Challenge 6 - Button Hero

« PREV NEXT »

You're a game developer and your last project, Button Hero, is almost done!

Button Hero is a rhythm game where the player has to time key strokes to the rhythm of a song. While in other games players imitate an instrument like a guitar, Button Hero has just one button that has to be pressed and released at the right time to match the notes.

But it's not as easy as it sounds. While the visual cues of other games are straight-forward, Button Hero can get pretty crazy:

- There's a single horizontal lane, which starts at position 0, and contains all the notes.
- A note is represented by a colored box of variable length. Each note has points assigned to it that you earn when you hit the note. To hit a note the button has to be pressed exactly when the note starts crossing the start of the lane and released exactly when the note finishes crossing it.
- Each note starts at some position in the lane and moves towards the start of the lane. Notes can have different speeds and can thus overlap each other at different moments of time.
- The notes always reach the start of the lane (notes cannot have a speed of zero)
- The notes arrangements can get pretty crazy. It might not be possible to hit all the notes.

The only thing left to implement is the leaderboard system. You are worried that some people might try to cheat, so you thought of just banning those who submit an impossible score. But to do that you need to know the maximum possible score for every song.

Input

The first line has an integer **C**, which is the number of cases for our problem.
Each case starts with a line with an integer **N**, which is the number of notes.
N lines follow, each with four integers **X**, **L**, **S**, **P**, indicating the initial position, length, speed and score of the note.

Output

There should be a line for each case starting with "Case #x: " followed by the maximum score possible. Every line is followed by a new line character.

Examples

```
Case 1:

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

```
Case 2:

2
2 2 2 1
1 1 1 1
```

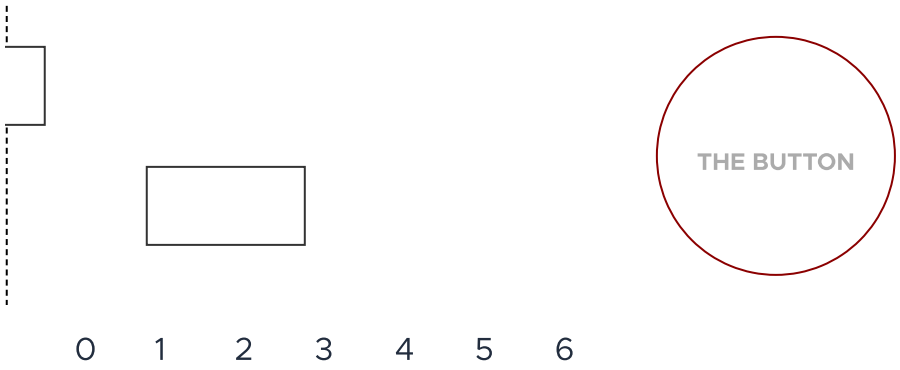
```
Case 3:

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

```
Case 4:

5
18 2 2 5
1 2 1 1
16 10 2 3
8 10 2 4
27 6 3 5
```

The answer for case 1 is 3 because you can hit both notes. Notice that even though they start at the same position, they reach the start of the lane separately.
The answer for case 2 is 2 because you can hit both notes at the same time.
The answer for case 3 is 2, because you can only hit one of the notes and the second one gives you the best score (note that you cannot hit both the first and the second note since you can't press and release the button at the same time).
The answer for case 4 is 6.



Graphical simulation for case 1

Limits

- $1 \leq \mathbf{C} \leq 100$
- $1 \leq \mathbf{N} \leq 10000$
- $1 \leq \mathbf{L}, \mathbf{S}, \mathbf{P} \leq 2^{10}$
- $1 \leq \mathbf{X} \leq 2^{30}$
- **X** and **L** are divisible by **S**

Sample Input

```
4
2
4 2 1 1
4 2 2 2
2
2 2 2 1
1 1 1 1
3
2 1 1 1
3 1 1 2
2 2 1 1
5
18 2 2 5
1 2 1 1
16 10 2 3
8 10 2 4
27 6 3 5
```

Sample Output

```
Case #1: 3
Case #2: 2
Case #3: 2
Case #4: 6
```

Test your code

You can test your program against both the input provided in the test phase and the input provided in the submit phase. A nice output will tell you if your program got the right solution or not. You can try as many times as you want to. Be careful with extra whitespaces, the output should be exactly as described.

Test your program against the input provided in the test phase

Download test input

Program output:

Choose file No file chosen

Submit test output