

ΑΝΑΦΟΡΑ 2ης ΕΡΓΑΣΤΗΡΙΑΚΗΣ ΑΣΚΗΣΗΣ ΣΤΟ ΜΑΘΗΜΑ ΤΗΣ ΤΕΧΝΗΤΗΣ ΝΟΗΜΟΣΥΝΗΣ

Κωνσταντίνος Κασφίκης 2013030108
Αλέξανδρος Τερζής 2013030184

Εισαγωγή

Ζητούμενο της άσκησης είναι η ανάπτυξη ενός προγράμματος το οποίο θα είναι ικανό να “παίζει” το επιτραπέζιο TUC-Chess. Το πρόγραμμα αυτό (client) θα πρέπει να μπορεί:

- Να λαμβάνει τα δεδομένα απο τον server σχετικά με την κατάσταση στην οποία βρίσκεται το παιχνίδι (αν δηλαδή το παιχνίδι μόλις ξεκίνησε, βρίσκεται σε εξέλιξη ή τελείωσε), το σκόρ των δύο παικτών, την θέση τυχόν Bonus καθώς και τις κινήσεις που πραγματοποιούνται από τους δύο παίκτες κατά την διάρκεια του παιχνιδιού. (1^ο Μέρος)
- Τα παραπάνω δεδομένα θα πρέπει να επεξεργαστούν από τον client. (2^ο Μέρος)
- Ο client πρέπει να αποφασίζει ποια κίνηση από τις διαθέσιμες θα επιλέξει, με σκοπό να νικήσει στο τέλος του παιχνιδιού. (3^ο Μέρος)
- Η κίνηση αυτή στέλνεται στον server (4^ο Μέρος)

1^ο Μέρος

Ο client επικοινωνεί με τον server μέσω TCP/IP connection.

Η κλάση που θα δημιουργήσει και θα υποστηρίξει την επικοινωνία αυτή είναι η Client. Ο client, σε αυτό το στάδιο, αφού καταφέρει να συνδεθεί με τον server περιμένει να δεχθεί μήνυμα από τον server με πληροφορία για το ποιο θα είναι το χρώμα του client/παίκτη στο παρόν παιχνίδι. Μόλις καθοριστεί το χρώμα ο client περιμένει μήνυμα που σηματοδοτεί την έναρξη του παιχνιδιού.

Με την έναρξη του παιχνιδιού, ο client περιμένει για να ενημερωθεί για τις κινήσεις που πραγματοποιήθηκαν στην σκακιέρα ή για την λήξη του παιχνιδιού.

Τα δεδομένα που λαμβάνονται από τον server περνάνε στην συνέχεια στην κλάση World για επεξεργασία.

2ο Μέρος

Η κλάση World θα δεχθεί τα δεδομένα που λήφθηκαν από τον server και θα αναλύσει το περιεχόμενο τους. Σημαντικό στοιχείο της υλοποίησης αποτελεί ο πίνακας board.

Ο πίνακας αυτός αποτελείται από στοιχεία τύπου Piece. Η κλάση Piece περιέχει την πληροφορία ως προς το τι είδους πεσσοί περιέχει το συγκεκριμένο κελί της σκακιάρας (πιόνι, βασιλιά, πύργο, κενό) αλλά και πληροφορία για το χρώμα του πεσσού.

Όταν το παιχνίδι ξεκινήσει η κλάση World θα αρχικοποιήσει τον πίνακα board σε ένα δυσδιάστατο πίνακα τύπου Piece, ενώ οι πεσσοί τοποθετούνται σε κατάλληλες θέσεις. Ο πίνακας αυτός θα αποτελέσει την ιδεατή αναπαράσταση του παιχνιδιού για τον client.

Μόλις ο client δεχθεί κάποια καινούργια κίνηση που πραγματοποιήθηκε στην σκακιάρα (μαζί με πληροφορία ως προς το ποιος παίκτης παίζει) ενημερώνει τον πίνακα board ενώ στην συνέχεια καλούνται οι κλάσεις MiniMax και AlphaBeta προκειμένου να αποφασιστεί η βέλτιστη διαθέσιμη κίνηση.

Τόσο η MiniMax όσο και AlphaBeta αποτελούν υποκλάσεις της κλάσης MovementAlgo. Η κλάση MovementAlgo περιέχει μεθόδους για την εύρεση τόσο των κινήσεων όσο και των επιθέσεων του κάθε πεσσού στην σκακιάρα.

Η MiniMax και AlphaBeta μέσω των μεθόδων που περιέχονται στην MovementAlgo θα αποθηκεύσουν σε μια λίστα τύπου Move όλες τις πιθανές κινήσεις που υπάρχουν για τον συγκεκριμένο παίκτη/client στην συγκεκριμένη διάταξη των πεσσών, βάσει του πίνακα board.

Η κλάση Move περιέχει πληροφορία για την αρχική θέση ενός πεσσού (x_1, y_1) και για την θέση στην οποία τελικά κατέληξε (x_2, y_2).

3ο Μέρος

Δεδομένης μίας κατάστασης του παιχνιδιού, ο αλγόριθμος αναζήτησης Minimax καλείται να αποφασίσει ποια θα είναι η επόμενη κίνησή του έναντι του αντιπάλου.

Το μέτρο της υπεροχής του ενός ή του άλλου αντιπάλου δίνεται από μία ευριστική συνάρτηση που καλείται συνάρτηση αξιολόγησης (evaluation function) και η οποία εφαρμόζεται στα φύλλα του δένδρου του παιχνιδιού.

Στην πράξη ο αλγόριθμος MiniMax ξεκινάει σε μια κατάσταση όπου υποδεικνύεται από το περιεχόμενο του board (κατάσταση A). Ανάλογα με το χρώμα του παίκτη θα κληθεί η συνάρτηση Min αν είναι μαύρο αλλιώς θα κληθεί η συνάρτηση Max.

Έστω ότι ο παίκτης ελέγχει τους μαύρους πεσσούς. Αρχικά θα κληθεί η Min με παράμετρο το βάθος στο οποίο θα ψάξουμε n , η οποία αποθηκεύει σε μία λίστα όλες τις πιθανές κινήσεις των μαύρων πεσσών, ενώ στην συνέχεια πραγματοποιεί κάθε μια από αυτές σε ένα αντίγραφο του board (το οποίο και θα κρατήσει ως βασικό board (κατάσταση B)). Έχοντας ως board αυτό της κατάστασης B καλείται η Max() η οποία θα αποθηκεύει σε μία λίστα όλες τις πιθανές κινήσεις των άσπρων πεσσών, ενώ στην συνέχεια πραγματοποιεί κάθε μια από αυτές σε ένα αντίγραφο του board (το οποίο και θα κρατήσει ως βασικό board (κατάσταση Γ)). Κάθε φορά που καλείται η Min ή η Max μέσω της Max ή της Min αντίστοιχα, το βάθος μειώνεται κατά ένα. Έτσι η παραπάνω διαδικασία σταματάει όταν το βάθος φτάσει στο 0, όπου και καλείται η συνάρτηση αξιολόγησης. Η τιμή που θα επιστρέψει η συνάρτηση αξιολόγησης περνάει στην κατάσταση A και αποθηκεύεται. Η κίνηση στην κατάσταση A που θα αποφέρει την μεγαλύτερη τιμή συνάρτησης αξιολόγησης για βάθος n θεωρείται και η βέλτιστη κίνηση. Για κάθε κατάσταση, επιλέγεται όχι η κίνηση που μπορεί να αποφέρει το

υψηλότερο κέρδος, αλλά η κίνηση η οποία στην απολύτως χειρότερη περίπτωση θα έχει την μικρότερη ζημία.

Η συνάρτηση αξιολόγησης που χρησιμοποιείται είναι η `estimate()`. Η `estimate()` στην κατάσταση A θα μετρήσει τους διαθέσιμους πόντους που υπάρχουν τόσο για τους λευκούς όσο και για τους μαύρους πεσσούς (οι πόντοι αποτελούνται από τους συνολικούς πόντους των διαθέσιμων λευκών η μαύρων πεσσών – η αξία αυτή ορίζεται : 1 για το πιόνι, 3 για τον πύργο , 7 για τον βασιλιά - , τα Bonus καθώς και η μετακίνηση πιονιών σε τερματική θέση).

Μετά την κλήση της `minimax` για κάθε περίπτωση όπου το βάθος είναι 0 ξανακαλείται η `estimate()` και η τιμή της αφαιρείται από αυτή την αρχικής κατάστασης.

Όσον αφορά τον αλγόριθμο AB-pruning, παρατηρούμε ότι ο αλγόριθμος MiniMax αναλύει υποδέντρα που δεν επηρεάζουν την MiniMax τιμή, και κατ' επέκταση ούτε την απόφαση. Το άλφα-βήτα κλάδεμα είναι μια βελτίωση του αλγορίθμου MiniMax, που του δίνει την δυνατότητα να «κλαδεύει» αυτά τα υποδέντρα.

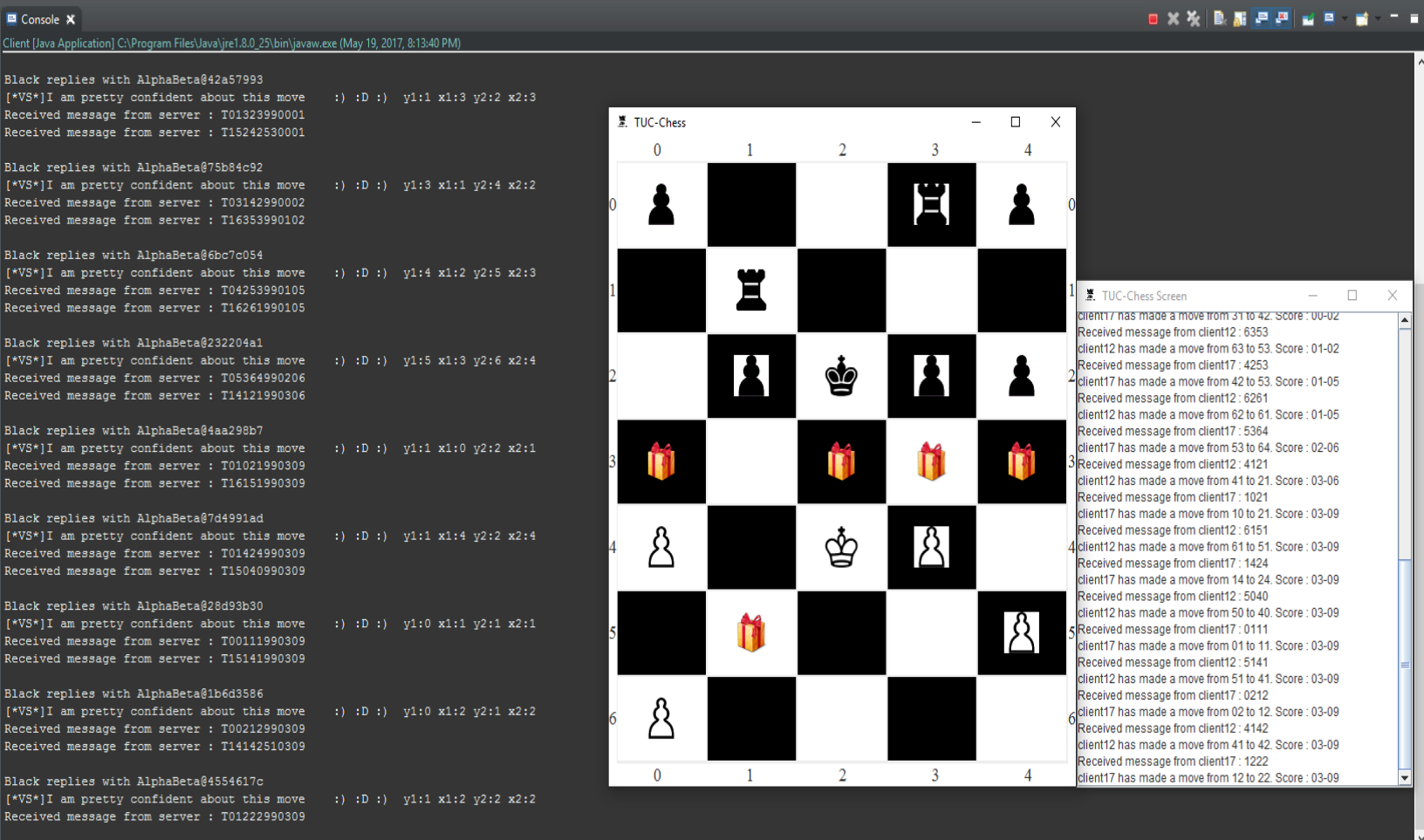
Η κλάση AlphaBeta δουλεύει με αντίστοιχο τρόπο με την `minimax`. Στις περισσότερες περιπτώσεις οι τιμές που προκύπτουν από την `minimax` και την `alphabeta` είναι ίδιες. Για την κλήση των αλγορίθμων καλείται η συνάρτηση `getReply()` που παίρνει ως όρισμα το χρώμα των πεσσών που ελέγχονται από τον παίκτη.

4ο Μέρος

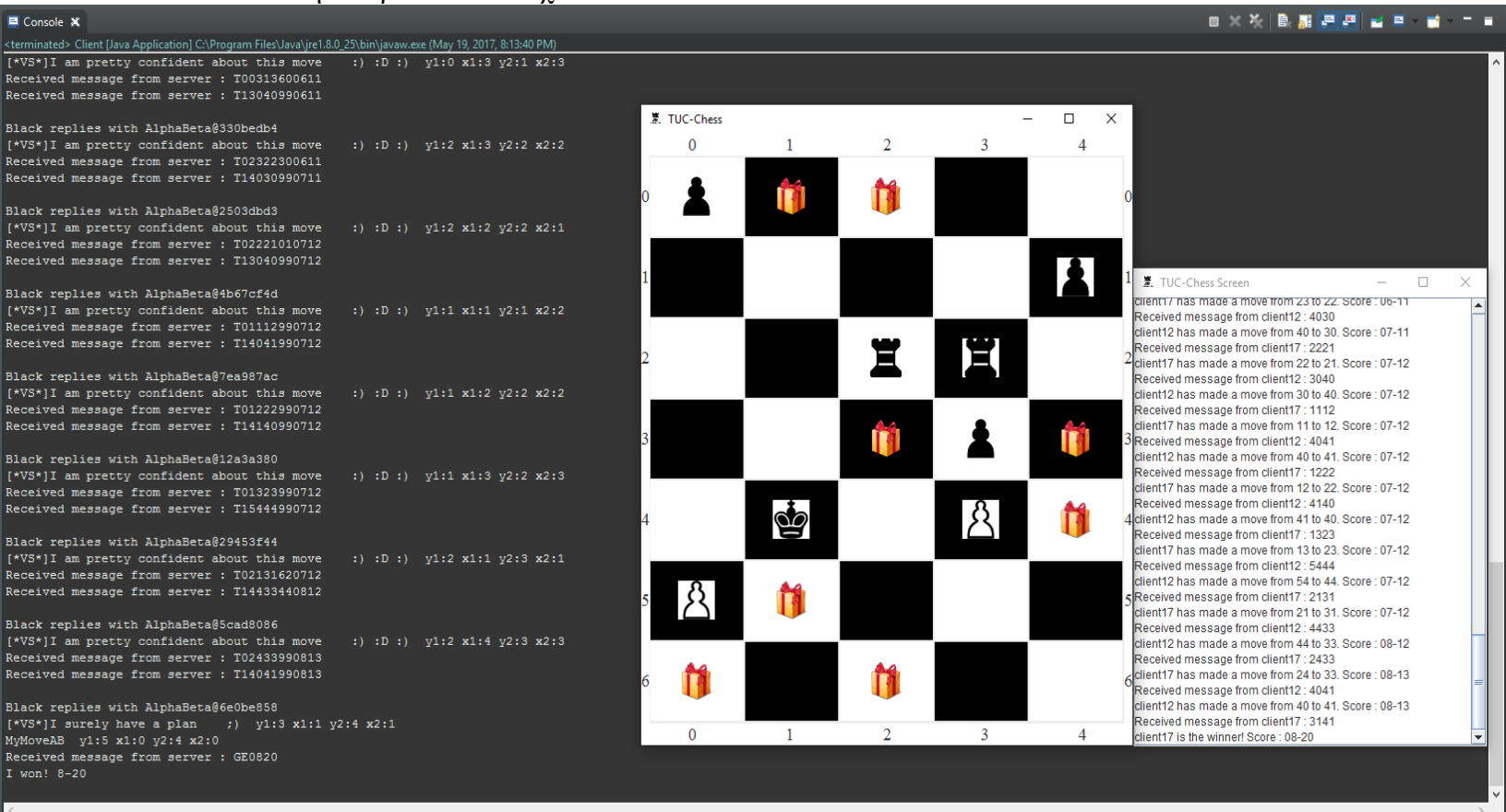
Μόλις η κλάση MiniMax καταλήξει σε μία κίνηση, η κίνηση ελέγχεται ως προς την εγκυρότητα της και στην συνέχεια στέλνεται στον server.

Ιδιαιτερότητες Κώδικα:

- Η επαλήθευση των αποτελεσμάτων με AB-pruning μπορεί να απενεργοποιηθεί με την αλλαγή της μεταβλητής `abPruning` στην κλάση `World` σε `False`.
- Παρότι ο αλγόριθμος του `client` λειτουργεί επαρκώς στην περίπτωση που ο παίκτης ελέγχει τους μαύρους πεσσούς, φαίνεται να λειτουργεί λιγότερο ικανοποιητικά στην περίπτωση που ελέγχει τους λευκούς.



Εικόνα κατά την διάρκεια του παιχνιδιού



Λήξη Παιχνιδιού