9. 复习String的实现过程

写一个String Class

防卫式

我们需要什么数据？

指针

构造函数。与class同名

是否有return type？

参数：默认参数

const

class with pointer

Big Three

拷贝构造 String(const String& str);

拷贝赋值

能否reference return 是否是local objects

析构函数 ~String

char* get_c_str() const { return m_data; }

char* m_data

C风格的字符串

return { & value const

pass

ctor dtor

构造函数 析构函数

{ Strlen()

Strcpy() }

inline

copy ctor

copy assignment operator

你的东西的来源端与目的端

我们return 只要传出去

是value还是reference看函数声明

String& String::operator=(const String& str)

return *this

10. 扩展补充：类模板，函数模板及其他

static

this pointer？不断熟悉

{ data members
  static data members
  member functions
  static member functions

```
complex c1, c2, c3;
cout << c1.real();
cout << c2.real();
```

this ⇒

```
complex c1, c2, c3;
cout << c1.real(&this);
cout << c2.real(&this);
```

一个成员函数处理很多变量：借助this pointer

但是static脱离了类（该）对象存储在了其他地方

地址：中国 安徽 合肥市金寨路96号 邮编：230026
电话：0551-63602184 传真：0551-63631760 Http://www.ustc.edu.cn

静态函数与普通函数的区别：
没有 this pointer，只能处理 static member

如果要使用静态数据，我们需要在类内 public 外定义初值。

调用 static 函数的方式：
① 通过 object 调用
② 通过 class name 调用

不希望外界创建 class，只希望有一份：
把 ctors 放在 private 区，且定义 ctors 为 static

这是一种设计模式 Singleton → Meyers Singleton

cout -> _IO_ostream_withassign -> ostream

C++ 标准库里的算法都是 function template

namespace

```
namespace std
{
  ...
}
```

包装在一个单元里。

using directive：using namespace std;
using declaration：using std::cout;

模板 template。
class template
template <typename T>
function template 函数模板

```
template <class T>
inline
const T& min(const T& a, const T& b)
{
  return b<a ? b:a;
}
```

与前面不同，函数模板时编译器会对 fuction template 进行参数推导，然后会在 class T 中寻找操作符重载

更多细节：
operator type() const;
auto (since C++11);
...