

Parcel Post (multihop)

Per consegnare i pacchi in modo più efficiente, l'ufficio postale ha costruito una rete di tubi pneumatici sotto le strade di Londra. La rete è composta da N stazioni di smistamento collegate da $N - 1$ tubi bidirezionali. C'è un percorso unico tra ogni coppia di stazioni e i pacchi verranno inviati lungo seguendo il percorso dalla sorgente alla destinazione.

Quando un pacco si trova nella stazione di smistamento i , ci sono due opzioni per inviarlo a destinazione. Si può sparare a *bassa potenza* al costo di A_i , in tal caso viaggerà lungo un unico tubo fino alla stazione successiva del suo percorso. In alternativa, può essere sparato ad *alta potenza*. In questo caso, l'operatore selezionerà la pressione $k \geq 1$ e il pacco attraverserà i successivi k tubi del suo percorso, al costo di $B_i + k \cdot C$.

L'ufficio postale instraderà i pacchi in modo da ridurre al minimo il costo totale, ma per evitare congestione sulla rete i pacchi devono rimanere sul percorso diretto dalla sorgente alla destinazione. Il tuo compito è trovare i costi minimi per l'invio di Q pacchi in questa rete.

Implementazione

Dovrai inviare un singolo file sorgente `.cpp`.

📁 Tra gli allegati di questo problema troverai un template `multihop.cpp` con un esempio di implementazione.

Dovrai implementare la seguente funzione:

```
C++    void init(int N, int C, vector<int> A, vector<int> B, vector<int> U,
        vector<int> V);
C++    long long query(int X, int Y);
```

- L'intero N rappresenta il numero di stazioni di smistamento.
- L'intero C rappresenta il costo incrementale per unità di pressione quando si spara ad alta potenza, come descritto sopra.
- L'array A , indicizzato da 0 a $N - 1$, contiene il costo di sparare a bassa potenza da ogni nodo.
- L'array B , indicizzato da 0 a $N - 1$, contiene il costo base di sparare ad alta potenza da ogni nodo, come descritto sopra.
- Gli array U e V descrivono i tubi nella rete: c'è un tubo tra la stazione di smistamento $U[i]$ e la stazione di smistamento $V[i]$.
- `query` deve restituire il costo minimo per spedire un pacco dalla stazione di smistamento X alla stazione di smistamento Y .

Il grader chiamerà la funzione `init` e poi chiamerà `query` Q volte, stampando il suo valore di ritorno sul file di output.

Grader di prova

La cartella del problema contiene una versione semplificata del grader, che puoi usare per testare la tua soluzione in locale. Il grader semplificato legge i dati di input da `stdin`, chiama le funzioni che devi implementare, e scrive l'output su `stdout`.

L'input è composto da $N + Q + 2$ righe, contenenti:

- Riga 1: gli interi N, Q, C .
- Riga 2: gli interi A_i , separati da uno spazio.
- Riga 3: gli interi B_i , separati da uno spazio.
- Riga $4 + i$ ($0 \leq i < N - 1$): gli interi U_i, V_i .
- Riga $4 + (N - 1) + i$ ($0 \leq i < Q$): gli interi X_i, Y_i .

L'output è composto da Q righe, contenenti i valori restituiti dalla funzione `query`.

Assunzioni

- $1 \leq N \leq 100\,000$.
- $1 \leq Q \leq 100\,000$.
- $1 \leq C \leq 1\,000\,000\,000$.
- $1 \leq A_i \leq 1\,000\,000\,000$ per ogni $i = 0, \dots, N - 1$
- $1 \leq B_i \leq 1\,000\,000\,000$ per ogni $i = 0, \dots, N - 1$
- $0 \leq U_i < N$.
- $0 \leq V_i < N$.

Assegnazione del punteggio

Il tuo programma verrà testato su più test case raggruppati in subtask. Per ottenere il punteggio associato a un subtask, devi risolvere correttamente tutti i casi di test che contiene.

- **Subtask 1** [0 punti]: Casi di esempio.
- **Subtask 2** [5 punti]: $A_i \leq 10, B_i \leq 10$ per ogni $i = 0, \dots, N - 1, C \leq 10, N \leq 10, Q \leq 10$.
- **Subtask 3** [10 punti]: $N \leq 5000, Q = 1$.
- **Subtask 4** [25 punti]: $N \leq 100\,000, Q = 1$.
- **Subtask 5** [25 punti]: $N \leq 5000$.
- **Subtask 6** [35 punti]: Nessuna limitazione aggiuntiva.

Esempi di input/output

stdin	stdout
5 1 4 2 8 6 9 2 2 5 9 5 2 3 0 2 3 4 2 1 4 0 1	16
5 5 3 9 7 9 4 5 5 10 8 9 7 4 3 0 4 2 0 1 2 4 0 3 1 0 3 3 0 1 4	5 20 11 9 19

Spiegazione

Nel **primo caso di esempio**, possiamo spedire il pacco dalla stazione di smistamento 0 alla 4 ad alta potenza, per un costo di 14, e poi dalla 4 alla 1 a bassa potenza energia, per un costo di 2. Il costo finale è quindi 16 che è ottimale.