5-1 作用域是标识符的有效范围，是一个标识符在程序正文中有效的区域。

类型：函数原型作用域、局部作用域、类作用域、命名空间作用域

5-2 可见性是指标识符是否可以被引用。

一般规则：标识符要声明在前，引用在后；在同一作用域中，不能声明同名的标识符；在没有相互包含关系的不同的作用域中声明的同名标识符，互不影响，且外层标识符在内层不可见。

5-5 静态数据成员具有静态生存期，是类的数据成员的一种特例，采用static关键字来声明。

特点：它是描述类的所有对象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。必须在类外定义和初始化，并且需要用（：：）来指明所属类。

5-6 静态函数成员是使用static关键字声明的函数成员，它属于整个类，由同一个类的所有对象共同拥有，为这些对象所共享，节省系统开销，提高程序效率。

特点：调用时可以通过类名或对象名，可以不依赖任何对象，直接访问该类的静态数据和函数成员。一般用于访问同一个类的静态数据成员。

5-9 类元关系是单向的，因此类B不是类A的友元；类元关系是不能传递的，所以类C不是类A的友元；类元关系是不能继承的，因此类D不是类B的友元。

5-10 可以。private：

static int a；

5-16编译的输入文件是源文件，输出文件是目标文件；连接的输入文件是目标文件，输出是可执行文件。

不同：编译是指编译器将源代码翻译为机器语言形式的目标文件。连接是指将各个编译单元的目标文件和运行库中被调用过的单元加以合并后生成可执行文件。

1. 编译时会报错，因为函数的实参和形参不匹配。
2. 报错，因为函数必须先定义才能使用，这里函数没有定义。
3. 不会报错。
4. 连接报错，函数重复定义了。

6-6 \*是指针运算符，也称解析，表示获取指针所指向的变量的值。

&是取地址运算符，用来得到一个对象的地址。

6-7 指针是一种数据类型，具有指针类型的变量成为指针变量。指针中存储的地址是指针所指向的对象的内存单元地址。这个地址中的值是这个对象的内容对象值本身。

6-9 结束符是’/0’字符。

6-11 引用是其他变量的别名，隐藏了“地址”这一概念，且只能在初始化时指定被引用的对象，不能再更改。

指针存储的是地址，普通指针可以多次被赋值，多次更改它所指向的对象。

6-15 const int\*p1是指向常量的指针，不能通过指针p1来改变它所指向的整型值。

int \*const p2是指针型常量，在初始化之后，这个指针本身的值不能被改变。

6-18 问题：定义a的时候没有定义为指针类型

没有delete函数，\*p被分配的内存没有被释放。

改正：int a改为int\*a

在return 0之前加上delete [] a;

6-19 （1）long (\*func) (int a);

long (A::\*func2) (int b);

6-21 6-22见源文件。