

Problema C

Abriu, fechou

Arquivo fonte: abriufechou.{c | cc | java | py2 | py3}

Autor: Prof. Rodrigo Bossini Tavares Moreira

Lembra do Charles? Não?! Ele é um apaixonado por máquinas de calcular. Já há muito tempo ele trabalha na implementação de uma dessas. Hoje, a máquina dele já é capaz de realizar operações aritméticas. Isso por conta da ajuda que ele recebeu de um grupo de alunos maratonistas. Depois dessa ajuda, ele percebeu que esse tipo de evento realmente revela talentos e, como de bobo não tem nada, resolveu pedir ajuda novamente para este seletor grupo. Desta vez, ele quer estender o funcionamento de seu equipamento. Ele gostaria que fosse possível utilizar os símbolos “(“ e “)”, claro, sem as aspas. A ideia é permitir que a precedência de operações seja alterada a gosto do usuário sem, entretanto, permitir que sejam utilizadas expressões inválidas. Lembre-se que uma expressão é válida se, para cada símbolo “(“ existe um símbolo “)”. Lembre-se, também, que a ordem importa. No momento, Charles pede apenas que o algoritmo implementado verifique a validade do uso de parênteses. Você pode ajudá-lo nesta tarefa?

Entrada

A primeira linha da entrada contém um número inteiro $1 \leq t \leq 100$. t linhas seguem. Cada linha possui uma sequência composta apenas por símbolos “(“ e “)”, sem as aspas. Cada uma delas possui, no máximo, 500 símbolos.

Saída

Para cada sequência de símbolos “(“ e “)” da entrada, seu programa deve produzir o texto “OK”, caso a sequência seja válida e “NOK” caso a sequência não seja válida, sempre sem as aspas. Não produza linha em branco no final.

Exemplo

OBS: Os exemplos de entrada possuem um espaço em branco entre cada par de símbolos, para facilitar a leitura do problema. Não há espaço em branco na entrada usada pelo ambiente da maratona.

Entrada

```
4
()
)(
((()))
(()(())
```

Saída

```
OK
NOK
OK
OK
```