

CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT CONTEST 8 – HÀNG ĐỘI

BÀI 1. CẤU TRÚC DỮ LIỆU HÀNG ĐỘI 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	

BÀI 2. CẤU TRÚC DỮ LIỆU HÀNG ĐỘI 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.

Dữ liệu vào:

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

Mỗi truy vấn có dạng như trên.

Kết quả:

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

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

BÀI 3. 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.

Dữ liệu vào:

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

Mỗi truy vấn có dạng như trên.

Kết quả:

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

Ví dụ:

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

BÀI 4. GIÁ TRỊ NHỎ NHẤT CỦA XÂU

Cho chuỗi ký tự $S[]$ bao gồm các ký tự in hoa $[A, B, \dots, Z]$. Ta định nghĩa giá trị của chuỗi $S[]$ là tổng bình phương số lần xuất hiện mỗi ký tự trong chuỗi. Ví dụ với chuỗi $S[] = \text{"AAABBCD"}$ ta có $F(S) = 3^2 + 2^2 + 1^2 + 1^2 = 15$. Hãy tìm giá trị nhỏ nhất của chuỗi $S[]$ sau khi loại bỏ K ký tự trong chuỗi.

Input:

- Dòng đầu tiên đưa vào số lượng test T ($T \leq 100$).
- Mỗi test được tổ chức thành 2 dòng. Dòng thứ nhất ghi lại số K . Dòng thứ 2 ghi lại chuỗi ký tự $S[]$ có độ dài không vượt quá 10^6 .

Output:

- Đưa ra giá trị nhỏ nhất của mỗi test theo từng dòng.

Input	Output
2	6
0	3
ABCC	
1	
ABCC	

BÀI 5. SỐ NHỊ PHÂN TỪ 1 ĐẾN N

Cho số tự nhiên n . Hãy in ra tất cả các số nhị phân từ 1 đến n .

Input:

- Dòng đầu tiên ghi lại số lượng test T ($T \leq 100$).
- Mỗi test là một số tự nhiên n được ghi trên một dòng ($n \leq 10000$).

Output:

- Đưa ra kết quả mỗi test trên một dòng.

Ví dụ:

Input	Output
2	1 10
2	1 10 11 100 101 110
5	

BÀI 6. SỐ 0 VÀ SỐ 9

Cho số tự nhiên N . Hãy tìm số nguyên dương X nhỏ nhất được tạo bởi số 9 và số 0 chia hết cho N . Ví dụ với $N = 5$ ta sẽ tìm ra $X = 90$.

Input:

- Dòng đầu tiên ghi lại số lượng test T ($T \leq 100$).
- Những dòng kế tiếp mỗi dòng ghi lại một test. Mỗi test là một số tự nhiên N được ghi trên một dòng ($N \leq 100$).

Output:

- Đưa ra theo từng dòng số X nhỏ nhất chia hết cho N tìm được.

Ví dụ:

Input	Output
2	90
5	9009
7	

BÀI 7. SỐ BDN 1

Ta gọi số nguyên dương K là một số BDN nếu các chữ số trong K chỉ bao gồm các 0 hoặc 1 có nghĩa. Ví dụ số $K = 1, 10, 101$. Cho số tự nhiên N ($N < 2^{63}$). Hãy cho biết có bao nhiêu số BDN nhỏ hơn N . Ví dụ $N=100$ ta có 4 số BDN bao gồm các số: 1, 10, 11, 100.

Input:

- Dòng đầu tiên ghi lại số tự nhiên T là số lượng Test;
- T dòng kế tiếp mỗi dòng ghi lại một bộ Test. Mỗi test là một số tự nhiên N .

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
3	2
10	4
100	7
200	

BÀI 8. SỐ BDN 2

Ta gọi số nguyên dương K là một số BDN nếu các chữ số trong K chỉ bao gồm các 0 hoặc 1 có nghĩa. Ví dụ số $K = 101$ là số BDN, $k=102$ không phải là số BDN.

Số BDN của N là số $P = M \times N$ sao cho P là số BDN. Cho số tự nhiên N ($N < 1000$), hãy tìm số BDN nhỏ nhất của N .

Ví dụ. Với $N=2$, ta tìm được số BDN của N là $P = 5 \times 2 = 10$. $N = 17$ ta tìm được số BDN của 17 là $P = 653 \times 17 = 11101$.

Input:

- Dòng đầu tiên ghi lại số tự nhiên T là số lượng Test;
- T dòng kế tiếp mỗi dòng ghi lại một bộ Test. Mỗi test là một số tự nhiên N .

Output:

Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
3	10
2	11100
12	11101
17	

BÀI 9. BIẾN ĐỔI S – T

Cho hai số nguyên dương S và T ($S, T < 10000$) và hai thao tác (a), (b) dưới đây:

Thao tác (a): Trừ S đi 1 ($S = S - 1$);

Thao tác (b): Nhân S với 2 ($S = S * 2$);

Hãy dịch chuyển S thành T sao cho số lần thực hiện các thao tác (a), (b) là ít nhất. Ví dụ với $S = 2, T = 5$ thì số các bước ít nhất để dịch chuyển S thành T thông qua 4 thao tác sau:

Thao tác (a): $2 * 2 = 4$;

Thao tác (b): $4 - 1 = 3$;

Thao tác (a): $3 * 2 = 6$;

Thao tác (b): $6 - 1 = 5$;

Input:

- Dòng đầu tiên ghi lại số tự nhiên T là số lượng Test;
- T dòng kế tiếp mỗi dòng ghi lại một bộ Test. Mỗi test là một bộ đôi S và T.

Output: Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
3	4
2 5	4
3 7	3
7 4	

BÀI 10. BIẾN ĐỔI SỐ TỰ NHIÊN

Cho số tự nhiên N ($N < 10^9$) và hai phép biến đổi (a), (b) dưới đây.

- **Thao tác (a):** Trừ N đi 1 ($N = N - 1$). Ví dụ $N = 17$, thao tác (a) biến đổi $N = N - 1 = 16$.
- **Thao tác (b):** $N = \max(u, v)$ nếu $u * v = N$ ($u > 1, v > 1$). Ví dụ $N = 16$, thao tác (b) có thể biến đổi $N = \max(2, 8) = 8$ hoặc $N = \max(4, 4) = 4$.

Chỉ được phép sử dụng hai thao tác (a) hoặc (b), hãy biến đổi N thành 1 sao cho số các thao tác (a), (b) được thực hiện ít nhất. Ví dụ với $N = 17$, số các phép (a), (b) nhỏ nhất biến đổi N thành 1 là 4 bước như sau:

Thao tác (a): $N = N - 1 = 17 - 1 = 16$

Thao tác (b): $16 = \max(4, 4) = 4$

Thao tác (b): $4 = \max(2, 2) = 2$

Thao tác (a): $2 = 2 - 1 = 1$

Input:

- Dòng đầu tiên ghi lại số tự nhiên T là số lượng Test;
- T dòng kế tiếp mỗi dòng ghi lại một bộ Test. Mỗi test là một số N.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
3	4
17	5
50	5
100	

BÀI 11. BIẾN ĐỔI SỐ NGUYÊN TỐ

Cho cặp số S và T là các số nguyên tố có 4 chữ số (Ví dụ S = 1033, T = 8197 là các số nguyên tố có 4 chữ số). Hãy viết chương trình tìm cách dịch chuyển S thành T thỏa mãn đồng thời những điều kiện dưới đây:

- Mỗi phép dịch chuyển chỉ được phép thay đổi một chữ số của số ở bước trước đó (ví dụ nếu S=1033 thì phép dịch chuyển S thành 1733 là hợp lệ);
- Số nhận được cũng là một số nguyên tố có 4 chữ số (ví dụ nếu S=1033 thì phép dịch chuyển S thành 1833 là không hợp lệ, và S dịch chuyển thành 1733 là hợp lệ);
- Số các bước dịch chuyển là ít nhất.

Ví dụ số các phép dịch chuyển ít nhất để S = 1033 thành T = 8179 là 6 bao gồm các phép dịch chuyển như sau:

8179 \leftarrow 8779 \leftarrow 3779 \leftarrow 3739 \leftarrow 3733 \leftarrow 1733 \leftarrow 1033.

Input:

- Dòng đầu tiên đưa vào số lượng test T ($T \leq 100$).
- Những dòng kế tiếp mỗi dòng đưa vào một test. Mỗi test là một bộ đôi S, T.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	6
1033 8179	5
1033 8779	

BÀI 12. KHOẢNG CÁCH XÂU KÝ TỰ

Cho tập n xâu ký tự S và hai xâu s, t \in S. Ta giả thiết các xâu ký tự S[i] \in S có độ dài bằng nhau. Hãy tìm khoảng cách đường đi ngắn nhất từ s đến t. Biết từ một xâu ký tự bất kỳ ta chỉ được phép dịch chuyển đến xâu khác với nó duy nhất 1 ký tự. Ví dụ ta có tập các từ S = { POON, TOON, PLEE, SAME, POIE, PLEA, PLIE, POIN }, s = TOON, t = PLEA ta có độ dài đường đi ngắn nhất là 7 tương ứng với các phép dịch chuyển : TOON \rightarrow POON \rightarrow POIN \rightarrow POIE \rightarrow PLIE \rightarrow PLEE \rightarrow PLEA.

Input:

- Dòng đầu tiên đưa vào số lượng test T ($T \leq 100$).
- Mỗi test được tổ chức thành 2 dòng. Dòng thứ nhất ghi lại n là số từ trong S và hai từ s, t. Dòng thứ 2 đưa vào n xâu ký tự của S; các xâu ký tự được viết cách nhau một vài khoảng trống, có độ dài không vượt quá 10 ký tự.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
1 8 TOON PLEA POON TOON PLEE SAME POIE PLEA PLIE POIN	7

BÀI 13. TÌM SỐ K THỎA MÃN ĐIỀU KIỆN

Cho hai số nguyên dương L, R . Hãy đưa ra số các số K trong khoảng $[L, R]$ thỏa mãn điều kiện:

- Tất cả các chữ số của K đều khác nhau.
- Tất cả các chữ số của K đều nhỏ hơn hoặc bằng 5.

Ví dụ với $L = 4, R = 13$ ta có 5 số thỏa mãn yêu cầu là 4, 5, 10, 12, 13,

Input:

- Dòng đầu tiên đưa vào số lượng test T .
- Dòng tiếp theo đưa vào các bộ test. Mỗi bộ test được là một cặp L, R được viết trên một dòng.
- T, L, R thỏa mãn ràng buộc: $1 \leq T \leq 100; 0 \leq L \leq R \leq 10^5$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2 4 13 100 1000	5 100

BÀI 14. SỐ NGUYÊN THỦY

Cho số nguyên N . Nhiệm vụ của bạn hãy đưa ra N số nguyên thủy đầu tiên. Số K được gọi là số nguyên thủy nếu số đó thỏa mãn điều kiện:

- Số các chữ số của K là một số chẵn.
- Tất cả các chữ số của K chỉ bao gồm số 4 hoặc 5.
- K là một số đối xứng.

Input:

- Dòng đầu tiên đưa vào số lượng test T .
- Dòng tiếp theo đưa vào các bộ test. Mỗi bộ test được là một N được viết trên một dòng.
- T, N thỏa mãn ràng buộc: $1 \leq T \leq 100; 1 \leq N \leq 10^4$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	44 55 4444 4554
4	44 55 4444 4554 5445 5555 444444 454454 544445 554455
10	

BÀI 15. DI CHUYỂN TRONG MA TRẬN *time*

Cho ma trận $A[M][N]$. Nhiệm vụ của bạn hãy tìm số bước đi ít nhất dịch chuyển từ vị trí $A[1][1]$ đến vị trí $A[M][N]$. Biết mỗi bước đi ta chỉ được phép dịch chuyển đến vị trí $A[i][j+A[i][j]]$ hoặc vị trí $A[i+A[i][j]][j]$ bên trong ma trận.

Input:

- Dòng đầu tiên đưa vào số lượng test T .
- Dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất là hai số M, N ; phần thứ hai là các phần tử của ma trận $A[i][j]$; các số được viết cách nhau một vài khoảng trống.
- $T, M, N, A[i][j]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq M, N, A[i][j] \leq 10^3$.

Output:

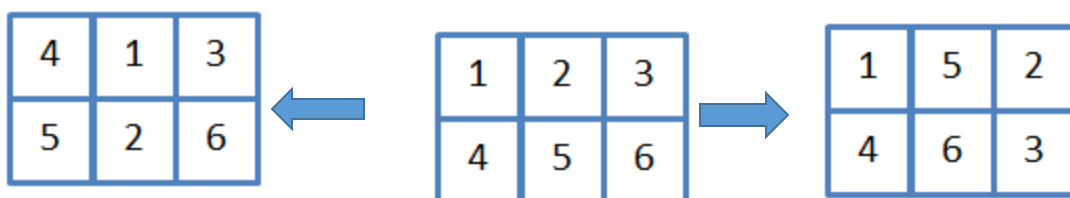
- Đưa ra kết quả mỗi test theo từng dòng. In ra -1 nếu không tìm được đáp án.

Ví dụ:

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

BÀI 16. 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

BÀI 17. DI CHUYỂN TRÁNH VẬT CẢN

Cho một bảng kích thước $N \times N$, trong đó có các ô trống '.' và vật cản 'X'. Các hàng và các cột được đánh số từ 0.

Mỗi bước di chuyển, bạn có thể đi từ ô (x, y) tới ô (u, v) nếu như 2 ô này nằm trên cùng một hàng hoặc một cột, và không có vật cản nào ở giữa.

Cho điểm xuất phát và điểm đích. Bạn hãy tính số bước di chuyển ít nhất?

Input:

- Dòng đầu tiên là số nguyên dương N ($1 \leq N \leq 100$).
- N dòng tiếp theo, mỗi dòng gồm N kí tự mô tả bảng.
- Cuối cùng là 4 số nguyên a, b, c, d với (a, b) là tọa độ điểm xuất phát, (c, d) là tọa độ đích. Dữ liệu đảm bảo hai vị trí này không phải là ô có vật cản.

Output:

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

Ví dụ:

Input	Output
3 .X. .X. ... 0 0 0 2	3

BÀI 18. GIEO MẦM

Trên một giá có kích thước $R \times C$ (R hàng, C cột), một số hạt mầm đã được tra vào các ô. Một số hạt mầm được bón thêm chất dinh dưỡng, nên đã nảy mầm sớm thành cây non.

Mỗi ngày, các cây non sẽ lan truyền chất dinh dưỡng của nó cho các mầm ở ô xung quanh (trái, trên, phải, dưới), làm cho các hạt mầm này phát triển thành cây non. Tuy nhiên, có thể có một

số hạt mầm được gieo ở vị trí lẻ loi, do không nhận được chất dinh dưỡng nên không thể nảy mầm.

Các bạn hãy xác định xem cần ít nhất bao nhiêu ngày để tất cả các hạt đều mầm?

Input:

- Dòng đầu tiên gồm 2 số nguyên R và C ($1 \leq R, C \leq 500$).
- R dòng tiếp theo, mỗi dòng gồm C số nguyên $A[i][j]$.
- $A[i][j] = 0$, ô (i, j) là ô trống.
- $A[i][j] = 1$, ô (i, j) là hạt chưa nảy mầm.
- $A[i][j] = 2$, ô (i, j) là cây non.

Output:

- In ra thời gian ngắn nhất để tất cả các hạt đều nảy mầm. Nếu có hạt nào chưa nảy mầm, in ra -1.

Ví dụ:

Test 1	Test 2
Input : 3 5 2 1 0 2 1 1 0 1 2 1 1 0 0 2 1 Output : 2	Input : 3 5 2 1 0 2 1 0 0 1 2 1 1 0 0 2 1 Output : -1

BÀI 19. DI CHUYỂN TRONG KHÔNG GIAN

Cho một hình hộp chữ nhật có kích thước $A \times B \times C$, trong đó A là chiều cao, B là chiều rộng và C là chiều dài. Mỗi ô có thể là một ô trống '.' hoặc vật cản '#'.
 Mỗi bước, bạn được phép di chuyển sang một ô kề bên cạnh (không được đi chéo). Nhiệm vụ của bạn là tìm đường đi ngắn nhất bắt đầu 'S' tới vị trí kết thúc 'E'.

Mỗi bước, bạn được phép di chuyển sang một ô kề bên cạnh (không được đi chéo). Nhiệm vụ của bạn là tìm đường đi ngắn nhất bắt đầu 'S' tới vị trí kết thúc 'E'.

Input:

- Dòng đầu tiên là số lượng bộ test T ($1 \leq T \leq 50$).
- Mỗi test bắt đầu bởi 3 số nguyên A, B, C ($A, B, C \leq 30$).
- Tiếp theo là A khối, mỗi khối gồm $B \times C$ kí tự mô tả một lát cắt của hình hộp chữ nhật. Giữa 2 khối có một dấu xuống dòng.

Output:

- In ra một số nguyên là đường đi ngắn nhất từ S tới E. Nếu không di chuyển được, in ra -1.

Ví dụ:

Input	Output
2	11
3 4 5	-1
S.....	
.###.	
.##..	
###. #	
#####	
#####	
##.##	
##...	
#####	
#####	
#.###	
#####E	
1 3 3	
S##	
#E#	
###	

BÀI 20. HEXGAME

HEXGAME là một trò chơi xếp hình gồm 10 miếng ghép hình lục giác đều, trên mỗi miếng ghép được điền một số nguyên, có 8 miếng được điền số từ 1 đến 8 và có hai miếng điền số 0. Các miếng liên kết với nhau tạo thành lưới tổ ong. Ban đầu các miếng ghép ở vị trí như hình vẽ. Tại mỗi bước, chọn một miếng ghép có đúng 6 miếng ghép kề cạnh làm tâm, rồi xoay một nấc 6 miếng ghép kề cạnh đó theo chiều kim đồng hồ. Như vậy chỉ có hai cách chọn tâm, đó là chọn tâm bên trái và chọn tâm bên phải.



Yêu cầu: Cho một trạng thái của trò chơi (nhận được sau một dãy biến đổi từ trạng thái ban đầu), hãy tính số phép biến đổi ít nhất để đưa về trạng thái ban đầu.

Input:

- Dòng đầu tiên chứa 3 số ở 3 miếng ghép dòng thứ nhất (thứ tự từ trái qua phải).
- Dòng đầu tiên chứa 4 số ở 4 miếng ghép dòng thứ hai (thứ tự từ trái qua phải).

- Dòng đầu tiên chứa 3 số ở 3 miếng ghép dòng thứ ba (thứ tự từ trái qua phải).

Output:

- In ra một số nguyên là số phép biến đổi ít nhất để đưa được về trạng thái ban đầu.

Ví dụ:

Input	Output
1 0 2 8 6 0 3 7 5 4	5