



循环与数组



- for循环体中有多个语句，却不加大括号。
- 试图在for循环结束后使用for循环体或控制行中定义的变量。
- 混淆break和continue（break直接结束循环，continue不跳过++i）。
- 用变量定义普通数组的大小：`int a[n];`
- 用int变量储存过大的数（比如11位手机号）。
- 把空字符‘\0’与空格字符‘ ’混淆，或误写成了‘0’、‘/0’。
- 混淆键盘输入的结尾‘\n’与内存中字符串的结尾‘\0’。
- 误把字符数组长度当作字符串的长度。（要看空字符，或用strlen函数）
- 误将二维字符数组名用作cin.getline函数的第一个参数。



函数



- 函数的调用语句中多写了参数的类型、数组的中括号。
- 函数有返回值却把返回类型写成void，或返回类型不是void却没有return。
- 定义静态变量时，写上了static却忘了写数据类型，如 `static x = 1;`。
- 在形式参数中写static（没有这种语法）。
- 函数模板的声明前没加template那行，或定义前没加template那行。
- 递归函数中没有判断递归结束的条件和递归结束时的操作（出口）。
- 递归函数中把递归过程中的操作写到了递归调用的后面，或把回溯过程中的操作写到了递归调用的前面。
- 在递归函数中只记得处理参数，忘记处理返回语句。



指针



- 没掌握用不同的参数可以区分同名函数。（函数重载机制）
- 把数组名赋值给指针时，在数组名前面加&。（数组名自己就是地址！）
- 混淆定义引用的&和取地址的&。
- 没掌握引用传参的写法。（形参包括类型、&、变量名，实参只有变量名）
- 没掌握用两个指针相减可以得到数组中两个元素的下标之差。
- 两个指针（或两个地址，或一个指针和一个地址）相加。（不能相加！）
- 给地址表达式解引用时漏加括号，例如把 `*(p+1)` 写成了 `*p+1`。
- 没掌握函数如何返回一个字符串。（返回类型是 `char *`，返回值是地址）
- 没掌握指针作为参数。（形参有类型、*、指针名，实参只有指针名）



指针



- 误以为字符数组元素的值默认为空字符。（是随机值，无论是否动态数组）
- 忘了可以用循环来给数组元素赋值。（只记得初始化表了）
- 把 `if...else if...else` 写成 `if...if...else`。
- 排序算法（特别是冒泡排序）中 `j+1` 导致数组下标越界。
- 试图把字符串赋值给单个字符变量/字符数组元素。
- 试图 `delete` 非动态变量或数组。（只有动态的才能被 `delete`）
- `delete` 后面的地址不是 `new` 操作分配的空间的起始地址。
- 为了再次使用一个函数的返回值而再次调用函数，导致多余操作。
- 函数返回局部变量的地址或引用，或试图返回两个不同类型的值。

字符串与结构体

- 没掌握strcpy、strcat。（dst是数组起始地址，src是字符串起始地址）
- 没掌握如何把常量字符串存入字符数组。（用strcpy函数）
- 字符数组的长度不够，导致存不下字符串末尾的空字符。
- 试图在cin.getline的参数表里只写第一和第三个参数。（不能空中楼阁）
- 混淆cin>>、cin.get()、cin.getline()。
- 试图通过没有指向结构体变量的结构体指针访问成员。
- 误以为结构体是一种数组。（结构体类型是类型，结构体变量是变量）
- 误把struct当作数据类型。（struct是用来定义结构体类型的关键字）
- 混淆结构体指针之间的赋值（在链表中常见）和结构体变量之间的赋值。



面向对象



- 试图对存储数字字符的字符数组直接做加减法。（别为了偷懒而开脑洞）
- 没掌握如何在类的定义之外定义成员函数。（要写上 类名::）
- 没掌握什么时候会调用构造函数。（定义对象时自动调用）
- 当构造函数没有操作需要写在函数体里时，漏写大括号。
- 知道某个成员指针要指向动态空间，但在构造函数里忘了写new操作。
- 试图在外部访问类的私有成员。（只有友元才可以）
- 试图直接用cout<<输出类的对象。（需要重载<<运算符）
- 没掌握类的对象在值传递时会调用拷贝构造函数。
- 声明和定义拷贝构造函数时参数没有用引用传递。



面向对象



- 拷贝构造函数漏给某个数据成员初始化。
- 试图在常量成员函数中调用非常量成员函数。
- 误以为定义类指针数组时会调用构造函数。（指针不是对象）
- 误以为类的非静态成员变量会在各对象之间共享。（只有静态的才会）
- 在成员函数中手动调用析构函数来撤销当前对象。（还没过河就拆桥）
- 重载>>运算符时给第二个参数加上const。（导致不能存入数据了）
- 试图在派生类的构造函数中直接对从基类继承的数据成员初始化。（要在初始化列表中调用基类的构造函数）
- 想用运行时的多态性，但忘了把基类的成员函数设为虚函数。



其它 Bug



- 宏定义错写成：`#define 标识符 = 数值;` //别加等号!也别加分号!
- 漏写或写错`using namespace std;`
- 定义变量后没有初始化也没有赋值，就用于运算。
- 在`if()`中把 `==` 写成 `=`
- 指针传参和引用传参的比较（`x`是`int`变量）：
 - 形参是`int *p`，实参可以是`&x`，不能是`x`
 - 形参是`int &p`，实参可以是`x`，不能是`&x`
- 遍历一行包含多个单词的文本时忘了结尾不是空格符而是空字符或换行符
- 遍历多行文本时没注意结尾不是换行符而是EOF