

# Programmazione I – Informatica Generale

## Prova di Programmazione - 10 Giugno 2009 – 2 ore 15 min

Definiamo *raccolta di lettere* un insieme di lettere minuscole in cui ciascuna lettera può apparire più volte. Rappresentiamo tale insieme come una sequenza di coppie (*lettera, numero occorrenze*), ordinate in base al campo *lettera* delle coppie ed in cui, per ciascuna coppia, *numero occorrenze* rappresenta il numero di volte in cui *lettera* appare nella raccolta. Ad esempio  $\{(a, 3), (d, 5), (j, 1)\}$  è una raccolta in cui sono presenti tre occorrenze della lettera *a*, cinque della lettera *d* ed una della lettera *j*. Definiamo *sequenza di lunghezza l* un vettore di *l* raccolte di lettere. Ad esempio  $[\{(a, 3), (d, 5), (j, 1)\}, \{(k, 10), (z, 3)\}]$  è una sequenza di lunghezza 2. Partendo dal frammento di codice fornito, scrivere un programma che fornisca le seguenti funzionalità:

1. Inizializzazione di una raccolta di lettere contenente solo *m* occorrenze di una data lettera.
2. Stampa di una raccolta di lettere come sequenza di coppie (*lettera, numero occorrenze*), ordinate in base al campo *lettera* e separate da virgole. Se, ad esempio, la raccolta contiene solo cinque occorrenze della lettera *g*, si dovrà stampare  $\{(g, 5)\}$ , mentre se la raccolta contiene tre a e quattro d, si dovrà stampare  $\{(a, 3), (d, 4)\}$ .
3. Aggiunta di *m* occorrenze di una data lettera ad una raccolta di lettere. Ad esempio aggiungendo due *a* alla raccolta  $\{(a, 3), (d, 4)\}$  si ottiene  $\{(a, 5), (d, 4)\}$ , mentre aggiungendo tre *b* alla stessa raccolta si ottiene  $\{(a, 3), (b, 3), (d, 4)\}$ .
4. Salvataggio di una raccolta di lettere in un file di testo.
5. Caricamento di una raccolta di lettere da un file di testo.
6. Inizializzazione di una sequenza di lunghezza *l* a partire da una stringa della stessa lunghezza: ogni elemento della sequenza sarà inizializzato con una raccolta di lettere contenente solo una occorrenza della corrispondente lettera nella stringa. Ad esempio, l'inizializzazione di una sequenza di lunghezza tre con la stringa “*not*” inizIALIZZEREBBE la sequenza a  $[\{(n, 1)\}, \{(o, 1)\}, \{(t, 1)\}]$ . Si assuma che questa funzionalità venga invocata una sola volta su una sequenza non inizializzata.
7. Stampa di una sequenza.
8. Aggiunta di una stringa di lunghezza *l* ad una sequenza di lettere della stessa lunghezza: ad ogni elemento della sequenza si aggiunge un'occorrenza della corrispondente lettera nella stringa. Ad esempio, l'aggiunta della stringa “*xor*” alla sequenza  $[\{(n, 1)\}, \{(o, 1)\}, \{(t, 1)\}]$  porta alla sequenza  $[\{(n, 1), (x, 1)\}, \{(o, 2)\}, \{(r, 1), (t, 1)\}]$ .

Gestire opportunamente le situazioni di errore, tranne l'*overflow* e l'inserimento di dati in formato errato da *stdin*.

---

### REGOLE

- Si può utilizzare ogni genere di materiale di manuale o altro materiale didattico
- Per superare la prova, il programma deve essere perfettamente funzionante almeno nelle parti 1, 2, 3 e 4. Il voto ottenuto in questo caso è 18.
- Ciascuna funzionalità DEVE essere implementata mediante almeno una funzione.
- Il voto massimo (almeno 30) si ottiene se
  - a) il programma è perfettamente funzionante in ogni sua parte
  - b) tutti i principi di ingegneria del codice visti nel corso sono stati applicati