A Few IUPC Problems

Ruhan Habib, A.J.M Istiaque

May 14, 2025

BRAC University

Equal

Problem Statement

Given four positive integers a, b, c, and d. You will have to find whether

$$\frac{a}{b} = \frac{c}{d}$$

Constraints

$$0 < a, b, c, d \leq 18 \times 10^{18}$$

Sample Input

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

Sample Output

```
Not Equal
Equal
```

Equal

Problem Statement

Given four positive integers a, b, c, and d. You will have to find whether

$$\frac{a}{b} = \frac{c}{d}$$

Constraints

$$0 < a, b, c, d \leq 18 \times 10^{18}$$

Sample Input

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

Sample Output

```
Not Equal
Equal
```

Equal

Problem Statement

Given four positive integers a, b, c, and d. You will have to find whether

$$\frac{a}{b} = \frac{c}{d}$$

Constraints

$$0 < a, b, c, d \leq 18 \times 10^{18}$$

Sample Input

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

Sample Output

```
Not Equal
Equal
```

Equal

Problem Statement

Given four positive integers *a*, *b*, *c*, and *d*. You will have to find whether

$$\frac{a}{b} = \frac{c}{d}$$

Constraints

$$0 < a, b, c, d \leq 18 \times 10^{18}$$

Sample Input

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

Sample Output

```
Not Equal
Equal
```

Equal

Point Table

Problem Statement

Given the total points of three football teams *A*, *B*, and *C* after each played exactly two matches against the others, determine if the point table could be valid based on standard football scoring rules:

- Win = 3 points
- Draw = 1 point
- Loss = 0 points

Constraints

⇒ T test cases ($1 \leq T \leq 350$),

⇒ Each test case will have three integers P_A, P_B, P_C such that $0 \leq P_A, P_B, P_C \leq 6$, denoting their points in a single line.

Point Table

Problem Statement

Given the total points of three football teams *A*, *B*, and *C* after each played exactly two matches against the others, determine if the point table could be valid based on standard football scoring rules:

- Win = 3 points
- Draw = 1 point
- Loss = 0 points

Constraints

⇒ T test cases ($1 \leq T \leq 350$),

⇒ Each test case will have three integers P_A, P_B, P_C such that $0 \leq P_A, P_B, P_C \leq 6$, denoting their points in a single line.

Point Table

Sample Input

```
3
6  1  1
3  3  3
6  6  6
```

Sample Output

```
Yes
Yes
No
```

Point Table

Qwiksort

Problem Statement

You're given an array of size $2n$ containing all numbers from 1 to $2n$ exactly once. You can perform a *Qwiksort* operation: choose any contiguous subarray of size n and sort it in place.

You may do this operation up to **10** times to sort the entire array in increasing order.

N.B: You do not need to minimize the number of operations. It is guaranteed that it's possible to sort the arrays with at most 10 *Qwiksort* operations.

Constraints

- $1 \leq T \leq 40000$
- $2 \leq n \leq 1000$
- Sum of n over all test cases does not exceed 2×10^5

Qwiksort

Problem Statement

You're given an array of size $2n$ containing all numbers from 1 to $2n$ exactly once. You can perform a *Qwiksort* operation: choose any contiguous subarray of size n and sort it in place.

You may do this operation up to **10** times to sort the entire array in increasing order.

N.B: You do not need to minimize the number of operations. It is guaranteed that it's possible to sort the arrays with at most 10 *Qwiksort* operations.

Constraints

- $1 \leq T \leq 40000$
- $2 \leq n \leq 1000$
- Sum of n over all test cases does not exceed 2×10^5

Qwiksort

Output Format

For each test case, print the number of *Qwiksort* operations $0 \le k \le 10$, followed by k lines each with two integers l and r (1-based indices), representing a sorted subarray $[l, r]$ of size n.

Sample Input

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

Sample Output

```
2
6 10
2 6
0
```

Qwiksort

Output Format

For each test case, print the number of *Qwiksort* operations $0 \leq k \leq 10$, followed by k lines each with two integers l and r (1-based indices), representing a sorted subarray $[l, r]$ of size n.

Sample Input

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

Sample Output

```
2
6   10
2   6
0
```

Qwiksort

Boat-Flix

Problem Statement

There are N swimmers and K boats on one side of a river of width D. Swimmers can swim at speed X, and boats move at speed Y (only when operated by a swimmer).

Each swimmer can swim, use a boat, or both. Boats can be reused but carry only one swimmer at a time.

All swimmers and boats start together. Find the minimum time for all swimmers to reach the opposite bank.

Constraints

- $1 \leq \mathsf{T} \leq 100$
- $1 \leq \mathsf{N}, \mathsf{K} \leq 10^6$
- $1 \leq \mathsf{D}, \mathsf{X}, \mathsf{Y} \leq 10^9$

Boat-Flix

Problem Statement

There are N swimmers and K boats on one side of a river of width D. Swimmers can swim at speed X, and boats move at speed Y (only when operated by a swimmer).

Each swimmer can swim, use a boat, or both. Boats can be reused but carry only one swimmer at a time.

All swimmers and boats start together. Find the minimum time for all swimmers to reach the opposite bank.

Constraints

- $1 \leq \mathsf{T} \leq 100$
- $1 \leq \mathsf{N}, \mathsf{K} \leq 10^6$
- $1 \leq \mathsf{D}, \mathsf{X}, \mathsf{Y} \leq 10^9$

Boat-Flix

Sample Input

```
2
3   2
100   10   15
5   6
120   10   15
```

Sample Output

```
7.7777777778
8.0000000000
```

Boat-Flix

Memoir of Sifat

Problem Statement

Sifat has a hidden binary string of length N $(1 \leq N \leq 1000)$. You are allowed to make at most 1024 queries to identify the exact string.

In each query, you submit a non-empty binary string S (length $\leq N$). Sifat will respond with:

- `"Correct"` - if S matches the hidden string exactly.
- `"Yes"` - if S is a subsequence of the hidden string.
- `"No"` - otherwise.

A subsequence is a sequence derived by deleting zero or more characters without changing the order.

Your goal is to identify the hidden string in at most **1024** queries.

Memoir of Sifat

Sample Interaction

```
>4
<010
>Yes
<0101
>No
<011
>Yes
<0110
>Correct
```

Memoir of Sifat

The Beast

Problem Statement

There are N shooters, where the i^{th} shooter fires a bullet every i minutes. The beast will be destroyed exactly when K bullets have been fired in total. Determine the exact minute when the beast is destroyed.

Constraints

- $1 \leq \mathsf{T} \leq 1000$
- $1 \leq \mathsf{N}, \mathsf{K} \leq 10^6$

The Beast

Problem Statement

There are N shooters, where the i^{th} shooter fires a bullet every i minutes. The beast will be destroyed exactly when K bullets have been fired in total. Determine the exact minute when the beast is destroyed.

Constraints

- $1 \leq \mathsf{T} \leq 1000$
- $1 \leq \mathsf{N}, \mathsf{K} \leq 10^6$

The Beast

Sample Input

```
4
2   10
10   10
5   10
5   11
```

Sample Output

```
7
5
5
6
```

The Beast

LCM Factorization

Problem Statement

For a positive integer x, define $f(x)$ as the sum of all distinct prime factors of x.

- For example: $f(60) = f(2^2 \times 3 \times 5) = 2 + 3 + 5 = 10$, and $f(1) = 0$.

You are given a sequence of n positive integers $a_1, a_2, \ldots, a_n$, and an integer k. Your task is to compute the sum of $f(\mathrm{LCM}(S))$ over all possible subsequences S of length k from the array.

Since the answer can be large, output it modulo 998244353.

Constraints

- $1 \leq T \leq 10000$
- $1 \leq k \leq n \leq 3 \cdot 10^5$

LCM Factorization

Problem Statement

For a positive integer x, define $f(x)$ as the sum of all distinct prime factors of x.

- For example: $f(60) = f(2^2 \times 3 \times 5) = 2 + 3 + 5 = 10$, and $f(1) = 0$.

You are given a sequence of n positive integers $a_1, a_2, \ldots, a_n$, and an integer k. Your task is to compute the sum of $f(\mathrm{LCM}(S))$ over all possible subsequences S of length k from the array.

Since the answer can be large, output it modulo 998244353.

Constraints

- $1 \leq T \leq 10000$
- $1 \leq k \leq n \leq 3 \cdot 10^5$

LCM Factorization

Sample Input

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

Sample Output

```
19
2
0
```

LCM Factorization

Four More Problems

- An Interesting Problem
- Litmus Test
- Distinct of Distincts
- The Last Bit of Us