

Esame di Algoritmi e Laboratorio

Modulo di Laboratorio

Appello del 9 dicembre 2020

Note

Completata la prova, spedire il codice sorgente, costituito da un singolo file chiamato “main.cpp”, all’indirizzo `santamaria@dmi.unict.it`. Prove inviate oltre l’orario concordato non saranno valutate. Indicare nell’oggetto del messaggio “Esame Laboratorio di Algoritmi”, nel corpo del messaggio nome, cognome, matricola e compilatore utilizzato.

Specifiche

Si implementi, sfruttando la tecnica di *Programmazione Dinamica*, un algoritmo per calcolare il minor numero possibile di monete di resto, disponendo di un numero illimitato di monete di diverso taglio.

L’input è suddiviso in task, uno per ogni riga. Ogni riga del file di input è formata da $2 + M$ elementi. I primi due elementi sono dei valori numerici interi, che rappresentano rispettivamente in numero M di tagli di monete disponibili e il resto da erogare. Seguono gli M tagli di monete disponibili.

L’output dovrà essere formato da una riga per ogni task. Qualora fosse possibile calcolare il resto, ogni riga conterrebbe $1+k$ valori. In tale ipotesi, il primo valore è il numero di monete da erogare, seguito dalle k monete necessarie ad erogare il resto, in ordine non decrescente. Qualora non fosse possibile calcolare il resto, ogni riga conterrebbe soltanto il valore -1 . Si noti che:

- Il sistema monetario potrebbe non essere *canonico*.
- Il numero di tagli di monete disponibili è sempre minore di 99.

Esempio

Il seguente esempio presenta un file di input, contenente 4 linee di input ed il corrispondente file di output (vedere file allegato).

```
6 42 5 7 3 8 1 4
3 6 4 3 1
```

2 6 5 2
3 8 3 6 7

Il corrispondente file di output sarà:

6 3 7 8 8 8 8
2 3 3
3 2 2 2
-1