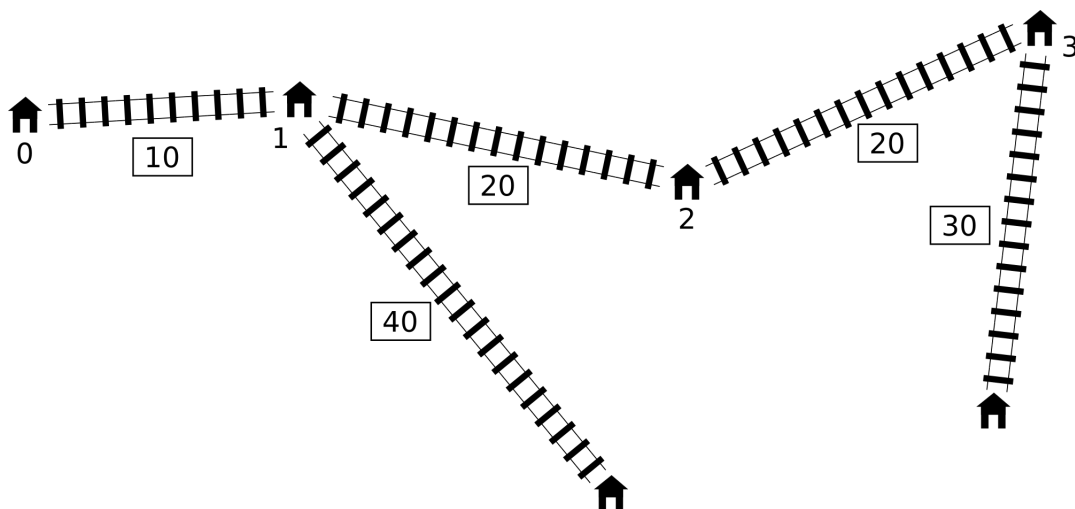


Συντομότερη Διαδρομή

Ο Παύλος έχει ένα παιχνίδι σιδηρόδρομο. Είναι πολύ απλό. Υπάρχει μία μοναδική κύρια γραμμή που αποτελείται από n σταθμούς. Οι σταθμοί αριθμούνται από 0 έως $n - 1$ στη σειρά κατά μήκος της γραμμής. Η απόσταση μεταξύ των σταθμών i και $i + 1$ είναι l_i εκατοστά ($0 \leq i < n - 1$).

Εκτός από την κύρια γραμμή μπορεί να υπάρχουν και κάποιες δευτερεύουσες γραμμές. Κάθε δευτερεύουσα γραμμή είναι μία σιδηροδρομική γραμμή μεταξύ ενός σταθμού στην κύρια γραμμή και ενός νέου σταθμού που δεν βρίσκεται στην κύρια γραμμή. (Αυτοί οι νέοι σταθμοί δεν είναι αριθμημένοι.) Το πολύ μία δευτερεύουσα γραμμή μπορεί να ξεκινά από κάθε σταθμό της κύριας γραμμής. Το μήκος της δευτερεύουσας γραμμής ξεκινώντας από τον σταθμό i είναι d_i εκατοστά.

Χρησιμοποιούμε $d_i = 0$ για να συμβολίσουμε ότι δεν υπάρχει καμία δευτερεύουσα γραμμή που ξεκινά από το σταθμό i .



Ο Παύλος τώρα σχεδιάζει να χτίσει μια συντομότερη διαδρομή (shortcut): μια ταχεία γραμμή μεταξύ των δύο (πιθανά γειτονικών) σταθμών της **κύριας γραμμής**. Η ταχεία γραμμή θα έχει μήκος ακριβώς c εκατοστά, ανεξάρτητα από το ποιους δύο σταθμούς θα συνδέει.

Κάθε τομέας του σιδηρόδρομου, συμπεριλαμβανομένης και της νέας ταχείας γραμμής, μπορεί να χρησιμοποιηθεί και στις δύο κατευθύνσεις. Η *απόσταση* μεταξύ δύο σταθμών είναι το μικρότερο μήκος μιας διαδρομής που πηγαίνει από τον ένα σταθμό στον άλλο κατά μήκος των σιδηροδρόμων. Η *διάμετρος* όλου του

σιδηροδρομικού δικτύου είναι η μέγιστη απόσταση μεταξύ όλων των ζευγών των σταθμών. Με άλλα λόγια, είναι ο μικρότερος αριθμός t , έτσι ώστε η απόσταση για κάθε ζεύγος σταθμών είναι το πολύ t .

Ο Παύλος θέλει να χτίσει την ταχεία γραμμή με τέτοιο τρόπο έτσι ώστε η διάμετρος του δικτύου που θα προκύψει να ελαχιστοποιείται.

Λεπτομέρειες υλοποίησης

Πρέπει να υλοποιήσετε τη συνάρτηση

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

- n : πλήθος των σταθμών στην κύρια γραμμή,
- l : αποστάσεις μεταξύ των σταθμών στην κύρια γραμμή (πίνακας μήκους $n - 1$),
- d : μήκη των δευτερευουσών γραμμών (πίνακας μήκους n),
- c : μήκος της νέας ταχείας γραμμής.
- η συνάρτηση πρέπει να επιστρέφει τη μικρότερη πιθανή διάμετρο του σιδηροδρομικού δικτύου μετά την προσθήκη της ταχείας γραμμής.

Παρακαλώ, χρησιμοποιείτε τα πρότυπα των αρχείων που σας δίνονται για λεπτομέρειες σχετικά με την υλοποίηση στην γλώσσα προγραμματισμού σας.

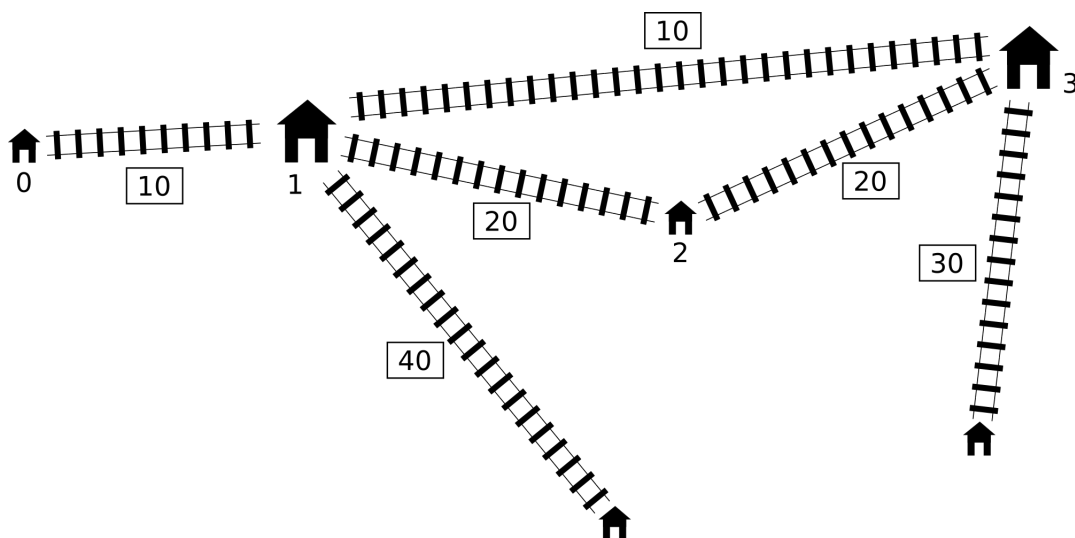
Παραδείγματα

Παράδειγμα 1

Για το παραπάνω σιδηροδρομικό δίκτυο, ο βαθμολογητής (grader) θα κάνει την παρακάτω κλήση συνάρτησης:

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

Η βέλτιστη λύση είναι να χτίσει την ταχεία γραμμή μεταξύ των σταθμών 1 και 3, όπως φαίνεται παρακάτω.



Η διάμετρος του νέου σιδηροδρομικού δικτύου είναι 80 εκατοστά, έτσι η

συνάρτηση πρέπει να επιστρέψει **80**.

Παραδείγμα 2

Ο βαθμολογητής (grader) κάνει την παρακάτω κλήση συνάρτησης:

```
find_shortcut(9, [10, 10, 10, 10, 10, 10, 10, 10, 10],  
              [20, 0, 30, 0, 0, 40, 0, 40, 0], 30)
```

Η βέλτιστη λύση είναι να συνδεθούν οι σταθμοί **1** και **6**, και σε αυτή την περίπτωση η διάμετρος είναι **110**.

Παράδειγμα 3

Ο βαθμολογητής (grader) κάνει την παρακάτω κλήση συνάρτησης:

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

Η βέλτιστη λύση είναι να συνδεθούν οι σταθμοί **2** και **3**, μειώνοντας τη διάμετρο σε **21**.

Παράδειγμα 4

Ο βαθμολογητής (grader) κάνει την παρακάτω κλήση συνάρτησης:

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

Συνδέοντας οποιουδήποτε δύο σταθμούς με ταχεία γραμμή μήκους **2** δε βελτιώνει την αρχική διάμετρο του σιδηροδρομικού δικτύου, το οποίο είναι **4**.

Υποπροβλήματα

Σε όλα τα υποπροβλήματα $2 \leq n \leq 1\,000\,000$, $1 \leq l_i \leq 10^9$, $0 \leq d_i \leq 10^9$, $1 \leq c \leq 10^9$.

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

Υπόδειγμα βαθμολογητή (Sample grader)

Το υπόδειγμα βαθμολογητή διαβάζει την είσοδο με την παρακάτω μορφή:

- γραμμή 1: integers n και c ,
- γραμμή 2: integers l_0, l_1, \dots, l_{n-2} ,
- γραμμή 3: integers d_0, d_1, \dots, d_{n-1} .