

Problem S3: Lunch Concert

Problem Description

It's lunchtime at your school! Your N friends are all standing on a long field, as they usually do. The field can be represented as a number line, with the i th friend initially at position P_i metres along it. The i th friend is able to walk in either direction along the field at a rate of one metre per W_i seconds, and their hearing is good enough to be able to hear music up to and including D_i metres away from their position. Multiple students may occupy the same positions on the field, both initially and after walking.

You're going to hold a little concert at some position c metres along the field (where c is any integer of your choice), and text all of your friends about it. Once you do, each of them will walk along the field for the minimum amount of time such that they end up being able to hear your concert (in other words, such that each friend i ends up within D_i units of c).

You'd like to choose c such that you minimize the sum of all N of your friends' walking times. What is this minimum sum (in seconds)? Please note that the result might not fit within a 32-bit integer.

Input Specification

The first line of input contains N .

The next N lines contain three space-separated integers, P_i , W_i , and D_i ($1 \leq i \leq N$).

The following table shows how the available 15 marks are distributed.

4 marks	$1 \leq N \leq 2000$	$0 \leq P_i \leq 2000$	$1 \leq W_i \leq 1000$	$0 \leq D_i \leq 2000$
9 marks	$1 \leq N \leq 200\,000$	$0 \leq P_i \leq 1\,000\,000$	$1 \leq W_i \leq 1000$	$0 \leq D_i \leq 1\,000\,000$
2 marks	$1 \leq N \leq 200\,000$	$0 \leq P_i \leq 1\,000\,000\,000$	$1 \leq W_i \leq 1000$	$0 \leq D_i \leq 1\,000\,000\,000$

Output Specification

Output one integer which is the minimum possible sum of walking times (in seconds) for all N of your friends to be able to hear your concert.

Sample Input 1

```
1
0 1000 0
```

Output for Sample Input 1

```
0
```

Explanation of Output for Sample Input 1

If you choose $c = 0$, your single friend won't need to walk at all to hear it.

La version française figure à la suite de la version anglaise.

Sample Input 2

2
10 4 3
20 4 2

Output for Sample Input 2

20

Explanation of Output for Sample Input 2

One possible optimal choice of c is 14, which would require your first friend to walk to position 11 (taking $4 \times 1 = 4$ seconds) and your second friend to walk to position 16 (taking $4 \times 4 = 16$ seconds), for a total of 20 seconds.

Sample Input 3

3
6 8 3
1 4 1
14 5 2

Output for Sample Input 3

43

Problème S3 : Déjeuner-concert

Énoncé du problème

C'est l'heure du déjeuner à votre école ! Comme d'habitude, vos N amis sont tous regroupés sur un long terrain adjacent à votre école. Le terrain peut être représenté par une droite numérique où votre i^{e} ami est situé à une position initiale (soit P_i) en mètres le long de cette droite numérique. Votre i^{e} ami est capable de marcher dans les deux sens le long du terrain à une vitesse de 1 mètre par W_i secondes. De plus, son ouïe est suffisamment bonne pour pouvoir entendre de la musique jusqu'à D_i mètres de sa position. Plusieurs élèves peuvent occuper les mêmes positions sur le terrain, à la fois avant et après avoir marché.

Vous organisez un petit concert à une certaine position (soit c) en mètres le long du terrain (c étant un entier de votre choix) et vous envoyez un SMS à tous vos amis pour les en informer. Une fois que vous l'avez fait, chacun d'eux marchera le long du terrain pendant le minimum de temps nécessaire pour pouvoir entendre votre concert (autrement dit, de sorte que chaque ami i soit situé à D_i unités ou moins de c).

Vous souhaitez choisir c de manière à minimiser la somme de tous les temps de marche de vos N amis. Quelle est cette somme minimale (en secondes) ? Veuillez noter que le résultat peut dépasser les paramètres d'un entier 32 bits.

Précisions par rapport aux données d'entrée

La première ligne des données d'entrée contient N .

Les N prochaines lignes contiennent trois entiers P_i , W_i et D_i ($1 \leq i \leq N$), dont chacun est séparé des autres par un espace.

Le tableau suivant indique la manière dont les 15 points disponibles sont répartis.

4 points	$1 \leq N \leq 2000$	$0 \leq P_i \leq 2000$	$1 \leq W_i \leq 1000$	$0 \leq D_i \leq 2000$
9 points	$1 \leq N \leq 200\,000$	$0 \leq P_i \leq 1\,000\,000$	$1 \leq W_i \leq 1000$	$0 \leq D_i \leq 1\,000\,000$
2 points	$1 \leq N \leq 200\,000$	$0 \leq P_i \leq 1\,000\,000\,000$	$1 \leq W_i \leq 1000$	$0 \leq D_i \leq 1\,000\,000\,000$

Précisions par rapport aux données de sortie

Les données de sortie ne devraient contenir qu'un seul entier. Ce dernier représente la somme minimale possible de tous les temps de marche (en secondes) de vos N amis afin qu'ils puissent tous entendre le concert.

Données d'entrée d'un 1^{er} exemple

```
1
0 1000 0
```

Données de sortie du 1^{er} exemple

```
0
```

English version appears before the French version

Justification des données de sortie du 1^{er} exemple

Si vous choisissez $c = 0$, votre seul ami n'aura pas besoin de se déplacer pour pouvoir entendre le concert.

Données d'entrée d'un 2^e exemple

2
10 4 3
20 4 2

Données de sortie du 2^e exemple

20

Justification des données de sortie du 2^e exemple

On peut avoir 14 comme choix optimal possible de c . Ceci obligerait votre premier ami à marcher jusqu'à la position 11 (en $4 \times 1 = 4$ secondes) et votre second ami à marcher jusqu'à la position 16 (en $4 \times 4 = 16$ secondes), soit un total de 20 secondes.

Données d'entrée d'un 3^e exemple

3
6 8 3
1 4 1
14 5 2

Données de sortie du 3^e exemple

43