A. Roda Gigante (`ferriswheel`)

Limite de tempo: 1 segundos
Limite de memória: 1024 MiB

Na praça principal de Cesenatico existe uma roda gigante colorida, uma das atrações características da cidade. Durante o inverno, a roda foi desmontada e guardada, mas agora que o verão está a chegar, é finalmente altura de a montar de novo! As peças acabaram de chegar à praça e, com a tua ajuda, estamos prontos para montá-las todas.

À tua frente, tens N cabines individuais que precisam de ser ligadas umas às outras, de forma circular, para formar a roda gigante. As cabines estão numeradas de 0 a $N-1$, mas não necessariamente na ordem em que devem ser ligadas.

Cada cabine vem com uma junção especial que é usada para a ligar à cabine seguinte no sentido horário. Cada junção tem um de dois tipos possíveis:

- Tipo [+]: só pode ser usada para ligar a uma cabine com um número superior;
- Tipo [−]: só pode ser usada para ligar a uma cabine com um número inferior.

No exemplo abaixo, a cabine 2 tem uma junção do tipo [+]. Isto significa que a cabine seguinte no sentido horário deve ser a cabine 3 ou a cabine 4.

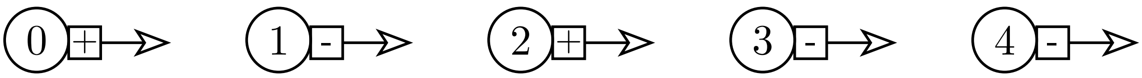

Figura 1: $N = 5$ e cinco cabines separadas, cada uma com uma junção do tipo [+] ou [−].

Recebes o número de cabines e os tipos das suas junções. A tua tarefa é determinar se é possível montar as N cabines numa roda gigante. Se sim, também tens de encontrar uma ordem na qual as cabines possam aparecer na roda.

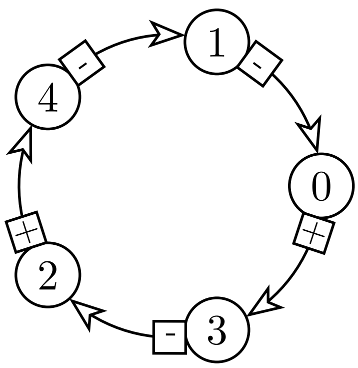

Figura 2: Uma roda gigante válida que pode ser montada a partir das cinco cabines mostradas acima.

A Figura 2 mostra uma roda gigante válida que pode ser montada a partir das cinco cabines mostradas na Figura 1.

Formalmente, uma ordem válida de cabines é uma sequência $C_0, C_1, \ldots, C_{N-1}$ de números com as seguintes propriedades.

- Cada número de 0 a $N-1$ aparece exatamente uma vez na sequência.
- Para cada $0 \le i \le N-2$, a cabine C_{i+1} deve satisfazer a condição imposta pela junção da cabine C_i. Ou seja, se o tipo da junção da cabine C_i for [+], então $C_{i+1} > C_i$; se for [−], então $C_{i+1} < C_i$.

- Adicionalmente, a cabine C_0 deve satisfazer a condição imposta pela junção da cabine C_{N-1}.

Entrada

A entrada consiste em duas linhas. A primeira linha contém um inteiro N, que indica o número de cabines.

A segunda linha contém uma string S de comprimento N, composta pelos caracteres '`+`' e '`-`'. Se $S_i =$ '`+`', então a cabine i tem uma junção do tipo [+]. Se $S_i =$ '`-`', então a cabine i tem uma junção do tipo [−].

Saída

Se não existir nenhuma ordem que satisfaça as restrições, escreve `NO`.

Caso contrário, escreve `YES`, seguido por uma linha com N inteiros, os índices das cabines na roda gigante no sentido horário, começando a partir de qualquer índice. Se existirem várias soluções, podes imprimir qualquer uma delas.

Restrições

- $3 \leq N \leq 300\,000$.
- $S_i =$ '`+`' ou '`-`'.

Pontuação

O teu programa será testado em vários casos de teste agrupados em subtarefas. Para obter a pontuação de uma subtarefa, tens de resolver corretamente todos os testes que ela contém.

- **Subtarefa 0** [**0 pontos**]: Exemplos.
- **Subtarefa 1** [**16 pontos**]: $N = 3$.
- **Subtarefa 2** [**13 pontos**]: Existe exatamente um '`+`' na string S.
- **Subtarefa 3** [**24 pontos**]: Os caracteres '`+`' e '`-`' alternam na string S; ou seja, para todo o $0 \leq i \leq N-2$, temos $S_i \neq S_{i+1}$.
- **Subtarefa 4** [**23 pontos**]: $N \leq 1000$.
- **Subtarefa 5** [**24 pontos**]: Sem restrições adicionais.

Exemplos

stdin	stdout
`3` `+++`	`NO`
`5` `+-+--`	`YES` `0 3 2 4 1`
`7` `------+`	`NO`
`8` `+-+-+-+-`	`YES` `3 2 4 6 7 1 0 5`
`11` `+++--+-++--`	`YES` `10 0 5 8 9 4 2 6 3 1 7`

Explicação

Primeiro Exemplo. Recebemos três cabines. Como todas as junções são do tipo [+], temos de organizar as cabines de modo a que cada cabine seja seguida por uma cabine com um número superior. Pode demonstrar-se que nenhuma ordem das três cabines satisfaz esta condição, portanto a resposta é `NO`.

Segundo Exemplo. Ver as Figuras 1 e 2 no enunciado do problema. Recebemos cinco cabines. Temos de as organizar no sentido horário de tal forma que:

- as cabines 0 e 2 (junção do tipo [`+`]) sejam imediatamente seguidas por uma cabine com um número superior;
- as cabines 1, 3 e 4 (junção do tipo [`-`]) sejam imediatamente seguidas por uma cabine com um número inferior.

Uma roda gigante que satisfaz todas estas condições é mostrada na figura abaixo. Para junções do tipo [`+`], os requisitos são cumpridos porque $0 < 3$ e $2 < 4$. Para junções do tipo [`-`], os requisitos são cumpridos porque $1 > 0$, $3 > 2$ e $4 > 1$. Existem várias saídas que correspondem a esta roda gigante: em vez de `0 3 2 4 1` podes também escrever `3 2 4 1 0`, `2 4 1 0 3`, `4 1 0 3 2`, ou `1 0 3 2 4`.

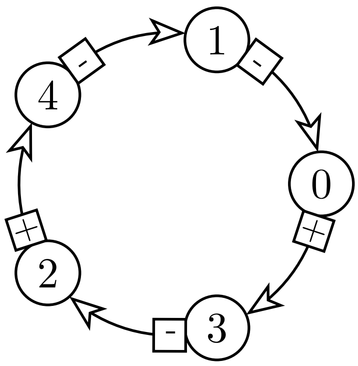

Figura 3: A roda gigante do segundo Exemplo (esta figura é idêntica à Figura 2).

No terceiro exemplo, recebemos sete cabines: todas as junções são do tipo [`-`], exceto a última, que é do tipo [`+`]. Assim, temos de organizar as cabines de modo a que cada cabine seja seguida por uma com um número inferior, exceto a cabine 6, que tem de ser seguida por uma cabine com um número superior. Pode demonstrar-se que não existe tal ordem, por isso a resposta é `NO`.

As figuras abaixo mostram as rodas gigantes que correspondem às duas últimas saídas de exemplo.

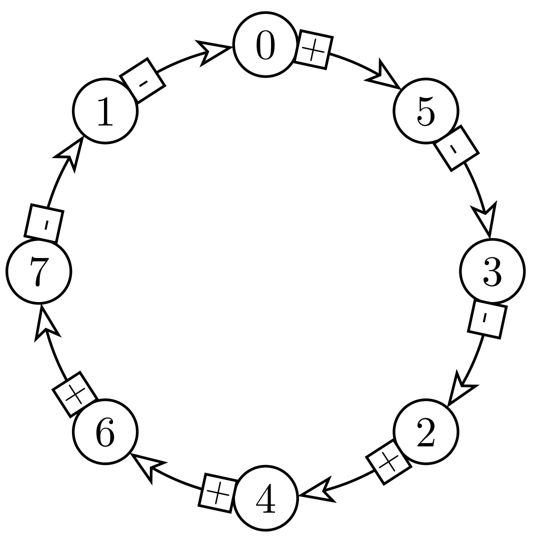

Figura 4: A roda gigante do quarto Exemplo.

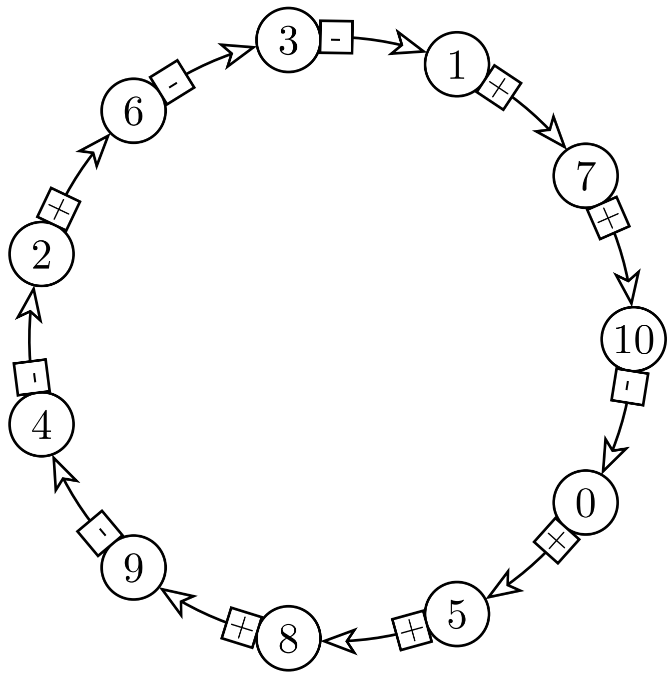

Figura 5: A roda gigante do quinto Exemplo.