

7-3 Postfix Expression (25 分)

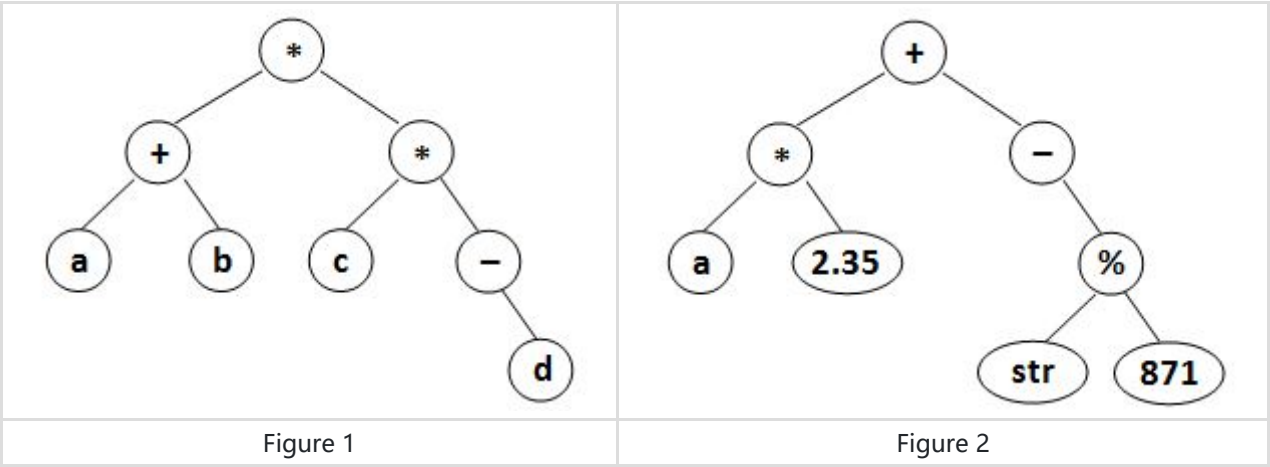
Given a syntax tree (binary), you are supposed to output the corresponding postfix expression, with parentheses reflecting the precedences of the operators.

Input Specification:

Each input file contains one test case. For each case, the first line gives a positive integer  $N$  ( $\leq 20$ ) which is the total number of nodes in the syntax tree. Then  $N$  lines follow, each gives the information of a node (the  $i$ -th line corresponds to the  $i$ -th node) in the format:

```
data left_child right_child
```

where `data` is a string of no more than 10 characters, `left_child` and `right_child` are the indices of this node's left and right children, respectively. The nodes are indexed from 1 to  $N$ . The NULL link is represented by  $-1$ . The figures 1 and 2 correspond to the samples 1 and 2, respectively.



Output Specification:

For each case, print in a line the postfix expression, with parentheses reflecting the precedences of the operators. There must be no space between any symbols.

Sample Input 1:

```
8
* 8 7
a -1 -1
* 4 1
+ 2 5
b -1 -1
d -1 -1
- -1 6
c -1 -1
```

Sample Output 1:

```
((a)(b)+)((c)((d))**)
```

Sample Input 2:

```
8
2.35 -1 -1
* 6 1
- -1 4
% 7 8
+ 2 3
a -1 -1
str -1 -1
871 -1 -1
```

Sample Output 2:

```
((a)(2.35)*)((str)(871)%)+)
```

作者: 陈越  
单位: 浙江大学  
时间限制: 400 ms  
内存限制: 64 MB  
代码长度限制: 16 KB