



Οικονομικό Πανεπιστήμιο Αθηνών

Τμήμα Πληροφορικής

Χειμερινό Εξάμηνο 2018-2019

Τεχνητή Νοημοσύνη – 1η εργασία

Διδάσκων: Ι. Ανδρουτσόπουλος

Θεοδώρου Ιωάννης (ΑΜ : 3150046)

ΕΙΣΑΓΩΓΗ

Στην παρούσα εργασία, κληθήκαμε να κατασκευάσουμε ένα πρόγραμμα που θα δίνει τη δυνατότητα στον χρήστη του να παίζει το παιχνίδι Reversi (γνωστό και ως Othello) εναντίον του υπολογιστή. Το πρόγραμμα αυτό χρησιμοποιεί τον αλγόριθμο MiniMax (QQ με πριόνισμα α - β), το μέγιστο βάθος αναζήτησης του οποίου μπορεί να καθορίσει ο χρήστης κατά την έναρξη του παιχνιδιού, προσαρμόζοντας έτσι και το επίπεδο δυσκολίας της κάθε παρτίδας.

Το παρόν πρόγραμμα έχει υλοποιηθεί σε γλώσσα Java.

ΑΡΧΙΤΕΚΤΟΝΙΚΗ

Το πρόγραμμά μας αποτελείται από τα εξής αρχεία:

- Board.java
- reversi.java
- Heureticals.java
- MinMax.java

Η κλάση *Board* χρησιμοποιείται για την κατασκευή και αρχικοποίηση του επιτραπέζιου παιχνιδιού. Για τον σκοπό αυτό, δηλώνεται ένας δισδιάστατος πίνακας που αναπαριστά το ταμπλό, καθώς και τα πιόνια με τις αξίες τους. Συγκεκριμένα, τα λευκά πιόνια έχουν αξία (-1) και τα μαύρα (+1), έτσι ώστε να δημιουργηθούν οι κατάλληλες ευρετικές συναρτήσεις. Ταυτόχρονα, οι κενές θέσεις στο ταμπλό ορίζονται κι αυτές ως πιόνια με μηδενική αξία. Επιπλέον, η κλάση περιέχει τους κατάλληλους constructors για την δημιουργία και αντιγραφή των διαφόρων καταστάσεων (δηλαδή των ταμπλό με τα πιόνια στις κατάλληλες θέσεις, μετά από κάθε κίνηση), καθώς και ελέγχους για τη δυνατότητα πραγματοποίησης κάθε πιθανής κίνησης. Τέλος, η κλάση περιέχει μεθόδους που επιτρέπουν στον υπολογιστή να κάνει κίνηση, εφόσον έχει τη δυνατότητα, και τυπώνουν στη κονσόλα τις νέες καταστάσεις που προκύπτουν μετά από κάθε κίνηση, με τη μορφή ενός ταμπλό.

Η κλάση *Reversi* περιέχει την μέθοδο *main*, η οποία αποτελεί την αφετηρία του προγράμματός μας. Σε αυτήν υλοποιείται η διεπαφή του προγράμματος με τον χρήστη, καλώντας τον να επιλέξει το επίπεδο δυσκολίας (μέσω του βάθους του αλγορίθμου Minimax) και τη σειρά που θα παίξει, καθώς και το σημείο που θα τοποθετήσει τα πιόνια του.

Η κλάση *Heureticals* περιέχει τις ευρετικές συναρτήσεις που χρησιμοποιούμε στην κλάση *Board* ώστε ο υπολογιστής να αξιολογεί την κάθε δυνατή κατεύθυνση και να προτιμάει αυτή με την υψηλότερη απόλυση αξία.

Η κλάση *MinMax* υλοποιεί τον αλγόριθμο αναζήτησης. Τα αντικείμενα *MinMax* προσφέρουν την μέθοδο *getMove()* που επιστρέφει το στιγμιότυπο παιχνιδιού στο οποίο θα βρεθεί μετά την κίνηση του. Ο *MinMax* χρησιμοποιεί τις

μεθόδους `getChildren()` και `evaluate()` της `Board` προκειμένου να χτίσει το δέντρο και να αξιολογίσει τις πιθανές του εκφάνσεις.

ΠΑΡΑΔΕΙΓΜΑΤΑ

Εδώ μπορούμε να δούμε ορισμένα παραδείγματα από την εκτέλεση του προγράμματός μας. Αρχικά, ο χρήστης καλείται να επιλέξει το μέγιστο βάθος αναζήτησης του αλγόριθμου MiniMax:

```
Welcome to riversi game!!!
Please enter an integer for the desired max depth(range between 3-7):
```

Στη συνέχεια, δίνεται στον χρήστη η επιλογή να παίξει με τα μαύρα ή τα λευκά πιόνια. Κατά σύμβαση, τα μαύρα πιόνια ξεκινούν πρώτα:

```
Welcome to riversi game!!!
Please enter an integer for the desired max depth(range between 3-7):3

The depth defined by the user is: 3
Now the user must choose whether to play first or not.First player starts always with BLACK.
If you wish to play first then press Y (YES), otherwise press N (NO) :
```

Τυπώνεται το ταμπλό στην κονσόλα και το πρόγραμμα αναμένει από τον χρήστη να εισάγει τη θέση που θέλει να τοποθετήσει το πιόνι του:

```
If you wish to play first then press Y (YES), otherwise press N (NO) :Y
You have chosen to play first!!

  1 2 3 4 5 6 7 8
  =====
A| . . . . . . . |
B| . . . . . . . |
C| . . . . . . . |
D| . . . W B . . . |
E| . . . B W . . . |
F| . . . . . . . |
G| . . . . . . . |
H| . . . . . . . |
  =====

In order to play your turn, type the tile you want to play e.g.: f5
>
```

Η επιλογή του χρήστη ακολουθείται από την κίνηση του υπολογιστή:

```
In order to play your turn, type the tile you want to play e.g.: f5
> f5
  1 2 3 4 5 6 7 8
  =====
A| . . . . . . . . |
B| . . . . . . . . |
C| . . . . . . . . |
D| . . . W B . . . |
E| . . . B B . . . |
F| . . . . B . . . |
G| . . . . . . . . |
H| . . . . . . . . |
  =====

Now it is the AI turn to AUTOplay!!!
>
  1 2 3 4 5 6 7 8
  =====
A| . . . . . . . . |
B| . . . . . . . . |
C| . . . . . . . . |
D| . . . W W W . . |
E| . . . B B . . . |
F| . . . . B . . . |
G| . . . . . . . . |
H| . . . . . . . . |
  =====

In order to play your turn, type the tile you want to play e.g.: f5
>
```

Στην περίπτωση που ο χρήστης εισάγει ένα πλακάκι του ταμπλό που οδηγεί σε μη επιτρεπτή κίνηση, λαμβάνουμε αντίστοιχο μήνυμα σφάλματος και καλείται ξανά να τοποθετήσει το πιόνι του:

```

If you wish to play first then press Y (YES), otherwise press N (NO) :Y
You have chosen to play first!!

      1 2 3 4 5 6 7 8
=====
A| . . . . . . . . |
B| . . . . . . . . |
C| . . . . . . . . |
D| . . . W B . . . |
E| . . . B W . . . |
F| . . . . . . . . |
G| . . . . . . . . |
H| . . . . . . . . |
=====

In order to play your turn, type the tile you want to play e.g.: f5
> A1
Movement is illegal!!!
java.lang.IllegalArgumentException
In order to play your turn, type the tile you want to play e.g.: f5
>

```

Όταν κανένας παίκτης δεν μπορεί να τοποθετήσει νέο πιόνι χωρίς να παραβιάσει τους κανονισμούς, το παιχνίδι τερματίζει και ανακοινώνεται ο νικητής:

```

In order to play your turn, type the tile you want to play e.g.: f5
> g3
      1 2 3 4 5 6 7 8
=====
A| W W W W W W W B |
B| W W W W W W W W |
C| W W W W W W W W |
D| W W W W W W W W |
E| W B B W B W B W |
F| W B B B W B W W |
G| W B B W W W . W |
H| W B B B B B B B |
=====

Now it is the AI turn to AUTOplay!!!
>
No more valid moves on the game or end of moves!!!
GAME TERMINATED!!!
WHITE WON

```

