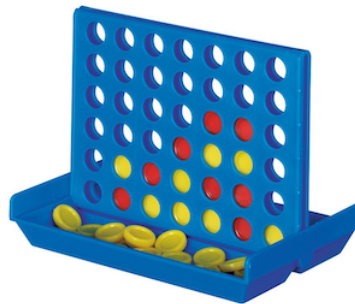


Mini progetto - Forza Quattro

Scrivete un programma che permetta a due persone di giocare al gioco "Forza quattro"



Forza Quattro

Il tabellone è costituito da una griglia di 7 colonne e 6 righe. A turno, i giocatori inseriscono un gettone in una delle colonne che non sia già piena. Il gettone cadrà nella colonna, occupando la prima posizione libera disponibile. Vince il giocatore che riesce a formare una serie consecutiva di 4 gettoni verticalmente, orizzontalmente o diagonalmente.

Descrizione

Il gioco dovrà svolgersi nel modo seguente:

1. All'inizio, i due giocatori dovranno inserire il loro nome ed il simbolo con il quale verranno identificati i propri gettoni.
2. Dovrà essere richiesto il numero totale di match che si intendono giocare, fino ad un massimo di 15. Per evitare condizioni di parità, in numero di match dovrà essere dispari.
3. Ad ogni match, dovrà essere sorteggiato casualmente il giocatore che inizia per primo.
4. A turno, ogni giocatore dovrà scegliere una colonna in cui inserire il suo gettone (da 1 a 7).
5. Se la colonna selezionata risulterà già essere piena, ne dovrà essere selezionata immediatamente un'altra.
6. Ad ogni mossa compiuta, il tabellone dovrà essere controllato per vedere se il giocatore ha generato una serie continua di 4 gettoni in orizzontale, verticale o in diagonale. In caso affermativo, il giocatore si aggiudicherà il match.
7. Nel caso il tabellone si dovesse riempire senza alcun vincitore, il match terminerà senza vincitore e dovrà essere ripetuto.

8. Quando tutti i match saranno stati giocati, verrà visualizzato a schermo il numero di match vinti da ciascun giocatore e la proclamazione del vincitore della partita.

Struttura

Il vostro programma dovrà essere composto almeno dalle seguenti classi:

- **Giocatore**: classe che rappresenta il giocatore che partecipa alla partita. Ogni giocatore è identificato dal nome e dal simbolo del gettone.
- **Match**: classe che si occupa di gestire il match tra due giocatori seguendo le regole descritte sopra.
- **Partita**: classe che si occupa di gestire la partita, dalla creazione dei giocatori alla gestione dei match.
- **GestioneInput**: classe di utilità che gestisce l'input da parte dell'utente. Dovrà esserne un'unica istanza in tutto il programma, e dovrà essere passata ove necessario alle classi che ne faranno uso tramite parametro del costruttore. Dovrà essere presente un metodo, `void rilascia()`, che al suo interno chiuda lo scanner. Questo metodo dovrà essere chiamato come ultima istruzione dell'intero programma.

Regole

- Per lo sviluppo del progetto, sfruttate al meglio tutte le conoscenze che avete acquisito nel corso del semestre. Questo vale non solo per la teoria, ma anche per tutte le best practices che abbiamo discusso (convenzioni, validazioni ecc...).
- Siete incoraggiati ad introdurre vostre personalizzazioni, stando attenti a non compromettere le logiche e le funzionalità del gioco richieste.
- Il progetto è, in linea di principio, **a svolgimento individuale**, ogni studente dovrà creare la propria consegna tramite l'apposita sezione sulla pagina iCorsi, e sarà responsabile del codice fornito. È perfettamente lecito confrontarsi e darsi una mano a vicenda, basta che sia fatto con **cognizione di causa**: in caso di blocchi di codice troppo simili tra due persone, potrebbe essere richiesto ad entrambi di spiegare in dettaglio il funzionamento degli stessi.
- L'utilizzo di generatori di codice (Copilot, ChatGPT, Google Bard e affini) tramite mero copia/incolla è fortemente scoraggiato. È abbastanza facile capire se uno snippet di codice arriva da una di queste piattaforme. Anche in questo caso potrebbe essere richiesta una spiegazione approfondita dei blocchi di codice sospetti.

Esempio di output

Di seguito un esempio di output. **L'output non deve necessariamente essere identico, basta mostrare le informazioni richieste:**

```
Inserisci il nome del giocatore 1
Andrea
Inserisci il simbolo
A
Inserisci il nome del giocatore 2
Loris
Inserisci il simbolo
L
Quante partite vuoi fare? (1-15, solo numeri dispari)
1
```

```
  1  2  3  4  5  6  7
-----
|  |  |  |  |  |  |
-----
|  |  |  |  |  |  |
-----
|  |  |  |  |  |  |
-----
|  |  |  |  |  |  |
-----
|  |  |  |  |  |  |
-----
|  |  |  |  |  |  |
-----
|  |  |  |  |  |  |
-----
```

```
È il turno di Loris
Scegli una colonna (1-7)
1
```

```
  1  2  3  4  5  6  7
-----
|  |  |  |  |  |  |
-----
|  |  |  |  |  |  |
-----
```


	L							

È il turno di Andrea
Scegli una colonna (1-7)
2

1	2	3	4	5	6	7

	L	A				

È il turno di Loris
Scegli una colonna (1-7)
1

1	2	3	4	5	6	7

| L | | | | | | |

| L | A | | | | | |

È il turno di Andrea

Scegli una colonna (1-7)

2

1 2 3 4 5 6 7

| | | | | | | |

| | | | | | | |

| | | | | | | |

| | | | | | | |

| L | A | | | | | |

| L | A | | | | | |

È il turno di Loris

Scegli una colonna (1-7)

1

1 2 3 4 5 6 7

| | | | | | | |

| | | | | | | |

| | | | | | | |

| L | | | | | | |

| L | A | | | | | |

```
| L | A |   |   |   |   |   |  
-----
```

È il turno di Andrea

Scegli una colonna (1-7)

2

```
  1  2  3  4  5  6  7  
-----
```

```
|   |   |   |   |   |   |   |  
-----
```

```
|   |   |   |   |   |   |   |  
-----
```

```
|   |   |   |   |   |   |   |  
-----
```

```
| L | A |   |   |   |   |   |  
-----
```

```
| L | A |   |   |   |   |   |  
-----
```

```
| L | A |   |   |   |   |   |  
-----
```

È il turno di Loris

Scegli una colonna (1-7)

1

```
  1  2  3  4  5  6  7  
-----
```

```
|   |   |   |   |   |   |   |  
-----
```

```
|   |   |   |   |   |   |   |  
-----
```

```
| L |   |   |   |   |   |   |  
-----
```

```
| L | A |   |   |   |   |   |  
-----
```

```
| L | A |   |   |   |   |   |  
-----
```

```
| L | A |   |   |   |   |   |  
-----
```

Il giocatore Loris ha vinto il match!

Il giocatore Andrea ha vinto 0 match
Il giocatore Loris ha vinto 1 match
IL VINCITORE DELLA PARTITA È LORIS