

**Trabajo Practico N°1  
Resuelto**

***Laboratorio de Computación I  
Tecnicatura Superior en Programación.  
UTN-FRRQ***

## CONSTANTES, VARIABLES Y OPERADORES

1- Dada la sigte línea: `# define VALOR 15` marcar en cada caso la/s respuestas correctas:

VALOR es:

- a. Constante
- b. Variable inicializada
- c. Constante de carácter
- d. Constante simbólica
- e. Ninguna de las anteriores

Cuál o cuales de las sptes. sentencias son válidas?

- a. `VALOR += i;`
- b. `printf ("%s", VALOR);`
- c. `++ VALOR;`
- d. `printf ("%d", VALOR);`
- e. ninguna de las anteriores

2- Definir la constante simbólica DOLAR dándole el valor 13.18.

```
#define DOLAR 13.18
```

.....

3- Habiendo definido la constante simbólica DOLAR, esta ¿puede cambiar su valor en tiempo de ejecución?

**SI - NO**

4- Decir si las siguientes expresiones son válidas:

- a. `#define PI 3.141592, NUMERO_E 2.718281`
  - a. NO ES VALIDA. Un `#define` para cada constante
- b. `#define CATEDRA "\"Sintaxis y Semántica de los Lenguajes\""`
  - a. NO ES VALIDA. `#define CATEDRA "\"Sintaxis y Semántica de los Lenguajes/\""`

5- Dada la siguiente directiva al preprocesador

```
#define CIUDAD "Rosario"
```

si escribimos:

```
printf( "Hola usted está en: %s", CIUDAD );
```

por pantalla se verá:

Respuesta: Hola usted esta en Rosario

6- Decir cuales de los siguientes identificadores son válidos, si no lo son decir porque:

a) registro1	d) return	g) nom y dir	j) 123-45-6
b) 1registro	e) \$impuesto	h) nom_y_dir	k) nom*ap
c) archivo_3	f) nombre	i) nom-y-dir	l) num1_art

7- Supongamos que la versión de C puede reconocer sólo los ocho primeros caracteres del nombre de un identificador, aunque los nombres de identificadores pueden ser de una longitud arbitraria. ¿Cuáles de los siguientes pares de nombres de identificadores se consideran como idénticos y cuáles se distinguirán?

a) nombre, nombres	c) identificador_1, identificador_2	e) respuesta, RESPUESTA
b) direccion, direcciones	d) lista1, lista2	f) num1, num_1

8- Determinar cuales de los sgtes. valores numéricos son constantes válidas. Si es una constante válida especificar si es entera o real.

a) 0.5	c) 9.3e12	e) 12345678
b) 27,822	d) 9.3e-12	f) 127.09

9- Determinar cuales de las sgtes. son constantes de carácter válidas.

a) 'a'	c) '\n'	e) '\\'	g) 'T'	i) 'xyz'
b) 'S'	d) '/n'	f) '\a'	h) '\0'	j) "a"

10- Determinar cuales de las sgtes. son constantes de cadena de caracteres válidas.

- '8:15 P.M.'
- "rojo, blanco y negro"
- "Nombre:"
- "2000 ROSARIO"
- "El profesor dijo: \\"Por favor, presten atención\\""

'a' = Carácter

"a" = Cadena

11- Escribir las declaraciones apropiadas y asignar los valores iniciales dados para cada grupo de variables

- Variables de coma flotante: a = -8.2, b = 0.005
  - float a = -8.5;
  - float b = 0.005;
- Variables enteras: x = 129, y = 87, z = -22
  - int x = 129, y = 87, z = -22;
- Variables de carácter: c1 = 'w', c2 = '\*'
  - char c1 = 'w', c2 = '\*';

- 12- Dentro de una función, si 2 variables, una local y una global, tienen el mismo nombre:
- La global prevalece sobre la local
    - NO
  - La local prevalece sobre la global
    - SI
  - Se producirá un error
    - NO

13- Detectar y justificar los errores en cada uno de los siguientes programas:

a. 

```
#include <stdio.h>
main()
{
    /* Aquí va el cuerpo del programa */
    Printf( "Hola mundo\n" );
}
```

b. 

```
#include <stdio.h>

main()
{
    int numero;
    numero = 2;
    printf( "El valor es %i", Numero );
}
```

c. 

```
#include <stdio.h>

main()
{
    int a, c;
    a = 5;
    c += a +5;
    printf ("c = %d \n", c);
}
```

Inicializar c = 0, o hacer C = a + 5

d. 

```
#include <stdio.h>

main()
{
    clrscr();
    gotoxy( 10, 10 );
    printf( "Estoy en la fila 10 columna 10" );
    return 0;
}
```

Clrscr() y gotoxy() no son soportadas por CodeBlocks

14-Cuál es el resultado de cada una de las siguientes expresiones

a)  $!(3 > 3 \ \&\& \ 5 != 4 \ || \ 9 = 9)$  \_\_\_\_\_ 0 \_\_\_\_\_

b)  $!(7 > 8) \ \&\& \ 7 > 8 \ || \ 7 > 8$  \_\_\_\_\_ 0 \_\_\_\_\_

c)  $13/13 * 2 * 5 + 5 \% 2$  \_\_\_\_\_ 11 \_\_\_\_\_

d)  $x = (y = 3, y + 1)$  \_\_\_\_\_ 4 \_\_\_\_\_

15- Dadas las sgtes declaraciones y asignaciones: **int a = 2, b =3, c = 4**; evaluar las sgtes. expresiones y determinar que quedará almacenado en **result** en cada caso.

1. `result = a * b % c;`  
a. 2
2. `result = 3 * (a -= 1) + 1 % a;`  
a. 3
3. `result = (a > b) || (b == c);`  
a. False

17- Que resultado se obtiene de evaluar los siguientes segmentos de programa:

a. `main()`

```

{
    int a = 3, b = 5, c = 4;
    if (a < b > c) printf("Correcto");
    else printf("Incorrecto");
}

```

b. `main()`

```

{ int a = 10, b = 3;
  (a % b == 0) ? printf("Azul\n"): printf("Rojo\n");
}

```

c. `main()`

```

{
    int a = 10, b = 5;
    char c;
    c = (a % b == 0) ? 'A' : 'B';
    printf("%c", c);
}

```

d. `main()`

```

{
    int a = 10;
    float c, d, b = 4;
    c = b / a;
    d = (int)b / a;
    printf(" %f \n ", c); 0.4
    printf(" %f \n ", d); 0.000000
    printf(" %g \n ", d); 0
    printf(" %g \n ", c); 4
}

```

```

e. main()
{
  char c; int i; float f;
  c = i = f = 60 + 5.8;
  printf( " %c %d %f \n", c, i, f);
}

```

A 65 65.800000

```

f. main()
{
  char c; int i; float f;
  c = i = f = 'A' + 5;
  printf( " %c %d %f ", c, i, f);
}

```

F 70 70.000000

```

g. main()
{
  char c; int i; float f;
  c = i = f = 'A' + 5.9;
  printf( " %c %d %f \n", c, i, f);
}

```

F 70 70.900000

```

h. main()
{
  char c; int i; float f;
  c = i = f = 'A' + 3.3 + 5;
  printf( " %c %d %f ", c, i, f);
}

```

I 73 73.300000

```

i. main( )
{
  int a=1, b = 5;
  printf( "%d\t", a == 2);
  printf( "%d\n", 5 / 2 * 3 + 8 % 5 + b);
}

```

0 (tab horizontal)

14

```

j. main( )
{
  int a=1, b = 5;
  printf( "%d\t", a == 1);
  printf( "%d\n", b + 7 / 3 * 3 + 7 % 5 );
}

```

1

13

```

k. main()
{
  int a, b, c;
  a = 5;
  b = ++a;
  c = ( a + 5 * 2 ) * ( b + 6 / 2 ) + ( a * 2 );
  printf( "%i, %i, %i", a, b, c);
}

```

6, 6, 156

- 18- ¿Qué valor contendrán las variables **s** y **c** luego de ejecutarse la secuencia de asignaciones que se mencionan en cada caso?
- a)  $c = 0;$   
 $s = (++c) + 1;$
- b)  $c = 0;$   
 $s = c ++ + 1;$
- c)  $c = 4;$   
 $c* = 2;$
- $c = 1$   
 $s = 2$
- $c = 1$   
 $s = 1$
- $c = 8$
- 19- ¿Qué valor queda en cada variable involucrada, después de ejecutar el enunciado que se indica, suponiendo que antes de ello todas las variables contienen el valor 5?
- a)  $product * = x++;$  b)  $result = ++x + x;$   
 $product = 25$   $x = 6$   
 $x = 5$   $result = 12$
- 20- Completar las líneas de punto
- a) El operador.....?..... permite realizar una operación si una condición es verdadera y otra si es falsa .
- b) El operador.....%..... devuelve el resto de la división entera.
- c) Las variables definidas fuera de toda función se denominan.....Globales.....
- d) Toda sentencia termina en.....;.....
- e) Un programa escrito en lenguaje C esta íntegramente formado por .....funciones.....
- f) Todo programa en C comienza su ejecución en la función.....main.....
- 22- Responder a cada una de las siguientes afirmaciones con V (verdadero) o F (falso) según corresponda. Justificar.
- a. El operador % puede ser utilizado con cualquier tipo de operandos.  
.....F (solo int).....
- b. En la expresión  $a = ('4' > 4)$  la variable **a** asume el valor 1.  
.....V.....'4' = 52.....
- c. La expresión  $(x > y || a < b)$  es verdadera ya sea que  $(x > y)$  es verdadero o  $(a < b)$  es verdadero  
.....V.....
- d. Una variable local puede ser definida en cualquier lugar dentro de una función.  
.....F (Solo al principio de la función).....
- 23- Escribir 2 sentencias diferentes para realizar:  
Dado un numero imprimir "n" si el número es  $< 0$  y "Z" si el número es  $> 0$ .
- `printf("%c", (nro < 0) ? 'n' : 'Z')`
- 24- Asignarle a la variable **var** el contenido de la variable **res** y luego disminuir **res** en 1(uno).  
**var = res--;**  
`var = res;`  
`ress = res - 1; => ress--`

25- Asignarle a la variable **p** un **1 si p es impar o un 0 si p es par**, esto en el caso de que **p sea mayor que 20**, sino asignarle un **6**.

```
(p<20) ? (p%2 == 0 ? 0 : 1) : 6;
```

```
(p<20) ? (p%2) : 6;
```

```
If()
```

```
{
```

```
    If(p%d == 0)
```

```
        Printf("%d",0)
```

```
    Else
```

```
        Printf("%d",0)
```

```
}
```

```
Else
```

```
{
```

```
    Printf("%d",6)
```

```
}
```