Assignment 1

Define a class String that could work as a user-defined string type.

Include constructors that will enable us to create an uninitialized string: String s1; // string with length 0 And also initialize an object with a string constant at the time of creation like String s2("Well done!");

Include a function that adds two strings to make a third string.

Note that the statement S2 = s1; will be perfectly reasonable expression to copy one string to another. Write a complete program to test your class to see that it does the following tasks:

(a) Creates uninitialized string objects.

(b) Creates objects with string constants.

(c) Concatenates two strings properly.

(d) Displays a desired string object

//don't use inbuilt string instead use char *str;

//str = new char[length]

Assignment 2

Create class Room

Member data : int length, int width

Constructors : Room(), Room(int value = 8)

Member function : void display()

In the main create object of Room with and without parameter and call the function display on it.

Assignment 3

Add copy constructor in Assignment 1 and invoke it in main

Assignment 4

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

class String
{
    private:
        char *str;
        int size;
```

```
    public:
        String(const char *valStr);
        ~String() { delete [] str;}
        String(const String &);
        void print();
        void change(const char *);
};

String :: String(const char *valStr)
{
    size = strlen(valStr);
    str = new char[size + 1];
    strcpy(str, valStr);
}

String :: String(const String &old_str)
{
    size = old_str.size;
    str = new char[size + 1];
    strcpy(str, old_str.str);
}

void String :: change(const char *ValStr)
{
    delete [] str;
    size = strlen(ValStr);
    str = new char[size + 1];
    strcpy(str, ValStr);
}

void String :: print()
{
    cout << str << endl;
}
int main()
{
    String str1("Test1");
    String str2 = str1;

    str1.print();
    str2.print();

    str2.change("Test2");

    str1.print();
    str2.print();
    return 0;
}
```

Assignment 5

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

class Test{
    int x;
    public:
        Test()
        {
            cout << "Default constructor called" << endl;
            x= 0;
        }
        Test(const Test &t)
        {
            cout << "Copy constructor called " << endl;
        }
};

void fun(Test t)
{
    //code
}

Test fun1(Test &t)
{
    return t;
}
int main()
{
    Test t1;
    Test t2;
    Test t3 = t1;//call copy constructor
    fun(t2);//call copy constructor as pass by value

    Test t4;
    Test t5 = fun1(t4); //call copy constructor as the value returned is an
object of class Test
    return 0;
}
```