The International Collegiate Programming Contest
Sponsored by ICPC Foundation

SCPC TEENS

The 2025 SCPC Teens Online Qual Collegiate Programming Contest

(Contest Problems)

Syria
August 2025

Problem A. Max it

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Silver`

David is trying to decide which operation gives a larger result: adding or multiplying two numbers. He needs your help to figure this out.

You are given two numbers. Determine which is larger: the sum of the two numbers or their product.

Input

Only one line contains two integers a and b ($1 \leq a, b \leq 100$).

Output

Print the larger value between the sum and the product.

Examples

standard input	standard output
2 3	6
1 5	6

Problem B. Xor To The End

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Lim Green`

Given T test cases, each with a pair of non-negative integers L and R, compute the bitwise XOR of all integers in the range $[L, R]$ inclusive.

Input

The first line contains a single integer T $(1 \leq T \leq 10^4)$ — the number of test cases.

Each of the next T lines contains two space-separated integers L and R $(0 \leq L \leq R \leq 10^{18})$.

Output

For each test case, output a single line containing the XOR of all integers from L to R, inclusive:

$$[L \oplus (L+1) \oplus (L+2) \oplus \cdots \oplus (R-1) \oplus R]$$

Example

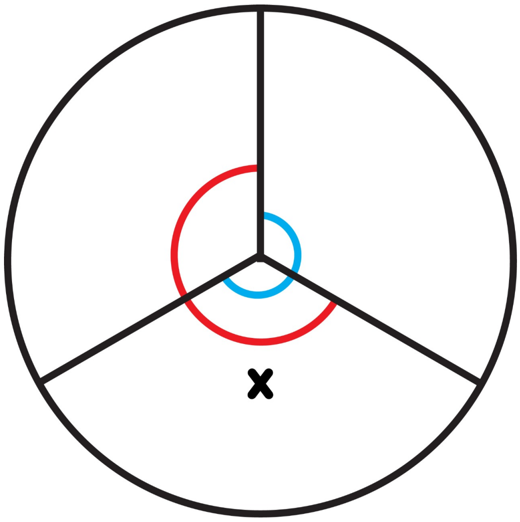
standard input	standard output
2 2 4 5 9	5 5

Problem C. The Hidden Sector

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Pink`

You have a circle that contains two arcs. These arcs intersect with an angle x.

Given the degrees of the two arcs, a (the red one) and b (the blue one), your task is to find the value of x, the measure of the angle where both arcs overlap.



Input

The first line contains a single integer T ($1 \le T \le 10^4$) — the number of test cases.

Each of the next T lines contains two integers a and b ($0 \le a, b \le 360$) — the degrees of the two arcs.

It is guaranteed that $360 \le a + b \le 720$.

Output

For each test case, print a single integer — the degrees where both arcs overlap.

Example

standard input	standard output
1 260 180	80

Problem D. caesar string

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Blue`

You are given a string S of lowercase Latin letters and q queries. Each query consists of two integers l and r, asking you to do the following operations on the substring $S[l \ldots r]$:

1. Find the most frequent character in the substring $S[l \ldots r]$.
2. Perform a Caesar shift on the substring $S[l \ldots r]$ by shifting each character forward by the frequency of the most frequent character. If a character reaches past `z`, it wraps around back to `a`.

After all queries are applied, you must output the final version of the string S.

Input

The first line contains a single integer T ($1 \le T \le 10^4$) — the number of test cases.

Each test case starts with a line containing two integers n and q ($1 \le n, q \le 10^5$) — the length of the string and the number of queries.

The next line contains a string S of length n, consisting only of lowercase Latin letters.

Each of the next q lines contains two integers l and r ($1 \le l \le r \le n$), describing a query.

It is guaranteed that the sum of $n + q$ across all tests does not exceed $2 \cdot 10^5$.

Output

For each test case, output a single line containing the final version of the string S after all queries are applied.

Example

standard input	standard output
2 5 2 ababa 1 3 2 5 3 1 xyz 1 3	cedcb yza

Problem E. Candy Distribution

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Orange`

There are n identical candies and n distinct boxes, numbered from 1 to n. The candies must be distributed according to the following rules:

- Each box can hold at most 2 candies.
- If any box contains 2 candies, then all preceding boxes (i.e., boxes 1 to k, for some k) must also contain 2 candies.

Determine the total number of valid ways to distribute the candies while satisfying these constraints. The answer should be computed modulo $10^9 + 7$.

Input

The first line contains an integer t ($1 \le t \le 10^5$), the number of test cases.

Each test case consists of a single integer n ($1 \le n \le 10^{18}$).

Output

For each test case, print a single integer — the number of valid ways to distribute the candies into the boxes modulo $10^9 + 7$.

Example

standard input	standard output
2 6 210	13 323454292

Problem F. Limit It

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Purple`

Given a binary tree of n nodes rooted at node 1, and each node i has a value a_i on it.

Recall that a binary tree is such a tree that no vertex has more than 2 children.

You are given q queries; in each query, you are given two integers L, R. Your task is to find the maximum possible sum of a group of nodes such that:

- The size of the group is at least L and at most R.
- If you remove all of the nodes from the tree except the nodes that are in the group, the resultant graph will still be a single connected tree.

Input

The first line contains one integer n ($1 \le n \le 10^4$), the number of nodes in the binary tree.

The second line contains n space-separated integers a_i ($-10^9 \le a_i \le 10^9$), the value on each node.

The next $n-1$ lines contain two space-separated integers u and v ($1 \le u,\ v \le n$), the edge between node u and node v.

The next line contains one integer q ($1 \le q \le 5050$), the number of queries.

The next q lines contain two space-separated integers $L,\ R$ ($1 \le L \le R \le min(n, 100)$).

Output

For each query, print the maximum sum of a connected group of nodes, where the size of the group is at least L and at most R.

Example

standard input	standard output
6 1 10 5 -2 -3 4 1 2 2 6 2 4 1 5 5 3 3 1 3 2 5 6 6	15 17 15

Problem G. The Great MEX

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Green`

You are given a positive integer n. Consider all possible permutations of the integers from 0 to $n-1$ (inclusive), of which there are $n!$ in total.

For a permutation p of length n, define the function $f(p)$ as the number of subarrays $p[i \ldots j]$ (where $0 \le i \le j < n$) such that $\mathrm{MEX}(p[i \ldots j]) = j - i + 1$. In other words, $f(p)$ is the number of subarrays where the minimum excluded value (MEX) of the subarray is equal to the length of that subarray.

Your task is to compute the number of permutations of length n that maximize the value of $f(p)$ among all possible permutations of length n.

Since the answer can be large, print it modulo $10^9 + 7$.

Input

The first line contains a single integer T ($1 \le T \le 10^5$) — the number of test cases.

Each of the next T lines contains a single integer n ($1 \le n \le 10^9$) — the length of the permutation.

Output

For each test case, print a single line containing one integer — the number of permutations that achieve the **maximum** possible value of $f(p)$, modulo $10^9 + 7$.

Example

standard input	standard output
3 1 2 1000000000	1 2 570312504

Note

— The MEX (minimum excluded value) of a multiset of non-negative integers is the smallest non-negative integer not present in the multiset.

— A subarray is a contiguous segment of the permutation.

Problem H. Greedy or Generous

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Light Blue`

You are given a sequence of n balls, each with an integer value a_i.

There are two players:

- Player 1 can only take balls from the front of the sequence.
- Player 2 can only take balls from the end of the sequence.

Players take turns, and Player 1 goes first. On each player's turn, they have two options:

- **Take** the current ball's value (front for Player 1, end for Player 2), add it to their own total, and pass the turn to the other player.
- **Give** the current ball's value to the other player (the opponent adds it to their total), and retain the turn.

Both players play optimally: Player 1 aims to maximize the profit, and Player 2 aims to minimize the profit.

The profit is defined as

Profit = (total value collected by Player 1) − (total value collected by Player 2).

Your task is to determine the maximum profit Player 1 can achieve if both players play optimally.

Input

The first line contains a single integer n ($1 \leq n \leq 2000$), the number of balls.

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($-10^9 \leq a_i \leq 10^9$), where a_i is the value written on the i-th ball.

Output

The output should consist of a single integer — the maximum profit Player 1 can achieve if both players play optimally.

Examples

standard input	standard output
4 1 2 3 4	0
5 3 9 1 2 7	4

Problem I. Mirror

Input file: `standard input`
Output file: `standard output`
Balloon Color: `Yellow`

Imagine a person named **Number** standing in front of a mirror. The mirror doesn't speak — it simply reflects. Number looks closely and wonders: "If what's on the left moves to the right, and what's on the right moves to the left… what will I look like now?"And just like that, he figures out the new version of himself — his mirrored self.

"Can you tell what **Number** saw in the mirror?"

Input

The input consists of a single integer $n(10 \leq n \leq 99)$ — a two-digit number.

Output

Print the number formed by swapping the digits.

Examples

standard input	standard output
97	79
10	01

Problem J. Binary Strings

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Red`

You are given an array S consisting of n binary strings. Your task is to find the index of the greatest binary string in the array.

A binary string a is considered greater than a binary string b if the decimal value of a is greater than the decimal value of b.

If there are multiple binary strings with the maximum value, print the index of the first (smallest index) occurrence.

Input

The first line contains a single integer n $(1 \leq n \leq 10^6)$ — the number of binary strings in the array.

The following n lines each contain a binary string S_i $(1 \leq |S_i| \leq 10^6)$.

- It is guaranteed that there are no leading zeros in the binary strings.

- The sum of the lengths of all binary strings in the array will not exceed 10^6.

Output

Print a single integer — the index of the greatest binary string in the array.

Example

standard input	standard output
4 11 101 1000 111	3

Problem K. Stations

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Gold`

In a futuristic city, there are n power stations connected in a network, each with an "energy code"(an integer). Connected stations have a "compatibility factor"(a positive integer), and the network is represented as an undirected graph where each edge has its compatibility factor.

The goal is to divide the stations into two groups, a and b, such that:

Every edge between two stations in the same group (both in a or both in b) must have a compatibility factor divisible by 3, i.e., for all edges $e = (u, v)$ where $u, v \in a$ or $u, v \in b$, $w_e \mod 3 = 0$.

The sum of the energy codes in group a minus the sum in group b must be a prime number greater than 5, i.e., $\text{sum}_a - \text{sum}_b$ is prime and $\text{sum}_a - \text{sum}_b > 5$.

Both groups a and b must contain at least one station.

You need to determine if such a division is possible.

Input

The first line contains two integers n $(1 \le n \le 20)$ and m $(0 \le m \le n(n-1)/2)$, the number of stations and edges.

The second line contains n integers, the energy codes for stations 1 to n, each in the range $[-10^6, 10^6]$.

The next m lines each contain three integers u, v, and w $(1 \le u, v \le n,\ u \ne v,\ 1 \le w \le 10^6)$, indicating an edge between stations u and v with a compatibility factor w.

Output

Output "YES"if a valid division exists, and "NO"otherwise.

Examples

standard input	standard output
3 3 1 2 3 1 2 4 2 3 5 1 3 10	NO
4 4 -1 2 5 7 1 2 6 3 4 9 1 3 4 2 4 7	YES