



Tax avoidance (tax)

Autore: *Dario Ostuni*

Narco è di nuovo alle prese con la giustizia, questa volta quella tributaria. L'azienda che ha fondato, la *Pear*, è stata condannata per evasione fiscale, e ora Narco deve trovare nuovi modi più "legali" per far circolare il denaro. Nello specifico *Pear* è composta di N sedi sparse in giro per il globo, e Narco è riuscito a stabilire M connessioni monodirezionali in modo che si possa inviare soldi dalla sede A_i alla sede B_i con una tassazione del C_i per mille. Narco è con questa rete di connessioni in grado di spostare denaro da qualunque sede a qualunque altra sede (a volte dovendo fare più di un passaggio per farlo arrivare).

Narco ora si chiede quale sia la coppia di sedi tali che, assumendo di instradare sempre nel modo migliore il denaro in modo da minimizzare le tasse pagate, la quantità di denaro pagato in tasse per trasferire denaro dalla prima alla seconda sia massima. Aiuta Narco a scoprirlo!

Implementazione

Il tuo programma deve leggere da `stdin` l'input nel seguente formato:

- la prima riga contiene i due interi N ed M separati da spazio;
- le successive M righe, che indicizziamo con i da 0 a $M-1$, contengono tre interi A_i , B_i e C_i , indicanti che esiste una connessione per trasferire denaro dalla sede A_i alla sede B_i con una tassazione del C_i per mille.

Il tuo programma deve scrivere in `stdout` l'output nel seguente formato:

- la prima riga deve contenere la coppia tale per cui la tassazione è massima (nel caso esistano più coppie con la stessa tassazione, restituire quella lessicograficamente minore).

Assunzioni

- $2 \leq N \leq 300$
- $2 \leq M \leq 50000$
- $0 \leq C_i < 1000$ per ogni $0 \leq i < M$
- Esiste sempre un percorso che connette ogni coppia di sedi

Esempi di input/output

stdin	stdout
3 5 0 2 400 2 1 400 1 0 300 0 1 600 2 0 100	0 1
4 7 1 3 900 3 0 150 2 1 800 0 2 400 1 0 400 3 1 200 0 3 0	2 0