

Atletismo

Prova Fase 2 – OBI2024



Os Jogos Olímpicos de Paris 2024 acabaram de terminar e mais uma vez reviveram na população o interesse por esportes. Uma das modalidades que mais receberam atenção dos espectadores foi o atletismo, no qual a corrida de 100 metros é uma das principais provas.

O sucesso foi tanto que o estádio onde Tiago trabalha recebeu muitas inscrições para as aulas de atletismo. Tiago está empolgado com o novo grupo, mas teme que a grande quantidade de alunos torne difícil determinar os resultados das corridas.

Em uma corrida com N atletas, os atletas são numerados de 1 a N e o sistema automatizado do estádio é capaz de registrar a ordem na qual os atletas passaram pela linha de chegada. Vale ressaltar que este sistema é muito preciso, e portanto nunca há empate entre dois atletas. Tiago gostaria de usar essas informações para descobrir em qual posição cada atleta ficou no *ranking* da corrida.

Por exemplo, se $N = 6$ e a ordem em que os atletas cruzaram a chegada foi, do primeiro ao último,

5, 2, 4, 6, 3, 1

então o atleta com número 1 ficou na posição 6, o atleta com número 2 ficou na posição 2, o atleta com número 3 ficou na posição 5 e assim em diante. A tabela abaixo indica a posição de cada atleta.

| Número do atleta | 1 | 2 | 3 | 4 | 5 | 6 |
|---------------------------|---|---|---|---|---|---|
| Posição no <i>ranking</i> | 6 | 2 | 5 | 3 | 1 | 4 |

Dadas a quantidade N de atletas em uma corrida e a ordem em que os N atletas cruzaram a linha de chegada, sua tarefa é escrever um programa que determine, para cada um dos N atletas, a posição dele no *ranking* da corrida.

Entrada

A primeira linha de entrada contém um único inteiro N representando a quantidade de atletas que participaram da corrida.

As próximas N linhas contém cada uma um inteiro e representam a ordem em que os atletas cruzaram a linha de chegada, do primeiro ao último. Ou seja, a i -ésima dessas linhas contém o número do i -ésimo atleta a cruzar a linha de chegada.

Saída

Seu programa deverá imprimir N linhas, cada uma contendo um único inteiro. A i -ésima linha deverá conter a posição no *ranking* do atleta com número i .

Restrições

- $1 \leq N \leq 100\,000$
- Cada inteiro de 1 a N aparece exatamente uma vez na ordem de chegada

Informações sobre a pontuação

A tarefa vale 100 pontos. Estes pontos estão distribuídos em subtarefas, cada uma com suas **restrições adicionais** às definidas acima.

- **Subtarefa 1 (0 pontos):** Esta subtarefa é composta apenas pelos exemplos mostrados abaixo. Ela não vale pontos, serve apenas para que você verifique se o seu programa imprime o resultado correto para os exemplos.
- **Subtarefa 2 (40 pontos):** $N \leq 1000$.
- **Subtarefa 3 (60 pontos):** Sem restrições adicionais.

Seu programa pode resolver corretamente todas ou algumas das subtarefas acima (elas não precisam ser resolvidas em ordem). Sua pontuação final na tarefa é a soma dos pontos de todas as subtarefas resolvidas corretamente por qualquer uma das suas submissões.

Exemplos

| Exemplo de entrada 1 | Exemplo de saída 1 |
|----------------------|--------------------|
| 6 | 6 |
| 5 | 2 |
| 2 | 5 |
| 4 | 3 |
| 6 | 1 |
| 3 | 4 |
| 1 | |

Explicação do exemplo 1: Este é o exemplo dado no enunciado.

| Exemplo de entrada 2 | Exemplo de saída 2 |
|----------------------|--------------------|
| 3 | 3 |
| 2 | 1 |
| 3 | 2 |
| 1 | |

Explicação do exemplo 2: A corrida teve 3 atletas. O atleta com número 1 chegou em terceiro, o atleta com número 2 chegou em primeiro e o atleta com número 3 chegou em segundo.

| Exemplo de entrada 3 | Exemplo de saída 3 |
|----------------------|--------------------|
| 9 | 3 |
| 9 | 7 |
| 5 | 4 |
| 1 | 6 |
| 3 | 2 |
| 6 | 5 |
| 4 | 8 |
| 2 | 9 |
| 7 | 1 |
| 8 | |