

# Đề cấu trúc dữ liệu và giải thuật

## Generation

### CTDL\_001 - THUẬT TOÁN SINH

Một xâu nhị phân độ dài  $n$  được gọi là thuận nghịch hay đối xứng nếu đảo ngược xâu nhị phân đó ta vẫn nhận được chính nó. Cho số tự nhiên  $n$  ( $n$  nhập từ bàn phím). Hãy viết chương trình liệt kê tất cả các xâu nhị phân thuận nghịch có độ dài  $n$ . Hai phần tử khác nhau của xâu thuận nghịch được ghi cách nhau một khoảng trống.

Ví dụ với  $n = 4$  ta tìm được 4 xâu nhị phân thuận nghịch như dưới đây.

```
0 0 0 0
0 1 1 0
1 0 0 1
1 1 1 1
```

**input**

4

**output**

```
0 0 0 0
0 1 1 0
1 0 0 1
1 1 1 1
```

### CTDL\_002 - Tổng dãy con = K

Cho dãy  $A[]$  gồm  $N$  số tự nhiên khác nhau và số tự nhiên  $K$ . Hãy viết chương trình liệt kê tất cả các dãy con của dãy số  $A[]$  sao cho tổng các phần tử trong dãy con đó đúng bằng  $K$ . Dữ liệu vào trên bàn phím ( $n=5, K=50$ ), 5 số dòng thứ 2 là các phần tử dãy  $A$ :

```
5 50
5 10 15 20 25
```

Các dãy con thoả mãn điều kiện tìm được liệt kê trên màn hình:

- Mỗi dòng ghi lại một dãy con. Hai phần tử khác nhau của dãy con được viết cách nhau bởi một khoảng trống.
- Dòng cuối cùng ghi lại số các dãy con có tổng các phần tử đúng bằng  $K$  tìm được.

```
10 15 25
5 20 25
5 10 15 20
3
```

### CTDL\_003 - PHƯƠNG ÁN TỐI ƯU

Cho  $a_i, c_i, W, N$  ( $i = 1, 2, \dots, N; N \leq 100$ ) là những số nguyên dương và tập hợp

$$D = \left\{ X = (x_1, x_2, \dots, x_N) : \sum_{j=1}^N a_j x_j \leq W; x_j = 0,1 \right\}.$$

Hãy viết chương trình tìm phương án tối ưu  $XOPT = (x_1, x_2, \dots, x_N)$  và giá trị tối ưu  $FOPT = F(XOPT)$  của hàm mục tiêu

$$F(x_1, x_2, \dots, x_N) = \sum_{j=1}^N c_j x_j \rightarrow \max$$

Dữ liệu vào cho bởi file data.in theo khuôn dạng sau:

- Dòng đầu tiên ghi lại số tự nhiên N và W. Hai số được viết cách nhau một vài khoảng trống;
- Dòng kế tiếp ghi lại N số  $c_j$  ( $j=1,2,\dots,N$ ). Hai số được viết cách nhau một vài khoảng trống;
- Dòng kế tiếp ghi lại N số  $a_j$  ( $j=1,2,\dots,N$ ). Hai số được viết cách nhau một vài khoảng trống;

Giá trị tối ưu FOPT và phương án XOPT tìm được sẽ liệt kê trên màn hình theo khuôn dạng:

- Dòng đầu tiên ghi lại giá trị tối ưu FOPT;
- Dòng kế tiếp ghi lại phương án tối ưu XOPT. Hai phần tử khác nhau của phương án tối ưu được viết cách nhau bởi một khoảng trống.

**input**

```
4 10
6 5 3 7
5 4 6 5
```

**output**

```
13
1 0 0 1
```

## CTDL\_004 - DÃY CON TĂNG DẦN BẬC K

Cho dãy gồm N số phân biệt  $AN = \{a_1, a_2, \dots, a_N\}$  và số tự nhiên K ( $K \leq N \leq 100$ ). Ta gọi một dãy con tăng dần bậc K của dãy số AN là một dãy các số gồm K phần tử trong dãy đó thỏa mãn tính chất tăng dần. Bài toán được đặt ra là in ra màn hình số các dãy con tăng dần bậc K của dãy số AN. Ví dụ :

**Input:**

- Dòng đầu tiên ghi lại hai số N và K tương ứng với số phần tử của dãy số và bậc của dãy con.
- Dòng kế tiếp : N số của dãy số AN, các số trong dãy không lớn hơn 100.

**Output:**

- In ra ra kết quả tìm được.

**input**

5 3  
2 5 15 10 20  
**output**

7

## DSA01001 - XÂU NHỊ PHÂN KẾ TIẾP

Cho xâu nhị phân  $X[]$ , nhiệm vụ của bạn là hãy đưa ra xâu nhị phân tiếp theo của  $X[]$ . Ví dụ  $X[] = "010101"$  thì xâu nhị phân tiếp theo của  $X[]$  là  $"010110"$ .

Input:

- Dòng đầu tiên đưa vào số lượng test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một xâu nhị phân  $X$ .
- $T, X[]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{length}(X) \leq 10^3$ .

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

2  
010101  
111111

**output**

010110  
000000

## DSA01002 - TẬP CON KẾ TIẾP

Cho hai số  $N, K$  và một tập con  $K$  phần tử  $X[] = (X_1, X_2, \dots, X_K)$  của  $1, 2, \dots, N$ . Nhiệm vụ của bạn là hãy đưa ra tập con  $K$  phần tử tiếp theo của  $X[]$ . Ví dụ  $N=5, K=3, X[] = \{2, 3, 4\}$  thì tập con tiếp theo của  $X[]$  là  $\{2, 3, 5\}$ .

Input:

- Dòng đầu tiên đưa vào số lượng test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một xâu nhị phân  $X$ .
- $T, X[]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{length}(X) \leq 10^3$ .

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

2  
5 3  
1 4 5  
5 3

3 4 5  
**output**

2 3 4  
1 2 3

## DSA01003 - HOÁN VỊ KẾ TIẾP

Cho số tự nhiên  $N$  và một hoán vị  $X[]$  của  $1, 2, \dots, N$ . Nhiệm vụ của bạn là đưa ra hoán vị tiếp theo của  $X[]$ . Ví dụ  $N=5$ ,  $X[] = \{1, 2, 3, 4, 5\}$  thì hoán vị tiếp theo của  $X[]$  là  $\{1, 2, 3, 5, 4\}$ .

Input:

- Dòng đầu tiên đưa vào số lượng test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất là số  $N$ ; dòng tiếp theo đưa vào hoán vị  $X[]$  của  $1, 2, \dots, N$ .
- $T, N, X[]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ .

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

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

**output**

1 2 3 5 4  
1 2 3 4 5

## DSA01004 - SINH TỔ HỢP

Cho hai số nguyên dương  $N$  và  $K$ . Nhiệm vụ của bạn là hãy liệt kê tất cả các tập con  $K$  phần tử của  $1, 2, \dots, N$ . Ví dụ với  $N=5$ ,  $K=3$  ta có 10 tập con của  $1, 2, 3, 4, 5$  như sau:  $\{1, 2, 3\}, \{1, 2, 4\}, \{1, 2, 5\}, \{1, 3, 4\}, \{1, 3, 5\}, \{1, 4, 5\}, \{2, 3, 4\}, \{2, 3, 5\}, \{2, 4, 5\}, \{3, 4, 5\}$ .

Input:

- Dòng đầu tiên đưa vào số lượng test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một cặp số tự nhiên  $N, K$  được viết trên một dòng.
- $T, n$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq k \leq n \leq 15$ .

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

2  
4 3  
5 3

#### output

123 124 134 234  
123 124 125 134 135 145 234 235 245 345

## DSA01005 - SINH HOÁN VỊ

Cho số nguyên dương  $N$ . Nhiệm vụ của bạn là hãy liệt kê tất cả các hoán vị của  $1, 2, \dots, N$ . Ví dụ với  $N = 3$  ta có kết quả: 123, 132, 213, 231, 312, 321.

Input:

- Dòng đầu tiên đưa vào số lượng test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một số tự nhiên  $N$  được viết trên một dòng.
- $T, n$  thỏa mãn ràng buộc:  $1 \leq T, N \leq 10$ .

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

2  
2  
3

#### output

12 21  
123 132 213 231 312 321

## DSA01006 - HOÁN VỊ NGƯỢC

Cho số nguyên dương  $N$ . Nhiệm vụ của bạn là hãy liệt kê tất cả các hoán vị của  $1, 2, \dots, N$  theo thứ tự ngược. Ví dụ với  $N = 3$  ta có kết quả: 321, 312, 231, 213, 132, 123.

Input:

- Dòng đầu tiên đưa vào số lượng test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một số tự nhiên  $N$  được viết trên một dòng.
- $T, n$  thỏa mãn ràng buộc:  $1 \leq T, N \leq 10$ .

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

2

2  
3

**output**

21 12  
321 312 231 213 132 123

## DSA01007 - XÂU AB CÓ ĐỘ DÀI N

Xâu ký tự str được gọi là xâu AB nếu mỗi ký tự trong xâu hoặc là ký tự 'A' hoặc là ký tự 'B'. Ví dụ xâu str="ABBABB" là xâu AB độ dài 6. Nhiệm vụ của bạn là hãy liệt kê tất cả các xâu AB có độ dài n.

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một số tự nhiên n.
- T, n thỏa mãn ràng buộc:  $1 \leq T \leq 10$ ;  $1 \leq n \leq 10$ .

Output:

- Đưa ra kết quả mỗi test theo từng dòng. Mỗi xâu cách nhau 1 khoảng trống.

**input**

2  
2  
3

**output**

AA AB BA BB  
AAA AAB ABA ABB BAA BAB BBA BBB

## DSA01008 - XÂU NHỊ PHÂN CÓ K BIT 1

Hãy in ra tất cả các xâu nhị phân độ dài N, có K bit 1 theo thứ tự từ điển tăng dần.

Input: Dòng đầu tiên là số lượng bộ test T ( $T \leq 20$ ). Mỗi test gồm 2 số nguyên N, K ( $1 \leq K \leq N \leq 16$ ).

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

**input**

2  
4 2  
3 2

**output**

0011  
0101  
0110  
1001  
1010  
1100

011  
101  
110

## DSA01009 - XÂU AB ĐẶC BIỆT

Một xâu kí tự  $S = (s_1, s_2, \dots, s_n)$  được gọi là xâu AB độ dài  $n$  nếu với mọi  $s_i \in S$  thì  $s_i$  hoặc là kí tự A hoặc  $s_i$  là kí tự B. Ví dụ xâu  $S = "ABABABAB"$  là một xâu AB độ dài 8. Cho số tự nhiên  $N$  và số tự nhiên  $K$  ( $1 \leq K < N \leq 15$  được nhập từ bàn phím), hãy viết chương trình liệt kê tất cả các xâu AB có độ dài  $N$  chứa duy nhất một dãy  $K$  kí tự A liên tiếp.

**Input:** chỉ có một dòng ghi hai số  $N$  và  $K$ .

**Output:**

- Dòng đầu tiên ghi lại số các xâu AB thỏa mãn yêu cầu bài toán;
- Những dòng kế tiếp, mỗi dòng ghi lại một xâu AB thỏa mãn. Các xâu được ghi ra theo thứ tự từ điển.

**input**

5 3

**output**

5

AAABA

AAABB

ABAAA

BAAAB

BBAAA

## DSA01011 - HOÁN VỊ TIẾP THEO CỦA CHUỖI SỐ

Hãy viết chương trình nhận vào một chuỗi (có thể khá dài) các ký tự số và đưa ra màn hình hoán vị kế tiếp của các ký tự số đó (với ý nghĩa là hoán vị có giá trị lớn hơn tiếp theo nếu ta coi chuỗi đó là một giá trị số nguyên). Chú ý: Các ký tự số trong dãy có thể trùng nhau.

Ví dụ:

123 -> 132

279134399742 -> 279134423799

Cũng có trường hợp sẽ không thể có hoán vị kế tiếp. Ví dụ như khi đầu vào là chuỗi 987.

**Input:** Dòng đầu tiên ghi số nguyên  $t$  là số bộ test ( $1 \leq t \leq 1000$ ). Mỗi bộ test có một dòng, đầu tiên là số thứ tự bộ test, một dấu cách, sau đó là chuỗi các ký tự số, tối đa 80 phần tử.

**Output:** Với mỗi bộ test hãy đưa ra một dòng gồm thứ tự bộ test, một dấu cách, tiếp theo đó là hoán vị kế tiếp hoặc chuỗi "BIGGEST" nếu không có hoán vị kế tiếp.

**input**

3

```
1 123
2 279134399742
3 987
```

**output**

```
1 132
2 279134423799
3 BIGGEST
```

## DSA01012 - MÃ GRAY 1

Số nhị phân được xem là cách mặc định biểu diễn các số. Tuy nhiên, trong nhiều ứng dụng của điện tử và truyền thông lại dùng một biến thể của mã nhị phân đó là mã Gray. Mã Gray độ dài  $n$  có mã đầu tiên là  $n$  số 0, mã kế tiếp của nó là một chuỗi nhị phân độ dài  $n$  khác biệt với chuỗi trước đó một bit. Ví dụ với  $n=3$  ta có  $2^3$  mã Gray như sau: 000, 001, 011, 010, 110, 111, 101, 100. Hãy viết chương trình liệt kê các mã Gray có độ dài  $n$ .

**Input:**

- Dòng đầu tiên là số lượng test  $T$ .
- $T$  dòng kế tiếp ghi lại mỗi dòng một test. Mỗi test là một số tự nhiên  $n$ .
- $T, n$  thỏa mãn ràng buộc:  $1 \leq T, n \leq 10$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
3
4
```

**output**

```
000 001 011 010 110 111 101 100
0000 0001 0011 0010 0110 0111 0101 0100 1100 1101 1111 1110 1010 1011 1001 1000
```

## DSA01013 - MÃ GRAY 2

Số nhị phân được xem là cách mặc định biểu diễn các số. Tuy nhiên, trong nhiều ứng dụng của điện tử và truyền thông lại dùng một biến thể của mã nhị phân đó là mã Gray. Mã Gray độ dài  $n$  có mã đầu tiên là  $n$  số 0, mã kế tiếp của nó là một chuỗi nhị phân độ dài  $n$  khác biệt với chuỗi trước đó một bit. Ví dụ với  $n=3$  ta có  $2^3$  mã Gray như sau: 000, 001, 011, 010, 110, 111, 101, 100. Hãy viết chương trình chuyển đổi một chuỗi mã Gray  $X$  có độ dài  $n$  thành một chuỗi mã nhị phân.

**Input:**

- Dòng đầu tiên là số lượng test  $T$ .
- $T$  dòng kế tiếp ghi lại mỗi dòng một test. Mỗi test là một số tự nhiên  $n$ .
- $T, n$  thỏa mãn ràng buộc:  $1 \leq T, n \leq 10$ .

**Output:**



- Đưa ra kết quả mỗi test theo từng dòng.

**input**

2  
01101  
01011

**output**

01001  
01101

## DSA01014 - TẬP HỢP

Xét tất cả các tập hợp các số nguyên dương có các phần tử khác nhau và không lớn hơn số  $n$  cho trước. Nhiệm vụ của bạn là hãy đếm xem có tất cả bao nhiêu tập hợp có số lượng phần tử bằng  $k$  và tổng của tất cả các phần tử trong tập hợp bằng  $s$ ?

Các tập hợp là hoán vị của nhau chỉ được tính là một.

Ví dụ với  $n = 9, k = 3, s = 23$ ,  $\{6, 8, 9\}$  là tập hợp duy nhất thỏa mãn.

**Input:**

- Gồm nhiều bộ test (không quá 100 test).
- Mỗi bộ test gồm 3 số nguyên  $n, k, s$  với  $1 \leq n \leq 20, 1 \leq k \leq 10$  và  $1 \leq s \leq 155$ . Input kết thúc bởi 3 số 0.

**Output:**

- Với mỗi test in ra số lượng các tập hợp thỏa mãn điều kiện đề bài.

**input**

9 3 23  
9 3 22  
10 3 28  
16 10 107  
20 8 102  
20 10 105  
20 10 155  
3 4 3  
4 2 11  
0 0 0

**output**

1  
2  
0  
20  
1542  
5448  
1  
0  
0

## DSA01015 - TÌM BỘI SỐ

Cho số nguyên  $N$ . Nhiệm vụ của bạn cần tìm số nguyên  $X$  nhỏ nhất là bội của  $N$ , và  $X$  chỉ chứa hai chữ số 0 và 9.

**Input:** Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 10000$ ). Mỗi bộ test chứa số nguyên  $N$  trên một dòng ( $1 \leq N \leq 500$ ).

**Output:** Với mỗi test in ra đáp án tìm được trên một dòng.

**input**

```
3
2
5
11
```

**output**

```
90
90
99
```

## DSA01016 - PHÂN TÍCH SỐ 1

Cho số nguyên dương  $N$ . Nhiệm vụ của bạn là hãy liệt kê tất cả các cách phân tích số tự nhiên  $N$  thành tổng các số tự nhiên nhỏ hơn hoặc bằng  $N$ . Phép hoán vị vữa một cách được xem là giống nhau. Ví dụ với  $N = 5$  ta có kết quả là: (5), (4, 1), (3, 2), (3, 1, 1), (2, 2, 1), (2, 1, 1, 1), (1, 1, 1, 1, 1).

**Input:**

- Dòng đầu tiên đưa vào số lượng test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một số tự nhiên  $N$  được viết trên một dòng.
- $T, n$  thỏa mãn ràng buộc:  $1 \leq T, N \leq 10$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
4
5
```

**output**

```
(4) (3 1) (2 2) (2 1 1) (1 1 1 1)
(5) (4 1) (3 2) (3 1 1) (2 2 1) (2 1 1 1) (1 1 1 1 1)
```

## DSA01017 - MÃ GRAY 3

Số nhị phân được xem là cách mặc định biểu diễn các số. Tuy nhiên, trong nhiều ứng dụng của điện tử và truyền thông lại dùng một biến thể của mã nhị phân đó là mã Gray. Mã Gray độ dài  $n$

có mã đầu tiên là  $n$  số 0, mã kế tiếp của nó là một xâu nhị phân độ dài  $n$  khác biệt với xâu trước đó một bit. Ví dụ với  $n=3$  ta có  $2^3$  mã Gray như sau: 000, 001, 011, 010, 110, 111, 101, 100. Hãy viết chương trình chuyển đổi một xâu mã nhị phân  $X$  có độ dài  $n$  thành một xâu mã Gray.

**Input:**

- Dòng đầu tiên là số lượng test  $T$ .
- $T$  dòng kế tiếp ghi lại mỗi dòng một test. Mỗi test là một số tự nhiên  $n$ .
- $T, n$  thỏa mãn ràng buộc:  $1 \leq T, n \leq 10$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
01001
01101
```

**output**

```
01101
01011
```

## DSA01018 - TẬP CON LIÊN KÈ PHÍA TRƯỚC

Cho hai số  $N, K$  và một tập con  $K$  phần tử  $X[] = (X_1, X_2, \dots, X_K)$  của  $1, 2, \dots, N$ . Nhiệm vụ của bạn là hãy đưa ra tập con  $K$  phần tử trước đó của  $X[]$ . Ví dụ  $N=5, K=3, X[] = \{2, 3, 5\}$  thì tập con trước đó của  $X[]$  là  $\{2, 3, 4\}$ . Chú ý nếu tập con trong input là đầu tiên thì trước đó là tập con cuối cùng.

**Input:**

- Dòng đầu tiên đưa vào số lượng test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất là hai số  $N$  và  $K$ ; dòng tiếp theo đưa vào  $K$  phần tử của  $X[]$  là một tập con  $K$  phần tử của  $1, 2, \dots, N$ .
- $T, K, N, X[]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100; 1 \leq K \leq N \leq 10^3$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
5 3
2 3 5
5 3
1 2 3
```

**output**

```
2 3 4
3 4 5
```

## DSA01019 - HAHAHA

Khi chat chit vui vẻ, anh em chiến hữu thường hay gõ HAHA để thể hiện sự sảng khoái. Đôi khi cũng có thể gõ HAHAHAHA chẳng hạn cho thêm phần nhấn mạnh.

Giả sử một xâu ký tự được coi là HAHA nếu thỏa mãn các điều kiện:

- Ký tự đầu tiên phải là chữ H, ký tự cuối cùng phải là chữ A
- Không có hai chữ H nào liền nhau

Cho trước độ dài N, hãy liệt kê tất cả các xâu ký tự HAHA theo thứ tự từ điển.

### Input

- Dòng đầu ghi số bộ test T (không quá 10)
- Mỗi bộ test ghi độ dài N ( $2 \leq N < 16$ )

### Output

- Với mỗi bộ test, ghi ra tất cả các xâu HAHA tìm được theo thứ tự từ điển, mỗi xâu viết trên một dòng.

### Ví dụ

#### input

```
2
2
4
```

#### output

```
HA
HAAA
HAHA
```

## DSA01020 - XÂU NHỊ PHÂN TRƯỚC

Cho xâu nhị phân  $X[]$ , nhiệm vụ của bạn là hãy đưa ra xâu nhị phân trước của  $X[]$ . Ví dụ  $X[] = "111111"$  thì xâu nhị phân trước của  $X[]$  là  $"111110"$ . Với xâu  $X[] = "000001"$  thì xâu nhị phân trước của  $X[]$  là  $"000000"$ . Chú ý: nếu xâu dữ liệu trong input là xâu đầu tiên thì trước nó sẽ là xâu cuối cùng.

### Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test là một xâu nhị phân X.
- T,  $X[]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{length}(X) \leq 10^3$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

### Ví dụ:

**input**

```
2
010101
111111
output
```

```
010100
111110
```

**DSA01021 - TỔ HỢP TIẾP THEO - DSA01010 - TẬP QUÂN SỰ**

Cho số nguyên dương ( $1 < N < 40$ ) và số nguyên dương  $K < N$ . Với 1 tổ hợp chập K phần tử của N, hãy cho biết tổ hợp tiếp theo sẽ có bao nhiêu phần tử mới. Nếu tổ hợp đã cho là cuối cùng thì kết quả là K.

**Dữ liệu vào:** Dòng đầu ghi số bộ test, không quá 20. Mỗi bộ test viết trên hai dòng

- Dòng 1: hai số nguyên dương N và K ( $K < N$ )
- Dòng 2 ghi K số của tổ hợp ban đầu. Theo đúng thứ tự tăng dần, không có số nào trùng nhau.

**Kết quả:** Với mỗi bộ dữ liệu in ra số lượng phần tử mới.

**Ví dụ:****input**

```
3
5 3
1 3 5
5 3
1 4 5
6 4
3 4 5 6
output
```

```
1
2
4
```

**DSA01022 - SỐ THỨ TỰ HOÁN VỊ**

Cho một hoán vị của N số nguyên dương đầu tiên.

Hãy xác định xem đó là hoán vị thứ bao nhiêu nếu liệt kê theo thứ tự tăng dần (tính từ 1).

**Input**

Dòng đầu ghi số T là số bộ test ( $T < 10$ )

Mỗi bộ test gồm 2 dòng

- Dòng đầu ghi số nguyên dương N ( $3 < N < 10$ )

- Dòng tiếp theo ghi một hoán vị của các số nguyên dương từ 1 đến N.

### Output

Với mỗi bộ test, ghi ra trên một dòng số thứ tự của hoán vị (tính từ 1, theo thứ tự liệt kê tăng dần).

### Ví dụ

#### input

```
2
4
1 4 3 2
4
1 2 3 4
```

#### output

```
6
1
```

## DSA01023 - SỐ THỨ TỰ TỔ HỢP

Cho một tổ hợp chập K của N số nguyên dương đầu tiên ( $2 < K < N < 15$ ).

Hãy xác định xem đó là tổ hợp thứ bao nhiêu nếu liệt kê tất cả các tổ hợp theo thứ tự tăng dần (tính từ 1).

### Input

Dòng đầu ghi số T là số bộ test ( $T < 10$ )

Mỗi bộ test gồm 2 dòng

- Dòng đầu ghi 2 số nguyên dương N và K ( $2 < K < N < 15$ )
- Dòng tiếp theo ghi một tổ hợp chập K của các số nguyên dương từ 1 đến N.

### Output

Với mỗi bộ test, ghi ra trên một dòng số thứ tự của tổ hợp (tính từ 1, theo thứ tự liệt kê tăng dần).

### Ví dụ

#### input

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

#### output

```
9
12
```

## DSA01025 - ĐẶT TÊN - 2

Vương quốc PTIT sử dụng bảng chữ cái gồm N chữ cái Latinh viết hoa. Quy tắc đặt tên của gia đình Hoàng gia PTIT là chọn ra K chữ cái (không trùng nhau) và sắp xếp lại theo thứ tự từ điển.

Hãy liệt kê tất cả các cái tên có thể có của gia đình Hoàng gia PTIT

### Input

- Dòng đầu ghi số bộ test T (không quá 10)
- Mỗi bộ test ghi 2 số N và K ( $3 < K < N < 16$ )

### Output

- Với mỗi bộ test, ghi ra tất cả các cái tên có thể được tạo ra, mỗi kết quả viết trên một dòng.

### Ví dụ

#### input

1  
4 2

#### output

AB  
AC  
AD  
BC  
BD  
CD

## DSA01026 - PHÁT LỘC

Một xâu ký tự số chỉ bao gồm các chữ số 6 và 8 sẽ được gọi là "phát lộc" nếu thỏa mãn các điều kiện sau:

- Độ dài xâu ít nhất là 6
- Chữ số đầu tiên là chữ số 8, chữ số cuối cùng là chữ số 6
- Không có 2 chữ số 8 nào ở cạnh nhau
- Không có nhiều hơn 3 chữ số 6 ở cạnh nhau.

Viết chương trình liệt kê các xâu ký tự phát lộc độ dài N theo thứ tự tăng dần.

### Input

Chỉ có 1 dòng ghi số N ( $5 < N < 16$ ).

### Output

Ghi ra các xâu ký tự phát lộc độ dài N, mỗi xâu trên một dòng.

### Ví dụ

### input

6

### output

866686

866866

868666

868686

## DSA01027 - HOÁN VỊ DÃY SỐ

Cho dãy số  $A[]$  có  $N$  phần tử là các số nguyên dương khác nhau từng đôi một. Hãy liệt kê tất cả các hoán vị của dãy số  $A[]$  theo thứ tự tăng dần, tức là hoán vị đầu tiên có giá trị tăng dần từ trái qua phải, hoán vị cuối cùng giảm dần từ trái qua phải.

### Input

Dòng đầu ghi số  $N$  ( $1 < N < 9$ )

Dòng thứ 2 ghi  $N$  số của dãy  $A[]$  ( $0 < A[i] < 10000$ )

### Output

Ghi mỗi hoán vị của dãy số trên một dòng

### Ví dụ

#### input

3

88 77 99

#### output

77 88 99

77 99 88

88 77 99

88 99 77

99 77 88

99 88 77

## DSA01028 - LIỆT KÊ TỔ HỢP

Cho dãy số  $A[]$  có  $N$  phần tử. Hãy liệt kê tất cả các tổ hợp chập  $K$  của tập các phần tử khác nhau trong  $A[]$ . Các tổ hợp cần liệt kê theo thứ tự từ điển (tức là trong mỗi tổ hợp thì giá trị từ nhỏ đến lớn, và tổ hợp sau lớn hơn tổ hợp trước).

### Input

Dòng đầu ghi hai số  $N$  và  $K$ .

Dòng thứ 2 ghi  $N$  số của mảng  $A[]$ . Các giá trị không quá 1000.

Dữ liệu đảm bảo số phần tử khác nhau của  $A[]$  không quá 20 và  $K$  không quá 10.



## Output

Ghi ra lần lượt các tổ hợp tìm được, mỗi tổ hợp trên một dòng.

## Ví dụ

### input

```
8 3
2 4 4 3 5 1 3 4
```

### output

```
1 2 3
1 2 4
1 2 5
1 3 4
1 3 5
1 4 5
2 3 4
2 3 5
2 4 5
3 4 5
```

## DSA02008 - CHỌN SỐ TỪ MA TRẬN VUÔNG CẤP N

Cho ma trận vuông  $C_{i,j}$  cấp N ( $1 \leq i, j \leq N < 10$ ) gồm  $N^2$  số tự nhiên và số tự nhiên K (các số trong ma trận không nhất thiết phải khác nhau và đều không quá 100, K không quá  $10^4$ ). Hãy viết chương trình lấy mỗi hàng, mỗi cột duy nhất một phần tử sao cho tổng các phần tử này đúng bằng K.

**Input:** Dòng 1 ghi hai số N và K. N dòng tiếp theo ghi ma trận C.

**Output:** Dòng đầu ghi số cách tìm được. Mỗi dòng tiếp theo ghi một cách theo vị trí của số đó trong lần lượt từng hàng của ma trận. Xem ví dụ để hiểu rõ hơn.

### input

```
3 10
2 4 3
1 3 6
4 2 4
```

### output

```
2
1 3 2
3 2 1
```

## Backtracking

## DSA01024 - ĐẶT TÊN - 1

Kỳ thi ICPC có K đội của PTIT tham gia và đội tuyển đang rất đầu đầu không biết chọn các cái tên như thế nào cho các đội. Yêu cầu phải đảm bảo tên không có khoảng trống và không được trùng nhau. Sau khi thảo luận, có N cái tên được đề xuất (có thể bị trùng nhau). Với  $K < 15$  và  $4 < N < 30$ .

Hãy liệt kê tất cả danh sách các tổ hợp K cái tên khác nhau có thể được tạo ra theo thứ tự từ điển.

### Input

Dòng đầu ghi 2 số N và K.

Tiếp theo là 1 dòng ghi N cái tên, mỗi cái tên có độ dài không quá 15 và cách nhau một khoảng trống. Tất cả đều là ký tự in hoa.

### Output

Ghi ra tất cả các tổ hợp tên có thể được lựa chọn theo thứ tự từ điển.

Tức là các tên trong mỗi tổ hợp liệt kê theo thứ tự từ điển và các tổ hợp cũng được liệt kê theo thứ tự từ điển.

### Ví dụ

#### input

```
6 2
DONG TAY NAM BAC TAY BAC
```

#### output

```
BAC DONG
BAC NAM
BAC TAY
DONG NAM
DONG TAY
NAM TAY
```

## DSA02001 - DÃY SỐ 1

Cho dãy số  $A[]$  gồm  $n$  số nguyên dương. Tam giác đặc biệt của dãy số  $A[]$  là tam giác được tạo ra bởi  $n$  hàng, trong đó hàng thứ 1 là dãy số  $A[]$ , hàng  $i$  là tổng hai phần tử liên tiếp của hàng  $i-1$  ( $2 \leq i \leq n$ ). Ví dụ  $A[] = \{1, 2, 3, 4, 5\}$ , khi đó tam giác được tạo nên như dưới đây:

[1, 2, 3, 4, 5]

[3, 5, 7, 9]

[8, 12, 16]

[20, 28]

[48]

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào  $N$  là số lượng phần tử của dãy số  $A[]$ ; dòng tiếp theo đưa vào  $N$  số của mảng  $A[]$ .
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N, A[i] \leq 10$ ;

**Output:**

- Đưa ra tam giác tổng của mỗi test theo từng dòng. Mỗi dòng của tam giác tổng được bao bởi ký tự [, ].

**input**

```
1
5
1 2 3 4 5
```

**output**

```
[1 2 3 4 5]
[3 5 7 9]
[8 12 16]
[20 28]
[48]
```

**DSA02002 - DÃY SỐ 2**

Cho dãy số  $A[]$  gồm  $n$  số nguyên dương. Tam giác đặc biệt của dãy số  $A[]$  là tam giác được tạo ra bởi  $n$  hàng, trong đó hàng thứ  $n$  là dãy số  $A[]$ , hàng  $i$  là tổng hai phần tử liên tiếp của hàng  $i+1$  ( $1 \leq i \leq n-1$ ). Ví dụ  $A[] = \{1, 2, 3, 4, 5\}$ , khi đó tam giác được tạo nên như dưới đây:

```
[48]

[20, 28]

[8, 12, 16]

[3, 5, 7, 9 ]

[1, 2, 3, 4, 5 ]
```

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào  $N$  là số lượng phần tử của dãy số  $A[]$ ; dòng tiếp theo đưa vào  $N$  số của mảng  $A[]$ .
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N, A[i] \leq 10$ ;

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng. Mỗi dòng của tam giác tổng được bao bởi ký tự [, ].

**input**

```
1
5
1 2 3 4 5
```

**output**

```
[48] [20 28] [8 12 16] [3 5 7 9] [1 2 3 4 5]
```

## DSA02003 - DI CHUYỂN TRONG MÊ CUNG 1

Cho một mê cung bao gồm các khối được biểu diễn như một ma trận nhị phân  $A[N][N]$ . Một con chuột đi từ ô đầu tiên góc trái ( $A[0][0]$ ) đến ô cuối cùng góc phải ( $A[N-1][N-1]$ ) theo nguyên tắc:

- Down (D): Chuột được phép xuống dưới nếu ô dưới nó có giá trị 1.
- Right (R): Chuột được phép sang phải dưới nếu ô bên phải nó có giá trị 1.

Hãy đưa ra một hành trình của con chuột trên mê cung. Đưa ra -1 nếu chuột không thể đi đến đích.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất đưa vào số  $N$  là kích cỡ của mê cung; dòng tiếp theo đưa vào ma trận nhị phân  $A[N][N]$ .
- $T, N, A[i][j]$  thỏa mãn ràng buộc:  $1 \leq T \leq 10$ ;  $2 \leq N \leq 10$ ;  $0 \leq A[i][j] \leq 1$ .

### Output:

- Đưa ra tất cả đường đi của con chuột trong mê cung theo thứ tự từ điển. Đưa ra -1 nếu chuột không đi được đến đích.

### input

```
2
4
1 0 0 0
1 1 0 1
0 1 0 0
1 1 1 1
5
1 0 0 0 0
1 1 1 1 1
1 1 0 0 1
0 1 1 1 1
0 0 0 1 1
```

### output

```
DRDDRR
DDRDRRDR DDRRRRD DRDDRRDR DRDDRRRD DRRRRDDD
```

## DSA02004 - DI CHUYỂN TRONG MÊ CUNG 2

Cho một mê cung bao gồm các khối được biểu diễn như một ma trận nhị phân  $A[N][N]$ . Một con chuột đi từ ô đầu tiên góc trái ( $A[0][0]$ ) đến ô cuối cùng góc phải ( $A[N-1][N-1]$ ) theo nguyên tắc:

- Down (D): Chuột được phép xuống dưới nếu ô dưới nó có giá trị 1.
- Right (R): Chuột được phép sang phải dưới nếu ô bên phải nó có giá trị 1.
- Left (L): Chuột được phép sang trái dưới nếu ô bên trái nó có giá trị 1.
- Up (U): Chuột được phép lên trên nếu ô trên nó có giá trị 1.

Hãy đưa ra tất cả các hành trình của con chuột trên mê cung. Đưa ra -1 nếu chuột không thể đi đến đích.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất đưa vào số N là kích cỡ của mê cung; dòng tiếp theo đưa vào ma trận nhị phân  $A[N][N]$ .
- T, N,  $A[i][j]$  thỏa mãn ràng buộc:  $1 \leq T \leq 10$ ;  $2 \leq N \leq 8$ ;  $0 \leq A[i][j] \leq 1$ .

**Output:**

- Đưa ra các chuỗi ký tự được sắp xếp, trong đó mỗi chuỗi là một đường đi của con chuột trong mê cung. In ra đáp án theo thứ tự từ điển. Đưa ra -1 nếu chuột không đi được đến đích.

**input**

```
3
4
1 0 0 0
1 1 0 1
0 1 0 0
0 1 1 1
4
1 0 0 0
1 1 0 1
1 1 0 0
0 1 1 1
5
1 0 0 0 0
1 1 1 1 1
1 1 1 0 1
0 0 0 0 1
0 0 0 0 1
```

**output**

```
DRDDRR
DDRDRR DRDDRR
DDRRURRDDD DDRURRRDDD DRDRURRDDD DRRRRDDD
```

## DSA02005 - HOÁN VỊ XÂU KÝ TỰ

Cho chuỗi ký tự S bao gồm các ký tự in hoa khác nhau. Hãy đưa ra tất cả các hoán vị của chuỗi ký tự S. Ví dụ S="ABC" ta có kết quả {ABC ACB BAC BCA CAB CBA}.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test là một chuỗi ký tự S được viết trên 1 dòng.
- T, S thỏa mãn ràng buộc:  $1 \leq T \leq 10$ ;  $1 \leq \text{length}(S) \leq 10$ ;

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

2  
AB  
ABC

#### output

AB BA  
ABC ACB BAC BCA CAB CBA

## DSA02006 - DÃY CON TỔNG BẰNG K

Cho dãy số  $A[] = (a_1, a_2, \dots, a_n)$  và số tự nhiên  $K$ . Hãy đưa ra tất cả các dãy con của dãy số  $A[]$  sao cho tổng các phần tử của dãy con đó đúng bằng  $K$ . Các phần tử của dãy số  $A[]$  được giả thuyết là nguyên dương và không có các phần tử giống nhau. Ví dụ với dãy con  $A[] = \{5, 10, 15, 20, 25\}$ ,  $K = 50$  ta có 3 dãy con  $\{5, 10, 15, 20\}$ ,  $\{5, 20, 25\}$ ,  $\{10, 15, 25\}$ .

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất đưa vào số  $N$  là số lượng phần tử của dãy số  $A[]$  và số  $K$ ; dòng tiếp theo đưa vào  $N$  phần tử của dãy số  $A[]$ .
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10$ ;  $1 \leq K, A[i] \leq 100$ .

#### Output:

- Đưa ra tất cả các dãy con của dãy số  $A[]$  thỏa mãn yêu cầu bài toán theo thứ tự từ điển, trong đó mỗi dãy con được bao bởi các ký tự  $[, ]$ . Nếu không có dãy con nào thỏa mãn yêu cầu bài toán, hãy đưa ra -1.

#### input

2  
5 50  
5 10 15 20 25  
8 53  
15 22 14 26 32 9 16 8

#### output

[5 10 15 20] [5 20 25] [10 15 25]  
[8 9 14 22] [8 14 15 16] [15 16 22]

## DSA02007 - ĐỔI CHỖ CÁC CHỮ SỐ

Cho số tự nhiên  $K$  và xâu ký tự các chữ số  $S$ . Nhiệm vụ của bạn là đưa ra số lớn nhất bằng cách thực hiện nhiều nhất  $K$  lần đổi chỗ các ký tự trong  $S$ . Ví dụ  $K = 3$  và  $S = "1234567"$  ta được  $"7654321"$ .

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất là số K; dòng tiếp theo là xâu ký tự S.
- T, K, S thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq K \leq 10$ ;  $1 \leq \text{length}(S) \leq 7$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
3
4
1234567
3
3435335
2
1034
```

**output**

```
3
4
1234567
3
3435335
2
1034
```

## DSA02009 - CHIA MẢNG

Cho mảng các số nguyên A[] gồm N phần tử. Hãy chia mảng số nguyên A[] thành K tập con khác rỗng sao cho tổng các phần tử của mỗi tập con đều bằng nhau. Mỗi phần tử thuộc tập con xuất hiện duy nhất một lần trong tất cả các tập con. Ví dụ với A[] = {2, 1, 4, 5, 6}, K = 3 ta có kết quả {2, 4}, {1, 5}, {6}.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất là hai số N và K; dòng tiếp theo đưa vào N số của mảng A[]; các số được viết cách nhau một vài khoảng trống.
- T, N, K, A[i] thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N$ ,  $K \leq 20$ ,  $0 \leq A[i] \leq 100$ .

**Output:**

- Đưa ra 1 nếu có thể chia tập con thành K tập thỏa mãn yêu cầu bài toán, ngược lại đưa ra 0.

**input**

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

2 1 5 5 6

**output**

1

0

## DSA02010 - TỔ HỢP SỐ CÓ TỔNG BẰNG X

Cho mảng  $A[]$  gồm  $N$  số nguyên dương phân biệt và số  $X$ . Nhiệm vụ của bạn là tìm phép tổ hợp các số trong mảng  $A[]$  có tổng bằng  $X$ . Các số trong mảng  $A[]$  có thể được sử dụng nhiều lần. Mỗi tổ hợp các số của mảng  $A[]$  được in ra theo thứ tự không giảm các số. Ví dụ với  $A[] = \{2, 4, 6, 8\}$ ,  $X = 8$  ta có các tổ hợp các số như sau:

[2, 2, 2, 2], [2, 2, 4], [2, 6], [4, 4], [8].

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất là hai số  $N$  và  $X$ ; dòng tiếp theo đưa vào  $N$  số của mảng  $A[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, X, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 10$ ;  $1 \leq X, A[i] \leq 100$ .  $N \leq 20$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng. Mỗi đường tổ hợp được bao bởi cặp ký tự [, ].  
Đưa ra -1 nếu không có tổ hợp nào thỏa mãn yêu cầu bài toán.

**input**

1

4 8

2 4 6 8

**output**

[2 2 2 2][2 2 4][2 6][4 4][8]

## DSA02011 - MÁY ATM

Một máy ATM hiện có  $n$  ( $n \leq 30$ ) tờ tiền có giá trị  $t[1], t[2], \dots, t[n]$ . Hãy tìm cách trả ít tờ nhất với số tiền đúng bằng  $S$  (các tờ tiền có giá trị bất kỳ và có thể bằng nhau, mỗi tờ tiền chỉ được dùng một lần).

**Input:**

Dòng đầu tiên ghi số bộ test  $T$  ( $T < 10$ ). Mỗi bộ test gồm 2 số nguyên  $n$  và  $S$  ( $S \leq 10^9$ ). Dòng thứ hai chứa  $n$  số nguyên  $t[1], t[2], \dots, t[n]$  ( $t[i] \leq 10^9$ )

**Output:**

Với mỗi bộ test ghi ra số tờ tiền ít nhất phải trả.

Nếu không thể tìm được kết quả, in ra -1. **input**



1  
3 5  
1 4 5  
**output**

1

## DSA02012 - DI CHUYỂN TRONG MA TRẬN

Cho ma trận  $A[M][N]$ . Nhiệm vụ của bạn là đếm tất cả các đường đi từ phần tử  $A[0][0]$  đến phần tử  $A[M-1][N-1]$ . Bạn chỉ được phép dịch chuyển xuống dưới hoặc sang phải phần tử liền kề với vị trí hiện tại.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất là hai số  $M, N$  tương ứng với số hàng và số cột của ma trận; dòng tiếp theo đưa vào 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:  $1 \leq T \leq 10$ ;  $1 \leq M, N, A[i][j] \leq 100$ .

### Output:

- Đưa ra số cách di chuyển của mỗi test theo từng dòng.
- Giải thích test 1: Có 3 cách di chuyển là  $[1\ 4\ 5\ 6]$ ,  $[1\ 2\ 5\ 6]$  và  $[1\ 2\ 3\ 6]$ .

### input

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

### output

3  
2

## DSA02013 - SỐ NGUYÊN TỐ

Cho ba số  $N, P, S$ . Trong đó,  $P$  là một số nguyên tố. Nhiệm vụ của bạn là đưa ra tất cả  $N$  số nguyên tố tính từ  $P$  có tổng bằng  $S$ . Ví dụ với  $S = 28, P=7, N=2$  ta có kết quả  $11 + 17 = 28$ . Với  $N = 3, P = 2, S = 23$  ta có kết quả :  $\{3, 7, 13\}, \{5, 7, 11\}$

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là bộ ba số  $S, P, N$  được viết trên một dòng.
- $S, P, N$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10$ ;  $2 \leq S, P \leq 200$ .

**Output:**

- Với mỗi test, dòng đầu tiên in ra số lượng đáp án tìm được. Mỗi dòng tiếp theo in ra kết quả tìm được theo thứ tự từ điển.

**input**

```
2
2 7 28
3 2 23
```

**output**

```
1
11 17
2
3 7 13
5 7 11
```

**DSA02014 - TỪ ĐIỂN**

Cho tập từ ghi trong từ điển `dic[]` và một bảng hai chiều `A[M][N]` các ký tự. Hãy tạo nên tất cả các từ có mặt trong từ điển `dic[]` bằng cách nối các ký tự kề nhau trong mảng `A[][]`. Chú ý, phép nối các ký tự kề nhau trong mảng `A[][]` được thực hiện theo 8 hướng nhưng không có phần tử `A[i][j]` nào được lặp lại. Ví dụ với từ điển `dic[] = { "GEEKS", "FOR", "QUIZ", "GO" }` và mảng `A[][]` dưới đây sẽ cho ta kết quả: "GEEKS", "QUIZ"

G I Z

U E K

Q S E

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test `T`.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất đưa vào ba số `K, M, N` tương ứng với số từ của từ điển `dic[]`, số hàng và số cột của ma trận ký tự `A[M][N]`; dòng tiếp theo đưa vào `K` từ của từ điển `dic[]`; dòng cuối cùng đưa vào các phần tử `A[i][j]`.
- `T, K, M, N` thỏa mãn ràng buộc:  $1 \leq T \leq 10$ ;  $1 \leq K \leq 100$ ;  $1 \leq M, N \leq 3$ .

**Output:**

- Đưa ra theo thứ tự tăng dần các từ có mặt trong từ điển `dic[]` được tạo ra từ ma trận `A[][]`.  
Đưa ra -1 nếu không thể tạo ra từ nào thuộc `dic[]` từ `A[][]`.

**input**

```
1
4 3 3
GEEKS FOR QUIZ GO
G I Z
```

## DSA02015 - LOẠI BỎ DẤU NGOẶC

Cho biểu thức P chỉ chứa các ký tự '(', ')' và các ký tự. Không có phép toán nào trong biểu thức P. Nhiệm vụ của bạn là thực hiện ít nhất các phép loại bỏ các ký tự '(', ')' để P trở thành biểu thức đúng. Chú ý: một biểu thức chỉ có 1 ký tự chữ (không có dấu ngoặc) hoặc một biểu thức rỗng thì không được xem là biểu thức đúng.

Nếu có nhiều hơn một biểu thức đúng với cùng số phép loại bỏ ít nhất hãy đưa ra tất cả các biểu thức đúng theo thứ tự từ điển.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một biểu thức P được viết trên một dòng.
- T, P thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{length}(P) \leq 100$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng. Nếu không có đáp án, in ra -1.

### input

```
2
()()()
(u)()()
```

### output

```
((())) ()()
(u())() (u)()
```

## DSA02016 - SẮP XẾP QUÂN HẬU 1

Cho một bàn cờ vua có kích thước  $n * n$ , ta biết rằng quân hậu có thể di chuyển theo chiều ngang, dọc, chéo. Vấn đề đặt ra rằng, có n quân hậu, bạn cần đếm số cách đặt n quân hậu này lên bàn cờ sao cho với 2 quân hậu bất kì, chúng không "ăn" nhau.

**Input:** Dòng đầu ghi số bộ test T ( $T < 5$ ). Mỗi bộ test ghi một số nguyên dương n duy nhất (không quá 10)

**Output:** Ghi kết quả mỗi bộ test trên một dòng. Số cách đặt quân hậu.

Ví dụ:

### input

```
1
```

4

**output**

2

## DSA02017 - SẮP XẾP QUÂN HẬU 2

Cho một bàn cờ  $8 \times 8$ , mỗi ô có một giá trị  $A[i][j]$  nhất định ( $0 \leq A[i][j] \leq 100$ ), tương ứng với điểm số đạt được nếu như bạn đặt một quân cờ vào đó.

Nhiệm vụ của bạn là đặt 8 quân hậu lên bàn cờ, sao cho không có 2 quân nào ăn nhau, và số điểm đạt được là lớn nhất.

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test gồm 8 dòng, mỗi dòng 8 số nguyên mô tả bàn cờ.

**Output:**

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

**input**

1

1 2 3 4 5 6 7 8

9 10 11 12 13 14 15 16

17 18 19 20 21 22 23 24

25 26 27 28 29 30 31 32

33 34 35 36 37 38 39 40

41 42 43 44 45 46 47 48

49 50 51 52 53 54 55 56

57 58 59 60 61 62 63 64

**output**

260

## DSA02018 - BIỂU THỨC TOÁN HỌC

Cho 5 số nguyên dương  $A, B, C, D, E$ . Bạn có thể hoán vị các phần tử cho nhau, hãy đặt các dấu biểu thức  $+, -, *$  sao cho biểu thức sau đúng:

$$[[[A \circ(1) B] \circ(2) C] \circ(3) D] \circ(4) E = 23$$

Trong đó:  $\circ(1) \dots \circ(4)$  là các phép toán  $+, -, *$ .

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test gồm 5 số nguyên dương  $A, B, C, D, E$  có giá trị không vượt quá 100.

**Output:**

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

**input**

```
3
1 1 1 1 1
1 2 3 4 5
2 3 5 7 11
```

**output**

```
NO
YES
YES
```

**DSA02019 - ĐƯỜNG ĐI DÀI NHẤT**

Cho đồ thị vô hướng có  $N$  đỉnh và  $M$  cạnh. Bạn hãy tìm đường đi dài nhất trên đồ thị, sao cho mỗi cạnh chỉ được đi qua nhiều nhất 1 lần.

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 10$ ).
- Mỗi test bắt đầu bằng số nguyên  $N$  và  $M$  ( $1 \leq N, M \leq 20$ ).
- Các đỉnh đánh dấu từ  $0, 1, \dots, N-1$ .  $M$  dòng tiếp theo, mỗi dòng gồm 2 số  $u, v$  cho biết có cạnh nối giữa  $u$  và  $v$ .

**Output:**

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

**input**

```
2
3 2
0 1
1 2
15 16
0 2
1 2
2 3
3 4
3 5
4 6
5 7
6 8
7 8
7 9
8 10
9 11
10 12
11 12
10 13
12 14
```

**output**

```
2
12
```

## DSA02020 - SỐ NHỎ NHẤT CÓ N ƯỚC SỐ

Cho số nguyên dương  $N$ . Nhiệm vụ của bạn là tìm số  $K$  nhỏ nhất, sao cho  $K$  có đúng  $N$  ước. Input đảm bảo rằng đáp án không vượt quá  $10^{18}$ .

### Input:

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test gồm 1 số nguyên  $N$  ( $1 \leq N \leq 1000$ ).

### Output:

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

### input

2  
4  
6

### output

6  
12

## DSA02021 - KÝ TỰ ĐẶC BIỆT

Cho một chuỗi  $s$ . Chuỗi  $F(s)$  được xác định bằng cách ghép chuỗi  $s$  ban đầu với chuỗi  $s$  sau khi đã được quay vòng sang bên phải 1 ký tự (ký tự cuối cùng của  $s$  được chuyển lên đầu).

Thực hiện liên tiếp các bước cộng chuỗi như trên với chuỗi mới thu được, ta có được chuỗi  $X$ .

$$X = F_k(s) = F(F_{k-1}(s)) \text{ với } F_0(s) = s.$$

Nhiệm vụ của bạn là hãy xác định ký tự thứ  $N$  trong chuỗi  $X$  là ký tự nào?

**Input:** Dòng đầu ghi số bộ test  $T$  ( $T < 10$ ). Mỗi bộ test gồm một chuỗi  $s$  có độ dài không vượt quá 30 ký tự và số nguyên  $N$  ( $1 \leq N \leq 10^{18}$ ).

**Output:** Với mỗi bộ test ghi ra trên một dòng ký tự tìm được.

### input

1  
COW 8

### output

C

Giải thích test: COW -> COWWCO -> COWWCOOCOWWC. Ký tự thứ 8 là 'C'.

## DSA02022 - NGÀY ĐẶC BIỆT

Hôm nay là ngày 22/02/2022. Một ngày đặc biệt. Chỉ có hai chữ số là 0 và 2. Đồng thời viết xuôi viết ngược tám chữ số ta đều được cùng một kết quả.

Nhiệm vụ của bạn trong ngày đặc biệt này là hãy liệt kê hết tất cả các ngày hợp lệ chỉ có 2 chữ số 0 và 2.

Một số chú ý:

- Luôn có đủ 8 chữ số của ngày tháng năm và hai dấu gạch chéo.
- Ngày và tháng có thể có chữ số 0 ở đầu nhưng với năm thì không được phép. Tức là ta chỉ tính các ngày bắt đầu từ năm 2000 trở về sau.

Bài toán không có dữ liệu vào. Bạn chỉ cần liệt kê kết quả gồm tất cả các ngày tháng hợp lệ. Coi đây ký tự ngày tháng là một xâu ký tự. Hãy liệt kê theo đúng thứ tự từ điển tăng dần.

## DSA02023 - ĐẶT TÊN

Kỳ thi ICPC có K đội của PTIT tham gia và đội tuyển đang rất đau đầu không biết chọn các cái tên như thế nào cho các đội. Yêu cầu phải đảm bảo tên không có khoảng trống và không được trùng nhau. Sau khi thảo luận, có N cái tên được đề xuất (có thể bị trùng nhau). Với  $K < 15$  và  $4 < N < 30$ .

Hãy liệt kê tất cả danh sách các tổ hợp K cái tên khác nhau có thể được tạo ra theo thứ tự từ điển.

### Input

Dòng đầu ghi 2 số N và K.

Tiếp theo là 1 dòng ghi N cái tên, mỗi cái tên có độ dài không quá 15 và cách nhau một khoảng trống. Tất cả đều là ký tự in hoa.

### Output

Ghi ra tất cả các tổ hợp tên có thể được lựa chọn theo thứ tự từ điển.

Tức là các tên trong mỗi tổ hợp liệt kê theo thứ tự từ điển và các tổ hợp cũng được liệt kê theo thứ tự từ điển.

### input

```
6 2
DONG TAY NAM BAC TAY BAC
```

### output

```
BAC DONG
BAC NAM
BAC TAY
DONG NAM
DONG TAY
NAM TAY
```

## DSA02024 - DẪY CON TĂNG DẦN

Cho dãy số  $a[]$  có n phần tử là các số nguyên dương khác nhau từng đôi một. Hãy liệt kê tất cả các dãy con có từ 2 phần tử trở lên của dãy  $a[]$  thỏa mãn tính chất tăng dần.

Dãy con tạo được bằng cách lấy ra các phần tử trong dãy  $a[]$  nhưng vẫn giữ nguyên thứ tự ban đầu.

Coi mỗi dãy con như một xâu ký tự với các phần tử cách nhau một khoảng trống, hãy liệt kê theo thứ tự từ điển.

### Input

- Dòng đầu ghi số  $n$  (không quá 20)
- Dòng thứ 2 ghi  $n$  số của dãy  $a[]$ . Các số khác nhau từng đôi một và có giá trị không quá 100.

### Output

Ghi ra lần lượt các dãy con tăng dần theo thứ tự từ điển.

#### input

```
4
6 3 7 11
```

#### output

```
3 11
3 7
3 7 11
6 11
6 7
6 7 11
7 11
```

## DSA02025 - KÝ TỰ LẶP

Cho một dãy các xâu ký tự chỉ bao gồm các chữ cái in hoa từ A đến Z, trong đó các ký tự trong mỗi xâu đều đã được sắp xếp theo thứ tự từ điển và mỗi chữ cái chỉ xuất hiện nhiều nhất một lần (tức là độ dài xâu tối đa là 26). Nếu một ký tự xuất hiện trong hai xâu liên tiếp thì được coi là một lần lặp. Hãy tìm cách sắp xếp lại thứ tự các xâu sao cho số lần lặp là nhỏ nhất có thể. Ví dụ dưới đây là cùng một dãy xâu nhưng với cách sắp xếp lại thì số lần lặp chỉ còn 2.

ABC

ABEF

DEF

ABCDE

FGH

=> Số lần lặp là 6

ABEF

DEF

ABC



FGH

ABCDE

=> Số lần lặp là 2.

**Input** Dòng đầu tiên ghi số  $N$  ( $2 \leq N \leq 10$ ) là số xâu ký tự.  $N$  dòng tiếp theo, mỗi dòng ghi một xâu.

**Output** In ra trên một dòng số lần lặp nhỏ nhất có thể.

**input**

test 1

5  
ABC  
ABEF  
DEF  
ABCDE  
FGH

test 2

6  
BDE  
FGH  
DEF  
ABC  
BDE  
ABEF  
test 3

4  
XYZ  
XYZ  
ABYZ  
Z

**output**

test 1

2  
test 2

3  
test 3

4

## DSA02026 - PHÉP TOÁN CƠ BẢN

Cho một biểu thức trong phạm vi hai chữ số với các phép toán cộng trừ nhân chia. Các toán hạng và kết quả đảm bảo là số nguyên dương có hai chữ số, nếu có phép chia thì phải thỏa mãn tính chia hết.

Người ta có thể ẩn đi một số chữ số hoặc phép toán bằng cách điền dấu chấm hỏi (?). Nhiệm vụ của bạn là khôi phục các dấu chấm hỏi và in ra phép toán chính xác ban đầu. Nếu không thể có kết quả đúng thì ghi ra WRONG PROBLEM!

### Input

- Dòng đầu ghi số bộ test  $T$  ( $1 \leq T \leq 100$ ).
- $T$  dòng tiếp theo, mỗi dòng là một biểu thức có thể có các dấu ?.

*Nếu có thể có nhiều kết quả đúng thì in ra kết quả đầu tiên theo thứ tự từ điển, tức là số nhỏ nhất có thể tính từ trái sang phải.*

### Output

- Với mỗi bộ test, ghi ra biểu thức đúng tìm được. Hoặc WRONG PROBLEM!

### input

```
2
?0 ? 12 = 28
40 / ?3 = ??
```

### output

```
40 - 12 = 28
WRONG PROBLEM!
```

## DSA02027 - NGƯỜI DU LỊCH

Cho  $n$  thành phố đánh số từ 1 đến  $n$  và các tuyến đường giao thông hai chiều giữa chúng, mạng lưới giao thông này được cho bởi mảng  $C[1...n, 1...n]$  ở đây  $C[i][j] = C[j][i]$  là chi phí đi đoạn đường trực tiếp từ thành phố  $i$  đến thành phố  $j$ .

Một người du lịch xuất phát từ thành phố 1, muốn đi thăm tất cả các thành phố còn lại mỗi thành phố đúng 1 lần và cuối cùng quay lại thành phố 1. Hãy chỉ ra chi phí ít nhất mà người đó phải bỏ ra.

**Input** Dòng đầu tiên là số nguyên  $n$  – số thành phố ( $n \leq 15$ );  $n$  dòng sau, mỗi dòng chứa  $n$  số nguyên thể hiện cho mảng 2 chiều  $C$ .

**Output** Chi phí mà người đó phải bỏ ra.

### input

```
4
0 20 35 10
20 0 90 50
35 90 0 12
10 50 12 0
```

### output

```
117
```

## DSA02028 - CHIA ĐỀU

Cho dãy số A có N phần tử và số K. Hãy đếm số cách chia dãy A thành K nhóm các phần tử liên tiếp sao cho tổng giá trị của mỗi nhóm đều bằng nhau.

### Input

- Dòng đầu ghi hai số N và K ( $0 < N \leq 12$ ;  $0 < K < N$ ).
- Dòng thứ 2 ghi N số của dãy A ( $-10000 \leq A[i] \leq 10000$ )

### Output

- In ra số cách thỏa mãn

### input

test 1

3 2  
-2 0 -2

test 2

3 2  
1 2 3

### output

test 1

2

test 2

1

## DSA02029 - THÁP HÀ NỘI

Bài toán Tháp Hà Nội đã rất nổi tiếng. Bắt đầu có các đĩa xếp chồng lên cột A theo thứ tự kích thước giảm dần, nhỏ nhất ở trên cùng. Cột B và cột C ban đầu không có đĩa nào cả.

Mục tiêu của bạn là di chuyển toàn bộ các đĩa theo đúng thứ tự về cột C, tuân theo các quy tắc sau:

- Mỗi lần chỉ có thể di chuyển một đĩa.
- Mỗi lần di chuyển sẽ lấy đĩa trên từ một trong các cột và đặt nó lên trên một cột khác.
- Không được đặt đĩa lên trên đĩa nhỏ hơn..

### Input

Số tự nhiên  $0 < N < 10$

### Output

In ra lần lượt từng bước theo mẫu trong ví dụ. Chú ý giữa các chữ cái và dấu -> có khoảng trống.

### input

3

### output

A -> C  
A -> B  
C -> B  
A -> C  
B -> A  
B -> C  
A -> C

## DSA02030 - LIỆT KÊ XÂU KÝ TỰ

Cho chữ cái  $c$  in hoa ( $'A' < c < 'K'$ ) và số nguyên  $K$  ( $0 < K < (c - 'A')$ ).

Hãy tìm cách liệt kê tất cả các chuỗi ký tự khác nhau được tạo ra bởi các chữ cái tính từ  $'A'$  đến ký tự  $c$ . Các ký tự được phép lặp lại nhưng không tính các chuỗi là hoán vị của chuỗi nào đó đã liệt kê trước đó.

Xem ví dụ để hiểu thêm yêu cầu đề bài.

### Input

Chỉ có một dòng ghi chữ cái  $c$  và số nguyên  $K$  thỏa mãn ràng buộc đề bài.

### Output

Ghi ra lần lượt các chuỗi ký tự kết quả theo thứ tự từ điển, mỗi chuỗi trên một dòng.

### input

D 2

### output

AA  
AB  
AC  
AD  
BB  
BC  
BD  
CC  
CD  
DD

## DSA02031 - GHÉP CHỮ CÁI

Trong 8 chữ cái Tiếng Anh đầu tiên (ABCDEFGH) thì có 2 nguyên âm là  $'A'$ ,  $'E'$ , còn lại là phụ âm.

Người ta muốn ghép các chữ cái bắt đầu từ chữ cái  $'A'$  cho đến chữ cái  $c$  ( $'D' \leq c \leq 'H'$ ) sao cho mỗi chữ cái được sử dụng đúng một lần và chuỗi kết quả không có nguyên âm nào bị kẹp giữa bởi hai phụ âm.

Hãy liệt kê các chuỗi thỏa mãn theo thứ tự từ điển.

### Input

Chỉ có một dòng ghi chữ cái c (đảm bảo từ 'D' đến 'H').

### Output

Liệt kê các xâu thỏa mãn, mỗi xâu trên một dòng.

### input

D

### output

ABCD  
ABDC  
ACBD  
ACDB  
ADBC  
ADCB  
BCDA  
BDCA  
CBDA  
CDBA  
DBCA  
DCBA

## DSA02032 - TỔ HỢP SỐ CÓ TỔNG BẰNG X

Cho mảng  $A[]$  gồm  $N$  số nguyên dương phân biệt và số  $X$ . Nhiệm vụ của bạn là tìm phép tổ hợp các số trong mảng  $A[]$  có tổng bằng  $X$ . Các số trong mảng  $A[]$  có thể được sử dụng nhiều lần. Mỗi tổ hợp các số của mảng  $A[]$  được in ra theo thứ tự không giảm các số. Ví dụ với  $A[] = \{2, 4, 6, 8\}$ ,  $X = 8$  ta có các tổ hợp các số như sau:

$\{2, 2, 2, 2\}$ ,  $\{2, 2, 4\}$ ,  $\{2, 6\}$ ,  $\{4, 4\}$ ,  $\{8\}$ .

### Input

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất là hai số  $N$  và  $X$ ; dòng tiếp theo đưa vào  $N$  số của mảng  $A[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, X, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 10$ ;  $1 \leq X, A[i] \leq 100$ .  $N \leq 20$ .

### Output

- Đưa ra kết quả mỗi test theo từng dòng.
- Đầu tiên là số lượng tổ hợp thỏa mãn. Mỗi tổ hợp được bao bởi cặp ký tự  $\{ \}$  và cách nhau một dấu cách. Đưa ra -1 nếu không có tổ hợp nào thỏa mãn yêu cầu bài toán.

### input

2  
4 8  
2 4 6 8  
2 9  
10 11

### output

5 {2 2 2 2} {2 2 4} {2 6} {4 4} {8}  
-1

## DSA02033 - SỐ XA CÁCH

Cho số nguyên dương  $N$  ( $2 < N < 10$ ). Một số nguyên dương  $K$  có  $N$  chữ số được gọi là số xa cách nếu thỏa mãn:

- $K$  không chứa chữ số 0
- Tất cả các chữ số từ 1 đến  $N$  đều xuất hiện trong  $K$  đúng 1 lần
- Không có hai chữ số liên tiếp nào trong  $K$  có hiệu bằng 1.

Hãy liệt kê tất cả các số thỏa mãn theo thứ tự tăng dần.

### Input

- Dòng đầu ghi số bộ test (không quá 10)
- Mỗi bộ test là 1 số nguyên dương  $N$  ( $2 < N < 10$ )

### Output

Liệt kê tất cả các số thỏa mãn, mỗi số trên một dòng.

Sau mỗi test in ra một khoảng trống.

### Ví dụ

#### input

2  
3  
4

#### output

2413  
3142

## DSA02034 - TRÒ CHƠI VỚI CÁC CON SỐ

Nam có  $N$  thẻ bài, mỗi thẻ ghi một con số từ 1 tới  $N$ . Không có 2 thẻ nào ghi hai số trùng nhau. Nam muốn xếp thứ tự các thẻ bài sao cho không có hai con số nào liền kề nhau ở cạnh nhau.

Ví dụ với  $N = 4$  thì cách xếp 3142 là hợp lệ nhưng cách xếp 1324 không hợp lệ.

Hãy giúp Nam liệt kê tất cả các cách xếp thỏa mãn theo thứ tự từ điển.

### Input

- Dòng đầu ghi số bộ test  $T$  (không quá 10)
- Mỗi bộ test ghi số  $N$  ( $3 < N < 10$ )

### Output

- Với mỗi bộ test, ghi ra tất cả các cách xếp thứ tự thỏa mãn, mỗi cách trên một dòng.

**input**

1  
4

**output**

2413  
3142

## DSA02035 - HOÁN VỊ CÁC CHỮ SỐ

Độ chênh lệch của một dãy số nguyên dương được định nghĩa là hiệu của số lớn nhất trừ đi số nhỏ nhất trong dãy.

Cho hai số nguyên dương  $n$  và  $k$ , với  $1 \leq n, k \leq 8$ .

Với  $n$  xâu ký tự số có độ dài  $k$ , có thể có chữ số 0 ở đầu. Hãy tìm cách duyệt tất cả hoán vị các chữ số của từng số trong danh sách sao cho độ chênh lệch của dãy số tạo được là bé nhất có thể.

**Input:**

- Dòng đầu ghi hai số  $n$  và  $k$ .
- Tiếp theo là  $n$  dòng, mỗi dòng ghi một xâu ký tự đúng  $k$  chữ số.

**Output:**

- Ghi ra độ chênh lệch bé nhất có thể.

**input**

6 4  
5237  
2753  
7523  
5723  
5327  
2537

**output**

2700

## DSA02036 - DÃY CON CÓ TỔNG LẺ

Cho dãy số  $A[]$  có  $N$  phần tử là các số nguyên dương khác nhau từng đôi một.

Hãy sắp xếp dãy theo thứ tự giảm dần, sau đó liệt kê tất cả các dãy con (đúng thứ tự trước sau) của  $A[]$  có tổng các phần tử là số lẻ.

Các dãy con được liệt kê theo thứ tự từ điển tăng dần.

**Input:**

Dòng đầu ghi số bộ test, mỗi test có 2 dòng:

- Dòng đầu ghi số  $N$  ( $2 < N < 15$ )
- Dòng thứ 2 ghi  $N$  số của dãy  $A[]$ , các số đều nguyên dương, nhỏ hơn 100 và khác nhau từng đôi một.

**Output:** Với mỗi test, liệt kê tất cả các dãy con có tổng các phần tử là số lẻ theo thứ tự từ điển tăng dần, mỗi dãy con trên một dòng.

**input**

```
1
4
2 3 4 5
```

**output**

```
3
3 2
4 3
4 3 2
5
5 2
5 4
5 4 2
```

## DSA02037 - DÃY CON C  TỔNG NGUYÊN TỐ

Cho dãy số  $A[]$  có  $N$  phần tử là các số nguyên dương khác nhau từng đôi một. Hãy liệt kê tất cả các dãy con của  $A[]$  có tổng các phần tử là số nguyên tố.

Các dãy con được liệt kê theo thứ tự từ điển tăng dần.

**Input:**

Dòng đầu ghi số bộ test, mỗi test có 2 dòng:

- Dòng đầu ghi số  $N$  ( $2 < N < 15$ )
- Dòng thứ 2 ghi  $N$  số của dãy  $A[]$ , các số đều nguyên dương, nhỏ hơn 100 và khác nhau từng đôi một.

**Output:** Với mỗi test, liệt kê tất cả các dãy con có tổng các phần tử là số lẻ theo thứ tự từ điển tăng dần, mỗi dãy con trên một dòng.

**input**

```
1
4
3 2 5 4
```

**output**

```
2
3
3 2
4 3
5
```



5 2  
5 4 2

## DSA02038 - DÂY CON CÓ K PHẦN TỬ TĂNG DẦN

Cho dãy số  $A[]$  có  $N$  phần tử là các số nguyên dương khác nhau từng đôi một và một số  $K < N$ .

Hãy liệt kê tất cả các dãy con khác nhau có  $K$  phần tử của  $A[]$ , mỗi dãy đều được sắp xếp theo thứ tự tăng dần.

Các dãy con được liệt kê lần lượt theo thứ tự từ điển.

### Input:

Dòng đầu ghi số bộ test, mỗi test có 2 dòng:

- Dòng đầu ghi số  $N$  ( $2 < N < 15$ )
- Dòng thứ 2 ghi  $N$  số của dãy  $A[]$ , các số đều nguyên dương, nhỏ hơn 100 và khác nhau từng đôi một.

**Output:** Với mỗi test, liệt kê tất cả các dãy con thỏa mãn, mỗi dãy con trên một dòng.

### input

1  
4 3  
3 2 5 4

### output

2 3 4  
2 3 5  
2 4 5  
3 4 5

## DSA02039 - PHÂN TÍCH SỐ 2

Cho số nguyên dương  $N$ . Nhiệm vụ của bạn là hãy liệt kê tất cả các cách phân tích số tự nhiên  $N$  thành tổng các số tự nhiên nhỏ hơn hoặc bằng  $N$ . Phép hoán vị của một cách được xem là giống nhau. Ví dụ với  $N = 5$  ta có kết quả là: (5), (4, 1), (3, 2), (3, 1, 1), (2, 2, 1), (2, 1, 1, 1), (1, 1, 1, 1, 1) .

### Input:

- Dòng đầu tiên đưa vào số lượng test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một số tự nhiên  $N$  được viết trên một dòng.
- $T, n$  thỏa mãn ràng buộc:  $1 \leq T, N \leq 10$ .

### Output:

- Dòng đầu tiên là số lượng cách phân tích thỏa mãn. Dòng tiếp theo liệt kê đáp án theo mẫu ví dụ đã cho.

### input

2  
4  
5

**output**

5

(4) (3 1) (2 2) (2 1 1) (1 1 1 1)

7

(5) (4 1) (3 2) (3 1 1) (2 2 1) (2 1 1 1) (1 1 1 1 1)

## DSA02041 - BIẾN ĐỔI VỀ 1

Cho số nguyên dương  $N$ . Hãy đếm số bước ít nhất để đưa  $N$  về 1 bằng cách thực hiện ba thao tác dưới đây:

- Nếu  $N$  chia hết cho 2 bạn có thể giảm  $N = N/2$ .
- Nếu  $N$  chia hết cho 3 bạn có thể giảm  $N = N/3$ .
- Giảm  $N$  đi 1.

**Input:**

- Dòng đầu tiên đưa vào số lượng test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một số  $N$  được viết trên một dòng.
- $T, N$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 100000$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

2  
10  
6

**output**

3  
2

## DSA02045 - TẬP CON CỦA XÂU KÝ TỰ

Cho một chuỗi ký tự  $S$  không có ký tự lặp lại. Hãy đưa ra tất cả các tập con của chuỗi ký tự  $S$  theo thứ tự từ điển.

**Input:**

- Dòng đầu tiên đưa vào số lượng test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một chuỗi ký tự.
- $T, S$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{length}(S) \leq 16$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

1  
3  
abc

#### output

a ab abc ac b bc c

## DSA05009 - TẬP CON BẰNG NHAU

Cho tập các số  $A[] = (a_1, a_2, \dots, a_n)$ . Hãy kiểm tra xem ta có thể chia tập  $A[]$  thành hai tập con sao cho tổng các phần tử của hai tập con bằng nhau hay không. Đưa ra YES nếu có thể thực hiện được, ngược lại đưa ra NO.

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất đưa vào số  $N$  là số lượng phần tử của dãy số  $A[]$ ; dòng tiếp theo đưa vào  $N$  phần tử của dãy số  $A[]$ .
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 100$ ;  $1 \leq A[i] \leq 100$ .

#### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

2  
4  
1 5 11 5  
3  
1 3 5

#### output

YES  
NO

## DSAKT012 - ĐỔI TIỀN

Có  $n$  tờ tiền có giá trị  $t[1], t[2], \dots, t[n]$ .

Hãy tìm cách trả ít tờ tiền nhất với số tiền đúng bằng  $S$

Chú ý: Các tờ tiền có giá trị bất kỳ và có thể bằng nhau, mỗi tờ tiền chỉ được dùng một lần.

#### Input:

- Mỗi bộ test gồm 2 số nguyên  $n$  và  $S$  ( $n \leq 30$ ;  $S \leq 10^9$ ).
- Dòng thứ hai chứa  $n$  số nguyên  $t[1], t[2], \dots, t[n]$  ( $t[i] \leq 10^9$ )

**Output:**

- Ghi ra trên một dòng số tờ tiền ít nhất phải trả.
- Nếu không thể tìm được kết quả, in ra -1.

**input**

3 5

1 4 5

**output**

1

## Greedy

### DSA03001 - ĐỔI TIỀN

Tại ngân hàng có các mệnh giá bằng 1, 2, 5, 10, 20, 50, 100, 200, 500, 1000. Tổng số tiền cần đổi có giá trị bằng N. Hãy xác định xem có ít nhất bao nhiêu tờ tiền sau khi đổi tiền?

**input**

Dòng đầu tiên là số lượng bộ test T ( $T \leq 50$ ). Mỗi test gồm 1 số nguyên N ( $1 \leq N \leq 100\,000$ ).

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

**input**

2

70

121

**output**

2

3

### DSA03002 - NHẦM CHỮ SỐ

Trong một buổi học toán, giáo viên viết 2 số nguyên, A và B, và yêu cầu Tèo thực hiện phép cộng. Tèo không bao giờ tính toán sai, nhưng thỉnh thoảng cậu ta chép các con số một cách không chính xác. Lỗi duy nhất của là ghi nhầm '5' thành '6' hoặc ngược lại. Cho hai số, A và B, tính tổng nhỏ nhất và lớn nhất mà Tèo có thể nhận được.

**Input:** Có một dòng chứa hai số nguyên dương A và B ( $1 \leq A, B \leq 1\,000\,000$ ).

**Output:** In ra 2 số nguyên cách nhau một dấu cách, tổng nhỏ nhất và lớn nhất có thể nhận được.

	Test 1	Test 2	Test 3
input	11 25	1430 4862	16796 58786

	Test 1	Test 2	Test 3
output	36 37	6282 6292	74580 85582

### DSA03003 - TÌM MAX

Cho mảng  $A[]$  gồm  $N$  phần tử. Nhiệm vụ của bạn là tìm  $\max = \text{for}(i: 0 \text{ to } n - 1) A_i * i$  bằng cách sắp đặt lại các phần tử trong mảng. Chú ý, kết quả của bài toán có thể rất lớn vì vậy bạn hãy đưa ra kết quả lấy modulo với  $10^9+7$ .

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất đưa vào số phần tử của mảng  $N$ ; dòng tiếp theo đưa vào  $N$  số  $A[i]$  tương ứng với các phần tử của mảng  $A[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N$ ,  $A[i] \leq 10^7$ .

#### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
2
5
5 3 2 4 1
3
1 2 3
```

#### output

```
40
8
```

### DSA03004 - TỔNG NHỎ NHẤT

Cho mảng  $A[]$  gồm các số từ 0 đến 9. Nhiệm vụ của bạn là tìm tổng nhỏ nhất của hai số được tạo bởi các số trong mảng  $A[]$ . Chú ý, tất cả các số trong mảng  $A[]$  đều được sử dụng để tạo nên hai số.

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất đưa vào số phần tử của mảng  $N$ ; dòng tiếp theo đưa vào  $N$  số  $A[i]$  tương ứng với các phần tử của mảng  $A[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 20$ ;  $0 \leq A[i] \leq 9$ .

#### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
2
6
6 8 4 5 2 3
5
5 3 0 7 4
```

#### output

```
604
82
```

## DSA03005 - CHIA MẢNG THÀNH HAI MẢNG CON CÓ TỔNG LỚN NHẤT

Cho mảng  $A[]$  gồm  $N$  số nguyên không âm và số  $K$ . Nhiệm vụ của bạn là hãy chia mảng  $A[]$  thành hai mảng con có kích cỡ  $K$  và  $N-K$  sao cho hiệu giữa tổng hai mảng con là lớn nhất. Ví dụ với mảng  $A[] = \{8, 4, 5, 2, 10\}$ ,  $K=2$  ta có kết quả là 17 vì mảng  $A[]$  được chia thành hai mảng  $\{4, 2\}$  và  $\{8, 5, 10\}$  có hiệu của hai mảng con là  $23-6=17$  là lớn nhất.

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất đưa vào số phần tử của mảng  $N$  và số  $K$ ; dòng tiếp theo đưa vào  $N$  số  $A[i]$  tương ứng với các phần tử của mảng  $A[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, K, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq K < N \leq 50$ ;  $0 \leq A[i] \leq 1000$ .

#### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
2
5 2
8 4 5 2 10
8 3
1 1 1 1 1 1 1 1
```

#### output

```
17
2
```

## DSA03006 - SẮP XẾP THAM LAM

Cho mảng  $A[]$  gồm  $N$  số và thực hiện các thao tác theo nguyên tắc dưới đây:

- Ta chọn một mảng con sao cho phần tử ở giữa của mảng con cũng là phần tử ở giữa của mảng  $A[]$  (trong trường hợp  $N$  lẻ).
- Đảo ngược mảng con đã chọn trong mảng  $A[]$ . Ta được phép chọn mảng con và phép đảo ngược mảng con bao nhiêu lần tùy ý.

Ví dụ với mảng  $A[] = \{1, 6, \mathbf{3}, \mathbf{4}, \mathbf{5}, 2, 7\}$  ta có câu trả lời là Yes vì: ta chọn mảng con  $\{3, 4, 5\}$  và đảo ngược để nhận được mảng  $A[] = \{1, 6, \mathbf{5}, \mathbf{4}, \mathbf{3}, 2, 7\}$ , chọn tiếp mảng con  $\{6, 5, 4, 3, 2\}$  và đảo ngược ta nhận được mảng  $A[] = \{1, 2, 3, 4, 5, 6, 7\}$ . Hãy cho biết ta có thể sắp xếp được mảng  $A[]$  bằng cách thực hiện các thao tác kể trên hay không?

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất đưa vào số phần tử của mảng  $N$ ; dòng tiếp theo đưa vào  $N$  số  $A[i]$  tương ứng với các phần tử của mảng  $A[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 50$ ;  $0 \leq A[i] \leq 1000$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
7
1 6 3 4 5 2 7
7
1 6 3 4 5 7 2
```

**output**

```
Yes
No
```

## DSA03007 - GIÁ TRỊ NHỎ NHẤT CỦA BIỂU THỨC

Cho mảng  $A[], B[]$  đều có  $N$  phần tử. Nhiệm vụ của bạn là tìm giá trị nhỏ nhất của biểu thức  $P = A[0] * B[0] + A[1] * B[1] + \dots + A[N-1] * B[N-1]$  bằng cách trao đổi vị trí các phần tử của cả mảng  $A[]$  và  $B[]$ .

**Input:**

- Dòng đầu tiên đưa vào số lượng test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng thứ nhất đưa vào số phần tử của mảng  $N$ ; dòng tiếp theo đưa vào  $N$  số  $A[i]$ ; dòng cuối cùng đưa vào  $N$  số  $B[i]$  các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i], B[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^7$ ;  $0 \leq A[i], B[i] \leq 10^{18}$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
7
1 6 3 4 5 2 7
1 1 1 2 3 4 3
```

```
7
1 6 3 5 5 2 2
0 1 9 0 1 2 3
```

**output**

```
45
27
```

## DSA03008 - SẮP XẾP CÔNG VIỆC 1

Cho hệ gồm  $N$  hành động. Mỗi hành động được biểu diễn như một bộ đôi  $\langle S_i, F_i \rangle$  tương ứng với thời gian bắt đầu và thời gian kết thúc của mỗi hành động. Hãy tìm phương án thực hiện nhiều nhất các hành động được thực hiện bởi một máy hoặc một người sao cho hệ không xảy ra mâu thuẫn.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng thứ nhất đưa vào số lượng hành động  $N$ ; dòng tiếp theo đưa vào  $N$  số  $S_i$  tương ứng với thời gian bắt đầu mỗi hành động; dòng cuối cùng đưa vào  $N$  số  $F_i$  tương ứng với thời gian kết thúc mỗi hành động; các số được viết cách nhau một vài khoảng trống.
- $T, N, S_i, F_i$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N, F_i, S_i \leq 1000$ .

**Output:**

- Đưa số lượng lớn nhất các hành động có thể được thực thi bởi một máy hoặc một người.

**input**

```
1
6
1 3 0 5 8 5
2 4 6 7 9 9
```

**output**

```
4
```

## DSA03009 - SẮP XẾP CÔNG VIỆC 2

Cho  $N$  công việc. Mỗi công việc được biểu diễn như một bộ 3 số nguyên dương  $\langle \text{JobId}, \text{Deadline}, \text{Profit} \rangle$ , trong đó  $\text{JobId}$  là mã của việc,  $\text{Deadline}$  là thời gian kết thúc của việc,  $\text{Profit}$  là lợi nhuận đem lại nếu hoàn thành việc đó đúng hoặc trước thời gian. Thời gian tối thiểu để hoàn thành mỗi công việc là 1 đơn vị thời gian. Hãy cho biết lợi nhuận lớn nhất có thể thực hiện các việc với giả thiết mỗi việc được thực hiện đơn lẻ.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất là số lượng  $\text{Job}$   $N$ ; phần thứ hai đưa vào  $3 \times N$  số tương ứng với  $N$  job.
- $T, N, \text{JobId}, \text{Deadline}, \text{Profit}$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 1000$ ;  $1 \leq \text{JobId} \leq 1000$ ;  $1 \leq \text{Deadline} \leq 1000$ ;  $1 \leq \text{Profit} \leq 1000$ .



**Output:**

- Đưa số lượng công việc tương ứng và lợi nhuận lớn nhất có thể đạt được.

**input**

```

2
4
1 4 20
2 1 10
3 1 40
4 1 30
5
1 2 100
2 1 19
3 2 27
4 1 25
5 1 15

```

**output**

```

2 60
2 127

```

**DSA03010 - NỐI DÂY 1**

Cho N sợi dây với độ dài khác nhau được lưu trong mảng A[]. Nhiệm vụ của bạn là nối N sợi dây thành một sợi sao cho tổng chi phí nối dây là nhỏ nhất. Biết chi phí nối sợi dây thứ i và sợi dây thứ j là tổng độ dài hai sợi dây A[i] và A[j].

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào số lượng sợi dây N; dòng tiếp theo đưa vào N số A[i] là độ dài của các sợi dây; các số được viết cách nhau một vài khoảng trống.
- T, N, A[i] thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^6$ ;  $0 \leq A[i] \leq 10^6$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```

2
4
4 3 2 6
5
4 2 7 6 9

```

**output**

```

29
62

```

**DSA03011 - NỐI DÂY 2**

Cho N sợi dây với độ dài khác nhau được lưu trong mảng A[]. Nhiệm vụ của bạn là nối N sợi dây thành một sợi sao cho tổng chi phí nối dây là nhỏ nhất. Biết chi phí nối sợi dây thứ i và sợi dây thứ j là tổng độ dài hai sợi dây A[i] và A[j].

**Input:**

- Dòng đầu ghi số bộ test T ( $T < 10$ ). Mỗi bộ test gồm 2 dòng. Dòng đầu tiên là số nguyên N ( $N \leq 2 \cdot 10^6$ ).
- Dòng tiếp theo gồm N số nguyên dương c[i] ( $1 \leq A[i] \leq 10^9$ ).

**Output:**

- In ra đáp án của bộ test trên từng dòng, theo modulo  $10^9+7$ .

**input**

```
1
7
2 4 1 2 10 2 3
```

**output**

```
59
```

## DSA03012 - SẮP ĐẶT XÂU KÝ TỰ 1

Cho xâu ký tự S bao gồm các ký tự in thường. Nhiệm vụ của bạn là kiểm tra xem ta có thể sắp đặt lại các ký tự trong S để hai ký tự giống nhau đều không kề nhau hay không? Đưa ra 1 nếu có thể sắp đặt lại các ký tự trong S thỏa mãn yêu cầu bài toán, ngược lại đưa ra -1.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một xâu ký tự S được viết trên một dòng.
- T, S thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{length}(S) \leq 10000$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
3
geeksforgeeks
bbbabaaacd
bbbbbb
```

**output**

```
1
1
-1
```

## DSA03013 - SẮP ĐẶT XÂU KÝ TỰ 2

Cho xâu ký tự S bao gồm các ký tự và số D. Nhiệm vụ của bạn là kiểm tra xem ta có thể sắp đặt lại các ký tự trong S để tất cả các ký tự giống nhau đều có khoảng cách là D hay không? Đưa ra 1 nếu có thể sắp đặt lại các ký tự trong S thỏa mãn yêu cầu bài toán, ngược lại đưa ra -1.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất là số D; dòng tiếp theo là xâu S.
- T, S, D thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{length}(S) \leq 10000$ ;  $1 \leq D \leq 100$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
2
ABB
2
AAA
```

**output**

```
1
-1
```

## DSA03014 - SỐ KHỐI LẬP PHƯƠNG

Một số X được gọi là số khối lập phương nếu X là lũy thừa bậc 3 của số Y ( $X = Y^3$ ). Cho số nguyên dương N, nhiệm vụ của bạn là tìm số khối lập phương lớn nhất bằng cách loại bỏ đi các chữ số của N. Ví dụ số 4125 ta có kết quả là  $125 = 5^3$ .

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một số tự nhiên N được viết trên một dòng.
- T, N thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^{18}$ .

**Output:**

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

**input**

```
2
4125
976
```

**output**

```
125
-1
```

## DSA03015 - MUA LƯƠNG THỰC

Giả sử bạn là một người nghèo trong địa phương của bạn. Địa phương của bạn có duy nhất một cửa hàng bán lương thực. Cửa hàng của bạn mở cửa tất cả các ngày trong tuần ngoại trừ chủ nhật. Cho bộ ba số  $N, S, M$  thỏa mãn ràng buộc sau:

- $N$  : số đơn vị lương thực nhiều nhất bạn có thể mua trong ngày.
- $S$  : số lượng ngày bạn cần được sử dụng lương thực để tồn tại.
- $M$  : số đơn vị lương thực cần có mỗi ngày để bạn tồn tại.

Giả sử bạn đang ở ngày thứ 2 trong tuần và cần tồn tại trong  $S$  ngày tới. Hãy cho biết số lượng ngày ít nhất bạn cần phải mua lương thực từ cửa hàng để tồn tại hoặc bạn sẽ bị chết đói trong  $S$  ngày tới.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là bộ 3 số  $N, S, M$  được viết trên một dòng.
- $T, N, S, M$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N, S, M \leq 30$ .

### Output:

- Đưa ra số ngày ít nhất bạn có thể mua lương thực để tồn tại hoặc đưa ra -1 nếu bạn bị chết đói.

### input

```
2
16 10 2
20 10 30
```

### output

```
2
-1
```

## DSA03016 - SỐ NHỎ NHẤT

Cho hai số nguyên dương  $S$  và  $D$ , trong đó  $S$  là tổng các chữ số và  $D$  là số các chữ số của một số. Nhiệm vụ của bạn là tìm số nhỏ nhất thỏa mãn  $S$  và  $D$ ? Ví dụ với  $S = 9, D = 2$  ta có số nhỏ nhất thỏa mãn  $S$  và  $D$  là 18.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là bộ 2 số  $S$  và  $D$  được viết trên một dòng.
- $T, S, D$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq S, D \leq 1000$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng. Nếu không có đáp án, in ra -1.

**input**

2  
9 2  
20 3

**output**

18  
299

## DSA03017 - GIÁ TRỊ NHỎ NHẤT CỦA XÂU

Cho xâu ký tự  $S$ . Ta gọi giá trị của xâu  $S$  là tổng bình phương số lần xuất hiện mỗi ký tự trong  $S$ . Hãy tìm giá trị nhỏ nhất của xâu  $S$  sau khi thực hiện  $K$  lần loại bỏ ký tự.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất là số  $K$ ; phần thứ hai là một xâu ký tự  $S$  được viết trên một dòng.
- $DT, S, K$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{length}(S) \leq 10000$ ;  $1 \leq K \leq 1000$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

2  
2  
ABCCBC  
2  
AAAB

**output**

6  
2

## DSA03018 - SỐ MAY MẮN

Hoàng yêu thích các số may mắn. Ta biết rằng một số là số may mắn nếu biểu diễn thập phân của nó chỉ chứa các chữ số may mắn là 4 và 7. Ví dụ, các số 47, 744, 4 là số may mắn và 5, 17, 467 không phải. Hoàng muốn tìm số may mắn bé nhất có tổng các chữ số bằng  $n$ . Hãy giúp anh ấy

**Input:** Dòng đầu ghi số bộ test, mỗi bộ test có một dòng chứa số nguyên  $n$  ( $1 \leq n \leq 106$ ) — tổng các chữ số của số may mắn cần tìm.

**Output:** In ra trên 1 dòng số may mắn bé nhất, mà tổng các chữ số bằng  $n$ . Nếu không tồn tại số thỏa mãn, in ra -1.

**input**

2  
11  
10  
**output**

47  
-1

## DSA03019 - PHÂN SỐ ĐƠN VỊ

Một phân số đơn vị nếu tử số của phân số đó là 1. Mọi phân số nguyên dương đều có thể biểu diễn thành tổng các phân số đơn vị. Ví dụ  $2/3 = 1/2 + 1/6$ . Cho phân số nguyên dương  $P/Q$  bất kỳ ( $P < Q$ ), hãy biểu diễn phân số nguyên dương thành tổng phân số đơn vị với số hạng tử là ít nhất.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là bộ đôi tử số  $P$  và mẫu số  $Q$  của phân số nguyên dương được viết trên một dòng.
- $T, P, Q$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq P, Q \leq 100$ .

### Output:

- Đưa ra đáp án tìm được trên 1 dòng, theo dạng " $1/a + 1/b + \dots$ "

### input

2  
2 3  
1 3  
**output**

$1/2 + 1/6$   
 $1/3$

## DSA03020 - BIỂU THỨC ĐÚNG

Cho một mảng  $S$  gồm  $2 \times N$  ký tự, trong đó có  $N$  ký tự '[' và  $N$  ký tự ']'. Xâu  $S$  được gọi là viết đúng nếu  $S$  có dạng  $S_2[S_1]$  trong đó  $S, S_2$  là các xâu viết đúng. Nhiệm vụ của bạn là tìm số các phép đổi chỗ ít nhất các ký tự kề nhau của xâu  $S$  viết sai để  $S$  trở thành viết đúng. Ví dụ với xâu  $S = "[]][["$  ta có số phép đổi chỗ kề nhau ít nhất là 2.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một xâu  $S$  viết sai theo nguyên tắc kể trên.
- $T, S$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{length}(S) \leq 100000$ .

### Output:

- Đưa kết quả trên một dòng.

**input**

```
2
[ ][ ][ ]
[ ][ ][ ]
```

**output**

```
2
0
```

## DSA03021 - TÌM DÃY SỐ

Cho dãy số nguyên  $A[]$  gồm có  $N$  phần tử. Nhiệm vụ của bạn là tìm dãy số  $B[]$  có tổng phần tử nhỏ nhất thỏa mãn tính chất  $A[i] / B[i] = A[i+1] / B[i+1]$  với mọi chỉ số  $i$  ( $0 \leq i \leq N-2$ ).

Phép chia trong bài toán này là phép chia nguyên (tức là chỉ lấy phần nguyên của kết quả: ví dụ  $5/3 = 1$ ).

**Input:**

- Dòng đầu tiên là số lượng phần tử  $N$  ( $1 \leq N \leq 1000$ ).
- Dòng tiếp theo gồm  $N$  số nguyên  $A[i]$  ( $1 \leq A[i] \leq 2000$ ).

**Output:**

- In ra một số nguyên là tổng các phần tử của dãy số  $B[]$  tìm được.

**input**

```
5
18 27 16 22 6
```

**output**

```
25
```

*Giải thích test: Dãy  $B[]$  tìm được là 5, 7, 5, 6, 2.*

## DSA03022 - TÍCH LỚN NHẤT

Cho dãy số  $A$  gồm  $N$  phần tử là các số nguyên. Hãy tính tích lớn nhất của **2 hoặc 3** phần tử trong dãy.

**Input:**

- Dòng đầu tiên ghi số  $N$  ( $3 \leq N \leq 10000$ )
- Dòng thứ 2 ghi  $N$  số của dãy  $A$  ( $|A_i| \leq 1000$ )

**Output:**

- Ghi ra kết quả trên một dòng

**input**

6  
5 10 -2 3 5 2  
**output**

250

## DSA03023 - SẮP XẾP VIÊN BI

Cho một dãy  $N$  viên bi gồm 3 màu xanh, trắng, đỏ xếp lẫn lộn. Bằng cách đổi chỗ từng cặp viên bi cho nhau có thể xếp lại dãy bi trên sao cho các viên bi xanh đứng trước, sau đó đến các viên bi trắng và cuối cùng là các viên bi đỏ. Tìm số lượng ít nhất các phép đổi chỗ cần thực hiện

### Input:

- Dòng đầu tiên ghi  $N$  ( $N \leq 100$ )
- Dòng thứ hai ghi sáu ký tự mô tả dãy bi (T-trắng, X-xanh, D-đỏ).

### Output:

- Một dòng duy nhất ghi số phép đổi chỗ tối thiểu cần thực hiện

### input

9  
TTXDDDTDX  
**output**

4

## DSA03024 - LỰA CHỌN TỐI ƯU

Bạn được giao cho  $N$  công việc, công việc thứ  $i$  có thời gian bắt đầu là  $A[i]$  và kết thúc tại  $B[i]$ . Tại một thời điểm, bạn chỉ có thể làm một công việc.

Bạn hãy lựa chọn các công việc một cách tối ưu sao cho số công việc làm được là nhiều nhất.

### Input:

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 10$ ).
- Mỗi test gồm 1 số nguyên  $N$  ( $1 \leq N \leq 100\,000$ ).
- $N$  dòng tiếp theo, mỗi dòng gồm 2 số  $A[i]$  và  $B[i]$  ( $0 \leq A[i] < B[i] \leq 10^6$ ).

### Output:

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

### input

1  
6  
5 9  
1 2  
3 4



0 6  
5 7  
8 9

**output**

4

*Giải thích test: Lựa chọn công việc 2, 3, 5, 6.*

## DSA03025 - TRỤC TỌA ĐỘ

Trên trục Ox tính từ vị trí 0, người ta muốn xếp nhiều nhất các đoạn thẳng sao cho không đoạn nào chồng lấn lên nhau. Đoạn thẳng thứ  $i$  có vị trí bắt đầu là  $X1[i]$  và kết thúc tại  $X2[i]$ , với  $X1[i] \leq X2[i]$ .

Hãy tính số đoạn thẳng nhiều nhất có thể được lựa chọn để đưa lên trục Ox và không có đoạn nào chồng lấn lên nhau.

**Input:**

- Dòng đầu tiên ghi số bộ test, không quá 10.
- Với mỗi bộ test: dòng đầu ghi số  $N$  là số đoạn thẳng (không quá  $10^5$ )
- Tiếp theo là  $N$  dòng, mỗi dòng có 2 số nguyên mô tả đoạn thẳng. Các giá trị tọa độ đều là các số nguyên không âm và không quá  $10^6$ .

**Output:**

- Với mỗi test, viết trên 1 dòng số lượng đoạn thẳng nhiều nhất có thể được lựa chọn thỏa mãn điều kiện đề bài.

**input**

1  
10  
39 55  
37 74  
0 1  
19 25  
65 76  
51 52  
19 21  
5 94  
46 65  
32 40

**output**

5

## DSA03026 - LỰA CHỌN THAM LAM

Cho hai số nguyên dương  $N$  và  $S$ . Hãy lựa chọn các chữ số phù hợp để tạo ra số nhỏ nhất và số lớn nhất có  $N$  chữ số sao cho tổng chữ số đúng bằng  $S$ .

**Input:**

Chỉ có một dòng ghi hai số N và S. ( $0 < N \leq 100$ ;  $0 \leq S \leq 900$ )

**Output:**

Ghi ra hai số nhỏ nhất và lớn nhất tìm được, cách nhau một khoảng trống.

Nếu không thể tìm được thì ghi ra "-1 -1"

**input**

3 20

---

2 900

---

3 0

**output**

299 992

---

-1 -1

---

-1 -1

## DSA03027 - GIÁ TRỊ LỚN NHẤT

Cho dãy số nguyên A[] có N phần tử.

- Gọi  $f(i,j) = |a_i| + |a_{i+1}| + \dots + |a_j|$
- Gọi  $g(i,j) = a_i + a_{i+1} + \dots + a_j$

Với tất cả các cặp  $1 \leq i \leq j \leq N$ .

Hãy tính giá trị lớn nhất của  $f(i,j) + g(i,j)$ .

**Input:**

- Dòng đầu ghi số N ( $1 \leq N \leq 50000$ )
- Dòng thứ 2 ghi N số nguyên của dãy A[]

**Output:**

- Ghi ra giá trị lớn nhất của  $f(i,j) + g(i,j)$

**input**

5

-3 5 -10 8 -2

**output**

26

## DSA03033 - ATM THẾ HỆ MỚI

Tại một cây rút tiền ATM thế hệ mới, Tý muốn rút ra W đồng. Các tờ tiền trong máy ATM có mệnh giá từ 1000, 2000, 3000, 5000,  $1000 * 10$ ,  $2000 * 10$ ,  $3000 * 10$ ,  $5000 * 10$ , ...,  $1000 * 10^c$ ,  $2000 * 10^c$ ,  $3000 * 10^c$ ,  $5000 * 10^c$ .

Chiếc máy ATM đã được tối ưu hóa sao cho số lượng tờ tiền đưa ra cho khách hàng là nhỏ nhất.

Các bạn hãy thử tính xem Tý sẽ nhận được bao nhiêu tờ tiền? Và có bao nhiêu cách thỏa mãn?

**Input:**

- Số đầu tiên là số lượng bộ test T ( $T \leq 1000$ ). Mỗi test gồm 2 dòng:
- Dòng thứ nhất chứa số nguyên W.
- Dòng thứ hai chứa số nguyên c.

**Output:**

- Với mỗi test, hãy in ra số lượng tờ tiền nhỏ nhất mà máy ATM sinh ra và số tổ hợp các tờ tiền khác nhau thỏa mãn tổng giá trị của chúng bằng W.
- Nếu không có đáp án, in ra số 0 trên một dòng.

**input**

```
4
2000
1
7000
1
1500
1
110000
1
```

**output**

```
1 1
2 1
0
3 2
```

Subtask 1: 30% số test đầu tiên,  $c \leq 5$ ,  $W \leq 10^9$ .

Subtask 2: 70% số test còn lại,  $c \leq 15$ ,  $W \leq 10^{18}$ .

*Giải thích test 4: 2 cách tìm được là:  $50000+50000+10000$  và  $50000+30000+30000$*

## DSA03034 - DÂY CON CHUNG -- DSA06040 - GIAO CỦA BA DÃY SỐ

Cho dãy số A[], B[] và C[] là dãy không giảm và có lần lượt N, M, K phần tử. Nhiệm vụ của bạn là hãy tìm các phần tử chung của 3 dãy số này.

**Input:**

- Dòng đầu tiên là số lượng bộ test T ( $T \leq 20$ ).
- Mỗi test gồm số nguyên N, M và K ( $1 \leq N, M, K \leq 100\,000$ ).

- Dòng tiếp theo gồm N số nguyên  $A[i]$ , rồi M số nguyên  $B[i]$  và K số nguyên  $C[i]$ . ( $0 \leq A[i], B[i], C[i] \leq 10^9$ ).

**Output:**

- Với mỗi test, in ra trên một dòng là đáp án thu được. Nếu không tìm được đáp án, in ra "NO".

**input**

```
3
6 5 8
1 5 10 20 40 80
5 7 20 80 100
3 4 15 20 30 70 80 120
3 5 4
1 5 5
3 4 5 5 10
5 5 10 20
3 3 3
1 2 3
4 5 6
7 8 9
```

**output**

```
20 80
5 5
NO
```

## DSA03035 - SẮP XẾ ĐƠN GIẢN

Cho một dãy số  $a[]$  có n phần tử gồm các số từ 1 đến n theo 1 thứ tự ngẫu nhiên. Nhiệm vụ của bạn là sắp xếp lại dãy số này theo thứ tự tăng dần với điều kiện : ở mỗi bước sắp xếp, bạn chỉ được chọn 1 số ở 1 vị trí bất kì và chuyển số đó lên đầu dãy hoặc về cuối dãy.

Hãy tính số bước tối thiểu cần thực hiện để hoàn thành việc sắp xếp.

**Input:**

- Dòng đầu tiên ghi 1 số n: số lượng phần tử của dãy a ( $1 \leq n \leq 100000$ )
- Dòng tiếp theo gồm n số từ 1 đến n theo thứ tự ngẫu nhiên

**Output:**

- Một số nguyên duy nhất là số bước tối thiểu cần thực hiện để hoàn thành việc sắp xếp.

**input**

```
5
4 1 2 5 3
```

**output**

```
2
```

# Divide and Conquer

## DSA04001 - LŨY THỪA

Cho số nguyên dương  $N$  và  $K$ . Hãy tính  $N^K$  modulo  $10^9+7$ .

### Input:

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

Mỗi test gồm 1 số nguyên  $N$  và  $K$  ( $1 \leq N \leq 1000$ ,  $1 \leq K \leq 10^9$ ).

### Output:

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

#### input

```
2
2 3
4 2
```

#### output

```
8
16
```

## DSA04002 - LŨY THỪA ĐẢO

Cho mảng số  $N$ . Ta gọi số đảo của  $N$  là  $R$ . Hãy tìm lũy thừa  $R$  của  $N$ . Đưa ra kết quả của bài toán dưới dạng modulo với  $10^9 + 7$ .

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm là số  $N$  được ghi trên một dòng.
- $T, N$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^{10}$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
2
2
12
```

#### output

```
4
864354781
```

## DSA04003 - ĐẾM DẤY

Cho số nguyên dương  $n$ . Hãy cho biết có bao nhiêu dãy số nguyên dương có tổng các phần tử trong dãy bằng  $n$ .

**Dữ liệu vào:** dòng đầu tiên chứa số nguyên  $T$  là số bộ dữ liệu, mỗi bộ dữ liệu ghi một số nguyên dương  $n$  duy nhất không qua 1018.

**Kết quả:** Mỗi bộ dữ liệu ghi ra một số nguyên duy nhất là số dư của kết quả tìm được khi chia cho 123456789.

**input**

1

3

**output**

4

## DSA04004 - GẤP ĐÔI DÃY SỐ

Một dãy số tự nhiên bắt đầu bởi con số 1 và được thực hiện  $N-1$  phép biến đổi "gấp đôi" dãy số như sau:

Với dãy số  $A$  hiện tại, dãy số mới có dạng  $A, x, A$  trong đó  $x$  là số tự nhiên bé nhất chưa xuất hiện trong  $A$ .

Ví dụ với 2 bước biến đổi, ta có  $[1] \rightarrow [1\ 2\ 1] \rightarrow [1\ 2\ 1\ 3\ 1\ 2\ 1]$ .

Các bạn hãy xác định số thứ  $K$  trong dãy số cuối cùng là bao nhiêu?

**Input:**

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

Mỗi test gồm số nguyên dương  $N$  và  $K$  ( $1 \leq N \leq 50, 1 \leq K \leq 2N - 1$ ).

**Output:**

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

**input**

2

3 2

4 8

**output**

2

4

## DSA04005 - DÃY XÂU FIBONACI

Một dãy xâu ký tự  $G$  chỉ bao gồm các chữ cái  $A$  và  $B$  được gọi là dãy xâu Fibonacci nếu thỏa mãn tính chất:  $G(1) = A$ ;  $G(2) = B$ ;  $G(n) = G(n-2) + G(n-1)$ . Với phép cộng (+) là phép nối hai xâu với nhau. Bài toán đặt ra là tìm ký tự ở vị trí thứ  $i$  (tính từ 1) của xâu Fibonacci thứ  $n$ .

**Dữ liệu vào:** Dòng 1 ghi số bộ test. Mỗi bộ test ghi trên một dòng 2 số nguyên  $N$  và  $i$  ( $1 < N < 93$ ). Số  $i$  đảm bảo trong phạm vi của xâu  $G(N)$  và không quá 18 chữ số.

**Kết quả:** Ghi ra màn hình kết quả tương ứng với từng bộ test.

**input**

2  
6 4  
8 19

**output**

A  
B

## DSA04006 - ĐẾM SỐ BÍT 1

Cho số nguyên dương  $N$ . Mỗi bước, bạn sẽ biến đổi  $N$  thành  $\lfloor N/2 \rfloor$ ,  $N \bmod 2$ ,  $\lfloor N/2 \rfloor$ . Sau khi thực hiện một cách triệt để, ta thu được một dãy số chỉ toàn số 0 và 1.

Nhiệm vụ của bạn là hãy đếm các số bằng 1 trong đoạn  $[L, R]$  của dãy số cuối cùng.

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test gồm 3 số nguyên  $N, L, R$  ( $1 \leq N, L, R < 2^{50}$ ,  $0 \leq R-L \leq 100\,000$ ).

**Output:**

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

**input**

2  
7 2 5  
10 3 10

**output**

4  
5

Giải thích test 1:  $[7] \rightarrow [3, 1, 3] \rightarrow [1, 1, 1, 1, 3] \rightarrow [1, 1, 1, 1, 1, 1]$ .

Giải thích test 2: Dãy số sau khi biến đổi là  $[1, 0, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 1]$ .

## DSA04007 - HỆ CƠ SỐ K

Cho hai số  $A, B$  ở hệ cơ số  $K$ . Hãy tính tổng hai số đó ở hệ cơ số  $K$ .

**Input:**

- Dòng đầu ghi số bộ test  $T$  ( $T < 10$ ). Mỗi bộ test ghi 3 số  $K, A, B$ .
- $2 \leq K \leq 10$ ;  $A$  và  $B$  nếu biểu diễn trong hệ cơ số 10 đều nhỏ hơn  $10^9$

**Output:**

- In ra tổng của A và B trong hệ cơ số K

**input**

1  
2 1 10

**output**

11

**DSA04008 - SỐ FIBONACCI THỨ N**

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

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

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

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.

**input**

3  
2  
6  
20

**output**

1  
8  
6765

**DSA04009 - LŨY THỪA MA TRẬN 1**

Cho ma trận vuông A kích thước N x 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  $X$ .

**input**

```
2
2 5
1 1
1 0
3 1000000000
1 2 3
4 5 6
7 8 9
```

**output**

```
8 5
5 3
597240088 35500972 473761863
781257150 154135232 527013321
965274212 272769492 580264779
```

## DSA04010 - DÃY CON LIÊN TIẾP CÓ TỔNG LỚN NHẤT

Cho mảng  $A[]$  gồm  $N$  số có cả các số âm và số dương. Nhiệm vụ của bạn là tìm mảng con liên tục có tổng lớn nhất của mảng. Ví dụ với mảng  $A[] = \{-2, -5, 6, -2, -3, 1, 5, -6\}$  ta có kết quả là 7 tương ứng với dãy con  $\{6, -2, -3, 1, 5\}$ .

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất đưa vào hai số  $N$  tương ứng với số phần tử của mảng; dòng tiếp theo đưa vào  $N$  số  $A[i]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 100$ ;  $-100 \leq A[i] \leq 100$ .

**Output:**

- Đưa ra tổng con liên tục lớn nhất của mỗi test theo từng dòng.

**input**

```
1
8
-2 -5 6 -2 -3 1 5 -6
```

**output**

```
7
```

## DSA04011 - TÍCH HAI SỐ NHỊ PHÂN

Cho hai xâu nhị phân biểu diễn hai số. Nhiệm vụ của bạn là đưa ra tích của hai số. Ví dụ với xâu  $S1 = "1100"$  và  $S2 = "1010"$  ta sẽ có kết quả là 120.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 2 hai xâu nhị phân  $S1, S2$  được viết trên một dòng.
- $T, S1, S2$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{length}(S1), \text{length}(S2) \leq 30$ .

**Output:**

- Đưa ra tích của mỗi test theo từng dòng.

**input**

```
2
1100 01
01 01
```

**output**

```
12
1
```

## DSA04012 - TÍCH ĐA THỨC

Cho hai đa thức  $P$  và  $Q$  được biểu diễn như một mảng bao gồm các hệ số của đa thức. Ví dụ với  $P(x) = 5 + 0x^1 + 10x^2 + 6x^3$  được biểu diễn như mảng  $P[] = \{5, 0, 10, 6\}$ . Hãy đưa ra đa thức  $R = P \times Q$  theo các hệ số của  $R$  với cách biểu diễn như trên.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng thứ nhất đưa vào hai số  $M, N$  tương ứng với lũy thừa lớn nhất của đa thức  $P$  và  $Q$ ; dòng tiếp theo đưa vào  $M$  số là hệ số của đa thức  $P$ ; dòng cuối cùng đưa vào  $N$  số là hệ số của đa thức  $Q$ .
- $T, M, N, P[i], Q[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq M, N \leq 100$ ;  $1 \leq P[i], Q[i] \leq 100$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
4 3
1 0 3 2
2 0 4
5 4
1 9 3 4 7
4 0 2 5
```

**output**

2 0 10 4 12 8  
4 36 14 39 79 23 34 35

## DSA04013 - 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  $X \geq 2 \cdot Y$ .

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.

### input

2  
8  
2 5 7 6 9 8 4 2  
8  
9 1 6 2 6 5 8 3

### output

5  
5

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

## DSA04014 - CẶP NGHỊCH THỂ

Cho mảng  $A[]$  gồm  $N$  phần. Ta gọi cặp nghịch thể của mảng  $A[]$  là số các cặp  $i, j$  sao cho  $i < j$  và  $A[i] > A[j]$ . Đối với mảng đã được sắp xếp thì số cặp nghịch thể bằng 0. Mảng đã sắp theo thứ tự giảm dần có số đảo ngược cực đại. Nhiệm vụ của bạn là hãy đưa ra số cặp nghịch thể của mảng  $A[]$  gồm  $N$  phần tử.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất đưa vào số  $N$  tương ứng với số phần tử của mảng  $A[]$ ; phần thứ 2 là  $N$  số của mảng  $A[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^7$ ;  $1 \leq A[i] \leq 10^{18}$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
5
2 4 1 3 5
5
5 4 3 2 1
```

**output**

```
3
10
```

**DSA04015 - TÍNH FLOOR(X)**

Cho mảng đã được sắp xếp  $A[]$  gồm  $N$  phần tử không có hai phần tử giống nhau và số  $X$ . Nhiệm vụ của bạn là tìm  $\text{floor}(X)$ . Trong đó,  $K = \text{floor}(X)$  là phần tử lớn nhất trong mảng  $A[]$  nhỏ hơn hoặc bằng  $X$ .

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất đưa vào số  $N$  là số phần tử của mảng  $A[]$  và số  $X$ ; dòng tiếp theo đưa vào  $N$  số của mảng  $A[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^7$ ;  $1 \leq A[i] \leq 10^{18}$ .

**Output:**

- Đưa ra vị trí của  $\text{floor}(X)$  trong mảng  $A[]$  hoặc  $-1$  nếu không tồn tại  $\text{floor}(X)$  của mỗi test theo từng dòng.

**input**

```
3
7 0
1 2 8 10 11 12 19
7 5
1 2 8 10 11 12 19
7 10
1 2 8 10 11 12 19
```

**output**

```
-1
2
4
```

**DSA04016 - PHẦN TỬ THỨ K**

Cho hai mảng đã được sắp xếp  $A[], B[]$  gồm  $M, N$  phần tử theo thứ tự và số  $K$ . Nhiệm vụ của bạn là tìm phần tử ở vị trí số  $K$  sau khi trộn hai mảng để nhận được một mảng được sắp xếp.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng thứ nhất đưa vào số  $M, N, K$ ; dòng tiếp theo đưa vào  $M$  số của mảng  $A[]$ ; dòng tiếp theo đưa vào  $N$  số của mảng  $B[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, M, N, A[i], B[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N, A[i], B[i] \leq 10^6$ ;  $1 \leq K \leq N+M$ .

**Output:**

- Đưa ra giá trị phần tử thứ  $K$  của mỗi test theo từng dòng.

**input**

```
1
5 4 5
2 3 6 7 9
1 4 8 10
```

**output**

```
6
```

**DSA04017 - PHẦN TỬ KHÁC NHAU**

Cho hai mảng đã được sắp xếp  $A[]$  và  $B[]$  gồm  $N$  và  $N-1$  phần tử. Các phần tử của mảng  $A[]$  chỉ khác mảng  $B[]$  một phần tử duy nhất. Hãy tìm vị trí của phần tử khác nhau giữa  $A[]$  và  $B[]$ .

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng thứ nhất đưa vào số  $N$ ; dòng tiếp theo đưa vào  $N$  số của mảng  $A[]$ ; dòng tiếp theo đưa vào  $N-1$  số của mảng  $B[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i], B[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^7$ ;  $0 \leq A[i] \leq 10^{18}$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
7
2 4 6 8 9 10 12
2 4 6 8 10 12
6
3 5 7 9 11 13
3 5 7 11 13
```

**output**

```
5
4
```

## DSA04018 - ĐẾM SỐ 0

Cho mảng  $A[]$  gồm  $N$  phần tử chỉ bao gồm các số 0 và 1. Các số 0 được đặt trước các số 1. Hãy đếm các số 0 với thời gian  $\log(N)$ .

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất đưa vào số  $N$ ; dòng tiếp theo đưa vào  $N$  số của mảng  $A[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 1000$ ;  $0 \leq A[i] \leq 1$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

### input

```
3
12
0 0 0 0 0 0 0 0 1 1 1
5
0 0 0 0 0
6
1 1 1 1 1 1
```

### output

```
9
5
0
```

## DSA04019 - CẶP 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.

### input

```
2
6
2 3
12 30
```

```
40 50
5 1
12 10
3 4
3
0 0
3 0
4 0
```

#### output

```
1.414214
1.000000
```

## DSA04020 - TÌM KIẾM NHỊ PHÂN

Cho dãy số  $A[]$  gồm có  $N$  phần tử đã được sắp xếp tăng dần và số  $K$ .

Nhiệm vụ của bạn là kiểm tra xem số  $K$  có xuất hiện trong dãy số hay không. Nếu có hãy in ra vị trí trong dãy  $A[]$ , nếu không in ra "NO".

#### Input:

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 10$ ).
- Mỗi test bắt đầu bằng số nguyên  $N$  và  $K$  ( $N \leq 100\,000$ ,  $0 \leq K \leq 10^6$ ).
- Dòng tiếp theo gồm  $N$  số nguyên  $A[i]$  ( $0 \leq A[i] \leq 10^6$ ), các phần tử là riêng biệt.

#### Output:

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

#### input

```
2
5 3
1 2 3 4 5
6 5
0 1 2 3 9 10
```

#### output

```
3
NO
```

## DSA04021 - DÃY XÂU NHỊ PHÂN

Người ta tạo ra một dãy xâu ký tự nhị phân  $X[]$  trong đó:

$X[1] = "0"$

$X[2] = "1"$

$X[i] = X[i-2] + X[i-1]$  với  $i > 2$

Với phép cộng (+) là phép nối hai xâu với nhau.

Cho hai số tự nhiên  $N$  và  $K$  ( $1 < N < 93$ ;  $K$  đảm bảo trong phạm vi của xâu  $X[N]$ ).

Hãy xác định ký tự thứ K trong chuỗi X[N] là ký tự '0' hay ký tự '1'.

**Input:** Dòng 1 ghi số bộ test. Mỗi bộ test ghi trên một dòng 2 số nguyên N và K.

**Output:** Ghi ra màn hình kết quả tương ứng với từng bộ test.

**input**

2  
3 1  
4 3

**output**

0  
1

## DSA04022 - KÝ TỰ THỨ K TRONG XÂU

Xâu ký tự S được tạo ra bằng cách bổ sung dần các ký tự chữ cái Tiếng Anh in hoa như sau.

- Bước 1: Chỉ có chữ cái A
- Bước 2: Thêm chữ cái B vào giữa 2 chữ A => S = "ABA"
- Bước 3: Thêm chữ cái C vào giữa 2 chuỗi đã có ở bước 2: S = "ABACABA"

Cứ như vậy cho đến bước thứ N ( $0 < N < 26$ )

Hãy xác định ký tự thứ K trong bước biến đổi thứ N là chữ cái gì?

**Input:**

- Dòng đầu tiên là số lượng bộ test T ( $T \leq 20$ ).
- Mỗi test gồm số nguyên dương N và K ( $1 \leq N \leq 25, 1 \leq K \leq 2N - 1$ ).

**Output:**

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

**input**

2  
3 2  
4 8

**output**

B  
D

## DSA04023 - KHOẢNG CÁCH NHỎ HƠN K

Cho mảng A[] gồm n số nguyên dương và số K. Hãy đếm số các cặp phần tử có hiệu nhỏ hơn K. Ví dụ A[] = {1, 10, 4, 2}, K=3 ta nhận được kết quả là 2 tương ứng với các cặp (1, 2), (4, 2).

**Input:**



- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Mỗi bộ test gồm hai dòng: dòng đầu tiên ghi số  $n$  và số  $K$ ; dòng tiếp theo là  $n$  số  $A[i]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, n, k, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq n \leq 10^4$ ;  $1 \leq k \leq 10^3$ ;  $1 \leq A[i] \leq 10^5$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
4 3
1 10 4 2
3 5
2 3 4
```

**output**

```
2
3
```

## DSA04024 - LŨY THỪA MA TRẬN 2

Cho ma trận vuông  $A$  kích thước  $N \times N$ . Nhiệm vụ của bạn là hãy tính ma trận  $X = AK$  với  $K$  là số nguyên cho trước. Sau đó, tính tổng các phần tử của cột cuối cùng. 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  $X$ .

**input**

```
2
2 5
1 1
1 0
3 1000000000
1 2 3
4 5 6
7 8 9
```

**output**

```
8
581039956
```

## DSA04025 - DÃY SỐ VÔ HẠN

Dãy số nguyên dương vô hạn  $A[]$  được định nghĩa một cách đệ quy như sau:

$$A[0] = 0$$

$$A[1] = 1;$$

$$A[i] = A[i-1] + A[i-2] \text{ với mọi } n \geq 2.$$

Cho số nguyên dương  $N$  không quá  $10^9$ . Hãy xác định số thứ  $N$  trong dãy  $A[]$ . Nếu kết quả quá lớn, hãy in ra 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.

**input**

2  
2  
6

**output**

1  
8

## DSA04030 - SỐ TRIBONACCI

Dãy số Tribonacci được xây dựng như sau:

$$T(i) = i \text{ với } i \leq 3.$$

$$T(i) = T(i-1) + T(i-2) + T(i-3) \text{ với } i \geq 4.$$

Nhiệm vụ của bạn là hãy tính tổng  $N$  phần tử đầu tiên của dãy số này.

$$F(N) = T(1) + T(2) + \dots + T(N)$$

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 100$ ).
- Mỗi test gồm một số nguyên dương  $N$  ( $N \leq 10^9$ ).

**Output:**

- Với mỗi test, in ra đáp án tìm được theo modulo  $10^{15}+7$ .

**input**

5

1  
2  
3  
4  
5

**output**

1  
3  
6  
12  
23

## DSA04035 - TÍNH LŨY THỪA

Cho hai số nguyên không âm  $a$  và  $b$ . Hãy tính  $a^b$ .

Nếu kết quả quá lớn hãy chia dư cho  $10^9 + 7$ .

**Input**

Gồm không quá 20 bộ test, mỗi test ghi trên một dòng hai số  $a, b$ ;  $a$  không quá 9 chữ số,  $b$  không quá 18 chữ số.

Input kết thúc khi  $a = b = 0$

**Output**

Với mỗi test ghi ra kết quả tính được trên một dòng.

**input**

2 3  
2 4  
3 2  
0 0

**output**

8  
16  
9

## Dynamic Programming

### DSA05001 - XÂU CON CHUNG DÀI NHẤT

Cho 2 chuỗi  $S1$  và  $S2$ . Hãy tìm chuỗi con chung dài nhất của 2 chuỗi này (*các phần tử không nhất thiết phải liên tiếp nhau*).

**Input:** Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ). Mỗi test gồm hai dòng, mô tả chuỗi  $S1$  và  $S2$ , mỗi chuỗi có độ dài không quá 1000 và chỉ gồm các chữ cái in hoa.

**Output:** Với mỗi test, in ra độ dài dãy con chung dài nhất trên một dòng.

**input**

2  
AGGTAB  
GXTXAYB  
AA  
BB  
**output**

4  
0

## DSA05003 - DÃY CON CHUNG DÀI NHẤT CỦA BA XÂU

Cho ba xâu ký tự X, Y, Z. Nhiệm vụ của bạn là tìm độ dài dãy con chung dài nhất có mặt trong cả ba xâu.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào độ dài xâu X, Y, X; dòng tiếp theo đưa vào ba xâu X, Y, Z.
- T, X, Y, Z thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{size}(X), \text{size}(Y), \text{size}(Z) \leq 100$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

### input

2  
5 8 13  
geeks geeksfor geeksforgeeks  
7 6 5  
abcd1e2 bc12ea bd1ea

### output

5  
3

## DSA05004 - DÃY CON TĂNG DÀI NHẤT

Cho một dãy số nguyên gồm N phần tử  $A[1], A[2], \dots, A[N]$ .

Biết rằng dãy con tăng là 1 dãy  $A[i_1], \dots, A[i_k]$  thỏa mãn  $i_1 < i_2 < \dots < i_k$  và  $A[i_1] < A[i_2] < \dots < A[i_k]$ .

Hãy cho biết dãy con tăng dài nhất của dãy này có bao nhiêu phần tử?

**Input:** Dòng 1 gồm 1 số nguyên là số N ( $1 \leq N \leq 1000$ ). Dòng thứ 2 ghi N số nguyên  $A[1], A[2], \dots, A[N]$  ( $1 \leq A[i] \leq 1000$ ).

**Output:** Ghi ra độ dài của dãy con tăng dài nhất.

### input

6

1 2 5 4 6 2

**output**

4

## DSA05005 - SỐ BƯỚC ÍT NHẤT

Cho mảng  $A[]$  gồm  $N$  số nguyên. Nhiệm vụ của bạn là sắp xếp lại mảng số với số lượng bước là ít nhất. Tại mỗi bước, bạn chỉ được phép chèn phần tử bất kỳ của mảng vào vị trí bất kỳ trong mảng. Ví dụ  $A[] = \{2, 3, 5, 1, 4, 7, 6\}$  sẽ cho ta số phép chèn ít nhất là 3 bằng cách lấy số 1 chèn trước số 2, lấy số 4 chèn trước số 5, lấy số 6 chèn trước số 7 ta nhận được mảng được sắp.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất là một số  $N$ ; dòng tiếp theo đưa vào  $N$  số của mảng  $A[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 1000$ ;  $1 \leq A[i] \leq 1000$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

1

7

2 3 5 1 4 7 6

**output**

3

## DSA05006 - TỔNG LỚN NHẤT CỦA DÃY CON TĂNG DẦN

Cho dãy số  $A[]$  gồm  $N$  số. Nhiệm vụ của bạn là tìm tổng lớn nhất của dãy con được sắp theo thứ tự tăng dần của dãy  $A[]$ . Ví dụ với dãy  $A[] = \{1, 101, 2, 3, 100, 4, 5\}$  ta có kết quả là  $106 = 1 + 2 + 3 + 100$ . Với dãy  $A[] = \{10, 7, 5\}$  ta có kết quả là 10. Với dãy  $A[] = \{1, 2, 3, 5\}$  ta có kết quả là 11.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào  $N$  là số phần tử của dãy  $A[]$ ; dòng tiếp theo đưa vào  $N$  số  $A[i]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $0 \leq A[i] \leq 10^3$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
3
7
1 101 2 3 100 4 5
3
10 7 5
4
1 2 3 5
```

**output**

```
106
10
11
```

## DSA05007 - TỔNG LỚN NHẤT CỦA DÃY CON KHÔNG KÈ NHAU -- DSAKT110 - NHÀ KHÔNG KÈ NHAU

Có  $N$  ngôi nhà trên một dãy phố, mỗi ngôi nhà chứa đựng một số lượng tài sản khác nhau. Một tên trộm muốn ắp cắp được nhiều nhất tài sản của dãy phố nhưng không muốn lấy tài sản của hai nhà kề nhau. Hãy cho biết, bằng cách đó tên trộm có thể đánh cắp được nhiều nhất bao nhiêu tài sản.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất là một số  $N$  là số lượng ngôi nhà; dòng tiếp theo đưa vào  $N$  số là tài sản tương ứng trong mỗi ngôi nhà; các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^6$ ;  $1 \leq A[i] \leq 10^7$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
6
5 5 10 100 10 5
4
3 2 7 10
```

**output**

```
110
13
```

## DSA05008 - DÃY CON CÓ TỔNG BẰNG S

Cho  $N$  số nguyên dương tạo thành dãy  $A = \{A_1, A_2, \dots, A_N\}$ . Tìm ra một dãy con của dãy  $A$  (không nhất thiết là các phần tử liên tiếp trong dãy) có tổng bằng  $S$  cho trước.

**Input:** Dòng đầu ghi số bộ test  $T$  ( $T < 10$ ). Mỗi bộ test có hai dòng, dòng đầu tiên ghi hai số nguyên dương  $N$  và  $S$  ( $0 < N \leq 200$ ) và  $S$  ( $0 < S \leq 40000$ ). Dòng tiếp theo lần lượt ghi  $N$  số hạng của dãy  $A$  là các số  $A_1, A_2, \dots, A_N$  ( $0 < A_i \leq 200$ ).

**Output:** Với mỗi bộ test, nếu bài toán vô nghiệm thì in ra "NO", ngược lại in ra "YES"

**input**

```
2
5 6
1 2 4 3 5
10 15
2 2 2 2 2 2 2 2 2 2
```

**output**

```
YES
NO
```

## DSA05012 - TỔ HỢP $C(n, k)$

Cho 2 số nguyên  $n, k$ . Bạn hãy tính  $C(n, k)$  modulo  $10^9+7$ .

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test gồm 2 số nguyên  $n, k$  ( $1 \leq k \leq n \leq 1000$ ).

**Output:**

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

**input**

```
2
5 2
10 3
```

**output**

```
10
120
```

## DSA05013 - BẬC THANG -- DSA05034 - TỔNG SỐ CÁCH DI CHUYỂN

Một chiếc cầu thang có  $N$  bậc. Mỗi bước, bạn được phép bước lên trên tối đa  $K$  bước. Hỏi có tất cả bao nhiêu cách bước để đi hết cầu thang? (Tổng số bước đúng bằng  $N$ ).

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 100$ ).
- Mỗi test gồm hai số nguyên dương  $N$  và  $K$  ( $1 \leq N \leq 100000, 1 \leq K \leq 100$ ).

**Output:**

- Với mỗi test, in ra đáp án tìm được trên một dòng theo modulo  $10^9+7$ .

**input**

```
2
```

2 2

4 2

**output**

2

5

## DSA05017 - DÃY SỐ BI-TONIC

Một dãy số được gọi là Bi-tonic nếu nó được chia thành hai dãy đầu tăng dần và dãy tiếp theo giảm dần. Nhiệm vụ của bạn là tìm tổng lớn nhất dãy con Bi-tonic của dãy số  $A[]$ . Ví dụ với dãy  $A[] = \{1, 15, 51, 45, 33, 100, 12, 18, 9\}$  ta có kết quả là 194 tương ứng với dãy Bi-tonic  $\{1, 15, 51, 100, 18, 9\}$ .

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào  $N$  là số phần tử của dãy  $A[]$ ; dòng tiếp theo đưa vào  $N$  số  $A[i]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 100$ ;  $0 \leq A[i] \leq 100$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

2

6

80 60 30 40 20 10

9

1 15 51 45 33 100 12 18 9

**output**

210

194

## DSA05018 - XÂU CON ĐỐI XỨNG DÀI NHẤT

Cho chuỗi  $S$  chỉ bao gồm các ký tự viết thường và dài không quá 1000 ký tự.

Hãy tìm chuỗi con đối xứng dài nhất của  $S$ .

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 10$ ).
- Mỗi test gồm một chuỗi  $S$  có độ dài không vượt quá 1000, chỉ gồm các ký tự thường.

**Output:**

- Với mỗi test, in ra đáp án tìm được.



**input**

```
2
abcbadd
aaaaa
```

**output**

```
5
5
```

**DSA05019 - HÌNH VUÔNG LỚN NHẤT**

Cho một bảng số  $N$  hàng,  $M$  cột chỉ gồm 0 và 1. Bạn hãy tìm hình vuông có kích thước lớn nhất, sao cho các số trong hình vuông toàn là số 1.

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 10$ ).
- Mỗi test bắt đầu bởi 2 số nguyên  $N, M$  ( $1 \leq N, M \leq 500$ ).
- $N$  dòng tiếp theo, mỗi dòng gồm  $M$  số mô tả một hàng của bảng.

**Output:**

- Với mỗi test, in ra đáp án là kích thước của hình vuông lớn nhất tìm được trên một dòng.

**input**

```
2
6 5
0 1 1 0 1
1 1 0 1 0
0 1 1 1 0
1 1 1 1 0
1 1 1 1 1
0 0 0 0 0
2 2
0 0
0 0
```

**output**

```
3
0
```

**DSA05020 - ĐƯỜNG ĐI NHỎ NHẤT**

Cho bảng  $A[]$  kích thước  $N \times M$  ( $N$  hàng,  $M$  cột). Bạn được phép đi xuống dưới, đi sang phải và đi xuống ô chéo dưới. Khi đi qua ô  $(i, j)$ , điểm nhận được bằng  $A[i][j]$ .

Hãy tìm đường đi từ ô  $(1, 1)$  tới ô  $(N, M)$  sao cho tổng điểm là nhỏ nhất.

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test gồm số nguyên dương  $N$  và  $M$ .

- N dòng tiếp theo, mỗi dòng gồm M số nguyên  $A[i][j]$  ( $0 \leq A[i] \leq 1000$ ).

**Output:**

- Với mỗi test, in ra độ dài dãy con tăng dài nhất trên một dòng.

**input**

```
1
3 3
1 2 3
4 8 2
1 5 3
```

**output**

```
8
```

## DSA05022 - KÝ TỰ GIỐNG NHAU

Giả sử bạn cần viết N ký tự giống nhau lên màn hình. Bạn chỉ được phép thực hiện ba thao tác dưới đây với chi phí thời gian khác nhau:

- Thao tác insert: chèn một ký tự với thời gian là X.
- Thao tác delete: loại bỏ ký tự cuối cùng với thời gian là Y.
- Thao tác copying: copy và paste tất cả các ký tự đã viết để số ký tự được nhân đôi với thời gian là Z.

Hãy tìm thời gian ít nhất để có thể đưa ra màn hình N ký tự giống nhau. Ví dụ với  $N = 9$ ,  $X = 1$ ,  $Y = 2$ ,  $Z = 1$  ta có kết quả là 5 bằng cách thực hiện: insert, insert, copying, copying, insert.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào N là số các ký tự giống nhau cần viết lên màn hình; dòng tiếp theo đưa vào bộ ba số X, Y, Z tương ứng với thời gian thực hiện ba thao tác; các số được viết cách nhau một vài khoảng trống.
- T, N, X, Y, Z thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 100$ ;  $1 \leq X, Y, Z \leq 100$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
9
1 2 1
10
2 5 4
```

**output**

```
5
```

## DSA05023 - TỔNG CÁC XÂU CON

Cho số nguyên dương  $N$  được biểu diễn như một chuỗi ký tự số. Nhiệm vụ của bạn là tìm tổng của tất cả các số tạo bởi các chuỗi con của  $N$ . Ví dụ  $N = "1234"$  ta có kết quả là  $1670 = 1 + 2 + 3 + 4 + 12 + 23 + 34 + 123 + 234 + 1234$ .

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Mỗi bộ test bắt đầu bởi số  $N$  (không quá 500).
- $T, N$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^{12}$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

2  
1234  
421

#### output

1670  
491

## DSA05025 - CON ẾCH

Một con ếch có thể nhảy 1, 2, 3 bước để có thể lên đến một đỉnh cần đến. Hãy đếm số các cách con ếch có thể nhảy đến đỉnh.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là số  $n$  là số bước con ếch có thể lên được đỉnh.
- $T, n$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq n \leq 50$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

2  
1  
5

#### output

1  
13

## DSA05026 - XEM PHIM

John có một đàn bò. Một ngày đẹp trời, anh ta quyết định mua xe tải với khả năng chở được  $C$  kg ( $1000 \leq C \leq 25000$ ) để đưa những con bò đi xem phim. Cho số con bò là  $N$  ( $20 \leq N \leq 100$ ) và khối lượng  $w[i]$  của từng con (đều nhỏ hơn  $C$ ), hãy cho biết khối lượng bò lớn nhất mà John có thể đưa đi xem phim là bao nhiêu.

### Input:

- Dòng 1: 2 số nguyên  $C$  và  $N$  cách nhau bởi dấu cách
- Dòng 2.. $N+1$ : Ghi lần lượt các số nguyên:  $w[i]$

### Output:

- Một số nguyên là tổng khối lượng bò lớn nhất mà John có thể mang đi xem phim.

### input

```
259 5
81
58
42
33
61
```

### output

```
242
```

## DSA05027 - CÁI TÚI -- DSAKT055 - BÀI TOÁN CÁI TÚI 0-1

Một người có cái túi thể tích  $V$  ( $V < 1000$ ). Anh ta có  $N$  đồ vật cần mang theo ( $N \leq 1000$ ), mỗi đồ vật có thể tích là  $A[i]$  ( $A[i] \leq 100$ ) và giá trị là  $C[i]$  ( $C[i] \leq 100$ ). Hãy xác định tổng giá trị lớn nhất của các đồ vật mà người đó có thể mang theo, sao cho tổng thể tích không vượt quá  $V$ .

### Input:

- Dòng đầu ghi số bộ test  $T$  ( $T < 10$ )
- Mỗi bộ test gồm ba dòng. Dòng đầu ghi 2 số  $N$  và  $V$ . Dòng tiếp theo ghi  $N$  số của mảng  $A$ . Sau đó là một dòng ghi  $N$  số của mảng  $C$ .
- Dữ liệu vào luôn đảm bảo không có đồ vật nào có thể tích lớn hơn  $V$ .

### Output:

- Với mỗi bộ test, ghi trên một dòng giá trị lớn nhất có thể đạt được.

### input

```
1
15 10
5 2 1 3 5 2 5 8 9 6 3 1 4 7 8
1 2 3 5 1 2 5 8 7 4 1 2 3 2 1
```

### output

## DSA05029 - GIẢI MÃ

Một bản tin M đã mã hóa bí mật thành các con số theo ánh xạ như sau: 'A' → 1, 'B' → 2, ..., 'Z' → 26. Hãy cho biết có bao nhiêu cách khác nhau để giải mã bản tin M. Ví dụ với bản mã M="123" nó có thể được giải mã thành ABC (1 2 3), LC (12 3), AW(1 23).

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một xâu ký tự số M.
- T, M thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{length}(M) \leq 40$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

2  
123  
2563

#### output

3  
2

## DSA05031 - TỔNG BÌNH PHƯƠNG

Mọi số nguyên dương N đều có thể phân tích thành tổng các bình phương của các số nhỏ hơn N. Ví dụ số  $100 = 10^2$  hoặc  $100 = 5^2 + 5^2 + 5^2 + 5^2$ . Cho số nguyên dương N. Nhiệm vụ của bạn là tìm số lượng ít nhất các số nhỏ hơn N mà có tổng bình phương bằng N.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi test là một số tự nhiên N được viết trên 1 dòng.
- T, N thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10000$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

3  
100  
6  
25

#### output

1  
3  
1

## DSA05036 - TĂNG - GIẢM

Cho hai dãy số thực  $A[]$  và  $B[]$  đều có  $N$  phần tử, các giá trị là số thực và không quá 100.

Hãy tính độ dài dài nhất của dãy các vị trí (không cần liên tiếp) thỏa mãn cả hai điều kiện:

- Nếu xét các vị trí đó trên dãy  $A[]$  thì dãy con thu được thỏa mãn tính chất tăng dần (giá trị bằng nhau không được tính vào dãy tăng).
- Nếu xét các vị trí đó trên dãy  $B[]$  thì dãy con thu được thỏa mãn tính chất tăng dần (giá trị bằng nhau không được tính vào dãy tăng).

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Mỗi bộ test bắt đầu bởi số  $N$  (không quá 500).
- Tiếp theo là  $N$  dòng, mỗi dòng ghi 2 giá trị  $A[i]$  và  $B[i]$

### Output:

- Với mỗi test, ghi ra độ dài tính được trên một dòng.

### input

```
3
2
1.0 1.0
1.5 0.0
3
1.0 1.0
1.0 1.0
1.0 1.0
6
1.5 9.0
2.0 2.0
2.5 6.0
3.0 5.0
4.0 2.0
10.0 5.5
```

### output

```
2
1
4
```

## DSA05041 - BIẾN ĐỔI VỀ XÂU ĐỐI XỨNG

Cho chuỗi ký tự  $S$ . Nhiệm vụ của bạn là tìm số phép loại bỏ ít nhất các ký tự trong  $S$  để  $S$  trở thành chuỗi đối xứng. Chú ý, phép loại bỏ phải bảo toàn tính trước sau của các ký tự trong  $S$ .

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một xâu ký tự được viết trên một dòng
- T, str thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{length}(S) \leq 100$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
aebcbda
geeksforgeeks
```

**output**

```
2
8
```

## DSA05042 - DÃY CON LIÊN TIẾP TỔNG BẰNG K

Cho dãy số A[] gồm có N phần tử không âm và số K.

Nhiệm vụ của bạn là hãy xác định xem có tìm được 1 dãy con liên tiếp mà tổng các phần tử bằng K hay không?

**Input:**

- Dòng đầu tiên là số lượng bộ test T ( $T \leq 20$ ).
- Mỗi test gồm số nguyên N và K ( $1 \leq N \leq 100\,000$ ,  $0 \leq K \leq 10^{18}$ ).
- Dòng tiếp theo gồm N số nguyên A[i] ( $0 \leq A[i] \leq 10^9$ ).

**Output:**

- Với mỗi test, in ra trên một dòng là đáp án thu được. Nếu có hãy in ra "YES". Nếu không tìm được đáp án, in ra "NO".

**input**

```
3
6 33
1 4 20 3 10 5
7 7
1 4 0 0 3 10 5
2 0
1 4
```

**output**

```
YES
YES
NO
```

## DSAKT053 - SỐ NGUYÊN LỚN

Cho hai số nguyên lớn N và M có không quá 1000 chữ số. Người ta muốn tính xem liệu có thể lấy ra nhiều nhất bao nhiêu chữ số trong N (không cần liên tiếp) và giữ nguyên thứ tự của nó để tạo ra một số X sao cho ta cũng có thể tìm thấy X trong số M theo cách tương tự.

**Input:** Dòng đầu tiên là số lượng bộ test T ( $T \leq 20$ ). Mỗi test gồm hai dòng, dòng thứ nhất ghi số N, dòng thứ 2 ghi số M.

**Output:** Với mỗi test, hãy in ra số chữ số nhiều nhất có thể của X.

**input**

```
2
144615
4976135
44
88
```

**output**

```
4
0
```

## Sort and Search

### CTDL\_005 - XÓA DỮ LIỆU TRONG DSLK ĐƠN

Cho danh sách liên kết đơn lưu giữ các số nguyên được quản lý bởi con trỏ First. Viết chương trình con xóa tất cả các phần tử có giá trị bằng x trong danh sách liên kết đơn; chương trình con trả về số lượng các phần tử đã xóa. Sau khi xóa xong, liệt kê các phần tử còn lại trong danh sách liên kết đơn First.

**input**

```
14
1 1 1 4 5 1 1 1 7 1 8 1 9
1
```

**output**

```
4 5 7 8 9
```

### CTDL\_006 - LỌC DỮ LIỆU TRÙNG TRONG DSLK ĐƠN

Cho danh sách liên kết đơn lưu giữ các số nguyên được quản lý bởi con trỏ First. Viết chương trình con lọc tất cả các phần tử có giá trị trùng nhau trong danh sách liên kết đơn First, chỉ để lại 1 phần tử đại diện cho nhóm trùng. Sau khi lọc xong, liệt kê các phần tử trong danh sách liên kết đơn First.

**input**

```
12
1 1 1 4 5 1 4 7 7 8 1 9
```

**output**

```
1 4 5 7 8 9
```



## DSA06001 - SẮP XẾP XEN KẼ

Cho mảng  $A[]$  gồm  $n$  số nguyên khác nhau. Hãy đưa ra các phần tử của mảng theo khuôn dạng lớn nhất, nhỏ nhất, lớn thứ hai, nhỏ thứ 2, ... Ví dụ với  $A[] = \{9, 7, 12, 8, 6, 5\}$  ta đưa ra : 12, 5, 9, 6, 8, 7.

### Input:

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

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
2
7
7 1 2 3 4 5 6
8
1 6 9 4 3 7 8 2
```

#### output

```
7 1 6 2 5 3 4
9 1 8 2 7 3 6 4
```

## DSA06002 - SẮP XẾP THEO GIÁ TRỊ TUYỆT ĐỐI

Cho mảng  $A[]$  gồm  $n$  phần tử và số  $X$ . Hãy đưa sắp xếp các phần tử của mảng theo trị tuyệt đối của  $|X - A[i]|$ . Ví dụ với  $A[] = \{10, 5, 3, 9, 2\}$  và  $X = 7$  ta đưa ra mảng được sắp xếp theo nguyên tắc kể trên:  $A[] = \{5, 9, 10, 3, 2\}$  vì  $|7-10|=3$ ,  $|7-5|=2$ ,  $|7-3|=4$ ,  $|7-9|=2$ ,  $|7-2|=5$ .

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên là số phần tử của mảng  $n$  và  $X$ ; dòng tiếp theo là  $n$  số  $A[i]$  của mảng  $A[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, n, X$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq n, X, A[i] \leq 10^5$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
2
5 7
10 5 3 9 2
```

5 6  
1 2 3 4 5  
**output**

5 9 10 3 2  
5 4 3 2 1

## DSA06003 - ĐỔI CHỖ ÍT NHẤT

Cho mảng  $A[]$  gồm  $n$  phần tử. Hãy tìm số phép đổi chỗ ít nhất giữa các phần tử của mảng để mảng  $A[]$  được sắp xếp. Ví dụ với  $A[] = \{4, 3, 2, 1\}$  ta cần thực hiện ít nhất 2 phép đổi chỗ:  $\text{Swap}(A[0], A[3]), \text{Swap}(A[1], A[2])$ .

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên là số phần tử của mảng  $n$  và  $X$ ; dòng tiếp theo là  $n$  số  $A[i]$  của mảng  $A[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, n$  thỏa mãn ràng buộc:  $1 \leq T \leq 100; 1 \leq n \leq 10^3$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

### input

2  
4  
4 3 2 1  
5  
1 5 4 3 2  
**output**

2  
2

## DSA06004 - HỢP VÀ GIAO CỦA HAI DÃY SỐ 1

Cho mảng  $A[]$  gồm  $n$  phần tử, mảng  $B[]$  gồm  $m$  phần tử khác nhau. Các phần tử của mảng  $A[]$  và  $B[]$  đã được sắp xếp. Hãy tìm mảng hợp và giao được sắp giữa  $A[]$  và  $B[]$ . Ví dụ với  $A[] = \{1, 3, 4, 5, 7\}$ ,  $B[] = \{2, 3, 5, 6\}$  ta có mảng hợp  $\text{Union} = \{1, 2, 3, 4, 5, 6, 7\}$ , mảng giao  $\text{Intersection} = \{3, 5\}$ . In ra đáp án theo giá trị phần tử từ nhỏ đến lớn.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm ba dòng: dòng đầu tiên đưa vào  $n, m$  là số phần tử của mảng  $A[]$  và  $B[]$ ; dòng tiếp theo là  $n$  số  $A[i]$  của mảng  $A[]$ ; dòng tiếp theo là  $m$  số  $B[i]$  của mảng  $B[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, n, m, A[i], B[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100; 1 \leq n, m, A[i], B[i] \leq 10^5$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
1
5 3
1 2 3 4 5
1 2 3
```

#### output

```
1 2 3 4 5
1 2 3
```

## DSA06005 - HỢP VÀ GIAO CỦA HAI DÃY SỐ 2

Cho mảng  $A[]$  gồm  $n$  phần tử, mảng  $B[]$  gồm  $m$  phần tử khác nhau. Các phần tử của mảng  $A[]$  và  $B[]$  chưa được sắp xếp. Hãy tìm mảng hợp và giao được sắp giữa  $A[]$  và  $B[]$ . Ví dụ với  $A[] = \{7, 1, 5, 2, 3, 6\}$ ,  $B[] = \{3, 8, 6, 20, 7\}$  ta có mảng hợp  $\text{Union} = \{1, 2, 3, 5, 6, 7, 8, 20\}$ , mảng giao  $\text{Intersection} = \{3, 6\}$ .

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm ba dòng: dòng đầu tiên đưa vào  $n, m$  là số phần tử của mảng  $A[]$  và  $B[]$ ; dòng tiếp theo là  $n$  số  $A[i]$  của mảng  $A[]$ ; dòng tiếp theo là  $m$  số  $B[i]$  của mảng  $B[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, n, m, A[i], B[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq n, m, A[i], B[i] \leq 10^5$ .

#### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
1
6 5
7 1 5 2 3 6
3 8 6 20 7
```

#### output

```
1 2 3 5 6 7 8 20
2 6
```

## DSA06006 - SẮP XẾP [0 1 2]

Cho mảng  $A[]$  gồm  $n$  phần tử. Các phần tử của mảng  $A[]$  chỉ bao gồm các số 0, 1, 2. Hãy sắp xếp mảng  $A[]$  theo thứ tự tăng dần. Ví dụ với  $A[] = \{0, 2, 1, 2, 0\}$  ta kết quả  $A[] = \{0, 0, 1, 2, 2\}$ .

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .

- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào n là số phần tử của mảng A[]; dòng tiếp theo là n số A[i] của mảng A [] các số được viết cách nhau một vài khoảng trống.
- T, n, A[i] thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $0 \leq A[i] \leq 2$ ;  $1 \leq n \leq 10^6$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
5
0 2 1 2 0
3
0 1 0
```

**output**

```
0 0 1 2 2
0 1
```

## DSA06007 - SẮP XẾP DÃY CON LIÊN TỤC

Cho mảng A[] gồm n phần tử. Hãy tìm dãy con liên tục của mảng A[R], ..., A[L] sao cho khi sắp xếp lại dãy con ta nhận được một mảng được sắp xếp. Ví dụ với A[] = {10, 12, 20, 30, 25, 40, 32, 31, 35, 50, 60} ta chỉ cần sắp xếp lại dãy con từ A[4], ..., A[9]: {30, 25, 40, 32, 31, 35} để có mảng được sắp.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào n là số phần tử của mảng A[]; dòng tiếp theo là n số A[i] của mảng A [] các số được viết cách nhau một vài khoảng trống. T, n, A[i] thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq n \leq 10^6$ ;  $0 \leq A[i] \leq 10^7$ .
- T, N thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 1010$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
11
10 12 20 30 25 40 32 31 35 50 60
9
0 1 15 25 6 7 30 40 50
```

**output**

```
4 9
3 6
```

## DSA06008 - ĐẾM CẶP

Cho mảng  $X[]$  gồm  $n$  phần tử và mảng  $Y[]$  gồm  $m$  phần tử. Hãy đếm số các cặp  $xy > yx$ , trong đó  $x \in X[]$  và  $y \in Y[]$ . Ví dụ  $X[] = \{2, 1, 6\}$ ,  $Y[] = \{1, 5\}$  ta có kết quả là 3 cặp (2, 1), (2, 5), (6, 1).

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm ba dòng: dòng đầu tiên đưa vào  $n, m$  tương ứng với số phần tử của mảng  $X[]$  và  $Y[]$ ; dòng tiếp theo là  $n$  số  $X[i]$  của mảng  $X[]$ ; dòng cuối cùng là  $m$  số của mảng  $Y[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, n, m, X[i], Y[j]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq n, m \leq 10^5$ ;  $0 \leq X[i], Y[j] \leq 10^3$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
1
3 2
2 1 6
1 5
```

**output**

```
3
```

## DSA06009 - CẶP SỐ TỔNG BẰNG K

Cho mảng  $A[]$  gồm  $n$  phần tử và số  $k$ . Đếm tất cả các cặp phần tử của mảng có tổng bằng  $k$ . Ví dụ  $A[] = \{1, 5, 3, 4, 2\}$ ,  $k = 7$  ta có kết quả là 2 cặp (3, 4), (5, 2).

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào  $n$  là số phần tử của mảng  $A[]$  và  $k$ ; dòng tiếp theo là  $n$  số  $A[i]$  của mảng  $A[]$  các số được viết cách nhau một vài khoảng trống.
- $T, n, k, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq n \leq 100$ ;  $0 \leq k \leq 100$ ,  $0 \leq A[i] \leq 10^3$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
5 9
1 5 4 1 2
3 2
1 1 1
```

**output**

```
1
```

## DSA06010 - SẮP XẾP CHỮ SỐ

Cho mảng  $A[]$  gồm  $n$  phần tử. Nhiệm vụ của bạn là đưa ra mảng đã được sắp xếp bao gồm các chữ số của mỗi phần tử trong  $A[]$ . Ví dụ  $A[] = \{110, 111, 112, 113, 114\}$  ta có kết quả là  $\{0, 1, 2, 3, 4\}$ .

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào  $n$  là số phần tử của mảng  $A[]$ ; dòng tiếp theo là  $n$  số  $A[i]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, n, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq n \leq 10^7$ ;  $0 \leq A[i] \leq 10^{16}$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
2
3
131 11 48
4
111 222 333 446
```

#### output

```
1 3 4 8
1 2 3 4 6
```

## DSA06011 - TỔNG GẦN 0 NHẤT

### TỔNG GẦN 0 NHẤT

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào  $n$  là số phần tử của mảng  $A[]$ ; dòng tiếp theo đưa vào  $n$  số  $A[i]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, n, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $2 \leq N \leq 10^3$ ;  $-10^6 \leq A[i] \leq 10^6$ .

#### Output:

- Đưa ra tổng gần nhất với 0 của cặp phần tử.

#### input

```
2
3
-8 -66 -60
```

6  
-21 -67 -37 -18 4 -65  
**output**

-68  
-14

## DSA06012 - PHẦN TỬ LỚN NHẤT

Cho mảng  $A[]$  gồm  $n$  phần tử, hãy tìm  $k$  phần tử lớn nhất của mảng. Các phần tử được đưa ra theo thứ tự giảm dần.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào  $N$  và  $K$ ; dòng tiếp theo đưa vào  $n$  số  $A[i]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, K, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq K < N \leq 10^3$ ,  $1 \leq A[i] \leq 10^6$ .

### Output:

- Đưa ra  $k$  phần tử lớn nhất trên một dòng.

### input

2  
5 3  
10 7 9 12 6  
6 2  
9 7 12 8 6 5

### output

12 10 9  
12 9

## DSA06013 - SỐ LẦN XUẤT HIỆN

Cho mảng  $A[]$  gồm  $n$  phần tử đã được sắp xếp. Hãy tìm số lần xuất hiện số  $X$  trong mảng. Nếu số lần xuất hiện số  $x$  trong mảng là 0 hãy đưa ra -1.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào  $N$  và  $X$ ; dòng tiếp theo đưa vào  $n$  số  $A[i]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, X, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ,  $1 \leq A[i], X \leq 10^6$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
7 2
1 1 2 2 2 2 3
7 4
1 1 2 2 2 2 3
```

**output**

```
4
-1
```

**DSA06014 - TỔNG CẶP SỐ NGUYÊN TỐ**

Cho số tự nhiên  $N$ . Hãy tìm cặp số nguyên tố đầu tiên có tổng là  $N$ . Nếu không tồn tại cặp số nguyên tố có tổng bằng  $N$ , hãy đưa ra  $-1$ .

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm là một số  $N$  được ghi trên một dòng.
- $T, N$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^6$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
4
8
```

**output**

```
2 2
2 5
```

**DSA06015 - MERGE SORT**

Cho mảng  $A[]$  gồm  $N$  phần tử chưa được sắp xếp. Nhiệm vụ của bạn là sắp xếp các phần tử của mảng  $A[]$  theo thứ tự tăng dần bằng thuật toán Merge Sort.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất đưa vào số  $N$  tương ứng với số phần tử của mảng  $A[]$ ; phần thứ 2 là  $N$  số của mảng  $A[]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N$ ,  $A[i] \leq 10^6$ .

**Output:**



- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
2
5
4 1 3 9 7
10
10 9 8 7 6 5 4 3 2 1
```

#### output

```
1 3 4 7 9
1 2 3 4 5 6 7 8 9 10
```

## DSA06016 - TÍCH LỚN NHẤT - NHỎ NHẤT

Cho mảng  $A[]$  gồm  $n$  phần tử và mảng  $B[]$  gồm  $m$  phần tử. Nhiệm vụ của bạn là tìm tích giữa phần tử lớn nhất của mảng  $A[]$  và phần tử nhỏ nhất của mảng  $B[]$ . Ví dụ  $A[] = \{5, 7, 112, 9, 3, 6, 2\}$ ,  $B[] = \{1, 2, 6, -1, 0, 9\}$  ta có kết quả là  $-9 = 9 * (-1)$ .

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm ba dòng: dòng đầu tiên đưa vào  $n, m$  tương ứng với số phần tử của mảng  $A[]$  và  $B[]$ ; dòng tiếp theo là  $n$  số  $A[i]$ ; dòng cuối cùng là  $m$  số  $B[i]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, n, m, A[i], B[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq n, m \leq 10^6$ ;  $-10^8 \leq A[i] \leq 10^8$ .

#### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
2
6 6
5 7 9 3 6 2
1 2 6 -1 0 9
6 6
1 4 2 3 10 2
4 2 6 5 2 9
```

#### output

```
-9
20
```

## DSA06017 - TRỘN HAI DÃY

Cho mảng  $A[]$  gồm  $n$  phần tử và mảng  $B[]$  gồm  $m$  phần tử. Nhiệm vụ của bạn là hợp nhất hai mảng  $A[]$  và  $B[]$  để được một mảng mới đã được sắp xếp. Ví dụ  $A[] = \{5, 7, 112, 9, 3, 6, 2\}$ ,  $B[] = \{1, 2, 6, -1, 0, 9\}$  ta có kết quả là  $C[] = \{-1, 1, 0, 2, 3, 5, 6, 6, 7, .$

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm ba dòng: dòng đầu tiên đưa vào n, m tương ứng với số phần tử của mảng A[] và B[]; dòng tiếp theo là n số A[i] ; dòng cuối cùng là m số B[i]; các số được viết cách nhau một vài khoảng trống.
- T, n, m, A[i], B[i] thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq n, m \leq 10^6$ ;  $-10^8 \leq A[i] \leq 10^8$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
1
3 3
10 5 15
20 3 2
```

**output**

```
2 3 5 10 15 20
```

## DSA06018 - BỔ SUNG PHẦN TỬ

Cho mảng A[] gồm n số nguyên dương. Gọi L, R là max và min các phần tử của A[]. Nhiệm vụ của bạn là tìm số phần tử cần thiết cần thêm vào mảng để mảng có đầy đủ các số trong khoảng [L, R]. Ví dụ A[] = {5, 7, 9, 3, 6, 2 } ta nhận được kết quả là 2 tương ứng với các số còn thiếu là 4, 8.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào n, tương ứng với số phần tử của mảng A[]; dòng tiếp theo là n số A[i] ; các số được viết cách nhau một vài khoảng trống.
- T, n, A[i] thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq n, A[i] \leq 10^3$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
5
4 5 3 8 6
3
2 1 3
```

**output**

```
1
0
```

## DSA06019 - SẮP XẾP THEO SỐ LẦN XUẤT HIỆN

Cho mảng  $A[]$  gồm  $n$  số nguyên. Nhiệm vụ của bạn là sắp xếp mảng theo số lần xuất hiện các phần tử của mảng. Số xuất hiện nhiều lần nhất đứng trước. Nếu hai phần tử có số lần xuất hiện như nhau, số nhỏ hơn đứng trước. Ví dụ  $A[] = \{5, 5, 4, 6, 4\}$ , ta nhận được kết quả là  $A[] = \{4, 4, 5, 5, 6\}$ .

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào  $n$ , tương ứng với số phần tử của mảng  $A[]$  và số  $k$ ; dòng tiếp theo là  $n$  số  $A[i]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, n, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq n \leq 10^4$ ;  $1 \leq k \leq 10^3$ ;  $1 \leq A[i] \leq 10^5$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
5
5 5 4 6 4
5
9 9 9 2 5
```

**output**

```
4 4 5 5 6
9 9 9 2 5
```

## DSA06020 - TÌM KIẾM

Cho mảng  $A[]$  gồm  $n$  phần tử đã được sắp xếp. Hãy đưa ra 1 nếu  $X$  có mặt trong mảng  $A[]$ , ngược lại đưa ra -1.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào  $n, X$  là số các phần tử của mảng  $A[]$  và số  $X$  cần tìm; dòng tiếp theo đưa vào  $n$  số  $A[i]$  ( $1 \leq i \leq n$ ) các số được viết cách nhau một vài khoảng trống.
- $T, n, A, X$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N, X, A[i] \leq 10^6$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
5 16
2 4 7 9 16
7 98
1 22 37 47 54 88 96
```

### output

1  
-1

## DSA06021 - TÌM KIẾM TRONG DÃY SẮP XẾP VÒNG

Một mảng được sắp được chia thành hai đoạn tăng dần được gọi là mảng sắp xếp vòng. Ví dụ mảng  $A[] = \{5, 6, 7, 8, 9, 10, 1, 2, 3, 4\}$  là mảng sắp xếp vòng. Cho mảng  $A[]$  gồm  $n$  phần tử, hãy tìm vị trí của phần tử  $x$  trong mảng  $A[]$  với thời gian  $\log(n)$ .

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào  $n$  và  $x$ ; dòng tiếp theo đưa vào  $n$  số  $A[i]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, n, A[i], x$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N, x, A[i] \leq 10^7$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

### input

2  
10 3  
5 6 7 8 9 10 1 2 3 4  
10 3  
1 2 3 4 5 6 7 8 9 10

### output

9  
3

## DSA06022 - SỐ NHỎ NHẤT VÀ NHỎ THỨ HAI

Cho mảng  $A[]$  gồm  $n$  phần tử, hãy đưa ra số nhỏ nhất và số nhỏ thứ hai của mảng. Nếu không có số nhỏ thứ hai, hãy đưa ra -1.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào  $n$  là số phần tử của mảng  $A[]$ ; dòng tiếp theo đưa vào  $n$  số  $A[i]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, n, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N, A[i] \leq 10^7$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
10
5 6 7 8 9 10 1 2 3 4
5
1 1 1 1 1
```

**output**

```
1 2
-1
```

## DSA06023 - SẮP XẾP ĐỔI CHỖ TRỰC TIẾP

Hãy thực hiện thuật toán sắp xếp đổi chỗ trực tiếp trên dãy N số nguyên. Ghi ra các bước thực hiện thuật toán.

**Input:** Dòng 1 ghi số N (không quá 100). Dòng 2 ghi N số nguyên dương (không quá 100).

**Output:** Ghi ra màn hình từng bước thực hiện thuật toán. Mỗi bước trên một dòng, các số trong dãy cách nhau đúng một khoảng trống.

**input**

```
4
5 7 3 2
```

**output**

```
Buoc 1: 2 7 5 3
Buoc 2: 2 3 7 5
Buoc 3: 2 3 5 7
```

## DSA06024 - SẮP XẾP CHỌN

Hãy thực hiện thuật toán sắp xếp chọn trên dãy N số nguyên. Ghi ra các bước thực hiện thuật toán.

**Input:** Dòng 1 ghi số N (không quá 100). Dòng 2 ghi N số nguyên dương (không quá 100).

**Output:** Ghi ra màn hình từng bước thực hiện thuật toán. Mỗi bước trên một dòng, các số trong dãy cách nhau đúng một khoảng trống.

**input**

```
4
5 7 3 2
```

**output**

```
Buoc 1: 2 7 3 5
Buoc 2: 2 3 7 5
Buoc 3: 2 3 5 7
```

## DSA06025 - SẮP XẾP CHÈN

Hãy thực hiện thuật toán sắp xếp chọn trên dãy N số nguyên. Ghi ra các bước thực hiện thuật toán.

**Input:** Dòng 1 ghi số N (không quá 100). Dòng 2 ghi N số nguyên dương (không quá 100).

**Output:** Ghi ra màn hình từng bước thực hiện thuật toán. Mỗi bước trên một dòng, các số trong dãy cách nhau đúng một khoảng trống.

**input**

4  
5 7 3 2

**output**

Buoc 0: 5  
Buoc 1: 5 7  
Buoc 2: 3 5 7  
Buoc 3: 2 3 5 7

## DSA06026 - SẮP XẾP NỔI BỌT

Hãy thực hiện thuật toán sắp xếp chọn trên dãy N số nguyên. Ghi ra các bước thực hiện thuật toán.

**Input:** Dòng 1 ghi số N (không quá 100). Dòng 2 ghi N số nguyên dương (không quá 100).

**Output:** Ghi ra màn hình từng bước thực hiện thuật toán. Mỗi bước trên một dòng, các số trong dãy cách nhau đúng một khoảng trống.

**input**

4  
5 7 3 2

**output**

Buoc 1: 3 2 5 7  
Buoc 2: 2 3 5 7

## DSA06027 - SẮP XẾP ĐỔI CHỖ TRỰC TIẾP - LIỆT KÊ NGƯỢC

Cho dãy số A[] có n phần tử là các số nguyên dương. Hãy sắp xếp dãy số theo thuật toán sắp xếp đổi chỗ trực tiếp. Nhưng yêu cầu in ra lần lượt các bước chạy của thuật toán theo thứ tự ngược (bước cuối cùng in ra đầu tiên).

**Input:**

Dòng đầu ghi số bộ test (không quá 10). Mỗi test gồm 2 dòng:

- Dòng đầu ghi số n ( $1 < n < 100$ )
- Dòng thứ 2 ghi n số của dãy A (các số đều nguyên dương và nhỏ hơn 1000)

**Output:**

Ghi ra lần lượt các bước thuật toán theo thứ tự ngược (xem ví dụ mẫu để hiểu rõ hơn).

### input

```
1
6
5 8 3 9 1 4
```

### output

```
Buoc 5: 1 3 4 5 8 9
Buoc 4: 1 3 4 5 9 8
Buoc 3: 1 3 4 9 8 5
Buoc 2: 1 3 8 9 5 4
Buoc 1: 1 8 5 9 3 4
```

## DSA06028 - SẮP XẾP CHỌN - LIỆT KÊ NGƯỢC

Cho dãy số  $A[]$  có  $n$  phần tử là các số nguyên dương. Hãy sắp xếp dãy số theo thuật toán sắp xếp chọn. Nhưng yêu cầu in ra lần lượt các bước chạy của thuật toán theo thứ tự ngược (bước cuối cùng in ra đầu tiên).

### Input:

- Dòng đầu ghi số  $n$  ( $1 < n < 100$ )
- Dòng thứ 2 ghi  $n$  số của dãy  $A$  (các số đều nguyên dương và nhỏ hơn 100000)

### Output:

- Ghi ra lần lượt các bước thuật toán theo thứ tự ngược (xem ví dụ mẫu để hiểu rõ hơn).

### input

```
6
5 8 3 9 1 4
```

### output

```
Buoc 5: 1 3 4 5 8 9
Buoc 4: 1 3 4 5 9 8
Buoc 3: 1 3 4 9 5 8
Buoc 2: 1 3 8 9 5 4
Buoc 1: 1 8 3 9 5 4
```

## DSA06029 - SẮP XẾP CHÈN - LIỆT KÊ NGƯỢC

Cho dãy số  $A[]$  có  $n$  phần tử là các số nguyên dương. Hãy sắp xếp dãy số theo thuật toán sắp xếp chèn. Nhưng yêu cầu in ra lần lượt các bước chạy của thuật toán theo thứ tự ngược (bước cuối cùng in ra đầu tiên).

### Input:

- Dòng đầu ghi số  $n$  ( $1 < n < 100$ )
- Dòng thứ 2 ghi  $n$  số của dãy  $A$  (các số đều nguyên dương và nhỏ hơn 100000)

### Output:

- Ghi ra lần lượt các bước thuật toán theo thứ tự ngược (xem ví dụ mẫu để hiểu rõ hơn).

### input

6  
5 8 3 9 1 4

### output

Buoc 5: 1 3 4 5 8 9  
Buoc 4: 1 3 5 8 9  
Buoc 3: 3 5 8 9  
Buoc 2: 3 5 8  
Buoc 1: 5 8  
Buoc 0: 5

## DSA06030 - SẮP XẾP NỔI BỌT - LIỆT KÊ NGƯỢC

Cho dãy số  $A[]$  có  $n$  phần tử là các số nguyên dương. Hãy sắp xếp dãy số theo thuật toán sắp xếp nổi bọt. Nhưng yêu cầu in ra lần lượt các bước chạy của thuật toán theo thứ tự ngược (bước cuối cùng in ra đầu tiên).

*Chú ý: thuật toán sắp xếp nổi bọt có thể kết thúc khi tất cả dãy đã sắp xếp, không còn cặp số cạnh nhau nào sai thứ tự.*

### Input:

Dòng đầu ghi số bộ test (không quá 10). Mỗi test gồm 2 dòng:

- Dòng đầu ghi số  $n$  ( $1 < n < 100$ )
- Dòng thứ 2 ghi  $n$  số của dãy  $A$  (các số đều nguyên dương và nhỏ hơn 1000)

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

### input

1  
6  
4 1 3 9 8 5

### output

Buoc 2: 1 3 4 5 8 9  
Buoc 1: 1 3 4 8 5 9

## DSA06031 - PHẦN TỬ LỚN NHẤT TRONG DÃY CON

Cho dãy số  $A[]$  gồm có  $N$  phần tử và số nguyên  $K$ .

Với mỗi dãy con liên tiếp có độ dài bằng  $K$  (từ trái sang phải), bạn hãy in ra phần tử lớn nhất trong dãy con này.

### Input:

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 10$ ).
- Mỗi test gồm số nguyên  $N$  và  $K$  ( $1 \leq N \leq 100\,000$ ,  $1 \leq K \leq N$ ).



- Dòng tiếp theo gồm N số nguyên  $A[i]$  ( $0 \leq A[i] \leq 10^9$ ).

**Output:**

- Với mỗi test, in ra trên một dòng N-K+1 số nguyên là đáp án tìm được.

**input**

```
2
9 3
1 2 3 1 4 5 2 3 6
10 4
8 5 10 7 9 4 15 12 90 13
```

**output**

```
3 3 4 5 5 5 6
10 10 10 15 15 90 90
```

## DSA06032 - BỘ BA SỐ CÓ TỔNG NHỎ HƠN K

Cho dãy số  $A[]$  gồm có N phần tử.

Nhiệm vụ của bạn là hãy đếm số bộ ba  $(i, j, k)$  mà  $A[i] + A[j] + A[k] < K$  cho trước.

**Input:**

- Dòng đầu tiên là số lượng bộ test T ( $T \leq 20$ ).
- Mỗi test gồm số nguyên N và K ( $1 \leq N \leq 5000$ ,  $-10^9 \leq K \leq 10^9$ ).
- Dòng tiếp theo gồm N số nguyên  $A[i]$  ( $-10^9 \leq A[i] \leq 10^9$ ).

**Output:**

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

**input**

```
2
4 2
-2 0 1 3
5 12
5 1 3 4 7
```

**output**

```
2
4
```

## DSA06033 - KHOẢNG CÁCH XA NHẤT

Cho dãy số  $A[]$  gồm có N phần tử. Bạn cần tìm 2 vị trí  $i, j$  sao cho  $j-i$  lớn nhất và  $A[j] > A[i]$ .

**Input:**

- Dòng đầu tiên là số lượng bộ test T ( $T \leq 10$ ).

- Mỗi test gồm số nguyên  $N$  ( $1 \leq N \leq 100\,000$ ).
- Dòng tiếp theo gồm  $N$  số nguyên  $A[i]$  ( $-10^6 \leq A[i] \leq 10^6$ ).

**Output:**

- Với mỗi test, in ra trên một dòng là giá trị của hiệu  $j-i$  tìm được.

**input**

```
3
9
34 8 10 3 2 80 30 33 1
6
1 2 3 4 5 6
6
6 5 4 3 2 1
```

**output**

```
6
5
-1
```

## DSA06034 - CẶP SỐ CÓ TỔNG BẰNG K

Cho dãy số  $A[]$  gồm có  $N$  phần tử.

Nhiệm vụ của bạn là đếm xem có bao nhiêu cặp  $(i, j)$  mà  $A[i] + A[j] = K$  cho trước.

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 10$ ).
- Mỗi test gồm số nguyên  $N$  và  $K$  ( $1 \leq N \leq 100\,000$ ,  $0 \leq K \leq 10^9$ ).
- Dòng tiếp theo gồm  $N$  số nguyên  $A[i]$  ( $-10^9 \leq A[i] \leq 10^9$ ).

**Output:**

- Với mỗi test, in ra trên một dòng là đáp án thu được.

**input**

```
4
4 6
1 5 7 -1
5 6
1 5 7 -1 5
4 2
1 1 1 1
13 11
10 12 10 15 -1 7 6 5 4 2 1 1 1
```

**output**

```
2
3
6
9
```

## DSA06035 - DÃY TAM GIÁC DÀI NHẤT

Cho dãy số  $A[]$  gồm có  $N$  phần tử.

Một dãy con liên tiếp được gọi là dãy tam giác nếu như dãy đó tăng dần rồi lại giảm dần, hay tồn tại  $i, j, k$  sao cho  $A[i] \leq A[i+1] \leq \dots \leq A[k] \geq A[k+1] \geq \dots \geq A[j]$ .

Nhiệm vụ của bạn là hãy tìm dãy con liên tiếp là dãy tam giác có độ dài lớn nhất.

Lưu ý: Dãy đơn điệu không giảm hoặc không tăng cũng là dãy tam giác.

Ví dụ  $A[] = \{10, 20, 30, 40\}$  là một dãy tam giác.

### Input:

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 10$ ).
- Mỗi test gồm số nguyên  $N$  ( $1 \leq N \leq 100\,000$ ).
- Dòng tiếp theo gồm  $N$  số nguyên  $A[i]$  ( $0 \leq A[i] \leq 10^6$ ).

### Output:

- Với mỗi test, in ra trên một dòng là độ dài của dãy con tìm được.

### input

```
2
6
12 4 78 90 45 23
8
20 4 1 2 3 4 2 10
```

### output

```
5
5
```

## DSA06036 - BỘ BA SỐ BẰNG K

Cho dãy số  $A[]$  gồm có  $N$  phần tử.

Nhiệm vụ của bạn là hãy kiểm tra tồn tại bộ ba  $(i, j, k)$  mà  $A[i] + A[j] + A[k] = K$  cho trước hay không.

### Input:

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test gồm số nguyên  $N$  và  $K$  ( $1 \leq N \leq 5000$ ,  $0 \leq K \leq 10^9$ ).
- Dòng tiếp theo gồm  $N$  số nguyên  $A[i]$  ( $0 \leq A[i] \leq 10^9$ ).

### Output:

- Với mỗi test, in ra trên một dòng là "YES" nếu tìm được bộ ba thỏa mãn, "NO" trong trường hợp ngược lại.

**input**

```

2
6 22
1 4 15 6 10 8
3 4
1 1 1

```

**output**

```

YES
NO

```

**DSA06037 - SẮP XẾP ĐOẠN CON**

Cho dãy số nguyên  $A[]$  độ dài  $N$ . Xác định số lượng chỉ số  $M$  ( $M < N$ ) thỏa mãn: nếu sắp xếp đoạn con  $(A_1, \dots, A_M)$  và  $(A_{M+1}, \dots, A_N)$  theo thứ tự tăng dần thì được dãy số  $A$  tăng dần.

**Input:**

- Dòng đầu tiên ghi số bộ test  $T$
- Mỗi bộ test bắt đầu bằng một dòng chứa số  $N$  ( $2 \leq N \leq 10^5$ )
- Dòng tiếp theo ghi  $N$  số của dãy  $A$  ( $|A_i| \leq 10^9$ )

**Output:**

Với mỗi bộ test:

- Dòng đầu tiên ghi số  $K$  là số lượng vị trí  $M$  tìm được
- Dòng thứ 2 ghi ra  $K$  số theo thứ tự tăng dần lần lượt là các chỉ số thỏa mãn. Nếu  $K = 0$  thì dòng này bỏ trống.

**input**

```

2
2
2 1
5
2 1 3 5 4

```

**output**

```

0

2
2 3

```

**DSA06038 - CẶP SỐ**

Cho dãy số nguyên  $A$  có  $n$  phần tử.

Hãy đếm xem có bao nhiêu cặp  $(i, j)$  thỏa mãn:

- $i < j$
- $A[i] > A[j]$  và đều là số chẵn

- Tồn tại chỉ số  $k$  với  $i < k < j$  sao cho  $A[k]$  là số lẻ

**Input:**

Dòng đầu tiên ghi số  $n$  ( $1 \leq n \leq 10^5$ ).

**Output:**

Dòng thứ 2 ghi  $n$  số của dãy  $A$ , các giá trị  $A[i]$  không vượt quá  $10^6$ .

**input**

5  
4 3 2 5 1

**output**

1

## DSA06039 - SỐ ĐẦU TIÊN BỊ LẶP

Cho dãy số  $A[]$  gồm có  $N$  phần tử. Nhiệm vụ của bạn là hãy tìm số xuất hiện nhiều hơn 1 lần trong dãy số và số thứ tự là nhỏ nhất.

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 10$ ).
- Mỗi test gồm số nguyên  $N$  ( $1 \leq N \leq 100\,000$ ), số lượng phần tử trong dãy số ban đầu.
- Dòng tiếp theo gồm  $N$  số nguyên  $A[i]$  ( $0 \leq A[i] \leq 10^9$ ).

**Output:**

- Với mỗi test in ra đáp án của bài toán trên một dòng. Nếu không tìm được đáp án, in ra "NO".

**input**

2  
7  
10 5 3 4 3 5 6  
4  
1 2 3 4

**output**

5  
NO

## DSA06041 - SỐ XUẤT HIỆN NHIỀU NHẤT

Cho dãy số  $A[]$  gồm có  $N$  phần tử. Nhiệm vụ của bạn là hãy tìm một số có tần số xuất hiện nhiều nhất, yêu cầu lớn hơn  $N/2$  lần xuất hiện trong dãy số.

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 10$ ).
- Mỗi test gồm số nguyên  $N$  ( $1 \leq N \leq 100\,000$ ), số lượng phần tử trong dãy số ban đầu.
- Dòng tiếp theo gồm  $N$  số nguyên  $A[i]$  ( $1 \leq A[i] \leq 1\,000\,000$ ).

**Output:**

- Với mỗi test in ra đáp án của bài toán trên một dòng. Nếu không tìm được đáp án, in ra "NO".

**input**

```
2
9
3 3 4 2 4 4 2 4 4
8
3 3 4 2 4 4 2 4
```

**output**

```
4
NO
```

## DSA06043 - ĐIỂM CÂN BẰNG

Cho dãy số  $A[]$  gồm có  $N$  phần tử nguyên dương. Phần tử thứ  $i$  được gọi là điểm cân bằng của dãy số nếu như tổng các số bên trái bằng tổng các số bên phải của nó.

Nhiệm vụ của bạn là điểm cân bằng đầu tiên của dãy  $A[]$  cho trước. Nếu không có đáp án, in ra -1.

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 10$ ).
- Mỗi test gồm số nguyên  $N$  ( $1 \leq N \leq 100\,000$ ), số lượng phần tử trong dãy số ban đầu.
- Dòng tiếp theo gồm  $N$  số nguyên  $A[i]$  ( $-1000 \leq A[i] \leq 1000$ ).

**Output:**

- Với mỗi test, in ra trên một dòng vị trí của điểm cân bằng tìm được.

**input**

```
2
7
-7 1 5 2 -4 3 0
5
1 2 3 4 5
```

**output**

```
4
-1
```

## DSA06044 - SẮP XẾP CHẴN LẺ

Cho dãy số  $a[]$  có  $n$  phần tử, đánh số từ 1 đến  $n$ . Hãy sắp xếp các phần tử ở vị trí lẻ theo thứ tự tăng dần, các phần tử ở vị trí chẵn theo thứ tự giảm dần.

**Input:**

- Dòng đầu tiên ghi số  $n$ , không quá  $10^5$
- Dòng thứ 2 ghi  $n$  số của dãy  $a[]$  ( $a \leq a[i] \leq 10^9$ )

**Output:**

- Ghi ra dãy số kết quả trên một dòng

**input**

4  
1 2 3 4

**output**

1 4 3 2  
CODE

## DSA06045 - XÂU TỐT

Xâu đối xứng là chuỗi mà khi ta đảo ngược thứ tự của chuỗi thì nhận lại được chuỗi cũ.

Xâu tốt là chuỗi mà mỗi ký tự của nó thuộc về ít nhất 1 chuỗi đối xứng có độ dài lớn hơn 1.

Ví dụ: AABBA, AABA,.. là các chuỗi tốt.

Giá sử cho chuỗi  $s$  chỉ có 2 ký tự A và B. Hãy đếm số chuỗi con là chuỗi tốt trong  $s$  ( Chuỗi con là hợp các phần tử liên tiếp của chuỗi gốc ).

**Input:**

- Dòng đầu là số ký tự của  $s$  ( Không vượt quá  $10^5$  )
- Dòng thứ 2 là chuỗi  $S$  chỉ gồm các ký tự A và B

**Output:**

- Ghi ra kết quả đếm được

**input**

test 1  
  
7  
BABBAAB  
test 2

6  
BAABBA

**output**

test 1

13  
test 2

8

## DSA06046 - CHÊNH LỆCH NHỎ NHẤT

Cho dãy số  $A[]$  gồm có  $N$  phần tử. Bạn cần tìm chênh lệch nhỏ nhất giữa hai phần tử bất kì trong dãy số đã cho.

### Input:

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 10$ ).
- Mỗi test gồm số nguyên  $N$  ( $1 \leq N \leq 100\,000$ ).
- Dòng tiếp theo gồm  $N$  số nguyên  $A[i]$  ( $0 \leq A[i] \leq 10^9$ ).

### Output:

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

### input

```
3
6
1 5 3 19 18 25
4
30 5 20 9
7
1 19 2 31 38 25 100
```

### output

```
1
4
1
```

## DSA06047 - BỘ BA SỐ PYTAGO

Cho dãy số  $A[]$  gồm có  $N$  phần tử. Một bộ 3 số  $(a, b, c)$  được gọi là bộ ba số Pytago nếu như  $a^2 + b^2 = c^2$ .

Nhiệm vụ của bạn là kiểm tra xem có tồn tại bộ ba số Pytago trong dãy số  $A[]$  hay không?

### Input:

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test gồm số nguyên  $N$  và  $K$  ( $1 \leq N \leq 5000$ ,  $0 \leq K \leq 10^9$ ).
- Dòng tiếp theo gồm  $N$  số nguyên  $A[i]$  ( $1 \leq A[i] \leq 10^9$ ).

### Output:

- Với mỗi test, in ra trên một dòng "YES" nếu tìm được, và "NO" trong trường hợp ngược lại.



**input**

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

**output**

```
YES
NO
```

**DSA06048 - SỐ LẦN QUAY VÒNG**

Ban đầu dãy số  $A[]$  gồm  $N$  phần tử đã được sắp xếp tăng dần. Các phần tử của dãy  $A[]$  có thể giống nhau. Sau đó ta thực hiện quay vòng phải, mỗi lần lấy một ký tự ở cuối dãy đưa lên đầu dãy.

Cho trạng thái dãy số của  $A[]$  sau khi đã thực hiện quay vòng  $K$  lần. Hãy tìm  $K$ .

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất đưa vào số  $M, N, K$ ; dòng tiếp theo đưa vào  $N$  số của mảng  $A[]$  các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^7$ ;  $0 \leq A[i] \leq 10^{18}$ .

**Output:**

- Đưa ra số lần quay vòng  $K$  của mỗi test theo từng dòng.

**input**

```
2
5
5 1 2 3 4
5
1 2 3 4 5
```

**output**

```
1
0
```

**DSA06049 - ĐẾM CẶP**

Cho mảng  $A[]$  gồm  $n$  số nguyên dương và số  $k$ . Nhiệm vụ của bạn là đếm số các cặp phần tử có hiệu nhỏ hơn  $k$ . Ví dụ  $A[] = \{1, 10, 4, 2\}$ ,  $k=3$  ta nhận được kết quả là 2 tương ứng với hiệu các cặp  $(1, 2)$ ,  $(4, 2)$ .

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .

- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào n, tương ứng với số phần tử của mảng A[] và số k; dòng tiếp theo là n số A[i] ; các số được viết cách nhau một vài khoảng trống.
- T, n, k, A[i] thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq n \leq 10^4$ ;  $1 \leq k \leq 10^3$ ;  $1 \leq A[i] \leq 10^5$ .

#### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
32
4 3
1 10 4 2
3 5
2 3 4
```

#### output

```
2
3
```

## Stack

### DSA07001 - NGĂN XẾP 1

Cho một ngăn xếp các số nguyên. Các thao tác gồm 3 lệnh: push, pop và show. Trong đó thao tác push kèm theo một giá trị cần thêm (không quá 1000). Hãy viết chương trình ghi ra kết quả của các lệnh show.

**Input:** Gồm nhiều dòng, mỗi dòng chứa một lệnh push, pop hoặc show. Input đảm bảo số lượng phần tử trong stack khi nhiều nhất cũng không vượt quá 200.

**Output:** 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"

#### input

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

#### output

```
3 5
3 5 7
3
```

### DSA07002 - 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”.

**input**

```
9
PUSH 1
PUSH 2
POP
PRINT
PUSH 3
PRINT
POP
POP
PRINT
```

**output**

```
1
3
NONE
```

## DSA07003 - KIỂM TRA BIỂU THỨC SỐ HỌC

Cho biểu thức số học, hãy cho biết biểu thức số học có dư thừa các cặp ký hiệu ‘(,’) ’ hay không?

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức.

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**Ràng buộc:**

- T, exp thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $2 \leq \text{length}(\text{exp}) \leq 20$ .

**input**

```
3
((a+b))
(a+(b)/c)
(a+b*(c-d))
output
```

```
Yes
Yes
No
```

## DSA07004 - ĐẾM SỐ DẤU NGOẶC ĐỐI CHIỀU

Cho một chuỗi chỉ gồm các ký tự '(' , ')' và có độ dài chẵn. Hãy đếm số lượng dấu ngoặc cần phải đổi chiều ít nhất, sao cho chuỗi 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 chuỗi  $S$  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.

### input

```
4
))((
(((
((((
)())
)())((
output
```

```
2
2
1
3
```

## DSA07005 - BIỂU THỨC TƯƠNG ĐƯƠNG

Cho biểu thức đúng  $P$  chỉ bao gồm các phép toán  $+$ ,  $-$ , các toán hạng cùng với các ký tự '(' , ')'. Hãy bỏ tất cả các ký tự '(' , ')' trong  $P$  để nhận được biểu thức tương đương. Ví dụ với  $P = a - (b + c)$  ta có kết quả  $P = a - b - c$ .

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ ;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức  $P$  được viết trên một dòng.

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

### Ràng buộc:

- T, P thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{length}(P) \leq 10^3$ .

### input

2  
a-(b+c)  
a-(b-c-(d+e))-f

### output

a-b-c  
a-b+c+d+e-f

## DSA07006 - 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  $(2+22)+2$  và  $2+(22+2)$  không được chấp nhận vì không xóa đi các cặp dấu ngoặc tương ứng với nhau

### Input:

- 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.

### Output:

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

### input

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

### output

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

## DSA07007 - SO SÁNH BIỂU THỨC

Cho P1, P2 là hai biểu thức đúng chỉ bao gồm các ký tự mở ngoặc '(' hoặc đóng ngoặc ')' và các toán hạng in thường. Nhiệm vụ của bạn là định xem P1 và P2 có giống nhau hay không.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào P1, dòng tiếp theo đưa vào P2.

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**Ràng buộc:**

- T, P thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{length}(P) \leq 100$ .

**input**

```
2
- (a+b+c)
-a-b-c
a-b-(c-d)
a-b-c-d
```

**output**

```
YES
NO
```

## DSA07008 - BIẾN ĐỔI TRUNG TỐ - HẬU TỐ

Hãy viết chương trình chuyển đổi biểu thức biểu diễn dưới dạng trung tố về dạng hậu tố.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức tiền tố exp.

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**Ràng buộc:**

- T, exp thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $2 \leq \text{length}(\text{exp}) \leq 10$ .
- Các phép toán bao gồm +, -, \*, /, ^. Phép lũy thừa có ưu tiên cao hơn nhân chia và cộng trừ.

**input**

```
2
(A+(B+C))
```

$((A*B)+C)$

**output**

ABC++

AB\*C+

## DSA07009 - BIẾN ĐỔI TIỀN TỔ - TRUNG TỔ

Hãy viết chương trình chuyển đổi biểu thức biểu diễn dưới dạng tiền tố về dạng trung tố.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ ;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức tiền tố exp.

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**Ràng buộc:**

- $T, \text{exp}$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $2 \leq \text{length}(\text{exp}) \leq 10^6$ .

**input**

2

\*+AB-CD

\*-A/BC-/AKL

**output**

$((A+B)*(C-D))$

$((A-(B/C))*(A/K)-L)$

## DSA07010 - BIẾN ĐỔI TIỀN TỔ - HẬU TỔ

Hãy viết chương trình chuyển đổi biểu thức biểu diễn dưới dạng tiền tố về dạng hậu tố.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ ;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức tiền tố exp.

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**Ràng buộc:**

- $T, \text{exp}$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $2 \leq \text{length}(\text{exp}) \leq 10^6$ .

**input**

2

\*+AB-CD

\*-A/BC-/AKL

**output**

AB+CD-\*

ABC/-AK/L-\*

**DSA07011 - BIẾN ĐỔI HẬU TỔ - TIỀN TỔ**

Hãy viết chương trình chuyển đổi biểu thức biểu diễn dưới dạng hậu tố về dạng tiền tố.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ ;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức tiền tố exp.

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**Ràng buộc:**

- $T, \text{exp}$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $2 \leq \text{length}(\text{exp}) \leq 10^6$ .

**input**

2

AB+CD-\*

ABC/-AK/L-\*

**output**

\*+AB-CD

\*-A/BC-/AKL

**DSA07012 - BIẾN ĐỔI HẬU TỔ - TRUNG TỔ**

Hãy viết chương trình chuyển đổi biểu thức biểu diễn dưới dạng hậu tố về dạng trung tố.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ ;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức tiền tố exp.

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**Ràng buộc:**



- T, exp thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $2 \leq \text{length}(\text{exp}) \leq 10^6$ .

#### input

2  
ABC++  
AB\*C+

#### output

(A+(B+C))  
((A\*B)+C)

## DSA07013 - TÍNH GIÁ TRỊ BIỂU THỨC HẬU TỔ

Hãy viết chương trình chuyển tính toán giá trị của biểu thức hậu tố.

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức hậu tố exp. Các số xuất hiện trong biểu thức là các số đơn có 1 chữ số.

#### Output:

- Đưa ra kết quả mỗi test theo từng dòng, chỉ lấy giá trị phần nguyên.

#### Ràng buộc:

- T, exp thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $2 \leq \text{length}(\text{exp}) \leq 20$ .

#### input

2  
231\*+9-  
875\*+9-

#### output

-4  
34

## DSA07014 - TÍNH GIÁ TRỊ BIỂU THỨC TIỀN TỔ

Hãy viết chương trình tính toán giá trị của biểu thức tiền tố.

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức tiền tố exp. Các số xuất hiện trong biểu thức là các số đơn có 1 chữ số.

#### Output:

- Đưa ra kết quả mỗi test theo từng dòng, chỉ lấy giá trị phần nguyên.

### Ràng buộc:

- $T$ , exp thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $2 \leq \text{length}(\text{exp}) \leq 20$ .

### input

2  
-+8/632  
-+7\*45+20

### output

8  
25

## DSA07015 - TÍNH TOÁN GIÁ TRỊ BIỂU THỨC TRUNG TỐ

Cho biểu thức trung tố  $S$  với các toán tử  $+$ ,  $-$ ,  $*$ ,  $/$  và dấu ngoặc  $()$ . Các toán hạng là các số có giá trị không vượt quá 100. Hãy tính giá trị biểu thức  $S$ . Phép chia thực hiện với số nguyên, input đảm bảo số bị chia luôn khác 0, đáp số biểu thức có không quá 10 chữ số.

### Input:

- Dòng đầu tiên là số lượng bộ test ( $T \leq 100$ ).
- Mỗi dòng gồm một chuỗi  $S$ , không quá 100 kí tự. Các toán hạng là các số nguyên không âm.

### Output:

- Với mỗi test, in ra đáp án tìm được.

### input

4  
 $6*3+2-(6-4/2)$   
 $100+99*22$   
 $6*((4*3)+5)$   
 $1-2$

### output

16  
2278  
102  
-1

## DSA07016 - BIỂU THỨC TĂNG GIẢM

Cho dãy ký tự  $S$  chỉ bao gồm các ký tự  $I$  hoặc  $D$ . Ký tự  $I$  được hiểu là tăng (Increasing) ký tự  $D$  được hiểu là giảm (Decreasing). Sử dụng các số từ 1 đến 9, hãy đưa ra số nhỏ nhất được đoán nhận từ  $S$ . Chú ý, các số không được phép lặp lại. Dưới đây là một số ví dụ mẫu:

- $A[] = "I"$  : số tăng nhỏ nhất là 12.
- $A[] = "D"$  : số giảm nhỏ nhất là 21
- $A[] = "DD"$  : số giảm nhỏ nhất là 321

- $A[] = \text{"DDIDDIID"}:$  số thỏa mãn 321654798

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test là một xâu  $S$
- $T, S$  thỏa mãn ràng buộc:  $1 \leq T \leq 100; 1 \leq \text{length}(S) \leq 8; .$

#### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

4  
I  
D  
DD  
DDIDDIID

#### output

12  
21  
321  
321654798

## DSA07017 - PHẦN TỬ BÊN PHẢI NHỎ HƠN

Cho mảng  $A[]$  gồm  $n$  phần tử. Hãy đưa ra các phần tử nhỏ hơn tiếp theo của phần tử lớn hơn đầu tiên phần tử hiện tại. Nếu phần tử hiện tại không có phần tử lớn hơn tiếp theo ta xem là -1. Nếu phần tử không có phần tử nhỏ hơn tiếp theo ta cũng xem là -1. Ví dụ với mảng  $A[] = \{5, 1, 9, 2, 5, 1, 7\}$  ta có kết quả là  $\text{ans} = \{2, 2, -1, 1, -1, -1, -1\}$  vì:

#### Next Greater      Right Smaller

5 -> 9	9 -> 2
1 -> 9	9 -> 2
9 -> -1	-1 -> -1
2 -> 5	5 -> 1
5 -> 7	7 -> -1
1 -> 7	7 -> -1
7 -> -1	7 -> -1

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào  $n$  là số phần tử của mảng  $A[]$ , dòng tiếp theo đưa vào  $n$  số  $A[i]$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**Ràng buộc:**

- $T, n, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq n, A[i] \leq 10^6$ .

**input**

```
2
7
5 1 9 2 5 1 7
8
4 8 2 1 9 5 6 3
```

**output**

```
2 2 1 1 -1 -1 -1
2 5 5 5 -1 3 -1 -1
```

**DSA07018 - TỔNG ĐA THỨC**

Cho hai đa thức có bậc không quá 10000 (chỉ viết ra các phần tử có hệ số khác 0). Hãy sử dụng danh sách liên kết đơn để viết chương trình tính tổng hai đa thức đó.

**Input:**

- Dòng đầu ghi số bộ test. Mỗi bộ test có hai dòng, mỗi dòng ghi một đa thức theo mẫu như trong ví dụ. Số phần tử của đa thức không quá 20.
- Chú ý: Bậc của các hạng tử luôn theo thứ tự giảm dần, trong đa thức chỉ có phép cộng và luôn được viết đầy đủ hệ số + số mũ (kể cả mũ 0).

**Output:**

- Ghi ra một dòng đa thức tổng tính được (theo mẫu như ví dụ)

**input**

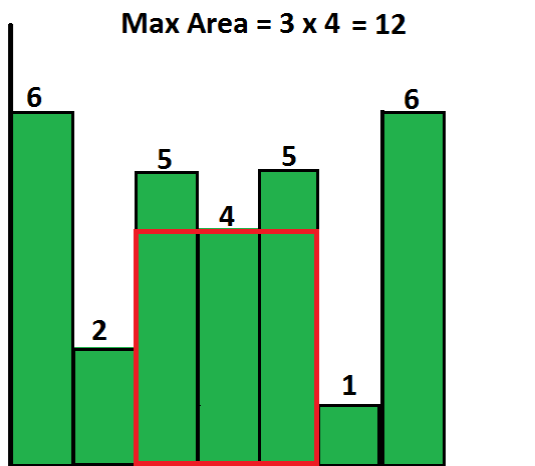
```
1
3*x^8 + 7*x^2 + 4*x^0
11*x^6 + 9*x^2 + 2*x^1 + 3*x^0
```

**output**

```
3*x^8 + 11*x^6 + 16*x^2 + 2*x^1 + 7*x^0
```

**DSA07019 - HÌNH CHỮ NHẬT LỚN NHẤT**

Cho N cột, mỗi cột có chiều cao bằng  $H[i]$ . Bạn hãy tìm hình chữ nhật lớn nhất bị che phủ bởi các cột?



**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test bắt đầu bởi số nguyên  $N$  ( $N \leq 100\,000$ ).
- Dòng tiếp theo gồm  $N$  số nguyên  $H[i]$  ( $1 \leq H[i] \leq 10^9$ ).

**Output:**

- Với mỗi test, in ra diện tích hình chữ nhật lớn nhất tìm được.

**input**

```
2
7
6 2 5 4 5 1 6
3
2 2 2
```

**output**

```
12
6
```

## DSA07021 - DÃY NGOẶC ĐÚNG DÀI 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.

**input**

```
3
(((
))())
()(((()))
```

**output**

```
2
4
6
```

## DSA07022 - PHẦN TỬ CÓ SỐ LẦN XUẤT HIỆN NHIỀU HƠN BÊN PHẢI

Cho mảng  $A[]$  gồm  $n$  phần tử. Nhiệm vụ của bạn là tìm phần tử gần nhất phía bên phải có số lần xuất hiện lớn hơn phần tử hiện tại. Nếu không có phần tử nào có số lần xuất hiện lớn hơn phần tử hiện tại hãy đưa ra -1.

Ví dụ với dãy  $A[] = \{1, 1, 2, 3, 4, 2, 1\}$ , ta nhận được kết quả  $ans[] = \{-1, -1, 1, 2, 2, 1, -1\}$  vì số lần xuất hiện mỗi phần tử trong mảng là  $F = \{3, 3, 2, 1, 1, 2, 3\}$  vì vậy phần tử  $A[0] = 1$  có số lần xuất hiện là 3 và không có phần tử nào xuất hiện nhiều hơn 3 nên  $ans[0] = -1$ , tương tự như vậy với  $A[2]=2$  tồn tại  $A[6]=1$  có số lần xuất hiện là 3 nên  $ans[2] = 1$ .

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ ;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test gồm hai dòng, dòng đầu tiên đưa vào số  $n$  là số các phần tử của mảng  $A[]$ ; dòng tiếp theo đưa vào  $n$  số của mảng  $A[]$ ; các phần tử được viết cách nhau một vài khoảng trống.

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**Ràng buộc:**

- $T, n, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $0 \leq n, A[i] \leq 10^6$ .

**input**

```
1
7
1 1 2 3 4 2 1
```

**output**

```
-1 -1 1 2 2 1 -1
```

## DSA07023 - ĐẢO TỪ

Cho xâu ký tự S. Nhiệm vụ của bạn là đảo ngược các từ trong S. Ví dụ S = "I like this program very much", ta nhận được kết quả là "much very program this like I".

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test là một xâu ký tự S.
- T, S thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{Length}(S) \leq 10^3$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

### input

```
2
I like this program very much
much very program this like I
```

### output

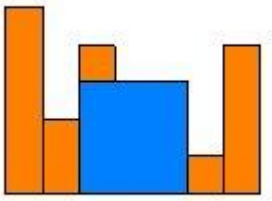
```
much very program this like I
I like this program very much
```

## DSA07024 - GHÉP VÁN GỖ

Trong giấc mơ, Tí thấy mình bị lạc trên hòn đảo với cư dân xa lạ. Không biết làm việc gì, Tí đành phải hái dừa đem bán. Tí muốn làm một tấm biển cho quán hàng của mình. Lục tung khắp nơi, Tí kiếm được N miếng gỗ hình chữ nhật, mỗi miếng có chiều dài bằng  $A[i]$  và chiều rộng bằng 1. Tí đã ghép N miếng gỗ thành một tấm ván lớn, rồi thực hiện cắt các miếng gỗ để thu được tấm biển.

Tí muốn tấm biển quảng cáo của mình thật đặc biệt, nên Tí đã chọn nhất định nó sẽ là hình vuông. Không có eke, thước, dây, ... nên Tí phải dựa vào các miếng gỗ ban đầu để căn nhát cửa. Việc cửa theo chiều dọc rất dễ dàng, nhưng theo chiều ngang, Tí chỉ có thể cửa theo một đường thẳng trùng với một cạnh của miếng gỗ nào đó.

Các bạn hãy giúp Tí tính thử xem tấm biển quảng cáo của Tí có kích thước lớn nhất bằng bao nhiêu (kích thước ở đây là độ dài cạnh hình vuông).



**Input:** Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ). Mỗi test bắt đầu bởi số nguyên  $N$  ( $N \leq 100\,000$ ). Dòng tiếp theo gồm  $N$  số nguyên  $A[i]$  ( $1 \leq A[i] \leq 10^9$ ).

**Output:** Với mỗi test, in ra chiều dài cạnh của tấm biển. Nếu Tí không thể tạo thành tấm biển, in ra 0.

**input**

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

**output**

```
4
2
0
3
```

## DSA07027 - PHẦN TỬ BÊN PHẢI ĐẦU TIÊN LỚN HƠN

Cho dãy số  $A[]$  gồm  $N$  phần tử. Với mỗi  $A[i]$ , bạn cần tìm phần tử bên phải đầu tiên lớn hơn nó. Nếu không tồn tại, in ra -1.

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test bắt đầu bởi số nguyên  $N$  ( $1 \leq N \leq 100\,000$ ).
- Dòng tiếp theo gồm  $N$  số nguyên  $A[i]$  ( $0 \leq A[i] \leq 10^9$ ).

**Output:**

- Với mỗi test, in ra trên một dòng  $N$  số  $R[i]$ , với  $R[i]$  là giá trị phần tử đầu tiên lớn hơn  $A[i]$ .

**input**

```
3
4
4 5 2 25
3
2 2 2
4
4 4 5 5
```

**output**

```
5 25 25 -1
-1 -1 -1
5 5 -1 -1
```

## DSA07028 - NHỊP CHỨNG KHOÁN



Bạn là một nhà đầu tư chứng khoán nổi tiếng. Nhiệm vụ hàng ngày của bạn là tính nhíp tăng giảm của phiên chứng khoán trong N ngày để có thể bắt kịp thị trường. Nhíp chứng khoán của ngày thứ i được định nghĩa là số ngày liên tiếp từ ngày thứ i trở về mà giá chứng khoán bé hơn hoặc bằng với giá chứng khoán của ngày i.

**Input:**

- Dòng đầu ghi số bộ test (không quá 10). Mỗi test có 2 dòng.
- Dòng đầu tiên gồm 1 số nguyên N ( $1 \leq N \leq 105$ ) là số ngày.
- Dòng tiếp theo gồm N số nguyên  $A_1, A_2, \dots, A_N$  ( $1 \leq A_i \leq 10^6$ ) là giá chứng khoán của các ngày.

**Output:**

- In ra N số  $B_1, B_2, \dots, B_N$  trong đó  $B_i$  là nhíp chứng khoán của ngày thứ i.

**input**

```
1
7
100 80 60 70 60 75 85
```

**output**

```
1 1 1 2 1 4 6
```

## DSA07029 - GIẢI MÃ XÂU KÝ TỰ

Cho xâu ký tự mã hóa str. Hãy viết chương trình giải mã xâu ký tự str. Xâu ký tự mã hóa được thực hiện theo số lần lặp các xâu con của str như sau:

Xâu đầu vào: "abbbababbbababbbab "

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một xâu mã hóa str được viết trên một dòng.

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**Ràng buộc:**

- T, str thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq \text{length}(\text{str}) \leq 100$ .

**input**

```
2
1[b]
3[b2[ca]]
```

**output**

b  
bcacabcacabcaca

## DSA07030 - HÌNH CHỮ NHẬT 0-1

Cho một bảng kích thước  $N \times M$ , được chia thành lưới ô vuông đơn vị  $N$  dòng  $M$  cột. Trên các ô của bảng ghi số 0 hoặc 1. Các dòng của bảng được đánh số 1, 2...  $N$  theo thứ tự từ trên xuống dưới và các cột của bảng được đánh số 1, 2...,  $M$  theo thứ tự từ trái qua phải

Yêu cầu: Hãy tìm một hình chữ nhật gồm các ô của bảng thoả mãn các điều kiện sau:

1. Hình chữ nhật đó chỉ gồm các số 1
2. Cạnh hình chữ nhật song song với cạnh bảng
3. Diện tích hình chữ nhật là lớn nhất có thể

### Input:

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test bắt đầu bởi hai số nguyên  $N$  và  $M$  ( $1 \leq M, N \leq 500$ ).
- $N$  dòng tiếp theo, mỗi dòng gồm  $M$  số nguyên 0 hoặc 1.

### Output:

- Với mỗi test, in ra diện tích hình chữ nhật lớn nhất tìm được.

### input

```
1
11 13
0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 1 1 1 0 0 0 0 0 0
0 0 1 1 1 1 1 1 1 0 0 0 0
0 0 1 1 1 1 1 1 1 0 0 0 0
0 1 1 1 1 1 1 1 1 1 0 0 0
1 1 1 1 1 1 1 1 1 1 1 0 0
0 1 1 1 1 1 1 1 1 1 0 0 0
0 0 1 1 1 1 1 1 1 0 0 0 0
0 0 1 1 1 1 1 1 1 0 0 0 0
0 0 0 0 1 1 1 0 0 0 0 1 1
0 0 0 0 0 1 0 0 0 0 0 1 1
```

### output

49

## DSA07041 - BIỂU THỨC ĐÚNG DÀI NHẤT

Cho biểu thức  $P$  chỉ bao gồm các ký tự mở ngoặc '(' hoặc đóng ngoặc ')'. Biểu thức  $P$  có thể viết đúng hoặc không đúng. Nhiệm vụ của bạn là tìm tổng độ dài lớn nhất của các biểu thức con viết đúng trong  $P$  (các biểu thức đúng không nhất thiết phải liên tiếp nhau).

Chú ý: Độ dài của biểu thức đúng ngắn nhất là 2.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T (không quá 100)
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức P được viết trên một dòng (độ dài của P không quá 100).

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
4
((()
()()((
(((()()()))
()()()()
```

**output**

```
2
4
10
6
```

## DSA07045 - BIỂU THỨC ĐÚNG DÀI NHẤT

Trong quá trình gõ một dòng văn bản, chúng ta thường sử dụng phím sang trái, sang phải hoặc xóa lùi (backspace). Cho một dãy ký tự mô tả các thao tác gõ phím, trong đó:

- Ký tự '-' mô tả phím backspace (xóa lùi). Ký tự ở phía trước con trỏ (nếu có) sẽ bị xóa.
- Ký tự '<' mô tả phím di chuyển sang trái. Con trỏ sẽ sang trái 1 ký tự nếu có thể.
- Ký tự '>' mô tả phím di chuyển sang phải. Con trỏ sẽ sang phải 1 ký tự nếu có thể.
- Các ký tự khác là các chữ cái Tiếng Anh (in hoa hoặc in thường). Bàn phím để ở chế độ Insert. Tức là nếu con trỏ không ở cuối dòng thì khi chèn các ký tự sẽ đẩy các ký tự khác sang phải một vị trí.

Hãy thử tính toán và viết ra kết quả tương ứng.

**Input:**

Có một dòng không quá  $10^6$  ký tự mô tả dãy gõ bàn phím.

**Output:**

Ghi ra kết quả.

**input**

```
<<PI<T>>Ta-
```

**output**

```
PTIT
```

## DSA07110 - 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.

**input**

```
2
[()]{ }{[( )]( )}
[( )]
```

**output**

```
YES
NO
```

## Queue

### DSA08001 - 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.

**Input:**

- 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  $10^6$ .

**Output:**

- In ra kết quả của các truy vấn..

#### input

```
1
14
3 1
3 2
3 3
5
6
4
4
4
4
4
3 5
3 6
5
1
```

#### output

```
1
3
5
2
```

## DSA08002 - CẤU TRÚC DỮ LIỆU HÀNG ĐỢI 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”.

#### input

```
9
PUSH 1
PUSH 2
POP
PRINTFRONT
PUSH 3
PRINTFRONT
POP
```

POP  
PRINTFRONT  
**output**

2  
2  
NONE

## DSA08003 - 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.

### input

10  
PUSHBACK 1  
PUSHFRONT 2  
PUSHBACK 3  
PRINTFRONT  
POPFRONT  
PRINTFRONT  
POPFRONT  
PRINTBACK  
POPFRONT  
PRINTBACK

### output

2  
1  
3  
NONE

## DSA08004 - GIÁ TRỊ NHỎ NHẤT CỦA XÂU

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

**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 xâu 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**

```
2
0
ABCC
1
ABCC
```

**output**

```
6
3
```

## DSA08005 - 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 đưa vào 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.

**input**

```
2
2
5
```

**output**

```
1 10
1 10 11 100 101
```

## DSA08006 - 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 đưa vào 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 .

**input**

2  
5  
7

**output**

90  
9009

**DSA08007 - 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.

**input**

3  
10  
100  
200

**output**

2  
4  
7

**DSA08008 - 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  $P = M * N$  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  $P = 5 * 2 = 10$ .  $N = 17$  ta tìm được số BDN của 17 là  $P = 653 * 17 = 11101$ .

**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.

**input**

3  
2  
12  
17

**output**

10  
11100  
11101

## DSA08009 - 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.

**input**

3  
2 5  
3 7  
7 4

**output**

4  
4  
3

**DSA08010 - 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 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.

**input**

3  
17  
50  
100

**output**

4  
5  
5

**DSA08011 - KHOẢNG CÁCH XÂU KÝ TỰ**

Cho tập  $n$  chuỗi ký tự  $S$  và hai chuỗi  $s, t \in S$ . Ta giả thiết các chuỗi 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 chuỗi ký tự bất kỳ ta chỉ được phép dịch chuyển đến chuỗi khác với nó duy nhất 1 ký tự. Ví dụ ta có tập các từ  $S = \{ \text{POON, TOON, PLEE, SAME, POIE, PLEA, PLIE, POIN} \}$ ,  $s = \text{TOON}$ ,  $t = \text{PLEA}$  ta có độ dài đường đi ngắn nhất là 7 tương ứng với các phép dịch chuyển :  $\text{TOON} \rightarrow \text{POON} \rightarrow \text{POIN} \rightarrow \text{POIE} \rightarrow \text{PLIE} \rightarrow \text{PLEE} \rightarrow \text{PLEA}$ .

#### 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  $n$  chuỗi ký tự của  $S$ ; các chuỗi 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.

#### input

```
1
8 TOON PLEA
POON TOON PLEE SAME POIE PLEA PLIE POIN
```

#### output

```
7
```

## DSA08012 - 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.

#### input

```
2
4 13
100 1000
```

#### output

5  
100

## DSA08013 - 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 ghi số bộ test (không quá 10). Mỗi test gồm:
  - 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:

- Với mỗi test, in ra một số nguyên là đáp số của bài toán.

### input

```
1
3
.X.
.X.
...
0 0 0 2
```

### output

```
3
```

## DSA08014 - 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 ghi số bộ test (không quá 10). Mỗi bộ test gồm:
  - 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:**

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

**input**

```
2
3 5
2 1 0 2 1
1 0 1 2 1
1 0 0 2 1
3 5
2 1 0 2 1
0 0 1 2 1
1 0 0 2 1
```

**output**

```
2
-1
```

## DSA08015 - 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'.

**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.

**input**

```
2
3 4 5
S....
.###.
.###.
###.#

#####
```

```
#####  
##.##  
##...
```

```
#####  
#####  
#.###  
#####E
```

```
1 3 3  
S##  
#E#  
###
```

**output**

```
11  
-1
```

## DSA08016 - 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 ghi số bộ test (không quá 10). Mỗi bộ test gồm:
  - Dòng đầu tiên chứa 3 số ở 3 miếng ghép dòng thứ nhất (thứ tự từ trái qua phải).
  - Dòng thứ hai chứa 4 số ở 4 miếng ghép dòng thứ hai (thứ tự từ trái qua phải).
  - Dòng thứ ba chứa 3 số ở 3 miếng ghép dòng thứ ba (thứ tự từ trái qua phải).

### Output:

- Với mỗi bộ test in ra một số nguyên là số phép biến đổi ít nhất để đưa được về trạng thái ban đầu.

### input

```
1  
1 0 2  
8 6 0 3  
7 5 4
```

### output

5

## DSA08017 - SỐ LỘC PHÁT 1

Một số được gọi là lộc phát nếu chỉ có 2 chữ số 6 và 8. Cho số tự nhiên N. Hãy liệt kê các số lộc phát có không quá N chữ số.

### Input:

- Dòng đầu tiên ghi lại số tự nhiên T là số lượng bộ test ( $T < 10$ );
- T dòng kế tiếp mỗi dòng ghi số N ( $1 < N < 15$ ).

### Output:

- In ra đáp án theo thứ tự giảm dần.

### input

2  
2  
3

### output

88 86 68 66 8 6  
888 886 868 866 688 686 668 666 88 86 68 66 8 6

## DSA08018 - SỐ LỘC PHÁT 2

Một số được gọi là lộc phát nếu chỉ có 2 chữ số 6 và 8. Cho số tự nhiên N. Hãy liệt kê các số lộc phát có không quá N chữ số.

### Input:

- Dòng đầu tiên ghi lại số tự nhiên T là số lượng bộ test ( $T < 10$ );
- T dòng kế tiếp mỗi dòng ghi số N ( $1 < N < 15$ ).

### Output:

- Dòng đầu tiên là số lượng số lộc phát tìm được. Dòng thứ hai in đáp án theo thứ tự tăng dần.

### input

2  
2  
3

### output

6  
6 8 66 68 86 88  
14  
6 8 66 68 86 88 666 668 686 688 866 868 886 888

## DSA08019 - SỐ LỘC PHÁT 3

Một số được gọi là lộc phát nếu chỉ có 2 chữ số 6 và 8. Cho số tự nhiên  $N$ . Hãy liệt kê các số lộc phát có không quá  $N$  chữ số.

### Input:

- Dòng đầu tiên ghi lại số tự nhiên  $T$  là số lượng bộ test ( $T < 10$ );
- $T$  dòng kế tiếp mỗi dòng ghi số  $N$  ( $1 < N < 15$ ).

### Output:

- Dòng đầu tiên là số lượng số lộc phát tìm được. Dòng thứ hai in ra đáp án theo thứ tự giảm dần.

### input

```
2
2
3
```

### output

```
6
88 86 68 66 8 6
14
888 886 868 866 688 686 668 666 88 86 68 66 8 6
```

## DSA08020 - 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 <- 8779 <- 3779 <- 3739 <- 3733 <- 1733 <- 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.



**input**

2  
1033 8179  
1033 8779

**output**

6  
5

**DSA08021 - DI CHUYỂN TRONG MA TRẬN**

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.

**input**

1  
3 3  
2 1 2  
1 1 1  
1 1 1

**output**

2

**DSA08022 - BỘI SỐ LỚN NHẤT CỦA 3**

Cho dãy số  $A[i]$  có  $N$  phần tử là các chữ số từ 0 đến 9. Nhiệm vụ của bạn là hãy chọn lấy một tổ hợp các phần tử và sắp xếp chúng sao cho thu được số lớn nhất chia hết cho 3.

Nếu không tìm được số nào, in ra -1.

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $1 \leq T \leq 50$ ).
- Mỗi test bắt đầu bởi số nguyên  $N$  ( $1 \leq N \leq 100\,000$ ).
- Dòng tiếp theo gồm  $N$  số nguyên  $A[i]$  ( $0 \leq A[i] \leq 9$ ).

**Output:**

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

#### input

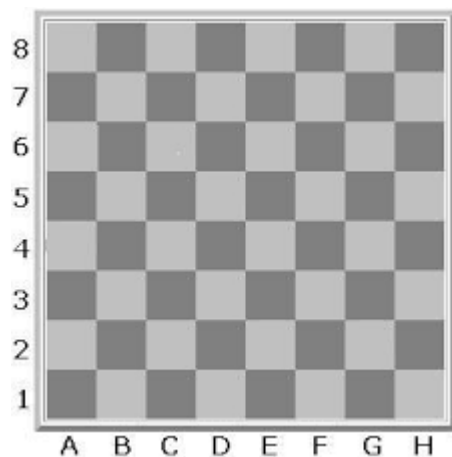
```
3
3
8 1 9
5
8 1 7 6 0
2
5 2
5
2 2 1 1 1
```

#### output

```
981
8760
-1
2211
```

## DSA08025 - QUÂN MÃ

Cho một quân mã trên bàn cờ vua tại vị trí ST. Nhiệm vụ của bạn là hãy tìm số bước di chuyển ít nhất để đưa quân mã tới vị trí EN.



#### Input:

- Dòng đầu tiên là số lượng bộ test T ( $T \leq 20$ ).
- Mỗi test gồm 2 xâu dạng "xy" và "uv", trong đó x, y là kí tự trong "abcdefgh" còn u, v là số thuộc 1, 2, 3, 4, 5, 6, 7, 8.

#### Output:

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

#### input

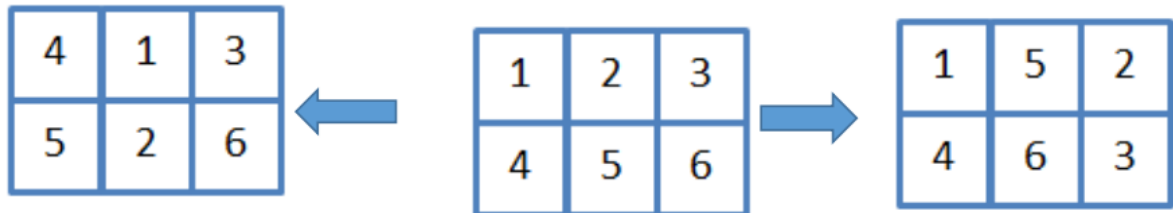
```
8
e2 e4
a1 b2
b2 c3
a1 h8
```

a1 h7  
h8 a1  
b1 c3  
f6 f6  
**output**

2  
4  
2  
6  
5  
6  
1  
0

## DSA08029 - 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 ghi số bộ test (không quá 10). Mỗi bộ test gồm hai dòng:
  - 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:

- Với mỗi test, in ra một số nguyên là đáp số của bài toán.

### input

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

### output

2

## DSA08045 - HÀNG RÀO

Trên cánh đồng có N hàng rào. Mỗi hàng rào được miêu tả bằng một đoạn thẳng đứng hoặc ngang trong mặt phẳng hai chiều. Nếu hai hàng rào gặp nhau, chúng chỉ có thể gặp nhau ở đầu mút của đoạn thẳng. Mỗi hàng rào chỉ giao với hai hàng rào khác tại các kiểu đầu mút của đoạn thẳng.

Có M chú bò đang ăn cỏ trên cánh đồng. Mỗi chú bò được thể hiện bằng một điểm trên mặt phẳng tọa độ hai chiều. Input đảm bảo không có chú bò nào nằm trên đoạn thẳng đại diện cho các hàng rào, và không có hai chú bò có cùng tọa độ. Các chú bò có thể di chuyển tự do trong vùng bao quanh bởi các hàng rào và tạo thành bầy đàn.

Các bạn hãy tìm xem đàn bò nào có số lượng nhiều nhất?

#### Input:

- Dòng đầu ghi số bộ test (không quá 10). Mỗi bộ test như sau:
  - Dòng đầu tiên là 2 số N, M ( $1 \leq N, M \leq 500$ ).
  - N dòng tiếp theo, mỗi dòng gồm 4 số nguyên  $A_x, A_y, B_x, B_y$  mô tả tọa độ của một hàng rào.
  - M dòng tiếp theo, mỗi dòng gồm 2 số nguyên  $C_x, C_y$  mô tả tọa độ của một chú bò.
  - Các tọa độ có giá trị trong khoảng từ 0 tới  $10^6$ .

#### Output:

- In ra số lượng lớn nhất có thể của một đàn bò.

#### input

```
1
7 3
0 0 10 0
10 0 10 5
12 5 10 5
10 5 1 5
12 5 12 7
0 7 12 7
0 7 0 0
3 4
6 6
17 3
```

#### output

```
2
```

## Graph

### DSA09001 - CHUYỂN DANH SÁCH CẠNH SANG DANH SÁCH KÈ

Cho đồ thị vô hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy viết chương trình thực hiện chuyển đổi biểu diễn đồ thị dưới dạng danh sách kề.

#### Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm  $|E| + 1$  dòng: dòng đầu tiên đưa vào hai số  $|V|, |E|$  tương ứng với số đỉnh và số cạnh của đồ thị;  $|E|$  dòng tiếp theo đưa vào các bộ đôi  $u \in V, v \in V$  tương ứng với một cạnh của đồ thị.
- T,  $|V|, |E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

**Output:**

- Đưa ra danh sách kề của các đỉnh tương ứng theo khuôn dạng của ví dụ dưới đây. Các đỉnh trong danh sách in ra theo thứ tự tăng dần.

**input**

```
1
6 9
1 2
1 3
2 3
2 5
3 4
3 5
4 5
4 6
5 6
```

**output**

```
1: 2 3
2: 1 3 5
3: 1 2 4 5
4: 3 5 6
5: 2 3 4 6
6: 4 5
```

## DSA09002 - CHUYỂN TỪ DANH SÁCH KỀ SANG DANH SÁCH CẠNH

Cho đơn đồ thị G vô hướng liên thông được mô tả bởi danh sách kề. Hãy in ra danh sách cạnh tương ứng của G.

**Input:**

- Dòng đầu tiên ghi số N là số đỉnh ( $1 < N < 50$ )
- N dòng tiếp theo mỗi dòng ghi 1 danh sách kề lần lượt theo thứ tự từ đỉnh 1 đến đỉnh N

**Output:**

- Ghi ra lần lượt từng cạnh của đồ thị theo thứ tự tăng dần.

**input**

```
3
2 3
1 3
1 2
```

**output**

1 2  
1 3  
2 3

## DSA09003 - BIỂU DIỄN ĐỒ THỊ CÓ HƯỚNG.

Cho đồ thị có hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy viết chương trình thực hiện chuyển đổi biểu diễn đồ thị dưới dạng danh sách kề.

### Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm  $|E| + 1$  dòng: dòng đầu tiên đưa vào hai số  $|V|, |E|$  tương ứng với số đỉnh và số cạnh của đồ thị;  $|E|$  dòng tiếp theo đưa vào các bộ đôi  $u \in V, v \in V$  tương ứng với một cạnh của đồ thị.
- T,  $|V|, |E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V| - 1)/2$ ;

### Output:

- Đưa ra danh sách kề của các đỉnh tương ứng theo khuôn dạng của ví dụ dưới đây. Các đỉnh trong danh sách in ra theo thứ tự tăng dần.

### input

1  
6 9  
1 2  
2 5  
3 1  
3 2  
3 5  
4 3  
5 4  
5 6  
6 4

### output

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

## DSA09004 - DFS TRÊN ĐỒ THỊ VÔ HƯỚNG

Cho đồ thị vô hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy viết thuật toán duyệt theo chiều sâu bắt đầu tại đỉnh  $u \in V$  ( $\text{DFS}(u) = ?$ )

### Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.

- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm  $|E| + 1$  dòng: dòng đầu tiên đưa vào ba số  $|V|$ ,  $|E|$  tương ứng với số đỉnh và số cạnh của đồ thị, và  $u$  là đỉnh xuất phát;  $|E|$  dòng tiếp theo đưa vào các bộ đôi  $u \in V$ ,  $v \in V$  tương ứng với một cạnh của đồ thị.
- $T$ ,  $|V|$ ,  $|E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

#### Output:

- Đưa ra danh sách các đỉnh được duyệt theo thuật toán DFS( $u$ ) của mỗi test theo khuôn dạng của ví dụ dưới đây.

#### input

```
1
6 9 5
1 2
1 3
2 3
2 4
3 4
3 5
4 5
4 6
5 6
```

#### output

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

## DSA09005 - BFS TRÊN ĐỒ THỊ VÔ HƯỚNG

Cho đồ thị vô hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy viết thuật toán duyệt theo chiều sâu bắt đầu tại đỉnh  $u \in V$  (BFS( $u$ )=?)

#### Input:

- Dòng đầu tiên đưa vào  $T$  là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm  $|E| + 1$  dòng: dòng đầu tiên đưa vào ba số  $|V|$ ,  $|E|$  tương ứng với số đỉnh và số cạnh của đồ thị, và  $u$  là đỉnh xuất phát;  $|E|$  dòng tiếp theo đưa vào các bộ đôi  $u \in V$ ,  $v \in V$  tương ứng với một cạnh của đồ thị.
- $T$ ,  $|V|$ ,  $|E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

#### Output:

- Đưa ra danh sách các đỉnh được duyệt theo thuật toán BFS( $u$ ) của mỗi test theo khuôn dạng của ví dụ dưới đây.

#### input

```
1
6 9 1
1 2 1 3 2 3 2 5 3 4 3 5 4 5 4 6 5 6
```

#### output

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

## DSA09006 - ĐƯỜNG ĐI THEO DFS VỚI ĐỒ THỊ VÔ HƯỚNG

Cho đồ thị vô hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy tìm đường đi từ đỉnh  $s \in V$  đến đỉnh  $t \in V$  trên đồ thị bằng thuật toán DFS.

### Input:

- Dòng đầu tiên đưa vào  $T$  là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm  $|E| + 1$  dòng: dòng đầu tiên đưa vào ba số  $|V|$ ,  $|E|$  tương ứng với số đỉnh và số cạnh của đồ thị, và  $u$  là đỉnh xuất phát;  $|E|$  dòng tiếp theo đưa vào các bộ đôi  $u \in V$ ,  $v \in V$  tương ứng với một cạnh của đồ thị.
- $T$ ,  $|V|$ ,  $|E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

### Output:

- Đưa ra đường đi từ đỉnh  $s$  đến đỉnh  $t$  của mỗi test theo thuật toán DFS của mỗi test theo khuôn dạng của ví dụ dưới đây. Nếu không có đáp án, in ra -1.

### input

```
1
6 9 1 6
1 2 1 3 2 3 2 5 3 4 3 5 4 5 4 6 5 6
```

### output

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

## DSA09007 - ĐƯỜNG ĐI THEO BFS VỚI ĐỒ THỊ VÔ HƯỚNG

Cho đồ thị vô hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy tìm đường đi từ đỉnh  $s \in V$  đến đỉnh  $t \in V$  trên đồ thị bằng thuật toán BFS.

### Input:

- Dòng đầu tiên đưa vào  $T$  là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm  $|E| + 1$  dòng: dòng đầu tiên đưa vào ba số  $|V|$ ,  $|E|$  tương ứng với số đỉnh và số cạnh của đồ thị, và  $u$  là đỉnh xuất phát;  $|E|$  dòng tiếp theo đưa vào các bộ đôi  $u \in V$ ,  $v \in V$  tương ứng với một cạnh của đồ thị.
- $T$ ,  $|V|$ ,  $|E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

### Output:

- Đưa ra đường đi từ đỉnh  $s$  đến đỉnh  $t$  của mỗi test theo thuật toán BFS của mỗi test theo khuôn dạng của ví dụ dưới đây. Nếu không có đáp án, in ra -1.

### input

```
1
6 9 1 6
1 2 1 3 2 3 2 5 3 4 3 5 4 5 4 6 5 6
```

### output



1 2 5 6

## DSA09008 - ĐẾM SỐ THÀNH PHẦN LIÊN THÔNG

Cho đồ thị vô hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy tìm số thành phần liên thông của đồ thị.

### Input:

- Dòng đầu tiên đưa vào  $T$  là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm  $|E| + 1$  dòng: dòng đầu tiên đưa vào ba số  $|V|$ ,  $|E|$  tương ứng với số đỉnh và số cạnh của đồ thị, và  $u$  là đỉnh xuất phát;  $|E|$  dòng tiếp theo đưa vào các bộ đôi  $u \in V, v \in V$  tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

### Output:

- Đưa ra số thành phần liên thông của đồ thị.

### input

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

### output

```
1
```

## DSA09009 - TÌM SỐ THÀNH PHẦN LIÊN THÔNG VỚI BFS

Cho đồ thị vô hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy tìm số thành phần liên thông của đồ thị bằng thuật toán BFS.

### Input:

- Dòng đầu tiên đưa vào  $T$  là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm  $|E| + 1$  dòng: dòng đầu tiên đưa vào ba số  $|V|$ ,  $|E|$  tương ứng với số đỉnh và số cạnh của đồ thị, và  $u$  là đỉnh xuất phát;  $|E|$  dòng tiếp theo đưa vào các bộ đôi  $u \in V, v \in V$  tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

### Output:

- Đưa ra số thành phần liên thông của đồ thị bằng thuật toán BFS.

### input

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

### output

## DSA09010 - KIỂM TRA TÍNH LIÊN THÔNG MẠNH

Cho đồ thị có hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy kiểm tra xem đồ thị có liên thông mạnh hay không?

### Input:

- Dòng đầu tiên đưa vào  $T$  là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm  $|E| + 1$  dòng: dòng đầu tiên đưa vào ba số  $|V|$ ,  $|E|$  tương ứng với số đỉnh và số cạnh của đồ thị, và  $u$  là đỉnh xuất phát;  $|E|$  dòng tiếp theo đưa vào các bộ đôi  $u \in V, v \in V$  tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

### Output:

- Đưa ra "YES", hoặc "NO" theo từng dòng tương ứng với test là liên thông mạnh hoặc không liên thông mạnh.

### input

```
1
6 9
1 2 2 4 3 1 3 2 3 5 4 3 5 4 5 6 6 3
```

### output

```
YES
```

## DSA09011 - SỐ LƯỢNG HÒN ĐẢO

Cho một bản đồ kích thước  $N \times M$  được mô tả bằng ma trận  $A[i][j]$ .  $A[i][j] = 1$  có nghĩa vị trí  $(i, j)$  là nổi trên biển. 2 vị trí  $(i, j)$  và  $(x, y)$  được coi là liên nhau nếu như nó có chung đỉnh hoặc chung cạnh. Một hòn đảo là một tập hợp các điểm  $(i, j)$  mà  $A[i][j] = 1$  và có thể di chuyển giữa hai điểm bất kì trong đó.

Nhiệm vụ của bạn là hãy đếm số lượng đảo xuất hiện trên bản đồ.

### Input:

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test bắt đầu bởi 2 số nguyên  $N$  và  $M$  ( $1 \leq N, M \leq 500$ ).
- $N$  dòng tiếp theo, mỗi dòng gồm  $M$  số nguyên  $A[i][j]$ .

### Output:

- Với mỗi test, in ra số lượng hòn đảo tìm được.

### input

```
1
5 5
```

```
1 1 0 0 0
0 1 0 0 1
1 0 0 1 1
0 0 0 0 0
1 0 1 0 1
```

**output**

5

## DSA09012 - LIỆT KÊ ĐỈNH TRỤ

Cho đồ thị vô hướng liên thông  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy đưa ra tất cả các đỉnh trụ của đồ thị?

**Input:**

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm  $|E| + 1$  dòng: dòng đầu tiên đưa vào ba số  $|V|$ ,  $|E|$  tương ứng với số đỉnh và số cạnh của đồ thị, và u là đỉnh xuất phát;  $|E|$  dòng tiếp theo đưa vào các bộ đôi  $u \in V, v \in V$  tương ứng với một cạnh của đồ thị.
- T,  $|V|$ ,  $|E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

**Output:**

- Đưa ra danh sách các đỉnh trụ của mỗi test theo từng dòng.

**input**

```
1
5 5
1 2 1 3 2 3 2 5 3 4
```

**output**

2 3

## DSA09013 - LIỆT KÊ CẠNH CẦU

Cho đồ thị vô hướng liên thông  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy đưa ra tất cả các cạnh cầu của đồ thị?

**Input:**

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm  $|E| + 1$  dòng: dòng đầu tiên đưa vào ba số  $|V|$ ,  $|E|$  tương ứng với số đỉnh và số cạnh của đồ thị, và u là đỉnh xuất phát;  $|E|$  dòng tiếp theo đưa vào các bộ đôi  $u \in V, v \in V$  tương ứng với một cạnh của đồ thị.
- T,  $|V|$ ,  $|E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

**Output:**

- Đưa ra danh sách các cạnh cầu của mỗi test theo từng dòng. In ra đáp án theo thứ tự từ điển, theo dạng "a b ..." với  $a < b$ .

### input

```
1
5 5
1 2
1 3
2 3
2 5
3 4
```

## DSA09014 - KIỂM TRA CHU TRÌNH TRÊN ĐỒ THỊ VÔ HƯỚNG

Cho đồ thị vô hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy kiểm tra xem đồ thị có tồn tại chu trình hay không?

### Input:

- Dòng đầu tiên đưa vào  $T$  là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm  $|E| + 1$  dòng: dòng đầu tiên đưa vào ba số  $|V|$ ,  $|E|$  tương ứng với số đỉnh và số cạnh của đồ thị, và  $u$  là đỉnh xuất phát;  $|E|$  dòng tiếp theo đưa vào các bộ đôi  $u \in V, v \in V$  tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

### Output:

- Đưa ra YES hoặc "NO" kết quả test theo từng dòng tương ứng với đồ thị tồn tại hoặc không tồn tại chu trình.

### input

```
1
6 9
1 2 1 3 2 3 2 5 3 4 3 5 4 5 4 6 5 6
```

### output

YES

## DSA09015 - KIỂM TRA CHU TRÌNH TRÊN ĐỒ THỊ CÓ HƯỚNG

Cho đồ thị có hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy kiểm tra xem đồ thị có tồn tại chu trình hay không?

### Input:

- Dòng đầu tiên đưa vào  $T$  là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm  $|E| + 1$  dòng: dòng đầu tiên đưa vào ba số  $|V|$ ,  $|E|$  tương ứng với số đỉnh và số cạnh của đồ thị, và  $u$  là đỉnh xuất phát;  $|E|$  dòng tiếp theo đưa vào các bộ đôi  $u \in V, v \in V$  tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

### Output:

- Đưa ra YES hoặc "NO" kết quả test theo từng dòng tương ứng với đồ thị tồn tại hoặc không tồn tại chu trình.

**input**

```
1
6 9
1 2 2 4 3 1 3 2 3 5 4 3 5 4 5 6 6 4
```

**output**

YES

## DSA09016 - KIỂM TRA CHU TRÌNH TRÊN ĐỒ THỊ CÓ HƯỚNG VỚI DFS

Cho đồ thị có hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Sử dụng thuật toán DFS, hãy kiểm tra xem đồ thị có tồn tại chu trình hay không?

**Input:**

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm  $|E| + 1$  dòng: dòng đầu tiên đưa vào ba số  $|V|$ ,  $|E|$  tương ứng với số đỉnh và số cạnh của đồ thị, và u là đỉnh xuất phát;  $|E|$  dòng tiếp theo đưa vào các bộ đôi  $u \in V, v \in V$  tương ứng với một cạnh của đồ thị.
- T,  $|V|$ ,  $|E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

**Output:**

- Đưa ra YES hoặc "NO" kết quả test theo từng dòng tương ứng với đồ thị tồn tại hoặc không tồn tại chu trình.

**input**

```
1
6 9
1 2 2 4 3 1 3 2 3 5 4 3 5 4 5 6 6 4
```

**output**

YES

## DSA09017 - KIỂM TRA ĐỒ THỊ CÓ PHẢI LÀ CÂY HAY KHÔNG

Một đồ thị N đỉnh là một cây, nếu như nó có đúng N-1 cạnh và giữa 2 đỉnh bất kì, chỉ tồn tại duy nhất 1 đường đi giữa chúng.

**Input:**

- Dòng đầu tiên là số lượng bộ test T ( $T \leq 20$ ).
- Mỗi test bắt đầu bởi số nguyên N ( $1 \leq N \leq 1000$ ).
- N-1 dòng tiếp theo, mỗi dòng gồm 2 số nguyên u, v cho biết có cạnh nối giữa đỉnh u và v.

**Output:**

- Với mỗi test, in ra "YES" nếu đồ thị đã cho là một cây, in ra "NO" trong trường hợp ngược lại.

#### input

```
2
4
1 2
1 3
2 4
4
1 2
1 3
2 3
```

#### output

```
YES
NO
```

## DSA09020 - CHUYỂN DANH SÁCH KỀ SANG MA TRẬN KỀ

Cho đơn đồ thị vô hướng có  $n$  đỉnh dưới dạng danh sách kề.

Hãy biểu diễn đồ thị bằng ma trận kề.

**Input:** Dòng đầu tiên chứa số nguyên  $n$  – số đỉnh của đồ thị ( $1 \leq n \leq 1000$ ).  $n$  dòng tiếp theo, dòng thứ  $i$  chứa các số nguyên là các đỉnh kề với đỉnh  $i$ .

**Output:** Ma trận kề của đồ thị.

#### input

```
3
2 3
1 3
1 2
```

#### output

```
0 1 1
1 0 1
1 1 0
```

## DSA09021 - CHUYỂN MA TRẬN KỀ SANG DANH SÁCH KỀ

Ma trận kề  $A$  của một đồ thị vô hướng là một ma trận chỉ có các số 0 hoặc 1 trong đó  $A[i][j] = 1$  có ý nghĩa là đỉnh  $i$  kề với đỉnh  $j$  (chỉ số tính từ 1).

Danh sách kề thì liệt kê các đỉnh kề với đỉnh đó theo thứ tự tăng dần.

Hãy chuyển biểu diễn đồ thị từ dạng ma trận kề sang dạng danh sách kề.

**Input:** Dòng đầu tiên chứa số nguyên  $n$  – số đỉnh của đồ thị ( $1 < n \leq 1000$ ).  $n$  dòng tiếp theo, mỗi dòng có  $n$  số nguyên có giá trị 0 và 1 mô tả ma trận kề của đồ thị.

**Output:** Gồm n dòng, dòng thứ i chứa các số nguyên là đỉnh có nối với đỉnh i và được sắp xếp tăng dần. Dữ liệu đảm bảo mỗi đỉnh có kết nối với ít nhất 1 đỉnh khác.

**input**

```
3
0 1 1
1 0 1
1 1 0
```

**output**

```
2 3
1 3
1 2
```

## DSA09022 - DFS TRÊN ĐỒ THỊ CÓ HƯỚNG

Cho đồ thị có hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy viết thuật toán duyệt theo chiều sâu bắt đầu tại đỉnh  $u \in V$  ( $DFS(u) = ?$ )

**Input:**

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào ba số  $|V|$ ,  $|E|$ ,  $u \in V$  tương ứng với số đỉnh, số cạnh và đỉnh bắt đầu duyệt; Dòng tiếp theo đưa vào các bộ đôi  $u \in V$ ,  $v \in V$  tương ứng với một cạnh của đồ thị.
- T,  $|V|$ ,  $|E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

**Output:**

- Đưa ra danh sách các đỉnh được duyệt theo thuật toán  $DFS(u)$  của mỗi test theo khuôn dạng của ví dụ dưới đây.

**input**

```
1
6 9 5
1 2 2 5 3 1 3 2 3 5 4 3 5 4 5 6 6 3
```

**output**

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

## DSA09024 - BFS TRÊN ĐỒ THỊ CÓ HƯỚNG

Cho đồ thị có hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy viết thuật toán duyệt theo chiều sâu bắt đầu tại đỉnh  $u \in V$  ( $BFS(u) = ?$ )

**Input:**

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào ba số  $|V|$ ,  $|E|$ ,  $u \in V$  tương ứng với số đỉnh, số cạnh và đỉnh bắt đầu duyệt; Dòng tiếp theo đưa vào các bộ đôi  $u \in V$ ,  $v \in V$  tương ứng với một cạnh của đồ thị.

- $T, |V|, |E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

**Output:**

- Đưa ra danh sách các đỉnh được duyệt theo thuật toán BFS(u) của mỗi test theo khuôn dạng của ví dụ dưới đây.

**input**

```
1
6 9 1
1 2 2 5 3 1 3 2 3 5 4 3 5 4 5 6 6 4
```

**output**

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

## DSA09025 - ĐƯỜNG ĐI THEO DFS VỚI ĐỒ THỊ CÓ HƯỚNG

Cho đồ thị có hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy tìm đường đi từ đỉnh  $s \in V$  đến đỉnh  $t \in V$  trên đồ thị bằng thuật toán DFS.

**Input:**

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào ba số  $|V|, |E|, u \in V$  tương ứng với số đỉnh, số cạnh và đỉnh bắt đầu duyệt; Dòng tiếp theo đưa vào các bộ đôi  $u \in V, v \in V$  tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

**Output:**

- Đưa ra đường đi từ đỉnh s đến đỉnh t của mỗi test theo thuật toán DFS của mỗi test theo khuôn dạng của ví dụ dưới đây. Nếu không có đáp án, in ra -1.

**input**

```
1
6 9 1 6
1 2 2 5 3 1 3 2 3 5 4 3 5 4 5 6 6 4
```

**output**

```
1 2 5 6
```

## DSA09026 - ĐƯỜNG ĐI THEO BFS VỚI ĐỒ THỊ CÓ HƯỚNG

Cho đồ thị có hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy tìm đường đi từ đỉnh  $s \in V$  đến đỉnh  $t \in V$  trên đồ thị bằng thuật toán BFS.

**Input:**

- Dòng đầu tiên đưa vào T là số lượng bộ test.



- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào ba số  $|V|$ ,  $|E|$ ,  $u \in V$  tương ứng với số đỉnh, số cạnh và đỉnh bắt đầu duyệt; Dòng tiếp theo đưa vào các bộ đôi  $u \in V$ ,  $v \in V$  tương ứng với một cạnh của đồ thị.
- $T$ ,  $|V|$ ,  $|E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 200$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

#### Output:

- Đưa ra đường đi từ đỉnh  $s$  đến đỉnh  $t$  của mỗi test theo thuật toán BFS của mỗi test theo khuôn dạng của ví dụ dưới đây. Nếu không có đáp án, in ra -1.

#### input

```
1
6 9 1 6
1 2 2 5 3 1 3 2 3 5 4 3 5 4 5 6 6 4
```

#### output

```
1 2 5 6
```

## DSA09027 - KIỂM TRA ĐƯỜNG ĐI

Cho đồ thị vô hướng có  $N$  đỉnh và  $M$  cạnh. Có  $Q$  truy vấn, mỗi truy vấn yêu cầu trả lời câu hỏi giữa 2 đỉnh  $x$  và  $y$  có tồn tại đường đi tới nhau hay không?

#### Input:

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test gồm 2 số nguyên  $N$ ,  $M$  ( $1 \leq N, M \leq 1000$ ).
- $M$  dòng tiếp theo, mỗi dòng gồm 2 số nguyên  $u$ ,  $v$  cho biết có cạnh nối giữa đỉnh  $u$  và  $v$ .
- Dòng tiếp là số lượng truy vấn  $Q$  ( $1 \leq Q \leq 1000$ ).
- $Q$  dòng tiếp theo, mỗi dòng gồm 2 số nguyên  $x$  và  $y$ .

#### Output:

- Với mỗi truy vấn, in ra "YES" nếu có đường đi từ  $x$  tới  $y$ , in ra "NO" nếu ngược lại.

#### input

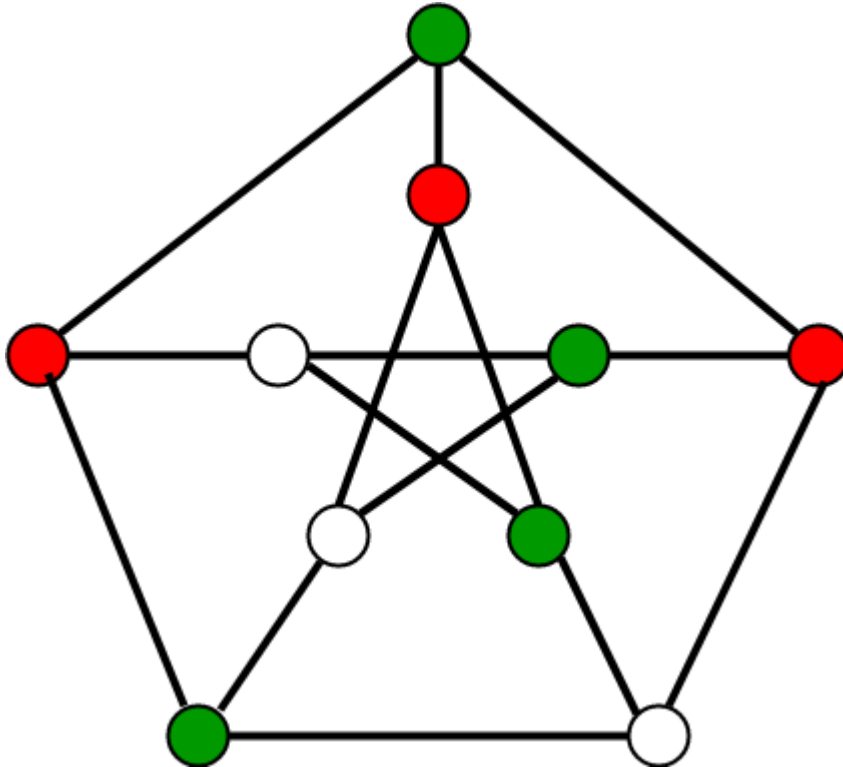
```
1
6 5
1 2
2 3
3 4
1 4
5 6
2
1 5
2 4
```

#### output

```
NO
YES
```

## DSA09028 - TÔ MÀU ĐỒ THỊ

Một trong những bài toán kinh điển của lý thuyết đồ thị là bài toán Tô màu đồ thị. Bài toán được phát biểu như sau: Cho đồ thị vô hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh và số  $M$ . Nhiệm vụ của bạn là kiểm tra xem đồ thị có thể tô màu các đỉnh bằng nhiều nhất  $M$  màu sao cho hai đỉnh kề nhau đều có màu khác nhau hay không?



### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất đưa vào ba số  $V, E, M$  tương ứng với số đỉnh, số cạnh và số màu; phần thứ hai đưa vào các cạnh của đồ thị.
- $T, V, E, M$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq V \leq 10$ ;  $1 \leq E \leq N(N-1)$ ,  $1 \leq V \leq N$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

### input

```
2
4 5 3
1 2
2 3
3 4
4 1
1 3
3 3 2
1 2
2 3
1 3
```

## output

YES  
NO

## DSA09029 - ĐƯỜNG ĐI HAMILTON

Đường đi đơn trên đồ thị có hướng hoặc vô hướng đi qua tất cả các đỉnh của đồ thị mỗi đỉnh đúng một lần được gọi là đường đi Hamilton. Cho đồ thị vô hướng  $G = \langle V, E \rangle$ , hãy kiểm tra xem đồ thị có đường đi Hamilton hay không?

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất đưa vào hai số  $V, E$  tương ứng với số đỉnh, số cạnh của đồ thị; phần thứ hai đưa vào các cạnh của đồ thị.
- $T, V, E$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq V \leq 10$ ;  $1 \leq E \leq 15$ .

### Output:

- Đưa ra 1 hoặc 0 tương ứng với test có hoặc không có đường đi Hamilton theo từng dòng.

## input

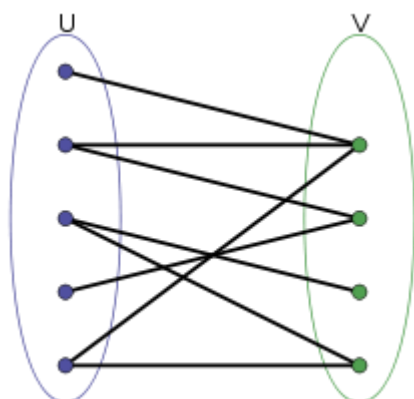
```
2
4 4
1 2 2 3 3 4 2 4
4 3
1 2 2 3 2 4
```

## output

```
1
0
```

## DSA09030 - ĐỒ THỊ HAI PHÍA

Đồ thị hai phía là một đồ thị đặc biệt, trong đó tập các đỉnh có thể được chia thành hai tập không giao nhau thỏa mãn điều kiện không có cạnh nối hai đỉnh bất kỳ thuộc cùng một tập. Cho đồ thị  $N$  đỉnh và  $M$  cạnh, bạn hãy kiểm tra đồ thị đã cho có phải là một đồ thị hai phía hay không?



**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test bắt đầu bởi số nguyên  $N$  và  $M$  ( $1 \leq N, M \leq 1000$ ).
- $M$  dòng tiếp theo, mỗi dòng gồm 2 số nguyên  $u, v$  cho biết có cạnh nối giữa đỉnh  $u$  và  $v$ .

**Output:**

- Với mỗi test, in ra "YES" nếu đồ thị đã cho là một đồ thị hai phía, in ra "NO" trong trường hợp ngược lại.

**input**

```
2
5 4
1 5
1 3
2 3
4 5
3 3
1 2
1 3
2 3
```

**output**

```
YES
NO
```

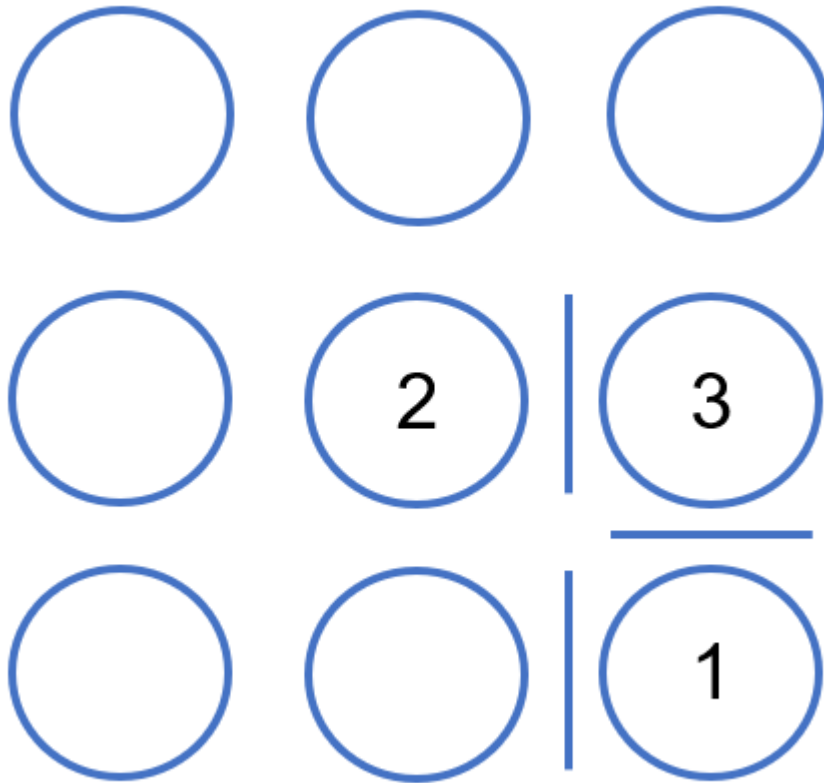
## DSA09031 - CHÚ CỪU XA CÁCH

Trên cánh đồng kích thước  $N \times N$  có  $K$  chú cừu. Người nông dân sợ các chú cừu đi lạc nên đã làm một số rào chắn giữa các khu vực. Các chú cừu chỉ có thể di chuyển lên trên, xuống dưới, sang trái, sang phải khu vực bên cạnh, và không thể vượt qua được hàng rào.

Hai chú cừu  $A$  và  $B$  được gọi là 'xa cách' nếu như chúng không thể di chuyển tới vị trí của nhau. Các bạn hãy xác định xem số cặp chú cừu xa cách bằng nhau nhiều?

**Input:** Dòng đầu tiên gồm 3 số nguyên dương  $N, K$  và  $M$  ( $1 \leq N \leq 100, K \leq 100, M \leq N^2$ ).  $M$  dòng tiếp theo, mỗi dòng gồm 4 số nguyên  $u, v, x, y$  cho biết có rào chắn ở giữa hai khu vực  $(u, v)$  và  $(x, y)$  (2 ô này cạnh nhau).  $K$  dòng cuối, mỗi dòng chứa 2 số nguyên là tọa độ của mỗi chú cừu.

**Output:** In ra số cặp chú cừu bị xa cách tìm được.



**input**

```
3 3 3
2 2 2 3
3 3 3 2
3 3 2 3
3 3
2 2
2 3
```

**output**

2

*Giải thích test: Cặp (3, 1) và (2, 1).*

## DSA09032 - KẾT BẠN

Trường học X có N sinh viên, trong đó có M cặp là bạn bè của nhau. Bạn của bạn cũng là bạn, tức là nếu A là bạn của B, B là bạn của C thì A và C cũng là bạn bè của nhau.

Các bạn hãy xác định xem số lượng sinh viên nhiều nhất trong một nhóm bạn là bao nhiêu?

**Input:**

- Dòng đầu tiên là số lượng bộ test T ( $T \leq 20$ ).
- Mỗi test bắt đầu bởi 2 số nguyên N và M ( $N, M \leq 100\,000$ ).
- M dòng tiếp theo, mỗi dòng gồm 2 số nguyên u, v ( $u \neq v$ ) cho biết sinh viên u là bạn của sinh viên v.

**Output:**

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

**input**

```
2
3 2
1 2
2 3
10 12
1 2
3 1
3 4
5 4
3 5
4 6
5 2
2 1
7 1
1 2
9 10
8 9
```

**output**

```
3
7
```

**DSA09033 - MẠNG XÃ HỘI**

Tý đang xây dựng một mạng xã hội và mời các bạn của mình dùng thử. Bạn của bạn cũng là bạn. Vì vậy, Tý muốn mạng xã hội của mình là hoàn hảo, tức với mọi bộ ba  $X, Y, Z$ , nếu  $X$  kết bạn với  $Y$ ,  $Y$  kết bạn với  $Z$  thì  $X$  và  $Z$  cũng phải là bạn bè của nhau trên mạng xã hội.

Các bạn hãy xác định xem mạng xã hội hiện tại của Tý có là hoàn hảo hay không? Nếu có hãy in ra "YES", "NO" trong trường hợp ngược lại.

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test bắt đầu bởi 2 số nguyên  $N$  và  $M$  ( $N, M \leq 100\,000$ ).
- $M$  dòng tiếp theo, mỗi dòng gồm 2 số nguyên  $u, v$  ( $u \neq v$ ) cho biết sinh viên  $u$  là bạn của sinh viên  $v$ .

**Output:**

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

**input**

```
3
4 3
1 3
3 4
1 4
4 4
```

```

3 1
2 3
3 4
1 2
10 4
4 3
5 10
8 9
1 2

```

**output**

```

YES
NO
YES

```

## DSA09034 - ĐẾM SỐ AO

Sau khi thi trượt môn Cấu trúc dữ liệu và giải thuật, một số sinh viên D19 CNTT - PTIT quyết định bỏ học, đầu tư thuê đất để trồng rau. Mảnh đất thuê là một hình chữ nhật  $N \times M$  ( $1 \leq N \leq 100$ ;  $1 \leq M \leq 100$ ) ô đất hình vuông. Nhưng chỉ sau đó vài ngày, trận lụt khủng khiếp đã diễn ra làm một số ô đất bị ngập. Mảnh đất bỗng biến thành các cái ao. Và sinh viên D19 lại dự định chuyển sang nuôi cá. Các bạn ấy muốn biết mảnh đất được chia thành bao nhiêu cái ao để có thể tính toán nuôi cá cho hợp lý. Hãy giúp các bạn ấy nhé. *Chú ý: Ao là gồm một số ô đất bị ngập có chung đỉnh. Để nhận thấy là một ô đất có thể có tối đa 8 ô chung đỉnh.*

Các bạn hãy xác định xem mạng xã hội hiện tại của Tý có là hoàn hảo hay không? Nếu có hãy in ra "YES", "NO" trong trường hợp ngược lại.

**Input:** Dòng 1: 2 số nguyên cách nhau bởi dấu cách: N và M. Dòng 2..N+1: M kí tự liên tiếp nhau mỗi dòng đại diện cho 1 hàng các ô đất. Mỗi kí tự là 'W' hoặc '.' tương ứng với ô đất đã bị ngập và ô đất vẫn còn nguyên.

**Output:** Một dòng chứa 1 số nguyên duy nhất là số ao tạo thành.

**input**

```

10 12
W.....WW.
.WWW.....WWW
....WW...WW.
.....WW.
.....W..
..W.....W..
.W.W.....WW.
W.W.W.....W.
.W.W.....W.
..W.....W.

```

**output**

```

3

```

## DSA09037 - HỌP MẶT

Có K người ( $1 \leq K \leq 100$ ) đứng tại vị trí nào đó trong N địa điểm cho trước ( $1 \leq N \leq 1,000$ ) được đánh số từ 1..N. Các điểm được nối với nhau bởi M đoạn đường một chiều ( $1 \leq M \leq 10,000$ ) (không có đoạn đường nào nối một điểm với chính nó).

Mọi người muốn cùng tụ họp tại một địa điểm nào đó. Tuy nhiên, với các đường đi cho trước, chỉ có một số địa điểm nào đó có thể được chọn là điểm họp mặt. Cho trước K, N, M và vị trí ban đầu của K người cùng với M đường đi một chiều, hãy xác định xem có bao nhiêu điểm có thể được chọn làm điểm họp mặt.

**Input:**

- Dòng 1: Ghi 3 số: K, N, và M
- Dòng 2 đến K+1: dòng i+1 chứa một số nguyên trong khoảng (1..N) cho biết địa điểm mà người thứ i đang đứng.
- Dòng K+2 đến M+K+1: Mỗi dòng ghi một cặp số A và B mô tả một đoạn đường đi một chiều từ A đến B (cả hai trong khoảng 1..N và  $A \neq B$ ).

**Output:**

- Số địa điểm có thể được chọn là điểm họp mặt.

**input**

```
2 4 4
2
3
1 2
1 4
2 3
3 4
```

**output**

```
2
```

*Giải thích Ví dụ: có thể họp mặt tại điểm 3 và điểm 4.*

## DSA09049 - LAN TRUYỀN DINH DƯỠNG

Cho một cây có N đỉnh và gốc tại đỉnh 1. Mỗi nút lá có chức năng tổng hợp chất dinh dưỡng, đồng thời lan truyền năng lượng cũng như các chất dinh dưỡng cho nút cha của nó để thực hiện quá trình tổng hợp.

Quá trình lan truyền năng lượng sẽ làm tăng giá trị dinh dưỡng lên 1. Tức là nếu một nút tổng hợp được giá trị dinh dưỡng bằng V, sẽ giúp cho nút cha của nó sẽ tổng hợp được giá trị dinh dưỡng bằng V+1. Nếu một nút cha có nhiều nút con, giá trị dinh dưỡng của nó bằng tổng của giá trị dinh dưỡng của các nút con gửi lên.

Biết rằng mỗi nút lá ban đầu tổng hợp được chất dinh dưỡng có giá trị 1. Nhiệm vụ của bạn là hãy xác định xem mỗi nút của cây tổng hợp được bao nhiêu giá trị dinh dưỡng?

**Input:**

- Dòng đầu tiên là số lượng đỉnh N ( $N \leq 200\,000$ ).
- Dòng tiếp theo gồm N-1 số nguyên,  $a[2]$ ,  $a[3]$ , ...,  $a[N]$  lần lượt là cha của các nút 2, 3, ..., N.

**Output:**



- In ra N số nguyên, số nguyên thứ  $i$  cho biết giá trị dinh dưỡng mà nút  $i$  tổng hợp được.

### Test1

input

3

1 1

output

5 1 1

### Test2

input

5

1 2 2 4

output

13 8 1 3 1

## DSA10003 - KIỂM TRA CHU TRÌNH SỬ DỤNG DISJOIN SET

Cho đồ thị vô hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Sử dụng **Disjoin Set**, hãy kiểm tra xem đồ thị có tồn tại chu trình hay không?

Input:

- Dòng đầu tiên đưa vào  $T$  là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào hai số  $|V|, |E|$  tương ứng với số đỉnh, số cạnh của đồ thị; Dòng tiếp theo đưa vào các bộ đôi  $u \in V, v \in V$  tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 100; 1 \leq |V| \leq 10^3; 1 \leq |E| \leq |V|(|V|-1)/2$ ;

Output:

- Đưa ra YES hoặc "NO" kết quả test theo từng dòng tương ứng với đồ thị tồn tại hoặc không tồn tại chu trình.

input

1

6 9

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

output

YES

## DSA10004 - ĐƯỜNG ĐI VÀ CHU TRÌNH EULER VỚI ĐỒ THỊ VÔ HƯỚNG

Cho đồ thị vô hướng liên thông  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy kiểm tra xem đồ thị có đường đi Euler hay chu trình Euler hay không?

Đường đi Euler bắt đầu tại một đỉnh, và kết thúc tại một đỉnh khác.

Chu trình Euler bắt đầu tại một đỉnh, và kết thúc chu trình tại chính đỉnh đó.

**Input:**

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào hai số  $|V|$ ,  $|E|$  tương ứng với số đỉnh, số cạnh của đồ thị; Dòng tiếp theo đưa vào các bộ đôi  $u \in V$ ,  $v \in V$  tương ứng với một cạnh của đồ thị.
- T,  $|V|$ ,  $|E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

**Output:**

- Đưa ra 1, 2, 0 kết quả mỗi test theo từng dòng tương ứng với đồ thị có đường đi Euler, chu trình Euler và trường hợp không tồn tại.

**input**

```
2
6 10
1 2 1 3 2 3 2 4 2 5 3 4 3 5 4 5 4 6 5 6
6 9
1 2 1 3 2 3 2 4 2 5 3 4 3 5 4 5 4 6
```

**output**

```
2
1
```

## DSA10005 - CHU TRÌNH EULER TRONG ĐỒ THỊ CÓ HƯỚNG

Cho đồ thị có hướng liên thông yếu  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy kiểm tra xem đồ thị có chu trình Euler hay không?

**Input:**

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào hai số  $|V|$ ,  $|E|$  tương ứng với số đỉnh, số cạnh của đồ thị; Dòng tiếp theo đưa vào các bộ đôi  $u \in V$ ,  $v \in V$  tương ứng với một cạnh của đồ thị.
- T,  $|V|$ ,  $|E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

**Output:**

- Đưa ra 1, 0 kết quả mỗi test theo từng dòng tương ứng với đồ thị có chu trình Euler và trường hợp không tồn tại đáp án.

**input**

```
2
6 10
1 2 2 4 2 5 3 1 3 2 4 3 4 5 5 3 5 6 6 4
3 3
```

1 2 2 3 1 3

**output**

1

0

## TN05009 - CHU TRÌNH THEO DFS

Cho đồ thị vô hướng  $G =$  được biểu diễn dưới dạng danh sách cạnh. Hãy kiểm tra xem đồ thị có chu trình từ đỉnh 1 hay không, nếu có, hãy in ra chu trình đầu tiên tìm thấy theo DFS.

**Input:**

- Dòng đầu tiên đưa vào  $T$  là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào hai số  $N, M$  tương ứng với số đỉnh, số cạnh.
- Dòng tiếp theo đưa vào các bộ đôi  $u, v$  tương ứng với một cạnh của đồ thị.
- $T, N, M$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq M \leq N(N-1)/2$ ;

**Output:**

- Đưa ra chu trình đầu tiên tìm thấy theo DFS từ đỉnh 1.
- Nếu không có chu trình từ đỉnh 1, ghi ra NO

**input**

1

6 9

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

**output**

1 2 3 1

## Graph (Advanced)

### DSA09018 - LIỆT KÊ ĐỈNH TRỤ

Cho đồ thị vô hướng liên thông  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy đưa ra tất cả các đỉnh trụ của đồ thị?

**Input:**

- Dòng đầu tiên đưa vào  $T$  là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào hai số  $|V|, |E|$  tương ứng với số đỉnh và số cạnh; Dòng tiếp theo đưa vào các bộ đôi  $u \in V, v \in V$  tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq |V| \leq 10^3$ ;  $1 \leq |E| \leq |V|(|V|-1)/2$ ;

**Output:**

- Đưa ra danh sách các đỉnh trụ của mỗi test theo từng dòng.

### input

```
1
5 5
1 2 1 3 2 3 2 5 3 4
```

### output

```
2 3
```

## DSA09018 - KIỂM TRA CHU TRÌNH

Cho đồ thị vô hướng  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh. Hãy kiểm tra xem đồ thị có tồn tại chu trình hay không?

### Input:

- Dòng đầu tiên đưa vào  $T$  là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào hai số  $|V|, |E|$  tương ứng với số đỉnh và số cạnh; Dòng tiếp theo đưa vào các bộ đôi  $u \in V, v \in V$  tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 100; 1 \leq |V| \leq 10^3; 1 \leq |E| \leq |V|(|V|-1)/2$ ;

### Output:

- Đưa ra YES hoặc "NO" kết quả test theo từng dòng tương ứng với đồ thị tồn tại hoặc không tồn tại chu trình.

### input

```
1
6 9
1 2 1 3 2 3 2 5 3 4 3 5 4 5 4 6 5 6
```

### output

```
YES
```

## DSA10001 - TÌM ĐƯỜNG

Cho một bảng  $S[][]$  kích thước  $N \times M$ , bao gồm các ô trống, các vật cản. Ban đầu bạn ở vị trí  $S$ . Nhiệm vụ của bạn là hãy di chuyển tới vị trí  $T$ , sao cho số lần đổi hướng không quá hai lần.

### Input:

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test bắt đầu bởi hai số nguyên  $N$  và  $M$  ( $1 \leq N, M \leq 500$ ).
- $N$  dòng tiếp theo, mỗi dòng gồm  $M$  kí tự mô tả bảng  $S$ . Trong đó: '.' là một ô trống, '\*' là vật cản, 'S' là vị trí xuất phát và 'T' là vị trí đích. (Chỉ có một vị trí  $S$  và  $T$  duy nhất).

### Output:

- Với mỗi test, in ra "YES" nếu tìm được đường đi, ra in "NO" trong trường hợp ngược lại.

**input**

```

2
5 5
..S..
****.
T....
****.
.....
5 5
S....
****.
.....
.****
..T..

```

**output**

```

YES
NO

```

**DSA10006 - CÂY KHUNG CỦA ĐỒ THỊ THEO THUẬT TOÁN DFS**

Cho đồ thị vô hướng  $G=(V, E)$ . Hãy xây dựng một cây khung của đồ thị  $G$  với đỉnh  $u \in V$  là gốc của cây bằng thuật toán DFS.

**Input:**

Dòng đầu tiên gồm một số nguyên  $T$  ( $1 \leq T \leq 20$ ) là số lượng bộ test.

Tiếp theo là  $T$  bộ test, mỗi bộ test có dạng sau:

- Dòng đầu tiên gồm 3 số nguyên  $N=|V|$ ,  $M=|E|$ ,  $u$  ( $1 \leq N \leq 10^3$ ,  $1 \leq M \leq 10^5$ ,  $1 \leq u \leq N$ ).
- Mỗi test bắt đầu bởi hai số nguyên  $N$  và  $M$  ( $1 \leq N, M \leq 500$ ).
- $M$  dòng tiếp theo, mỗi dòng gồm 2 số nguyên  $a, b$  ( $1 \leq a, b \leq N$ ,  $a \neq b$ ) tương ứng cạnh nối hai chiều từ  $a$  tới  $b$ .
- Dữ liệu đảm bảo giữa hai đỉnh chỉ tồn tại nhiều nhất một cạnh nối.

**Output:**

Với mỗi bộ test, nếu tồn tại cây khung thì in ra  $N - 1$  cạnh của cây khung với gốc là đỉnh  $u$  trên  $N - 1$  dòng theo thứ tự duyệt của thuật toán DFS. Ngược lại nếu không tồn tại cây khung thì in ra -1.

**input**

```

2
4 4 2
1 2
1 3
2 4
3 4
4 2 2
1 2
3 4

```

**output**

```
2 1
1 3
3 4
-1
```

## DSA10007 - CÂY KHUNG CỦA ĐỒ THỊ THEO THUẬT TOÁN BFS

Cho đồ thị vô hướng  $G=(V, E)$ . Hãy xây dựng một cây khung của đồ thị  $G$  với đỉnh  $u \in V$  là gốc của cây bằng thuật toán BFS.

### Input:

Dòng đầu tiên gồm một số nguyên  $T$  ( $1 \leq T \leq 20$ ) là số lượng bộ test.

Tiếp theo là  $T$  bộ test, mỗi bộ test có dạng sau:

- Dòng đầu tiên gồm 3 số nguyên  $N=|V|$ ,  $M=|E|$ ,  $u$  ( $1 \leq N \leq 10^3$ ,  $1 \leq M \leq 10^5$ ,  $1 \leq u \leq N$ ).
- Mỗi test bắt đầu bởi hai số nguyên  $N$  và  $M$  ( $1 \leq N, M \leq 500$ ).
- $M$  dòng tiếp theo, mỗi dòng gồm 2 số nguyên  $a, b$  ( $1 \leq a, b \leq N$ ,  $a \neq b$ ) tương ứng cạnh nối hai chiều từ  $a$  tới  $b$ .
- Dữ liệu đảm bảo giữa hai đỉnh chỉ tồn tại nhiều nhất một cạnh nối.

### Output:

Với mỗi bộ test, nếu tồn tại cây khung thì in ra  $N - 1$  cạnh của cây khung với gốc là đỉnh  $u$  trên  $N - 1$  dòng theo thứ tự duyệt của thuật toán BFS. Ngược lại nếu không tồn tại cây khung thì in ra -1.

### input

```
2
4 4 2
1 2
1 3
2 4
3 4
4 2 2
1 2
3 4
```

### output

```
2 1
2 4
1 3
-1
```

## DSA10008 - DIJKSTRA

Cho đồ thị có trọng số không âm  $G=<V, E>$  được biểu diễn dưới dạng danh sách cạnh trọng số. Hãy viết chương trình tìm đường đi ngắn nhất từ đỉnh  $u \in V$  đến tất cả các đỉnh còn lại trên đồ thị.

### Input:

- Dòng đầu tiên đưa vào  $T$  là số lượng bộ test.

- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm  $|E|+1$  dòng: dòng đầu tiên đưa vào hai ba số  $|V|, |E|$  tương ứng với số đỉnh và  $u \in V$  là đỉnh bắt đầu;  $|E|$  dòng tiếp theo mỗi dòng đưa vào bộ ba  $u \in V, v \in V, w$  tương ứng với một cạnh cùng với trọng số cạnh của đồ thị.
- $T, |V|, |E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 100; 1 \leq |V| \leq 10^3; 1 \leq |E| \leq |V|(|V|-1)/2$ ;

#### Output:

- Đưa ra kết quả của mỗi test theo từng dòng. Kết quả mỗi test là trọng số đường đi ngắn nhất từ đỉnh  $u$  đến các đỉnh còn lại của đồ thị theo thứ tự tăng dần các đỉnh.

#### input

```
1
9 12 1
1 2 4
1 8 8
2 3 8
2 8 11
3 4 7
3 6 4
3 9 2
4 5 9
4 6 14
5 6 10
6 7 2
6 9 6
```

#### output

```
0 4 12 19 26 16 18 8 14
```

### DSA10009 - FLOYD

Cho đơn đồ thị vô hướng liên thông  $G = (V, E)$  gồm  $N$  đỉnh và  $M$  cạnh, các đỉnh được đánh số từ 1 tới  $N$  và các cạnh được đánh số từ 1 tới  $M$ .

Có  $Q$  truy vấn, mỗi truy vấn yêu cầu bạn tìm đường đi ngắn nhất giữa đỉnh  $X[i]$  tới  $Y[i]$ .

#### Input:

- Dòng đầu tiên hai số nguyên  $N$  và  $M$  ( $1 \leq N \leq 100, 1 \leq M \leq N*(N-1)/2$ ).
- $M$  dòng tiếp theo, mỗi dòng gồm 3 số nguyên  $u, v, c$  cho biết có cạnh nối giữa đỉnh  $u$  và  $v$  có độ dài bằng  $c$  ( $1 \leq c \leq 1000$ ).
- Tiếp theo là số lượng truy vấn  $Q$  ( $1 \leq Q \leq 100\,000$ ).
- $Q$  dòng tiếp theo, mỗi dòng gồm 2 số nguyên  $X[i], Y[i]$ .

#### Output:

- Với mỗi truy vấn, in ra đáp án là độ dài đường đi ngắn nhất tìm được.

#### input

```
5 6
1 2 6
```

```
1 3 7
2 4 8
3 4 9
3 5 1
4 5 2
3
1 5
2 5
4 3
```

**output**

```
8
10
3
```

## DSA10011 - DI CHUYỂN TRÊN BẢNG SỐ

Cho một bảng số kích thước  $N \times M$ . Chi phí khi đi qua ô  $(i,j)$  bằng  $A[i][j]$ . Nhiệm vụ của bạn là hãy tìm một đường đi từ ô  $(1, 1)$  tới ô  $(N, M)$  sao cho chi phí là nhỏ nhất. Tại mỗi ô, bạn được phép đi sang trái, sang phải, đi lên trên và xuống dưới.

**Input:**

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test bắt đầu bởi hai số nguyên  $N$  và  $M$  ( $1 \leq N, M \leq 500$ ).
- $N$  dòng tiếp theo, mỗi dòng gồm  $M$  số nguyên  $A[i][j]$  ( $0 \leq A[i][j] \leq 9$ ).

**Output:**

- Với mỗi test, in ra một số nguyên là chi phí nhỏ nhất cho đường đi tìm được.

**input**

```
3
4
5
0 3 1 2 9
7 3 4 9 9
1 7 5 5 3
2 3 4 2 5
1
6
0 1 2 3 4 5
5 5
1 1 1 9 9
9 9 1 9 9
1 1 1 9 9
1 9 9 9 9
1 1 1 1 1
```

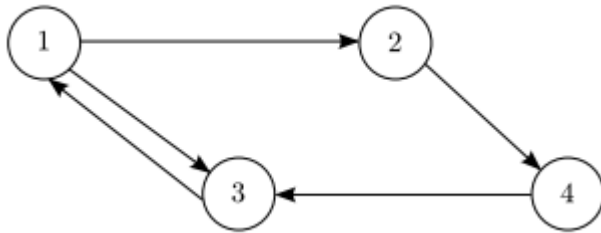
**output**

```
24
15
13
```

## DSA10012 - ĐƯỜNG ĐI TRUNG BÌNH



Cho một đồ thị có hướng gồm  $N$  đỉnh và  $M$  cạnh. Nhiệm vụ của bạn là hãy tính khoảng cách trung bình ngắn nhất giữa hai node bất kì nếu như chúng liên thông với nhau. Input đảm bảo rằng trong một nhóm liên thông, nếu như  $u$  đi tới được  $v$  thì  $v$  cũng đi tới được  $u$  với mọi cặp  $u, v$ .



**Input:** Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ). Mỗi test bắt đầu bởi hai số nguyên  $N$  và  $M$  ( $1 \leq N \leq 100$ ,  $M \leq N*(N-1)/2$ ).  $M$  dòng tiếp theo, mỗi dòng gồm 2 số nguyên  $u, v$  cho biết có cạnh nối đơn hướng từ  $u$  tới  $v$ .

**Output:** Với mỗi test, in ra đáp án tìm được với độ chính xác 2 chữ số sau dấu phẩy.

#### input

```

2
4 5
1 2
2 4
1 3
3 1
4 3
7 5
1 2
1 4
4 2
2 7
7 1

```

#### output

```

1.83
1.75

```

Giải thích test 1: Ta có

$d(1, 2) = 1$ ,  $d(1, 3) = 1$ ,  $d(1, 4) = 2$ ;  $d(2, 1) = 3$ ,  $d(2, 3) = 2$ ,  $d(2, 4) = 1$ ;

$d(3, 1) = 1$ ,  $d(3, 2) = 2$ ,  $d(3, 4) = 3$ ;  $d(4, 1) = 2$ ,  $d(4, 2) = 3$ ,  $d(4, 3) = 1$ .

Trung bình bằng  $22/12 = 1.83$

## DSA10015 - KRUSKAL

Cho đồ thị vô hướng có trọng số  $G = \langle V, E, W \rangle$ . Nhiệm vụ của bạn là hãy xây dựng một cây khung nhỏ nhất của đồ thị bằng thuật toán Kruskal.

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .

- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất đưa vào hai số  $V, E$  tương ứng với số đỉnh và số cạnh của đồ thị; phần thứ 2 đưa vào  $E$  cạnh của đồ thị, mỗi cạnh là một bộ 3: đỉnh đầu, đỉnh cuối và trọng số của cạnh.
- $T, S, D$  thỏa mãn ràng buộc:  $1 \leq T \leq 100; 1 \leq V \leq 100; 1 \leq E, W \leq 10000$ .

#### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
2
3 3
1 2 5
2 3 3
1 3 1
2 1
1 2 5
```

#### output

```
4
5
```

## DSA10016 - NỐI ĐIỂM

Cho  $N$  điểm trên mặt phẳng Oxy. Để vẽ được đoạn thẳng nối  $A$  và  $B$  sẽ tốn chi phí tương đương với khoảng cách từ  $A$  tới  $B$ .

Nhiệm vụ của bạn là nối các điểm với nhau, sao cho  $N$  điểm đã cho tạo thành 1 thành phần liên thông duy nhất và chi phí để thực hiện là nhỏ nhất có thể.

#### Input:

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test bắt đầu bởi số nguyên  $N$  ( $1 \leq N \leq 100$ ).
- $N$  dòng tiếp theo, mỗi dòng gồm 2 số thực  $x[i], y[i]$  là tọa độ của điểm thứ  $i$  ( $|x[i]|, |y[i]| \leq 100$ ).

#### Output:

- Với mỗi test, in ra chi phí nhỏ nhất tìm được với độ chính xác 6 chữ số thập phân sau dấu phẩy.

#### input

```
1
3
1.0 1.0
2.0 2.0
2.0 4.0
```

#### output

```
3.414214
```

## DSA10017 - CHU TRÌNH ÂM

Cho đồ thị có trọng số  $G = \langle V, E \rangle$  được biểu diễn dưới dạng danh sách cạnh trọng số âm hoặc dương. Hãy viết chương trình xác định xem đồ thị có chu trình âm hay không.

### Input:

- Dòng đầu tiên đưa vào  $T$  là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm  $|E| + 1$  dòng: dòng đầu tiên đưa vào hai số  $|V|, |E|$  tương ứng với số đỉnh và số cạnh của đồ thị;  $|E|$  dòng tiếp theo mỗi dòng đưa vào bộ ba  $u \in V, v \in V, w$  tương ứng với một cạnh cùng với trọng số cạnh của đồ thị.
- $T, |V|, |E|$  thỏa mãn ràng buộc:  $1 \leq T \leq 100; 1 \leq |V| \leq 10^3; 1 \leq |E| \leq |V|(|V|-1)/2$ ;

### Output:

- Đưa ra 1 hoặc 0 theo từng dòng của mỗi test tương ứng với đồ thị có hoặc không có chu trình âm.

### input

```
2
3 3
1 2 -1
2 3 4
3 1 -2
3 3
1 2 -1
2 3 2
3 1 -2
```

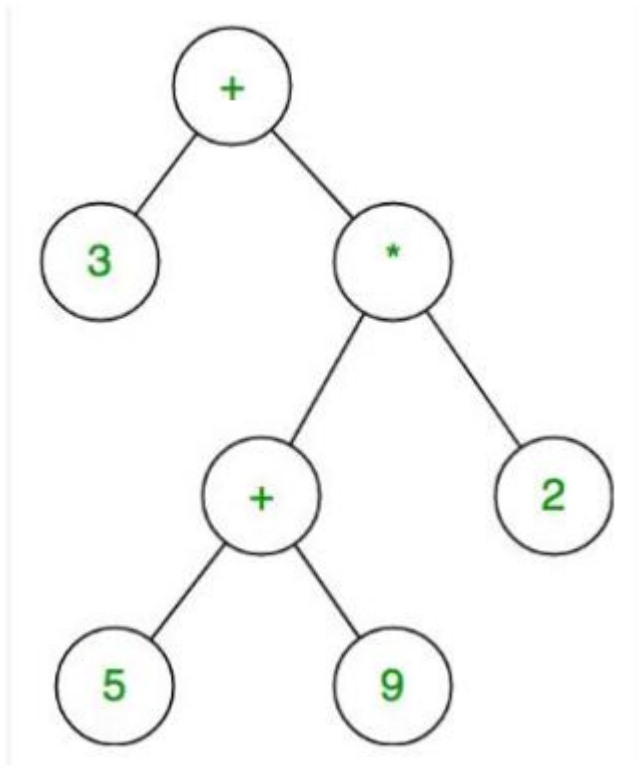
### output

```
0
1
```

## Tree

## DSA11001 - CÂY BIỂU THỨC 1

Cây biểu thức là một cây nhị phân trong đó mỗi node trung gian là một phép toán, mỗi node lá là một toán hạng. Ví dụ với biểu thức  $P = 3 + ((5+9)*2)$  sẽ được biểu diễn như cây dưới đây.



Đối với cây biểu thức, duyệt theo thứ tự trước ta sẽ được biểu thức tiền tố, duyệt theo thứ tự sau ta sẽ được biểu thức hậu tố, duyệt theo thứ tự giữa ta được biểu thức trung tố. Chú ý, cây biểu thức luôn là cây nhị phân đầy (mỗi node trung gian đều có hai node con).

Cho biểu thức hậu tố P, hãy sử dụng cây biểu thức để đưa ra biểu thức trung tố tương ứng với biểu thức P.

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test là một biểu thức hậu tố P.
- T, P thỏa mãn ràng buộc :  $1 \leq T \leq 100$ ;  $1 \leq \text{length}(P) \leq 100$ .

#### Output:

- Đưa ra biểu thức trung tố tương ứng với P.

#### input

```
2
ab+ef*g*-
wlrb+-*
```

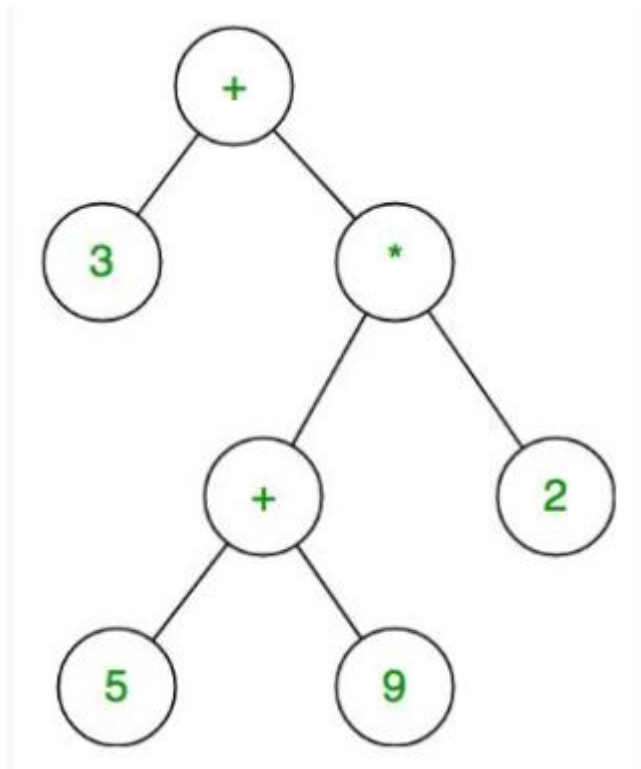
#### output

```
a + b - e * f * g
w * l - r + b
```

## DSA11002 - CÂY BIỂU THỨC 2

Cho một cây biểu thức là một cây nhị phân đầy đủ bao gồm các phép toán +, -, \*, / và một số toán hạng có giá trị nguyên. Nhiệm vụ của bạn là hãy tính toán giá trị biểu thức được biểu diễn

trên cây nhị phân đầy đủ. Ví dụ với cây dưới đây là biểu diễn của biểu thức  $P = ((5*4) + (100-20))$  sẽ cho ta giá trị là 100.



#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test là gồm hai dòng: dòng thứ nhất đưa vào N là số lượng node của cây; dòng thứ hai đưa vào nội dung các node của cây; các node được viết cách nhau một vài khoảng trống. Các số có giá trị nguyên không vượt quá 1000.
- T, N, P thỏa mãn ràng buộc :  $1 \leq T \leq 100$ ;  $1 \leq N$ ,  $\text{length}(P) \leq 100$ .

#### Output:

- Đưa ra giá trị của cây biểu thức.

#### input

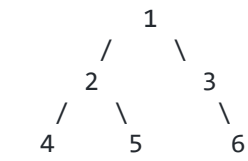
```
2
7
+ * - 5 4 100 20
3
- 4 7
```

#### output

```
100
-3
```

## DSA11003 - DUYỆT CÂY 1

Cho phép duyệt cây nhị phân Inorder và Preorder, hãy đưa ra kết quả phép duyệt Postorder của cây nhị phân. Ví dụ với cây nhị phân có các phép duyệt cây nhị phân của cây dưới đây:



Inorder: 4 2 5 1 3 6

Preorder: 1 2 4 5 3 6

Postorder: 4 5 2 6 3 1

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng đầu tiên đưa vào số N là số lượng node; dòng tiếp theo đưa vào N số theo phép duyệt Inorder; dòng cuối cùng đưa vào N số là kết quả của phép duyệt Preorder; các số được viết cách nhau một vài khoảng trống.
- T, N, node thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 1000$ ;  $1 \leq \text{giá trị node} \leq 10^4$ ;

### Output:

- Đưa ra kết quả phép duyệt Postorder theo từng dòng.

### input

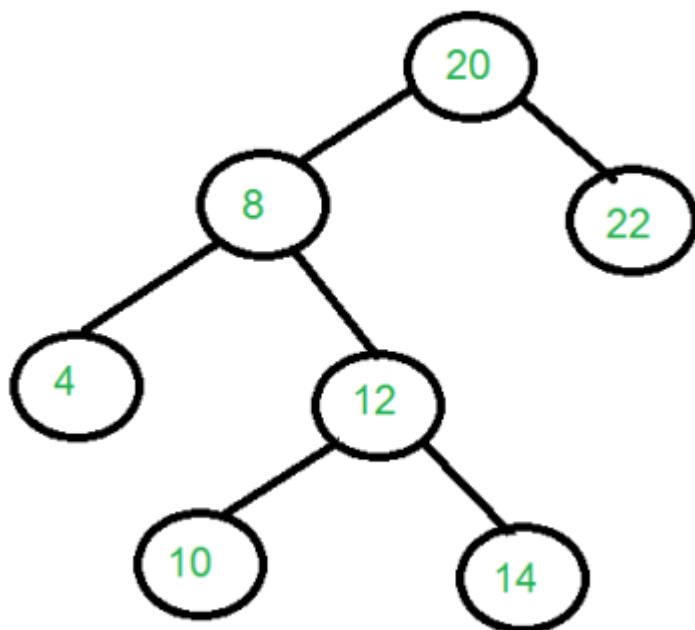
```
1
6
4 2 5 1 3 6
1 2 4 5 3 6
```

### output

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

## DSA11004 - DUYỆT CÂY THEO MỨC

Cho cây nhị phân, nhiệm vụ của bạn là duyệt cây theo Level-order. Phép duyệt level-order trên cây là phép thăm node theo từng mức của cây. Ví dụ với cây dưới đây sẽ cho ta kết quả của phép duyệt level-order: 20 8 22 4 12 10 14.



#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trống.
- T, N, u, v, thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq u, v \leq 10^4$ ;

#### Output:

- Đưa ra kết quả phép duyệt level-order theo từng dòng.

#### input

```

2
2
1 2 R 1 3 L
4
10 20 L 10 30 R 20 40 L 20 60 R

```

#### output

```

1 3 2
10 20 30 40 60

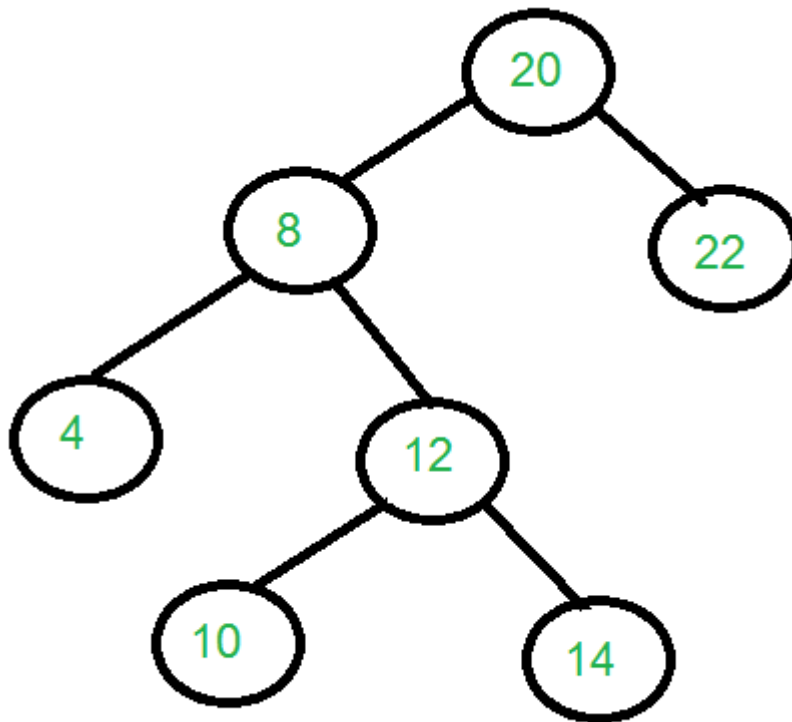
```

## DSA11005 - DUYỆT CÂY 2

Cho hai mảng là phép duyệt Inorder và Level-order, nhiệm vụ của bạn là xây dựng cây nhị phân và đưa ra kết quả phép duyệt Postorder. Level-order là phép duyệt theo từng mức của cây. Ví dụ như cây dưới đây ta có phép Inorder và Level-order như dưới đây:

Inorder : 4 8 10 12 14 20 22

Level order: 20 8 22 4 12 10 14



**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng đầu tiên đưa vào số N là số lượng node; dòng tiếp theo đưa vào N số là phép duyệt Inorder; dòng cuối cùng đưa vào N số là phép duyệt Level-order; các số được viết cách nhau một vài khoảng trống.
- T, N, node thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq A[i] \leq 10^4$ ;

**Output:**

- Đưa ra kết quả phép duyệt Postorder theo từng dòng.

**input**

```
2
3
1 0 2
0 1 2
7
3 1 4 0 5 2 6
0 1 2 3 4 5 6
```

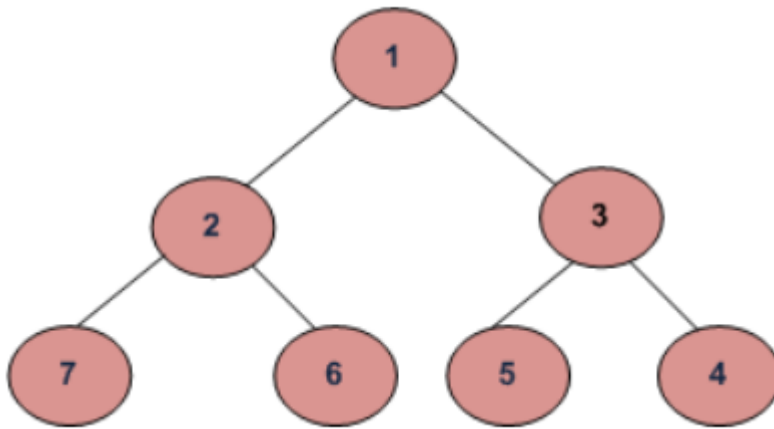
**output**

```
1 2 0
3 4 1 5 6 2 0
```



## DSA11006 - DUYỆT CÂY KIỂU XOẮN ỐC

Cho cây nhị phân, nhiệm vụ của bạn là duyệt cây theo xoắn ốc (spiral-order). Phép. Ví dụ với cây dưới đây sẽ cho ta kết quả của phép duyệt spiral-order: 1 2 3 4 5 6 7.



### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x=R nếu v là con phải, x=L nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trống.
- T, N, u, v, thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq u, v \leq 10^4$ ;

### Output:

- Đưa ra kết quả phép duyệt Postorder theo từng dòng.

### input

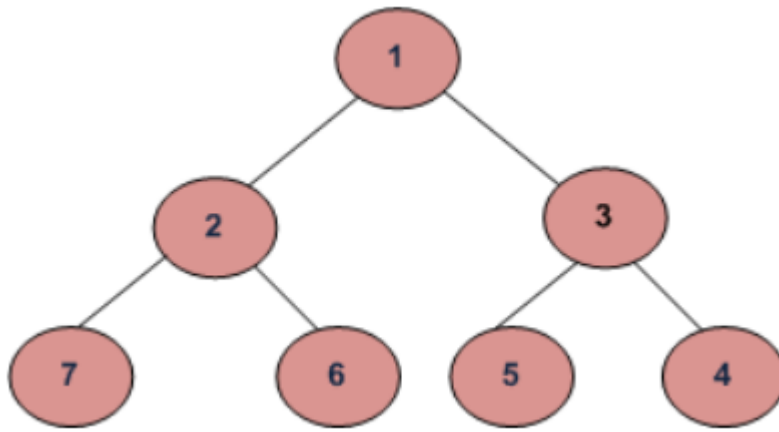
```
2
2
1 2 R 1 3 L
4
10 20 L 10 30 R 20 40 L 20 60 R
```

### output

```
1 3 2
10 0 30 60 40
```

## DSA11008 - KIỂM TRA NODE LÁ

Cho cây nhị phân, nhiệm vụ của bạn là kiểm tra xem tất cả các node lá của cây có cùng một mức hay không? Ví dụ với cây dưới đây sẽ cho ta kết quả là Yes.



#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trống.
- T, N, u, v, thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq u, v \leq 10^4$ ;

#### Output:

- Đưa ra kết quả phép duyệt Postorder theo từng dòng.

#### input

```

2
2
1 2 R 1 3 L
4
10 20 L 10 30 R 20 40 L 20 60 R

```

#### output

```

1
0

```

## DSA11010 - CÂY NHỊ PHÂN HOÀN HẢO

Cho cây nhị phân, nhiệm vụ của bạn là kiểm tra xem cây nhị phân có phải là một cây hoàn hảo hay không (perfect tree)? Một cây nhị phân được gọi là cây hoàn hảo nếu tất cả các node trung gian của nó đều có hai node con và tất cả các node lá đều có cùng một mức.

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trống.

- $T, N, u, v$ , thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq u, v \leq 10^4$ ;

**Output:**

- Đưa ra kết quả phép duyệt Postorder theo từng dòng.

**input**

```
3
6
10 20 L 10 30 R 20 40 L 20 50 R 30 60 L 30 70 R
2
18 15 L 18 30 R
5
1 2 L 2 4 R 1 3 R 3 5 L 3 6 R
```

**output**

```
Yes
Yes
No
```

## DSA11011 - CÂY NHỊ PHÂN ĐỦ

Cho cây nhị phân, nhiệm vụ của bạn là kiểm tra xem cây nhị phân có phải là một cây đủ hay không (full binary tree)? Một cây nhị phân được gọi là cây đủ nếu tất cả các node trung gian của nó đều có hai node con.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số  $N$  là số lượng cạnh của cây; dòng tiếp theo đưa vào  $N$  bộ ba  $(u, v, x)$ , trong đó  $u$  là node cha,  $v$  là node con,  $x = R$  nếu  $v$  là con phải,  $x = L$  nếu  $v$  là con trái;  $u, v, x$  được viết cách nhau một vài khoảng trống.
- $T, N, u, v$ , thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq u, v \leq 10^4$ ;

**Output:**

- Đưa ra kết quả phép duyệt Postorder theo từng dòng.

**input**

```
2
4
1 2 L 1 3 R 2 4 L 2 5 R
3
1 2 L 1 3 R 2 4 L
```

**output**

```
1
0
```

## DSA11012 - CÂY NHỊ PHÂN BẰNG NHAU

Cho hai cây nhị phân, nhiệm vụ của bạn là kiểm tra xem cây nhị phân có giống nhau hay không?

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái của mỗi cây; u, v, x được viết cách nhau một vài khoảng trống.
- T, N, u, v, thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq u, v \leq 10^4$ ;

**Output:**

- Đưa ra kết quả phép duyệt Postorder theo từng dòng.

**input**

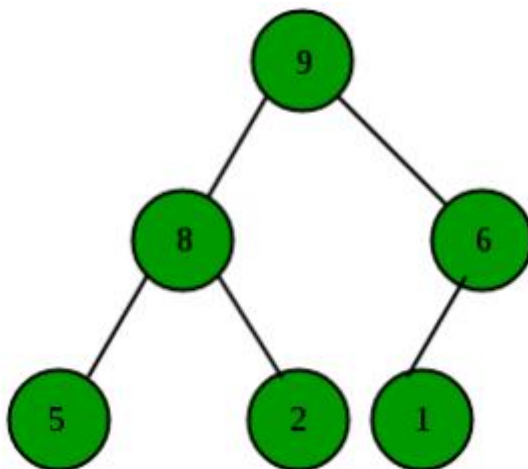
```
2
2
1 2 L 1 3 R
2
1 2 L 1 3 R
2
1 2 L 1 3 R
2
1 3 L 1 2 R
```

**output**

```
1
0
```

## DSA11014 - TỔNG NODE LÁ BÊN PHẢI

Cho cây nhị phân, nhiệm vụ của bạn là tính tổng của tất cả các node lá bên phải trên cây? Ví dụ với cây dưới đây ta có kết quả là 2.



**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trống.
- T, N, u, v, thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq u, v \leq 10^4$ ;

### Output:

- Đưa ra kết quả phép duyệt Postorder theo từng dòng.

### input

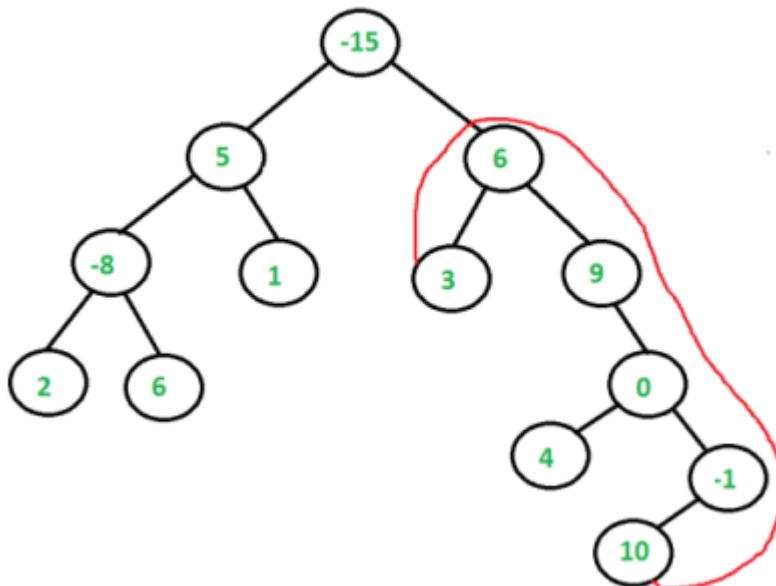
```
2
2
1 2 L 1 3 R
5
10 20 L 10 30 R 20 40 L 20 60 R 30 90 L
```

### output

```
3
60
```

## DSA11015 - TỔNG LỚN NHẤT

Cho cây nhị phân có giá trị mỗi node là một số, nhiệm vụ của bạn là tìm tổng lớn nhất từ một node lá này sang một node lá khác? Ví dụ với cây dưới đây ta có tổng lớn nhất là 27.



### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là

node cha, v là node con,  $x = R$  nếu v là con phải,  $x = L$  nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trống.

- T, N, u, v, thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq u, v \leq 10^4$ ;

#### Output:

- Đưa ra kết quả phép duyệt Postorder theo từng dòng.

#### input

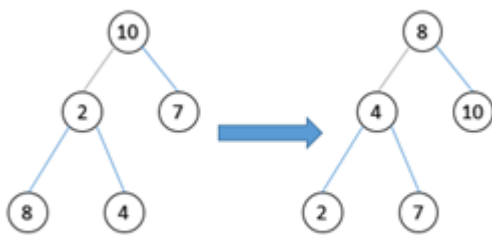
```
1
12
-15 5 L -15 6 R 5 -8 L 5 1 R -8 2 L -8 -3 R 6 3 L 6 9 R 9 0 R 0 4 L 0 -1 R -1 10 L
```

#### output

```
27
```

## DSA11016 - BIẾN ĐỔI SANG CÂY NHỊ PHÂN TÌM KIẾM

Cho cây nhị phân, nhiệm vụ của bạn là dịch chuyển cây nhị phân thành cây nhị phân tìm kiếm. Phép dịch chuyển phải bảo toàn được cấu trúc cây nhị phân ban đầu. Ví dụ dưới đây sẽ minh họa phép dịch chuyển:



#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con,  $x = R$  nếu v là con phải,  $x = L$  nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trống.
- T, N, u, v, thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq u, v \leq 10^4$ ;

#### Output:

- Đưa ra kết quả mỗi test theo từng dòng là phép duyệt Inorder của cây tìm kiếm.

#### input

```
2
2
1 2 R 1 3 L
4
10 20 L 10 30 R 20 40 L 20 60 R
```

### output

```
1 2 3
10 20 30 40 60
```

## DSA11017 - DUYỆT CÂY NHỊ PHÂN TÌM KIẾM 1

Cho mảng  $A[]$  gồm  $N$  node là biểu diễn phép duyệt theo thứ tự giữa (Preorder) của cây nhị phân tìm kiếm. Nhiệm vụ của bạn là đưa ra phép duyệt theo thứ tự sau của cây nhị phân tìm kiếm.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số  $N$  là số lượng node; dòng tiếp theo đưa vào  $N$  số  $A[i]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N$ , node thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq A[i] \leq 10^4$ ;

### Output:

- Đưa ra kết quả phép duyệt Postorder theo từng dòng.

### input

```
2
5
40 30 35 80 100
8
40 30 32 35 80 90 100 120
```

### output

```
35 30 100 80 40
35 32 30 120 100 90 80 40
```

## DSA11018 - XÂY DỰNG LẠI CÂY NHỊ PHÂN TÌM KIẾM

Cho một mảng là phép duyệt level-order của cây nhị phân tìm kiếm. Nhiệm vụ của bạn là xây dựng lại cây nhị phân tìm kiếm bảo toàn được cấu trúc cây nhị phân ban đầu.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm dòng: dòng đầu tiên đưa vào số  $N$  là số lượng node của cây tìm kiếm; dòng tiếp theo đưa vào phép duyệt level-order của cây tìm kiếm; các số được viết cách nhau một vài khoảng trống.
- $T, N$ , node thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq \text{node} \leq 10^4$ ;

### Output:

- Đưa ra kết quả mỗi test theo từng dòng là phép duyệt trước (preOrder) của cây tìm kiếm.

### input

```

2
9
7 4 12 3 6 8 1 5 10
6
1 3 4 6 7 8
output

7 4 3 1 6 5 12 8 10
1 3 4 6 7 8

```

## DSA11019 - DUYỆT CÂY NHỊ PHÂN TÌM KIẾM 2

Cho một mảng  $A[]$  gồm  $N$  phần tử biểu diễn phép duyệt preorder của cây nhị phân tìm kiếm. Nhiệm vụ của bạn là đưa ra phép duyệt postorder của cây nhị phân tìm kiếm.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số  $N$  là số lượng node của cây tìm kiếm; dòng tiếp theo đưa vào phép duyệt preorder của cây tìm kiếm; các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq A[i] \leq 10^4$ ;

### Output:

- Đưa ra kết quả mỗi test theo từng dòng là phép duyệt postorder của cây tìm kiếm.

### input

```

2
5
40 30 35 80 100
8
40 30 32 35 80 90 100 120
output

35 30 100 80 40
35 32 30 120 100 90 80 40

```

## DSA11020 - KIỂM TRA CÂY NHỊ PHÂN TÌM KIẾM

Cho một mảng  $A[]$  gồm  $N$  phần tử. Nhiệm vụ của bạn là đưa ra 1 nếu mảng  $A[]$  biểu diễn phép duyệt inorder của cây nhị phân tìm kiếm, ngược lại đưa ra 0.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$ .
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số  $N$  là số lượng node của cây tìm kiếm; dòng tiếp theo đưa vào  $N$  số  $A[i]$ ; các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i]$  thỏa mãn ràng buộc:  $1 \leq T \leq 100$ ;  $1 \leq N \leq 10^3$ ;  $1 \leq A[i] \leq 10^4$ ;

### Output:



- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
3
5
10 20 30 40 50
6
90 80 100 70 40 30
3
1 1 2
```

#### output

```
1
0
0
```

## DSA11021 - NODE LÁ CỦA CÂY NHỊ PHÂN TÌM KIẾM

Cho dãy số gồm N số là phép duyệt theo thứ tự trước (Preorder) của một cây nhị phân tìm kiếm. Hãy in ra tất cả các node lá của cây ?

Ví dụ với dãy A[] = {30, 20, 15, 25, 23, 28, 40, 35, 33, 38, 45} là phép duyệt cây theo thứ tự trước sẽ cho ta kết quả: 15, 23, 28, 33, 38, 45.

#### Input:

- Dòng đầu tiên đưa vào số lượng bộ test T ( $T \leq 100$ ).
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất là số tự nhiên N ( $N \leq 1000$ ). Dòng tiếp theo là N số là phép duyệt theo thứ tự trước của cây BST.

#### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
2
6
10 5 1 7 40 50
11
30 20 15 25 23 28 40 35 33 38 45
```

#### output

```
1 7 50
15 23 28 33 38 45
```

## DSA11022 - NODE TRUNG GIAN CỦA CÂY NHỊ PHÂN TÌM KIẾM

Cho dãy số gồm N số là phép duyệt theo thứ tự trước (Preorder) của một cây nhị phân tìm kiếm. Hãy đưa ra số các node trung gian của cây ?

Ví dụ với dãy A[] = {30, 20, 15, 25, 23, 28, 40, 35, 33, 38, 45} là phép duyệt cây theo thứ tự trước sẽ cho ta kết quả là 5 bao gồm các node: 30, 20, 25, 40, 35.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$  ( $T \leq 100$ ).
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất là số tự nhiên  $N$  ( $N \leq 1000$ ). Dòng tiếp theo là  $N$  số là phép duyệt theo thứ tự trước của cây BST.

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
6
10 5 1 7 40 50
11
30 20 15 25 23 28 40 35 33 38 45
```

**output**

```
3
5
```

**DSA11023 - ĐỘ SÂU CÂY NHỊ PHÂN TÌM KIẾM**

Cho dãy số gồm  $N$  số là phép duyệt theo thứ tự trước (Preorder) của một cây nhị phân tìm kiếm. Hãy tìm độ sâu của cây ?

Ví dụ với dãy  $A[] = \{30, 20, 15, 25, 23, 28, 40, 35, 33, 38, 45\}$  là phép duyệt cây theo thứ tự trước sẽ cho ta kết quả là 3.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$  ( $T \leq 100$ ).
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất là số tự nhiên  $N$  ( $N \leq 1000$ ). Dòng tiếp theo là  $N$  số là phép duyệt theo thứ tự trước của cây BST.

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
6
10 5 1 7 40 50
11
30 20 15 25 23 28 40 35 33 38 45
```

**output**

```
2
3
```

**DSA11024 - CÂY NHỊ PHÂN TÌM KIẾM CÂN BẰNG 1**

Hãy xây dựng một cây nhị phân tìm kiếm cân bằng từ dãy số  $A[] = \{a_0, a_1, \dots, a_{n-1}\}$ . Đưa ra nội dung node gốc của cây tìm kiếm cân bằng. Ví dụ với dãy  $A[] = \{40, 28, 45, 38, 33, 15, 25, 20, 23, 35, 30\}$  ta sẽ có cây nhị phân tìm kiếm cân bằng với node gốc là 33.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$  ( $T \leq 100$ ).
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất là số tự nhiên  $N$  ( $N \leq 10^6$ ). Dòng tiếp theo là  $N$  số của mảng  $A[]$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
11
40 28 45 38 33 15 25 20 23 35 30
10
1 2 3 4 5 6 7 8 9 10
```

**output**

```
30
5
```

## DSA11025 - CÂY NHỊ PHÂN TÌM KIẾM CÂN BẰNG 2

Hãy xây dựng một cây nhị phân tìm kiếm cân bằng từ dãy số  $A[] = \{a_0, a_1, \dots, a_{n-1}\}$ . Đưa ra phép duyệt theo thứ tự trước (preorder) của cây tìm kiếm cân bằng. Ví dụ với dãy  $A[] = \{40, 28, 45, 38, 33, 15, 25, 20, 23, 35, 30\}$  ta sẽ có phép duyệt theo thứ tự trước của cây nhị phân tìm kiếm cân bằng với node gốc là 33 : 33, 25, 20, 15, 23, 28, 30, 40, 38, 35, 45.

**Input:**

- Dòng đầu tiên đưa vào số lượng bộ test  $T$  ( $T \leq 100$ ).
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất là số tự nhiên  $N$  ( $N \leq 10^6$ ). Dòng tiếp theo là  $N$  số của mảng  $A[]$ .

**Output:**

- Đưa ra kết quả mỗi test theo từng dòng.

**input**

```
2
11
40 28 45 38 33 15 25 20 23 35 30
10
1 2 3 4 5 6 7 8 9 10
```

**output**

```
30 23 15 20 25 28 38 33 35 40 45
5 2 1 3 4 8 6 7 9 10
```

## DSA11026 - DUYỆT SAU CÂY TÌM KIẾM CÂN BẰNG

Hãy xây dựng một cây nhị phân tìm kiếm cân bằng từ dãy số  $A[] = (a_0, a_1, \dots, a_{n-1})$ . Đưa ra phép duyệt theo thứ tự sau (post-order) của cây tìm kiếm cân bằng. Ví dụ với dãy  $A[] = \{40, 28, 45, 38, 33, 15, 25, 20, 23, 35, 30\}$  ta sẽ có phép duyệt theo thứ tự trước của cây nhị phân tìm kiếm cân bằng với node gốc là 33 : 15, 23, 20, 30, 28, 25, 35, 38, 45, 40, 33.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$  ( $T \leq 100$ ).
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất là số tự nhiên  $N$  ( $N \leq 10^6$ ). Dòng tiếp theo là  $N$  số của mảng  $A[]$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
2
11
40 28 45 38 33 15 25 20 23 35 30
10
1 2 3 4 5 6 7 8 9 10
```

#### output

```
20 15 28 25 23 35 33 45 40 38 30
1 4 3 2 7 6 10 9 8 5
```

## DSA11027 - ĐẾM SỐ NODE LÁ CÂY NHỊ PHÂN TÌM KIẾM CÂN BẰNG

Hãy xây dựng một cây nhị phân tìm kiếm cân bằng từ dãy số  $A[] = (a_0, a_1, \dots, a_{n-1})$ . Đếm số node lá của cây nhị phân tìm kiếm cân bằng. Ví dụ với dãy  $A[] = \{40, 28, 45, 38, 33, 15, 25, 20, 23, 35, 30\}$  ta sẽ có phép duyệt theo thứ tự trước của cây nhị phân tìm kiếm cân bằng với số node lá là 5 : 15, 23, 30, 35 45.

### Input:

- Dòng đầu tiên đưa vào số lượng bộ test  $T$  ( $T \leq 100$ ).
- Những dòng kế tiếp đưa vào  $T$  bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất là số tự nhiên  $N$  ( $N \leq 1000$ ). Dòng tiếp theo là  $N$  số của mảng  $A[]$ .

### Output:

- Đưa ra kết quả mỗi test theo từng dòng.

#### input

```
2
11
40 28 45 38 33 15 25 20 23 35 30
10
1 2 3 4 5 6 7 8 9 10
```

### output

5  
4

## DSA11028 - ĐỘ CAO CỦA CÂY

Cho một cây có  $N$  node, gốc tại 1. Độ cao của một node  $u$  được tính bằng khoảng cách từ  $u$  tới node gốc. Độ cao của node gốc bằng 0. Nhiệm vụ của bạn là hãy tìm node xa node gốc nhất và in ra độ cao của node này.

### Input:

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test bắt đầu bằng số nguyên  $N$  ( $1 \leq N \leq 100\,000$ ).
- $N-1$  dòng tiếp theo, mỗi dòng gồm 2 số nguyên  $u, v$  cho biết node  $u$  nối với node  $v$ .

### Output:

- Với mỗi test, in ra một số nguyên là độ cao của cây.

### input

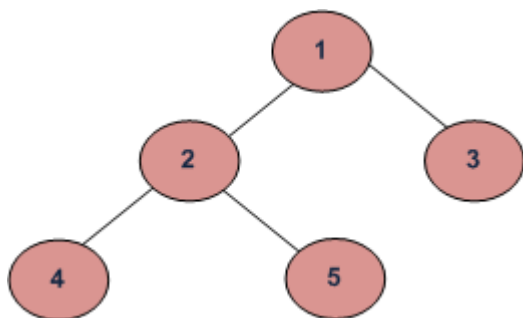
2  
5  
1 2  
1 3  
2 4  
2 5  
4  
1 2  
2 3  
3 4

### output

2  
3

## DSA11029 - ĐƯỜNG ĐI TỚI NODE LÁ

Cho một cây có  $N$  node, gốc tại 1. Với mỗi node lá, bạn hãy in ra đường đi từ node gốc tới nó.



### Input:

- Dòng đầu tiên là số lượng bộ test  $T$  ( $T \leq 20$ ).
- Mỗi test bắt đầu bằng số nguyên  $N$  ( $1 \leq N \leq 1000$ ).
- $N-1$  dòng tiếp theo, mỗi dòng gồm 2 số nguyên  $u, v$  cho biết node  $u$  nối với node  $v$ .

#### Output:

- Với mỗi test, in ra  $K$  dòng, trong đó  $K$  là số lượng node lá. Mỗi dòng là đường đi từ node gốc tới node lá  $X$ . Node lá nào có nhãn nhỏ hơn, in ra trước.

#### input

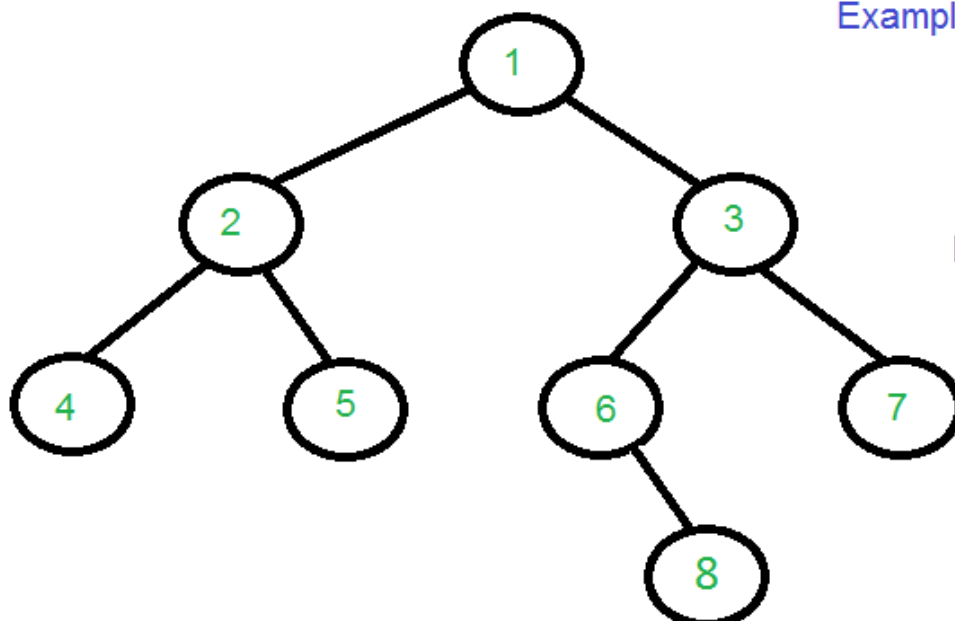
```
2
5
1 2
1 3
2 4
2 5
4
1 2
2 3
3 4
```

#### output

```
1 3
1 2 4
1 2 5
1 2 3 4
```

## DSA11030 - KHOẢNG CÁCH GIỮA HAI NODE

Cho một cây có  $N$  node, gốc tại 1. Có  $Q$  truy vấn, mỗi truy vấn yêu cầu bạn tìm khoảng cách giữa hai node  $u$  và  $v$ .



#### Examples

```

Dist(4, 5) = 2
Dist(4, 6) = 4
Dist(3, 4) = 3
Dist(2, 4) = 1
Dist(8, 5) = 5

```

#### Input:

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

- Mỗi test bắt đầu bằng số nguyên  $N$  ( $1 \leq N \leq 1000$ ).  $N-1$  dòng tiếp theo, mỗi dòng gồm 2 số nguyên  $u, v$  cho biết node  $u$  nối với node  $v$ .
- Dòng tiếp theo là số lượng truy vấn  $Q$  ( $1 \leq Q \leq 1000$ ).  $Q$  dòng tiếp theo, mỗi dòng gồm 2 số nguyên  $u$  và  $v$ .

**Output:**

- Với mỗi test, in ra  $Q$  dòng là đáp án với mỗi truy vấn.

**input**

```
1
8
1 2
1 3
2 4
2 5
3 6
3 7
6 8
5
4 5
4 6
3 4
2 4
8 5
```

**output**

```
2
4
3
1
5
```

## DSAKT060 - TRUY VẤN ĐỐI XỨNG

Cho số nguyên  $S$  gồm  $N$  chữ số. Có  $Q$  truy vấn, mỗi truy vấn thuộc một trong hai loại sau:

q L R: yêu cầu xác định xâu con  $S[L, L+1, \dots, R]$  có phải là một xâu đối xứng hay không?

c U x: thay đổi kí tự thứ  $U$  thành chữ số  $x$ .

**Input:**

- Dòng đầu tiên chứa số nguyên  $S$  gồm  $N$  kí tự ( $N \leq 10^5$ ).
- Tiếp theo là số lượng truy vấn  $Q$  ( $Q \leq 10^5$ )
- $Q$  dòng tiếp theo, mỗi dòng gồm một loại truy vấn.

**Output:**

- Với mỗi truy vấn loại 1, nếu xâu con là một số đối xứng, hãy in ra "YES", ngược lại in ra "NO".

**input**

12345

5

q 1 5

q 5 5

c 4 3

q 3 5

q 3 4

**output**

NO

YES

NO

YES