第五次作业

概念题

一、 C++ 中为什么需要对操作符进行重载？除了常用的操作符外，还可以对哪些特殊的操作符进行重载？

二、C++中系统提供的隐式赋值操作存在什么问题？如何解决？

编程题

现有一个 CustomString 类，请按照要求思考需要*重载的运算符操作*或者*构造函数相关的操作*，补全*string_operator.h*以及*string_operator.cpp*文件，得到相应的输出结果。

string_operator.h

```
#include <iostream>
#include <string.h>

using namespace std;

class CustomString
{
public:
    //构造函数
    CustomString();
    CustomString(const char* str);
    //析构函数
    ~CustomString();

private:
    char* p;   // 字符串的起始地址
    int len;   // 字符串的长度
};
```

string_operator.cpp

```

//析构函数
CustomString::~CustomString()
{
    delete[] p;
    p = NULL;
}


```

```
int main()
{
    CustomString mystr("this is e CustomString class for
testing!");
    cout << mystr[8] << endl;
    mystr[8] = 'a';
    cout << mystr <<endl;
    CustomString mystr2 = mystr;
    cout << mystr2 << endl;
    CustomString mystr3;
    mystr3 = mystr + mystr2;
    cout << mystr + mystr2 << endl;
    mystr3 += mystr;
    cout << mystr3 << endl;
    cout << (mystr == mystr2) << endl;
    cout << (mystr != mystr3) << endl;
    CustomString mystr4;
    cout << "Input any string to test the overloaded input
operator >>: " << endl;
    cin >> mystr4;
    cout << mystr4 << endl;
    cout << "Congratulations! testing passed!" << endl;

    return 0;
}
```

期望输出如下：

```
e
this is a CustomString class for testing!
this is a CustomString class for testing!
this is a CustomString class for testing!this is a
CustomString class for testing!
this is a CustomString class for testing!this is a
CustomString class for testing!this is a CustomString
class for testing!
1
1
Input any string to test the overloaded input operator >>:
hello
hello
Congratulations! testing passed!
```

要求输出必须和期望输出严格一致，提交完整的.h和.cpp文件，自己填写的部分代码高亮或者斜体显示。