

Course Instructor: Rana Waqas Ali

Lab Instructor: Khizer Hayat

# Data Structures and Algorithms

## Lab Manual (Lab 01)

---



**Topic: Arrays**

Session: Spring 2022

Faculty of Information Technology

UMT Lahore Pakistan

Course Instructor: Rana Waqas Ali

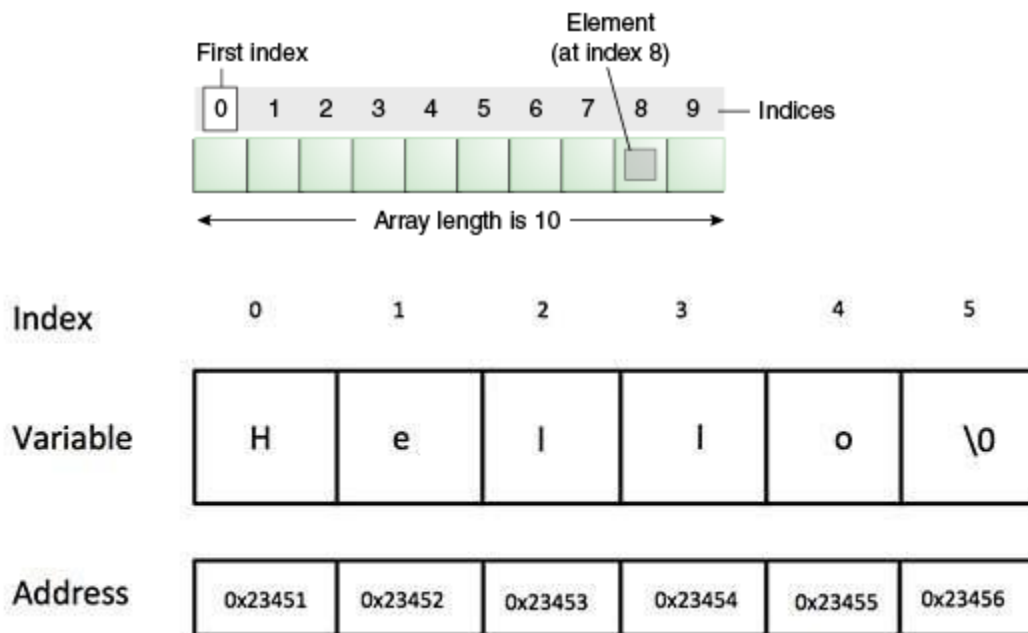
Lab Instructor: Khizer Hayat

## Objectives:

In this lab, we will implement a simple version of data structure, namely IntArray. We shall use MS Visual Studio or CodeBlocks for developing the programs. Learn how to implement Array data structure.

## Array:

An array is a series of elements of the same type placed in contiguous memory locations that can be individually referenced by adding an index to a unique identifier. For example, we can store 5 values of type int in an array without having to declare 5 different variables, each one with a different identifier. Instead of that, using an array we can store 5 different values of the same type, with a unique identifier.



## Sample Problem

Create your own implementation of the list class, called IntArray that implements a class where the elements are stored in an array.

```
Class IntArray {
    public:
        // operations performed on arrays
        IntArray (int size = ArraySize);
        ~IntArray ();
        int getValue( int index );
        void setValue( int index, int value );
        int getSize();
    protected:
        void checkBounds( int index );

    // internal data representation
```

```
        int size;
        int *data;
    };

IntArray::IntArray( int sz ) {
    // allocate an integer array of 'size' elements.
    // new returns a pointer to this array or 0
    // 0 indicates the program has exhausted its
    // available memory: a generally fatal error
    size = sz;
    data = new int[size];

    for ( int ix=0; ix < sz; ix++ )
        data[ix] = 0;
}

void IntArray::checkBounds (int index) {
    if (index < 0) {
        cout<<"An array index is out of bounds. Its value is"<< index<<endl;
        exit(1);
    }
    if (index >= size) {
        cout<<"An array index exceeds the bounds of its array. The size";
        cout<<"of the array is"<<size<<" but the value of the index is"<<index<<endl;
        exit(1);
    }
}

int IntArray::getValue (int index) {
    checkBounds(index);
    return data[index];
}

void IntArray::setValue (int index, int value) {
    checkBounds(index);
    data[index] = value;
}

int IntArray::getSize() { return size; }

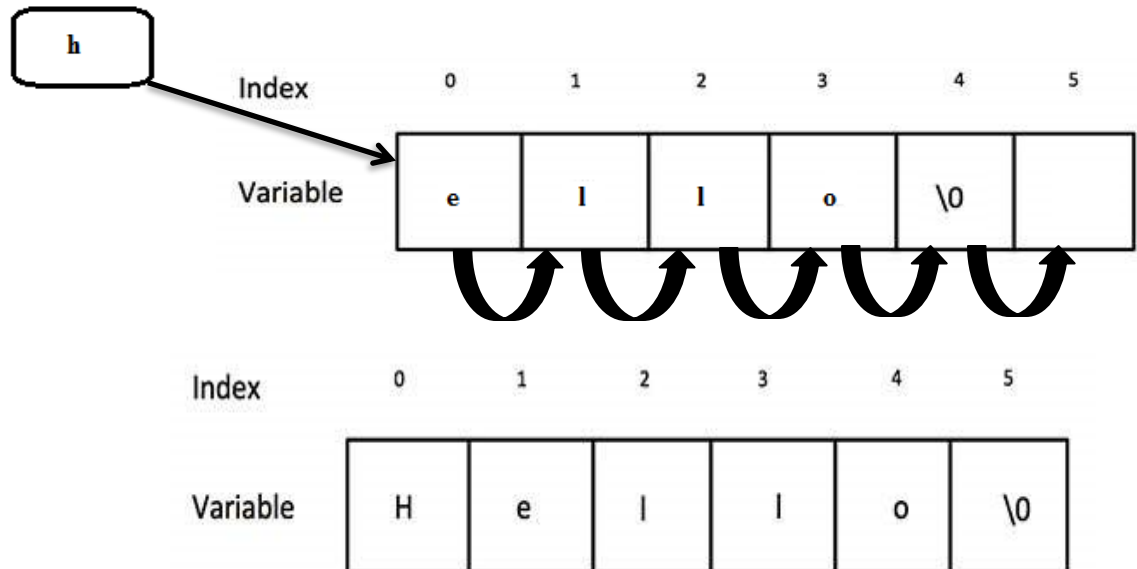
IntArray::~IntArray() { delete [] data; }
```

### Tasks:

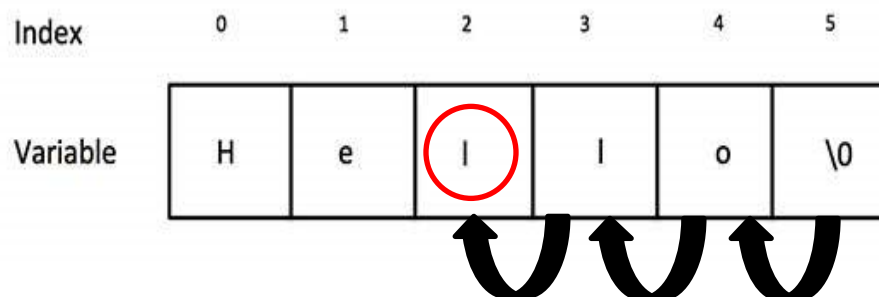
Use your programming skills and try to write the efficient code. The student with well-structured code will get more marks throughout the semester.

You have to do the following the tasks.

1. Change the above sample code from integer to template type.
2. Modify class array in Exercise 1 by adding the following:
  - a. Function called **InsertAtBegin** to add any value at the begging of list



- b. Function called **InsertAtEnd** to add vlaue at the last of list.
- c. Function called **InsertAfter** to inset new value after specific value in list.
- d. Function called **DeleteFromBegin** to delete value from the begging of list.
- e. Function called **DeleteFromEnd** to delete value from the last of list.
- f. Function called **Delete** to delete any specific index in list.



Index	0	1	2	3	4	5
Variable	H	e	l	o	\0	

- g. Function called **max** to return maximum value in the list.


### Bonus Questions

1. Write a Function called **reverseList** to reverse values in the list.

Index	0	1	2	3	4	5
Variable	H	e	l	l	o	\0

Index	0	1	2	3	4	5
Variable	o	l	l	e	h	\0

2. Write a Function called **reSizeList** to resize the list.

Index	0	1	2	3	4	5						
Variable	H	e	l	l	o	\0						
Index	0	1	2	3	4	5	6	7	8	9	10	
Variable	H	e	l	l	o	w	o	r	l	d	\0	