

Lab 9 Part II 栈和队列

Problem 1. 尝试在下面的代码基础上，实现一个具有奇偶翻转功能的栈类（允许修改代码），编写main函数进行测试。

```
#define MAX_SIZE 15
class myStack
{
    double data[MAX_SIZE];
    int top;
public:
    myStack(); //构造函数
    bool isEmpty(); //栈空判断
    bool isFull(); //栈满判断
    int length(); //返回栈中数据的个数
    double push(double a); //进栈
    double pop(); //出栈
    void operator~(); //运算符重载，倒序翻转栈数据;从0到top将栈中连续相邻位置中的数据交换，当栈顶top没有匹配交换位置时，保持top位置不变
    void printData(); //从0到top打印栈序列，空栈打印"empty stack"
};
```

Problem 2. 根据下面代码，自定义一个队列类 queue，队列中需要进行保存并处理的那一批同类型数据，使用一个大小为 max_size 的数组来存放，编写main函数进行测试。

```
const int max_size = 10;
class queue {
private:
    float data[max_size]; //data 中存放队列的实际数据，max_size 为事先定义的有名常量
    int front, rear; //首尾下标，队列数据放在 data[front+1]到 data[rear]之中
    int num;
public:
    queue(void); //构造函数，置 front=rear=-1, num=0, 意味着队列为空
    bool Empty(void); //判断队列是否为空（即判 num 是否为 0）
    bool Full(void); //判断队列是否已满（即判 num 是否等于 maxsize）
    void Add(float); //往队列尾部加入数据（加入到 data[rear+1]处，num 增 1）
    float Delete(void);
};
```

Problem 3 (OpenJudge 6263) . 利用Problem 1中实现的栈类，解决下面问题：

输入一个布尔表达式，请你输出它的真假值。

比如：(V | V) & F & (F | V)

V表示true，F表示false，&表示与，|表示或，!表示非。

上式的结果是F

输入

输入包含多行，每行一个布尔表达式，表达式中可以有空格，总长度不超过1000

输出

对每行输入，如果表达式为真，输出"V",否则输出"F"

样例输入

```
( V | V ) & F & ( F | V )
!V | V & V & !F & ( F | V ) & (!F | F | !V & V)
(F&F|V|!V&!F&!(F|F&V))
```

样例输出

```
F
V
V
```

Problem 4 (OpenJudge 1696). 利用Problem 1中实现的栈类，解决下面问题：

波兰表达式是一种把运算符前置的算术表达式，例如普通的表达式 $2 + 3$ 的波兰表示法为 $+ 2 3$ 。波兰表达式的优点是运算符之间不必有优先级关系，也不必用括号改变运算次序，例如 $(2 + 3) * 4$ 的波兰表示法为 $* + 2 3 4$ 。本题求解波兰表达式的值，其中运算符包括 $+ - * /$ 四个。

输入

输入为一行，其中运算符和运算数之间都用空格分隔，运算数是浮点数。

输出

输出为一行，表达式的值。

可直接用`printf("%f\n", v)`输出表达式的值v。

样例输入

```
* + 11.0 12.0 + 24.0 35.0
```

样例输出

```
1357.000000
```

提示

可使用`atof(str)`把字符串转换为一个double类型的浮点数。`atof`定义在`math.h`中。
此题可使用函数递归调用的方法求解。

Problem 5. 利用Problem 2中实现的队列类，解决下面问题：

大数学家高斯小时候偶然间发现一种有趣的自然数集合Blah，对于以a为基的集合Ba定义如下：

- (1) a是集合Ba的基，且a是Ba的第一个元素；
- (2) 如果x在集合Ba中，则 $2x+1$ 和 $3x+1$ 也都在集合Ba中；
- (3) 没有其他元素在集合Ba中了。

现在小高斯想知道如果将集合Ba中元素按照升序排列，第N个元素会是多少？

输入

输入包括很多行，每行输入包括两个数字，集合的基 $a(1 \leq a \leq 50)$ 以及所求元素序号 $n(1 \leq n \leq 1000000)$

输出

对于每个输入，输出集合 B_a 的第 n 个元素值

样例输入

```
1 100
28 5437
```

样例输出

```
418
900585
```

Problem 6 (洛谷 P3156) . 有 n ($n \leq 2 \times 10^6$) 名同学陆陆续续进入教室。我们知道每名同学的学号（在 1 到 10^9 之间），按进教室的顺序给出。上课了，老师想知道第 i 个进入教室的同学的学号是什么（最先进入教室的同学 $i=1$ ），询问次数不超过 10^5 次。

输入格式

第一行 2 个整数 n 和 m ，表示学生个数和询问次数。

第二行 n 个整数，表示按顺序进入教室的学号。

第三行 m 个整数，表示询问第几个进入教室的同学。

输入格式

m 个整数表示答案，用换行隔开。

输入样例

```
10 3
1 9 2 60 8 17 11 4 5 14
1 5 9
```

输出样例

```
1
8
5
```

Problem 7 (洛谷 P1996) . n 个人围成一圈，从第一个人开始报数，数到 m 的人出列，再由下一个人重新从 1 开始报数，数到 m 的人再出圈，依次类推，直到所有的人都出圈，请输出依次出圈人的编号。

输入格式

输入两个整数 n, m 。

输入格式

输出一行 n 个整数，按顺序输出每个出圈人的编号。

输入样例

```
10 3
```

输出样例

```
3 6 9 2 7 1 8 5 10 4
```