

Chú ý:

1. Các ngôn ngữ lập trình được sử dụng: C/C++, Java, Python 3
2. Giới hạn bộ nhớ cho tất cả các bài đều là 524288Kb

ĐỀ BÀI

BÀI 2. CHỌN ĐỒNG XU (Thời gian: 2s)

Cho A đồng xu vàng, B đồng xu bạc và C đồng xu bằng đồng ở trong hộp. Khi chưa có đủ 100 đồng xu nào đó cùng loại, ta thực hiện các thao tác sau:

Lấy ngẫu nhiên ra một đồng xu khỏi hộp (xác suất lấy được mỗi đồng xu là bằng nhau) rồi bỏ vào trong hộp 2 đồng xu cùng loại với nó. Hỏi số bước trung bình để có đủ được 100 đồng xu cùng loại bằng bao nhiêu?

Input

Chứa 3 số nguyên A, B, C ($0 \leq A, B, C \leq 99, A + B + C > 0$).

Output

In ra một số thực là giá trị kì vọng số thao tác sẽ được thực hiện, với độ chính xác 6 chữ số sau dấu phẩy.

Test ví dụ:

Input	Output
99 99 99	1.000000
99 99 98	1.331081
0 0 1	99.000000
46 28 96	7.115789

Giải thích test 2: Nếu thao tác đầu tiên chọn được đồng xu đồng, ta sẽ cần 2 bước thực hiện. Do vậy, số thao tác trung bình bằng $1 \times 99/(99+99+98) + 1 \times 99/(99+99+98) + 2 \times 98/(99+99+98) = 1.331081$

BÀI 3. XÂU AB (Thời gian: 2s)

Cho 3 số nguyên A, B và K. Trong các xâu có độ dài bằng A + B, gồm A kí tự 'a' và B kí tự 'b', hãy tìm xâu có thứ tự từ điển bằng K.

Input: Gồm 3 số nguyên A, B và K ($1 \leq A, B \leq 30, 1 \leq K \leq S$ với S là số lượng xâu có A kí tự 'a' và B kí tự 'b', S có giá trị không vượt quá phạm vi số nguyên 64 bit).

Output: In ra xâu tìm được

Test ví dụ:

Input	Output
2 2 4	baab

2 2 6	bbaa
26 26 371326006850843	babbbbbbbaaaabbbaabbabbbbbbbaaabaababbaaababbaaabaab

Giải thích test 1 và 2: Thứ tự 6 xâu lần lượt là: aabb, abab, abba, baab, baba, bbaa.

BÀI 4. TRÒ CHƠI VỚI CÁC ĐOẠN THẲNG (Thời gian: 2s)

Alice và Bob đang chơi một thể loại game tối ưu như sau:

Cho N đoạn thẳng có tọa độ $[L_i, R_i]$ trên trục Ox. 2 người lần lượt thay phiên nhau chơi, Alice đi trước. Mỗi lượt, một người chơi sẽ chọn ra một đoạn thẳng sao cho không giao nhau với bất kì đoạn nào đã được chọn. Ai không đi được nữa sẽ là người thua cuộc.

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test bắt đầu bởi số nguyên dương N ($1 \leq N \leq 100$).

N dòng tiếp theo, mỗi dòng gồm 2 số nguyên $L[i]$ và $R[i]$ ($1 \leq L[i] \leq R[i] \leq 100$).

Output:

Với mỗi test, hãy in ra người chiến thắng trên một dòng.

Test ví dụ:

Input	Output
4	Bob
3	Bob
50 90	Alice
8 45	Alice
10 49	
2	
10 20	
30 50	
1	
1 3	
10	
4 6	
5 6	
3 6	
4 4	
5 7	
6 8	
4 7	
5 9	
1 8	
5 9	

Giải thích test 1:

Ví dụ Alice chọn [10 49], Bob chọn [50 90], sau đó Alice không thể đi được nữa.

Hoặc Alice chọn [50 90] thì Bob chọn [8 45] hay [10 49] đều làm cho Alice hết lượt.

BÀI 5. CHỌN PHẦN TỬ TRÊN BẢNG (Thời gian: 2s)

Cho bảng số có kích thước $N \times N$ và số nguyên K , mỗi ô có một giá trị không âm $A[i][j]$. Nhiệm vụ của bạn là hãy chọn mỗi hàng, mỗi cột không quá K phần tử, sao cho tổng giá trị các ô được chọn là lớn nhất có thể.

Input:

Dòng đầu tiên chứa hai số nguyên N và K ($1 \leq N \leq 50$, $1 \leq K \leq N$).

N dòng tiếp theo, mỗi dòng gồm N số nguyên mô tả một hàng của bảng số ($0 \leq A[i][j] \leq 10^9$).

Output:

In ra một số nguyên là tổng các phần tử lớn nhất lựa chọn được.

Test ví dụ:

Input	Output
3 1 3 2 5 4 8 1 6 9 7	19
3 2 1 10 10 1 10 10 20 1 1	60

Giải thích test 1: Chọn các phần tử

..X

.X.

X..

Giải thích test 2: Chọn các phần tử

.XX

.XX

X..

BÀI 6. TRUY VẤN (Thời gian: 3s)

Cho dãy số $A[]$ có N phần tử. Có Q truy vấn dạng $L R$. Với mỗi truy vấn, hãy đếm xem trong dãy con $A[L]$, $A[L+1]$, ..., $A[R]$ có nhiều nhất bao nhiêu cặp hai phần tử có giá trị giống nhau?

Input:

Dòng đầu tiên là số nguyên N ($1 \leq N \leq 100000$).

Dòng tiếp theo gồm N số nguyên $A[i]$ ($1 \leq A[i] \leq N$).

Tiếp theo là số lượng truy vấn Q ($1 \leq Q \leq 1000000$).

Q dòng tiếp, mỗi dòng gồm 2 số nguyên L và R ($1 \leq L \leq R \leq N$).

Output:

Với mỗi truy vấn, hãy in ra đáp án tìm được trên một dòng.

Test ví dụ:

Input	Output
9	2
1 2 3 1 2 3 2 1 1	1
5	0
4 8	3
3 6	4
4 4	
1 6	
1 9	

Giải thích test 1:

Truy vấn 1: Có 2 cặp thỏa mãn $A[4] = A[8] = 1$ và $A[5] = A[7] = 2$

Truy vấn 4: Có nhiều nhất 3 cặp, ví dụ chọn (1, 4), (2, 5) và (3, 6).

Truy vấn 5: Có nhiều nhất 4 cặp, ví dụ chọn (1, 4), (2, 5), (3, 6) và (8, 9).

BÀI 7. NGÔI SAO (Thời gian: 2s)

Sau nhiều ngày quan sát các ngôi sao trên bầu trời, Alice đã liệt kê được $n+1$ ngôi sao. Để thuận lợi cho việc theo dõi và khảo sát, Alice đánh số các ngôi sao từ 0 đến n và xem xét các ngôi sao trên một hệ trục tọa độ Đề-các, trong đó, ngôi sao thứ i ($i = 0, 1, \dots, n$) có tọa độ nguyên $(x[i], y[i])$. Sau khi phân tích về mặt hình học, Alice đưa ra nhận định: Có lẽ các ngôi sao khác đang chi phối ngôi sao số 0. Alice định nghĩa một bộ ba ngôi sao sẽ chi phối ngôi sao số 0 nếu tọa độ ba ngôi sao đó tương ứng là ba đỉnh của một tam giác và tọa độ của ngôi sao số 0 nằm hoàn toàn bên trong tam giác đó.

Để tính toán sự chi phối của các ngôi sao với ngôi sao số 0, Alice muốn đếm số lượng bộ ba ngôi sao chi phối ngôi sao số 0.

Yêu cầu: Cho tọa độ $n+1$ ngôi sao, hãy đếm số lượng bộ ba ngôi sao chi phối ngôi sao số 0.

Input:

Dòng đầu tiên chứa số nguyên dương n ($n \leq 200000$);

Tiếp theo là $n+1$ dòng, mỗi dòng chứa hai số nguyên, lần lượt là tọa độ của các ngôi sao, từ ngôi sao số 0 đến ngôi sao thứ n .

Dữ liệu đảm bảo không có hai ngôi sao nào có cùng tọa độ và tọa độ các ngôi sao là số nguyên có trị tuyệt đối không vượt quá 10^9 .

Output:

In ra một số nguyên là số lượng bộ ba ngôi sao chỉ phối ngôi sao số 0.

Test ví dụ:

Input	Output
4 0 0 0 1 1 0 -1 -1 -1 0	1

BÀI 8. PHÂN HOẠCH CÂY (Thời gian: 5s)

Cho một đồ thị có dạng cây gồm n đỉnh, đỉnh thứ i ($1 \leq i \leq n$) có trọng số $w[i]$. Với một số nguyên k , Oliu muốn chia cây thành k vùng liên thông để phân có tổng trọng số nhỏ nhất là lớn nhất.

Input:

Dòng đầu chứa hai số nguyên n, k ($1 \leq k \leq n \leq 10^6$);

Dòng thứ hai chứa n số nguyên dương $w[1], w[2], \dots, w[n]$ ($w[i] \leq 10^6$);

Tiếp theo là $n-1$ dòng, mỗi dòng chứa hai số u, v mô tả một cạnh của cây.

Output:

Gồm một dòng chứa một số là tổng trọng số nhỏ nhất là lớn nhất.

Test ví dụ:

Input	Output
4 2 1 2 3 4 1 2 2 3 3 4	4

HẾT