

1. Ritratto (**ritratto**)

Descrizione del problema

La comitiva di turisti Tedeschi è ora in visita ad una città d'arte, nella quale si prestano a un ritratto da parte di Giangi, il pittore di strada. Come da tradizione per i ritratti, decidono di disporsi in una fila basso-ordinata.

Una fila basso-ordinata è una fila SbD , dove b è la persona più bassa della fila, S e D sono file basso-ordinate, e le persone in S sono tutte più basse delle persone in D .

Giangi vuole spillare più soldi possibile dalla comitiva, facendogli un gran numero di ritratti. Quanti ritratti diversi gli potrà fare, al massimo, sapendo che la comitiva si disporrà sempre in file basso-ordinate?

Dati di input

La prima e unica riga del file `input.txt` contiene un unico numero intero, la dimensione N della comitiva.

Dati di output

Il file `output.txt` consisterà di un'unica riga, contenente un unico numero intero R , il numero massimo di ritratti che Giangi può fare.

Assunzioni

- $1 \leq N \leq 100000$

Esempi di input/output

File <code>input.txt</code>	File <code>output.txt</code>
3	5

2. Homer (`homer`)

Descrizione del problema

Homer esce di casa per fare una passeggiata. Come tutti sanno, Homer si sposta prendendo soltanto strade in discesa, ed è particolarmente attratto dalle ciambelle.

La città di Mollacampo ha N incroci, dove l'incrocio numero 1 è la casa di Homer, e in ogni incrocio è presente una pasticceria. Qual'è il massimo numero di pasticcerie che Homer può visitare in un'unica passeggiata?

Dati di input

La prima riga del file `input.txt` contiene due numeri interi, N , M .

N è il numero di incroci presenti nella città di Mollacampo, M è il numero di strade dirette tra coppie di incroci. Le successive M righe contengono ciascuna due interi, x_i , y_i , che significano la presenza di una strada che parte dall'incrocio x_i e procede in discesa fino all'incrocio y_i .

Dati di output

Il file `output.txt` consisterà di un'unica riga, contenente un unico numero intero P , il massimo numero di pasticcerie che Homer può visitare in un'unica passeggiata.

Assunzioni

- $1 \leq N \leq 100000$
- $0 \leq M \leq 100000$
- $1 \leq x_i, y_i \leq N$

Esempi di input/output

File <code>input.txt</code>	File <code>output.txt</code>
5 5 1 2 1 3 2 4 4 5 3 5	4

3. Tik-tok III (revenge)

Descrizione del problema

Dopo il grande successo commerciale di Tik-tok Pro Evolution, la Mela[©] ha deciso di rilasciarne il sequel per iPhone: Tik-tok III - Revenge. In questo gioco, due robot (Tik e Tok) sono disposti in un labirinto $R \times C$.

In ogni turno, il giocatore sceglie una direzione tra le quattro direzioni cardinali N, S, O, E , e trasmette il comando via radio ai due robot. I due robot sono tarati sulle stesse frequenze, e non è quindi possibile controllarli individualmente: entrambi attuano il comando ricevuto muovendosi nella stessa direzione. Alcune caselle del labirinto contengono delle trappole, in grado di distruggere i robot che ci passano sopra. Altre caselle invece contengono dei muri: se un robot cerca di muoversi in una casella con un muro, rimarrà invece fermo senza venirne danneggiato (analogo effetto accade a un robot che cerca di uscire dal labirinto). Altre caselle contengono un interruttore, che viene automaticamente attivato quando la casella viene occupata da un robot.

Usa Tik e Tok per attivare due diversi interruttori contemporaneamente, e passa al livello successivo!

Dati di input

La prima riga del file `input.txt` contiene due numeri interi R, C , la dimensione del labirinto.

La seconda riga contiene quattro numeri interi, $Tik_R, Tik_C, Tok_R, Tok_C$, la posizione iniziale di Tik e Tok.

Le successive R righe contengono ciascuna C caratteri consecutivi rappresentanti il labirinto, dove:

. Casella libera
Muro
X Trappola
0 Interruttore

Dati di output

La prima riga del file `output.txt` conterrà un unico numero intero N , il minimo numero di mosse necessario per vincere la partita (-1 se non c'è modo di vincere la partita).

La seconda riga conterrà N caratteri consecutivi $c_i = N, S, O, E$, la sequenza di mosse da fare per vincere.

Assunzioni

- $1 \leq R, C \leq 500$
- $1 \leq Tik_R, Tok_R \leq R$
- $1 \leq Tik_C, Tok_C \leq C$
- I robot sono abbastanza piccoli da poter stare entrambi nella stessa casella.
- Un programma che trova una soluzione non minima prende il 50% del punteggio.

Esempi di input/output

File input.txt	File output.txt
5 6	7
1 2 1 3	SSEESES
X..#..#	
.#.#X..	
.#0..#.	
..##.0.	
#....#X	