

Scrabble (noto anche come “Scarabeo”) è un gioco da tavolo in cui il giocatore deve inserire delle parole all’interno di un tabellone avente $N \times N$ caselle. Le parole vengono inserite a partire da una data casella in direzione orizzontale o verticale. Ogni lettera della parola occupa una casella. Sono possibili incroci con parole già presenti nel tabellone, a patto che l’incrocio avvenga su lettere uguali.

Implementare le seguenti operazioni che possono essere fatte su Scrabble:

--- **PRIMA PARTE** --- *(qualora siano presenti errori di compilazione, collegamento o esecuzione in questa parte, l’intera prova sarà considerata insufficiente e pertanto non sarà corretta)*

✓ **Scrabble s(N);**

Costruttore che crea un tabellone di $N \times N$ caselle (con $N \geq 4$). Tutte le caselle sono inizialmente vuote.

✓ **s.aggiungi(str, r, c, dir);**

Operazione che aggiunge la parola `str` a partire dalla casella di coordinate (r, c) , nella direzione indicata da `dir` (le righe e le colonne sono numerate a partire da 0). In particolare, `dir` può assumere i valori ‘O’ o ‘V’ (lettere maiuscole), per indicare rispettivamente la direzione orizzontale e verticale. L’operazione può fallire nel caso in cui non ci sia spazio sufficiente per inserire la parola o se la parola non può incrociarsi con le altre parole già presenti sul tabellone. In caso di fallimento, il tabellone rimane inalterato.

✓ **s1 = s;**

Operatore di assegnamento per il tipo `Scrabble`, che sostituisce il contenuto di `s1` con quello di `s`.

✓ **cout << s;**

Operatore di uscita per il tipo `Scrabble`. L’output mostrato a video per un tabellone di dimensione 7×7 , nel quale sono state aggiunte 4 parole, ha il seguente formato (il carattere ‘-’ indica le caselle vuote):

```
- - - - t - -
- - a e r e o
- - - - e - -
- - m - n - -
- - o - o - -
a u t o - - -
- - o - - - -
```

--- **SECONDA PARTE** ---

✓ **s.esiste(str);**

Operazione che controlla se la parola `str` è presente nel tabellone. La parola può trovarsi in senso orizzontale o verticale. La funzione restituisce `true` se la parola esiste, `false` altrimenti.

✓ **!s;**

Operazione che restituisce il massimo numero di caselle vuote consecutive nel tabellone, in senso orizzontale o verticale. Nota: l’ultima casella di una riga(colonna) e la prima della riga(colonna) successiva non sono da considerarsi come consecutive. Nell’esempio, la funzione deve restituire il valore 5.

```
- - - - t - -
- - a e r e o
- - - - e - -
- - m - n - -
- - o - o - -
a u t o - - -
- - o - - - -
```

✓ ~Scrabble();

Distruttore per la classe Scrabble.

Mediante il Linguaggio C++, realizzare il tipo di dato astratto **Scrabble**, definito dalle precedenti specifiche. **Gestire le eventuali situazioni di errore.**

USCITA CHE DEVE PRODURRE IL PROGRAMMA

Test del costruttore

```
- - - - -  
- - - - -  
- - - - -  
- - - - -  
- - - - -  
- - - - -  
- - - - -  
- - - - -
```

Test s.aggiungi

```
- - - - t - -  
- - a e r e o  
- - - - e - -  
- - m - n - -  
- - o - o - -  
a u t o - - -  
- - o - - - -
```

Test s1.aggiungi

```
- c - - - -  
g a t t o -  
- n - - - -  
- e - - - -  
- - - - -  
- m u c c a
```

Test dell'operatore di assegnamento s1 = s

```
- - - - t - -  
- - a e r e o  
- - - - e - -  
- - m - n - -  
- - o - o - -  
a u t o - - -  
- - o - - - -
```

Primo test s1.esiste

La parola cercata esiste! :)

Secondo test s1.esiste

La parola cercata non esiste! :(

Test operator!

5

Test del distruttore

(s2 e' stato distrutto)