

下面的提示分为初级提示、中级提示和高级提示。

初级提示一般仅仅提示基础思路，适合大脑一片空白的小朋友；

中级提示会提示到一些易错点，适合写出了程序框架但是觉得似乎有哪里不对的小朋友；

高级提示大概就很接近剧透了，适合差一点点就豁然开朗的小朋友。

三类提示会分别放在三页上，大家按需查看。

初级提示：

第 1 题：

实验指导书上的题目描述已经说明了基础思路。

第 2 题：

实验指导书上的题目描述已经说明了基础思路。注意加法函数 add 需要重载。

第 3 题：

基础思路：本题需要一份 SavingAccount 类通用的数据来记录账户的总数，但在 main 函数执行 new SavingAccount[num]时，num 的值是无法直接传入 SavingAccount 的成员函数中的。因此，需要将账户总数定义为该类对象的计数器，并在构造函数中执行该计数器++的操作，从而实现每创建一个对象（账户）都能即时更新账户总数。同时，该计数器还能用来给每个新对象（账户）的账号赋值。

第 4 题：

基础思路：用单链表来存储角色学会的技能，每个节点是一个结构体，SkillList 类中包含一个成员指针 head 作为单链表的头指针。在 SkillList 类中设计一个整型常量数据成员来表示最大技能数，再用一个普通的整型数据成员对加入的技能计数，当后者达到前者之后不再允许加入新的技能。

中级提示：

第 1 题：

易错点：

- ① 误解了 getO 函数的作用。该函数应该只获取圆心的位置，而不设置圆心的位置。
- ② 忘了在值传递情况下，形参的变化不会影响实参的值。
- ③ 没仔细看题目，忘了写构造函数。
- ④ 成员函数参数与数据成员重名。

第 2 题：

易错点：

- ① 构造函数或拷贝构造函数没有另外开辟动态内存空间。
- ② 没仔细看题目，用整型数组而非字符数组来存放 LongLongInt 类的数据。
- ③ 当前对象与引用参数互相干扰。
- ④ 相加时左右顺序颠倒、忘记进位或算错。
- ⑤ 没考虑两数相加后的结果比两数的位数都多的情况。

第 3 题：

易错点：

- ① 看漏题目，没写账号（即一个 int 类型的数据成员）。
- ② 误以为静态成员函数不能通过参数来操作类中的非静态成员。
- ③ 忘写静态数据成员的定义初始化语句。

第 4 题：

易错点：

- ① 对指向定长数组或字符串常量的指针使用 delete。
- ② 在链表操作中混淆头指针、当前结点指针以及通过它们间接访问的 next 指针。
- ③ 在指针并未指向结点的情况下间接访问结构体成员。

高级提示：

第 1 题：

本题没有高级提示。

第 2 题：

本题没有高级提示。

第 3 题：

能通过“类名::”来调用的都是静态成员函数。静态成员函数没有 this 指针，因此没有当前对象，所以不能通过当前对象来访问其私有成员。但静态成员函数可以有参数，因此可以通过参数来访问同类对象的私有成员。

第 4 题：

不用对技能列表排序。只要在加入技能时在正确的位置插入（按名称升序排列），输出技能列表时就可以简单地按链表连接的顺序显示各个结点上的技能名称。在撤销对象时按从后往前的顺序删除链表中的结点，用递归函数来操作比较方便。