

Bài A. GOODS

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 1 giây

- Xâu rỗng là một dãy ngoặc đúng
- Nếu A là dãy ngoặc đúng thì (A) , $[A]$, $\{A\}$ cũng là dãy ngoặc đúng
- Nếu A, B là 2 dãy ngoặc đúng thì AB cũng là dãy ngoặc đúng

Cho xâu S gồm các ký tự thuộc $\{ '(', ')', '[,]', '\{, \}' \}$. Đếm số đoạn con của S là dãy ngoặc đúng. Đoạn con của S là một dãy liên tiếp các ký tự của S . Hai đoạn con được gọi là khác nhau nếu điểm bắt đầu hoặc điểm kết thúc của hai đoạn khác nhau

Dữ liệu vào

- Một dòng duy nhất chứa xâu S

Kết quả

- Một số nguyên duy nhất – kết quả bài toán

Ví dụ

<code>stdin</code>	<code>stdout</code>
<code>()</code>	<code>1</code>

Hạn chế

- $1 \leq n \leq 10^6$

Bài B. XQUERY

File dữ liệu vào: **stdin**
File kết quả: **stdout**
Hạn chế thời gian: 2 giây

Cho S là một tập các số nguyên không âm (các số có thể xuất hiện nhiều lần). Ban đầu tập S rỗng, cần thực hiện Q truy vấn trên S , mỗi truy vấn thuộc một trong các dạng sau:

- 0 x : Thêm một số x vào S ($0 \leq x \leq 10^5$). Nếu x đã có trong S thì truy vấn này vẫn được thực hiện như thường.
- 1 x : Xóa một số x khỏi S ($0 \leq x \leq 10^5$). Nếu $x \notin S$ thì truy vấn này không cần làm gì. Nếu x xuất hiện nhiều lần trong S thì truy vấn này chỉ xóa đi 1.
- 2 a : Thay thế tất cả phần tử của S , x thay bằng $x \wedge a$ với \wedge là phép toán xor/nim/hoặc triet tiêu ($0 \leq a \leq 10^5$)
- 3 k : Tính tổng của k phần tử nhỏ nhất trong S ($0 \leq k \leq |S|$)

Dữ liệu vào

- Dòng đầu tiên chứa Q ($1 \leq Q \leq 10^5$)
- Q dòng tiếp theo mỗi dòng chứa một truy vấn: $t \ val$

Kết quả

Với mỗi truy vấn loại 3, hãy in ra tổng của k phần tử nhỏ nhất trong S trên một dòng

Ví dụ

stdin	stdout
6	6
0 1	1
2 2	
0 3	
3 2	
2 2	
3 1	

Hạn chế

- Có 15% số test với $Q \leq 1000$
- Có 15% số test không có truy vấn loại 2
- Có 20% số test với $k \leq 10$
- Có 50% số test không có ràng buộc gì thêm

Bài C. FROBOT

File dữ liệu vào: **stdin**
File kết quả: **stdout**
Hạn chế thời gian: **1s**

Trên lưới ô vuông n dòng m cột có 1 con robot đứng ở vị trí (x, y) và ngoảnh mặt về hướng bên phải (theo hướng ô $(1, 1) \rightarrow (1, m)$). Tại mỗi bước di chuyển của mình, robot có thể đi sang ô phía trước (mất k_1 năng lượng) hoặc ô trước phải (mất k_2 năng lượng); hoặc nó có thể quay phải (mất k_3 năng lượng), quay trái (mất k_4 năng lượng) và vẫn đứng trong ô đó. Một số ô của bảng có chứa vật cản và không thể đi vào. Nhiệm vụ đặc biệt của nó là phải đến được ô (z, t) , hãy giúp nó tìm cách di chuyển để hoàn thành nhiệm vụ và tốn ít năng lượng nhất có thể.

Dữ liệu vào

- Dòng đầu chứa 10 số nguyên dương $n, m, x, y, z, t, k_1, k_2, k_3, k_4$
- n dòng tiếp theo, mỗi dòng chứa m ký tự mô tả trạng thái vật cản của một dòng của lưới, ký tự 1/0 tương ứng có/không có vật cản

Dữ liệu đảm bảo ô (x, y) không có vật cản

Kết quả

Nếu không có cách di chuyển, in ra -1, ngược lại:

- Dòng đầu chứa một số nguyên không âm là năng lượng ít nhất cần có để robot hoàn thành nhiệm vụ
- Dòng thứ 2 chứa số nguyên không âm k : Số bước di chuyển trong cách di chuyển tìm được
- Dòng thứ 3 chứa k số nguyên dương mô tả di chuyển: 1/2/3/4 tương ứng là robot sang ô phía trước/sang ô trước phải/quay phải/quay trái

Ví dụ

stdin	stdout
5 5 1 1 5 5 1 1 1 10	9
00000	9
00000	1 2 2 3 1 3 3 3 2
00100	
00001	
00000	

Hạn chế

- $1 \leq n, m \leq 100, 1 \leq k_i \leq 10^9$
- Có 50% test với $k_i = 1$

Bài D. PRUN

File dữ liệu vào: `stdin`
File kết quả: `stdout`
Hạn chế thời gian: 1 giây

Với $p = p_1, p_2, \dots, p_n$ là một hoán vị của n số nguyên dương từ 1 đến n , gọi $\beta(p)$ là số đoạn trong cách chia p thành ít nhất các đoạn con tăng. Cho n và k , hãy đếm số hoán vị p của n số từ 1 đến n có $\beta(p) = k$

Dữ liệu vào

- Một dòng duy nhất chứa n k

Kết quả

- Ghi số lượng hoán vị tìm được sau khi chia lấy dư cho $10^9 + 7$

Ví dụ

stdin	stdout
4 2	11

Hạn chế

- $1 \leq n \leq 1000, 0 \leq k \leq n$
- Có 30% số test với $1 \leq n \leq 9$
- Có 30% số test với $9 < n \leq 20$