

Τεχνητή Νοημοσύνη

OVERVIEW

Το συγκεκριμένο αρχείο δημιουργήθηκε για να περιγράψει γενικά το τι ακριβώς κάνει ο αλγόριθμος Riversi.py και τον τρόπο με τον οποίο λειτουργεί. Το πρόγραμμα είναι γραμμένο σε Python3

PROCEDURE

Αρχικά το πρόγραμμα με το που τρέξει δημιουργεί τον πίνακα που αποτελείται από απλές γραμμές και τα πούλια μέσω της μεθόδου drawBoard. Αντί για μαύρα και άσπρα πούλια προτιμήσαμε να βάλουμε X και O τα οποία γίνονται initialize στον παραπάνω πίνακα (όπως και όλος ο υπόλοιπος πίνακας) από την μέθοδο resetBoard.

Ο παίκτης επιλέγει το πiónι το οποίο θέλει μέσω της μεθόδου chooseTile. Μετά την επιλογή του παίκτη διαμορφώνεται και η σειρά με την οποία θα παίξουν. Αν ο παίκτης επιλέξει το O τότε θα παίξει πρώτος καθώς το O παίζει πρώτο. Αλλιώς θα παίξει πρώτα το AI.

Στην περίπτωση που είναι η σειρά του παίκτη να παίξει χρησιμοποιούνται 4 μέθοδοι (εμμεσα 5). Η getPlayerMove, η οποία επιστρέφει το input του χρήστη και το ελέγχει μέσω της isValidMove. Η makeMove η οποία χρησιμοποιεί την μέθοδο isValidMove, η οποία σε περίπτωση που δεν επιστρέψει false επιστρέφει έναν πίνακα με όλα τα πiónια που μπορούν να μετατραπούν (από X σε O και από O σε X). Αυτός ο Πίνακας γυρνάει στην makeMove και πραγματοποιεί τις αλλαγές στο board μας. Τέλος η drawBoard η οποία απλα ζωγραφίζει ξανά στο terminal το board με τις καινούριες του αλλαγές. Το currentTile αλλάζει και το AI παίρνει σειρά

Στην περίπτωση που είναι η σειρά του AI χρειαζόμαστε input από τον χρήστη για confirmation (κυρίως για λόγους ευκολίας gameplay). Εκτελείται η μέθοδος bestMove η οποία με την βοήθεια του αλγορίθμου AlphaBeta, βρίσκει την καλύτερη δυνατή κίνηση που μπορεί να κάνει στη συγκεκριμένη περίπτωση κάθε φορά το AI. Αυτό το καταφέρνει φτιάχνοντας έναν εικονικό πίνακα κάθε φορά, τον οποίο τον περνάει από τον AlphaBeta και μετράει τους πόντους σε αυτόν. Στο τέλος η πιο κερδοφόρα για το AI κίνηση είναι αυτή με την οποία θα κάνει και το makeMove

Το πρόγραμμα συνεχίζει με τον ίδιο ρυθμό, εναλλάσσοντας μεταξύ παίκτη και AI. Αυτό το οποίο είναι γρήγορα αντιληπτό είναι το ότι το AI καθυστερεί να κάνει την κίνηση του μετά από ένα σημείο και μετά, προφανώς επειδή έχει να κάνει περισσότερους υπολογισμούς καθώς οι πιθανές κινήσεις γίνονται όλο και περισσότερες όσο εξελίσσεται το παιχνίδι.

Το παιχνίδι τελειώνει όταν ούτε το AI ούτε ο παίκτης θα μπορούν πλέον να κάνουν έγκυρες κινήσεις δηλαδή όταν κανένας από τους δύο δεν μπορούν να κερδίσουν καινούρια αντίπαλα πιόνια. Αυτό φαίνεται από την `isTerminalNode`. Αμέσως μετά βγαίνει το `score` του AI και του παίκτη μέσω της μεθόδου `evalBoard`.