

计算机学院高级语言程序设计课程实验报告

实验题目：实验四、数据的共享与保护		学号：202300130183
日期：2024 年 3 月 4 日	班级：2023 级智能班	姓名：宋浩宇
Email：202300130183@mail.sdu.edu.cn		
<p>实验目的：</p> <ol style="list-style-type: none">1. 观察程序运行中变量的作用域、生存期和可见性。2. 学习类的静态成员的使用。3. 学习多文件结构在 C++ 程序中的使用。		
<p>实验软件和硬件环境：</p> <p>实验软件：Windows 11 家庭中文版(x64) Visual Studio 2022</p> <p>硬件环境：处理器：13th Gen Intel(R) Core(TM) i9-13980HX 2.20 GHz</p> <p>RAM 32.0 GB (31.6 GB 可用)</p>		
<p>实验步骤与内容：</p> <p>第 5 章 1. a) 首先是第一次输出的 1, 2, 这里的输出的是全局变量 x, y 的值；然后在 main 里定义了 x, y, 此时在 main 函数里这两个 x, y 取代全局变量 x, y 的可见性，因此第二次输出 10, 20；第三次输出的 3, 4, 是代码块里定义的变量 x, y, 这里的 x, y 取代了 main 函数里定义的 x, y 的可见性，因此代码块里输出的是代码块里定义的 x, y 的值；第四次输出，在函数 fn1 里，定义了 y 的值，它取代了全局变量 y 的可见性，而 x 依旧是全局变量 x, 因此输出全局变量 x 的值 1 和局部变量 y 的值 200；第五次输出是在 main 函数里的输出，因为代码块里的定义和 fn1 函数里定义的 x, y 对于它们之外的部分是不可见的，因此对于此处的输出来说，它还是按照 main 函数里定义的 x, y 来进行输出即输出 10, 20。</p> <p>第 5 章 1. b)</p>		

```
Microsoft Visual Studio 调试 × + - □ ×
test1's SeverName is:a
test2's SeverName is:a
test3's SeverName is:a
ClientNum is:3
change test1's SeverName to "b"
test1's SeverName is:b
test2's SeverName is:b
test3's SeverName is:b
change test2's SeverName to "c"
test1's SeverName is:c
test2's SeverName is:c
test3's SeverName is:c
change test3's SeverName to "d"
test1's SeverName is:d
test2's SeverName is:d
test3's SeverName is:d

F:\Homework\高程作业\实验4\codes\x64\Debug\codes.exe (
进程 30904)已退出, 代码为 0。
按任意键关闭此窗口 . . .|
```

第 5 章 2. a) 表示用 age 的值来初始化 itsAge 这个成员变量

第 5 章 2. b) numOfCats 仅在 Cat 类内可见, 而全局变量在整个定义或声明它的文件内都可见

第 5 章 2. c) 不可以

第 5 章 2. d) 还可以作为 Cat 类中 public 的函数来设计

第 5 章 3. 在 engine 类中加入 friend class fuel; 原因是通过将 fuel 类声明为 engine 类的友元类来实现 fuel 类对 engine 类私有和保护成员的访问

第 5 章 4.

```
Microsoft Visual Studio 调试 × + - □ ×
成员函数计算为 :
1.41421
友元函数计算为 :
1.41421
一般函数计算为 :
1.41421

F:\Homework\高程作业\实验4\codes\x64\Debug\codes
.exe (进程 32784)已退出, 代码为 0。
按任意键关闭此窗口 . . .|
```

第 5 章 5. const 不可以去掉, 去掉它就相当于对于 const R 类不存在 print() 成

员函数了。a 可以，b 不可以，因为 b 是 const 类型的，在初始化之后其中的成员都是不可修改的。

第 5 章 6. 不可以，因为 A 类中的 a 成员是 const 类型的，是不可修改的，可以用初始化列表进行初始化，但是 {a=i;} 本质上是一次赋值运算，这对 const 类型的对象是不可进行的。

第 5 章 7. 因为可以使用预处理指令来规避重复包含等问题。

第 5 章 8. a) 关于拆分为多文件结构: Point.h 和 Line.h 里分别写 Point 类和 Line 类的声明和定义。Point.cpp 和 Line.cpp 里分别写 Point 类和 Line 类的实现。Line_main.cpp 里写主函数内的程序。并使用 #ifndef/#define/#endif 的结构来防止头文件的重复包含。

第 5 章 8. b) 关于输出 Line 对象个数: 首先在 Line.h 中 Line 类的声明中在 private 类型中定义 static int count, 并声明一个析构函数在其中添加 count-- 语句, 再声明一个 static int counts() 函数用于返回 count 的值; 再在 Line.cpp 中初始化 count 为 0, 并在构造函数和复制构造函数的实现中添加 count++ 语句, 并实现 counts 函数的功能。最后在 Line_main.cpp 中添加一行 cout << "The number of Lines is " << Line::counts() << endl; 用于输出 Line 类的数量。

结论分析与体会:

掌握了程序运行中变量的作用域、生存期和可见性三种特性。学习和练习了类的静态成员的使用。对于多文件结构在 C++ 程序中的使用有了初步的了解, 并且知晓了在制作多文件项目时需要考虑的问题以及相应的解决方法。

就实验过程中遇到的问题及解决处理方法 (如有):

未遇到问题