

# Programmazione I – Informatica Generale

## Prova di Programmazione - 11 Febbraio 2009 - 2ore15min

Partendo dal frammento di codice fornito, scrivere un programma di ricerca di sottostringhe all'interno di un vettore di parole, ove per parola intendiamo una stringa non contenente spazi. Le parole hanno tutte la stessa lunghezza massima, nota a tempo di scrittura del programma. Inoltre la posizione dei caratteri all'interno delle parole è numerata a partire da 0. All'avvio il vettore di parole è vuoto. Il programma deve fornire le seguenti funzionalità:

1. Inserimento delle parole. Si chiede all'utente il numero di parole da inserire, quindi il contenuto di ciascuna parola (notare che un array dinamico **a** di  $N$  stringhe di lunghezza massima  $M$  si può definire ed allocare con l'istruzione `char (*a)[M+1] = new char [N][M+1];`, inoltre, per passare l'oggetto **a** per riferimento, è necessario un parametro formale, che chiamiamo per esempio **v**, definito come `char (* &v)[M+1]`). Ogni nuovo inserimento comporta la perdita del precedente contenuto del vettore. Trascurare problemi di spreco della memoria nel caso di parole molto più corte della lunghezza massima.
2. Stampa del vettore.
3. Salvataggio del vettore su di un file di testo dal nome definito a tempo di scrittura del programma.
4. Caricamento del vettore da un file di testo dal nome definito a tempo di scrittura del programma. Il precedente contenuto del vettore viene perso.
5. Ricerca di una sottostringa data all'interno della  $i$ -esima parola, e ritorno della prima posizione in cui compare (il primo carattere de) la sottostringa, se presente; o della comunicazione di fallimento (non effettuare alcuna stampa della sottostringa all'interno della funzione che realizza la ricerca). Se ad esempio il vettore contiene le parole “lampone”, “ananas” e “mela”, la ricerca della sottostringa “anas” nella seconda parola ritornerà la posizione 2, mentre la ricerca della sottostringa “ele” nella terza parola fallirà.
6. Stampa di tutte le sottostringhe di lunghezza data della  $i$ -esima parola. Per ciascuna sottostringa, stampare inoltre la posizione in cui appare all'interno della parola (il primo carattere della sottostringa). Se ad esempio il vettore contiene di nuovo le parole “lampone”, “ananas” e “mela”, la stampa delle sottostringhe di 3 caratteri della seconda parola deve produrre:  
`(ana, 0)`  
`(nan, 1)`  
`(ana, 2)`  
`(nas, 3)`

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, e 3. 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