

# Práctica 7. Vectores

## Objetivos de la práctica:

- Adquirir la capacidad de programar en lenguaje C usando vectores.
- Conocer el manejo y funcionamiento de los vectores.

## 1. Los vectores en C

Los vectores nos permiten almacenar datos de un mismo tipo. En C se declaran de la siguiente forma:

```
tipo_de_datos nombre_vector[TAMANYO];
```

Podemos inicializarlos dándole valores que van entre llaves y separados por comas:

```
tipo_de_datos nombre_vector[TAMANYO]={valor1, valor2, valor3, valorn};
```

Para acceder a una posición del vector, se utiliza la siguiente notación, teniendo en cuenta que el índice va desde 0 hasta el tamaño del vector menos 1:

```
nombre_vector[indice]
```

## 2. La librería math.h

La librería `math.h` contiene funciones útiles cuando queremos realizar operaciones matemáticas. Cuando usamos funciones de esta librería, hay que incluirla al principio de nuestro programa. Además, hay que compilar el programa indicando que usamos la librería matemática mediante el comando `$gcc -o ejecutable fuente.c -lm`

Entre las funciones incluidas en la librería `math.h` están:

- Logaritmo neperiano de x: `double log(double x);`
- Logaritmo en base 10 de x: `double log10(double x);`
- Potencia x elevado a y: `double pow(double x, double y);`
- Raíz cuadrada de x: `double sqrt(double x);`
- Redondear hacia arriba el número x: `double ceil(double x);`
- Redondear hacia abajo el número x: `double floor(double x);`
- Valor absoluto: `double fabs(double x);`
- Seno de x: `double sin(double x);`
- Coseno de x: `double cos(double x);`
- Tangente de x: `double tan(double x);`

### 3. Ejercicios propuestos

#### 3.1. Ejercicio 1

Realice un programa `maximo.c` en lenguaje C que haga lo siguiente:

- La función `main` llamará a una función `inicializa` y a una función `maximo` que le devolverá el valor máximo de un vector. La función `main` lo imprime por pantalla.
- La función `inicializa` recibe un vector de 10 elementos enteros y lee 10 números y los almacena en el vector. No devuelve nada.
- La función `maximo` recibe un vector de 10 elementos enteros y devuelve el valor máximo. No imprime nada por pantalla.

#### 3.2. Ejercicio 2

Realice un programa `estadistica.c` en lenguaje C que tenga 3 funciones: `varianza`, `media` y `desviaciontipica`. La función `main` leerá 15 enteros por teclado y lo guardará en un vector. A continuación llamará a esas 3 funciones e imprimirá los valores que devuelvan esas funciones. Una calculará la media de esos 15 números enteros. Otra calculará la varianza. La última, calculará la desviación típica. Compílo con el compilador `gcc`. Después, ejecútalo para comprobar que funciona correctamente. NOTA: Varianza  $\sigma^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}$ , Media  $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$ , Desviación típica  $\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}}$ .

#### 3.3. Ejercicio 3

Realice un programa `enteros.c` que lea 20 enteros positivos, negativos y 0 (se recomienda usar una función). La función `main` llamará a una función `enteros` a la cual le pasará ese vector de 20 enteros y otro vector de 3 elementos. Esa función `enteros` calculará cuántos números positivos, negativos y 0 tiene el vector de 20 elementos. Los resultados los almacenará en el vector de 3 elementos (un elemento para cada resultado). La función no devuelve nada. La función `main` imprime los resultados por pantalla.

#### 3.4. Ejercicio 4

Realice un programa `parImpar.c` que lea 15 enteros (se recomienda usar una función). La función `main` llamará a una función `parImpar` a la cual le pasará ese vector de 15 enteros y otro vector de 2 elementos. Esa función `parImpar` calculará cuántos números pares e impares tiene el vector de 15 elementos. Los resultados los almacenará en el vector de 2 elementos (un elemento para cada resultado). La función no devuelve nada. La función `main` imprime los resultados por pantalla.