

BÀI TẬP NGÀY 10/02/2022

Bài 1: DISTANCE

Cho N điểm trên trục tọa độ một chiều OX. Với mỗi điểm i ở vị trí x_i sẽ có vận tốc là v_i .

Tại thời điểm t thì điểm i sẽ ở vị trí mới là $x_i + v_i * t$.

Với hai điểm i và j , gọi $d(i, j)$ là khoảng cách ngắn nhất giữa hai điểm xét trong mọi thời điểm. Bạn hãy tính hàm sau:

$$\text{Sum}(\sum_{i=1}^N \sum_{j=i+1}^N d(i, j))$$

Dữ liệu vào: Từ tệp văn bản **DISTANCE.INP** gồm:

- + Dòng đầu tiên là gồm duy nhất một số nguyên N ($1 \leq N \leq 10^5$).
- + Dòng tiếp theo gồm N số x_i ($1 \leq x_i \leq 10^8$).
- + Dòng cuối cùng gồm N số v_i ($-10^8 \leq v_i \leq 10^8$).

Kết quả: Ghi vào tệp văn bản **DISTANCE.OUT** một số duy nhất là kết quả của bài toán.

Ví dụ:

DISTANCE.INP	DISTANCE.OUT
5 2 1 4 3 5 2 2 2 3 4	19

Giải thích:

- Điểm 1 và điểm 2 mọi thời điểm có khoảng cách đều bằng nên khoảng cách ngắn nhất là 1.
- Điểm 1 và điểm 3 khoảng cách ngắn nhất là 2.
-
- Tổng khoảng cách là 19.

Subtask

- 50% số test có $1 \leq N \leq 10^3$.
- 50% số test còn lại theo giới hạn gốc bài toán.

Bài 2: ICECREAM

Ở cỗ máy bán kem tự động, mỗi que kem được bán với giá 50 cent và máy chỉ chấp nhận các loại đồng xu 50 cent, 1 USD và 2 USD (1 USD = 100 cent). Ban đầu, máy có M50 xu 50 cent, M1 xu 1 USD và M2 xu 2 USD. Tiền thối khi trả 1 USD chỉ được trả nếu máy có đồng 50 cent. Tiền thối khi trả 2 USD chỉ được trả khi máy có (a) một xu 1 USD và một xu 50 cent hoặc (b) ba xu 50 cent. Nếu cả hai trường hợp thỏa mãn, máy luôn chọn phương án (a). Nếu thiếu đồng xu để trả tiền thừa, cỗ máy sẽ không bán kem. Chỗ chứa xu của máy là hữu hạn, nên ở mọi thời điểm, không được có quá MMAX xu ở mỗi mệnh giá trong máy.

Có N học sinh đứng trước cỗ máy, và thầy giáo đi kèm cũng có rất nhiều đồng 50 cent, 1 USD, 2 USD. Nhiệm vụ của thầy giáo là phát cho mỗi bạn học sinh một đồng xu để mua đúng một que kem, nếu có dư tiền, học sinh sẽ trả cho thầy giáo, học sinh không trao đổi tiền với nhau.

Hãy đếm xem, thầy giáo có bao nhiêu cách phát xu? Hai cách phát được coi là khác nhau nếu tồn tại một học sinh nhận được đồng xu có mệnh giá khác nhau ở hai cách.

Dữ liệu vào: Từ tệp **ICECREAM.INP** gồm:

- + Dòng đầu ghi số N ($1 \leq N \leq 300$) và MMAX ($1 \leq \text{MMAX} \leq 10000$).
- + Dòng thứ hai ghi ba số nguyên, M50, M1 và M2 ($0 \leq \text{M50}, \text{M1}, \text{M2} \leq \text{MMAX}$) – số lượng các xu mệnh giá 50 cent, 1 USD, 2 USD có trong máy lúc ban đầu.

Kết quả: Ghi vào tệp **ICECREAM.OUT** ghi ra số cách phát tiền mod (10^9+9) .

Ví dụ:

ICECREAM.INP	ICECREAM.OUT	Giải thích
2 2 2 0 0	3	3 cách trả là 1, 50 1, 1 1, 2

ICECREAM.INP	ICECREAM.OUT	Giải thích
4 3 0 0 0	8	50, 50, 50, 1; 50, 50, 50, 2; 50, 50, 1, 50; 50, 50, 1, 1; 50, 50, 1, 2; 50, 1, 50, 50; 50, 1, 50, 1; 50, 1, 50, 2;

Subtask:

- 30% số test có $N \leq 15$, $M_{MAX} \leq 10$
- 35% số test khác có $16 \leq N \leq 50$
- 35% số test còn lại không có giới hạn gì thêm.

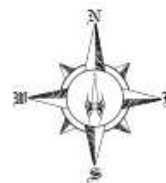
Bài 3: NAUTILUS

Nautilus là một con tàu ngầm bí mật đi vòng quanh các Đại dương. Bản đồ các Đại dương được biểu diễn dưới dạng một lưới có kích thước $R \times C$, với '#' là đảo '.' là nước và mô tả như hình sau đây:

```

...#...
..#.#.#
..#...##
.##...#
...#...

```



Cứ một phút, Tàu gửi tín hiệu cho biết hướng đi tiếp theo của Tàu. Hướng đi luôn là N(BẮC), S(NAM), E(ĐÔNG), W(TÂY).

Vytaytus đã thiết kế ra thiết bị Radar tiếp nhận các tín hiệu định kỳ của tàu ngầm Nautilus. Trong M phút cuối cùng thì Radar đã nhận được M tín hiệu được biểu diễn bởi các ký tự 'N', 'S', 'E', 'W', '?' tương ứng với các hướng đi của tàu và ký hiệu '?' chưa được giải mã bởi Radar.

Ví dụ như 'WS?EE??'

Vytautas không biết vị trí ban đầu của tàu ngầm, nhưng vẫn muốn dùng bản đồ biển để xác định vị trí hiện tại của nó. Cho biết tàu luôn ở dưới nước, hãy xác định số vị trí khác nhau tàu có thể ở tính tới hiện tại.

Dữ liệu vào: Từ tệp **NAUTILUS.INP** có cấu trúc sau:

- + Dòng 1 ghi số R, C, M
- + R dòng tiếp theo mỗi dòng ghi C kí tự '#' hoặc '.' Tương ứng với đảo hoặc mặt nước.
- + Dòng cuối cùng ghi chuỗi M kí tự {'N', 'S', 'E', 'W', '?' }.

Kết quả: Ghi vào file **NAUTILUS.OUT** số nguyên duy nhất là số vị trí có thể của tàu.

Ví dụ:

NAUTILUS.INP	NAUTILUS.OUT
5 9 7 ...##.... ..#.##..# ..##..#..#.... WS?EE??	22

Subtask:

- 29% test $1 \leq R, C, M \leq 100$ không có kí tự '?';
- 37% test $1 \leq R, C, M \leq 100$;
- 34% test có $1 \leq R, C \leq 500, 1 \leq M \leq 5000$