

第一章 栈

【上机练习】

1、表达式括号匹配(stack)

【问题描述】

假设一个表达式有英文字母（小写）、运算符（+，-，*，/）和左右小（圆）括号构成，以“@”作为表达式的结束符。请编写一个程序检查表达式中的左右圆括号是否匹配，若匹配，则返回“YES”；否则返回“NO”。表达式长度小于 255，左圆括号少于 20 个。

【输入文件】

输入文件 stack.in 包括一行数据，即表达式，

【输出文件】

输出文件 stack.out 包括一行，即“YES” 或“NO”。

【输入输出样例】

【样例输入 1】	【样例输出 1】	【样例输入 2】	【样例输出 2】
2*(x+y)/(1-x)@	YES	(25+x)*(a*(a+b+b)@	NO

2、括弧匹配检验(check)

【问题描述】

假设表达式中允许包含两种括号：圆括号和方括号，其嵌套的顺序随意，如 ([] ()) 或 [([] [])] 等为正确的匹配，[(]) 或 ([] () 或 (()) 均为错误的匹配。

现在的问题是，要求检验一个给定表达式中的括弧是否正确匹配？

输入一个只包含圆括号和方括号的字符串，判断字符串中的括号是否匹配，匹配就输出“OK”，不匹配就输出“Wrong”。输入一个字符串：[([] [])]，输出：OK

【输入格式】

输入仅一行字符（字符个数小于 255）

【输出格式】

匹配就输出“OK”，不匹配就输出“Wrong”。

【输入样例】

[()]

【输出样例】

Wrong

3、字符串匹配问题(strs)

【问题描述】

字符串中只含有括号 (),[],<>,{},判断输入的字符串中括号是否匹配。如果括号有互相包含的形式，从内到外必须是<>,(),[],{}，例如。输入:[()] 输出:YES，而输入([]), ([{]}都应该输出 NO。

【输入格式】

文件的第一行为一个整数 n，表示以下有多少个由括号组成的字符串。接下来的 n 行，每行都是一个由括号组成的长度不超过 255 的字符串。

【输出格式】

在输出文件中有 n 行，每行都是 YES 或 NO。

【输入样例】

5
{ } < > () []
{ } { } < > < > () () [] []
{ } { } < > < > () () [] []
< > { } < > < > < > (< >) (< >) [(< >)] []
> < { } < > < > < > < > (< >) (< >) [(< >)] []

【输出标例】

YES
YES
YES
YES
NO

4、计算(calc)

【问题描述】

小明在你的帮助下，破密了 Ferrari 设的密码门，正要往前走，突然又出现了一个密码门，门上有一个算式，其中只有“(”，“)”，“0-9”，“+”，“-”，“*”，“/”，“^”，求出的值就是密码。小明数学学得不好，还需你帮他的忙。(“/”用整数除法)

【输入】

输入文件 calc.in 共 1 行，为一个算式。

【输出】

输出文件 calc.out 共 1 行，就是密码。

【输入输出样例】

calc.in	calc.out
1+(3+2)*(7^2+6*9)/(2)	258

【限制】

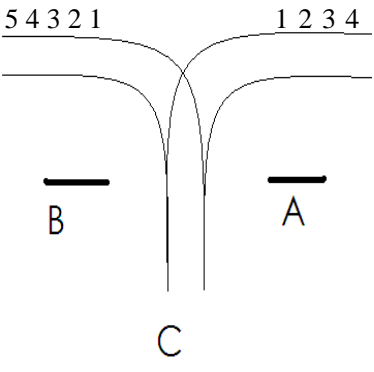
100%的数据满足：算式长度<=30 其中所有数据在 $2^{31}-1$ 的范围内。

5、车厢调度(train)

【问题描述】

有一个火车站，铁路如图所示，每辆火车从 A 驶入，再从 B 方向驶出，同时它的车厢可以重新组合。假设从 A 方向驶来的火车有 n 节 ($n \leq 1000$)，分别按照顺序编号为 1, 2, 3, ..., n。假定在进入车站前，每节车厢之间都不是连着的，并且它们可以自行移动到 B 处的铁轨上。另外假定车站 C 可以停放任意多节车厢。但是一旦进入车站 C，它就不能再回到 A 方向的铁轨上了，并且一旦当它进入 B 方向的铁轨，它就不能再回到车站 C。

负责车厢调度的工作人员需要知道能否使它以 a_1, a_2, \dots, a_n 的顺序从 B 方向驶出，请来判断能否得到指定的车厢顺序。



【输入】

输入文件的第一行为一个整数 n ，其中 $n \leq 1000$ ，表示有 n 节车厢，第二行为 n 个数字，表示指定的车厢顺序。

【输出】

如果可以得到指定的车厢顺序，则输出一个字符串“YES”，否则输出“NO”（注意要大写，不包含引号）。

【输入样例】

```
5
5 4 3 2 1
```

【输出样例】

```
YES
```

6、中缀表达式值(expr)**【问题描述】**

输入一个中缀表达式（由 0-9 组成的运算数、加+减-乘*除/四种运算符、左右小括号组成。注意“-”也可作为负数的标志，表达式以“@”作为结束符），判断表达式是否合法，如果不合法，请输出“NO”；否则请把表达式转换成后缀形式，再求出后缀表达式的值并输出。

注意：必须用栈操作，不能直接输出表达式的值。

【输入文件】

输入文件的第一行为一个以@结束的字符串。

【输出文件】

如果表达式不合法，请输出“NO”，要求大写。

如果表达式合法，请输出计算结果。

【输入样例】

```
1+2×8-9
```

【输出样例】

```
8
```