

Quy hoạch động trạng thái (DP + Bitmasks)

Biên soạn - Nguyễn Tiến Đạt

Nguồn bài - [codeforces](#)

Ngày 1 tháng 3 năm 2021

Mục lục

A. Kefa and Dishes - time limit per test: 2.5s	3
B. Little Pony and Harmony Chest - time limit per test: 4.5s	3
C. Roman and Numbers - time limit per test: 4.5s	4
D. Looking for Order - time limit per test: 5s	5
E. Rotate Columns - time limit per test: 3.5s	6

A. Kefa and Dishes - time limit per test: 2.5s

Khi Kefa tới nhà hàng và ngồi vào bàn, anh bồi bàn mang tới cho Kefa bảng menu. Ở đó có n món ăn. Kefa biết rằng anh ta cần ăn chính xác m món. Nhưng anh ta muốn thử nhiều món nhất có thể nên anh ta không muốn đặt một món hai lần.

Kefa biết rằng món ăn thứ i cho anh ta a_i đơn vị của sự thỏa mãn. Một vài món kết hợp với nhau thì rất ngon nhưng một vài món kết hợp lại rất tệ. Do đó, Kefa đặt cho bản thân k luật ăn như sau - nếu anh ta ăn món x ngay trước món y (không có món nào ở giữa món x và món y), thì sự thỏa mãn của anh ta tăng thêm c .

Hãy giúp Kefa đạt được sự thỏa mãn lớn nhất khi đến nhà hàng!!!.

Input

Dòng đầu tiên chứa ba số nguyên phân tách nhau bởi dấu cách, n , m và k ($1 \leq m \leq n \leq 18, 0 \leq k \leq n * (n - 1)$) - Số lượng món ăn trong menu, số lượng món Kefa phải ăn và số lượng luật ăn của Kefa.

Dòng thứ hai chứa n số nguyên phân a_i , ($0 \leq a_i \leq 10^9$) - Sự thỏa mãn từ món ăn thứ i .

Tiếp theo là k dòng chứa các luật. Dòng thứ i mô tả bằng ba số nguyên x_i, y_i, c_i ($1 \leq x_i, y_i \leq n, 0 \leq c_i \leq 10^9$). Nghĩa là nếu ăn món x_i ngay trước món y_i , thì sự thỏa mãn của Kefa tăng c_i . Thỏa mãn rằng, không có cặp chỉ số i và j ($1 \leq i < j \leq k$), mà $x_i = x_j$ và $y_i = y_j$.

Output

Một số nguyên duy nhất là mức độ thỏa mãn tối đa mà Kefa có thể đạt được.

Ví dụ

Input	Output
2 2 1 1 1 2 1 1	3
4 3 2 1 2 3 4 2 1 5 3 4 2	12

B. Little Pony and Harmony Chest - time limit per test: 4.5s

Công chúa Hoàng Hôn đã tới cung điện của Mặt Trời và Mặt Trăng để nghiên cứu về chiếc rương của sự Hải Hòa.

Một dãy số nguyên dương b_i là hài hòa nếu và chỉ nếu với mọi phần tử của dãy ước chung lớn nhất của chúng bằng 1. Theo một cuốn sách cổ, chìa khóa của chiếc rương là dãy hài hòa b_i cái mà cực tiểu biểu thức sau:

$$\sum_{i=1}^n |a_i - b_i|.$$

Bạn được cho dãy a_i , hãy giúp công chúa Hoàng Hôn tìm chìa khóa.

Input

Dòng đầu tiên chứa số nguyên n ($1 \leq n \leq 100$) - số lượng phần tử của dãy a và dãy b . Dòng tiếp theo chứa n số nguyên a_1, a_2, \dots, a_n ($1 \leq a_i \leq 30$).

Output

Đưa ra màn hình chìa khóa - dãy b_i cái mà cực tiểu tổng đã được mô tả bên trên. Nếu có nhiều kết quả đưa ra bất kì.

Ví dụ

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

C. Roman and Numbers - time limit per test: 4.5s

Roman là một nhà toán học trẻ, rất nổi tiếng ở Uzhland. Không may, Sereja lại không nghĩ vậy. Để làm Sereja thay đổi suy nghĩ, Roman sẵn sàng giải bất kì bài toán nào. Sau một hồi suy nghĩ, Sereja yêu cầu Roman phải giải, có bao nhiêu số thân thiết với số n , modulo m .

Số x được gọi là thân thiết với số n modulo m , nếu:

- nó có thể thu được từ việc hoán vị các chữ số của n ,
- nó không có số 0 ở đầu,
- số dư sau khi chia x cho m bằng 0.

Roman là nhà toán học giỏi, nhưng số lượng số phải tìm là quá lớn với anh ta. Vì thế anh ta nhờ bạn giải giúp.

Input

Dòng đầu tiên chứa hai số nguyên: n ($1 \leq n < 10^{18}$) và m ($1 \leq m \leq 100$).

Output

Một số nguyên duy nhất là số lượng số thân thiết với n modulo m .

Ví dụ

Input	Output
104 2	3
223 4	1
7067678 8	47

D. Looking for Order - time limit per test: 5s

Cô gái Lena thích khi mọi thứ có thứ tự, và tìm kiếm thứ tự ở mọi nơi. Một ngày nọ, khi đang chuẩn bị cho việc đi học, cô để ý thấy rằng căn phòng của cô thật bừa bộn - tất cả đồ vật từ cặp cô ý bị ném ra khắp phòng. Tất nhiên, cô gái muốn để chúng lại cặp. Vấn đề ở đây là cô gái không thể mang **nhiều hơn hai** đồ vật trong cùng một lúc, và cô gái không thể **di chuyển chiếc cặp**. Và một khi đã lấy được đồ vật, cô ta không thể để bất kì đâu ngoài chiếc cặp.

Bạn được cho tọa độ của chiếc cặp và tọa độ của các món đồ trong hệ tọa độ Descartes. Biết rằng thời gian cô gái di chuyển giữa hai đồ vật bất kì bằng bình phương khoảng cách của hai đồ vật đó. Cũng biết rằng, vị trí ban đầu của cô và chiếc cặp là như nhau. Bạn được yêu cầu tìm thứ tự các hành động, mà cô gái có thể để tất cả các đồ vật vào cặp trong thời gian ít nhất.

Input

Dòng đầu tiên chứa tọa độ chiếc cặp x_s, y_s . Dòng thứ hai chứa số nguyên n ($1 \leq n \leq 24$) - số lượng đồ vật cô gái có. Tiếp theo n dòng là tọa độ các đồ vật. Tất cả các tọa độ có giá trị tuyệt đối không vượt quá 100. Tất cả các vị trí được cho là đôi một khác nhau. Tất cả đều là số nguyên.

Output

Dòng đầu tiên chứa một số nguyên duy nhất - thời gian ít nhất để cô gái đặt hết đồ vật vào cặp.

Dòng thứ hai chứa thứ tự tối ưu của Lena. Chiếc cặp được đánh chỉ số là 0 và các đồ vật chỉ số (từ 1 tới n). Đường đi phải bắt đầu và kết thúc tại chiếc cặp. Nếu có nhiều kết quả tối ưu, in ra bất kì.

Ví dụ

Input	Output
0 0 2 1 1 -1 1	8 0 1 2 0
1 1 3 4 3 3 4 0 0	32 0 1 2 0 3 0

E. Rotate Columns - time limit per test: 3.5s

Bạn được cho một ma trận a kích thước $m \times n$. Trong một lượt di chuyển, bạn có thể chọn bất kì cột nào và dịch chuyển (cyclically shift) các phần tử trong cột này. Bạn có thể thực hiện các thao tác như đã nêu bao nhiêu lần tùy ý (có thể không lần). Bạn có thể thực hiện thao tác trên một cột vô số lần.

Sau khi dịch chuyển, bạn tính toán với mỗi hàng giá trị lớn nhất trong nó. Giả sử hàng thứ i có giá trị lớn nhất là r_i . Giá trị lớn nhất của $r_1 + r_2 + \dots + r_n$?

Input

Dòng đầu tiên là số nguyên t ($1 \leq t \leq 40$), số lượng test cases.

Dòng đầu tiên của mỗi test case chứa hai số nguyên n và m ($1 \leq n \leq 4, 1 \leq m \leq 100$) - Số lượng hàng và cột của ma trận a .

Tiếp theo là n dòng, mỗi dòng chứa m số nguyên là phần tử của ma trận a ($1 \leq a_{i,j} \leq 10^5$).

Output

In ra t số nguyên: câu trả lời của mỗi test case theo thứ tự trong input.

Ví dụ

Input	Output
2 2 3 2 5 7 4 2 4 3 6 4 1 5 2 10 4 8 6 6 4 9 10 5 4 9 5 8 7	12 29