



## 1 Comentarios

- Para usar las funciones de la librería matemática estándar hay que incluir el fichero `math.h` mediante la directiva `#include` y compilar con la opción `-lm`. Algunas de estas funciones son:

<code>ceil(x)</code>	<code>floor(x)</code>	<code>pow(x,y)</code>	<code>exp(x)</code>	<code>log(x)</code>	<code>log10(x)</code>	<code>fabs(x)</code>	<code>sin(x)</code>
$\lceil x \rceil$	$\lfloor x \rfloor$	$x^y$	$e^x$	$\ln x$	$\log_{10} x$	$ x $	$\text{seno}(x)$

Nótese que se pueden obtener logaritmos en cualquier base mediante la fórmula  $\log_b x = \log_b a \cdot \log_a x$ . No existe una función para redondear a enteros puesto que se puede usar de la misma manera el cambio de tipo, `int(x)`. No obstante esto se puede conseguir mediante la fórmula  $\lfloor x + 0.5 \rfloor$ .

- Cuando una función ha de devolver más de un valor, es normal devolverlos mediante el uso de argumentos de la función pasados por referencia.
- Para usar el generador de enteros aleatorios hay que incluir el fichero `stdlib.h`. La función `rand()` devuelve un entero aleatorio entre 0 y `MAX_RANDOM`. Estos valores se pueden modificar convenientemente con los operadores `+`, `*` y `%` para obtener enteros aleatorios en cualquier subrango.

En realidad, `rand` genera una secuencia (fija) de enteros pseudoaleatorios. Para que esta secuencia no sea siempre la misma, hay que inicializar este generador con “semillas” diferentes mediante `srand(int semilla)`. Si se quiere inicializar el generador de forma diferente sin necesidad de proporcionar explícitamente una semilla se puede hacer `srand(time(NULL))` ya que `time(NULL)` devuelve la hora actual en décimas de segundo.

## 2 Ejercicios

- a) Escribe una función que, dado un valor real, redondee usando únicamente la función `floor` de la librería estándar `math.h`. b) Modifica la función anterior para que devuelva el valor redondeado a la centésima. c) Modifica la función anterior para que dado el valor y un entero  $n$  devuelva este valor redondeado al  $n$ -ésimo decimal.

Usa todas estas funciones en un mismo programa y redondea los mismos números de diferentes formas. Escribe todos los valores con 6 decimales. Ejemplo: Valor? 2.010234 Resultado= 2.000000, 2.000000, 2.010000, 2.010000, 2.010200. En el ejemplo se ha redondeado con 0, 1, 2, 3 y 4 decimales.

Atención esto no coincide con modificar la precisión (`cout.precision`). Aquí se está cambiando el valor. Con `cout.precision`, se está escribiendo el mismo valor con diferente precisión.

- Escribe tu versión particular de la función `pow(x,y)` con funciones simples y que devuelva un resultado redondeado a milésimas.
- Escribe una función con cuatro argumentos que, dados dos reales  $x$  e  $y$ , calcule la división con dos decimales y el resto de esta división.

OPCIONAL: Generalizar la función anterior para  $n$  decimales.

- Usa el programa `grafica.cpp` para visualizar diferentes rangos de las siguientes funciones de la librería `math.h`: `ceil`, `floor`, `pow`, `exp`, `log`, `log10`. Visualiza también las funciones anteriores de redondeo así como la división y el resto. Sigue las instrucciones en el programa y las explicaciones del profesor.

Modifica el programa para que aparezca el valor correspondiente a  $f(x)$  a la derecha del símbolo '\*'.

OPCIONAL: Modifica el programa para que admita rangos con números negativos. Añadele cualquier otra funcionalidad que consideres oportuna.

MUY OPCIONAL: Extiende el programa para que se puedan visualizar 2 o más funciones a la vez.

5. Escribe un programa para aprender la tabla de multiplicar. Se generarán aleatoriamente dos enteros entre 2 y 9 y se preguntará ¿n x m?. Habrá entonces que introducir un valor y el ordenador tendrá que responder Bien o Mal. La pregunta se repetirá indefinidamente hasta que se conteste 0. Entre otras cosas, habrá que escribir una función `aleat(n,m)` que genere enteros aleatorios en el rango n..m.

Modifica el programa para que el programa responda Bien o Mal de cuatro o más formas diferentes de forma aleatoria. Ejemplo: ¿8 x 3? 20 ``Fatal`` ¿2x1? 2 ``Muy Bien`` ¿5 x 2? 10 ``De categoría`` etc.

6. Escribe un programa que lea *cualquier cosa* de la entrada estándar y sea capaz de encontrar enteros (separados por espacios, comas o cualquier otro carácter incluídos los no imprimibles) y calcular sus valores. Usa el carácter '/' como centinela para acabar. Habrá que escribir una función `siguiente_entero` que devuelva el siguiente entero encontrado. Esta función deberá tener un argumento de tipo bandera que será usado por el programa principal para detectar que se ha llegado al carácter '/'. Ejemplo: Entrada= ``aaQ90r,+12.3 <return> <tab>aaa000z/'``. Ante esta entrada el programa hará 5 llamadas a la función. Los resultados que se deba imprimir son 90, 12, 3 y 0.

OPCIONAL: Modificar el programa para que haga lo mismo pero con enteros o reales en formato fijo. Decide tú mismo el comportamiento del programa ante casos como ``12..3''`.

```
/* GRAFICA.CPP
 * Programa per a dibuixar grafiques
 * Autor: prof_lab
 */

#include <iostream.h>
#include <math.h>

const int max_pantalla=75; // columnes utils per dibuixar

double funcio ( double x ); // funcio a dibuixar
int conv_pant ( double val, double maxval ); // conversio a punts de pantalla
void dibuixa_eix_y( double maxval ); // dibuixa l'eix
void dibuixa_punt ( double x, double y, double maxval ); // dibuixa un punt

int main()
{
    double first,last;
    double step;
    double maxval;
    double w;

    cout << "Escriu primer i ultim valor de x\n";
    cin >>first>> last;

    step= (last - first) / 20.0; // increment de la variable x
    step= int (ceil (4.0 * step)) / 4.0; // redondeig a 0.25, 0.5, 0.75 o enters

    //decimals en els eixos
    cout.setf(ios::fixed);
    if (step>=1)
    {
        step=ceil(step);
        cout.precision(0);
    }
    else
        cout.precision(2);

    maxval = funcio(last); // valor maxm de y
    dibuixa_eix_y(maxval); // dibuixa eix
    for (w=first;w<=last;w+=step)
        dibuixa_punt(w,funcio(w),maxval); //bucle per a dibuixar

    return 0;
}
```

```
double funcio(double x)
// funcio a pintar
{
    return pow(x,3.0);
}

int conv_pant(double val,double maxval)
// converteix un valor 0..maxval en un valor 0..max_pantalla
{
    return int (max_pantalla * val / maxval) ;
}

void dibuixa_eix_y(double maxval)
//dibuixa l'eix y amb els valors 0, maxval/2, maxval
{
    cout.fill('-');
    cout << " y=";
    cout << 0;
    cout.width(max_pantalla / 2);
    cout << maxval / 2;
    cout.width(max_pantalla / 2);
    cout << maxval << endl;
    cout.fill(' ');
}

void dibuixa_punt(double x, double y, double maxval)
//pinta un punt x,y on y esta en el rang 0..maxval
{
    cout.width(4);
    cout << x << '|';
    cout.width(conv_pant(y,maxval));
    cout << "*" << endl;
}

////////////////////////////////////

Escriu primer i ultim valor de x
1 30
y=0-----13500-----27000
1 *
3 *
5 *
7 *
9 *
11 *
13 *
15 *
17 *
19 *
21 *
23 *
25 *
27 *
29 *
```