







**(503) 2209-2930** 



## **ESTRUCTURAS DE DATOS**

Arreglo:

```
[*] array.cpp
     #include <iostream>
     #include <array>
     using namespace std;
 6 ☐ int main(int argc, char *argv[]){
         array<int, 3 > a = \{8, 9, 10\};
         cout << a.size() << endl;</pre>
         cout << a.front() << endl;</pre>
10
         cout << a.back() << endl;</pre>
11
         cout << a[0] << a[1] << a[2] << endl;
13
15
         return 0;
16
```





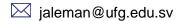






```
array.cpp [*] vector.cpp listaEnteros.cpp lista.h
     #include <iostream>
     #include <vector>
     using namespace std;
 6 ☐ int main(int argc, char *argv[]){
         vector<int> v;
         v.push back(2);
         v.push back(3);
         v.push_back(4);
10
         v.push back(5);
         v.push back(6);
12
         v.push back(7);
14
         cout << v.size() << endl;</pre>
15
         cout << v.front() << endl;</pre>
16
         cout << v.back() << endl;</pre>
18
         for(int i = 0; i < v.size();i++){
19 -
20
              cout << v[i] << endl;
21
22
         return 0;
23
```

Página 2 de 6



Edificio de Atención al Estudiante, Nivel 3, 55 Av. Sur, Condominio Centro Roosevelt, entre Av. Olímpica y Alameda Roosevelt, San Salvador, El Salvador, C.A.









**(503) 2209-2930** 

## Lista enteros:

```
array.cpp [*] vector.cpp lista.h listaEnteros.cpp
  int cuenta;
               int capacidad;
               int *items;
void agrandar();
              ListaEnteros(int capacidad);
              ListaEnteros();
              ListaEnteros(const ListaEnteros &otraLista);
              ~ListaEnteros();
              bool estaVacia();
              int tamanio();
              void insertar(int indice, int item);
              void adjuntar(int item);
               int obtener(int indice);
               bool contiene(int item);
               int remover(int indice);
               void imprimirLista();
 26 = ListaEnteros::ListaEnteros(int capacidad){
          this->cuenta = 0;
          this->capacidad = capacidad;
          this->items = new int[capacidad];
 32 ListaEnteros::~ListaEnteros(){
          delete[] items;
 36 	☐ bool ListaEnteros::estaVacia(){
          return this->cuenta == 0;
 40 = int ListaEnteros::tamanio(){
          return this->cuenta;
 42 \ }
 44 = int ListaEnteros::obtener(int indice){
          if(indice < 0 || indice >= this->cuenta) throw "Indice fuera de rango";
if(this->estaVacia()) throw "No se puede recuperar elementos de una lista vacia";
          return this->items[indice];
```

Edificio de Atención al Estudiante, Nivel 3, 55 Av. Sur, Condominio Centro Roosevelt, entre Av. Olímpica y Alameda Roosevelt, San Salvador, El Salvador, C.A.

Página 3 de 6











```
array.cpp [*] vector.cpp lista.h listaEnteros.cpp
           if(this->estaVacia()) throw "No se puede recuperar elementos de una lista vacia";
          return this->items[indice];
 50 	☐ void ListaEnteros::insertar(int indice, int item){
          if(indice < 0 || indice > this->cuenta) throw "Indice fuera de rango";
          if (this->cuenta >= this->capacidad) this->agrandar();
 54
          for(int i = cuenta - 1; i >= indice; i--){
              this->items[i+1] = this->items[i];
          this->items[indice] = item;
          this->cuenta++;
 60 - }
 62 - void ListaEnteros::agrandar(){
          int *temp = this->items;
this->capacidad *=2;
          this->items = new int[capacidad];
 66 🖃
          for(int i =0; i < this->cuenta; i++){
              this->items[i] = temp[i];
          delete[] temp;
 70 \}
 this->insertar(this->cuenta, item);
 76 bool ListaEnteros::contiene(int item){
          for(int i = 0; i < this->cuenta; i++){
               if(this->items[i] == item) return true;
 81 }
 83 int ListaEnteros::remover(int indice){
          if(this->estaVacia()) throw "No se puede recuperar elementos de una lista vacia";
if(indice < 0 || indice >= this->cuenta) throw "Indice fuera de rango";
          int valor = this->items[indice];
 88 🖃
          for(int i = indice; i < cuenta - 1; i++){</pre>
              this->items[i] = this->items[i+1];
          return valor;
```











```
int valor = this->items[indice];
for(int i = indice; i < cuenta - 1; i++){
    this->items[i] = this->items[i+1];
               this->cuenta--;
               return valor;
95 void ListaEnteros::imprimirLista(){
               std::cout << "| ";
for(int i = 0; i < this->cuenta; i++){
    std::cout << this->obtener(i) << " | ";</pre>
               std::cout << std::endl;</pre>
```

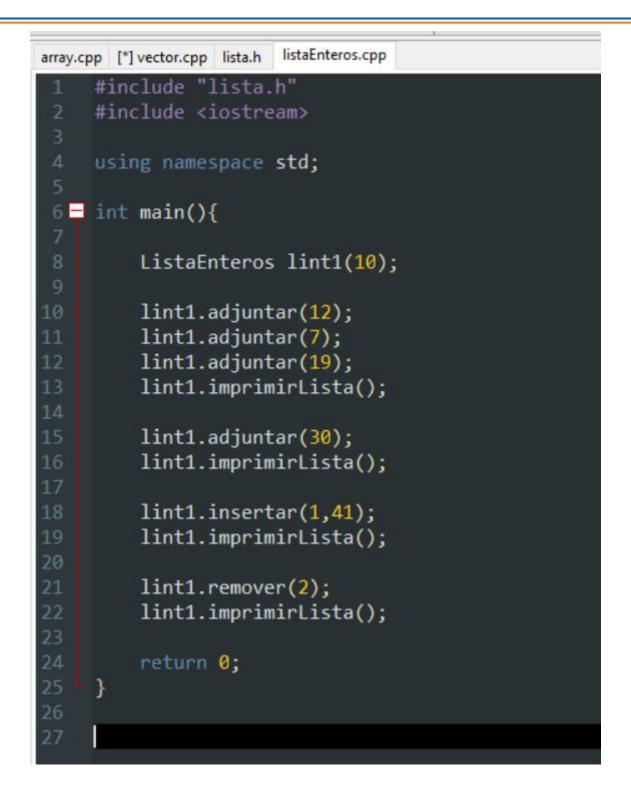
Implementación lista enteros:











Edificio de Atención al Estudiante, Nivel 3, 55 Av. Sur, Condominio Centro Roosevelt, entre Av. Olímpica y Alameda Roosevelt, San Salvador, El Salvador, C.A. Página 6 de 6