

Balanco de Parênteses



(++)

Dada uma expressão qualquer com parênteses, indique se os parênteses estão corretos ou não, sem levar em conta o restante da expressão. Por exemplo:

$a+(b*c)-2-a$ está correto $(a+b*(2-c)-2+a)*2$ está correto

enquanto

$(a*b-(2+c)$ está incorreto $2*(3-a))$ está incorreto $)3+b*(2-c)($ está incorreto

Ou seja, todo parênteses que fecha deve ter um outro parênteses que abre correspondente e não pode haver parênteses que fecha sem um previo parenteses que abre e a quantidade total de parenteses que abre e fecha deve ser igual.

Obs: a estrutura de dados deve ser implementada de forma dinâmica.

Entrada

A entrada contém uma expressão de no máximo 1000 caracteres.

Saída

A saída deverá conter Correta se estiver correta e Incorreta e estiver incorreta.

Exemplo

Entrada	Saída
) 3+b* (2-c) (Incorreta

Fila do SUS



(++)

Os pacientes que chegam na fila do SUS passam por uma triagem imediatamente e vão para a fila de atendimento. Na triagem a enfermeira anota o horário de entrada do paciente e quantos minutos ele tem até que sua condição de saúde se torne crítica. Sabe-se que os pacientes são atendidos de 30 em 30 minutos (sempre nas horas cheias ou meias horas) quando na fila de atendimento. O início da triagem e do atendimento se dá às 7h da manhã, se não há nenhum paciente sendo atendido e a fila está vazia, o primeiro paciente é atendido no instante que chega na triagem. O médico atende até o último paciente na fila. A preocupação é se algum paciente atingiu uma condição crítica enquanto não tenha sido atendido. Para tanto você foi convidado para verificar na fila quantos pacientes atingem a condição crítica.

Entrada

A entrada começa com uma linha com o número inteiro N , $0 < N < 25$; o número de pacientes que chegam à triagem. A seguir são N linhas com os valores inteiros H , M e C , com $7 < H < 19$, e $0 \leq M < 60$, a hora e minuto que o paciente chega à triagem. O paciente da linha i sempre chega antes que, e no máximo junto com, o paciente da linha $i + 1$. E $0 \leq C \leq 720$ o número de minutos antes do paciente atingir a condição crítica de saúde.

Saída

Imprima o número de pacientes que atingiram a condição crítica ainda na fila de atendimento.

Exemplo

Entrada	Saída
4 7 0 20 7 0 30 7 30 20 8 15 30	1

Entrada	Saída
5 10 20 50 10 30 30 11 10 20 12 0 0 12 10 30	0

Trilhos



(++)

Há uma famosa estação de trem na cidade PopPush. Esta cidade fica em um país incrivelmente acidentado e a estação foi criada no último século. Infelizmente os fundos eram extremamente limitados naquela época. Foi possível construir somente uma pista. Além disso, devido a problemas de espaço, foi feita uma pista apenas até a estação.

A tradição local é que todos os comboios que chegam vindo da direção A continuam na direção B com os vagões reorganizados, de alguma forma. Suponha que o trem que está chegando da direção A tem $N \leq 1000$ vagões numerados sempre em ordem crescente 1,2, ..., N. O primeiro que chega é o 1 e o último que chega é o N. Existe um chefe de reorganizações de trens que quer saber se é possível reorganizar os vagões para que os mesmos saiam na direção B na ordem a_1, a_2, \dots .

O chefe pode utilizar qualquer estratégia para obter a saída desejada. Por exemplo se o chefe quer saber se a saída 5,4,3,2,1 é possível em B, nesse caso, basta o chefe deixar todos os vagões entrarem na estação (do 1 ao 5) e depois retirar um a um: retira o 5, retira o 4, retira o 3, retira o 2 e por último retira o 1. Desta forma a resposta seria Yes. Vagão que entra na estação só pode sair para a direção B e é possível incluir quantos forem necessários para retirar o primeiro vagão desejado.

Entrada

Na primeira linha de cada bloco há um inteiro N que é a quantidade de vagões. Em cada uma das próximas linhas de entrada haverá uma permutação dos valores 1,2,..., N.

Saída

A saída deve conter Yes se for possível e No se não for.

Exemplo

Entrada	Saída
5 5 4 3 2 1	Yes
Entrada	Saída
5 5 4 1 2 3	No