**Code Jam to I/O for Women 2022 - Code Jam to I/O for Women**

# Ingredient Optimization

## Problem

Hathai is proud that her catering service provides the freshest food in town. To accomplish that, she gets fresh ingredients with no preservatives delivered constantly. This brings about the challenge of preventing the ingredients from spoiling. Her current special uses exactly $\mathbf{U}$ leaves of Thai basil, that need special care.

Hathai has the schedule of the deliveries of Thai basil. The $i$-th delivery comes at the beginning of time $\mathbf{M_i}$ (in minutes since opening time), and brings exactly $\mathbf{L_i}$ leaves of Thai basil that can be stored for at most $\mathbf{E_i}$ minutes since arriving. Hathai has orders to prepare her specialty dish at times $\mathbf{O_1}, \mathbf{O_2}, \dots, \mathbf{O_N}$. Order $i$ can only be fulfilled if she has $\mathbf{U}$ unspoiled leaves of Thai basil at time $\mathbf{O_i}$. Note that if leaves would spoil at the same time an order comes in, those leaves cannot be used to fulfill that order. If an order is fulfilled, $\mathbf{U}$ of the leaves available have to be used and cannot be used for future orders. Once Hathai gets an order that she cannot fulfill, all of the remaining orders will also be canceled because she needs to close her kitchen and think about how to improve the fulfillment schedule.

For example, suppose Hathai's schedule has the following $4$ deliveries:

- Delivery time: $1$. Amount: $10$. Time remaining until spoiled: $2$.
- Delivery time: $3$. Amount: $4$. Time remaining until spoiled: $2$.
- Delivery time: $5$. Amount: $1$. Time remaining until spoiled: $4$.
- Delivery time: $10$. Amount: $6$. Time remaining until spoiled: $3$.

And also suppose she has $4$ orders placed at times $3$, $4$, $6$ and $10$. Each order requires using $\mathbf{U} = 2$ leaves in this example.

The first delivery become spoiled at time $3$, so it cannot be used on any order. Then the first order and the second order at time $3$ and time $4$ can be fulfilled and use up the $4$ leaves from the second delivery. For the third order at time $6$, there is only $1$ leaf in the storage, so Hathai cannot fulfill this order. Note that although there is a delivery at time $10$, she still cannot fulfill the fourth order at time $10$ because she has already closed her kitchen. In this example, Hathai managed to fulfill $2$ orders in total.

Given the delivery and order schedules, find out the maximum number of orders Hathai can fulfill if she optimizes the use of the Thai basil leaves.

## Input

The first line of the input gives the number of test cases, $\mathbf{T}$. $\mathbf{T}$ test cases follow. Each test case starts with a line containing three integers $\mathbf{D}$, $\mathbf{N}$, and $\mathbf{U}$: the number of deliveries, the number of orders and the amount of leaves needed for each order, respectively. Then, $\mathbf{D}$ lines follow. The $i$-th of these lines contains three integers $\mathbf{M_i}$, $\mathbf{L_i}$, and $\mathbf{E_i}$: the time of the delivery in minutes since opening time, the amount of Thai basil leaves delivered, and the number of minutes those leaves can be stored and remain fresh, respectively, of the $i$-th delivery. Then, the last line contains $\mathbf{N}$ integers $\mathbf{O_1}, \mathbf{O_2}, \dots, \mathbf{O_N}$, where $\mathbf{O_j}$ is the time, in minutes since opening time, at which the $j$-th order must be prepared.

## Output

For each test case, output one line containing `Case #`$x$`:` $y$, where $x$ is the test case number (starting from 1) and $y$ is an integer representing the maximum number of orders Hathai can fulfill.

## Limits

Time limit: 5 seconds.
Memory limit: 1 GB.
$1 \leq \mathbf{T} \leq 100.$
$1 \leq \mathbf{D} \leq 100.$
$1 \leq \mathbf{N} \leq 100.$
$1 \leq \mathbf{U} \leq 100.$
$1 \leq \mathbf{M_i} \leq 10^9$, for all $i$.
$\mathbf{M_i} < \mathbf{M_{i+1}}$, for all $i$. (Deliveries are given in increasing order of time.)
$1 \leq \mathbf{L_i} \leq 100$, for all $i$.
$1 \leq \mathbf{O_j} \leq 10^9$, for all $j$.
$\mathbf{O_j} < \mathbf{O_{j+1}}$, for all $j$. (Orders are given in increasing order of time.)

### Test Set 1 (Visible Verdict)

$\mathbf{E_i} = 10^9$, for all $i$. (No Thai basil leaf expires before an order comes.)

### Test Set 2 (Hidden Verdict)

$1 \leq \mathbf{E_i} \leq 10^9$, for all $i$.

## Sample

*Note: there are additional samples that are not run on submissions down below.*

Sample Input

```
2
2 4 5
20 8 1000000000
60 4 1000000000
10 30 50 70
3 5 5
20 8 1000000000
50 3 1000000000
60 100 1000000000
30 50 59 70 90
```

Sample Output

```
Case #1: 0
Case #2: 2
```

In Sample Case #1, Hathai cannot fulfill the first order, because it came too early, before any deliveries brought in Thai leaves. So she has to close her kitchen immediately without fulfilling any order.

In Sample Case #2, Hathai can fulfill the first order, and her second delivery arrives just in time to help her fulfill the second order. However, the huge third delivery does not get to her in time to help her with the third order, so that one goes unfulfilled and so do the remaining ones.

## Additional Sample - Test Set 2

*The following additional sample fits the limits of Test Set 2. It will not be run against your submitted solutions.*

Sample Input

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

Sample Output

```
Case #1: 2
```

This additional sample is the one explained in the problem statement.