TetrAIs

|  |  |  |
| --- | --- | --- |
| **Christian Miccolis** | **Davide Paduanelli** | **Mattia Patruno** |
| Matricola: 683313 | Matricola: | Matricola: |

Indice

**1 Introduzione ……………………………………………………………………. 2**

**2 Funzionalità ……………………………………………………………………….**

**3 Intelligenze Artificiali ……………………………………………………………**

3.1 Deep First Search …………………………………………………………………..

3.2 Stochastic Gradient Descent (Q-Learning) …………………………………….

3.3 Genetico …………………………………….…………………………………….….

3.4 Blind Bandit Monte Carlo ………………………………………………………...

3.5 Basato su Regole logiche …………………………………………………………..

3.6 Ricerca Locale …………………………………….…………………………………

**4 Implementazione …………………………….…………………………………**

**5 Valutazione e confronto ……………………….…………………………………**

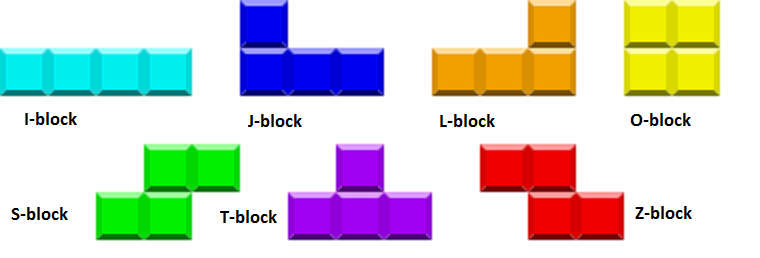
**6 Conclusioni …………………………………….…………………………………**

**7 Bibliografia …………………………………….…………………………………**

**1. Introduzione**

Tetris è un videogioco classico pubblicato a metà degli anni '80. Il gioco prevede la disposizione di sette diversi tipi di blocchi (anche noti come “Tetramini”) per creare righe lungo l'area di gioco. I blocchi cadono dall'alto verso il basso sullo schema di gioco e possono solo essere ruotati o spostato a sinistra o a destra dal giocatore. Ogni volta che viene posizionato un blocco, uno nuovo blocco casuale inizierà a scendere dalla parte superiore dello schermo. A seconda della variante del gioco, è possibile che lo schema di gioco sia a conoscenza del tipo di blocco successivo oppure no. I blocchi cadono in una griglia alta 20 quadrati e larga 10. Ogni volta che una riga viene occupata interamente, essa scompare lasciando scoperta la riga inferiore. Quando la pila di blocchi raggiunge la cima alla griglia, il gioco termina. Poiché i blocchi in arrivo non possono essere previsti, il pianificatore di blocchi deve essere in grado di adattarsi a diversi modelli. Il gioco tetris risulta essere un gioco invincibile in modo complessivo, poiché dipende strettamente dalla sequenza di blocchi che vengono generati, di conseguenza un eventuale combinazione infausta costituita da blocchi come la S e la Z condurrebbe la partita ad una conclusione rapida e inevitabile.

* I sette “Tetramini” presenti nel gioco:



* La schermata di gioco fi TetrAIs:



**2. Funzionalità**

TetrAIs dispone di alcune funzionalità aggiuntive che permettono maggiore comprensione riguardo le azioni svolte dalle intelligenze artificiali durante la loro esecuzione.

2.1 Guide Side Panel

2.2 Decisional Tree Plot

2.3 Result Tree Plot

2.4 Real Time Console Prints

**3. Intelligenze Artificiali**

TetrAIs ha a disposizione sei differenti agenti autonomi basati su sei differenti algoritmi di intelligenza artificiale, in grado di operare sullo schema di gioco seguendo diversi approcci per l’ottenimento dello score più alto possibile.

3.1. Deep First Search

….

3.2. Stochastic Gradient Descent (Q-Learning)

…

3.3. Genetico

…

3.4. Blind Bandit Monte Carlo

…

3.5 Basato su Regole logiche

…

3.6. Ricerca Locale

…

**4. Implementazione**

L’ implementazione adottata per TetrAIs è basata interamente sull’utilizzo del linguaggio python 3, sia per il lato (front -end) grafico sia per la parte (back-end) dedicata al ragionamento, controllo e scelta delle mosse da svolgere, l’unica eccezione è rappresentata dall’AI Basata su regole logiche che presenta un bridge per la comunicazione con una base di conoscenza scritta nel linguaggio prolog (Swi-prolog)

3.1. Deep First Search

….

3.2. Stochastic Gradient Descent (Q-Learning)

…

3.3. Genetico

…

3.4. Blind Bandit Monte Carlo

…

3.5 Basato su Regole logiche

…

3.6. Ricerca Locale

…

**5. Valutazione e confronto**

… sezione “core” dedicata ai grafici e i risultati ottenuti durante i test

**6. Conclusioni**

… resoconto sulle AI

**7. Bibliografia**

… riferimenti e link a risorse utili