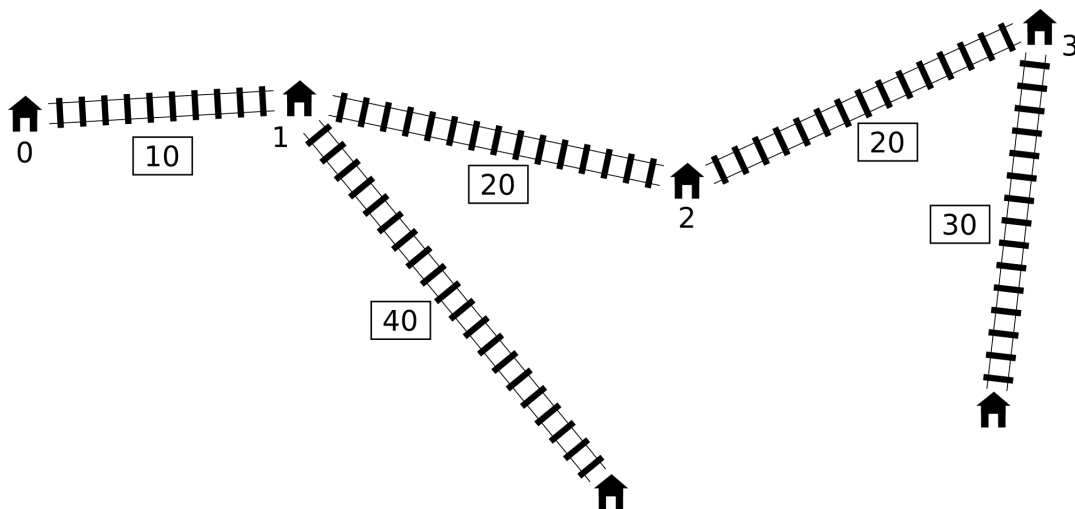


Shortcut

Павел има детско влакче. То се състои от една главна линия, по която има n гари, последователно номерирани с целите числа от 0 до $n - 1$. Гарите 0 и $n - 1$ са върху двата края на главната линия. Разстоянието между гарите i и $i + 1$ е l_i сантиметра ($0 \leq i < n - 1$).

Освен главната линия има и няколко второстепенни линии. Всяка второстепенна линия свързва гара от главната линия и нова станция, която не е на главната линия (Тези нови гари не са номерирани). Най-много една второстепенна линия може да започва от всяка гара на главната линия. Дължината на второстепенната линия, започваща от гара i е d_i сантиметра. Означаваме $d_i = 0$, за да отбележим, че не съществува второстепенна линия, започваща от гара i .



Павел планира да построи един пряк път за експресна линия между две различни гари (възможно съседни) от **главната линия**. Експресната линия ще има дължина точно c сантиметра, независимо от това, кои две гари свързва.

Всеки участък от линията, включително новата експресна линия може да се ползва в двете си посоки. *Разстоянието* между две гари е равно на по-малката дължина на път, измежду пътищата от едната гара до другата. *Диаметър* на цялата система от линии е максимумът на разстоянията между всички двойки гари. С други думи, това е най-малкото число t , такова, че разстоянието между

всеки две гари е най-много t .

Павел иска да построи експресна линия така, че диаметърът на получената система от линии да е минимален.

Детайли за реализацията

Трябва да реализирате функцията

```
int64 find_shortcut(int n, int[] l, int[] d, int c)
```

- n : брой на гарите на главната линия,
- l : разстояние между гарите от главната линия (масив с дължина $n - 1$),
- d : дължини на второстепенните линии (масив с дължина n),
- c : дължина на новата експресна линия.
- Функцията трябва да върне най-малкия възможен диаметър на цялата система от линии след добавянето на новата експресна линия.

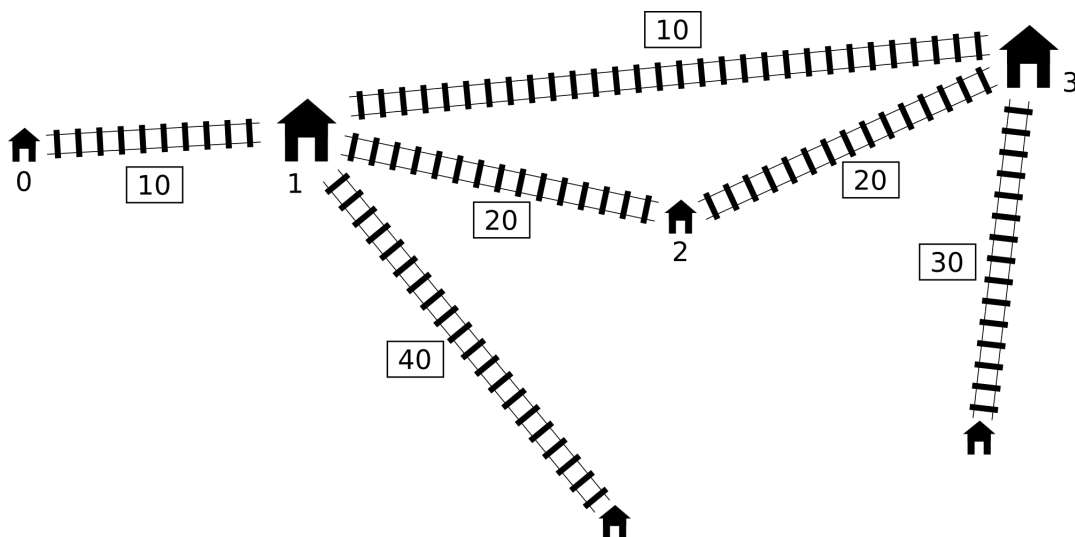
Примери

Пример 1

За системата от линии, показани по-горе, грейдерът ще направи следното извикване:

```
find_shortcut(4, [10, 20, 20], [0, 40, 0, 30], 10)
```

Оптималното решение е да се построи експресна линия, свързваща гари 1 и 3, както е показано по-долу.



Диаметърът на получената система от линии е **80** сантиметра, така че функцията трябва да върне **80**.

Пример 2

Грейдерът прави следното извикване:

```
find_shortcut(9, [10, 10, 10, 10, 10, 10, 10, 10],
```

```
[20, 0, 30, 0, 0, 40, 0, 40, 0], 30)
```

Оптимално решение е свързването на гари **2** и **7**, и в този случай диаметърът е **110**.

Пример 3

Грейдерът прави следното извикване:

```
find_shortcut(4, [2, 2, 2],  
              [1, 10, 10, 1], 1)
```

Оптимално решение е свързването на гари **2** и **3**, и в този случай диаметърът е **21**.

Пример 4

Грейдерът прави следното извикване:

```
find_shortcut(3, [1, 1],  
              [1, 1, 1], 3)
```

Свързването на кои да е две гари с експресна линия с дължина **3** не подобрява първоначалния диаметър на системата от линии, който е **4**.

Подзадачи

Във всички подзадачи $2 \leq n \leq 1\,000\,000$, $1 \leq l_i \leq 10^9$, $0 \leq d_i \leq 10^9$, $1 \leq c \leq 10^9$.

1. (9 точки) $2 \leq n \leq 10$,
2. (14 точки) $2 \leq n \leq 100$,
3. (8 точки) $2 \leq n \leq 250$,
4. (7 точки) $2 \leq n \leq 500$,
5. (33 точки) $2 \leq n \leq 3000$,
6. (22 точки) $2 \leq n \leq 100\,000$,
7. (4 точки) $2 \leq n \leq 300\,000$,
8. (3 точки) $2 \leq n \leq 1\,000\,000$.

Примерен грейдер

Примерният грейдер чете вход в следния формат:

- ред 1: чели числа n и c ,
- ред 2: чели числа l_0, l_1, \dots, l_{n-2} ,
- ред 3: чели числа d_0, d_1, \dots, d_{n-1} .