

Programmazione I
Prova di programmazione – 28 Gennaio 2021 – 2 ore

Partendo dal frammento di codice fornito, realizzare un programma per la gestione di una matrice di parole. Le dimensioni della matrice sono note a tempo di scrittura del programma. La lunghezza massima di ciascuna parola nella matrice non è invece nota a tempo di scrittura del programma, e può differire dalla lunghezza di una delle altre parole. All'avvio del programma, le parole sono tutte vuote, ossia hanno lunghezza zero. Il programma fornisce le seguenti funzionalità.

1. **sostituisci_parola(p, i, j)** Sostituisce, con la parola **p**, la parola presente nella posizione **(i, j)** nella matrice. La parola **p** deve avere lunghezza almeno uno.
2. **[1 punto extra] stampa_matrice** Stampa la matrice di parole, separate da spazi e con le parole vuote rappresentate da uno spazio. Si ottiene il punto extra se si stampa ciascuna parola con una larghezza pari al numero di caratteri della parola più lunga presente nella matrice, giustificando ciascuna parola a sinistra o a destra (equivalente), ed usando il carattere spazio come riempitivo (non usare altri caratteri rispetto allo spazio). Ecco un esempio con matrice 3x2, e con due parole vuote:

```
Pippo      Paperino
Minnie
```

Quo

Se utile, si ricordano i manipolatori **setw(int n)** per stabilire il numero di caratteri per la prossima operazione di uscita, e **setfill(char c)** per decidere il carattere di riempimento. Entrambi sono presentati in *iomanip*.

3. **[3] salva_matrice** Salva la matrice in un file BINARIO dal nome predefinito. Tralasciare il salvataggio delle dimensioni della matrice. Ricordare che salvare l'indirizzo contenuto in un puntatore ad un array non equivale a salvare l'array stesso.
4. **[3] carica_matrice** Carica la matrice dal file. L'eventuale precedente contenuto è perso.
5. **[3] scambia(i, j, i2, j2)** Scambia la parola in posizione **(i, j)** con la parola in posizione **(i2, j2)**. Ad esempio, dato il contenuto mostrato nell'esempio al punto 2, se si scambia la parola in posizione (0, 1) con la parola in posizione (2, 1), il nuovo contenuto della matrice diviene:

```
Pippo      Quo
Minnie
```

Paperino

6. **[5] scambia_con_minimo(i, j)** Scambia la parola in posizione **(i, j)** con la parola non vuota di lunghezza minima tra quelle successive all'interno della matrice. Se non viene trovata alcuna parola non vuota, non si effettua alcuno scambio. Ad esempio, se si invoca questa funzionalità per la parola in posizione (0, 0) nella matrice mostrata nell'esempio al punto 2, si ottiene:

```
Quo      Paperino
Minnie
```

Pippo

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

Per ottenere un output compatibile con il tester, definire matrici 3x2.

REGOLE

- Si può utilizzare ogni genere di manuale e di materiale didattico
- Per superare la prova, bisogna svolgere almeno i punti 1 e 2. Se si svolgono solo tali punti, il programma deve essere perfettamente funzionante. 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
 - c) sono state seguite eventuali altre indicazioni presenti nella traccia in merito al voto finale