

Carátula para entrega de prácticas	
Facultad de Ingeniería	Laboratorio de docencia

Laboratorios de computación salas A y B

<i>Profesor:</i>	Alejandro Pimentel
<i>Asignatura:</i>	Fundamentos de programación
<i>Grupo:</i>	3
<i>No de Práctica(s):</i>	7
<i>Integrante(s):</i>	Uno Karin Natalia
<i>No. de Equipo de cómputo empleado:</i>	26
<i>No. de Lista o Brigada:</i>	8723 #50
<i>Semestre:</i>	1
<i>Fecha de entrega:</i>	30 de septiembre del 2019
<i>Observaciones:</i>	

CALIFICACIÓN: _____

PRÁCTICA 7

FUNDAMENTOS DE LENGUAJE C

OBJETIVO

Se elaboraran programas con lenguaje C utilizando instrucciones para declarar variables, así ejecutando entradas y salidas, mostrar los valores de las variables.

INTRODUCCIÓN

El lenguaje de programación en C, es un lenguaje conocido como de alto nivel. Una de las características del lenguaje de programación en C, es que es un lenguaje estructurado, lo que permite generar código claro y sencillo, ya que esta basado en la modularidad.

El lenguaje de programación en C, esta estructurado en tres partes fundamentales, las cuales son, una librería estándar, un programa compilador y un preprocesador.

La librería estándar en el lenguaje de programación C, no es más que librerías realizadas en código objeto y puede haber sido realizada en otro lenguaje diferente que el C. Estas librerías se deben colocar en un programa de lenguaje programación en C, en la instrucción conocida como INCLUDE.

El programa compilador en el lenguaje de programación en C, es el que tiene como función traducir las instrucciones del programa fuente en C a el lenguaje conocido por las computadoras u ordenadores, el llamado lenguaje maquina. El programa compilador, depura y detecta los posibles errores en el lenguaje fuente, y es capaz de notificar el error ocurrido al programador, mediante un mensaje de texto.

En el lenguaje de programación en C, el preprocesador es un componente perteneciente propiamente al lenguaje C, el cual transforma el programa fuente traduciendo cada instrucción del programa fuente, de la siguiente forma: Elimina los comentarios colocados por el programador, incluye en el programa fuente el contenido de los archivos que se encuentran declarados en el INCLUDE, a estos archivos se le suele llamar cabeceras, y por ultimo, sustituye los valores de las constantes declaradas en el define.

Recopilado en. <https://www.lenguajes-de-programacion.com/programacion-en-c.shtml>

Una variable en C es un espacio que reservamos en memoria para poder guardar información, las variables se utilizan a menudo para guardar números caracteres, entre otros tipos de datos, no solo en C si no en cualquier lenguaje de programación. En C cada variable (espacio reservado en memoria) puede utilizarse para guardar información, y dicha información se representa por medio de un tipo de dato específico, que determina como la interpretará el lenguaje en sí.

Recopilado en: <http://codigoprogramacion.com/cursos/tutoriales-c/variables-y-tipos-de-datos->

enc.html#targetText=Variables%20y%20tipos%20de%20datos,en%20cualquier%20lenguaje%20de%20programaci%C3%B3n.

Los tipos de variables utilizados en C son:

DATA TYPE	MEMORY (BYTES)	RANGE
short int	2	-32,768 to 32,767
unsigned short int	2	0 to 65,535
unsigned int	4	0 to 4,294,967,295
int	4	-2,147,483,648 to 2,147,483,647
long int	4	-2,147,483,648 to 2,147,483,647
unsigned long int	4	0 to 4,294,967,295
long long int	8	$-(2^{63})$ to $(2^{63})-1$
unsigned long long int	8	0 to 18,446,744,073,709,551,615

En el caso de los reales son:

<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

<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

Operadores utilizados en lenguaje C

<i>Operador</i>	<i>Operación</i>	<i>Uso</i>	<i>Resultado</i>
+	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

<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

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

DESARROLLO/RESULTADOS

En las siguientes imágenes se muestran ejemplos de programas de textos planos en donde se determinan variables utilizando el lenguaje en C, incluyendo entradas y salidas.

PRACT7.c UNREGISTERED

PRACT7.c

```
1 int main() {
2
3     // Variables enteras
4     short numero Entero1;
5     signed int numeroEntero2;
6     unsigned long numeroEntero3;
7
8     // Carácter
9     char caracter;
10
11    // Variables reales
12    float puntoFlotante1;
13    double puntoFlotante2;
14
15    return 0;
16 }
```

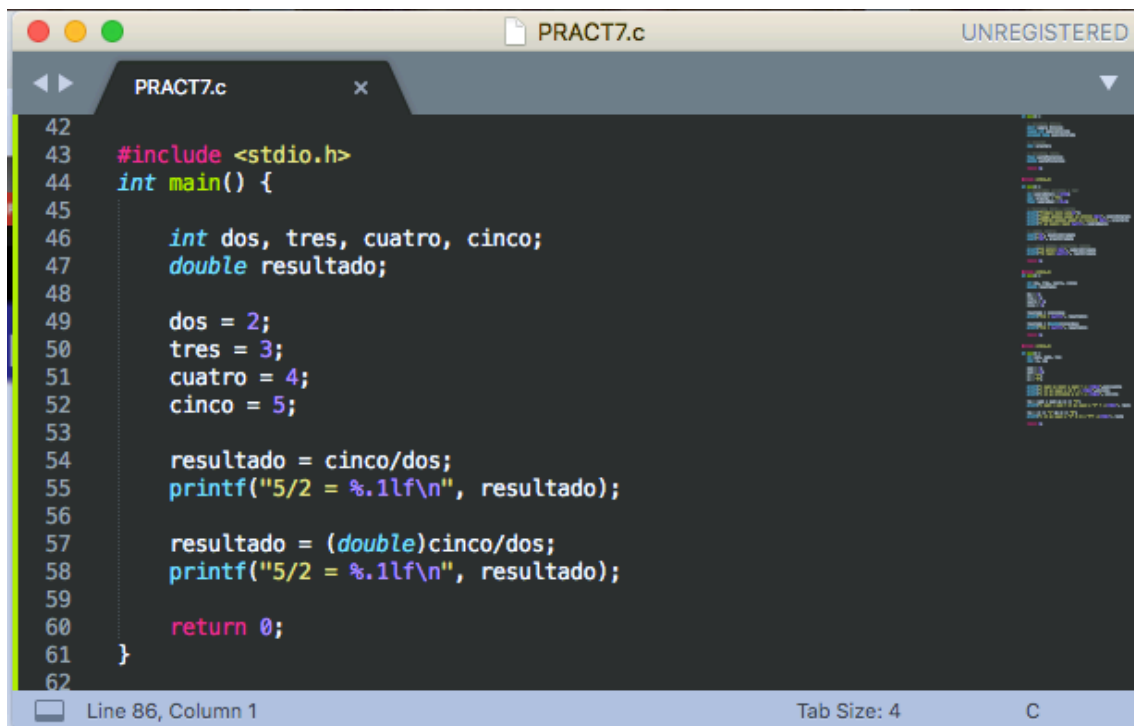
Line 86, Column 1 Tab Size: 4 C

PRACT7.c UNREGISTERED

PRACT7.c

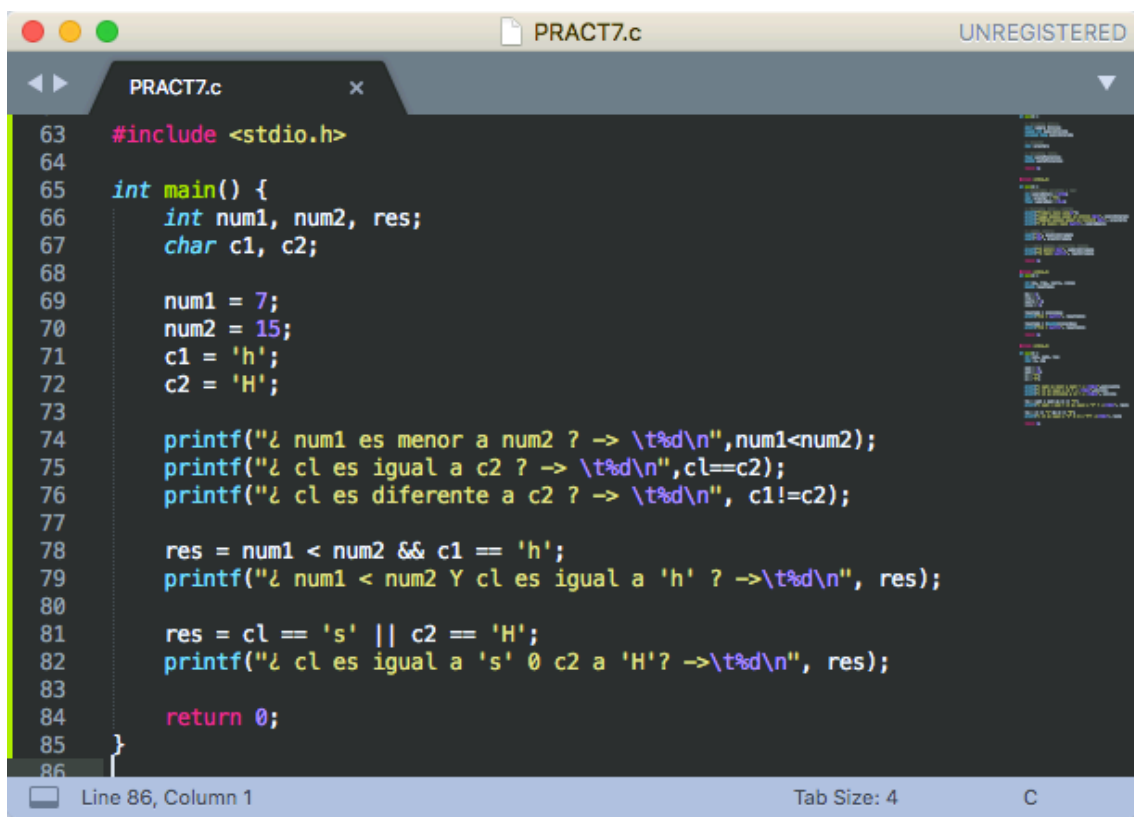
```
17
18 #include <stdio.h>
19
20 int main() {
21     // Declaramos variables a leer
22     int numeroEntero = 32768;
23     char caracter = 'B';
24     float numeroReal = 89.8;
25
26     // Mostramos texto y valores
27     printf("Primero texto solo\n");
28     printf("Luego podemos poner un entero: %i\n", numeroEntero);
29     printf("También podemos poner un carácter: %c\n", caracter);
30     printf("Y un número real: %.2f\n", numeroReal);
31
32     // Leemos valores
33     scanf("%i", &numeroEntrada);
34     scanf("%lf", &realEntrada);
35
36     // Y ahora podemos mostrarlos también
37     printf("Tu entero: %i\n", numeroEntrada);
38     printf("Tu real: %.31f\n", realEntrada);
39
40     return 0;
41 }
```

Line 86, Column 1 Tab Size: 4 C



```
42
43 #include <stdio.h>
44 int main() {
45     int dos, tres, cuatro, cinco;
46     double resultado;
47
48     dos = 2;
49     tres = 3;
50     cuatro = 4;
51     cinco = 5;
52
53     resultado = cinco/dos;
54     printf("5/2 = %.1lf\n", resultado);
55
56     resultado = (double)cinco/dos;
57     printf("5/2 = %.1lf\n", resultado);
58
59     return 0;
60 }
61
62
```

Line 86, Column 1 Tab Size: 4 C



```
63 #include <stdio.h>
64
65 int main() {
66     int num1, num2, res;
67     char c1, c2;
68
69     num1 = 7;
70     num2 = 15;
71     c1 = 'h';
72     c2 = 'H';
73
74     printf("\n num1 es menor a num2 ? -> \t%d\n", num1<num2);
75     printf("\n c1 es igual a c2 ? -> \t%d\n", c1==c2);
76     printf("\n c1 es diferente a c2 ? -> \t%d\n", c1!=c2);
77
78     res = num1 < num2 && c1 == 'h';
79     printf("\n num1 < num2 Y c1 es igual a 'h' ? ->\t%d\n", res);
80
81     res = c1 == 's' || c2 == 'H';
82     printf("\n c1 es igual a 's' 0 c2 a 'H'? ->\t%d\n", res);
83
84     return 0;
85 }
86
```

Line 86, Column 1 Tab Size: 4 C

CONCLUSIÓN

Para concluir, el lenguaje de programación en C , comúnmente utilizado en textos planos como “Sublime Text” o “Notepad++”, son muy útiles para realizar programas sin formato. En dichos programas para ejecutar un proceso, cálculo

o solución de un problema, se utilizan operadores, variables, entradas y salidas.