

Clase Vector – Lenguaje C++

El tipo vector se debe declarar en la cabecera

```
#include <vector>
```

La declaración general de un vector:

```
vector<tipo_de_dato> nombre_vector(tamaño,valor_inicial);
```

Donde en primer lugar se inicia la declaración con la palabra '*vector*' seguida entre '<>' para el tipo de dato del cual queremos que esté compuesto el array, posteriormente entre paréntesis tenemos dos valores (**ambos opcionales**) uno de ellos "*tamaño*" el cual indica la longitud del vector que recién se está creando, esto, aunque es opcional es bastante útil ya que la ejecución se hace más óptima, si se da por adelantado la longitud máxima que puede llegar a usar dicho vector en memoria entonces menos desplazamientos en memoria dinámica se tendrán que hacer en medio de la ejecución (tengan en cuenta esto para hacer más óptimo su código). El segundo parámetro, también opcional es "*valor_inicial*" cómo su nombre lo indica todos las posiciones del vector que se ha creado será inicializadas a ese valor.

Ejemplo 1

Para definir un vector llamado v1 de tipo int, para este caso no se especifica el tamaño y tampoco el valor inicial:

```
vector<int> v1;
```

Ejemplo 2

Vector denominado v2 para valores float con tamaño 5 y componentes iniciados en 1.5

```
vector<float> v2(5,1.5);
```

Métodos más utilizados de la clase Vector

assign	asigna los elementos al vector
at	devuelve el elemento de una posición específica
back	devuelve una referencia al último elemento del vector
begin	devuelve un iterador al principio del vector
capacity	devuelve el número de elementos que pueden ser contenidos por el vector
clear	elimina todos los elementos del vector
empty	true si el vector está vacío
end	devuelve un iterador al final del vector
erase	elimina elementos del vector
front	regresa una referencia al primer componente del vector
insert	insertar componentes en el vector
max_size	regresa el número máximo de elementos soportados por el vector
pop_back	elimina el último elemento del vector
push_back	añade un elemento al final del vector
rbegin	devuelve un reverse_iterator hacia el final del vector
rend	devuelve un reverse_iterator hacia el inicio del vector
reserve	establece la capacidad mínima del vector
resize	cambia el tamaño del vector
size	devuelve el número de componentes en el vector
swap	intercambia el contenido de un vector con el de otro