ESO207: Data Structures and Algorithms

Programming Assignment 2

Due Date: 24th September 2025, 11:59 PM

Total Number of Pages: 5 Total Points 100

Note :

- The questions have to be answered through a contest in Hackerrank. The contest has 3 challenges, each corresponding to a question. You have to submit your code through the contest.
- Contest Link: `https://www.hackerrank.com/eso207-au25-pa2`
- You do not need to upload anything on Hello IITK.
- Your codes will be checked for possible plagiarism of any sorts. If we find such cases, then we will possibly award an F grade.
- Allowed Languages for challenge code submission : C, C++
- Allowed libraries : `stdio.h` for C and `<bits/stdc++.h>` for C++

Question 1. (20 points) **Play While You Can**

You manage a game centre. There are n players. The i-th player arrives at time a_i and requires p_i minutes of continuous play to feel satisfied.

To prevent overcrowding, whenever a new player arrives at time a_i, all the players who came before time $a_i - d$ have to leave the centre.

Additionally, the servers shut down completely at time T.

A player is considered **satisfied** if they can finish their required p_i minutes of play before being forced to stop (either due to overcrowding or the shutdown time T).

Input

The first line contains three integers n, d, T.

The next n lines contain two integers a_i, p_i.

Constraints

- $1 \leq n \leq 2 \cdot 10^5$
- $1 \leq d \leq 10^9$
- $1 \leq T \leq 10^9$
- $1 \leq a_i \leq 10^9$
- $1 \leq p_i \leq 10^9$

Output

Print a single integer, the number of satisfied players.

Sample Input

```
5 5 30
1 10
3 5
7 10
15 5
27 10
```

Sample Output

```
2
```

Explanation

- $t = 1$: Player 1 arrives (needs until 11).
- $t = 3$: Player 2 arrives (needs until 8). No removals.
- $t = 7$: Player 3 arrives. Remove arrivals $< 2 \Rightarrow$ Player 1 removed (unsatisfied). Player 2 still active.
- $t = 8$: Player 2 completes $\Rightarrow$ satisfied.
- $t = 15$: Player 4 arrives. Remove arrivals $< 10 \Rightarrow$ Player 3 removed (unsatisfied). Player 4 needs until 20.
- $t = 20$: Player 4 completes $\Rightarrow$ satisfied.
- $t = 27$: Player 5 arrives. Remove arrivals $< 22 \Rightarrow$ no removals. Player 5 needs until 37, but shutdown $T = 30$ occurs first $\Rightarrow$ unsatisfied.
- $t = 30$: Shutdown.

Question 2. (40 points) **Light it Up!**

You are given a square grid of size $n \times n$. There are t lighthouses placed on some cells of the grid. The i-th lighthouse is located at (x_i, y_i).

A lighthouse spreads light in a square area of side $2r + 1$ centered at its position.

Formally, a lighthouse at (x, y) lights up all cells (u, v) such that

$$|u - x| \leq r \quad \text{and} \quad |v - y| \leq r.$$

Find the minimum r such that **all cells of the grid** are lit up by at least one lighthouse.

Input

The first line contains a single integer t, the number of test cases.

Each test case begins with two integers n and k, the size of the grid and the number of lighthouses.

The next k lines each contain two integers x_i, y_i, the position of the i-th lighthouse.

Constraints

- $1 \leq t \leq 10$
- $1 \leq n \leq 1000$
- $1 \leq k \leq n^2$
- $1 \leq x_i, y_i \leq n$

Output

Print a single integer, the minimum r such that the entire grid is lit.

Sample Input

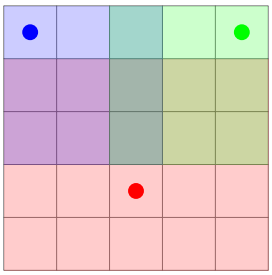
```
1
5 3
1 1
1 5
4 3
```

Sample Output

```
2
```

Explanation

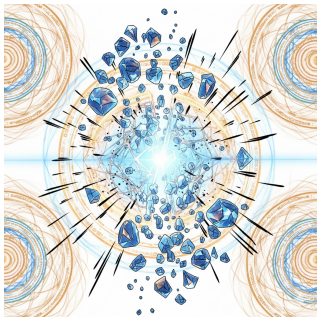
For $r = 2$:



- Lighthouse at $(1, 1)$ covers rows $[1, 3]$, cols $[1, 3]$.
- Lighthouse at $(1, 5)$ covers rows $[1, 3]$, cols $[3, 5]$.
- Lighthouse at $(4, 3)$ covers rows $[2, 5]$, cols $[1, 5]$.

Together, they cover every cell of the 5×5 grid. Smaller r fails to cover all corners.

Question 3. (40 points) **Doctor Strange and the Crystal of Futures**



In *Avengers: Infinity War*, Doctor Strange looks into the future and sees 14,000,605 possible realities, but only one where the Avengers can defeat Thanos.

Now, he encounters a magical crystal of power, represented as an array a of length n, and wishes to determine whether a particular reality (energy product) exists.

At each step of his foresight, Strange computes

$$mid = \left\lfloor \frac{\max(a) + \min(a)}{2} \right\rfloor$$

and the crystal fractures into two realities:

- **Left reality**: all elements $\leq mid$, preserving order,
- **Right reality**: all elements $> mid$, preserving order.

He must choose exactly one reality to keep, discarding the other. Along this multiversal branching, each reality has an **energy product** equal to the product of its elements.

For each query p_i, Doctor Strange must decide:

Does there exist a timeline in which the crystal's energy product equals p_i modulo 998244353?

Input

The first line contains a single integer t, the number of test cases.

Each test case begins with two integers n and q, the number of elements in the crystal and the number of queries.

The second line contains n integers $a_1, a_2, \ldots, a_n$, the crystal's energy distribution.

The next q lines each contain a single integer p_i, the target energy product for the i-th query.

Constraints

- $1 \leq t \leq 100$
- $1 \leq n, q \leq 10^5$
- $1 \leq a_i \leq 10^6$
- $1 \leq p_i \leq 10^9$

It is guaranteed that $\sum n + \sum q \leq 2 \cdot 10^5$.

Output

For each query, print `Yes` if such a reality exists, otherwise print `No`.

Sample Input

```
1
5 4
1 2 3 4 5
2
10
20
50
```

Sample Output

```
Yes
No
Yes
No
```

Explanation

- **Query:** 2 — `Yes`.
  Steps:

$$[1,2,3,4,5] \xrightarrow{mid=3} \text{choose Left } [1,2,3],$$
$$[1,2,3] \xrightarrow{mid=\lfloor(3+1)/2\rfloor=2} \text{choose Left } [1,2],$$
$$[1,2] \xrightarrow{mid=\lfloor(2+1)/2\rfloor=1} \text{choose Right } [2] \quad (\text{product} = 2).$$

- **Query:** 10 — `No`.
  It can be shown this cannot be achieved.
- **Query:** 20 — `Yes`.
  Steps:

$$[1,2,3,4,5] \xrightarrow{mid=3} \text{choose Right } [4,5] \quad (\text{product } 4 \cdot 5 = 20).$$

- **Query:** 50 — `No`.
  It can be shown this cannot be achieved.