

LUYỆN TẬP CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT – LẦN 3

CHIA VÀ TRỊ + STACK CƠ BẢN – QUEUE CƠ BẢN

A. SẮP XẾP KANGURU

Có N con kanguru trong vườn thú, con thứ i có chiều cao bằng $A[i]$. Con kanguru có chiều cao X có thể chứa được một con có chiều cao bằng Y trong túi của nó nếu như $X \geq 2 * Y$.

Một con đã chứa một con kanguru rồi, thì không thể nhảy vào túi một con kanguru khác.

Bầy Kanguru rất thích chơi trốn tìm, vì vậy chúng thường xuyên nhảy vào túi của nhau. Các bạn hãy tính toán xem trong trường hợp tối ưu, số con kanguru nhìn thấy trong vườn thú ít nhất bằng bao nhiêu?

Input:

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

Mỗi test gồm số nguyên N ($1 \leq N \leq 100\,000$).

Dòng tiếp theo gồm N số nguyên $A[i]$ ($1 \leq A[i] \leq 100\,000$).

Output:

Với mỗi test, in ra đáp án trên một dòng.

Ví dụ:

Input:	Output
2	5
8	5
2 5 7 6 9 8 4 2	
8	
9 1 6 2 6 5 8 3	

Giải thích test 1: Nhóm 2 – 5, 2 – 6, 4 – 8, 7, 9.

B. CẬP ĐIỂM GẦN NHẤT

Cho N điểm trên mặt phẳng tọa độ Oxy. Bạn cần tìm khoảng cách ngắn nhất giữa hai điểm trong số N điểm đã cho.

Input:

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

Mỗi test bắt đầu bởi một số nguyên N ($1 \leq N \leq 100\,000$).

N dòng tiếp theo, mỗi dòng gồm 2 số nguyên $X[i]$, $Y[i]$ ($-10^6 \leq X[i]$, $Y[i] \leq 10^6$).

Output:

Với mỗi test, in ra đáp án trên một dòng với độ chính xác 6 chữ số sau dấu phẩy.

Ví dụ:

Input:	Output
--------	--------

2	1.414214
6	1.000000
2 3	
12 30	
40 50	
5 1	
12 10	
3 4	
3	
0 0	
3 0	
4 0	

C. SỐ FIBONACCI THỨ N

Dãy số Fibonacci được xác định bằng công thức như sau:

$F[0] = 0, F[1] = 1;$

$F[n] = F[n-1] + F[n-2]$ với mọi $n \geq 2$.

Các phần tử đầu tiên của dãy số là 0, 1, 1, 2, 3, 5, 8, ...

Nhiệm vụ của bạn là hãy xác định số Fibonacci thứ n. Do đáp số có thể rất lớn, in ra kết quả theo modulo 10^9+7 .

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 1000$).

Mỗi test bắt gồm một số nguyên N ($1 \leq N \leq 10^9$).

Output:

Với mỗi test, in ra đáp án trên một dòng.

Ví dụ:

Input:	Output
3	1
2	8
6	6765
20	

D. LŨY THỪA MA TRẬN

Cho ma trận vuông A kích thước $N \times N$. Nhiệm vụ của bạn là hãy tính ma trận $X = A^K$ với K là số nguyên cho trước. Đáp số có thể rất lớn, hãy in ra kết quả theo modulo 10^9+7 .

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 100$).

Mỗi test bắt gồm một số nguyên N và K ($1 \leq N \leq 10, 1 \leq K \leq 10^9$) là kích thước của ma trận và số mũ.

Output:

Với mỗi test, in ra kết quả của ma trận X.

Ví dụ:

Input:	Output
2	8 5
2 5	5 3
1 1	597240088 35500972 473761863
1 0	781257150 154135232 527013321
3 1000000000	965274212 272769492 580264779
1 2 3	
4 5 6	
7 8 9	

E. TÍNH TỔNG BỘ BA

Cho dãy số $F[n]$ được xác định bằng công thức $F[n] = a^n + b^n + c^n$ với a, b, c là các số nguyên dương và $a < b < c$.

Cho biết $F[2], F[3], F[4]$, bạn hãy xác định a, b, c (dữ liệu đảm bảo tồn tại duy nhất một bộ ba a, b, c thỏa mãn). Sau đó, tính giá trị biểu thức $S = F[L] + F[L+1] + \dots + F[R]$ theo modulo 10^9+7 .

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 2500$).

Mỗi test gồm 5 số nguyên $F[2], F[3], F[4], L, R$.

$F[1] = a+b+c \leq 15000$.

$1 \leq F[2], F[3], F[4], L, R \leq 10^{15}$.

Output:

Với mỗi test, in ra đáp án tìm được trên một dòng.

Ví dụ:

Input:	Output
4	1070
14 36 98 5 6	72592824
49 251 1393 7 10	30124
14 36 98 6 9	1686433
49 251 1393 8 8	

Giải thích test 1: $a = 1, b = 2, c = 3$.

Giải thích test 3: $a = 2, b = 3, c = 6$.

F. TRÒ CHƠI VỚI CÁC CON SỐ

Tí có N miếng bìa và muốn viết lên mỗi miếng bìa một con số 1 hoặc 2 hoặc 3. Tí băn khoăn không biết liệu có bao nhiêu cách viết số sao cho tổng các số trên tất cả N miếng bìa sẽ chia hết cho 3. Bạn tính giúp Tí nhé

Input

Có duy nhất một dòng chứa số nguyên dương N không quá 10^{18}

Output

Ghi ra số nguyên dương là số cách viết thỏa mãn (chia lấy dư cho $10^9 + 7$)

Ví dụ

Input	Output
2	3

G. ĐẾM XÂU NHỊ PHÂN

Cho số nguyên dương N không quá 10^5 . Bạn hãy đếm xem có bao nhiêu xâu nhị phân độ thỏa mãn một trong hai điều kiện sau:

- Tồn tại ít nhất 2 chữ số 1 liên tiếp
- Tồn tại ít nhất 3 xâu nhị phân liên tiếp giống nhau

Ví dụ với N=3 thì xâu 000 và 011 là thỏa mãn, xâu 010 không thỏa mãn

Input

Dòng đầu tiên ghi số bộ test T (không quá 10)

Mỗi bộ test ghi trên một dòng duy nhất một số nguyên dương N

Output

Gồm T dòng, mỗi dòng ghi kết quả tìm được của một bộ test (kết quả chia phần dư cho 1000).

Ví dụ

Input	Output
2	0
1	1
2	

H. NGĂN XẾP 1

Cho một ngăn xếp các số nguyên. Các thao tác được mô tả trong file văn bản gồm 3 lệnh: push, pop và show. Trong đó thao tác push kèm theo một giá trị cần thêm. Hãy viết chương trình ghi ra kết quả của các lệnh show.

Dữ liệu vào: Gồm nhiều dòng, mỗi dòng chứa một lệnh push, pop hoặc show.

Kết quả: Ghi ra màn hình các phần tử đang có trong stack theo thứ tự lưu trữ mỗi khi gặp lệnh show. Các số viết cách nhau đúng một khoảng trống. Nếu trong stack không còn gì thì in ra dòng “empty”

Ví dụ:

Input	Output
push 3	3 5

push 5	3 5 7
show	3
push 7	
show	
pop	
pop	
show	

I. NGĂN XẾP 2

Yêu cầu bạn xây dựng một stack với các truy vấn sau đây:

“PUSH x”: Thêm phần tử x vào stack ($0 \leq x \leq 1000$).

“PRINT”: In ra phần tử đầu tiên của stack. Nếu stack rỗng, in ra “NONE”.

“POP”: Xóa phần tử đầu tiên của stack. Nếu stack rỗng, không làm gì cả.

Input:

Dòng đầu tiên là số lượng truy vấn Q ($Q \leq 100000$).

Mỗi truy vấn có dạng như trên.

Output:

Với mỗi truy vấn “PRINT”, hãy in ra phần tử đầu tiên của stack. Nếu stack rỗng, in ra “NONE”.

Ví dụ:

Input:	Output
9	1
PUSH 1	3
PUSH 2	NONE
POP	
PRINT	
PUSH 3	
PRINT	
POP	
POP	
PRINT	

J. BIỂU THỨC HẬU TỔ 1

Hãy sử dụng ngăn xếp để thực hiện việc chuyển các biểu thức trung tố sang hậu tố

Dữ liệu vào: Dòng 1 ghi số N là số biểu thức trung tố (đúng khuôn dạng) chỉ bao gồm các phép cộng, trừ, nhân, chia, các chữ cái thường từ a đến z và các dấu ngoặc đơn. N dòng tiếp theo mỗi dòng ghi một biểu thức.

Kết quả: Ghi ra màn hình các biểu thức hậu tố kết quả.

Ví dụ:

Input	Output
1 ((a+b) * (c+d))	ab+cd+*

K. BIỂU THỨC HẬU TỔ 2

Cho một dãy biểu thức dạng trung tố có thể chưa đúng khuôn dạng với các phép toán cộng trừ nhân chia, lũy thừa.

Nhiệm vụ của bạn là hãy chuyển biểu thức đã cho về dạng hậu tố.

Chú ý xem ví dụ để hiểu sự khác biệt với bài trước.

Input:

Dòng đầu tiên là số lượng bộ test ($T \leq 100$).

Mỗi dòng gồm một xâu S, không quá 50 kí tự. Các số trong biểu thức đại diện bởi một chữ cái thường.

Output:

Với mỗi test, in ra biểu thức hậu tố tìm được.

Ví dụ:

Input:	Output
2 a+b*c+d a+b*(c^d-e)^(f+g*h)-i	abc*+d+ abcd^e-fgh*+^*+i-

L. DÃY NGOẶC ĐÚNG ĐẠI NHẤT

Cho một xâu chỉ gồm các kí tự '(' và ')'. Một dãy ngoặc đúng được định nghĩa như sau:

- Xâu rỗng là 1 dãy ngoặc đúng.
- Nếu A là 1 dãy ngoặc đúng thì (A) là 1 dãy ngoặc đúng.
- Nếu A và B là 2 dãy ngoặc đúng thì AB là 1 dãy ngoặc đúng.

Cho một xâu S. Nhiệm vụ của bạn là hãy tìm dãy ngoặc đúng dài nhất xuất hiện trong xâu đã cho.

Input:

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

Mỗi test gồm một xâu S có độ dài không vượt quá 10^5 kí tự.

Output:

Với mỗi test in ra một số nguyên là độ dài dãy ngoặc đúng dài nhất tìm được.

Ví dụ:

Input:	Output
3 ((()) () () () (())))	2 4 6

M. KIỂM TRA DÃY NGOẶC ĐÚNG

Cho một xâu chỉ gồm các kí tự ‘(’, ‘)’, ‘[’, ‘]’, ‘{’, ‘}’. Một dãy ngoặc đúng được định nghĩa như sau:

- Xâu rỗng là 1 dãy ngoặc đúng.
- Nếu A là 1 dãy ngoặc đúng thì (A), [A], {A} là 1 dãy ngoặc đúng.
- Nếu A và B là 2 dãy ngoặc đúng thì AB là 1 dãy ngoặc đúng.

Cho một xâu S. Nhiệm vụ của bạn là xác định xâu S có là dãy ngoặc đúng hay không?

Input:

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

Mỗi test gồm 1 xâu S có độ dài không vượt quá 100 000.

Output:

Với mỗi test, in ra “YES” nếu như S là dãy ngoặc đúng, in ra “NO” trong trường hợp ngược lại.

Ví dụ:

Input:	Output
2	YES
[()] { } { [() ()] () }	NO
[()]	

N. SỬA LẠI DÃY NGOẶC

Cho một xâu chỉ gồm các kí tự ‘(’, ‘)’ và có độ dài chẵn. Hãy đếm số lượng dấu ngoặc cần phải đổi chiều ít nhất, sao cho xâu mới thu được là một dãy ngoặc đúng.

Input:

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

Mỗi test gồm 1 xâu S có độ dài không vượt quá 100 000, chỉ gồm dấu (và).

Output:

Với mỗi test, in ra đáp án tìm được trên một dòng.

Ví dụ:

Input:	Output
4	2
)) ((2
((((1
(((())	3
) (()) (((

O. XÓA DẤU NGOẶC

Cho biểu thức toán học đúng, bạn cần tìm tất cả các biểu thức đúng có thể bằng cách xóa bỏ các cặp dấu ngoặc tương ứng với nhau từ biểu thức ban đầu.

Ví dụ: Cho biểu thức: $(2+(2*2)+2)$ Các biểu thức tìm được:

$$(2+2*2+2)$$

$$2+(2*2)+2$$

$$2+2*2+2$$

Các biểu thức $(2+2*2)+2$ và $2+(2*2+2)$ không được chấp nhận vì không xóa đi các cặp dấu ngoặc tương ứng với nhau

Dữ liệu vào: Một dòng chứa biểu thức gồm các số nguyên không âm, các dấu +, -, *, / và dấu ngoặc đơn.

Biểu thức không quá 200 kí tự, có chứa ít nhất 1 và không quá 10 cặp dấu ngoặc.

Kết quả: In ra tất cả các biểu thức khác nhau thỏa mãn đầu bài theo thứ tự từ điển

Ví dụ

Input	Output
$(1 + (2 * (3 + 4)))$	$(1 + (2 * 3 + 4))$ $(1 + 2 * (3 + 4))$ $(1 + 2 * 3 + 4)$ $1 + (2 * (3 + 4))$ $1 + (2 * 3 + 4)$ $1 + 2 * (3 + 4)$ $1 + 2 * 3 + 4$

P. CẤU TRÚC DỮ LIỆU HÀNG ĐỢI 1

Ban đầu cho một queue rỗng. Bạn cần thực hiện các truy vấn sau:

1. Trả về kích thước của queue
2. Kiểm tra xem queue có rỗng không, nếu có in ra “YES”, nếu không in ra “NO”.
3. Cho một số nguyên và đẩy số nguyên này vào cuối queue.
4. Loại bỏ phần tử ở đầu queue nếu queue không rỗng, nếu rỗng không cần thực hiện.
5. Trả về phần tử ở đầu queue, nếu queue rỗng in ra -1.
6. Trả về phần tử ở cuối queue, nếu queue rỗng in ra -1.

Dữ liệu vào

Dòng đầu tiên chứa số nguyên T là số bộ dữ liệu, mỗi bộ dữ liệu theo dạng sau.

Dòng đầu tiên chứa số nguyên n - lượng truy vấn ($1 \leq n \leq 1000$)

N dòng tiếp theo, mỗi dòng sẽ ghi loại truy vấn như trên, với truy vấn loại 3 sẽ có thêm một số nguyên, không quá 10^6 .

Kết quả: In ra kết quả của các truy vấn..

Ví dụ:

Input	Output
-------	--------

1	1
1 4	3
3 1	5
3 2	2
3 3	
5	
6	
4	
4	
4	
4	
4	
3 5	
3 6	
5	
1	

Q. CẤU TRÚC DỮ LIỆU HÀNG ĐOỊ 2

Yêu cầu bạn xây dựng một queue với các truy vấn sau đây:

“PUSH x”: Thêm phần tử x vào cuối của queue ($0 \leq x \leq 1000$).

“PRINTFRONT”: In ra phần tử đầu tiên của queue. Nếu queue rỗng, in ra “NONE”.

“POP”: Xóa phần tử ở đầu của queue. Nếu queue rỗng, không làm gì cả.

Input:

Dòng đầu tiên là số lượng truy vấn Q ($Q \leq 100000$).

Mỗi truy vấn có dạng như trên.

Output:

Với mỗi truy vấn “PRINT”, hãy in ra phần tử đầu tiên của queue. Nếu queue rỗng, in ra “NONE”.

Test ví dụ:

Input:	Output
9	2
PUSH 1	2
PUSH 2	NONE
POP	
PRINTFRONT	
PUSH 3	
PRINTFRONT	
POP	
POP	
PRINTFRONT	

R. ĐIỂM XUẤT PHÁT

Có N thành phố liên kết với nhau tạo thành một vòng tròn và các đường đi của chúng chỉ là một chiều, khoảng cách từ thành phố thứ i tới i+1 (trạm đến tiếp theo) bằng D[i]. Bạn có một chiếc xe tải có khả năng chứa rất nhiều (vô hạn xăng), tuy nhiên lại rất ngắn xăng, tận 1 lít xăng / 1 đơn vị chiều dài. Tại mỗi thành phố có một trạm xăng có lượng xăng bằng A[i] lít.

Mục tiêu của bạn là có một hành trình đi hết N thành phố rồi quay lại đúng điểm xuất phát. Tuy nhiên, việc lựa chọn điểm xuất phát là không hề dễ dàng, vì nếu không tối ưu, có thể bạn sẽ bị hết xăng khi đi giữa đường.

Nhiệm vụ của bạn là xác định xem có thể tìm được điểm xuất phát sao cho có thể đi trọn vẹn một vòng quanh các thành phố hay không?

Input:

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

Mỗi test gồm một số nguyên N ($N \leq 100\,000$).

N dòng tiếp theo, mỗi dòng gồm 2 số nguyên A[i] và D[i] ($1 \leq A[i], D[i] \leq 10^6$).

Output:

Với mỗi test, in ra “YES” nếu có thể đi một vòng quanh N thành phố, in ra “NO” trong trường hợp ngược lại.

Test ví dụ:

Input:	Output
3	YES
3	YES
6 4	NO
3 6	
7 3	
4	
5 5	
5 5	
5 5	
5 5	
2	
1 2	
2 3	

S. HÀNG ĐỢI HAI ĐẦU (DEQUEUE)

Yêu cầu bạn xây dựng một hàng đợi hai đầu với các truy vấn sau đây:

“PUSHFRONT x”: Thêm phần tử x vào đầu của dequeue ($0 \leq x \leq 1000$).

“PRINTFRONT”: In ra phần tử đầu tiên của dequeue. Nếu dequeue rỗng, in ra “NONE”.

“POPFRONT”: Xóa phần tử đầu của dequeue. Nếu dequeue rỗng, không làm gì cả.

“PUSHBACK x”: Thêm phần tử x vào cuối của dequeue ($0 \leq x \leq 1000$).

“PRINTBACK”: In ra phần tử cuối của dequeue. Nếu dequeue rỗng, in ra “NONE”.

“POPBACK”: Xóa phần tử cuối của dequeue. Nếu dequeue rỗng, không làm gì cả.

Input:

Dòng đầu tiên là số lượng truy vấn Q ($Q \leq 100000$).

Mỗi truy vấn có dạng như trên.

Output:

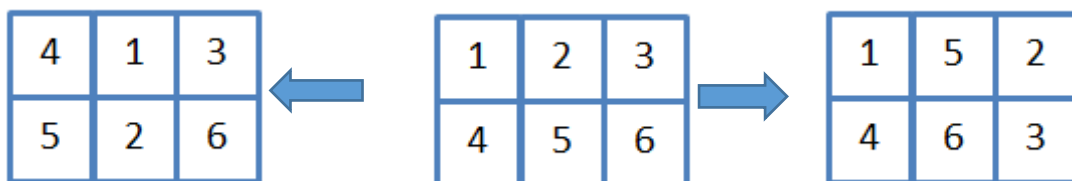
Với mỗi truy vấn “PRINTFRONT” và “PRINTBACK”, hãy in ra kết quả trên một dòng.

Test ví dụ:

Input:	Output
10	2
PUSHBACK 1	1
PUSHFRONT 2	3
PUSHBACK 3	NONE
PRINTFRONT	
POPFRONT	
PRINTFRONT	
POPFRONT	
PRINTBACK	
POPFRONT	
PRINTBACK	

T. QUAY HÌNH VUÔNG

Có một chiếc bảng hình chữ nhật với 6 miếng ghép, trên mỗi miếng ghép được điền một số nguyên trong khoảng từ 1 đến 6. Tại mỗi bước, chọn một hình vuông (bên trái hoặc bên phải), rồi quay theo chiều kim đồng hồ.



Yêu cầu: Cho một trạng thái của bảng, hãy tính số phép biến đổi ít nhất để đưa bảng đến trạng thái đích.

Input:

Dòng đầu tiên chứa 6 số là trạng thái bảng ban đầu (thứ tự từ trái qua phải, dòng 1 tới dòng 2).

Dòng thứ hai chứa 6 số là trạng thái bảng đích (thứ tự từ trái qua phải, dòng 1 tới dòng 2).

Output:

In ra một số nguyên là đáp số của bài toán.

Ví dụ:

Input	Output
1 2 3 4 5 6 4 1 2 6 5 3	2