

C. Aeroporto

time limit per test: 1 second

memory limit per test: 256 megabytes

O aeroporto de Brasília recebe todos os dias uma média de 400 poucos e decolagens. No último ano, foram realizados em torno de 146.000 movimentos aéreos.

Para organizar todo o fluxo de aviões que chegam e saem de lá, a torre de controle funciona o tempo inteiro com nível máximo de atenção. Para descartar qualquer possibilidade de erro humano, o chefe do controle de tráfego aéreo contratou você para desenvolver um programa que organize automaticamente o fluxo de aviões na entrada do campo de pouso. Para isso, basta seguir o seguinte protocolo: os aviões que veem do Norte da pista têm maior prioridade de serem colocados na fila, pois são aqueles que estão mais próximo do localizador (início da pista); em seguida, a ordem de prioridade dos aviões é Sul, Leste e Oeste. Assim, dada essa ordem de prioridades, os aviões são inseridos na fila 1 por vez, ou seja, insere-se 1 avião do lado Norte, depois um avião do lado Sul, em seguida 1 avião do lado Leste e por último 1 avião do lado Oeste, até que não haja mais aviões para serem inseridos na fila.

Dessa forma, o seu programa deve receber as informações de todos os aviões que chegarão ao aeroporto de Brasília e organizar a fila de aviões que irão poussar na pista.

Input

A entrada é composta por um único caso de testes. Nesse caso de testes, a primeira linha contém um character D, que poderá ser "N", "S", "L" ou "O", representando o ponto cardeal do avião que está solicitando entrar na fila de pouso. Em seguida, serão lidas um número desconhecido de linhas onde cada uma terá o perfil "BX", onde "B" é um character que indica se tratar de um voo, e "X" é um número inteiro que representa o identificador do avião. Sucessivamente, aparecerão linhas com o carácter de identificação de ponto cardeal, indicando que as linhas a seguir são referentes a aviões que estão solicitando entrar na fila de pouso por aquela direção. As informações de entrada terminam com uma linha contendo o carácter "0".

Output

A saída é composta de uma linha contendo as aeronaves enfileiradas pela ordem do protocolo estabelecido pelo aeroporto.

Examples

input	<input type="button" value="Copy"/>
N B1 B26 B38 B23 L B357 B2 B99 0 B3 B4 B8 B74 S B44 B6 0	

output	<input type="button" value="Copy"/>
B1 B44 B357 B3 B26 B6 B2 B4 B38 B99 B8 B23 B74	

input	<input type="button" value="Copy"/>
N B10 S B9 B8	

IDP - TAA - 2025/02
Private
Participant


→ About Group

Este grupo tem o objetivo de organizar as atividades de programação da disciplina de Técnicas de Programação e Análise de Algoritmos.
Group website

→ Group Contests	<input type="button" value="▼"/>
<ul style="list-style-type: none"> • TAA - LEA 05 • TAA - LEE 05 • TAA - LEA 04 • TAA - LEE 04 • TAA - AS 01 • TAA - LEA 03 • TAA - LEE 03 • TAA - LEA 02 • TAA - LEE 02 • TAA - LEA 01 • TAA - LEE 01 • ET - Exercício de Testes 	

TAA - LEE 03
Finished
Practice


→ Submit?
Language: <input type="button" value="GNU G++17 7.3.0"/> <input type="button" value="▼"/>

O
B7
L
B6
S
B5
B4
N
B3
B2
B1
0

output

B10 B9 B6 B7 B3 B8 B2 B5 B1 B4

Choose file:

Escolher Arquivo

Nenhu...colhido

Submit

→ Last submissions

Submission	Time	Verdict
339800489	Sep/21/2025 22:26	Accepted

[Codeforces](#) (c) Copyright 2010-2025 Mike Mirzayanov

The only programming contests Web 2.0 platform

Server time: Nov/11/2025 18:22:40^{UTC-3} (j2).

Desktop version, switch to [mobile version](#).

[Privacy Policy](#) | [Terms and Conditions](#)

Supported by



ITMO