

Programmazione I

Prova di Programmazione – 3 febbraio 2010 – 2ore 40 min

Partendo dal frammento di codice fornito, scrivere un programma di gestione di un *crucipuzzle*, in cui bisogna trovare delle parole in una matrice quadrata di lettere maiuscole di dimensioni definite a tempo di scrittura del programma. Le parole da trovare possono essere disposte in orizzontale, verticale o in diagonale, e vanno lette da sinistra verso destra oppure dall'alto in basso. Il programma deve fornire le seguenti funzionalità:

1. **nuova_partita** Si legge da *stdin* il contenuto della matrice, il numero di parole da trovare e le parole da trovare (che si da per scontato siano presenti nella matrice). Supporre che il numero massimo possibile di parole da trovare sia definito a tempo di scrittura del programma. Prima che questa funzionalità sia stata invocata per la prima volta, la matrice contiene solo lettere **A** e non c'è nessun parola da trovare. Ogni volta che questa funzionalità viene invocata la precedente partita in corso è persa. Esempio: si potrebbe inserire la matrice

T	R	E
S	F	V
R	E	A

 e le parole TRE, RE, SE ed EVA da trovare.
2. **stampa_partita** Si stampa la matrice di caratteri e le parole da trovare.
3. **salva_partita** Si salvano la matrice e la parole da trovare in un file dal nome predefinito.
4. **carica_partita** Si caricano la matrice e l'elenco delle parole da trovare dal file. La precedente partita è persa.
5. **confronta_parola(par, pos, dir)** Ritorna vero se la parola **par** è presente nella matrice a partire dalla posizione **pos** e lungo la direzione **dir**. Ad esempio, data la matrice dell'esempio al punto 1, dovrebbe ritornare vero per ciascuno dei seguenti confronti: parola TRE con la parola presente a partire dall'angolo in alto a sinistra ed in direzione orizzontale, parola SE con la parola presente a partire dalla prima colonna/seconda riga ed in direzione diagonale.
6. **trova_parola(par, pos, dir)** Se la parola **par** è una di quelle da trovare, si controlla se è presente nella matrice a partire dalla posizione **pos** e lungo la direzione **dir**, e se le cose stanno così si elimina tale parola dall'elenco delle parole da trovare e, se non ci sono più parole da trovare comunica l'avvenuta vittoria.
7. **[PUNTO EXTRA] elenca_pos_parola(par, dir)** Stampa su *stdout* tutte le posizioni in cui la parola **par** compare nella matrice nella direzione **dir**. Ciascuna posizione va stampata su una riga separata, con gli indici di riga e colonna separati dallo spazio. Se ad esempio, data la matrice dell'esempio al punto 1, si immettesse la parola RE e la direzione orizzontale, si dovrebbe stampare
0 1
2 0

I parametri di ingresso delle funzionalità sono solo indicativi. Gestire opportunamente le situazioni di errore, tranne gli *overflow* dovuti a valori assoluti troppo elevati e l'inserimento di dati in formato errato da *stdin*.

REGOLE

- Si può utilizzare ogni genere di manuale o di 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