

FPQLAEDW

Cartas do Jogo

Você e seus amigos estão jogando um jogo de tabuleiro com inúmeras cartas de poder, que podem ou não ser repetidas. Pelo manual, para o setup, você deve pegar o monte de cartas da caixa e, a partir dele, gerar um novo monte. Este novo monte, contém todas as cartas do primeiro, contudo este é formado da seguinte forma: no monte original da caixa, pegue uma carta da base. Se o valor de poder desta carta for maior do que a última carta inserida no novo monte, então essa deve ser incluída no final do novo monte. Por outro lado, se o valor de poder desta carta **não** for maior do que a última carta inserida no novo monte, então essa deve ser incluída no topo do novo monte. Sua tarefa é, dada a sequência de cartas, crie um programa que faça essa organização no menor tempo possível.

Input

Cada entrada é formada por um único caso de teste, sendo este formado por diversas linhas (presentando o monte de cartas da caixa). Cada linha contém um número inteiro ($1 \leq p \leq 10^5$) que apresenta o poder da carta. Contudo, não sabemos quantas cartas há no monte de cartas da caixa.

Output

Para cada caso de teste, seu programa deve imprimir a sequência de cartas do novo monte.

Restrições da Solução C

- Seu programa deve usar algum tipo de lista encadeada (use apenas as que estudamos), com alocação dinâmica de memória, composta por struct(s).
- Seu programa NÃO deve utilizar variáveis globais.
- GARANTA que não haverá desperdício de memória, seja por alocação extra, seja por não liberar a memória.
- Seu programa deve utilizar no mínimo duas funções e/ou procedimentos, criados por você.

Example

Input:

8
2
7
9
4
6
3
5

Output:

2
7
4
3
5
6
9
8

Example

Input:

4
5
8
7
6
9
3
2

Output:

4
5
6
2
3
9
7
8

Example

Input:

<https://justpaste.it/7axwd>

Output:

<https://justpaste.it/5ez6s>