

# Histórico de Comandos

Uma *interface por linha de comando* (ILC) é um dos tipos de interface humano-computador mais antigos que existem. Uma ILC permite a interação com o software através de um *interpretador de comandos*, sendo normalmente acessível em um terminal (ou janela) de texto. A vantagem de um interpretador de comandos é que ele permite que o usuário opere o sistema usando apenas o teclado. Ainda hoje em dia, em que estamos acostumados com interfaces gráficas sofisticadas, muitos aplicativos e sistemas operacionais incluem algum tipo de interface por linha de comando, e muitos usuários ainda preferem usá-la para grande parte das tarefas.

Um dos recursos mais úteis de um interpretador de comandos é o *histórico* de comandos. Quando um comando é digitado e executado, ele é colocado no histórico de comandos do terminal. O comando pode ser exibido novamente no terminal apertando a tecla `↑`; a tecla Enter executa o comando novamente quando o comando está sendo exibido no terminal. Todos os comandos executados são guardados no histórico: pressionar a tecla `↑` duas vezes exibe o penúltimo comando executado, pressioná-la três vezes exibe o antepenúltimo comando, e assim sucessivamente.

Por exemplo, se o histórico inicial é (A, B, C, D), para repetir o comando C basta pressionar duas vezes a tecla `↑`. O histórico será então atualizado para (A, B, C, D, C). Nesse ponto, para repetir o comando A será necessário pressionar cinco vezes a tecla `↑`; o histórico será atualizado para (A, B, C, D, C, A). Nesse ponto, para repetir mais uma vez o comando A basta pressionar uma vez a tecla `↑`; o histórico será atualizado para (A, B, C, D, C, A, A).

Leandro é administrador de sistemas e usa freqüentemente o interpretador de comandos para gerenciar remotamente os servidores que administra. Em geral, ele precisa apenas repetir comandos que já havia digitado antes. Enquanto estava trabalhando em um servidor, ele teve uma curiosidade: quantas vezes ele precisa pressionar a tecla `↑` para executar uma determinada sequência de comandos? Ele sabe quais são as posições no histórico dos comandos que ele necessita executar, mas não sabe resolver esse problema. Por isso, pediu que você fizesse um programa que respondesse à pergunta dele.

## Entrada

A entrada é composta de vários casos de teste. A primeira linha de cada caso de teste contém um número inteiro  $N$ , indicando o número de comandos que Leandro deseja executar ( $1 \leq N \leq 1.000$ ). A segunda linha de um caso de teste contém  $N$  inteiros  $P_1, P_2, \dots, P_N$ , que indicam as posições dos comandos no histórico ( $1 \leq P_i \leq 1.000.000$ ) no momento inicial, na ordem em que os comandos devem ser executados. Ou seja, o primeiro comando que deve ser executado está inicialmente na posição  $P_1$  do histórico; depois deve ser executado o comando que está inicialmente na posição  $P_2$  no histórico, e assim por diante, até  $P_N$ , que é a posição inicial do último comando que deve ser executado. Note que pode haver  $P_i = P_j$ .

As posições são dadas em função do número de vezes que a tecla `↑` deve ser pressionada: um comando na posição 5 necessita que a tecla `↑` seja pressionada cinco vezes antes de aparecer no terminal (note que à medida que comandos vão sendo executados, a posição de um dado comando no histórico pode mudar).

O final da entrada é indicado por  $N = 0$ .

*A entrada deve ser lida da entrada padrão.*

## Saída

Para cada caso de teste, seu programa deve imprimir apenas uma linha, contendo o número de vezes que Leandro precisa pressionar a tecla `↑` para executar todos os comandos.

*A saída deve ser escrita na saída padrão.*

## Exemplo

### Entrada:

```
3
2 5 3
4
2 1 4 3
5
1 2 3 4 5
4
1 3 1 3
0
```

### Saída

```
13
16
25
9
```