

# Pacman: Project 3

## Logic and Classical Planning

Κωνσταντίνος Χούσος

### Περίληψη

Το συγκεκριμένο report αναλύει τις λύσεις μου στο project 2 των Pac-Man Projects του Berkeley [1], στα πλαίσια του προπτυχιακού μαθήματος “Τεχνητή Νοημοσύνη”. Τα ερωτήματα 1, 2 και 3 αποτελούν μια απλή “μετάφραση” των λογικών εκφράσεων σε εκφράσεις τύπου **Expr**, όπου οι πρώτες εκφράζονται στο 1 με λογική σημειογραφία, στο 2 με γραπτό λόγο και στο 3 με μορφή αλγορίθμου. Τα ερωτήματα 4 με 8 αποτελούν μια πιστή και “τυφλή” υλοποίηση των δοσμένων αλγορίθμων.

### 1: LOGIC WARM-UP

Τα **sentence1** και **sentence2** είναι μία απλή μεταφορά των λογικών προτάσεων της εκφώνησης σε εκφράσεις της κλάσης **Expr**.

Το **sentence3** είναι παρόμοιο με τα προηγούμενα δύο, όπου όμως σε αυτήν την περίπτωση χρησιμοποιούμε την συνάρτηση **PropSymbolExpr()** για την αρχικοποίηση των λεκτικών.

Η **findModelCheck()** επιστρέφει **{dummyClass("a"): True}**, δηλαδή ό,τι θα επέστρεφε **findModel(Expr('a'))** αν το όρισμα 'a' ήταν αποδεκτό. Χρησιμοποιεί την εσωτερική στην συνάρτηση κλάση **DummyClass**.

Η **entails()** χρησιμοποιεί το θεώρημα της μη ικανοποιησιμότητας του συγγράμματος των Russell και Norvig [2]:

$\alpha \models \beta \Leftrightarrow \eta \ (\alpha \wedge \neg \beta)$  είναι μη ικανοποιήσιμη.

Η **plTrueInverse()** αποτελείται από μόνο μια γραμμή κώδικα, όπου χρησιμοποιείται η ήδη υπάρχουσα συνάρτηση **pl\_true()**.

### 2: LOGIC WORKOUT

Η συνάρτηση **atLeastOne()** εκμεταλλεύεται την συνάρτηση **disjoin()** και για τα λεκτικά **A, B, C** επιστρέφει την λογική έκφραση

$$A \vee B \vee C.$$

Η συνάρτηση **atMostOne()** επιστρέφει την λογική έκφραση

$$(\neg A \vee \neg B) \wedge (\neg A \vee \neg C) \wedge (\neg B \vee \neg C).$$

Η συνάρτηση **exactlyOne()** είναι ο συνδυασμός των παραπάνω, δηλαδή επιστρέφει

$$\text{atLeastOne}() \wedge \text{atMostOne}().$$

### 3: PACPHYSICS AND SATISFIABILITY

Όσον αφορά την **pacmanSuccessorAxiomSingle()**, η λογική έκφραση που επιστρέφει περιγράφει το εξής: Ο Pacman είναι στο σημείο ανν (αν και μόνο αν) τουλάχιστον ένας από τους πιθανούς λόγους για να είναι σε εκείνο το σημείο αληθεύει.

Οι συναρτήσεις **pacphysicsAxioms()** και **checkLocationSatisfiability()** ακολουθούν κι υλοποιούν πιστά τον εκάστοτε αλγόριθμο της εκφώνησης [1].

### ΕΡΩΤΗΜΑΤΑ 4, 5, 6, 7 & 8

Τα ερωτήματα 4 (Path Planning with Logic), 5 (Eating All the Food), 6 (Localization), 7 (Mapping) και 8 (SLAM) ακολουθούν πιστά τους εκάστοτε αλγορίθμους που παραθέτει η εκφώνηση [1].

Οι αλγόριθμοι των βοηθητικών “συναρτήσεων” *Add pacphysics, action, and percept information to KB, Find possible pacman locations with updated KB* και *Find provable*

*wall locations with updated KB* υλοποιού- ορισμάτων που θέλουν ανά περίπτωση θα νται σε κάθε ερώτημα. Δεν αποτελούν ξε- δυσκόλεψε και θα περίπλεκε την—τελικά χωριστές συναρτήσεις καθώς είναι αμελη- περιττή—υλοποίηση. τέου μεγέθους και η διαφοροποίηση των

### Αναφορές

- [1] John DeNero, Dan Klein και Pieter Abbeel. *Projects - CS 188: Introduction to Artificial Intelligence, Spring 2022*. 2022. URL: <https://inst.eecs.berkeley.edu/~cs188/sp22/projects/>.
- [2] Stuart J. Russell και Peter Norvig. *Artificial Intelligence: A Modern Approach*. Συνεργασία από Ming-wei Chang κ.ά. Fourth edition, global edition. Pearson Series in Artificial Intelligence. Harlow: Pearson, 2022. ISBN: 978-1-292-40113-3.