	<b>Fundamentos de Lenguaje C</b>	
Facultad de Ingeniería	Laboratorio de docencia	

Laboratorios de computación  
salas A y B

*Profesor:* Alejandro Pimentel

*Asignatura:* Fundamentos de Programación

*Grupo:* 135

*No de Práctica(s):* 7

*Integrante(s):* Lorena Basurto Amezcua

*No. de Equipo de  
cómputo empleado:*

*No. de Lista o* 2858  
*Brigada:*

*Semestre:* 2020-1

*Fecha de entrega:* Octubre 03, 2019.

*Observaciones:*

CALIFICACIÓN: \_\_\_\_\_

## Objetivo:

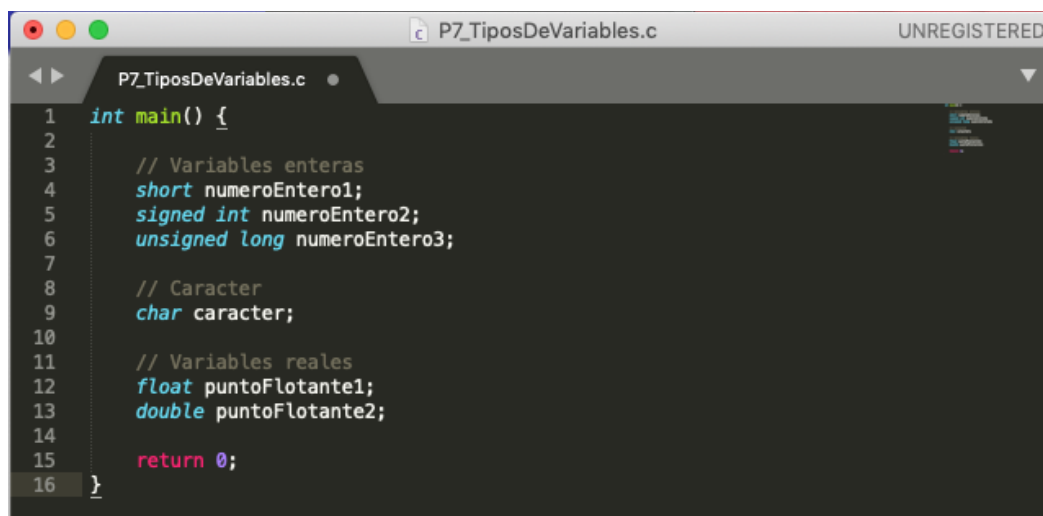
Elaborar programas en lenguaje C utilizando las instrucciones de control de tipo secuencia, para realizar la declaración de variables de diferentes tipos de datos, así como efectuar llamadas a funciones externas de entrada y salida para asignar y mostrar valores de variables y expresiones.

## Desarrollo:

- **Tipos de variables**

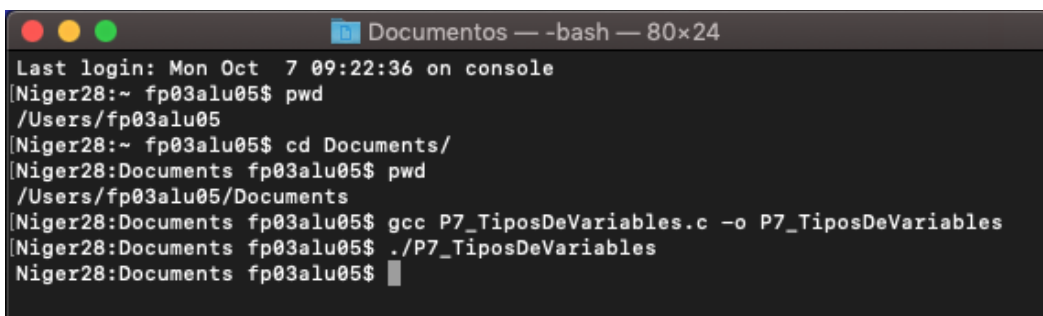
Para los reales, se tienen también diferentes tipos de variables que asignan más bits para tener mayor rango y mayor precisión. Las variables reales siempre poseen signo.

<i>Tipo</i>	<i>Bits</i>	<i>Valor Mínimo</i>	<i>Valor Máximo</i>
<i>float</i>	32	3.4 E-38	3.4 E38
<i>double</i>	64	1.7 E-308	1.7 E308
<i>long double</i>	80	3.4 E-4932	3.4 E4932



```
P7_TiposDeVariables.c
UNREGISTERED

P7_TiposDeVariables.c
1 int main() {
2
3     // Variables enteras
4     short numeroEntero1;
5     signed int numeroEntero2;
6     unsigned long numeroEntero3;
7
8     // Caracter
9     char caracter;
10
11    // Variables reales
12    float puntoFlotante1;
13    double puntoFlotante2;
14
15    return 0;
16 }
```



```
Documentos — -bash — 80x24
Last login: Mon Oct 7 09:22:36 on console
[Niger28:~ fp03alu05$ pwd
/Users/fp03alu05
[Niger28:~ fp03alu05$ cd Documents/
[Niger28:Documents fp03alu05$ pwd
/Users/fp03alu05/Documents
[Niger28:Documents fp03alu05$ gcc P7_TiposDeVariables.c -o P7_TiposDeVariables
[Niger28:Documents fp03alu05$ ./P7_TiposDeVariables
[Niger28:Documents fp03alu05$ ]
```

- Mostrar y Leer

<i>Tipo de dato</i>	<i>Especificador de formato</i>
<i>Entero</i>	%d, %i, %ld, %li, %o, %x
<i>Flotante</i>	%f, %lf, %e, %g
<i>Carácter</i>	%c, %d, %i, %o, %x
<i>Cadena de caracteres</i>	%s

```

1  #include <stdio.h>
2
3  int main () {
4
5      // Declaramos variables a leer
6      int numeroEntrada;
7      double realEntrada;
8
9      // Asignamos variables
10     int numeroEntero = 32768;
11     char caracter = 'B';
12     float numeroReal = 89.8;
13
14     // Mostramos texto y valores
15     printf("Primero texto solo\n");
16     printf("Luego podemos poner un entero: %i\n", numeroEntero);
17     printf("También podemos poner un caracter: %c\n", caracter);
18     printf("Y un número real: %.2f\n", numeroReal);
19
20     // Leemos valores
21     scanf("%i", &numeroEntrada);
22     scanf("%lf", &realEntrada);
23
24     // Y ahora podemos mostrarlos también
25     printf("Tu entero: %i\n", numeroEntrada);
26     printf("Tu real: %.3lf\n", realEntrada);
27
28     return 0
29 }

```

```

Documents — -bash — 80x24
Last login: Mon Oct 7 09:40:14 on ttys000
[Niger28:~ fp03alu05$ pwd
/Users/fp03alu05
[Niger28:~ fp03alu05$ cd Documents/
[Niger28:Documents fp03alu05$ gcc P7_MostrarYLeer.c -o P7_MostrarYLeer
[Niger28:Documents fp03alu05$ ./P7_MostrarYLeer
Primero texto solo
Luego podemos poner un entero: 32768
También podemos poner un caracter: B
Y un número real: 89.80
7
8.45
Tu entero: 7
Tu real: 8.450
Niger28:Documents fp03alu05$

```

- Operadores

Operador	Operación	Uso	Resultado
+	Suma	125.78 + 62.5	188.28
-	Resta	65.3 - 32.33	32.97
*	Multiplicación	8.27 * 7	57.75
/	División	15 / 4	3.75
%	Módulo	4 % 2	0

```

P7_Operadores.c
UNREGISTERED

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

```

```

Documents — -bash — 80x24

Last login: Mon Oct 7 09:49:50 on ttys000
[Niger28:~ fp03alu05$ pwd
/Users/fp03alu05
[Niger28:~ fp03alu05$ cd Documents/
[Niger28:Documents fp03alu05$ pwd
/Users/fp03alu05/Documents
[Niger28:Documents fp03alu05$ gcc P7_Operadores.c -o P7_Operadores
[Niger28:Documents fp03alu05$ ./P7_Operadores
5 / 2 = 2.0
5 / 2 = 2.5
Niger28:Documents fp03alu05$

```

- Comparaciones

<i>Operador</i>	<i>Operación</i>	<i>Uso</i>	<i>Resultado</i>
==	Igual que	'h' == 'H'	Falso
!=	Diferente a	'a' != 'b'	Verdadero
<	Menor que	7 < 15	Verdadero
>	Mayor que	11 > 22	Falso
<=	Menor o igual	15 <= 22	Verdadero
>=	Mayor o igual	20 >= 35	Falso

- Operadores lógicos

<i>Operador</i>	<i>Operación</i>
!	No
&&	Y
	O

```

P7_OperadoresLógicos.c  UNREGISTERED
P7_OperadoresLógicos.c
1  #include <stdio.h>
2
3  int main() {
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? -> %d\n", num1 < num2);
13     printf("¿c1 es igual a c2? -> %d\n", c1 == c2);
14     printf("¿c1 es diferente a c2? -> %d\n", c1 != c2);
15
16     res = num1 < num2 && c1 == 'h';
17     printf("¿num1 < num2 Y c1 es igual a 'h'? -> %d\n", res);
18
19     res = c1 == 's' || c2 == 'H';
20     printf("¿c1 es igual a 's' O c2 a 'H' -> %d\n", res);
21
22     return 0;
23 }
  
```

Line 22, Column 14      Tab Size: 4      C

```
Documents — -bash — 80x24
Last login: Mon Oct  7 09:56:01 on ttys000
[Niger28:~ fp03alu05$ cd Documents/
[Niger28:Documents fp03alu05$ gcc P7_OperadoresLógicos.c -o OperadoresLógicos
[Niger28:Documents fp03alu05$ ./OperadoresLógicos
¿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
Niger28:Documents fp03alu05$
```

## Conclusión:

As escribir código en lenguaje C se debe cuidar el uso estricto de la sintaxis correcta para que al compilar y correr no exitan errores y el programa haga lo que nosotros deseamos. También, al compilar, es importante estar ubicado en la carpeta donde se encuentra el archivo *nombre.c*; de lo contrario no se encontrará el archivo y no se podrá compilar ni correr.