

FECBE1953789456

Festa Estranha com Regra Esquisita

Restrição de Complexidade Assintótica na Solução: SIM

Tempo máximo: 60 segundos

Haverá uma festa no local conhecido como Espaço Infinito e diversas pessoas receberam convites numerados, sendo estes diferentes uns dos outros. No dia da festa, observou-se que o controle era realizado por um sistema digital. Esse sistema liberava a entrada, sempre, uma pessoa por vez e tomando nota de seu convite. Além disso, as pessoas adentravam ao evento na ordem de chegada. Por outro lado, a saída de pessoas seguia a mesma ordem de entrada, ou seja, o primeiro a entrar será o primeiro a sair e sempre uma pessoa por vez. Por questões de controle, de tempos em tempos e usando a informação do convite, o sistema convocava a pessoa com mais tempo de permanência na festa para se retirar. Neste momento, o sistema também informa a quantidade de pessoas que continua desfrutando do evento.

Entrada:

Cada entrada é formada por um único caso de teste. Cada caso de teste contém múltiplas linhas (L_i) e termina com o fim do arquivo (EOF). L_i denota a operação de entrada de pessoa, se somente se, ocorrer no seguinte formato: “Entrada **ConviteID**”, onde o primeiro termo é uma sequência constante e o segundo termo é numérico ($10^{-9} \leq \text{ConviteID} \leq 10^9$). Por outro lado, L_i denota a operação de saída de pessoa, se somente se, ocorrer no seguinte formato: “Saida”, onde o único termo é uma sequência constante.

Saída:

Para cada linha L_i que denota saída de pessoa, imprima uma linha no formato: “Saindo **ConviteID** e resta **N** pessoas”, onde **ConviteID** é o número que identifica a pessoa com mais tempo de permanência na festa e **N** é a quantidade de pessoas que continua a desfrutar do evento. Complementarmente, se não houver pessoas na festa, imprima apenas: “resta **0** pessoas”.

Restrições da Implementação C:

1. Restrição de Complexidade Assintótica: DEVE SER **$O(n)$** em qualquer operação.
2. A solução deve ser modelada usando a estrutura de dados lista encadeada com alocação dinâmica de memória, composta por struct(s).
3. Seu programa/código NÃO deve usar variáveis globais.
4. GARANTA que não haverá desperdício de memória.

Example

Input:

Saida
Entrada 1
Entrada 2

Saida
Entrada 3
Entrada 5
Saida
Saida
Entrada 6
Entrada 4
Entrada 7
Saida

Output:

resta 0 pessoas
Saindo 1 e resta 1 pessoas
Saindo 2 e resta 2 pessoas
Saindo 3 e resta 1 pessoas
Saindo 5 e resta 3 pessoas

Example

Input:

Entrada -42364
Entrada -83677
Entrada -85817
Entrada -59516
Entrada -74577
Entrada -51269
Saida
Entrada -71475
Entrada 79113
Saida
Entrada 83421
Entrada 5183
Entrada -78662
Saida

Output:

Saindo -42364 e resta 5 pessoas
Saindo -83677 e resta 6 pessoas
Saindo -85817 e resta 8 pessoas