



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: M.C. ALEJANDRO ESTEBAN PIMENTEL ALARCON

Asignatura: FUNDAMENTOS DE PROGRAMACION

Grupo: 3

No de Práctica(s): PRACTICA 7

Integrante(s): 1

*No. de Equipo de
cómputo empleado:* Equipo 3

No. de Lista o Brigada: 420054913

Semestre: 2020-1

Fecha de entrega: 30 DE SEPTIEMBRE 2019

Observaciones:

CALIFICACIÓN: _____

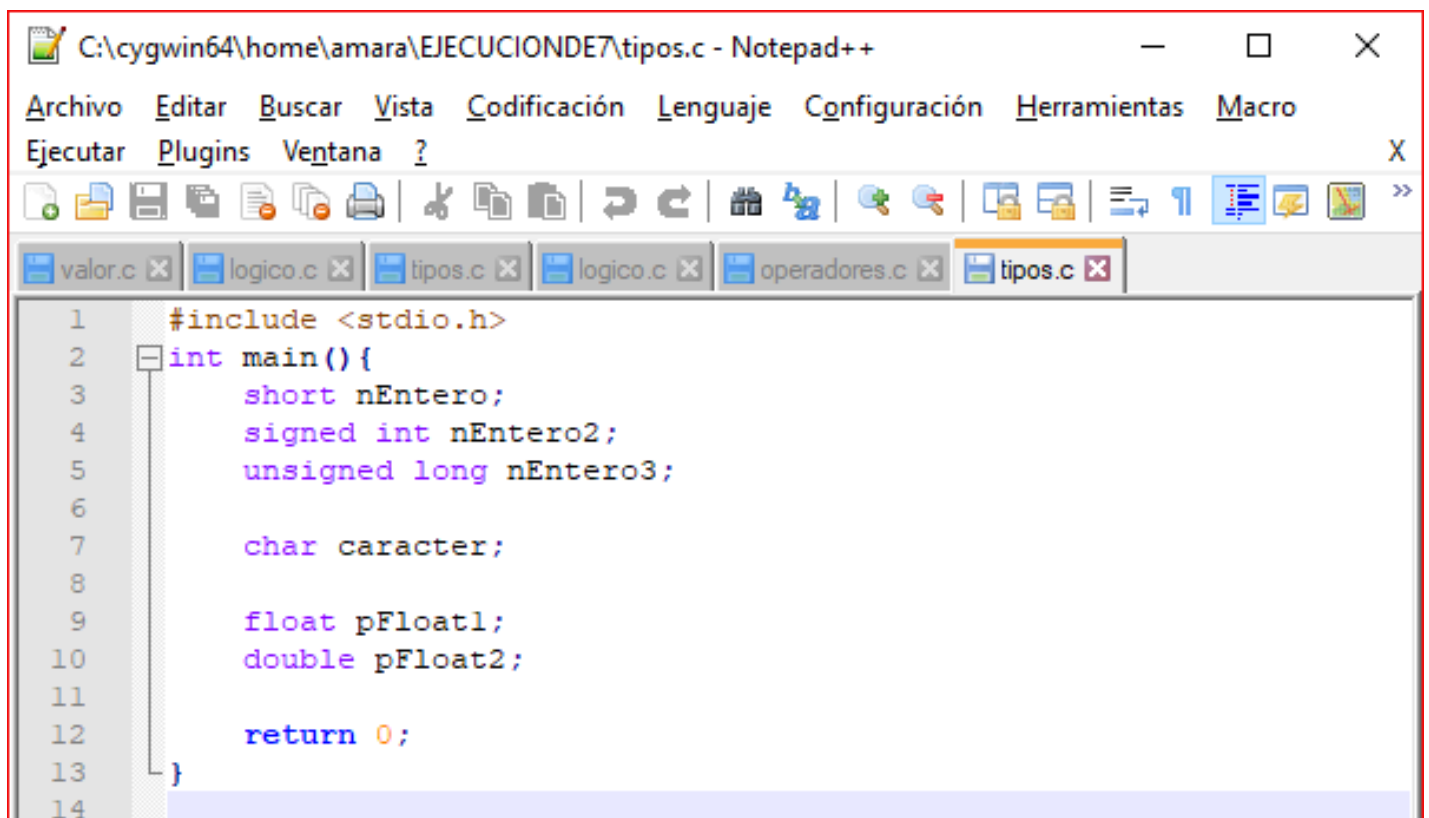
SE BUSCARA COMPRENDER EL FUNCIONAMIENTO DE TODO LO APRENDIDO EN CLASE COMO LOS TIPOS DE VARIABLES, OPERADORES, COMPARACIONES Y OPERADORES LÓGICOS; SOBRE COMO FUNCIONAN LOS PROGRAMAS PONIENDOLOS EN ESTA OCASION EN FUNCIONAMIENTO, ENTENDER LA SINTAXIS DE CODIGO QUE USA C, EN GENERAL O HA GRANDES RASGOS.

Objetivo

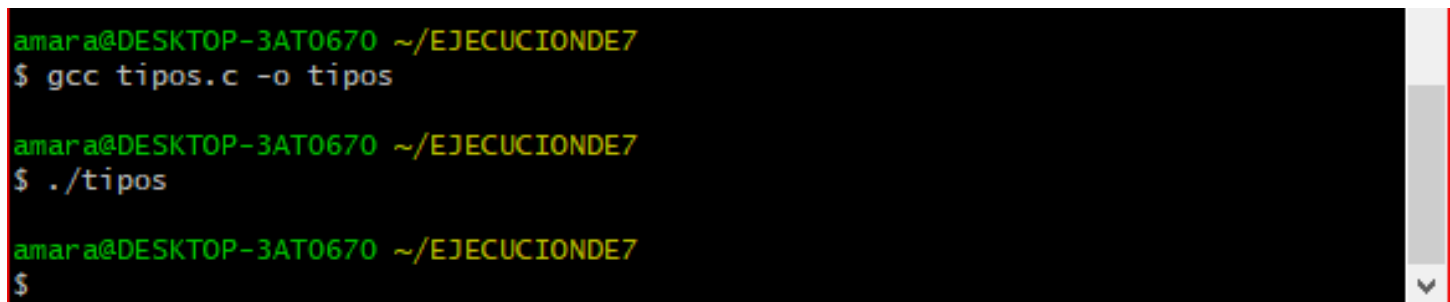
Elaborar programas en lenguaje C utilizando las instrucciones de control de tipo secuencia, para realizar la declaración de variables de diferentes tipos de datos, así como efectuar llamadas a funciones externas de entrada y salida para asignar y mostrar valores de variables y expresiones.

Mostrar y leer.

En este código lo que se ve es la declaración de variables de diferentes tipo, como es programa no regresa nada cuando este se ejecuta no muestra nada en la pantalla.



```
C:\cygwin64\home\amara\EJECUCIONDE7\tipos.c - Notepad++
Archivo  Editar  Buscar  Vista  Codificación  Lenguaje  Configuración  Herramientas  Macro
Ejecutar  Plugins  Ventana  ?
valor.c x logico.c x tipos.c x logico.c x operadores.c x tipos.c x
1  #include <stdio.h>
2  int main() {
3      short nEntero;
4      signed int nEntero2;
5      unsigned long nEntero3;
6
7      char character;
8
9      float pFloat1;
10     double pFloat2;
11
12     return 0;
13 }
14
```



```
amara@DESKTOP-3AT0670 ~/EJECUCIONDE7
$ gcc tipos.c -o tipos

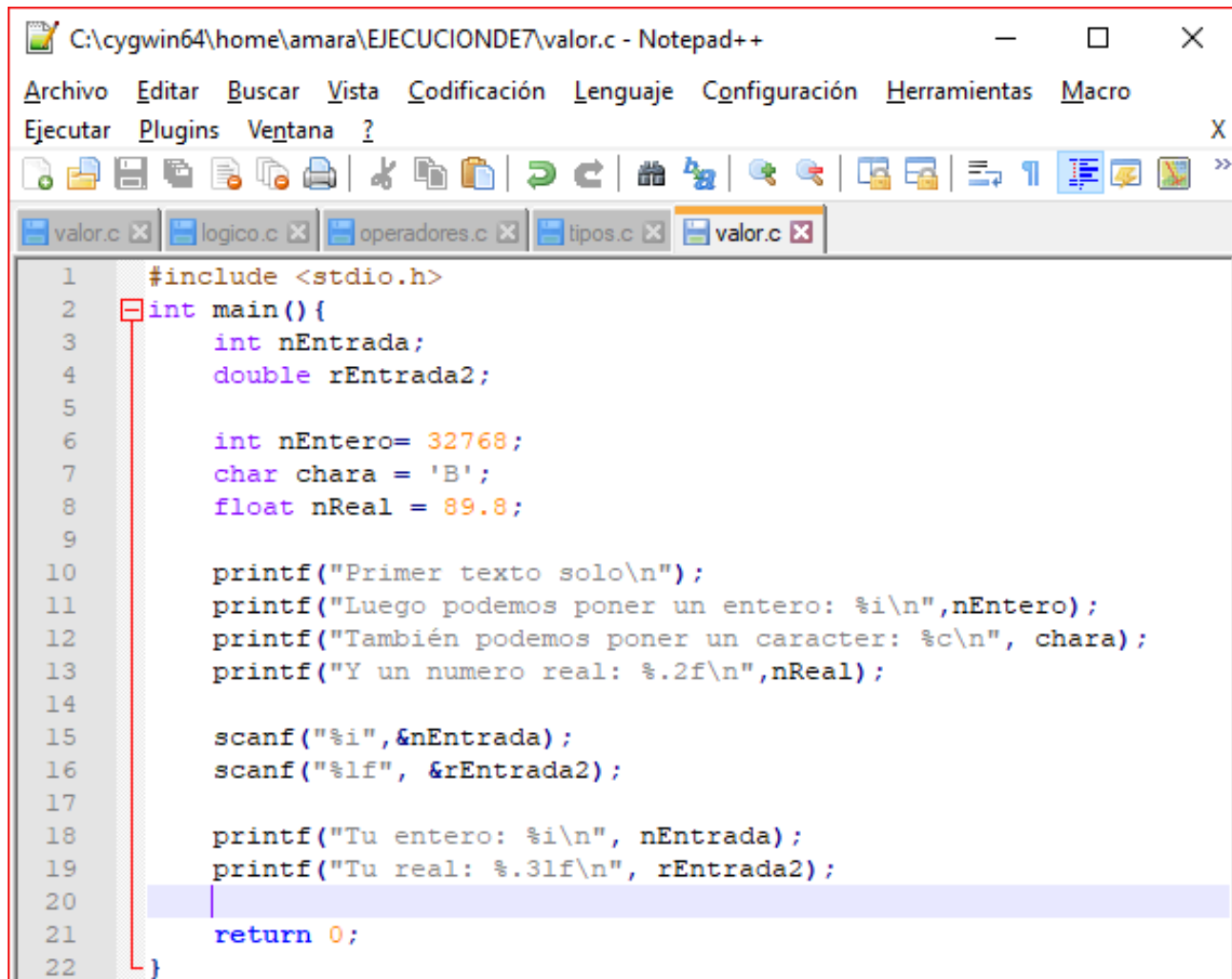
amara@DESKTOP-3AT0670 ~/EJECUCIONDE7
$ ./tipos

amara@DESKTOP-3AT0670 ~/EJECUCIONDE7
$
```

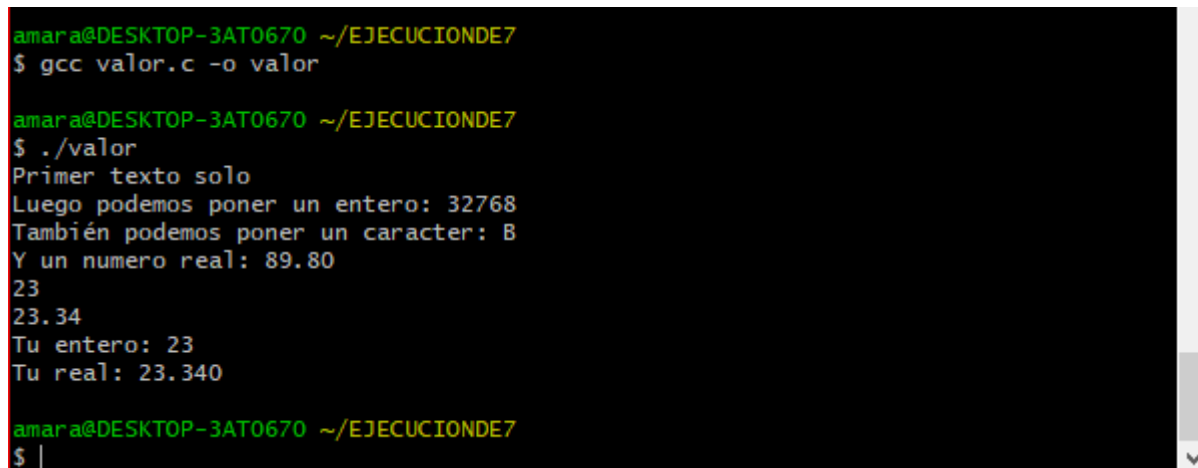
Tipos de variable

En este código se muestran 2 cosas fundamentales, que es el llenado de un variable y el modo de mostrar la información.

El uso del printf es para mostrar texto y en este caso, es el texto con el valor de alguna variable. También se pone en uso el scanf que lo que hace es pedir los datos al usuario desde consola para que después con un printf se muestre lo que se puso.



```
1  #include <stdio.h>
2  int main() {
3      int nEntrada;
4      double rEntrada2;
5
6      int nEntero= 32768;
7      char chara = 'B';
8      float nReal = 89.8;
9
10     printf("Primer texto solo\n");
11     printf("Luego podemos poner un entero: %i\n",nEntero);
12     printf("También podemos poner un caracter: %c\n", chara);
13     printf("Y un numero real: %.2f\n",nReal);
14
15     scanf("%i",&nEntrada);
16     scanf("%lf", &rEntrada2);
17
18     printf("Tu entero: %i\n", nEntrada);
19     printf("Tu real: %.3lf\n", rEntrada2);
20
21     return 0;
22 }
```



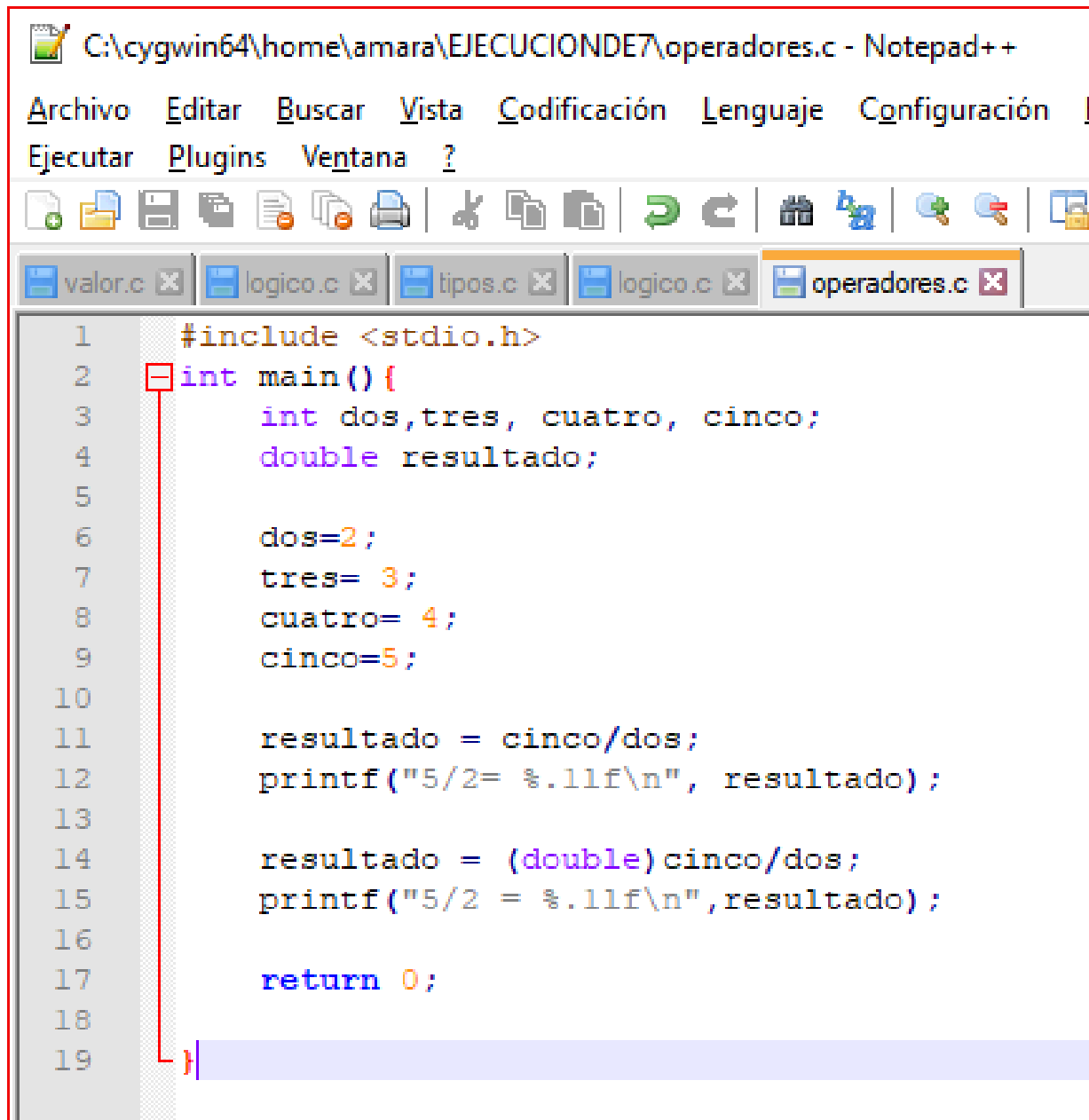
```
amara@DESKTOP-3AT0670 ~/EJECUCIONDE7
$ gcc valor.c -o valor

amara@DESKTOP-3AT0670 ~/EJECUCIONDE7
$ ./valor
Primer texto solo
Luego podemos poner un entero: 32768
También podemos poner un caracter: B
Y un numero real: 89.80
23
23.34
Tu entero: 23
Tu real: 23.340

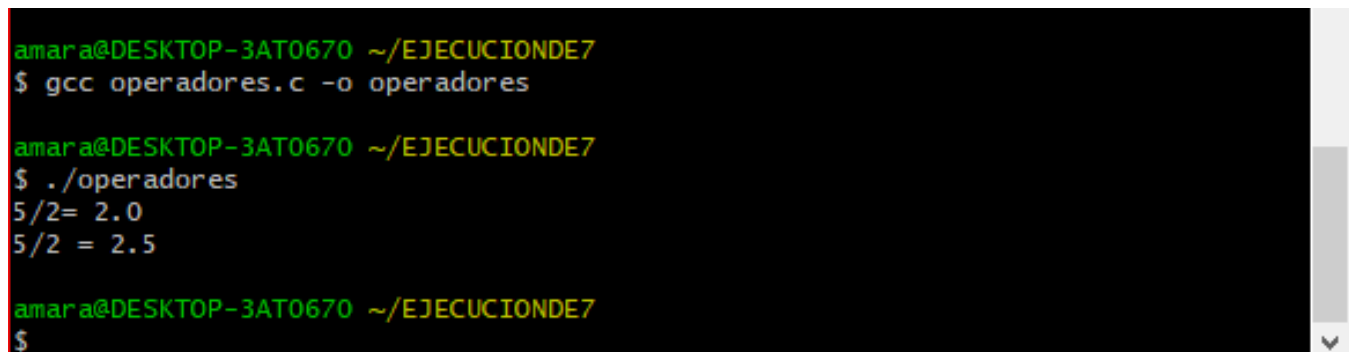
amara@DESKTOP-3AT0670 ~/EJECUCIONDE7
$ |
```

Operadores

En este código se pone en aplicación el hacer operaciones simples, introduciéndola en una nueva variable para un mejor uso, esta nueva variable que contiene el valor de la operación se une con el printf para mostrarla(regresar el valor a la hora de ejecutar el programa). Se observa de 2 maneras uno usando números enteros y otra con un tipo de dato double (con decimales), observando cómo se puede cambiar de un tipo de dato a otro sin alterar las variables originales.



```
1  #include <stdio.h>
2  int main(){
3      int dos,tres, cuatro, cinco;
4      double resultado;
5
6      dos=2;
7      tres= 3;
8      cuatro= 4;
9      cinco=5;
10
11     resultado = cinco/dos;
12     printf("5/2= %.11f\n", resultado);
13
14     resultado = (double)cinco/dos;
15     printf("5/2 = %.11f\n",resultado);
16
17     return 0;
18
19 }
```



```
amar a@DESKTOP-3AT0670 ~/EJECUCIONDE7
$ gcc operadores.c -o operadores

amar a@DESKTOP-3AT0670 ~/EJECUCIONDE7
$ ./operadores
5/2= 2.0
5/2 = 2.5

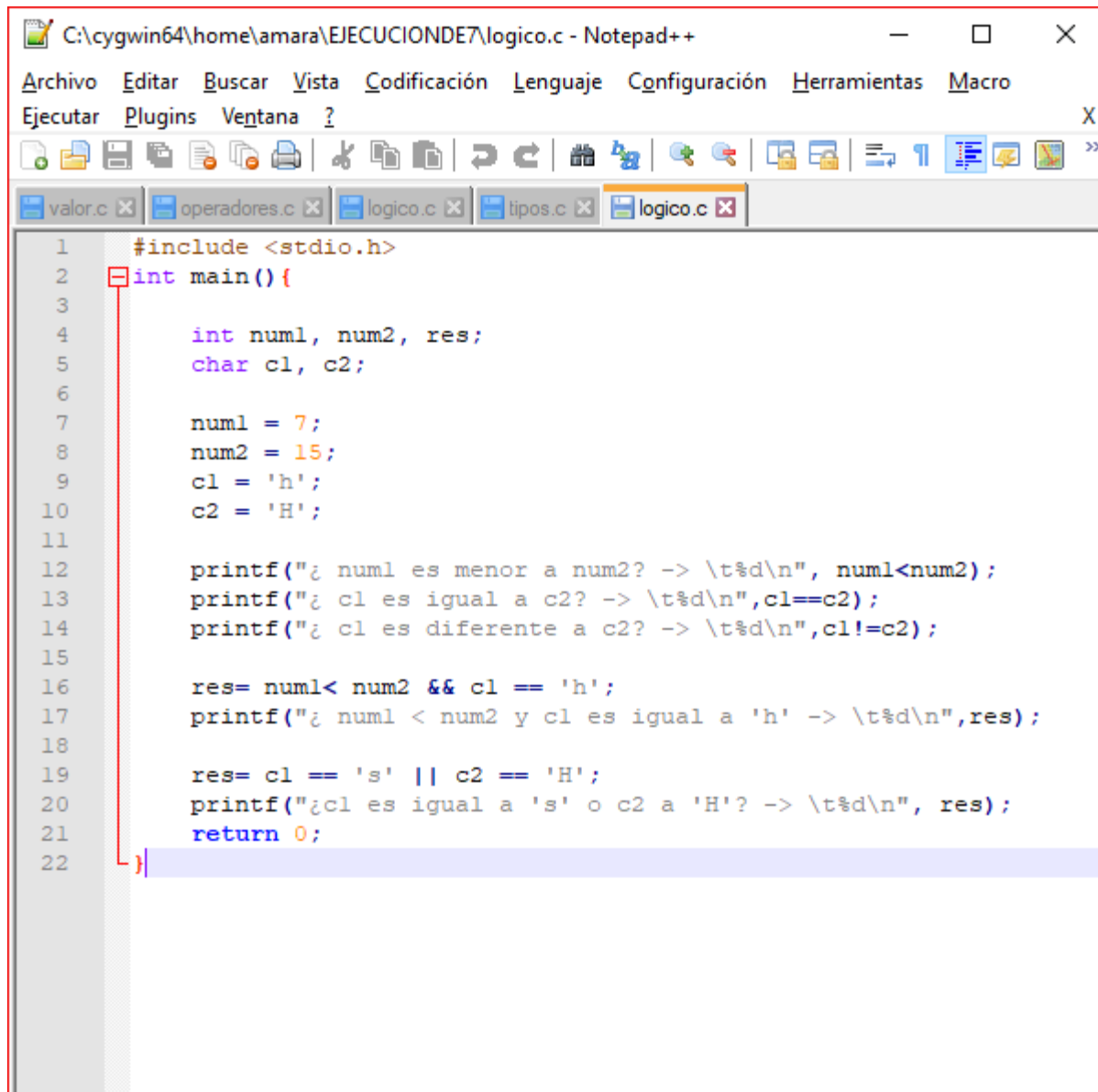
amar a@DESKTOP-3AT0670 ~/EJECUCIONDE7
$
```

Operadores lógicos

En este ejemplo usando datos tipo char y int se hacen operaciones usando operadores lógicos como bien lo son la comparación y lo que es AND y OR.

En este caso la condición se encuentra dentro de un printf sin usar una variable extra para poner la condición, por lo cual el valor sale junto con el texto (Esto también funciona sin se pone la condición dentro de una nueva variable).

Todos estos casos son tipo verdadero o falso, cuando la condición da verdadero al ejecutar en la terminal se mostrara como un 1 y cuando este es falso como un 0.



```
1  #include <stdio.h>
2  int main() {
3
4      int num1, num2, res;
5      char c1, c2;
6
7      num1 = 7;
8      num2 = 15;
9      c1 = 'h';
10     c2 = 'H';
11
12     printf("¿ num1 es menor a num2? -> \t%d\n", num1<num2);
13     printf("¿ c1 es igual a c2? -> \t%d\n", c1==c2);
14     printf("¿ c1 es diferente a c2? -> \t%d\n", c1!=c2);
15
16     res= num1< num2 && c1 == 'h';
17     printf("¿ num1 < num2 y c1 es igual a 'h' -> \t%d\n", res);
18
19     res= c1 == 's' || c2 == 'H';
20     printf("¿c1 es igual a 's' o c2 a 'H'? -> \t%d\n", res);
21     return 0;
22 }
```

```
~/EJECUCIONDE7
$ cd EJECUCIONDE7

amara@DESKTOP-3AT0670 ~/EJECUCIONDE7
$ gcc logico.c -o logico
gcc: error: logico: No such file or directory
gcc: error: no se reconoce la opción de línea de órdenes '-o'

amara@DESKTOP-3AT0670 ~/EJECUCIONDE7
$ ls
logico.c operadores.c tipos.c valor.c

amara@DESKTOP-3AT0670 ~/EJECUCIONDE7
$ gcc logico.c -o logico

amara@DESKTOP-3AT0670 ~/EJECUCIONDE7
$ ./logico
¿ 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

amara@DESKTOP-3AT0670 ~/EJECUCIONDE7
$
```

CON LA EJECUCIÓN DE ESTOS EJERCICIOS SE PUEDE VER MÁS CLARAMENTE LA SINTAXIS BASE QUE USA C EN SUS PROGRAMAS, ESTO AYUDA PARA LA COMPRENSIÓN GENERAL DEL PROGRAMA, Y EL CÓMO OTROS HECHOS ANTERIORMENTE, SE PUEDEN PASAR A TEXTO COMPLETAMENTE PARA SU COMPROBACIÓN Y EJECUCIÓN. A PARTE ESTOS EJERCICIOS AYUDAN EN EL SENTIDO QUE SE PUEDE VER, COMO SE VERÁN PROGRAMAS MÁS COMPLEJOS EN PANTALLA PUESTO, ESTOS SIENDO MÁS GENERALES, PERMITE APRECIAR CÓMO SE VEN LOS PROGRAMAS YA FUNCIONANDO EN LA TERMINAL.