5-1、5-2、5-5、5-6、5-9、5-10、5-16、6-6、6-7、6-9、6-11、6-15、6-18、6-19

<https://github.com/jinrongsyb17/homework-space>

第3次作业 非程序题

徐菡2017310719

5-1

作用域是一个标识符在程序正文中有效的区域。

有局部作用域、类作用域和文件作用域。

5-2

程序运行到某一点。能够引用到的标识符，就是该处可见的标识符。

可见性是从对标识符的引用的角度来谈的概念

可见性表示从内层作用域向外层作用域“看”时能看见什么。

如果标识在某处可见，则就可以在该处引用此标识符。

可见性（续）一般原则：标识符应声明在先，引用在后。

如果某个标识符在外层中声明，且在内层中没有同一标识符的声明，则该标识符在内层可见。

对于两个嵌套的作用域，如果在内层作用域内声明了与外层作用域中同名的标识符，则外层作用域的标识符在内层不可见。

5-5

静态数据成员：使用static关键字声明的类的所有对象共同拥有的数据成员。

一个类的所有对象具有相同的属性，但属性值不同。

（类属性：描述类的所有对象的共同特征的一个数据项，对于任何对象实例。它的属性值是相同的。）

静态数据成员的特点：

静态数据成员具有静态生存期。静态数据成员不属于任何一个对象，因此可以通过类名对它进行访问，一般的用法是“类名：：标识符”。

该类的所有对象维护静态数据成员的同一个拷贝。

静态数据成员必须在类外定义和初始化。

5-6

静态函数成员：使用static关键字声明的由同一个类的所有对象所共有的函数成员。

静态成员函数的特点：

静态成员函数可以通过类名或对象名来调用（而非静态成员函数只能通过对象名来调用）。

静态成员函数可以直接访问该类的静态数据和函数成员。而访问非静态成员，必须通过对象名。

静态成员函数只能引用属于该类的静态数据成员或静态成员函数。

5-9

类B不是类A的友元；

类C不是类A的友元；

类D不是类B的友元（友元关系不能被继承）。

5-10

可以，如：

private:

static int I;

6-6

\*的作用是指向指针所指向的对象值；

&的作用是得到一个对象的地址。

6-7

指针是一种专门用来存放内存单元地址的类型，具有指针类型的变量成为指针变量。

指针变量存放的是另一个对象的地址，这个地址中的值就是另一个对象中的内容。

6-9

/0

6-11

引用不能被重新分配，也不能为空值；指针是存放地址的变量，可以改变变量的值，也可以为空指针。

6-15

const int \* p1 是一个常量指针，即指向整型常量P1的指针，因此不能通过指针P1来改变其指向的整型变量的值；而 int \* const p2是一个指针常量，用来存放整型变量的地址，并且这个指针的值一旦被初始化后，它的值就不能再改变了，即不能被重新赋值。

6-18

给p分配的内存没有被释放掉。

int \*fn1();

int main ()

{

int \*a =\*fn1();

cout <<”the value of a is: “<<\*a;

delete a;

return 0;

}

int \*fn1()

{

int \*p = new int (5);

return p;

}

6-19

long ( \*p ) (int);

long ( A::\*p) (int);