Array.h

```h
#ifndef ARRAY_H
#define ARRAY_H
#include <iostream>
using namespace std;

// Class definition
class Array
{
private:
	int* arr;
	int capacity;
	int size;
	void moveTowardFront(int index);
	void moveTowardEnd(int index);
public:
	Array(int capacity);
	~Array();
	void insert(int value);
	void remove(int value);
	void print() const;
};
#endif
```

Array.cpp

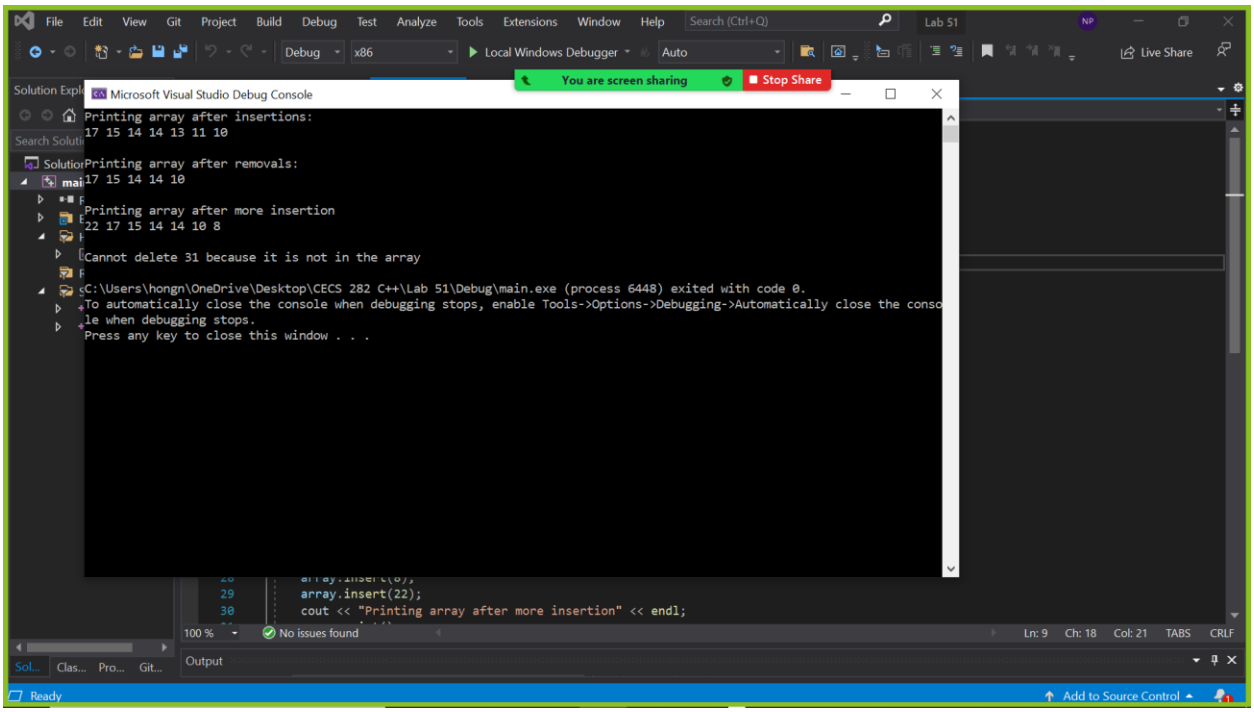
```cpp
#include "Array.h"
Array::Array(int cap)
{
	capacity = cap;
	size = 0;
	arr = new int[capacity];
}

Array::~Array() {}

void Array::moveTowardFront(int index) //when delete
{
	for (int i = index; i < size-1; i++) {
		*(arr + i) = *(arr + i + 1);
	}
}

void Array::moveTowardEnd(int index) //when insert
{
	for (int i = size; i >= index; i--) {
		*(arr + i + 1) = *(arr + i);
	}
}

void Array::insert(int value)
{
	int index = 0;
```

```
	for (int i = 0; i < size; i++) {
		if (*(arr + i) < value) {
			index = i;
			break;
		}
		else {
			index = size;
		}
	}
	moveTowardEnd(index);
	*(arr + index) = value;
	size++;
}

void Array::remove(int value)
{
	int index = 0;
	bool found = false;
	for (int i = 0; i < size; i++) {
		if (*(arr + i) == value) {
			index = i;
			found = true;
			moveTowardFront(index);
			size--;
		}
	}
	if (found == false) {
		cout << "Cannot delete " << value << " because it is not in the array" <<
endl;
	}
}

void Array::print() const
{
	for (int i = 0; i < size; i++) {
		cout << *(arr + i) << " ";
	}
	cout << endl;
}
```

Main.cpp

```
/*****************************************************
 * The application file to test the sorted array class *
 *****************************************************/
#include "Array.h"

int main()
{
	// Declaration of any array of capacity 20
	Array array(20);
	// Inserting some elements and printing array
	array.insert(15);
```

```
        array.insert(13);
        array.insert(10);
        array.insert(14);
        array.insert(11);
        array.insert(17);
        array.insert(14);
        cout << "Printing array after insertions: " << endl;
        array.print();
        cout << endl;
        // Removing two elements and printing array
        array.remove(13);
        array.remove(11);
        cout << "Printing array after removals: " << endl;
        array.print();
        cout << endl;
        // Inserting two more elements and printing array
        array.insert(8);
        array.insert(22);
        cout << "Printing array after more insertion" << endl;
        array.print();
        cout << endl;
        // Try to remove an element, which is not in the array
        array.remove(31);
        return 0;
}
```



Demonstrated at 11:04am on September 16, 2021