CSC2111 Computer Science I Lab

Lab 11

Objectives

1. Learn how to work with pointers in class
2. Learn how to allocate and deallocate dynamic memory
3. Learn how to write copy constructor function

Dynamic memory allocation

```
1  int main()
2  {
3      int size;
4
5      cout << "Class size: ";
6      cin >> size;
7
8      int* array = new int[size]; //Dynamically allocated "array" of size 'size'
9
10     for(int i = 0; i < size; i++)
11     {
12         array[i] = i+1;
13         cout << array[i] << " ";
14     }
15
16     delete [] array;
17
18     return 0;
19 }
```

Dynamic memory allocation

```
#include <iostream>
#include <string>
using namespace std;

class MyString
{
private:
    char *arr;
    int size;
public:
    MyString(char *s):size(strlen(s)) // Constructor no. 1
    {
        arr = new char[size + 1]; // one additional byte for '\0' null terminator
        strcpy_s(arr,size+1,s);
    }

    void show()
    {
        std::cout << arr << std::endl;
    }

    ~MyString()
    {
        delete [] arr;
    }
};

int main(void)
{
    char *name = "Mark Addie";
    MyString myName(name);
    myName.show();
    return 0;
}
```

Copy Constructor

Recall that the copy constructor automatically executes in the following three situations:

1. When an object is declared and initialized by using the value of another object
2. When, as a parameter, an object is passed by value
3. When the return value of a function is an object