

Project 1 για το μάθημα της Τεχνητής νοημοσύνης Ι

Ονοματεπώνυμο: Απόστολος Καρβέλας

AM: 1115201800312

READ_ME:

Ο κώδικας δημιουργήθηκε και εκτελέστηκε σε python 3.6.6 σε περιβάλλον Windows 10

Πρόβλημα 1: Project 1 search

Q1 (search.py DFS):

Υλοποίηση του ψευδοκώδικα από τις διαφάνειες με την διαφορά ότι βλέπει όταν ο κόμβος είναι ο τελικός έξω από την for loop λόγω του autograder.

Q2 (search.py BFS):

Ίδια με την Q1 αλλά αντί για Stack η αναζήτηση κατά πλάτος χρησιμοποιεί Queue.

Q3 (search.py UCS):

Υλοποίηση του ψευδοκώδικα από τις διαφάνειες.

Q4 (search.py A*):

Ίδια με την Q3 αλλά στην εύρεση κόστους του επόμενου κόμβου μαζί με την απόσταση προσθέτουμε και την τιμή από την ευρετική συνάρτηση.

Q5 (searchAgents.py cornersproblem):

Για την επίσκεψη κάθε γωνίας χρησιμοποιεί μια λίστα με 4 δεδομένα Boolean, ένα για κάθε κόμβο. Κατά την δημιουργία της κλάσης αρχικοποιεί την λίστα αυτή και στην συνάρτηση getStartState την επιστρέφει με την αρχική θέση. Επίσης, σαν συνάρτηση στόχου (getGoalState) κοιτάει αν έχουν επισκεφτεί κάθε γωνία και για την επιστροφή των successors βλέπει αν η επόμενη επιτρεπόμενη κίνηση είναι τοίχος και αν βρίσκεται στις γωνίες που δεν έχουμε επισκεφτεί τότε αλλάζει την Boolean τιμή.

Q6 (searchAgents.py cornersHeuristic):

Για την ευρετική συνάρτηση βρίσκει την απόσταση manhattan (που ξέρουμε ότι είναι ευρετική) από κάθε γωνία που δεν έχει επισκεφτεί και επιστρέφει την υψηλότερη τιμή.

Q7 (searchAgents.py foodHeuristic):

Χρησιμοποιεί την συνάρτηση `mazeDistance` το οποίο βρίσκει την απόσταση μεταξύ 2 σημείων (όπως `manhattan`).

Q8 (searchAgents.py ClosestDotSearchAgent):

Χρησιμοποιεί `greedy` αναζήτηση μέσω της UCS από την Q3 έχοντας ως `goalstate` να μην υπάρχει στοιχείο στην λίστα με τα φαγητά.