



Práctica 4: Funciones (paso por valor)

Hoja de trabajo PREVIÓ del estudiante en casa

Cuestión 1: Dado el siguiente trozo de código:

<pre>float z; int main (void) { float x; int y; return 0; }</pre>	<pre>int Func (int y, int x) { int z; return z; }</pre>
--	---

Indica cuántas variables distintas aparecen:

Indica el ámbito (global, local a la función main o local a la función Fun) de las siguientes variables:

Variable	Ámbito
La variable z de tipo float	
La variable x de tipo float	
La variable x de tipo int	
La variable z de tipo int	

Cuestión 2: Analiza el siguiente programa y modifícalo hasta que funcione correctamente.

```
#include <iostream>
```

```
using namespace std;
```

```
FncDivisible (                ); // línea a modificar
```

```
int main()
{
```

```
    int valor1, valor2;
    bool res;
```

```
    cout << "Introduce un valor: ";
    cin >> valor1;
    cout << "Introduce otro valor: ";
    cin >> valor2;
```

```
FncDivisible (                ); // línea a modificar
```

```
    if ( res == true )
        cout << valor1 << " es divisible por " << valor2 << endl;
    else
        cout << valor1 << " no es divisible por " << valor2 << endl;
```



```

    return 0;
}

// Averigua si valor es divisible por n.
FncDivisible (          ) // línea a modificar
{
    bool r = false;

    if (valor % n == 0)
        r = true;

    return r;
}

```

Cuestión 3: Realiza la traza del siguiente programa y averigua cuál es su salida por pantalla

```

#include <iostream>

using namespace std;

int miFuncion (int, int);

int main ()
{
    int z, x = 1, y = 10;          //1

    z = miFuncion (x, y);          //2

    cout << z << " " << x << " " << y << endl;    //3

    return 0;
}

int miFuncion (int x, int y)
{
    int r;

    x = x * 2;          //4
    y = y % 5;          //5
    r = x + y;          //6

    return r;
}

```

Indica el contenido de cada una de las variables en las líneas que se indican.

Los parámetros formales se denotan por x_1 , y_1 . El subíndice hace referencia al orden de la llamada.

Mensaje por pantalla:_____

Línea	x	y	z	x_1	y_1	r_1
(en la llamada)	1					
	2					
	4					
	5					
	6					
(después de la llamada)	2					
	3					



Cuestión 4: Para generar números aleatorios en C++ se usan las funciones `rand()` y `srand()` incluidas en la librerías `<cstdlib>` y `<ctime>`. Por ejemplo, para generar un número aleatorio en el intervalo $[0, N - 1]$:

```
srand (time (NULL)); // inicializa el generador de números aleatorios  
val = rand() % N;    // val es una variable uniformemente distribuida en [0, N-1]
```

Dado el siguiente programa:

```
#include <iostream>  
#include <cstdlib>  
#include <ctime>  
  
using namespace std;  
  
int main()  
{  
    short valor;  
  
    srand (time (NULL));  
  
    for (int i = 0; i <= 10; i++)  
    {  
        valor = rand () % (100 + 1);  
        cout << valor << endl;  
    }  
  
    return 0;  
}
```

Indica brevemente qué hace.

Comenta la línea de inicialización de la semilla y ejecuta el programa varias veces. ¿Qué ocurre?