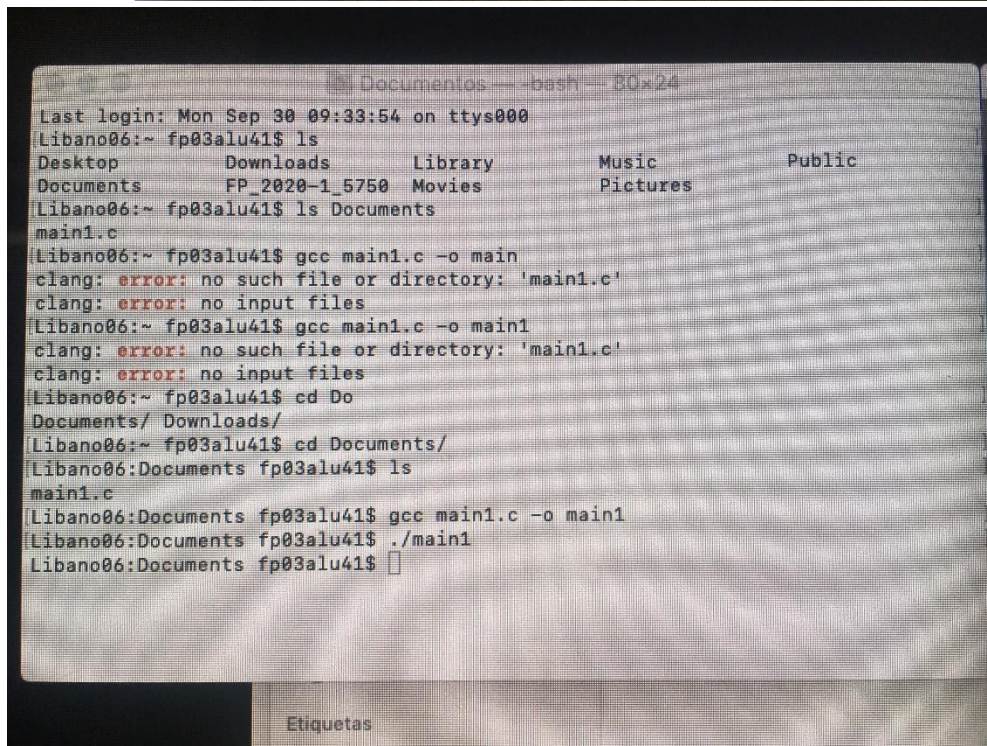
#### Tipos de datos (II)

Tipo	Bytes	Desde	Hasta
unsigned int	2	0	65535
signed long	4	-2147483648	2147483647
unsigned long	4	0	4294967295
float	4	$3,4 \times 10^{-38}$	$3,4 \times 10^{38}$
double	8	$1,7 \times 10^{-308}$	$1,7 \times 10^{308}$
long double	10	$3,4 \times 10^{-4932}$	$3,4 \times 10^{4932}$

## 1.- Tipos de variables



```
1 int main () {
2     // Variables enteras
3     short numeroEntero1;
4     signed int numeroEntero2;
5     unsigned long numeroEntero3;
6     // caracter
7     char caracter;
8     // Variables reales
9     float puntoFlotante1;
10    double puntoFlotante2;
11    return 0;
12 }
```

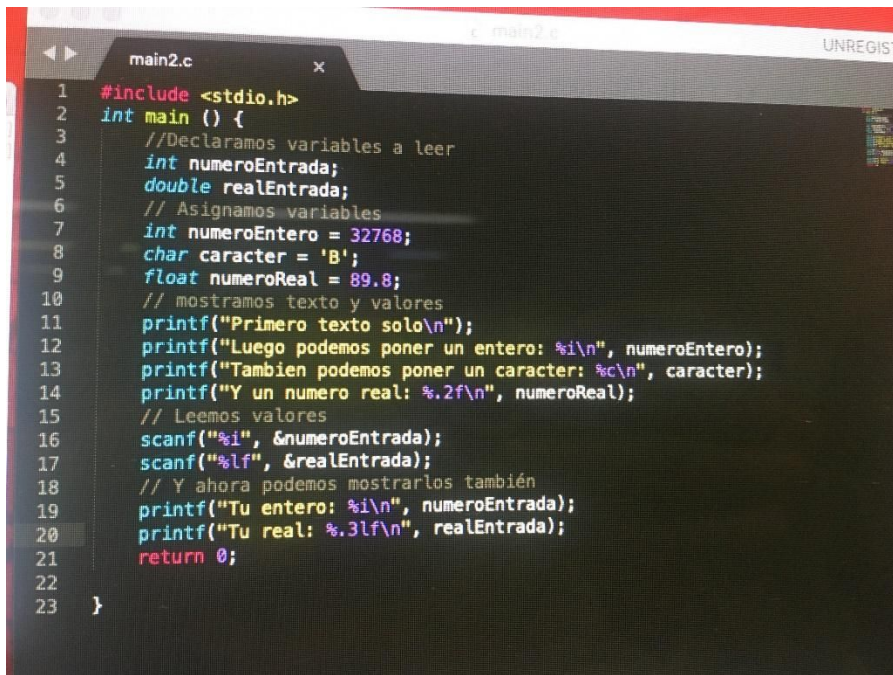


```
Documents -- bash -- 80x24
Last login: Mon Sep 30 09:33:54 on ttys000
Libano06:~ fp03alu41$ ls
Desktop      Downloads    Library      Music        Public
Documents    FP_2020-1_5750  Movies      Pictures
Libano06:~ fp03alu41$ ls Documents
main1.c
Libano06:~ fp03alu41$ gcc main1.c -o main1
clang: error: no such file or directory: 'main1.c'
clang: error: no input files
Libano06:~ fp03alu41$ gcc main1.c -o main1
clang: error: no such file or directory: 'main1.c'
clang: error: no input files
Libano06:~ fp03alu41$ cd Do
Documents/ Downloads/
Libano06:~ fp03alu41$ cd Documents/
Libano06:Documents fp03alu41$ ls
main1.c
Libano06:Documents fp03alu41$ gcc main1.c -o main1
Libano06:Documents fp03alu41$ ./main1
Libano06:Documents fp03alu41$
```

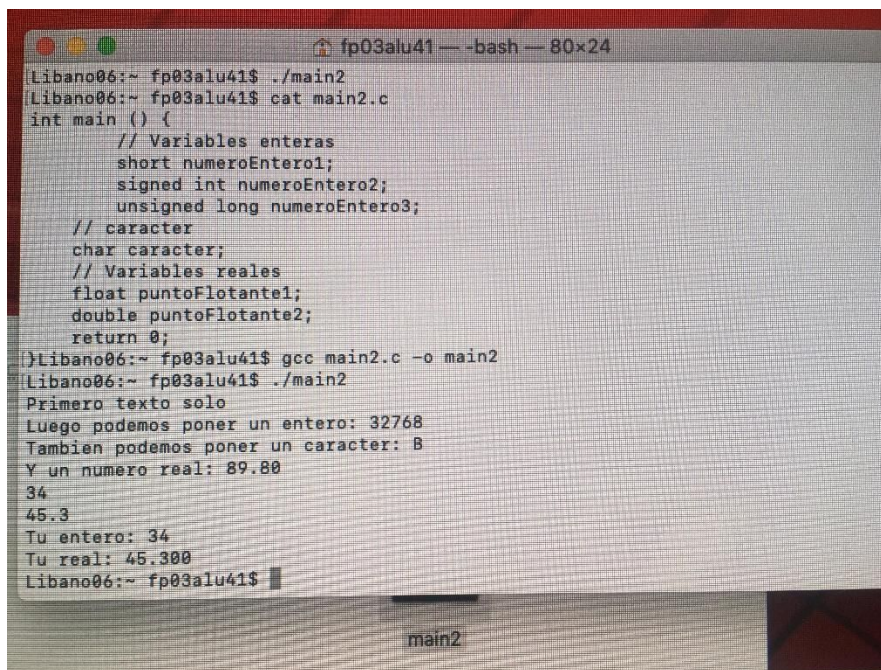
1.-Esta es una introducción a cómo se utilizan los comandos en C, primero se declaran las variables, se pone primero el tipo de variable y luego la variable que se desea, las // nos sirven para saber qué es lo que estamos haciendo nosotros.



## 2.- Mostrar y leer: el programa muestra un número real, entero o un caracter.



```
main2.c
1  #include <stdio.h>
2  int main () {
3      //Declaramos variables a leer
4      int numeroEntrada;
5      double realEntrada;
6      // Asignamos variables
7      int numeroEntero = 32768;
8      char caracter = 'B';
9      float numeroReal = 89.8;
10     // mostramos texto y valores
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     // Leemos valores
16     scanf("%i", &numeroEntrada);
17     scanf("%lf", &realEntrada);
18     // Y ahora podemos mostrarlos también
19     printf("Tu entero: %i\n", numeroEntrada);
20     printf("Tu real: %.3lf\n", realEntrada);
21     return 0;
22 }
23 }
```



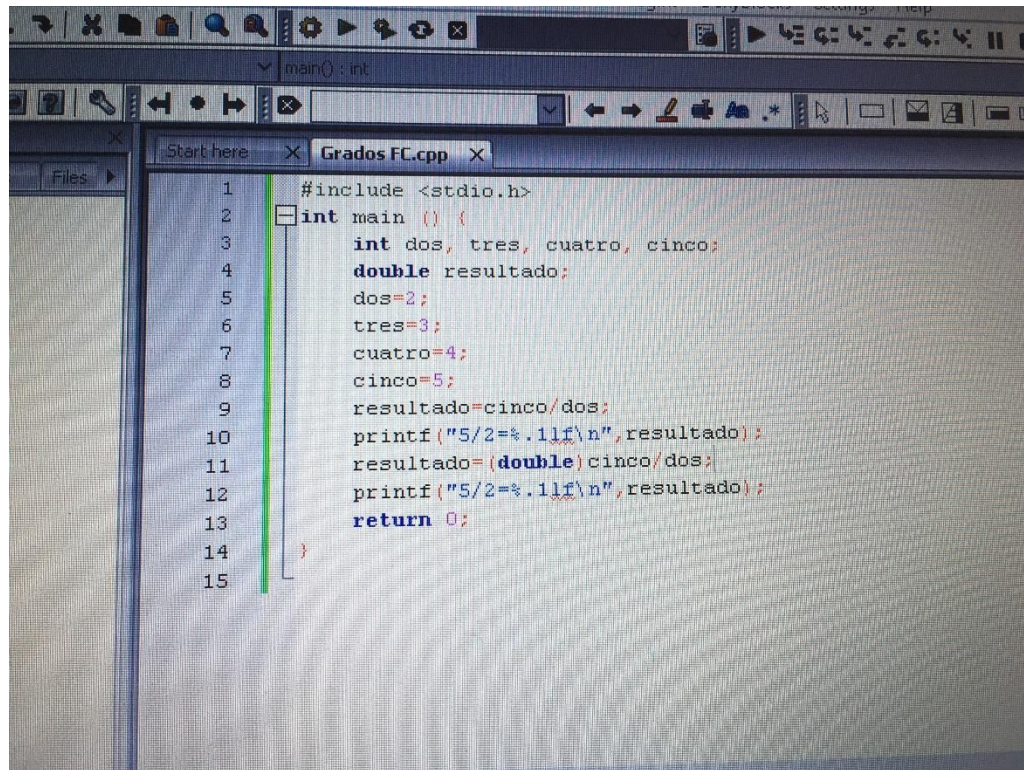
```
fp03alu41 ~ -bash — 80x24
Libano06:~ fp03alu41$ ./main2
Libano06:~ fp03alu41$ cat main2.c
int main () {
    // Variables enteras
    short numeroEntero1;
    signed int numeroEntero2;
    unsigned long numeroEntero3;
    // caracter
    char caracter;
    // Variables reales
    float puntoFlotante1;
    double puntoFlotante2;
    return 0;
}
Libano06:~ fp03alu41$ gcc main2.c -o main2
Libano06:~ fp03alu41$ ./main2
Primero texto solo
Luego podemos poner un entero: 32768
Tambien podemos poner un caracter: B
Y un numero real: 89.80
34
45.3
Tu entero: 34
Tu real: 45.300
Libano06:~ fp03alu41$
```

2.-Es importante que al final de cada línea después de las llaves se ponga un punto y coma para que el programa pueda correr, También se cierran las llaves siempre, si queremos poner dígitos decimales, se coloca el símbolo % seguido de un punto y el número de decimales después del punto que queremos.

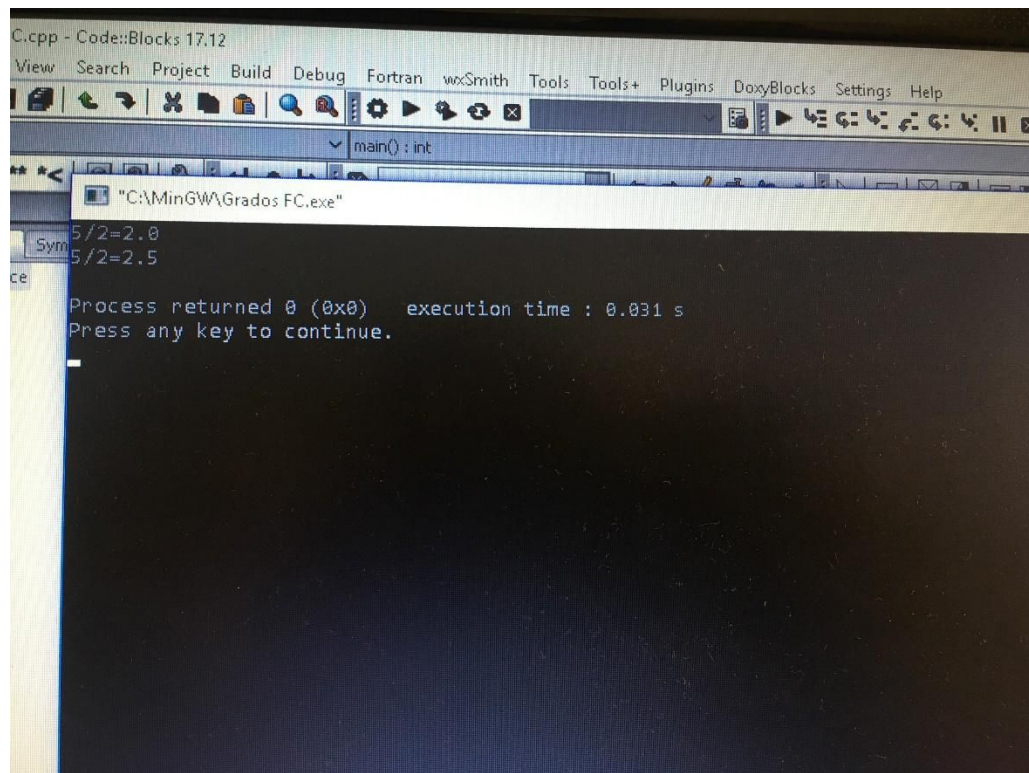
Al momento de compilar y correr Podemos probar utilizando cualquier número entero, real o una letra y el programa leerá de que tipo de variable estas metiendo.



### 3.-Operadores: el programa muestra el número que sale al hacer una division.



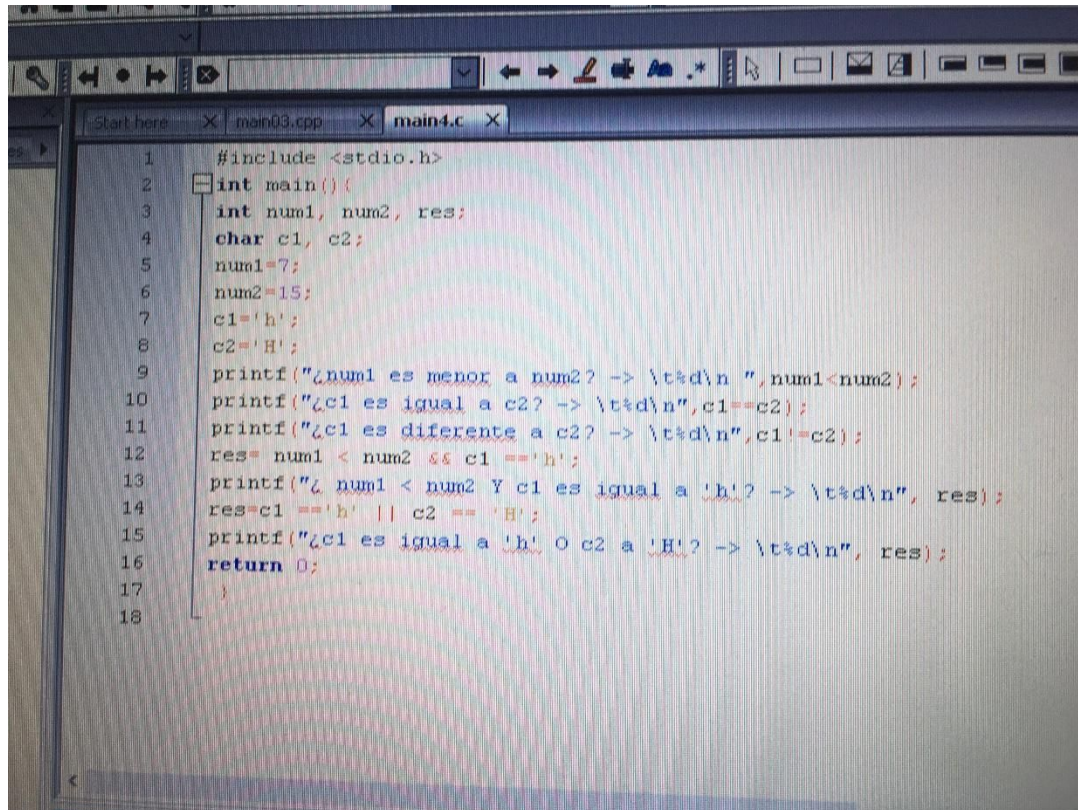
```
1  #include <stdio.h>
2  int main () {
3      int dos, tres, cuatro, cinco;
4      double resultado;
5      dos=2;
6      tres=3;
7      cuatro=4;
8      cinco=5;
9      resultado=cinco/dos;
10     printf("5/2=%.1lf\n", resultado);
11     resultado=(double)cinco/dos;
12     printf("5/2=%.1lf\n",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



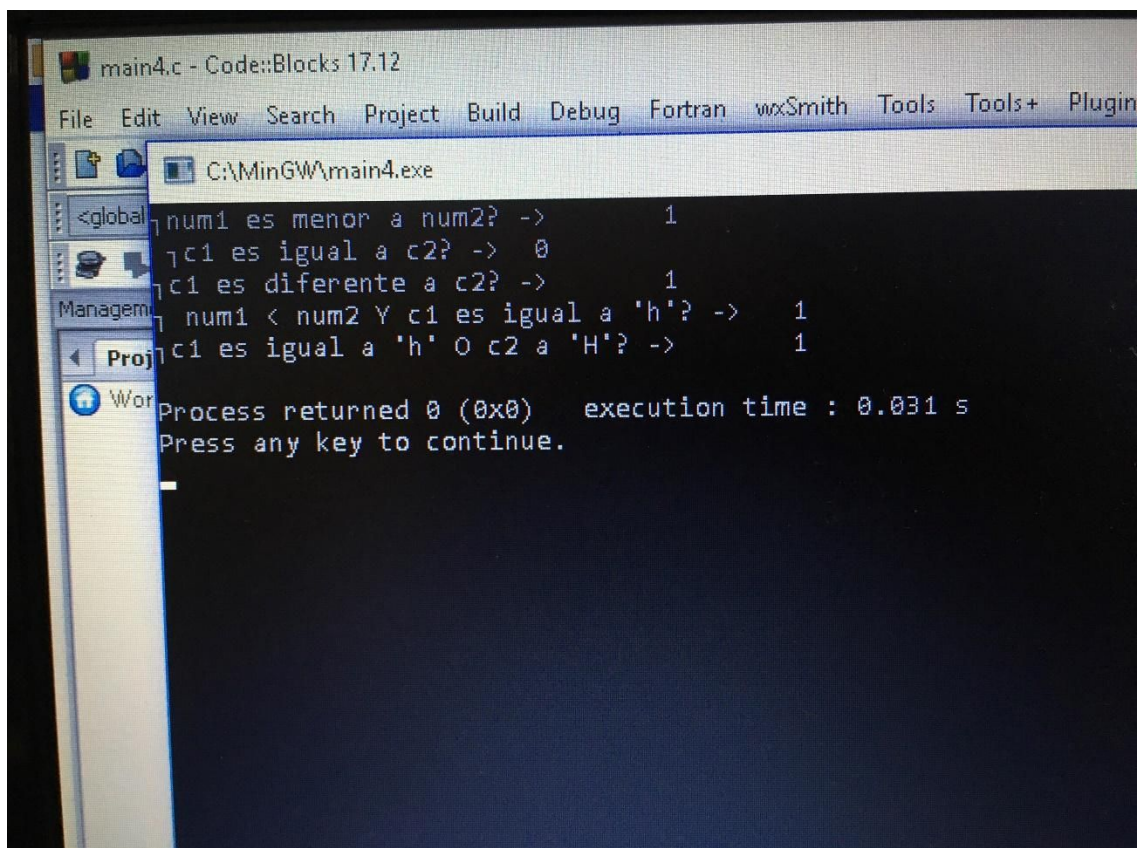
```
C.cpp - Code::Blocks 17.12
View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
main(): int
"C:\MinGW\Grados FC.exe"
5/2=2.0
5/2=2.5
Process returned 0 (0x0)   execution time : 0.031 s
Press any key to continue.
```



#### 4.- Operadores lógicos: el programa muestra si un número es mayor que otro y si un caracter es igual o diferente a otro.



```
1 #include <stdio.h>
2 int main(){
3     int num1, num2, res;
4     char c1, c2;
5     num1=7;
6     num2=15;
7     c1='h';
8     c2='H';
9     printf("\num1 es menor a num2? -> \t%d\n", num1<num2);
10    printf("\nc1 es igual a c2? -> \t%d\n", c1==c2);
11    printf("\nc1 es diferente a c2? -> \t%d\n", c1!=c2);
12    res= num1 < num2 && c1 == 'h';
13    printf("\num1 < num2 Y c1 es igual a 'h'? -> \t%d\n", res);
14    res=c1 == 'h' || c2 == 'H';
15    printf("\nc1 es igual a 'h' O c2 a 'H'? -> \t%d\n", res);
16    return 0;
17 }
18
```



```
main4.c - Code::Blocks 17.12
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugin
C:\MinGW\main4.exe
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 'h' O c2 a 'H'? -> 1
Process returned 0 (0x0) execution time : 0.031 s
Press any key to continue.
```

[https://www.google.com/search?q=tipos+de+variables+en+c&rlz=1C1CHBF\\_esMX868MX868&sx=srf=ACYBGNTCTIaZF9Rbm6WTP\\_wPzQdoA9Xa4g:1570399516318&source=lnms&tbm=isch&s=a=X&ved=0ahUKEwi1hc2K0ojAhVPmK0KHUC-CQYQ\\_AUIEigB&biw=1366&bih=625#imgsrc=v-g\\_RaKkRMR83jM:](https://www.google.com/search?q=tipos+de+variables+en+c&rlz=1C1CHBF_esMX868MX868&sx=srf=ACYBGNTCTIaZF9Rbm6WTP_wPzQdoA9Xa4g:1570399516318&source=lnms&tbm=isch&s=a=X&ved=0ahUKEwi1hc2K0ojAhVPmK0KHUC-CQYQ_AUIEigB&biw=1366&bih=625#imgsrc=v-g_RaKkRMR83jM:)