email at: sdi2000055@di.uoa.gr

Περιεχομενα

1. Ασκηση 1

Για την ερώτηση 1, στην δημιουργία των sentences (1,2,3) ακολούθησα απλά τις εντολές που μας δίνονται, χωρίς να κάνω κάτι άλλο ιδιαίτερο. Προσπάθησα να ακολουθήσω την οδηγία για χρήση της conjoin, disjoin όσο περισσότερο γίνεται για να έχουμε αποδοτικότητα.
findModelCheck

Απλά ακολούθησα την οδηγία, φτιάχνοντας ένα dictionary βασισμένο στην μορφή που επιστρέφει η findModel

entails

Όπως γνωρίζουμε από την θεωρία της προτασιακής λογικής, για να αποδείξουμε ότι από μια KB έπειται λογικά μία πρόταση ϕ, δηλαδή $KB \vDash \phi$, τότε αρκεί να φτάσουμε σε contradiction χρησιμοποιώντας την άρνηση της πρότασης μας. Άρα χρησιμοποιώντας την συνάρτηση findModel, και ως είσοδο την $KB \wedge \neg\phi$, αν φτάσουμε σε αποτέλεσμα $False$, δηλαδή contradiction, θα πρέπει να επιστρέψουμε $True$, άρα για αυτό η άρνηση στο αποτέλεσμα.

plTrueInverse
Χρησιμοποιώντας πάλι την συνάρτηση pl_true , απλά δίνουμε ως όρισμα την άρνηση της inverse_statement και έχουμε το επιθυμητό αποτέλεσμα.

2. Ασκηση 2

atLeastOne
Αφού ζητάμε CNF, και θέλουμε ένα να είναι τουλάχιστον αληθές, η λύση είναι μπροστά μας, καθώς με ένα απλό disjoin έχουμε και την μορφή έτοιμη και το επιθυμητό αποτέλεσμα.

atMostOne

Σε γενική μορφή το at most one μπορεί να αναπαρασταθεί ώς: $\neg((x_i \wedge x_j) \vee (x_j \wedge x_k) \vee (x_i \wedge x_k) \vee \cdots)$ αλλά δεν είναι σε CNF. Φέρνοντας την άρνηση εντός καταλήγουμε στο αποτέλεσμα που θέλουμε δηλαδή μια μορφή συζεύξεων από όρους της μορφής $(\neg x_i \vee \neg x_j)$. Για να πάρουμε όλα τα ζεύγη των x_i, x_j χρησιμοποιούμε την συνάρτηση itertools.combinations ανά 2.

3. Ασκηση 3

pacmanSuccessorSingle

Όπως αναφέρεται και στην εκφώνηση αρκεί να κάνουμε fill in το return statement. Όπως αναφέρεται και σε σχόλια του κώδικα η λογική που πρέπει να ακολουθήσουμε είναι ότι $Current \iff$ (Ποσιτιον ατ τ-1) $\wedge$ (Αχτιον το μοε)
Τον όρο (Ποσιτιον ατ τ-1) $\wedge$ (Αχτιον το μοε) μας το δίνει ήδη ο κώδικας στα if statements.
Ο όρος $Current$ είναι απλά η θέση του agent pacman την στιγμή που εξετάζουμε, δηλαδή $t = now$.
Από τα δύο παραπάνω βγαίνει το return statement.

pacphysicsAxioms

Για την συνάρτηση αυτή ακολούθησα πιστά τα βήματα που μας δίνεται στην εκφώνηση. Αρχικά λοιπόν (1) ορίζω την λογική για τους τοίχους και το ότι εκεί που υπάρχουν αυτοί αποκλείεται να υπάρχει pacman. Έπειτα, (2) ορίζω οτι από τις πιθανές θέσεις που μπορούν να υπάρξουν εκτός των τοίχων, στο πολύ μια εξ αυτών βρίσκεται ο pacman, ενώ (3) από τις 4 πιθανές κινήσεις, κάνει ακριβώς μια από αυτές. Στην συνέχεια προσθέτουμε (4) τα αξιώματα από τους σένσορες, σιγουρεύοντας ότι δεν είναι None, και τα αξιώματα που δίνονται από το successorAxioms αν ωστόσο δεν έχουμε $t = 0$, γιατί πριν από αυτή δεν έχουμε κάποια κατάσταση.

checkLocationSatisfiability

Ομοίως με την παραπάνω συνάρτηση, ακολούθησα πιστά τα βήματα που αναφέρονται στην εκφώνηση. Αρχικά ξεκίνησα προσθέτοντας στην βάση γνώσης τα αξιώματα των timesteps, άρα για $t = 0, 1$, με τα κατάλληλα successorAxioms που δίνονται από την allLegalSuccessorAxioms (1). Έπειτα προθέτω στην βάση γνώσης (2) την αρχική θέση του πακμαν, και (3) τις 2 κινήσεις (action0, action1). Τελικώς ορίζω το query το οποίο αναφέρεται στην θέση του πακμαν την χρονική στιγμή 1, και έπειτα χρησιμοποιώ την findModel για να βρω τα μοντέλα με βάσει την βάσης γνώσης και το ερώτημα για τα 2 που ζητούνται.

4. Ασκηση 4

positionLogicPlan

Για να μην επαναλαμβάνω και εδώ την λογική, είναι ίδια με των δύο παραπάνω ασκήσεων , δηλαδή ακολούθησα σε μεγάλο βαθμό αυτό που μας δίνεται από την εκφώνηση. Πρόσθεσα ωστόσο το ότι αν σε μια από αυτές τις θέσεις υπάρχει τοίχος τότε δεν

μπορούμε να βρισκόμαστε σε εκείνο το σημείο, κάτι το οποίο δεν αναφέρεται ρητά αλλά χωρίς αυτόν τον περιορισμό δεν μπορούμε να συνεχίσουμε. Στο τέλος ελέγχουμε με το κατάλληλο ερώτημα αν έχουμε βρεί ένα μοντέλο για την λύση και αν ναι τότε βγάζουμε την ακολουθία κινήσεων, αλλιώς συνεχίζουμε.

5. Ασκηση 5

Όπως αναφέρεται και στην εκφώνηση, η 5 βασίζεται (¨πατάει πάνω¨) στην 4, άρα μερικά σημεία θα είναι ίδια. Συγκεκριμένα, στην αρχή θα προσθέσουμε την αρχική θέση του πράκτορα στην βάση γνώσης, και στην συνέχεια θα συνεχίσουμε να έχουμε μόνο ένα βήμα την φορά, ενώ τα αξιώματα για επίσης παραμένουν τα ίδια. Οι αλλαγές έχουν να κάνουν με το φαγητό. Αφού φτάνουμε τον στόχο μας όταν όλες οι κουκίδες φαγητού είναι $False$, δηλαδή όταν ισχύει $\bigwedge^n(\neg Food_i) \equiv \neg(\bigvee^n Food_i)$, με μια απλή σύζευξη με την βάση γνώσης θα έχουμε την συνθήκη που πρέπει να ισχύει για να είμαστε σε κατάσταση στόχου. Ελέγχοντας το αποτέλεσμα από την findModel, αν φτάσαμε σε λύση καλούμε την extractActionSequence και επιστρέφουμε. Αλλιώς προχωράμε στο επόμενο βήμα, το οποίο τώρα απαιτεί την επόμενη πρόσθεση, η οποία αφορά την ενημέρωση των κουκίδων φαγητού. Η σκέψη είναι απλή: Αν ο πράκτορας βρίσκεται σε θέση που πριν υπήρχε φαγητό, πλέον δεν υπάρχει φαγητό, αφού το έφαγε. Άρα η λογική είναι ότι για την στιγμή $t+1$, η κουκίδα φαγητού παραμένει φαγωμένη αν ήταν ήδη φαγωμένη στο παρελθόν (alreadyEaten), ή αν την φάγαμε τώρα (nowEaten = $\neg$ alreadyEaten $\wedge$ $P[x][y]_t$).

6. Ασκηση 6

Ακολουθώντας πιστά την εκφώνηση, αρχικά πρόσθεσα στην βάση γνώσης μας την πληροφορία για τους τοίχους. Ξεκινώντας από όλες τις συντεταγμένες (x, y) μπορούμε με δεδομένες τις λίστες που μας δίνονται (walls_list, all_coords) να ελέγξουμε το δεδομένο που θα προσθέσουμε. Συγκεκριμένα, αν $(x, y) \in$ walls_list τότε ξέρουμε ότι εκεί υπάρχει τοίχος άρα θα πρέπει να προσθέσουμε το δεδομένο ότι $WALL[x][y]_t$ αλλιώς δεν υπάρχει τοίχος άρα προσθέτουμε το $\neg WALL[x][y]_t$.
Έπειτα για κάθε χρονικό βήμα, αρχικά προσθέτω τα φυσικά αξιώματα, μετά την κίνηση που κάναμε, και έπειτα τα αποτελέσματα από τον σένσορα fourBitPerceptRules. Πάλι, δεν υπάρχουν πολλές εναλλακτικές, αφού τα βήματα δίνονται strictly από την εκφώνηση.
Συνεχίζω προσπαθώντας να βρω τις επόμενες θέσεις του πράκτορα pacman. Η λίστα non_outer_wall_coords έχει όλες τις πιθανές θέσεις εκτός των εξωτερικών τοίχων. Περνώντας από τις συντεταγμένες, κάνω 2 ερωτήματα στην βάση γνώσης, σχετικά με το αν μπορούμε να έπουμε λογικά την θέση του pacman με την γνώση μας (Υλοποιώντας δηλαδή το ότι αν $Q =$ ύπαρξη του πράκτορα στο (x, y) την στιγμή t, θέλω να ελέγξω αν $KB \vDash Q$ και $KB \vDash \neg Q$). Για αυτό χρησιμοποιώ την συνάρτηση entails όπως αναφέρεται και στην εκφώνηση. Τι μπορούμε να πούμε τώρα για μια πιθανή θέση του pacman· Ας προσέξουμε τα παρακάτω για να εξάγουμε και το αντίστοιχο αποτέλεσμα:

• Αν $KB \vDash Q = False$, τότε δεν μπορούμε να εξάγουμε συμπέρασμα ότι ο πράκτορας βρίσκεται στην θέση, άρα μπορεί να μην βρίσκεται εκεί. Ομοίως,
• Αν $KB \vDash \neg Q = False$, τότε δεν μπορούμε να εξάγουμε συμπέρασμα ότι ο πράκτορας δεν βρίσκεται στην θέση, άρα μπορεί να βρίσκεται εκεί.
Με αυτήν την λογική επομένως για να εξάγουμε το συμπέρασμα ότι **ο πράκτορας δεν βρίσκεται στην θέση εκείνη**, πρέπει να ξέρουμε ότι μπορούμε να βγάλουμε το συμπέρασμα ότι δεν βρίσκεται εκεί ($KB \vDash \neg Q = True$) και ότι δεν μπορούμε να πούμε με σιγουριά ότι βρίσκεται εκεί ($\neg(KB \vDash Q = False)$). Ομοιώς, για να εξάγουμε το συμπέρασμα ότι **ο πράκτορας βρίσκεται στην θέση εκείνη** πρέπει να ξέρουμε ότι μπορούμε να βγάλουμε το συμπέρασμα ότι βρίσκεται εκεί ($KB \vDash Q = True$) και ότι δεν μπορούμε να πούμε με σιγουριά ότι δεν βρίσκεται εκεί ($\neg(KB \vDash \neg Q = False)$). Αν δεν ισχύει τίποτα από τα παραπάνω, έχουμε πιθανή θέση, αλλά δεν είμαστε σίγουροι, άρα απλά προσθέτουμε στην πιθανή θέση αλλά όχι στην βάση γνώσης.

7. Ασκηση 7

Η λύση είναι σχεδόν ολόιδια με της 6, μόνο που μας ζητάται πλέον να εισάγουμε την αρχικές θέση του πράκτορα, καθώς και το ότι αφού υπάρχει εκεί πράκτορας, σίγουρα δεν υπάρχει τοίχος εκεί (wall_at_start_loc = $P[x][y]_t \rightarrow \neg WALL[x][y]$). Όσο αφορά το υπόλοιπο κομμάτι, ακολουθώντας πάλι τον ψευδοκώδικα που μας δίνεται και τα hints, με παρόμοια λογική με την άσκηση 6, δημιουργούμε τον πίνακα που ζητείται.

8. Ασκηση 8

Η ερώτηση 8 είναι εύκολη αν έχουμε υλοποίησει τις 6,7. Η λογική είναι ίδια με των δύο παραπάνω και συνδυαστικά παράγουν τον αλγόριθμο που χρειαζόμαστε. Το μόνο που αλλάζει είναι ο σένσοράς μας και τα αξιώματα.