

CORSO DI ALGORITMI E STRUTTURE DATI

Prof. ROBERTO PIETRANTUONO

Prova del 16/02/2022

Indicazioni

Si consegnino un file in **formato editabile (.txt, .docx, .rtf, etc.)** nominandolo "*CognomeNome*", in cui è riportata l'implementazione (nel linguaggio scelto) seguita da una indicazione della complessità temporale dell'algoritmo implementato (complessità nel caso peggiore, è sufficiente il limite superiore $O(f(n))$). Se si utilizzano librerie di cui non si conosce la complessità, lo si indichi nella spiegazione (ad esempio, "la complessità è $O(n \log n)$ al netto della complessità dell'algoritmo x , che è non nota"). Se si utilizza la randomizzazione, si indichi anche il tempo di esecuzione atteso.

PROBLEMA

Si supponga di avere N classi di prodotti, e dobbiamo acquistare un solo prodotto per ogni classe. Si supponga di disporre di un budget B , e di voler spendere la massima quantità possibile ($\leq B$) per acquistare un solo prodotto per classe.

Si scriva un algoritmo che calcoli la massima quantità di denaro, minore o uguale ad B , che spendiamo per acquistare un prodotto per ogni classe. Nota: è possibile che il budget B non sia sufficiente ad acquistare un prodotto per ogni classe; in tal caso il programma stampi "*denaro insufficiente*".

INPUT

La prima riga dell'input contiene il numero N di casi di test.

Per ogni caso di test: la prima riga contiene due interi, il budget B ed il numero di classi di prodotti C ($1 \leq B \leq 200$, e $1 \leq C \leq 20$); le C righe successive contengono ciascuna un primo intero M ($1 \leq M \leq 20$), che indica il numero di prodotti differenti per ogni classe, seguito da M interi che indicano il prezzo di ogni prodotto. Ad esempio la riga 3 8 6 4 indica che ci sono 3 prodotti per quella classe, e che costano rispettivamente 8, 6 e 4.

OUTPUT

Per ogni test case, l'output da riportare è l'importo massimo di denaro necessario per acquistare un prodotto di ciascuna classe, senza superare il budget. Se non c'è soluzione, è necessario stampare "*denaro insufficiente*".

Sample Input

```
3
100 4
3 8 6 4
2 5 10
4 1 3 3 7
4 50 14 23 8
20 3
3 4 6 8
2 5 10
4 1 3 5 5
```

5 3

3 6 4 8

2 10 6

4 7 3 1 7

Sample Output

75

19

denaro insufficiente