

TRABAJO PRÁCTICO N° 2

Ejercicios 1

Fecha de entrega:

Forma de entrega: Documento digital en PDF, que incluya las consignas, los datos del alumno, webgrafía y licencia Creative Commons, con permisos de modificación e impresión, cuyo nombre debe ser “Nombre_Apellido-TP2.pdf” enviado a consultas@profmatiasgarcia.com.ar el cual debe tener como asunto “Estructurada TM - tu Nombre y Apellido - TP 2”.

Ayuda: En caso de no comprender alguna consigna o tener dudas, puede solicitarse asistencia enviando un email a consultas@profmatiasgarcia.com.ar con el asunto “Estructurada TM Nombre y Apellido TP2 CONSULTA”.

Modalidad: Individual

Ejercicio 1

¿Qué definición de constantes es la correcta en lenguaje C?

- a) `int N=25;`
- b) `int N 25;`
- c) `const N 25;`
- d) `const int N=25;`
- e) `const int N 25;`
- f) `#define N=25;`

Ejercicio 2

La expresión que es falsa si y sólo si un valor real x está en el intervalo $[3.0,4.0]$ (ambos inclusive) es:

- a) `(x<3.0) && (x>4.0)`
- b) `! (x<3.0) || ! (x>4.0)`
- c) `(x>=3.0) || (x<=4.0)`
- d) `! (x<=3.0) && ! (x>=4.0)`
- e) `(x<3.0) || (x>4.0)`

Ejercicio 3

Un número entero se dice elemental si es divisible por 3 ó 5 pero no lo es por 4. Una expresión lógica que devuelva cierto si un número entero i es elemental es:

- a) `(i % 3 == 0 || i % 5 == 0) && (i % 4 != 0)`
- b) `(i % 3 == 0 || i % 5 == 0) && ! (i % 4 != 0)`
- c) `(i % 3 == 0 || i % 5 == 0) && (i % 4 == 0)`
- d) `(i % 3 == 0 || i % 5 == 0) || ! (i % 4 != 0)`
- e) `(i % 3 == 0 || i % 5 == 0) || (i % 4 != 0)`

Ejercicio 4

La expresión que es falsa si y sólo si un valor real x está en el intervalo abierto $(3.0,4.0)$

(sin incluir los extremos) es:

- a) `(x<=3.0) || (x>=4.0)`
- b) `(x<3.0) && (x>4.0)`
- c) `(x>=3.0 || x<=4.0)`
- d) `! (x<=3.0) && !(x>=4.0)`
- e) `(x<3.0) || (x>4.0)`

Ejercicio 5

Dado el siguiente programa:

```
#include <stdio.h>
int main()
```

```
{  
    int a=8.8;  
    printf("%d",a);  
    return 0;  
}
```

- a) Se ejecutará sin errores pero no imprimirá ningún valor concreto
- b) Al compilar da un error en la definición de la variable a
- c) Se ejecutará sin problemas e imprimirá un 8
- d) Al compilar dará un error en la línea de la sentencia printf
- e) No da error de compilación pero si un error al ejecutarlo

Ejercicio 6

Determina qué vale x después de acabar el bucle

```
int i=0, x=0;  
while (i<100)  
{  
    x=x+1;  
    i=i+2;  
}
```

- a) 49
- b) 50
- c) 51
- d) 100

Ejercicio 7

Determina qué vale x después de acabar el bucle

```
int i, x=0;  
for (i=0 ; i <100 ; i=i+2)  
    x=x+1;
```

- a) 49
- b) 50
- c) 51
- d) 100

Ejercicio 8

Determina qué vale x después de acabar el bucle

```
int i=100, x=0;  
while (i>0)  
{  
    x=x+1;  
    i=i-1;  
}
```

- a) 1
- b) 99
- c) 100
- e) 50

Ejercicio 9

Determina qué vale x después de acabar el bucle

```
int i, x;
for (i=100 ; i>0 ; i--)
{
    x=0;
    x=x+1;
}
```

- a) 1
- b) 99
- c) 100
- d) 50

Ejercicio 10

¿Cuántas iteraciones del bloque se realizan al ejecutar el siguiente bucle?:

```
for (i=0 ; i <10 ; i=i+2)
{
    for (j=0 ; j <10 ; j=j+2) { /* bloque*/ }
}
```

- a) 16
- b) 100
- c) 50
- d) 25

Ejercicio 11

¿Cuántas iteraciones del bloque se realizan al ejecutar el siguiente bucle?:

```
for (i=0 ; i <10 ; i++)
{
    for (j=0 ; j <i ; j++) { /* bloque*/ }
}
```

- a) 45
- b) 36
- c) 28
- d) 100

Ejercicio 12

Dado el siguiente trozo de código:

```
int i=0, suma=0;
for (i=3; i<10; i++)
    suma=suma+i;
```

¿Cuánto vale la variable suma tras la ejecución del bucle?

- a) 42
- b) 55
- c) 0
- d) No es posible saberlo
- e) 45

Ejercicio 13

Sea el siguiente trozo de código:

```
...
suma=0;
for (i=0; i<10; i++)
    if (i%2!=0)
        suma = suma+i;
printf ("%d", suma);
...
```

El resultado impreso del mismo es el siguiente:

- a) No se puede ejecutar el programa porque hay un error de compilación
- b) 0
- c) El programa no termina
- d) 25
- e) 50

Ejercicio 14

Sea el siguiente trozo de código:

```
...
suma=0;
for (i=0; i<10; i++)
    if (i%2==0)
        printf ("%d", i/2);
    else
        suma = suma+i;
printf ("%d", suma);
...
```

El resultado impreso del mismo es el siguiente:

- a) No se puede ejecutar el programa porque hay un error de compilación
- b) 0
- c) El programa no termina
- d) 0123425
- e) 50

Ejercicio 15

Sea el siguiente trozo de código:

```
int i,j;
suma=0;
for(i=1; i<=10; i++)
    if (i%2==0)
        j = i/2;
    else
        j = i;

suma = suma + j;
printf ("%d", suma);
```

El resultado impreso del mismo es el siguiente:

- a) No se puede ejecutar el programa porque hay un error de compilación
- b) 40
- c) El programa se queda bloqueado
- d) 5
- e) 0

Ejercicio 16

Escribe un programa que tome 3 números enteros introducidos por el usuario mediante el teclado y determine cuántos de dicho números son diferentes.

Ejercicio 17

Escribe un programa que tome 3 números enteros introducidos por el usuario mediante el teclado, determine cuál es el mayor, además de cuántos de dicho números son diferentes.

Ejercicio 18

Escribe un programa que tome un año introducido por el usuario y diga si es bisiesto o no. Recuerda que los años múltiplos de 4 son bisiestos, excepto aquellos que son múltiplos de 100 y no lo son de 400. Es decir el año 2000 si es bisiesto, pero no lo son el 1990 ni el 2100.

Ejercicio 19

Escribe un programa que calcule el factorial de n donde n es un número entero mayor o igual que cero dado por el usuario.

Ejercicio 20

Escribe un programa que escriba todos los divisores de un número n entero positivo introducido por el usuario.

Ejercicio 21

Escribe un programa que escriba la tabla de multiplicar de cualquier número entero entre 1 y 10, dado por el usuario.

Ejercicio 22

Escribe un programa que calcule la suma de todos los números múltiplos de 5 comprendidos entre dos enteros positivos leídos por teclado

Ejercicio 23

Escribe un programa que calcule el producto de dos números a y b enteros positivos dados por el usuario, sin usar el operador $*$.

Ejercicio 24

Escribe un programa que calcule la división entera entre dos números a y b enteros positivos dados por el usuario, sin usar el operador $/$.

Ejercicio 25

Escribe el programa para un juego que consista en averiguar un número. El programa nos indicará cada vez si el número introducido es mayor o menor que la constante almacenada que tendremos que averiguar. Cuando se descubre el número mostrara un mensaje de felicitación.