

Impila le scatole

Descrizione del problema

Mojito e i suoi amici sono a caccia di cibo nascosto nei più disparati angoli della casa di Monica!



In casa di Monica sono disponibili N diverse scatole che è possibile utilizzare, e la scatola i -esima è rappresentata da un parallelepipedo i cui tre lati misurano A_i , B_i , C_i metri.

I quadrupedi, in assenza di Monica, hanno deciso utilizzare le N scatole a disposizione per raggiungere le più alte credenze in cui Monica ha nascosto del cibo! Per farlo, hanno intenzione di impilare tutte le scatole una sopra l'altra scegliendo, per ognuna di essa, il lato da utilizzare in altezza e sul quale saltare.

Le scatole saranno impilate appoggiando le facce scelte tra di loro, e l'altezza finale della pila di scatole così creata sarà la somma dei lati corrispondenti alle altezze delle scatole disposte.

Aiuta Mojito e i suoi amici a trovare l'altezza massima raggiungibile impilando le N scatole a disposizione!

Nota: Quando si parla di cibo i cani sono maestri di agilità, e saranno in grado di impilare le scatole in qualsiasi configurazione sfidando la legge di gravità!

Dati di input

La prima riga del file di input contiene un intero T , il numero di casi di test. Seguono T casi di test, numerati da 1 a T . Ogni caso di test è preceduto da una riga vuota.

Ogni caso di test è composto da $N + 1$ righe:

- la prima riga contiene l'intero N ;
- le successive N righe contengono ciascuna tre interi A_i , B_i e C_i .

Dati di output

Il file di output deve contenere la risposta ai casi di test che sei riuscito a risolvere. Per ogni caso di test che hai risolto, il file di output deve contenere una riga con la dicitura:

```
Case #t: h
```

dove t è il numero del caso di test (a partire da 1) e il valore h è la massima altezza ottenibile impilando le N scatole.

Assunzioni

- $T = 6$, nei file di input che scaricherai saranno presenti esattamente 6 casi di test.
- $1 \leq N \leq 100$.
- $1 \leq A_i, B_i, C_i \leq 100$ per ogni $0 \leq i \leq N - 1$.

Nei primi 2 casi di test valgono le seguenti assunzioni aggiuntive:

- $A_i = B_i = C_i$ (tutte le scatole sono cubiche).

Esempi di input/output

Input:

```
2
2
2 3 4
2 2 2

3
1 3 7
4 2 1
3 9 5
```

Output:

```
Case #1: 6
Case #2: 20
```

Spiegazione

Nel **primo caso d'esempio** è possibile raggiungere l'altezza di 6 metri impilando la prima scatola con altezza 4 e la seconda scatola con altezza 2.

Nel **secondo caso d'esempio** è possibile raggiungere l'altezza di 20 metri impilando la prima scatola con altezza 7, la seconda scatola con altezza 4 e la terza scatola con altezza 9.

Collezionismo di robot

Descrizione del problema

Luigi è un collezionista di modellini di robot. Finalmente si è deciso a sistemare la sua collezione in una serie di bacheche, ma ha bisogno del tuo aiuto per disporla ottimalmente!



Luigi possiede in totale N modellini, ognuno dei quali ha un valore di collezionismo C_i , che vuole disporre su K scaffali. Ogni modellino deve essere messo su esattamente uno scaffale e ogni scaffale deve contenere almeno un modellino.

Poiché Luigi non vuole far sfigurare nessun modellino della sua preziosa collezione, vuole assicurarsi che la differenza dei valori di collezionismo C_i dei modellini su uno stesso scaffale non sia troppo alta. Luigi quindi assegna ad ogni scaffale un fattore di discrepanza D_j , definito come la differenza tra il massimo ed il minimo valore di collezionismo dei modellini su quello scaffale.

Luigi, per riuscire nel suo intento, vuole quindi cercare di minimizzare i fattori di discrepanza D_j dei K scaffali. Qual è il valore minimo della somma di questi fattori?

Dati di input

La prima riga del file di input contiene un intero T , il numero di casi di test. Seguono T casi di test, numerati da 1 a T . Ogni caso di test è preceduto da una riga vuota.

Ogni caso di test è composto da 2 righe:

- la prima riga contiene i due interi N e K ;
- la seconda riga contiene N interi, i valori di collezionismo C_i .

Dati di output

Il file di output deve contenere la risposta ai casi di test che sei riuscito a risolvere. Per ogni caso di test che hai risolto, il file di output deve contenere una riga con la dicitura

Case #t: s

dove t è il numero del caso di test (a partire da 1) e il valore s è la somma minima dei fattori di discrepanza dei K scaffali.

Assunzioni

- $T = 15$, nei file di input che scaricherai saranno presenti esattamente 15 casi di test.
- $1 \leq K \leq N \leq 10000$.

- $0 \leq C_i \leq 10^9$ per ogni $0 \leq i < N$.

Nei primi 6 casi di test valgono le seguenti assunzioni aggiuntive:

- $N \leq 50$.
- $K \leq 6$.

Esempi di input/output

Input:

```
2
4 2
7 9 3 1

6 3
4 42 23 0 21 2
```

Output:

```
Case #1: 4
Case #2: 6
```

Spiegazione

Nel **primo caso d'esempio** Luigi può mettere il primo ed il secondo modellino nel primo scaffale, ottenendo un fattore di discrepanza **2**, e il terzo e quarto modellino nel secondo scaffale, ottenendo di nuovo **2**. La somma minima è quindi **4**.

Nel **secondo caso d'esempio** Luigi può mettere il secondo modellino da solo nel primo scaffale, ottenendo **0**, il primo, quarto e sesto modellino nel secondo scaffale, ottenendo **4**, e il terzo e quinto modellino nel terzo scaffale, ottenendo **2**. La somma minima è quindi **6**.

Corsa ad ostacoli

Descrizione del problema

Monica ha deciso di portare Mojito ad una corsa ad ostacoli!



La corsa si svolge in un tracciato rettilineo lungo L metri, e la partenza si trova all'inizio del tracciato nel punto 0 . Nel tracciato sono presenti N ostacoli: ciascun ostacolo si trova a X_i metri dalla partenza ed ha una difficoltà P_i , per cui superando tale ostacolo Mojito guadagna P_i punti.

Per rendere la corsa più difficile, gli organizzatori hanno però deciso che ciascun ostacolo può essere attraversato solo in un breve intervallo. In particolare, l' i -esimo ostacolo appare dopo S_i secondi dall'inizio della gara e scompare immediatamente dopo, quindi per poter superare l'ostacolo Mojito deve trovarsi esattamente nel punto X_i all'istante S_i . Mojito può correre avanti e indietro nel tracciato oppure può stare fermo a rincorrersi la coda, purtroppo però è in grado di correre solo fino a 1 metro al secondo.

La corsa dura esattamente D secondi, al termine dei quali si sommano i punteggi degli ostacoli che Mojito è riuscito a superare. Mojito si è allenato duramente, ed è quindi in grado di superare tutti gli ostacoli che riesce a raggiungere in tempo senza difficoltà. Qual è il massimo numero di punti che Mojito riesce a totalizzare?

Dati di input

La prima riga del file di input contiene un intero T , il numero di casi di test. Seguono T casi di test, numerati da 1 a T . Ogni caso di test è preceduto da una riga vuota.

Ogni caso di test è composto da $N + 1$ righe:

- la prima riga contiene i tre interi N , L , D ;
- le successive N righe contengono ciascuna tre interi X_i , P_i e S_i .

Dati di output

Il file di output deve contenere la risposta ai casi di test che sei riuscito a risolvere. Per ogni caso di test che hai risolto, il file di output deve contenere una riga con la dicitura

Case #t: x

dove t è il numero del caso di test (a partire da 1) e il valore x è il massimo punteggio che Mojito può totalizzare.

Assunzioni

- $T = 19$, nei file di input che scaricherai saranno presenti esattamente 19 casi di test.
- $1 \leq N \leq 1000$.
- $1 \leq L, D \leq 10^9$.
- $0 \leq X_i \leq L$ per ogni $0 \leq i \leq N - 1$.
- $0 \leq P_i \leq 10^6$ per ogni $0 \leq i \leq N - 1$.
- $0 \leq S_i \leq S_{i+1} \leq D$ per ogni $0 \leq i \leq N - 2$.

Nei primi 11 casi di test valgono le seguenti assunzioni aggiuntive:

- $L \leq 100$.

Nei primi 6 casi di test valgono, inoltre, le seguenti assunzioni aggiuntive:

- $N, D \leq 100$.

Esempi di input/output

Input:

```
2
2 20 20
7 30 2
9 10 11

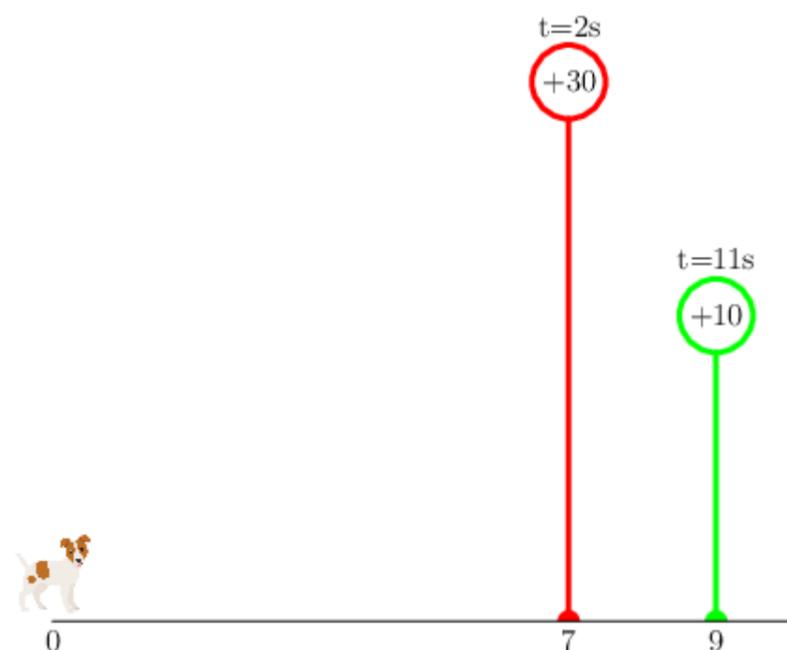
4 10 10
7 20 7
4 6 7
3 7 8
2 5 9
```

Output:

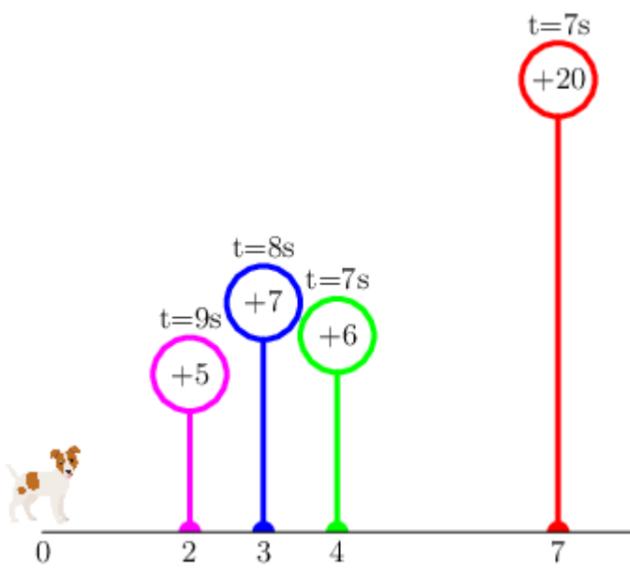
```
Case #1: 10
Case #2: 20
```

Spiegazione

Nel **primo caso d'esempio** Mojito non è in grado di raggiungere il primo ostacolo all'istante 2 in quanto è troppo distante dalla partenza. Quindi, può solo prendere il secondo ostacolo guadagnando 10 punti. Per farlo, per esempio, può correre fino al punto dove comparirà l'ostacolo, rincorrersi la coda sul posto per 2 secondi e all'istante 11 superare l'ostacolo.



Nel **secondo caso d'esempio** Mojito può andare all'istante 7 nel punto 4 e poi spostarsi nei punti 3 e 2 prendendo gli ultimi 3 ostacoli, oppure può andare all'istante 7 nel punto 7. La prima scelta permette a Mojito di guadagnare $6 + 7 + 5 = 18$ punti mentre la seconda scelta gli permette di guadagnare 20 punti, la risposta è quindi 20.



Percorso palindromo

Descrizione del problema

Per festeggiare i 40 anni di Monica, Mojito le ha preparato una sorta di caccia al tesoro.

Come ben noto, l'infrastruttura di Milano (la città dove vive Monica) è formata da N incroci, numerati da 0 a $N - 1$, e da M strade bidirezionali numerate da 0 a $M - 1$ che collegano tali incroci. La i -esima strada collega gli incroci A_i e B_i .

Mojito ha disseminato una pallina in ciascuna delle M strade. Su ogni pallina è impressa una lettera: in particolare, la pallina posta nella i -esima strada reca la lettera L_i .



L'obiettivo di Monica è andare (a piedi) dal suo appartamento, che si trova all'incrocio X , all'incrocio Y in cui la aspetta Mojito. Ogni volta che percorre una strada, Monica deve segnare su un foglio la lettera scritta sulla pallina che si trova in quella strada (lasciando la pallina dov'è). Quando raggiunge Mojito, Monica avrà vinto se sul foglio è scritta una parola palindroma (una parola si dice *palindroma* quando rimane invariata se la si legge al contrario).

Aiuta Monica a vincere il gioco percorrendo il minimo numero possibile di strade, per evitare di infrangere il coprifuoco!

Nota: Il percorso di Monica può passare più volte da uno stesso incrocio o anche da una stessa strada.

Dati di input

La prima riga del file di input contiene un intero T , il numero di casi di test. Seguono T casi di test, numerati da 1 a T . Ogni caso di test è preceduto da una riga vuota.

Ogni caso di test è composto da $M + 2$ righe:

- la prima riga contiene i due interi N e M , il numero di incroci e il numero di strade.
- la seconda riga contiene i due interi X e Y , l'incrocio di partenza e l'incrocio di destinazione.
- le successive M righe contengono ciascuna i due interi A_i , B_i e il carattere L_i che rappresentano l' i -esima strada.

Dati di output

Il file di output deve contenere la risposta ai casi di test che sei riuscito a risolvere. Per ogni caso di test che hai risolto, il file di output deve contenere una riga con la dicitura

```
Case #t: s
```

dove t è il numero del caso di test (a partire da 1) e il valore s è la lunghezza minima di un percorso palindromo tra X e Y . Se tale percorso non esiste, s deve valere -1 .

Assunzioni

- $T = 24$, nei file di input che scaricherai saranno presenti esattamente 24 casi di test.
- $1 \leq N \leq 300$.

- $1 \leq M \leq 300$.
- $0 \leq X, Y \leq N - 1$.
- $0 \leq A_i, B_i \leq N - 1$ per ogni $0 \leq i \leq M - 1$.
- L_i è una lettera minuscola dell'alfabeto inglese (a-z) per ogni $0 \leq i \leq M - 1$.
- Per ogni coppia di incroci esiste al più una strada che li collega, e non esistono strade che iniziano e terminano nello stesso incrocio. Tutte le strade possono essere percorse in qualsiasi direzione.

Nei primi 5 casi di test valgono le seguenti assunzioni aggiuntive:

- Tutte le lettere L_i sono uguali.

Esempi di input/output

Input:

```

3

5 5
0 4
0 1 b
1 3 c
0 2 a
2 1 c
3 4 a

6 5
3 4
0 5 a
4 5 a
3 1 z
2 3 a
0 3 b

4 3
0 2
0 1 o
1 2 i
2 3 i

```

Output:

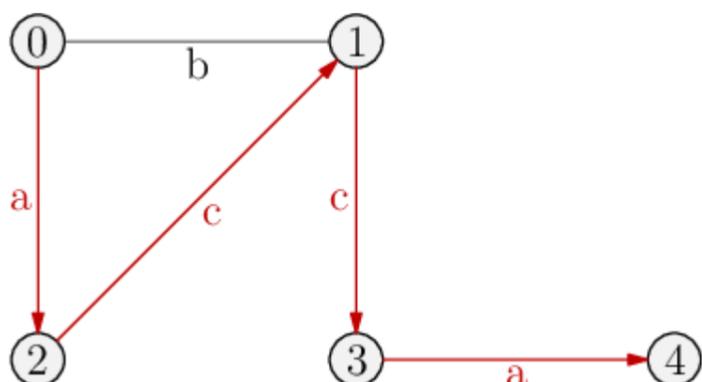
```

Case #1: 4
Case #2: 5
Case #3: -1

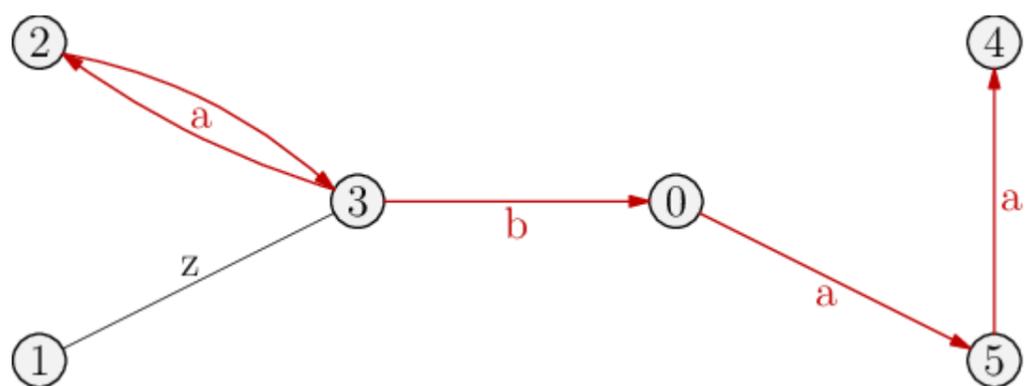
```

Spiegazione

Nel **primo caso d'esempio** il percorso palindromo di lunghezza minima tra i nodi 0 e 4 è $0 \rightarrow 2 \rightarrow 1 \rightarrow 3 \rightarrow 4$, che compone la parola **acca**. Esiste un percorso più breve, $0 \rightarrow 1 \rightarrow 3 \rightarrow 4$, che passa per 3 strade ma non è palindromo e quindi non risulta valido.



Nel **secondo caso d'esempio** la soluzione è il percorso palindromo $3 \rightarrow 2 \rightarrow 3 \rightarrow 0 \rightarrow 5 \rightarrow 4$, per un totale di 5 strade percorse, che compone la parola **aabaa**. Si noti come questo percorso passa due volte per la strada che collega gli incroci 2 e 3.



Nel **terzo caso d'esempio**, qualunque percorso che collega gli incroci 0 e 2 inizia con la lettera **o** e termina con la lettera **i**, e pertanto non può essere palindromo.

