# Diagrama de clases

Diagrama

Descripción generada automáticamente

# Documentación clase plantilla: VECTOR

La clase plantilla VECTOR se hace con el animo de tener una clase a disposición con algunos atributos y métodos del contenedor vector de la STL, nuestra implementación claramente no maneja la misma eficiencia que dicho contenedor, pero para la dimensión de este programa es una buena opción.

La clase VECTOR consta de dos atributos, que son:

* **longitud:** Este atributo guardara la longitud del arreglo VECTOR y que además es un dato de tipo unsigned short ya que según nuestro criterio nuestro programa no llegará a tener VECTORES con una longitud mayor 65.5365 que es el máximo numero entero que soporta el tipo de dato elegido, de hecho, se espera contar con VECTORES con tamaños muy por debajo de este valor.
* **estructura:** Es un puntero que apunta a el arreglo dinámico que contiene los elementos del VECTOR, es un puntero de tipo T\*, aquí es donde se usa la plantilla T para que se pueda crear un arreglo dinámico de cualquier tipo de dato.

Los métodos que contiene dicha clase y sus respectivas descripciones se listan a continuación:

* **-reservarEspacio():void**

Este método aumenta un espacio más a la capacidad del arreglo al que apunta el atributo estructura, lo que hace es crear un arreglo dinámico auxiliar del mismo tamaño que la estructura, luego se copia todo el contenido de la estructura a el arreglo auxiliar, después se libera estructura y se le reserva nuevamente un arreglo dinámico de tipo T pero ahora de longitud + 1, se copia nuevamente todo lo del arreglo auxiliar en estructura y se libera el arreglo auxiliar.

* **-recortarEspacio():void**

Este método quita un espacio de la capacidad del arreglo al que apunta el atributo estructura, lo que hace es crear un arreglo dinámico auxiliar del mismo tamaño que la estructura, luego se copia todo el contenido de la estructura a el arreglo auxiliar, después se libera estructura y se le reserva nuevamente un arreglo dinámico de tipo T, pero ahora de longitud - 1, se copia nuevamente todo lo del arreglo auxiliar en estructura y se libera el arreglo auxiliar.

* -copiarConte(T \*estructOrigin, T \*estrucDestin):void

Hace una copia uno a uno de los elementos de la estructura Origen en la estructura destino.

* +append(T element):void

Este método permite agregar un elemento a la última posición de memoria del arreglo al que apunta el atributo estructura, internamente llama al método reservarEspacio y luego asigna a la última posición del arreglo el elemento pasado como parametro

* +delPos(T pos): T

Elimina el elemento almacenado en la posición pasada por parámetro, lo que se hace es correr todos los elementos del arreglo al que apunta el atributo estructura hacia la izquierda a partir de la posición ingresada como parámetro y al final llama al método recortarEspacio para recortar la última posición del arreglo y luego disminuye el atributo longitud en una unidad.

* +len():int

Retorna el valor del atributo longitud del VECTOR.

* +getfirst():T \*

Retorna la dirección de memoria del primer elemento del arreglo dinámico de tipo T al que apunta el atributo estructura.

Los constructores de esta clase platilla son dos:

* VECTOR ()

Este es el constructor por defecto, es decir, si no ingresan parámetros en la inicialización del objeto se usará este, el atributo longitud se inicializará en cero y el atributo estructura apuntará a un arreglo dinámico de tipo T de longitud cero.

* VECTOR (int tam)

Este constructor se podrá usar si se desea reservar un tamaño especifico desde la inicialización para el arreglo al que apunta el atributo estructura, el atributo longitud se inicializará con el unsigned short pasado por parámetro y el atributo estructura apuntará a un arreglo dinámico de tipo T de tamaño igual al atributo longitud.