Recitation 7 - Arrays

CSCI1300
Instructor: Dr.Knox

Working With Arrays

Arrays are a series of elements consisting of the same type (For ex: all integers or all floats), in a contiguous chunk of memory. The elements can be individually referenced by indexing the array. They are variables that can have multiple elements. Some of the important aspects of array are:

- You must specify the type of data stored in an array, and each element in any array must be of the same type.
- An array has a static size -- they cannot grow any larger than the size you tell it to be when you create it.

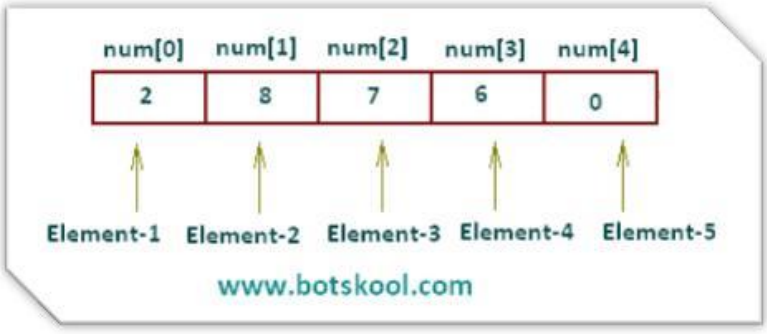
To access the value at a certain index of the array:

```
for (int i = 0; i < 10; i++){
        intArray[i] = i;                //allows the value to be updated at each index
        cout<< intArray[i] << endl;     //prints the value at each index of the array
}
```

To access the memory address of the array element:

```
for (int i = 0; i < 10; i++){
      cout << intArray + i << endl;     //prints the address
}
```

Example:



This shows us an example of an array with 5 elements and how the computer starts as 0 for counting.

Creating and Initializing the Array

When you declare an array, you will get an uninitialized section of memory. If you want to create an array of numbers, you would do: For ex: int arrayName[arraySize];

- int intNumbers [10];
- double dbNumbers[10];

Then, putting elements in the array can be set up by using the index of the array element you want to change: Note: This isn't very efficient if you have a large array, but this is the idea.

```
intNumbers[0] = 3;
intNumbers[1] = 5;
intNumbers[2] = 6;
intNumbers[3] = 8;
```

When you create an array, you do so to capture a group of items that all need to be treated separately, but also need to grouped together for some purpose. For example, when you think about a song, each note in the song is a separate sound, but they all need to be included together in a certain order to make up the song.

To initialize the array:

```
int intNum [5] = {16, 3, 20, 4, 21};
```

Hint: notice the syntax difference initializing this array versus the one above. It is very close.

Retrieving/Adding/Changing elements

To *access or retrieve* individual elements in an array

```
intNumbers[0];    will be the first item in the intNumbers array.
intNumbers[1];    will be the second item in the intNumbers array.
```

We can *change or update* the values in the array using the index of the individual items, such as:

Example 1:

```
intNumber[1] = 10;                    //this updates the value to 10 at that index
```

Example 2:

```
int intNumbers[2];

intNumbers[0] = 5;
intNumbers[1] = 7;                          //initializes
intNumbers[1] = intNumbers[1] * 2           //updates

cout << intNumbers[1] << endl;              //What value is printed?
```

Example 3:

```
for (int i = 0; i < 10; i++){
        intArray[i] = i;                    //Sets what the values are in that position
        cout << intArray[i] << endl;        //prints out the value of i at each index
 }
```

Recitation Activity:

1. Create an array of 20 doubles. Assign values to your array using a for loop, where myDoubles is the name of the array created.

2) Print out the values using another for loop.

3) Now, using your array and another loop, calculate the average of the values in the myDoubles array and print out the average using the statement below:

"The average value of myDoubles is: X" *(Hint: the answer should be 9.5)*

Once you are done with your program, show it to your TA and submit it on Moodle through the Recitation 7 Submit link by Sunday at 5pm.