Olimpiadi Italiane di Informatica – Salerno 2013

Maree a Venezia (maree)

Descrizione del problema

I gondolieri di Venezia si spostano lungo i canali, seguendo la corrente. Ogni canale è quindi percorribile in una sola direzione, nota ai gondolieri. Quando però arriva l'alta marea, le dinamiche della laguna sono tali che il verso della corrente di ogni canale si inverte. Dovete aiutare i gondolieri a pianificare il loro percorso per spostarsi in gondola da un punto di origine a un punto di destinazione, sapendo che sta per arrivare l'alta marea.

Data la mappa degli N punti di intersezione e degli M canali della città di Venezia che li collegano, con le correnti che inizialmente ne fluiscono, dovete determinare il minimo tempo necessario per spostarsi in gondola dal punto 0 al punto 0, tenendo presente che:

- per attraversare un canale cosa possibile solo nel senso della corrente si impiega esattamente 1 minuto;
- dopo T minuti arriva l'alta marea e quindi la direzione di percorrenza di tutti i canali si inverte;
- nei punti di intersezione di due o più canali è possibile fermarsi per un numero di minuti a piacere.



Per esempio, considerate questa mappa. Per andare dal punto 0 al punto N-1=8, sapendo che l'alta marea arriverà al minuto T=5, si può seguire il seguente percorso (che richiede 7 minuti):

- da t = 0 a $t = 1: (0) \rightarrow (1)$
- da t=1 a $t=2: (1) \rightarrow (4)$
- da t = 2 a t = 3: $\boxed{4} \rightarrow \boxed{2}$
- da t = 3 a $t = 4: 2 \to 3$
- da t = 4 a t = 5: si rimane fermi in 3
- in t = 5: alta marea
- da t = 5 a t = 6: (3) \rightarrow (6)
- da t = 6 a t = 7: $6 \rightarrow 8$

Un percorso alternativo, sebbene richieda 8 minuti e quindi non sia una soluzione valida, è il seguente:

```
• da t = 0 a t = 1: (0) \to (1)
```

- da t = 1 a $t = 2: (1) \to (4)$
- da t = 2 a $t = 3: (4) \rightarrow (5)$
- da t = 3 a t = 5: si rimane fermi in (5)
- in t = 5: alta marea
- da t = 5 a $t = 6: (5) \to (7)$
- da t = 6 a $t = 7: (7) \to (6)$
- da t = 7 a $t = 8: (6) \rightarrow (8)$

È facile verificare che, nella mappa mostrata, non ci sono modi di arrivare in N-1 in meno di 7 minuti. Per risolvere il problema, dovete scrivere una funzione solve(N, M, T, S, E), che restituisca un singolo numero intero: il minimo tempo necessario per arrivare dal punto 0 al punto N-1, sapendo che l'*i*-esimo degli M canali collega i punti di intersezione S_i ed E_i , che la corrente inizialmente fluisce da S_i a E_i , e che l'alta marea arriva al minuto T. Se non è possibile andare dal punto 0 al punto N-1, allora la funzione deve restituire il valore -1.

Subtask

- Subtask 0 [5 punti]: caso di esempio.
- Subtask 1 [11 punti]: $2 \le N \le 100\,000$, $1 \le M \le 2\,000\,000$ e T = 0.
- Subtask 2 [25 punti]: $2 \le N \le 3000$, $1 \le M \le 20000$ e $0 \le T \le 1000000000$.
- Subtask 3 [38 punti]: $2 \le N \le 100\,000$, $1 \le M \le 2\,000\,000$ e $0 \le T \le 1\,000\,000\,000$.
- Subtask 4 [21 punti]: $2 \le N \le 200\,000$, $1 \le M \le 3\,000\,000$ e $0 \le T \le 1\,000\,000\,000$.

Dettagli di implementazione

Dovrai sottoporre esattamente un file con estensione: .c, .cpp o .pas. Questo file deve implementare la funzione solve utilizzando uno dei seguenti prototipi.

Programma in C o C++

```
int solve(int N, int M, int T, int* S, int* E);
```

Programma in Pascal

```
function solve(N, M, T: longint; var S, E: array of longint): longint;
```

La funzione implementata dovrà comportarsi come sopra descritto. Naturalmente sei libero di implementare altre routine ausiliarie per uso interno. La tua sottoposizione non deve effettuare direttamente operazioni di input o output, via console o via file di testo.

Funzionamento del grader di esempio

Nella directory relativa a questo problema è presente una versione semplificata del grader usato durante la correzione, che potete usare per testare le vostre soluzioni in locale. Il grader di esempio legge i dati di input dal file input.txt, chiama la funzione che dovete implementare, e scrive il risultato restituito dalla vostra funzione sul file output.txt. Il file input.txt deve essere composto da M+1 righe, in questo formato:

- nella prima riga devono essere presenti 3 interi: N, M, T;
- nella i-esima delle successive M righe devono essere presenti 2 interi: S_i e E_i .

Esempio di input/output

File input.txt	File output.txt
9 10 5	7
0 1	
1 4	
4 2	
2 0	
2 3	
4 5	
7 5	
6 7	
6 3	
8 6	

Nota: l'esempio riportato è relativo alla situazione mostrata in figura.