

青少年软件编程（C语言）等级考试试卷（六级）

分数：100 题数：4

一、编程题(共 4 题，共 100 分)

1. 波兰表达式

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。

时间限制：1000

内存限制：65536

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
```

试题编号：20230318-6-01

试题类型：编程题

标准答案：

试题难度：一般

试题解析：

展示地址：[点击浏览](#)

考生答案：（此题已作答）

考生得分：25

是否评分：已评分

评价描述：

考生答案：

```
#include<bits/stdc++.h>
using namespace std;
double exp(){
    char s[20];
    cin>>s;
```

```
switch(s[0]){
    case '+':return exp()+exp();
    case '-':return exp()-exp();
    case '*':return exp()*exp();
    case '/':return exp()/exp();
    default:return stof(s);
}
}
int main(){
    printf("%.6f",exp());
    return 0;
}
```

2. 括号配对

字符串中可能有三种成对的括号，"()"、"[]"、"{}"。请判断字符串的括号是否都正确配对。无括号也算正确配对。括号交叉算不正确配对，例如"12{34[78]ab}"就不算正确配对。但是一对括号被包含在另一对括号里面，例如"12{ab[8]}"不影响正确性。

时间限制：1000

内存限制：65536

输入

第一行为整数 n ($n < 40$)，接下来有 n 行，每行一个无空格的字符串，长度小于 1000。

输出

对每行字符串，如果括号配对，输出"YES"，否则输出"NO"。

样例输入

```
2
12{ab[8]}
12{34[78]ab}
```

样例输出

```
YES
NO
```

试题编号：20230318-6-02

试题类型：编程题

标准答案：

试题难度：一般

试题解析：

展示地址：[点击浏览](#)

考生答案：（此题已作答）

考生得分：0

是否评分：已评分

评价描述：

3. 扑克牌排序

假设这里有 36 张扑克牌，分别为 A1~A9,B1~B9,C1~C9,D1~D9，其中 A 代表方片，B 代表草花，C 代表红桃，D 代表黑桃，那么，设定如下的排序规则：

- 1.对于两张卡牌，X1Y1 与 X2Y2，X1 与 X2 表示 A~D，Y1 与 Y2 表示 1~9，如果 X1 与 X2 不同，那么依照 $D > C > B > A$ 的方式进行排序
- 2.假如有 X1 与 X2 相同时，那么就比较 Y1 与 Y2 的大小。

例如，对于如下的四张牌，有如下的升序排序结果：

D3，C4，A4，C1

升序排序的结果为 A4，C1，C4，D3

有人提出了如下的排序策略：

先建立 9 个队列，用于存放点数的大小，将卡牌依点数放入各自的队列之中，然后再按队列 1 到队列 9 依次出队。

例如，对于上面的结果，依次进队后，结果如下：

队列 1：C1；队列 3：D3，队列 4：C4，A4

将其依次出队后，结果为 C1，D3，C4，A4

然后，再建立 4 个队列，用于存放花色。将卡牌依花色 A~D 放入队列 1~4 中，然后再按队列 1 到队列 4 依次出队。

例如，对于上面刚刚出队的序列 C1，D3，C4，A4，将其依次进队，结果如下：

队列 1：A4；队列 3：C1，C4；队列 4：D3

将其依次出队后，结果为 A4，C1，C4，D3，排序结束。

请根据上面的算法，编写一个用队列对扑克牌排序的程序，要求依照上面的排序规则，根据先花色后点数的方法进行排序。

时间限制：1000
内存限制：65536

输入

输入分为两行，第一行为一个整数 n，表示一共有 n 张牌（ $1 \leq n \leq 100$ ） 第二行用 XY 的形式表示每一张牌，其中 X 为 A~D，Y 为 1~9

输出

输出三个部分 第一个部分为第一次进队出队的结果，用 Queue1:...表示，共 9 行，结果用空格分隔，下同 第二部分为第二次进队出队的结果，用 QueueA:...表示，共 4 行 三部分为一行，即将卡牌排序后的结果（升序排序）

样例输入

```
8
D8 A6 C3 B8 C5 A1 B5 D3
```

样例输出

```
Queue1:A1
Queue2:
Queue3:C3 D3
Queue4:
Queue5:C5 B5
Queue6:A6
Queue7:
```

```
Queue8:D8 B8
Queue9:
QueueA:A1 A6
QueueB:B5 B8
QueueC:C3 C5
QueueD:D3 D8
A1 A6 B5 B8 C3 C5 D3 D8
```

提示

第二次入队出队时，可以复用第一次时 9 个队列中的 4 个。所以其实只需要开辟 9 个队列即可。

运行测试

```
1
2
3
4
5
#include<bits/stdc++.h>
using namespace std;
int main(){
}
```

控制台:清理控制台

14:

试题编号 : 20230318-6-03

试题类型 : 编程题

标准答案 :

试题难度 : 一般

试题解析 :

展示地址 : [点击浏览](#)

考生答案 : (此题已作答)

考生得分 : 0

是否评分 : 已评分

评价描述 :

4. 滑动窗口

给定一个长度为 n ($n \leq 10^6$) 的数组。有一个大小为 k 的滑动窗口从数组的最左端移动到最右端。你可以看到窗口中的 k 个数字。窗口每次向右滑动一个数字的距离。

下面是一个例子：

数组是 `[1 3 -1 -3 5 3 6 7]`， $k = 3$ 。

窗口位置	最小值	最大值
<code>[1 3 -1]</code> <code>-3 5 3 6 7</code>	<code>-1</code>	<code>3</code>
<code>1 [3 -1 -3]</code> <code>5 3 6 7</code>	<code>-3</code>	<code>3</code>
<code>1 3 [-1 -3 5]</code> <code>3 6 7</code>	<code>-3</code>	<code>5</code>
<code>1 3 -1 [-3 5 3]</code> <code>6 7</code>	<code>-3</code>	<code>5</code>
<code>1 3 -1 -3 [5 3 6]</code> <code>7</code>	<code>3</code>	<code>6</code>
<code>1 3 -1 -3 5 [3 6 7]</code>	<code>3</code>	<code>7</code>

你的任务是得到滑动窗口在每个位置时的最大值和最小值。

时间限制：20000
内存限制：65536

输入

输入包括两行。第一行包括 n 和 k ，分别表示数组的长度和窗口的大小。第二行包括 n 个数字。

输出

输出包括两行。第一行包括窗口从左至右移动的每个位置的最小值。第二行包括窗口从左至右移动的每个位置的最大值。

样例输入

```
8 3
1 3 -1 -3 5 3 6 7
```

样例输出

-1 -3 -3 -3 3 3

3 3 5 5 6 7

试题编号：20230318-6-04

试题类型：编程题

标准答案：

试题难度：一般

试题解析：

展示地址：点击浏览

考生答案：（此题已作答）

考生得分：0

是否评分：已评分

评价描述：