Laboratorio di Algoritmi

Progetto "Lessico" (novembre 2023)

Nota: La scadenza del progetto è fissata per venerdì 10 novembre **compreso**.

Nota: Si consiglia di controllare gli eventuali aggiornamenti di questo documento, che potrebbero riportare le risposte a dubbi degli studenti e la correzione di eventuali errori, e di consultare il documento con le indicazioni tassative e le informazioni utili per svolgere e consegnare correttamente il progetto.

Il problema Lo studio dei manoscritti antichi tenta di ricostruire la storia della loro trasmissione, confrontando parola per parola e lettera per lettera le versioni disponibili di ciascun testo. L'idea fondamentale è che il processo di copia a mano produca piccoli errori, che si accumulano via via di copia in copia. Di conseguenza, manoscritti fra loro più vicini sono in genere anche lessicalmente più simili. Insieme all'evidenza storica e a una serie di regole empiriche sulla probabilità di diversi generi di errore, queste analisi consentono di ipotizzare, se non di dimostrare, l'età di ciascun manoscritto, dunque la sua distanza dal testo originale.

Il progetto considera una "variante giocattolo" di questo campo di studi. Si ha un elenco di singole parole, anziché di manoscritti, e si vogliono analizzare le relazioni di "parentela" fra queste parole. Semplificando drasticamente, definiremo tale parentela come somiglianza nella loro forma scritta: due parole sono direttamente legate fra loro quando una si può ottenere dall'altra modificando, aggiungendo o sottraendo una singola lettera. Per esempio: `cane` e `pane` sono parole direttamente legate, e così pure lo sono `fato` e `fiato`. Ricostruita la relazione di parentela fra le parole dell'insieme dato, costruiremo una serie di indicatori che potrebbero essere utili nell'esecuzione di studi successivi.

Il progetto Il progetto richiede la stesura di un programma che legga da un file di testo un elenco di parole. Il file contiene una parola per ogni riga. Ogni parola è composta al massimo da l lettere, tutte minuscole, e tutte le parole sono diverse tra loro. Per semplificare la gestione delle stringhe di caratteri che rappresentano le parole, è lecito assumere $l = 30$ nel codice, ma **le analisi di complessità devono considerare l come non data a priori**. L'esempio seguente fornisce un breve elenco di 15 parole:

```
rutto
pilata
vane
casa
velo
vaso
pilota
cane
pelo
melo
pirata
rane
brutto
case
pineta
```

La relazione di parentela definisce due parole legate quando la seconda si può ottenere dalla prima modificando, aggiungendo o cancellando una sola lettera. Il

programma deve per prima cosa ricostruire la relazione e stampare su una riga il numero di parole seguito dalle parole chiave `parole con` e il numero di coppie in relazione seguito dalla parola chiave `parentele`[1]. Siccome la relazione di parentela è simmetrica, ogni coppia va contata una volta sola. Nell'esempio si ha:

```
15 parole con 11 parentele
```

Si deve quindi calcolare per ogni parola un indice di parentela pari il numero di altre parole di cui essa è parente. Per ciascun valore dell'indice si deve stampare in una diversa riga il numero di parole che lo possiedono seguito dalle parole chiave `parole di indice` e dal valore dell'indice. Le righe vanno ordinate per valori decrescenti dell'indice, saltando i valori a cui non corrisponde alcuna parola. Nell'esempio, una delle parole ha indice pari a 3, sette parole hanno indice pari a 2, cinque parole indice pari a 1 e le ultime due hanno indice di valore nullo. Quindi:

```
1 parole di indice 3
7 parole di indice 2
5 parole di indice 1
2 parole di indice 0
```

Oltre alle coppie di parole legate direttamente, la relazione di parentela crea gruppi di parole legate indirettamente, attraverso catene di parole parenti. I singoli gruppi sono invece tra loro isolati, cioè non è possibile raggiungere le parole dell'uno da quelle dell'altro attraverso parentele dirette. Si deve calcolare la cardinalità di ogni gruppo e il suo "diametro", cioè la massima "distanza" fra due parole del gruppo. Definiremo la distanza fra due parole come il minimo numero di passaggi richiesto per raggiungere l'una dall'altra. Al termine, si stamperanno i gruppi in ordine di cardinalità decrescente; i gruppi di pari cardinalità saranno ordinati per diametro decrescente; in caso di ulteriore parità si porranno i gruppi in ordine alfabetico rispetto alla prima parola di ciascun gruppo[2]. Il formato di stampa richiesto è il seguente. La prima riga fornisce il numero di gruppi seguito dalla parola chiave `gruppi`. Seguono le informazioni dei singoli gruppi, uno per riga. Per ciascuno si stampa la cardinalità seguita dalle parole chiave `parole da`, dalla prima parola del gruppo in ordine alfabetico, dalla parola chiave `a`, dall'ultima parola del gruppo in ordine alfabetico (eventualmente, coincidente con la prima, se ve n'è una sola), dalle parole chiave `con diametro` e dal valore del diametro. Nel consueto esempio, l'elenco genera 6 gruppi. Il più numeroso consiste di 5 parole (`cane`, `casa`, `case`, `rane`, `vane`) e ha diametro pari a 3, perché trasformare `casa` in `vane` richiede un minimo di tre passaggi: `casa -> case -> cane -> vane`. Si avrà quindi:

```
7 gruppi
5 parole da cane a vane con diametro 3
3 parole da pilata a pirata con diametro 2
3 parole da melo a velo con diametro 1
2 parole da brutto a rutto con diametro 1
1 parole da pineta a pineta con diametro 0
1 parole da vaso a vaso con diametro 0
```

dove, ovviamente, i gruppi di una sola parola hanno diametro nullo, perché non occorrono passaggi per raggiungere l'unica parola da sé stessa.

Per concludere, si stamperà l'intero elenco di parole in ordine alfabetico, una parola per riga. Nell'esempio considerato si ha:

[1] Le parole chiave restano plurali anche qualora i valori numerici fossero unitari

[2] Per ordinamento alfabetico si intende quello generato dalla funzione di confronto `strcmp` della libreria standard del C.

```
brutto
cane
casa
case
melo
pelo
pilata
pilota
pineta
pirata
rane
rutto
vane
vaso
velo
```

Chiarimenti In questa sezione saranno riportate le risposte a domande e dubbi.