

# Eu Posso Adivinhar a Estrutura de Dados em Java

Existe uma estrutura de dados do tipo sacola, suportando duas operações:

1  $x$

Jogue um elemento  $x$  na sacola.

2

Tire um elemento da sacola.

Dada uma sequência de operações que retornam valores, você vai adivinhar a estrutura de dados. É uma pilha (último-dentro, primeiro-fora), uma fila (primeiro-dentro, primeiro-fora), uma fila de prioridade (sempre tire os elementos grandes por primeiro) ou qualquer outra coisa que você dificilmente consegue imaginar!

## Entrada

Existem muitos casos de testes. Cada caso de teste começa com a linha contando um único inteiro  $n$  ( $1 \leq n \leq 1000$ ). Cada uma das seguintes  $n$  linhas é um comando do tipo 1, ou um número inteiro 2, seguido de um número inteiro  $x$ . Isso significa que depois de executar um comando do tipo 2, obtemos um elemento  $x$  *sem erros*. O valor de  $x$  é sempre um número inteiro, positivo e não maior do que 100. O final da entrada é determinado pelo final do arquivo (EOF). O tamanho do arquivo de entrada não excede 1MB.

## Saída

Para cada caso de teste, mostre um dos seguintes:

stack

É definitivamente uma pilha.

queue

É definitivamente uma fila.

priority queue

É definitivamente uma fila de prioridade.

impossible

Não pode ser uma pilha, uma fila ou uma fila de prioridade.

not sure

Pode ser mais de uma das três estruturas mencionadas acima.

Exemplo de Entrada	Exemplo de Saída
6	queue
1 1	not sure
1 2	impossible
1 3	stack
2 1	priority queue
2 2	
2 3	
6	
1 1	
1 2	
1 3	
2 3	
2 2	
2 1	
2	
1 1	
2 2	
4	
1 2	
1 1	
2 1	
2 2	
7	
1 2	
1 5	
1 1	
1 3	
2 5	
1 4	
2 4	

# Ordenando a Lista de Crianças do Papai Noel em C

Papai Noel está nos preparativos finais para a entrega dos presentes para as crianças do mundo todo pois o natal está chegando mais uma vez. Olhando suas novas listas de crianças que irão ganhar presentes neste ano ele percebeu que o duende estagiário (que havia ficado responsável por fazer as listas) não havia colocado os nomes em ordem alfabética.

Como o Papai Noel é um homem muito organizado ele deseja que cada lista de crianças possua, no seu final, o total de crianças que foram bem comportadas neste ano e um total das que não foram. Assim ele pode comparar a quantidade de crianças que se comportam este ano com as dos anos anteriores.

Para ajudar o bom velhinho, seu dever é criar um programa que leia todos os nomes da lista e imprima os mesmos nomes em ordem alfabética. No final da lista, você deve imprimir o total de crianças que foram e não foram comportadas neste ano.

## Entrada

A entrada é composta por vários nomes. O primeiro valor **N** ( $0 \leq N \leq 100$ ), indica quantos nomes tem na lista. As **N** linhas seguintes, contém um caractere especial correspondente ao comportamento da criança (+ indica que a criança foi bem comportada, - indica que a criança não foi bem comportada). Após o caractere especial, segue o nome da criança com no máximo 20 caracteres.

## Saída

Para cada lista de crianças, você deve imprimir os nomes em ordem alfabética. Após imprimir os nomes das crianças, você deve mostrar o total de crianças que se comportaram bem ou mal durante o ano.

Exemplos de Entrada	Exemplos de Saída
16 + Tininha + Dudinha - Carlinhos - Marquinhos + Joaozinho + Bruninha - Leandrinho - Fernandinha + Rafinha - Pedrinho + Aninha - Tamirinha - Gaguinho - Zezinho - Luquinhas + Julhinha	Aninha Bruninha Carlinhos Dudinha Fernandinha Gaguinho Joaozinho Julhinha Leandrinho Luquinhas Marquinhos Pedrinho Rafinha Tamirinha Tininha Zezinho Se comportaram: 7   Nao se comportaram: 9

Exemplos de Entrada	Exemplos de Saída
10 + Zezinho + Mangojatinha + Bruninha - Joaozinho - Fernandinha + Aninha - Luquinhas + Pedrinho - Tininha - Marquinhos	Aninha Bruninha Fernandinha Joaozinho Luquinhas Mangojatinha Marquinhos Pedrinho Tininha Zezinho Se comportaram: 5   Nao se comportaram: 5