

# ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ

## 2η Εργασία Pacman Project2(multiagent)

ΟΝΟΜΑΤΕΠΩΝΥΜΟ: Παπαγεωργίου Βασίλειος-Νικηφόρος  
ΑΜ=1115201400140

### Γενικές Πληροφορίες:

Η εργασία μου υλοποιεί το ζητούμενο στο αρχείο multiagents.py.  
Για το **πρώτο ερώτημα**(Reflex Agent) ,σκέφτηκα οτι δύο επαρκείς αποστάσεις για μια αξιόπιστη ευρετική είναι η ελάχιστη απόσταση του Pacman από φαγητό(την υπολογίζω) καθώς και η ελάχιστη απόσταση από φάντασμα(την υπολογίζω).Επίσης σημαντικό είναι στην ευρετική να προσμετράται το υπάρχον Score(με βάση την έτοιμη συνάρτηση `getScore()`).Στον τελικό υπολογισμό παίρνω τους αντίστροφους των παραπάνω αποστάσεων(σύμφωνα με το hint της εκφώνησης) και τελικά προσθέτω τον αντίστροφο της ελάχιστης απόστασης απ το φαγητό(όσο μικρότερη η απόσταση τόσο μεγαλύτερο το κλάσμα άρα θέλουμε να αυξάνεται η τελική τιμή της ευρετικής για μικρές αποστάσεις από φαγητά) και αφαιρώ τον αντίστροφο της ελάχιστης απόστασης απ το φάντασμα(όσο μικρότερη η απόσταση τόσο μεγαλύτερο το κλάσμα άρα θέλουμε να μειώνεται η τελική τιμή της ευρετικής για μικρές αποστάσεις από φαντάσματα).

Για το **δεύτερο , το τρίτο και το τέταρτο ερώτημα** ακολουθώ παρόμοια λογική-δομή κάνοντας τις μικρές απαιτούμενες αλλαγές για τις παραλλαγές του αλγορίθμου.Γενικά και στους τρεις αλγορίθμους ακολουθώ το πρότυπο των διαφάνειων ,με την διαφοροποίηση οτι δεν παίζει εναλλάξ ο min με τον max,αλλά οτι παίζει ένας max και μετά πολλοί min.Ουσιαστικά,αυξάνω το depth ,μόνο αφού παίζουν όλοι οι min άρα κάθε στρώμα αποτελείται απο έναν max και κάποιους min.Στη συνάρτηση για τον max,δεν χρειάζεται να κάνω κάτι ιδιαίτερο καθώς πάντα ακολουθεί min,οπότε πάντα απ τον max καλώ τον min.Ωστόσο στη συνάρτηση για τον min ,μπορεί να χρειάζεται να καλέσω είτε τον max είτε τον min.Για τον λόγο αυτό έχω δημιουργήσει τη συνάρτηση `decision`,η οποία παίρνει ως ορίσματα το τρέχον βάθος,και τον τρέχοντα αριθμό agent.Ωστόσο,χρησιμοποιώντας τη συνάρτηση `getNumAgents()`,μπορούμε

να βρούμε το συνολικό πλήθος agents, και συνεπώς όταν ο index φτάσει τον συνολικό αριθμό από agents, αυξάνουμε το βάθος κατά ένα και μηδενίζουμε φυσικά τον index αφού ουσιαστικά ξεκινάμε πάλι απ την αρχή για το επόμενο επίπεδο. Κατά συνέπεια όταν ο index είναι 0 θα πρέπει να παίζει ο max και σε κάθε άλλη περίπτωση ο min. Στον expectimax, η διαφοροποίηση είναι ότι δεν έχουμε min, αλλά έχουμε τον πιθανοτικό expectimax, ο οποίος παίζει ισοπίθανα για κάθε πιθανό action με πιθανότητα  $1.0/\text{πλήθος πιθανών actions}$ .

Στο **5ο ερώτημα** χρησιμοποιώ ακριβώς την ευρετική του πρώτου, με την προσθήκη μιας ακόμα απόστασης, της ελάχιστης απόστασης απο capsule, τον αντίστροφο της οποίας προσμετράω θετικά στην ευρετική τιμή που επιστρέφεται (αφού όσο μικρότερη η απόσταση τόσο μεγαλύτερο το κλάσμα άρα θέλουμε να αυξάνεται η τελική τιμή της ευρετικής για μικρές αποστάσεις από κάψουλες αφού συμφέρει τον Pacman να “εξαφανίσει” και να μπορεί να φάει τα φαντάσματα όσο είναι φοβισμένα λόγω της κάψουλας).

**Βαθμολόγηση Autograder: 25/25**