



Burger Happiness

by gdisastery

Problem

Submissions

Leaderboard

Discussions

Editorial

In Burger Town new burger restaurants will be opened! Concretely, N restaurants will open in N days, while restaurant i will be opened on day i and will be located at X_i . The town should be imagined as an one dimensional line in which every object's location can be described by the x coordinate.

Tim has just recently arrived the town after a very bad result in a programming contest. Thus he wants to cheer himself up by starting a trip to try out some new burgers.

Every burger restaurant i is associated with two integers A_i and B_i . If Tim eats a burger from i , then his happiness will increase by A_i , which can also be negative, depending on the deliciousness of the burger. On the other hand, if Tim looks through the window of an opened restaurant i , from which he will *not* eat a burger, then his happiness decreases by B_i , since Tim gets sad by only seeing the burgers.

Tim's journey can start from any day d at the burger restaurant d and eats a burger from there. On each subsequent day $n > d$, Tim has the following options:

- Stay at the previous restaurant p .
- Or go to the new restaurant n to eat a burger from there.

If he decides for the latter option, then on the path from p to n he will look through all the windows that are on his path and maybe lose some happiness. Concretely, if $X_p < X_n$, then he will look through the window of every *opened* restaurant i , having $X_p \leq X_i < X_n$. Similar for the case $X_n < X_p$.

Since Tim is a very good friend of yours you should help him finding a trip that will maximize his happiness. If he should stay at home since no trip would cheer him up, then print 0 .

Note: Tim's happiness is 0 at the beginning of the trip and is allowed to be negative throughout the time.

Input Format

N will be given on the first line, then N lines will follow, describing the restaurants numbered from 1 to N accordingly. Restaurant i will be described by X_i , A_i and B_i separated by a single space.

Output Format

Output the maximum happiness on one line.

Constraints

- $1 \leq N \leq 10^5$
- $|A_i| \leq 10^6$
- $0 \leq B_i \leq 10^6$
- $0 \leq X_i \leq 10^9$ and no two restaurants will have the same X coordinates.

Sample Input

```
3
2 -5 1
```

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

Sample Output

8

Sample Input

```
4
4 10 0
1 -5 0
3 0 10
2 10 0
```

Sample Output

15

Sample Input

```
3
1 -1 0
2 -2 0
3 -3 0
```

Sample Output

0

First testcase: His trip starts on day 2 at restaurant 2 located at $X_2 = 1$. He gains $A_2 = 5$ happiness points there by eating a burger. On the next day he goes from restaurant 2 to 3, but will look through the window of restaurant 2 and 1. Therefore he loses $B_2 = 1$ and $B_1 = 1$ points on the way to restaurant 3. There he eats a burger and gains another $A_3 = 5$ points. In total his happiness is equal to $5 - 1 - 1 + 5 = 8$ and this is optimal.

Second testcase: His trip starts on day 1 at restaurant 1. Then his actions on day 2, 3 and 4 will be go to restaurant 2, stay at restaurant 2 and go to restaurant 4 respectively. The happiness of this optimal trip is equal to $10 - 5 + 10 = 15$.

Third testcase: It's not worth to start the trip from any of the restaurant since he will only have negative happiness. That's why he should stay at home and 0 should be printed.



[f](#) [t](#) [in](#)
Submissions: [125](#)

Max Score: 100

Difficulty: Hard

Rate This Challenge:

☆☆☆☆☆

[More](#)Current Buffer (saved locally, editable)  

Java 7



```
1 import java.io.*;
2 import java.util.*;
3 import java.text.*;
4 import java.math.*;
5 import java.util.regex.*;
6
7 public class Solution {
8
9     public static void main(String[] args) {
```

```
10 ▾      /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */  
11      }  
12  }
```

Line: 1 Col: 1

 [Upload Code as File](#)☐ Test against custom input[Run Code](#)[Submit Code](#)

Join us on IRC at [#hackerrank](#) on freenode for hugs or bugs.

[Contest Calendar](#) | [Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Support](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#) | [Request a Feature](#)