



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: 35

No de Práctica(s): 7

Integrante(s): Páez Martínez Karen

*No. de Equipo de
cómputo empleado:* 2

No. de Lista o Brigada: 3781

Semestre: 1

Fecha de entrega: 02/10/2019

Observaciones:

CALIFICACIÓN: _____

PRACTICA#7

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.

Actividad #1:

- Primero descargamos sublime text ,despues de esto con la ayuda de la presentación del profesor nos vamos guiando en que tipos de variables se van a utilizar y como estas trabajan en este programa.

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

```
1 #Include <stdio.h>
2 int main () {
3     int nEntrada;
4     doble nEntrada2;
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("Tambien podemos poner un caracter: %c\n",chara);
13    printf("Y un numero real: %.2f\n",numeroReal);
14
15    scanf("%i",&nEntrada);
16    scanf("%if",&nEntrada2);
17
18    printf("Tu entero: %i\n",nEntrada);
19    printf("Tu real: %3lf\n",nEntrada2);
20
21    return 0;
22 }
23 }
```

no command for selector: noop:

Line 5, Column 1 Tab Size: 4 C

```

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

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

amar@DESKTOP-3AT0670 ~/EJECUCIONDE7
$

```

```

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

amar@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

amar@DESKTOP-3AT0670 ~/EJECUCIONDE7
$ |

```

- b) En esta parte seguimos realizando los ejercicios que el profesor dejo en la practica, utilizando las diferentes variables.

<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
*	Multipliación	8.27 * 7	57.75
/	División	15 / 4	3.75
%	Módulo	4 % 2	0

```
File Edit Selection Find View Goto Tools Project Preferences Help
# stdio.h x
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 }
20
```

Line 19, Column 2 Tab Size: 4 C++

```
ACER@EQUIPO /cygdrive/c/Users/ACER/Documents
$ gcc operador.c -o operador

ACER@EQUIPO /cygdrive/c/Users/ACER/Documents
$ ./operador
5 / 2 = 2.000000
5 / 2 = 2.500000

ACER@EQUIPO /cygdrive/c/Users/ACER/Documents
$
```

c) En este paso seguí las demás instrucciones faltantes , y con las variables.

<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

Operador Operación

!

No

&&

Y

||

O

```

File Edit Selection Find View Goto Tools Project Preferences Help
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? -> %td\n",num1<num2);
13    printf("¿c1 es igual a c2? -> %td\n",c1==c2);
14    printf("¿c1 es diferente a c2? -> %td\n",c1!=c2);
15
16    res= num1 < num2 && c1=='h';
17    printf("¿num1< num2 Y c1 es igual a 'h'? -> %td\n",res);
18
19    res= c1 == 's' || c2 == 'H';
20    printf("¿c1 es igual a 's' 0 c2 a 'H'? -> %td\n",res);
21
22    return 0;
23 }
24

```

```

$ 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
$

```

Conclusion: En esta parte cabe destacar que me gusto mucho aprender las diferentes variables y es que todas estas practicas han sido nuevas para mi pero se ve como programar en C no es tan dificil como aparenta y apesar que tengo dificultades con las capturas de pantalla, los programas si corrieron.