

# 1. Atollo di Nueva Esperancia (atollo)

## Descrizione del problema

Il governo dell'atollo di Nueva Esperancia ha intenzione di recuperare consensi intraprendendo un'imponente opera pubblica: collegare tutto l'arcipelago di pertinenza della repubblica con dei lunghi ponti, per favorire una maggiore cooperazione e industrializzazione del territorio.

In verità, già qualche anno prima il partito di opposizione aveva iniziato un progetto simile, costruendo alcuni ponti. Tuttavia l'esaurimento prematuro dei fondi ha fatto sì che non tutte le isole dell'arcipelago siano collegate, mentre ora il governo in carica vorrebbe completare il progetto consentendo di passare da una qualunque isola a una qualunque altra spostandosi solamente via terra.

L'unico problema, questa volta, è che solo alcuni collegamenti tra isole sarebbero costruibili, perchè il fondale tra le isole spesso è troppo profondo per consentire un collegamento diretto. Aiuta il ministro dei trasporti calcolando il minimo numero di ponti che sarebbe necessario costruire (e quali ponti), affinché possa verificare la disponibilità di fondi prima di impegnarsi in un progetto di questa portata!

## Dati di input

La prima riga del file `input.txt` contiene tre numeri interi  $N$ ,  $M$ ,  $L$ , rispettivamente il numero di isole, il numero di ponti già costruiti, e il numero di ponti che si possono costruire.

Le successive  $M$  righe contengono ciascuna due interi,  $x_i$ ,  $y_i$ , che indicano la presenza di un ponte già costruito tra l'isola  $x_i$  e l'isola  $y_i$ . Le ultime  $L$  righe contengono ciascuna due interi,  $x_i$ ,  $y_i$ , che indicano la possibilità di costruire un ponte tra l'isola  $x_i$  e l'isola  $y_i$ .

## Dati di output

La prima riga del file `output.txt` conterrà un numero intero  $P$ , il minimo numero di ponti che è necessario costruire per collegare tutte le isole via terra. Le successive  $P$  righe conterranno ciascuna due interi,  $x_i$ ,  $y_i$ , che rappresentano il progetto di costruire un ponte tra l'isola  $x_i$  e l'isola  $y_i$ .

## Assunzioni

- $1 \leq N \leq 10000$
- $0 \leq M, L \leq 1000000$
- $1 \leq x_i < y_i \leq N$ , per ogni  $i$ ;
- Ogni isola è collegata con al più 1000 altre; ed esiste un modo per collegare tutte le isole con i ponti dati.

## Esempi di input/output

File input.txt	File output.txt
6 5 3	1
1 2	3 5
2 3	
3 4	
1 4	
5 6	
1 3	
2 4	
3 5	

## 2. Worms Karmagheddon (worms)

### Descrizione del problema

Una colossale partita di *Worms Karmagheddon*<sup>©</sup> sta avendo luogo in una mappa costituita da  $N$  isolotti, collegati tra di loro da  $N - 1$  ponti, in modo tale che tra ogni due isolotti c'è uno e un unico percorso per passare da un isolotto all'altro (attraversando eventualmente altri isolotti). In ogni isolotto c'è un verme nemico.

Hai appena raccolto la rarissima arma *Vecchietta di Cemento da Lancio*, che puoi scagliare in un isolotto a tua scelta e poi condurre in un qualunque percorso distruggendo tutto quello che trova nel suo percorso, senza mai tornare mai sui tuoi passi (perché altrimenti affonderebbe, essendo quella parte già distrutta).

Qual'è il numero massimo di vermi nemici che puoi uccidere in questo modo?

### Dati di input

La prima riga del file `input.txt` contiene un unico numero intero,  $N$ .

Le successive  $N - 1$  righe contengono ciascuna due interi,  $x_i, y_i$ , che significano la presenza di un ponte tra la zona  $x_i$  e la zona  $y_i$ .

### Dati di output

Il file `output.txt` consisterà di un'unica riga, contenente un unico numero intero  $W$ , il numero massimo di vermi nemici che la *Vecchietta di Cemento da Lancio* può sterminare.

### Assunzioni

- $1 \leq N \leq 100000$
- $1 \leq x_i < y_i \leq N$ , per ogni  $i$ ;

### Esempi di input/output

File <code>input.txt</code>	File <code>output.txt</code>
10 1 2 1 3 3 4 4 5 5 6 4 7 3 8 8 9 9 10	7

### 3. Tre giorni in Tibet (tibet)

#### Descrizione del problema

Vi aspetta una magnifica vacanza in ritiro nel monastero di Whou Tzun! Per raggiungere il monastero, tuttavia, dovete ancora attraversare l'impervia vallata di Shi-Ho. La vallata ha forma rettangolare ed è suddivisa in  $N \times M$  cocuzzoli, voi siete nell'angolo in alto a destra  $(1, 1)$  e il monastero di Whou Tzun è nell'angolo opposto  $(N, M)$ .

Fortunatamente avete a disposizione una mappa molto accurata della vallata, che riporta per ogni cocuzzolo l'altitudine corrispondente  $H_{ij}$ . Nel cercare di raggiungere il monastero, potete ogni volta muovervi di un passo a Nord, a Sud, a Ovest, o a Est (ma non in diagonale), e l'attraversamento vi richiederà un'ora se il cocuzzolo su cui vi state muovendo è ad un'altitudine pari o inferiore al cocuzzolo su cui eravate precedentemente, oppure due ore se il nuovo cocuzzolo è più in alto del vecchio.

Non vedete l'ora di raggiungere l'agognato monastero: quante ore, al minimo, vi richiederà il tragitto?

#### Dati di input

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

Le successive  $N$  righe contengono ciascuna  $M$  interi  $H_{ij}$ , che indicano l'altitudine del corrispondente cocuzzolo.

#### Dati di output

Il file `output.txt` consisterà di un'unica riga, contenente un unico numero intero  $T$ , il numero di ore che è necessario impiegare per raggiungere il monastero di Whou Tzun.

#### Assunzioni

- $1 \leq N \leq 2000$
- $1 \leq M \leq 2000$
- $1 \leq H_{ij} \leq 1000000$

#### Esempi di input/output

File input.txt	File output.txt
10 4 1 10 9 6 8 1 8 7 9 2 7 8 10 3 6 9 11 4 5 10 10 10 1 11 11 9 2 12 9 8 3 13 11 7 4 14 12 6 5 1	16