

Traccia prova d'esame

L'organizzazione umanitaria E. Scrooge et Co. ha deciso, per il Natale del 2018, di distribuire dei pacchi regalo per il mondo, in modo da aiutare le persone più in difficoltà e donare un regalo a chi non può permetterselo. Nei database dell'organizzazione sono presenti varie directory compilate grazie alle varie sedi nel mondo: ogni directory contiene uno o più identificativi, ove un identificativo rappresenta univocamente una persona e due persone diverse hanno sempre identificativi diversi. Le directory non sono sincronizzate tra loro, quindi può capitare che una persona sia contenuta in più directory (fortunatamente con lo stesso identificativo) o anche che una persona sia contenuta più volte nella stessa directory. La E. Scrooge et Co. può stanziare dei fondi per almeno una directory e al più k e inviare un regalo ad ogni persona contenuta nell'insieme delle persone specificato dall'unione di queste directory. Ovviamente, se una persona si trova in due o più directory o più volte nella stessa directory, riceverà comunque un solo regalo. Inoltre, la E. Scrooge et Co., per motivi di marketing, vuole raggiungere almeno p persone distinte.

Sia S la lista di directory nel database dell'organizzazione e sia k il numero massimo di directory da poter scegliere e p il numero minimo di persone da raggiungere. Si deve scrivere un programma che scelga un sottoinsieme di directory T tale che

- Il numero di directory in T sia almeno uno e al più k ,
- Il numero di persone coperte da T sia almeno p .

Struttura dell'input

L'input consiste in un certo numero di righe. La prima riga è nel formato $n \ k \ p$, dove n è il numero di directory presenti nel database dell'organizzazione, k è il massimo numero di directory da poter scegliere e p il numero minimo di persone da coprire. Le successive n righe sono nel formato $d_i : id_1 id_2 id_3 \dots id_m$ dove d_i è una stringa rappresentante il nome della directory e i vari $id_1, id_2, id_3, \dots, id_m$ sono dei numeri interi (separati da uno spazio) dove ogni numero è l'identificativo di una persona. Si può assumere che l'input sia sempre corretto e che esista sempre almeno una soluzione.

Struttura dell'output

L'output della soluzione consiste in una riga, formata dai nomi delle directory in T separati da uno spazio.

Esempio input – output

```
3 2 6
ab : 4 8 12 8
xyz : 7 4 7 4 12 4
q : 320 11 7
```

```
ab q
```

Regole e istruzioni

- Si può scegliere se usare C++ o Java; in entrambi i casi, si presuppone che lo studente sappia compilare il codice sorgente e avviare l'eseguibile ottenuto tramite terminale/prompt dei comandi.
- Si può assumere che l'input sia sempre corretto.
- Il vostro programma deve **leggere da stdin** e **scrivere su stdout**. La lettura da file o l'hardcoding di un input nel vostro programma **non rappresenta una soluzione corretta**.
- Per effettuare le (vostre) varie prove, potete creare dei file testuali contenenti input di prova e
 - o Scrivere o copiare riga per riga il vostro input al programma,
 - o Reindirizzare il contenuto del file di input al vostro programma (consigliato).
- Il vostro programma verrà valutato su vari input utilizzando il secondo metodo.

Come posso reindirizzare su stdin?

- Supponendo di utilizzare C++ e di essere su Linux/OS X

```
cat input.txt | ./programma
```

dove input.txt è un file testuale contenente un input e programma è l'eseguibile ottenuto dalla compilazione del vostro codice sorgente (input.txt e programma devono essere nella stessa cartella)

- Supponendo di utilizzare C++ e di essere su Windows

```
type input.txt | programma.exe
```

dove input.txt è un file contenete un input e programma.exe è l'eseguibile ottenuto dalla compilazione del vostro codice sorgente (input.txt e programma.exe devono essere nella stessa cartella).