

Il gioco del Tris

Progetto di programmazione procedurale in C#

Specifiche in sintesi del progetto:

- 3 modalità di gioco, selezionabili in alternativa ad inizio gioco.
- Interfaccia grafica mediante WPF application.
- Relazione del progetto: va esplicitata una breve documentazione delle scelte implementative effettuate.
- Effetto speciale (facoltativo): un elemento da aggiungere a discrezione di ciascuno che rappresenti un valore aggiunto, cioè la propria “firma”.

1° modalità di gioco: giocatore umano contro giocatore umano

- Scrivere un programma che realizza il gioco del tris (o filetto) tra due giocatori umani.
- In particolare, ciascun giocatore ha un simbolo ed una mossa consiste nel posizionare, a turno, il proprio simbolo in una cella libera.
- Vince chi per primo riesce a comporre una sequenza di tre simboli uguali, allineati in orizzontale, verticale oppure diagonale.
- Se la scacchiera viene riempita senza alcun vincitore, è pareggio

2° modalità di gioco: giocatore umano contro giocatore virtuale che utilizza una tattica prestabilita

- Modificare il programma precedente in modo che uno dei due giocatori sia virtuale, e conduca la partita seguendo le seguenti tattiche prestabilite.
- Controlla prima se può vincere (provando a posizionare il proprio simbolo in una cella libera (e provando per tutte le celle libere) per vedere se è possibile un tris in una sola mossa. In tal caso la esegue.
- Se non è possibile vincere in una sola mossa, il giocatore virtuale prova a vedere se è l'avversario a poter vincere in una sola mossa occupando una data cella. In tal caso cerca di ostacolarlo occupando la cella mediante il proprio simbolo. Se vi fossero due celle di questo tipo l'avversario vincerebbe comunque, in quanto in una sola mossa se ne potrebbe occupare una soltanto.
- Se non avviene alcuno dei casi precedenti, il giocatore esegue una mossa casuale.

3° modalità di gioco: giocatore umano contro giocatore artificiale in grado di apprendere

- Modificare il programma precedente in modo che il giocatore virtuale (macchina) conduca la partita imitando il giocatore reale (umano). In particolare, la macchina memorizza tutte le mosse del giocatore reale in un file. Quando è il suo turno, cerca in tale file se in passato (nelle partite precedenti) vi è stata una situazione identica per l'umano. In tal caso esegue la medesima mossa che eseguì l'umano, altrimenti esegue una mossa a caso.
- Per semplicità, la macchina memorizza ogni mossa senza controllare se la medesima mossa era stata già memorizzata, quindi è possibile che vi siano dei duplicati. Inoltre essa memorizza ogni mossa senza controllare che sia una mossa buona o cattiva, per cui l'umano dovrebbe essere un giocatore esperto.
- Per fare in modo che le situazioni presentate all'umano si ripropongano alla macchina, la scelta di chi inizia per primo è casuale.
- Per ogni mossa, viene salvato nel file lo stato della scacchiera e le coordinate della mossa eseguita. Lo stato della scacchiera va codificato in modo efficiente.

Compiti vari:

- Implementazione in C#
- Realizzazione dell'interfaccia grafica
- Documentare il lavoro svolto

Il codice sorgente sviluppato in C#:

- Deve essere originale, ovvero non copiato da altri studenti, da siti web
- Deve essere compilabile senza che vengano segnalati errori o messaggi di warning.
- Deve essere leggibile, ovvero:
 - privo di identificatori (cioè variabili) non evocativi di ciò che rappresentano
 - ben strutturato
 - ben commentato
 - ben indentato
 - ben spaziato
- Deve essere basato sui principi della programmazione modulare:
 - È consigliato e incoraggiato il riutilizzo del codice.

Ogni inosservanza di quanto stabilito sopra a proposito della preparazione della relazione e dello sviluppo del software determina una riduzione nel voto attribuito al progetto.