1 Problem Description

To complete the implementation of the mystring class, the declaration of which is provided in mystring.h file in Lab 5. The mystring class mimics the behavior of C++ string class.

To find out the behavior of each method to be implemented, we are encouraged to experiment with the corresponding method in C++ string class or to ask question in class. You may be allowed to use c-style string library functions, such as strcpy, in your implementation. However, you should not use C++ string! The reason is that such a class is assumed to be not available and we are going to implement it. Vector or any class of data structures in C++ should not be used either.

2 Purpose

Understand class, member function and field, constructor, operator overloading as member functions or basic functions, memory management, array, and string class. To gain experience and understanding related to how a complex and useful class such as string may be designed and implemented using basic c++ language features, data structures, and algorithms.

3 Design

The design is provided to us in the two files: mystring.h and mystring.cpp in Lab 5.

4 Implementation

The subsections 4.1 to 4.3 help us to understand the assignment operator and provide hint as to how to implement it. The material used to be part of the lab 5. You are encouraged to carry out the tasks.

4.1 Testing using C++ string for assignment operator

1. Comment out the line `#define string mystring` in main.cpp

2. Add the following before `cout << "Lab 5 ends" << endl;` statement

   ```cpp
   cout << "---Testing assignment operator---\n\n";
   {
   ```
string s2;
s2 = s1;
string s2name("s2");
check(s2, s2name);
}
check(s1, s1name);

We should get the following output:

This is Lab 5
checking s1
s1 contains Hello, World!
s1 capacity() is 13
s1 length() is 13
s1 size() is 13
s1 max_size() is 1073741820

checking s1
s1 contains Hello, World!
s1 capacity() is 13
s1 length() is 13
s1 size() is 13
s1 max_size() is 1073741820

---Testing assignment operator---

checking s2
s2 contains Hello, World!
s2 capacity() is 13
s2 length() is 13
s2 size() is 13
s2 max_size() is 1073741820

checking s1
s1 contains Hello, World!
s1 capacity() is 13
s1 length() is 13
s1 size() is 13
s1 max_size() is 1073741820

Lab 5 ends
Press [Enter] to close the terminal ...

4.2 Testing using mystring for default assignment operator

1. Uncomment the line #define string mystring in main.cpp
2. compile and run the same program

Do we get the same output as the previous step? Do we get a nasty run time error? Please try to explain why we get a different output in your report.

4.3 Implement assignment operator for mystring class

The following provides the pseudo code or hint about how to implement the assignment operator for mystring class as well as how to adjust the current code base.

1. Uncomment the assignment operator prototype line in mystring.h
2. Uncomment the definition assignment operator in mystring.cpp
3. If the object of assignment operator (note = is a member function conceptually) is the same as the argument object, do nothing and otherwise do the following.
4. Free space of the object
5. Allocation space based on argument object and adjust member fields (similar to copy constructor code)
6. The last statement is return *this
7. Compile and run the same program

We should get the following:

This is Lab 5
checking s1
s1 contains Hello, World!
s1 capacity() is 14
s1 length() is 13
s1 size() is 13
s1 max_size() is 1073741820

checking s1
s1 contains Hello, World!
s1 capacity() is 14
s1 length() is 13
s1 size() is 13
s1 max_size() is 1073741820

---Testing assignment operator---

checking s2
s2 contains Hello, World!
s2 capacity() is 14
s2 length() is 13
s2 size() is 13
s2 max_size() is 1073741820

checking s1
s1 contains Hello, World!
s1 capacity() is 14
s1 length() is 13
s1 size() is 13
s1 max_size() is 1073741820

Lab 5 ends
Press [Enter] to close the terminal ...

4.4 Implement the remaining methods or operators

Implement the member functions, operators, as well as non-member operators, declared in mystring.h, except for iterator methods begin and end. The implementation goes in mystrng.cpp. We should study the relationship between the methods and think about how some operation might be implemented by other operation(s). The idea is reuse and some method might be implemented by calling other method(s) and a bit of additional logic.

5 Test and evaluation

In the main program, demonstrate the functionalities of the implemented functions or operators to be similar to that of string class.

6 Report and documentation

A short report about things observed and things learned and understood. The report should also describe the test cases used in the main program and the reasons for each test case selected. Properly document and indent the source code. The source code must include the author name and as well as a synopsis of the file.

7 Lab submission

Use the same website (http://csce.uark.edu/~mmoccaro/2014/F13/) and procedure as you have done to submit your homework1.