

Traccia prova d'esame

Gli shamani del regno di Kul Tiras sono alle prese con la creazione di nuovi incantesimi e per farlo hanno chiesto il nostro aiuto. Per creare nuovi incantesimi, essi usano una collezione di parole sacre. Data una collezione P di parole, e date tue parole iniziali p_1 e p_2 (appartenenti a P), il problema che ci chiedono di risolvere consiste nel trasformare la parola p_1 nella parola p_2 utilizzando il numero minimo di mosse valide possibili. Una mossa in una parola consiste nel cambiare un solo simbolo nella parola, e una mossa è valida quando la parola ottenuta appartiene a P . Ad esempio, se ho la collezione $\{luce, luse, duce\}$, per la parola $luce$ una possibile mossa valida è cambiare c in s , ottenendo $luse$ (che è nella collezione), mentre una mossa non valida è cambiare c in x , ottenendo $luxe$ (che non è nella collezione). Dati in input P , p_1 e p_2 , si deve restituire il numero minimo di mosse valide che si devono effettuare per trasformare p_1 in p_2 . Se non si può trasformare la parola p_1 in p_2 , allora si deve restituire -1 . Tutte le parole in P hanno la stessa lunghezza.

Struttura dell'input

La prima riga contiene tre valori n p_1 p_2 , dove n è il numero di parole in P , mentre p_1 e p_2 sono le parole iniziali. Ognuna delle successive n righe rappresenta una parola. Si assuma che tutte le parole abbiano la stessa lunghezza. Inoltre, le parole p_1 e p_2 sono anche contenute in due righe comprese tra le n in input.

Struttura dell'output

L'output consiste in un numero intero k rappresentante il numero minimo di mosse valide per trasformare la parola p_1 nella parola p_2 . Se non è possibile trasformare la parola p_1 in p_2 (ovvero non esiste un numero di mosse valido per farlo), si deve restituire -1 .

Esempi input – output

<pre>7 hope role peta atpe hope sole rope role sope</pre>	<pre>2</pre>
---	--------------

In questo esempio, la soluzione è data, applicando due mosse, da $hope$ a $rope$ a $role$.

<pre>5 abc def abc def axx yyf qbq</pre>	<pre>-1</pre>
--	---------------

In questo esempio, non esiste un numero minimo di mosse per trasformare abc in def .

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 contenente 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).