

思考题 1: 假如我们写一个程序, 程序在运行时会通过输出流来输出日志的形式来记录运行状态, 但不幸的是, 某个时刻程序崩溃了, 此时我们是否一定能看到程序崩溃时最新输出的日志吗? 如果能, 为什么? 如果不能, 为什么? 有哪些方法可以应对这一情况? 思考一下这对精准调试大型复杂程序的意义

思考题 2: 一般多个程序打开同一个文件时, 需要调用 clear 函数, 如果不调用这个函数, 会出现什么问题?

编程题 1:

请编写一个程序用于在控制台和文件之间进行输入输出转换。该程序将文件中的内容读出, 文件中内容是一个 string 的句子, 对句子中每个单词求反, 并打印到控制台。

代码结构如下:

```
class ReverseWords {
public:
    string getSentence(string filename); //获得句子
    string reverseString(); //倒置句子里面每个单词, 并作为返回值
private:
    //根据需要自行添加成员变量
};
```

(一) 注意事项:

1. 文件不为空
2. 以单个空格为断句方式
3. 使用文本读入的方式

(二) 具体示例:

```
ReverseWords *r = new ReverseWords();
string sentence = r->getSentence("filename");
cout << r->reverseString() << endl;
```

(三) 用例 1:

文件中句子: "I'am fine, thank you!"

输出: "ma'l ,enif dna !uoy"

(四) 提交要求:

1. 提交一个源码文件 ReverseWords.h, 直接打包成 zip 压缩格式。
2. 实现和声明都写在.h 头文件中。
3. 请严格按照给定的接口进行编码, 否则无法调用测试用例。
4. 提交的源码文件中, 不能包含 main 函数, 否则无法编译通过

编程题 2:

以二进制方式读取一个文件，除了开头第一个字符可能是+-号，文件里其余都是都是数字，用随机读取的方法获得所有奇数位的数组成的整数（从左往右数是奇数位）。

代码结构如下：

```
class Palindrome {
public:
    int getOddNumber(string filename); //获得需要的奇数位的数组成的整数
    bool IsOddNumberPalindrome(); //判断整数是否为回文数
private:
    //根据需要自行添加成员变量
};
```

(一) 注意事项：

- 1.负数不是回文数
- 2.正数去掉符号判断是否为回文数
- 3.正数去掉最前面的 0
- 4.使用二进制方式读取

(二) 具体示例：

```
Palindrome *p = new Palindrome();
int number = p->GetOddNumber(filename);
cout << number << "\t" << p->IsOddNumberPalindrome();
```

(三) 用例 1:

输入：123456
输出：135 False

用例 2:

输入：+121
输出:11 True

用例 3:

输入: -121
输出:-11 False

用例 4:

输入：+0011
输出：1 True

(四) 提交要求：

1. 提交一个源码文件 Palindrome.h，直接打包成 zip 压缩格式。

2. 实现和声明都写在.h 头文件中。
3. 请严格按照给定的接口进行编码，否则无法调用测试用例。
4. 提交的源码文件中，不能包含 main 函数，否则无法编译通过

出题人：陈旭