



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ**

**Σχεδιασμός, υλοποίηση GPS εφαρμογής σε
περιβάλλον visual prolog**

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**ΗΑΧΗΑJ THEODHORA
(ΑΕΜ: 1131)**

**Επιβλέπων : Δόσης Μιχαήλ
Καθηγητής**

Καστοριά - Νοέμβριος 2023

Ευχαριστίες

Θα ήθελα να εκφράσω τις ευχαριστίες μου στον κ.Δόση Μιχαήλ και στον κ.Μπάτο Παναγιώτη για την καθοδήγηση και την στήριξη κατά τη διάρκεια της εκπόνησης της παρούσας εργασίας. Τέλος, ένα μεγάλο ευχαριστώ σε όλους τους ανθρώπους που στάθηκαν δίπλα μου κατά τη διάρκεια των σπουδών μου, στους καθηγητές μου και στους οικείους μου.

Περίληψη

Όλοι γνωρίζουμε το γεγονός της διαρκώς εξελισσόμενης τεχνολογίας γι' αυτό και η εποχή που διανύουμε θεωρείται εδώ και μερικές δεκαετίες ως η μεταβιομηχανική εποχή και συγκεκριμένα της πληροφορίας. Αυτό φέρει ως αποτέλεσμα την μεταμόρφωση της πληροφορίας ως έννοια, σε κύριο προϊόν. Οι κυβερνήσεις, η επιστήμη και πλέον και οι απλοί πολίτες λαμβάνουν αποφάσεις πριν δράσουν για κάποιο έργο αφού πρώτα επεξεργαστούν και εκτιμήσουν διάφορες πληροφορίες και κυρίως γεωγραφικά χαρακτηριστικά όπως αυτό της ακριβής θέσης ενός σημείου. Η αντιστοίχιση χώρου και πληροφοριών οδηγεί στην έννοια της γεωγραφικής πληροφορίας. Η εργασία αυτή έχει ως στόχο την δημιουργία μιας GPS εφαρμογής σε γλώσσα Prolog στην οποία ο χρήστης θα διατυπώνει ερωτήματα σχετικά με την εύρεση διαδρομής ανάμεσα σε δύο σημεία και η εφαρμογή θα αναζητά και θα εμφανίζει τη πιθανή διαδρομή εφόσον υπάρχει.

Λέξεις κλειδιά: prolog, GPS, διαδρομή, μονοπάτια

Abstract

We are all aware of the fact of the constantly evolving technology, which is why the era we are going through has been considered for a few decades as the post-industrial era, specifically of information. This results in the transformation of information as a concept into a primary product. Governments, science and now even ordinary citizens make decisions before acting on a project after first processing and evaluating various information and mainly geographical features such as the exact location of a point. The mapping of space and information leads to the concept of geographic information. This diploma thesis aims to create a GPS application in Prolog language in which the user will formulate questions about finding a route between two points and the application will search for the possible route and show it if it exists.

Key words: prolog, GPS, paths

Πίνακας περιεχομένων

Εισαγωγή.....	6
Κεφάλαιο 1ο: Οι εφαρμογές GPS και τα εργαλεία δημιουργίας τους.....	7
Περίληψη.....	7
1.1. Υπάρχον λογισμικό - Μελέτες περιπτώσεων.....	8
1.2. Εργαλεία.....	14
1.3. Παράδειγμα προγράμματος σε Prolog.....	20
1.4. SWI-Prolog.....	21
1.5. Εκτέλεση παραδείγματος social media στη SWI-Prolog.....	23
1.6. Visual Prolog.....	28
Κεφάλαιο 2ο: Σχεδίαση εφαρμογής GPS “WayToHome”-Βάση γνώσεων-UML.....	30
2.1. Βάση γνώσης (Knowledge base).....	30
2.2. UML σχέδιο εφαρμογής GPS “WayToHome”.....	31
2.3. Ανάλυση βάσης γνώσης.....	32
Κεφάλαιο 3ο: Παρουσίαση κώδικα εφαρμογής GPS “WayToHome”.....	34
3.1. Κώδικας εφαρμογής GPS “WayToHome”.....	34
3.2. Ανάλυση κώδικα εφαρμογής GPS “WayToHome”.....	35
3.3. Εκτέλεση του κώδικα εφαρμογής GPS “WayToHome” με παραδείγματα.....	36
3.3.1. Βήματα εκτέλεσης της εφαρμογής.....	36
3.3.2. Παραδείγματα εκτέλεσης της εφαρμογής.....	37
Κεφάλαιο 4ο: Εκτέλεση του κώδικα εφαρμογής GPS “WayToHome” στο περιβάλλον Visual Prolog.....	43
4.1. Βήματα εκτέλεσης του κώδικα εφαρμογής GPS “WayToHome”.....	43
4.2.Κώδικας εφαρμογής “WayToHome” στο περιβάλλον Visual Prolog.....	46
Συμπεράσματα.....	49
Βιβλιογραφία.....	50

Εισαγωγή

Η παρούσα πτυχιακή εργασία πραγματεύεται στην διαδικασία εύρεσης διαδρομής. Αρχικά γίνεται μελέτη σε ήδη υπάρχοντα λογισμικά με σκοπό την εξοικείωση του αναγνώστη με την εργασία. Στην συνέχεια αναφέρονται μερικά παραδείγματα από εμπορικές εφαρμογές με τα χαρακτηριστικά και τις λειτουργίες τους. Ακόμη, παρουσιάζεται η εφαρμογή μας πρώτα σε επίπεδο διαγραμμάτων UML και σε δεύτερο επίπεδο η ανάλυση του κώδικα. Η εφαρμογή έχει υλοποιηθεί σε περιβάλλον SWI-Prolog και λειτουργεί με ερωτήματα διαδρομής με σκοπό την εύρεση είτε μοναδικής είτε πολλών πιθανών διαδρομών προς το σημείο ενδιαφέροντος.

Στο κεφάλαιο 1 παρουσιάζεται τι είναι το GPS, κάποια παραδείγματα από ήδη υπάρχουσες εφαρμογές όπως και τα εργαλεία δημιουργίας τους.

Στο κεφάλαιο 2 παρουσιάζεται ο σχεδιασμός της GPS εφαρμογής μέσω των διαγραμμάτων UML, οι συσχετίσεις τους όπως και η βάση γνώσης που χρησιμοποιήθηκε στην Prolog.

Στο κεφάλαιο 3 πραγματοποιείται η παρουσίαση και η ανάλυση του κώδικα της εφαρμογής. Συγκεκριμένα παρουσιάζεται η υλοποίηση της εφαρμογής βήμα προς βήμα με επεξήγηση της.

Τέλος, εκτελούνται παραδείγματα της εφαρμογής.

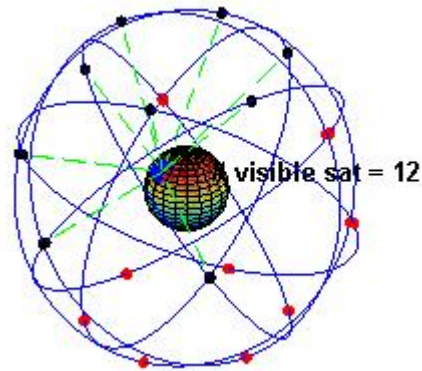
Κεφάλαιο 1ο: Οι εφαρμογές GPS και τα εργαλεία δημιουργίας τους

Περίληψη

Το Παγκόσμιο Σύστημα Στιγματοθέτησης (ή Θεσιθεσίας) - GPS (Global Positioning System) είναι ένα παγκόσμιο σύστημα εντοπισμού γεωγραφικής θέσης το οποίο ξεκίνησε από το Υπουργείο Άμυνας των ΗΠΑ και ονομάστηκε NAVSTAR GPS (Navigation Signal Timing and Raging Global Positioning System). Στη πορεία δόθηκε για εμπορική χρήση, περίπου στα μέσα της δεκαετίας του 1990. Στην ουσία είναι ένα σύστημα πλοήγησης που παρέχει πληροφορίες σχετικά με τη τοποθεσία, το υψόμετρο, τη κατεύθυνση και την ταχύτητα ενός ακίνητου ή κινούμενου χρήστη. Για να το επιτύχει αυτό βασίζεται σε ένα δίκτυο 24-32 δορυφόρων της Γης, οι οποίοι είναι εφοδιασμένοι με ειδικές συσκευές εντοπισμού, τους “πομποδέκτες GPS”. Το GPS αποτελείται από τρία λειτουργικά μέρη:

- i. Το διαστημικό τμήμα
- ii. Το επίγειο τμήμα ελέγχου
- iii. Το τμήμα τελικού χρήστη

Το τμήμα τελικού χρήστη, το οποίο είναι και αυτό που μας ενδιαφέρει περισσότερο στη προκειμένη περίπτωση, αποτελείται από τους τελικούς χρήστες, δηλαδή τους πολίτες και τον στρατό. Οι δέκτες GPS που χρησιμοποιούν οι πολίτες εμφανίζονται σε κατανοητή μορφή δια μέσου χάρτη στην οθόνη της συσκευής GPS αφού συνδυαστούν με ειδικό λογισμικό [6].



Εικόνα 1: Αναπαράσταση του αρχικού σχεδίου του συστήματος GPS, με 24 δορυφόρους GPS



Εικόνα 2: Χάρτης του επίγειου τμήματος ελέγχου GPS

Πλέον υπάρχουν τόσες πολλές εφαρμογές πλοήγησης GPS διαθέσιμες για εγκατάσταση στα smartphones που για μερικούς είναι δύσκολο να αποφασίσουν ποιες εξυπηρετούν καλύτερα τις ανάγκες τους διότι θα πρέπει να καταλάβουν αποθηκευτικό χώρο και ίσως και χρήμα. Για τον λόγο αυτό θα πρέπει κάποιος να “ζυγίσει” τα θετικά και τα αρνητικά κάθε εφαρμογής, αφού πρώτα φυσικά καταλήξει στο ποια είναι η κατάλληλη για τις εργασίες που χρειάζονται [14].

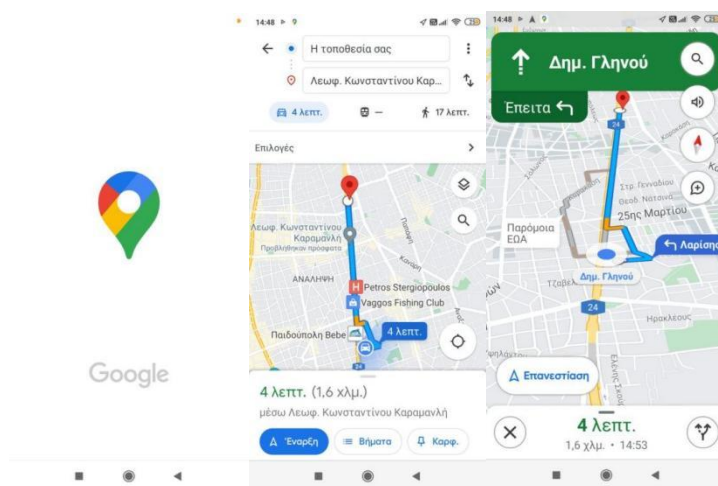
1.1. Υπάρχον λογισμικό - Μελέτες περιπτώσεων

Χάρτες Google - Google Maps

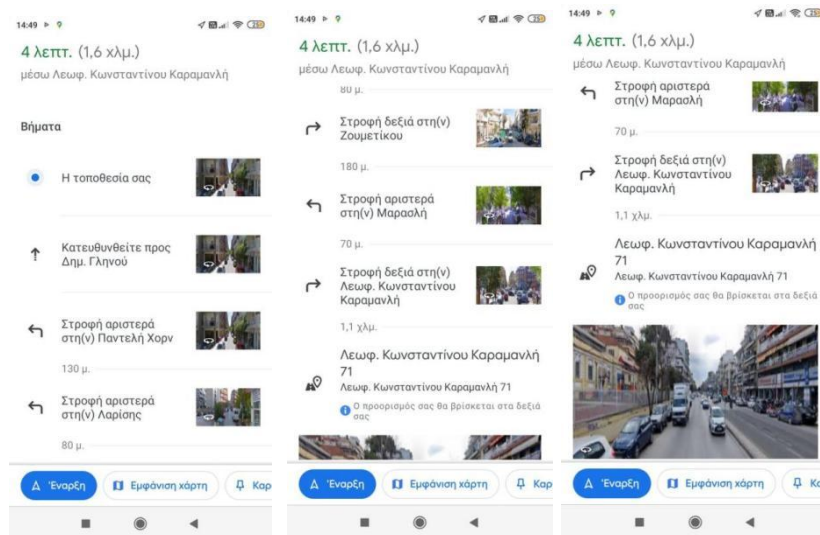
Η παλαιότερη εφαρμογή που θεωρείται ο “παππούς” των GPS εφαρμογών είναι το Google Maps. Αρχικά ξεκίνησε ως πρόγραμμα σταθερού υπολογιστή με γλώσσα προγραμματισμού C++. Εξαγοράστηκε από την Google τον Οκτώβριο του

2004 η οποία το μετέτρεψε σε εφαρμογή και μετά από πρόσθετες εξαγορές μιας εταιρίας οπτικοποίησης γεωγραφικών δεδομένων (Keyhole) και ενός αναλυτή κίνησης σε πραγματικό χρόνο (ZipDash), οι Χάρτες Google κυκλοφόρησαν τον Φεβρουάριο του 2005.

Η διεπαφή της υπηρεσίας χρησιμοποιεί JavaScript, XML, Json για λόγους καλύτερης απόδοσης και Ajax. Οι χάρτες προσφέρουν μια διεπαφή (API) που επιτρέπει την ενσωμάτωση χαρτών σε ιστοτόπους τρίτων και προσφέρει έναν εντοπιστή για επιχειρήσεις και άλλους οργανισμούς σε πολλές χώρες σε όλο τον κόσμο. Η εφαρμογή για android διατέθηκε πρώτη φορά στο κοινό τον Σεπτέμβριο του 2008. Ανοίγοντας την εφαρμογή στο κινητό τηλέφωνο, στο κάτω μέρος της οθόνης υπάρχει η επιλογή “μετάβαση”. Επιλέγοντας την μπορεί ο χρήστης να δει τη διαδρομή για το εκάστοτε σημείο ενδιαφέροντος από οποιαδήποτε τοποθεσία. Δίδονται οι επιλογές εμφάνισης διαδρομής μέσω οδήγησης οχήματος, συγκοινωνίας, περπατήματος, ποδηλασίας και πτήσεων. Ο χρήστης μπορεί να επιλέξει το κουμπί “Έναρξη” μόλις αρχίσει να κινείται για να βλέπει ανά πάσα στιγμή σε ποιο σημείο βρίσκεται και μια φωνή θα τον καθοδηγεί στις στροφές. Μπορεί όμως και να επιλέξει το κουμπί “Βήματα” το οποίο θα εμφανίσει ένα προς ένα τα βήματα της διαδρομής, τις στροφές και τα ονόματα των οδών [9].



Εικόνα 3: Άνοιγμα εφαρμογής Google Maps και επιλογή σημείου μετάβασης



Εικόνα 4: Εμφάνιση διαδρομής στην εφαρμογή Google Maps

Τα θετικά της εφαρμογής είναι:

- Λειτουργεί για πολλές μορφές μετακίνησης.
- Ενημέρωση για θέματα κυκλοφοριακής συμφόρησης.
- Δίνει ώρες αναχώρησης - άφιξης για την οδήγηση.
- Δίνει εκτιμώμενη ώρα άφιξης βάσει των συνθηκών κυκλοφορίας.
- Προσφέρει αυτόματη αλλαγή διαδρομής λόγω κυκλοφορίας, ατυχημάτων ή άλλων οδικών συνθηκών.
- Διαθέσιμο εκτός σύνδεσης.

Τα αρνητικά της εφαρμογής είναι:

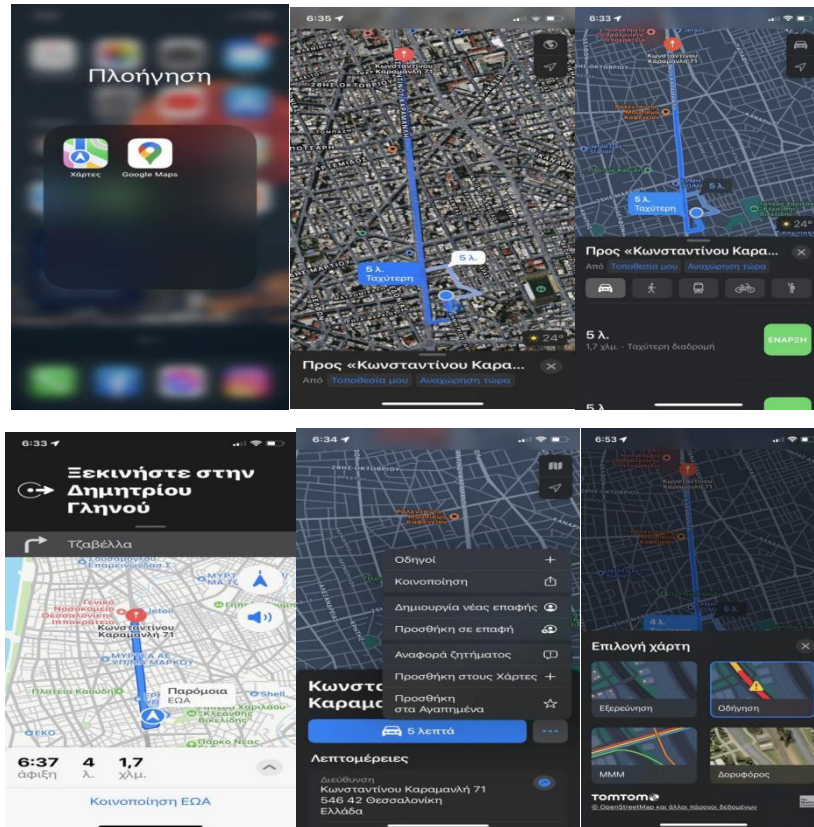
- Ορισμένες κριτικές παραπονιούνται για οδηγίες της επόμενης στροφής που λείπουν ή είναι λανθασμένες, καθώς οι άνθρωποι χάνουν τις εξόδους τους.
- Μπορεί να εξαντλήσει τη μπαταρία του τηλεφώνου.
- Εάν ο χρήστης κάνει λάθος στροφή, συχνά θα τον ανακατευθύνει πίσω στην αρχική διαδρομή ακόμη και αν μια νέα διαδρομή θα ήταν συντομότερη.
- Οι κριτικές χρηστών από το 2021 αναφέρουν προβλήματα με τις ενημερώσεις εφαρμογών, όπως η κατάργηση λειτουργιών, οι επερχόμενες στροφές και η αυτόματη αλλαγή της διαδρομής λόγω ατυχημάτων.

Χάρτες Apple - Apple Maps

Οι χάρτες Apple είναι υπηρεσία χαρτογράφησης που αναπτύχθηκε από την Apple Inc. Κυκλοφόρησε για πρώτη φορά τον Σεπτέμβριο του 2012 και αντικατέστησε την

υπηρεσία Google Maps ως προεπιλογή σε όλες τις συσκευές Apple (iOS, iPadOS, macOS, watchOS). Χρησιμοποιεί κυρίως τη γλώσσα προγραμματισμού Swift, η οποία δημιουργήθηκε από την ίδια την Apple και την κοινότητα ανοιχτού κώδικα (open source community). Χρησιμοποιεί και τεχνολογίες όπως JavaScript, .NET και διάφορα API για ανανέωση των δεδομένων (κίνηση, καιρός κλπ.), όπως το πιο βασικό με την TomTom. Οι Χάρτες Apple μπορούν να χρησιμοποιηθούν για τον σχεδιασμό διαδρομών. Η υπηρεσία πλοήγησης διαθέτει πλοήγηση στροφή προς στροφή με προφορικές οδηγίες και εκτιμώμενες ώρες άφιξης σε πραγματικό χρόνο για οχήματα, πεζούς και μέσα μαζικής μεταφοράς, εμφανίζονται ο καιρός και η ποιότητα του αέρα του προορισμού. Η λειτουργία “Flyover” δείχνει ορισμένα πυκνοκατοικημένα αστικά κέντρα και άλλα σημεία ενδιαφέροντος σε ένα τρισδιάστατο τοπίο που αποτελείται από μοντέλα κτιρίων και κατασκευών. Επιπλέον, διατίθεται ενσωματωμένη η εικονική βοηθός της Apple, “Siri”, και κάποιες επιπλέον λειτουργίες όπως η αποθήκευση της διαδρομής διεύθυνσης σε τηλεφωνική επαφή. Σύμφωνα με την Apple η λειτουργία πλοήγησης είναι διαθέσιμη σε 56 χώρες παγκοσμίως. Ο χάρτης εμφανίζει σημεία ενδιαφέροντος που παρέχονται από περίπου 20 εταιρίες, συμπεριλαμβανομένων των Booking.com, TripAdvisor, Yelp και Foursquare. Οι χρήστες έχουν εντοπίσει παρ’ όλα αυτά κάποια αρνητικά σημεία της εφαρμογής:

- Ορισμένοι χρήστες λένε πως οι οδηγίες εξόδου και οι πληροφορίες σχετικά με τις επιχειρήσεις είναι ανακριβείς.
- Ενδέχεται να λείπουν σημεία ενδιαφέροντος στους χάρτες σε σύγκριση με άλλες εφαρμογές.
- Δεν υπάρχει επιλογή για χρήση χαρτών εκτός σύνδεσης.
- Η λειτουργία “Look Around street view” δεν έχει εικόνες για παντού στις ΗΠΑ και είναι διαθέσιμη μόνο σε 5 χώρες [12].



Εικόνα 5: Εκτέλεση εφαρμογής Apple Maps

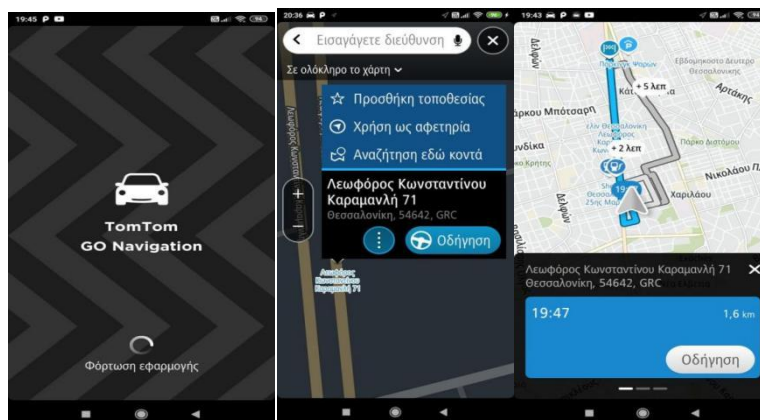
TomTom Go Navigation

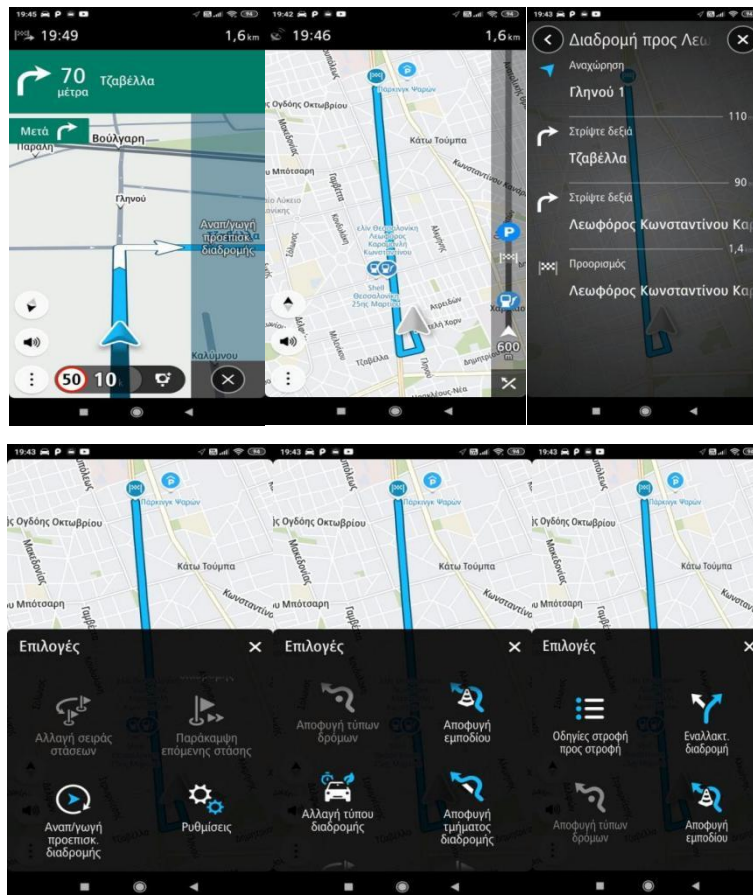
Το TomTom Go Navigation είναι μια εφαρμογή που έχει δημιουργηθεί από την ολλανδική εταιρία TomTom N.V με βάση το Άμστερνταμ. Οι μονάδες TomTom παρέχουν μια ιπτάμενη διασύνδεση με λοξή πανοραμική άποψη του δρόμου, καθώς και απευθείας προβολή χάρτη. Χρησιμοποιούν έναν δέκτη GPS για να δείχνουν την ακριβή τοποθεσία και να παρέχουν οπτικές και προφορικές οδηγίες για το πώς να οδηγήσει ο χρήστης προς τον καθορισμένο προορισμό. Ορισμένα συστήματα TomTom ενσωματώνονται επίσης με κινητά τηλέφωνα που χρησιμοποιούν Bluetooth, χάρτες κυκλοφοριακής συμφόρησης ή για να δέχονται κλήσεις και να διαβάζουν δυνατά τα μηνύματα sms. Η TomTom χρησιμοποιεί λογισμικό ανοιχτού κώδικα (open source) στα προϊόντα της και δεσμεύεται να υποστηρίζει τη κοινότητα ανοιχτών προγραμματιστών. Μπορεί κανείς να επισκεφθεί την αποθήκη Github της TomTom στο <http://github.com/tomtom-international> για να μπορέσει να δει τον

πηγαίο κώδικα και τις προϋποθέσεις του. Κάποιες από τις γλώσσες που μπορεί να παρατηρήσει ο επισκέπτης ότι χρησιμοποιούνται είναι οι Python, Java, JavaScript, HTML, Swift.

Λογισμικά TomTom με βάση το λειτουργικό σύστημα

Υπάρχει το TomTom for iOS που είναι μόνο για συσκευές iOS και λειτουργεί με το iPhone (όλα τα μοντέλα), το iPodTouch (όλα τα μοντέλα) και το iPad (όλα τα μοντέλα). Ωστόσο, η Apple απέσυρε την υποστήριξη για τα πρώιμα μοντέλα και οι πιο πρόσφατες εκδόσεις της TomTom for iOS ενδέχεται να έχουν προβλήματα σε ορισμένες συσκευές. Τον Μάιο του 2015 ανακοινώθηκε το TomTom Go Mobile που είναι για AndroidOS, με ένα μοντέλο συνδρομής premium για χάρτες με τα πρώτα 50 μίλια/75 χλμ τον μήνα να είναι δωρεάν, συμπεριλαμβανομένων όλων των διαθέσιμων χαρτών TomTom Traffic και Speed Cameras. Παρακάτω παρουσιάζονται βήμα-βήμα η εγκατάσταση και η χρήση της εφαρμογής TomTom Go Navigation σε AndroidOS. Η εφαρμογή όπως παρατηρείται απευθύνεται κυρίως σε οδηγούς. Στην αναζήτηση γράφει ο χρήστης τη διεύθυνση προορισμού του, στο side menu εμφανίζεται ένα μίνι μενού με επιλογές, ειδικά επιλέγει το κουμπί “οδήγηση” και εμφανίζονται προτεινόμενες διαδρομές ξεκινώντας πάντα από τη συντομότερη. Μόλις φορτώσει το κουμπί “οδήγηση” ο χάρτης μεγενθύνεται και η φωνή καθοδήγησης αρχίζει να εκφωνεί τις οδηγίες. Στη δεξιά πλευρά της οθόνης υπάρχουν κάποια εικονίδια όπως το εικονίδιο με το γράμμα “P” το οποίο εμφανίζει τα κοντινότερα διαθέσιμα σημεία στάθμευσης (parking), όπως και το εικονίδιο με το σχήμα μιας αντλίας το οποίο εμφανίζει τα διαθέσιμα πρατήρια καυσίμων στη περιοχή. Στις επιλογές ο χρήστης μπορεί να βρει τις οδηγίες της διαδρομής γραμμένες στροφή προς στροφή, εναλλακτικές διαδρομές κλπ.





Εικόνα 6: Εκτέλεση εφαρμογής Tom Tom Go Navigation

Τα θετικά της εφαρμογής είναι:

- Διαθέτει απ'ευθείας πληροφόρηση σχετικά με τη κυκλοφορία, φωνητική πλοήγηση, λειτουργίες εκτός σύνδεσης & τρισδιάστατους χάρτες.
- Προειδοποιεί για κάμερες ταχύτητας όπου επιτρέπεται.
- Περιλαμβάνει χάρτες για 150 χώρες σε περίπτωση που θέλει ο χρήστης να κάνει διεθνές οδικό ταξίδι.
- Επιτρέπει τη διαμοίραση της εκτιμώμενης ώρας άφιξης του χρήστη με την οικογένεια και τους φίλους του.
- Η λειτουργία "Drive to a photo" επιτρέπει να ανεβάσει ο χρήστης μια φωτογραφία στην εφαρμογή και να λάβει οδηγίες για αυτήν.

Τα αρνητικά της εφαρμογής είναι:

- Μόνο τα 50 μίλια πλοήγησης είναι δωρεάν κάθε μήνα. Μετά από αυτό χρειάζεται μηνιαία ή ετήσια συνδρομή.
- Η εφαρμογή δεν ενημερώνεται συχνά.

- Δωρεάν μεν αλλά μόνο για δοκιμαστική περίοδο. Στη συνέχεια πρέπει ο χρήστης να πληρώσει συνδρομή.
- Οι χρήστες έχουν αναφέρει πως η λειτουργία αναζήτησης χρειάζεται βελτίωση. Κάποιες φορές φορτώνει αργά και μερικές άλλες δίνει μονές διαδρομές.

1.2. Εργασία

Prolog

Η ιστορία της γλώσσας προγραμματισμού Prolog, ως μία ειδική περίπτωση υλοποίησης ενός συστήματος Λογικού Προγραμματισμού, ξεκινάει το 1971-1972 με τη δημιουργία του πρώτου διερμηνέα (interpreter) στο πανεπιστήμιο της Μασσαλίας, στη Γαλλία από τους Alain Colmerauer και Philippe Roussel (Colmerauer et.al, 1973; Battani and Meloni, 1973). Ο αρχικός στόχος της ερευνητικής ομάδας ήταν να αναπτύξει ένα σύστημα επεξεργασίας φυσικής γλώσσας (της Γαλλικής), στη συνέχεια όμως το σύστημα αυτό εξελίχθηκε σε γλώσσα προγραμματισμού (Colmerauer and Roussel, 1993). Το όνομα Prolog αναφέρεται ότι προτάθηκε από τη Jacqueline Roussel, σύζυγο του Philippe Roussel ως συγκοπτόμενη λέξη της φράσης “PROgrammation en LOGique”. Η ομάδα της Μασσαλίας συνεργάστηκε στενά με τον Kowalski και με τον τρόπο αυτό συνδυάστηκε η προσπάθεια της ανάπτυξης του συστήματος επεξεργασίας φυσικής γλώσσας των Γάλλων με την εξειδικευμένη εφαρμογή του κανόνα της ανάλυσης. Έχει τις ρίζες της στη κατηγορηματική λογική πρώτης τάξεως και σχετίζεται κυρίως με την τεχνητή νοημοσύνη και την υπολογιστική γλωσσολογία. Χρησιμοποιείται σε σχεσιακές βάσεις δεδομένων, μαθηματική λογική, αυτοματοποίηση σχεδιασμού καθώς και για το αρχικό πεδίο χρήσης της, την επεξεργασία φυσικής γλώσσας. Είναι εντελώς διαφορετική από τις συμβατικές γλώσσες όπως η C, C++, Java. Είναι μια περιγραφική γλώσσα, περιγράφει γεγονότα και σχέσεις και πολύ λιγότερο τα βήματα που εκτελούνται από τον υπολογιστή για να λύσει ένα πρόβλημα. Αυτό σημαίνει πως η Prolog σας λέει τι να κάνετε και όχι πώς να το κάνετε. Στην C++ και στην Java στην ουσία λέμε στον μεταγλωττιστή πώς να λύσει ένα πρόβλημα, πώς να φτάσει σε μια λύση. Στη Prolog δεν θα πούμε στον μεταγλωττιστή πώς να φτάσει στη λύση αλλά τι να κάνει για να φτάσει εκεί. Όλη αυτή η λογική εκφράζεται με γεγονότα, κανόνες και όρους σχέσεων. Χρησιμοποιώντας αυτά τα συστατικά ο χρήστης μπορεί να θέσει στο πρόγραμμα ερωτήματα του ώστε να λάβει απάντηση σχετικά με αυτά. Η Prolog είναι κατάλληλη

για συγκεκριμένες εργασίες που επωφελούνται από λογικά ερωτήματα που βασίζονται σε κανόνες, όπως αναζήτηση βάσεων δεδομένων, συστήματα φωνητικού ελέγχου και συμπλήρωση προτύπων. Τα σύγχρονα περιβάλλοντα Prolog υποστηρίζουν τη δημιουργία γραφικών διεπαφών χρήστη καθώς και διοικητικών και δικτυωμένων εφαρμογών [2].

Επιπλέον πεδία εφαρμογών στα οποία η γλώσσα Prolog έχει χρησιμοποιηθεί με επιτυχία και αξίζει να σημειωθούν είναι τα ακόλουθα:

- **Η Επεξεργασία φυσικής Γλώσσας:** Οι δυνατότητες ταιριάσματος μορφότυπων και η δηλωτική φύση των ορισμών γραμματικής, κάνουν την Prolog εύχρηστο και ισχυρό εργαλείο για την επεξεργασία φυσικής γλώσσας. Άλλωστε όπως αναφέρθηκε ένα σύστημα επεξεργασίας φυσικής γλώσσας οδήγησε στη “γέννηση” της Prolog.
- **Τα Έμπειρα Συστήματα/Συστήματα βασισμένα στη Γνώση:** Η υλοποίηση των συστημάτων αυτών βασίζεται κατά κύριο λόγο σε μία βάση γνώσης (knowledge base) και σε μία μηχανή επαγωγής συμπερασμάτων (inference engine). Τις περισσότερες φορές, η βάση γνώσης εκφράζεται με τη μορφή κανόνων παραγωγής, η υλοποίηση των οποίων γίνεται άμεσα με τη χρήση των κανόνων της Prolog. Επίσης ο μηχανισμός επαγωγής συμπερασμάτων της Prolog που βασίζεται στην αρχή της ανάλυσης είναι αντίστοιχος με την τεχνική της ανάστροφης συλλογιστικής (backward reasoning) που χρησιμοποιείται, σε πολλές περιπτώσεις, για την υλοποίηση της μηχανής επαγωγής συμπερασμάτων. Άλλες μορφές συλλογιστικής είναι επίσης εύκολο να υλοποιηθούν με τη βοήθεια των μεταγλωσσικών δυνατοτήτων της Prolog. Μία ειδική κατηγορία συστημάτων γνώσης είναι τα Συστήματα Λήψης Αποφάσεων (Decision Support Systems) για τα οποία η Prolog επίσης αποτελεί κατάλληλο εργαλείο ανάπτυξης τους, καθώς οι κανόνες απόφασης στους οποίους βασίζονται μπορούν να υλοποιηθούν κατάλληλα.
- **Κατάστρωση Πλάνων και Χρονοπρογραμματισμός (Planning and Scheduling):** Μία άλλη περιοχή της Τεχνητής Νοημοσύνης στην οποία η Prolog αποτελεί κατάλληλο εργαλείο, είναι αυτή της κατάστρωσης πλάνων και του χρονοπρογραμματισμού. Ο μηχανισμός ενοποίησης με τις δυνατότητες ταιριάσματος περιπτώσεων, η δυνατότητα εφαρμογής τεχνικών παραγωγής και δοκιμής (generate and test) και η χρήση των κανόνων της γλώσσας αποτελούν στοιχεία κατάλληλα για αυτού του είδους τις εφαρμογές. Σημαντικό ρόλο επίσης στην υλοποίηση αυτών των εφαρμογών παίζουν και οι μηχανισμοί επαύξησης της Prolog για τη δυνατότητα έκφρασης και επίλυσης προβλημάτων με ικανοποίηση περιορισμών (constraint satisfaction).

- Βάσεις Δεδομένων: Η Prolog έχει επίσης συνεισφέρει στην ευφυή αναζήτηση πληροφορίας σε μεγάλες βάσεις δεδομένων, λόγω της φύσης της κατά την οποία ένα οποιοδήποτε πρόγραμμά της ενεργοποιείται με τη βοήθεια στόχων/ερωτήσεων. Η γλώσσα Datalog, που αποτελεί ένα υποσύνολο της Prolog με απόλυτη δηλωτική σημασιολογία, έχει χρησιμοποιηθεί ως γλώσσα ερωτήσεων σε επαγωγικές βάσεις δεδομένων (deductive databases).

- Σημασιολογικός Ιστός (Semantic Web): Την τελευταία δεκαετία οι προσπάθειες συνδυασμού των τεχνικών του λογικού προγραμματισμού με τις τεχνολογίες του σημασιολογικού ιστού, έχουν δημιουργήσει μία καινούργια μορφή προγραμματισμού με την επωνυμία Περιγραφικός Λογικός Προγραμματισμός (Description Logic Programming). Η ανάπτυξη του βασίστηκε στην διαπίστωση ότι η χρήση λογικών κανόνων που αναφέρονται σε κάποιο συγκεκριμένο πεδίο μπορεί να βελτιώσει τον σημασιολογικό ιστό. Ο περιγραφικός λογικός προγραμματισμός παρέχει ένα βαθμό εκφραστικότητας σημαντικά μεγαλύτερο από αυτόν της περιγραφικής λογικής του RDF-σχήματος.

Υπάρχουν μόνο τρία βασικά στοιχεία στη Prolog: Γεγονότα, κανόνες και ερωτήματα. Μια συλλογή γεγονότων και κανόνων ονομάζεται “βάση γνώσης” (knowledge base). Τα γεγονότα δεν είναι τίποτα άλλο από οικουμενικές αλήθειες. Αναπαριστούν μια ιδιότητα ενός αντικειμένου, μια σχέση μεταξύ αντικειμένων ή ένα αντικείμενο [1].

- Άτομα: Σταθερές που αποτελούνται από αλφαριθμητικούς χαρακτήρες με τις εξής μορφές:
 - ‘Vincent’
 - burger
 - dora_chatzai28
- Αριθμοί:
 - 2
 - 100
 - -1
- Μεταβλητές: Είναι μια συμβολοσειρά από κεφαλαία γράμματα, πεζά, ψηφία και τον χαρακτήρα “_” ο οποίος είναι ιδιαίτερα σημαντικός και ονομάζεται “ανώνυμη μεταβλητή”. Το όρισμα ενός κατηγορήματος ή σύνθετου όρου μπορεί εκτός από άτομο ή σύνθετος όρος να είναι και μεταβλητή. Μία μεταβλητή μπορεί να ταυτιστεί με οποιοδήποτε άτομο, κατηγορημα, σύνθετο όρο ή αριθμό. Η

χρήση τους είναι εξαιρετικής σημασίας καθώς εντοπίζεται κυρίως στους κανόνες (rules) των προγραμμάτων και στα ερωήματα (queries) που θέτει ο κάθε χρήστης.

- X
- Y
- List25
- _head
- _input

➤ Ατομικοί Τύποι ή Σύνθετοι όροι: Οι σύνθετοι όροι δημιουργούνται από έναν συντελεστή που ακολουθείται από μια ακολουθία ορισμάτων. Τα ορίσματα τοποθετούνται σε συνηθισμένες παρενθέσεις, χωρίζονται με κόμματα και τοποθετούνται μετά τον συντελεστή (functor). Ο συντελεστής πρέπει να είναι άτομο.

- loves(vincent,catherine).
- mother(maria,antonis).
- mother(mother(mother(antonis))).

➤ Κανόνες: Οι κανόνες (rules) αποτελούν μία μέθοδο αναπαράστασης διαδικαστικής γνώσης που αντιστοιχεί σε έγκυρους συλλογισμούς και έχει τη μορφή IF-THEN. Η δήλωση ή το σύνολο των δηλώσεων μετά τον όρο IF αντιπροσωπεύει κάποιο παρατηρήσιμο πρότυπο. Η δήλωση ή το σύνολο των δηλώσεων μετά τον όρο THEN αντιπροσωπεύει κάποιο εξαγωγίμο συμπέρασμα ή κάποια εκτελεστή ενέργεια . Επομένως, ένα κανόνας:

- Προσδιορίζει ένα πρότυπο και εξάγει συμπεράσματα σχετικά με το τι αυτό σημαίνει
- Προσδιορίζει ένα πρότυπο και συμβουλεύει τι πρέπει να γίνει γι' αυτό
- Προσδιορίζει ένα μοτίβο και πραγματοποιεί τις κατάλληλες ενέργειες. Οι δηλώσεις μετά το IF ονομάζονται συνθήκες, ισχυρισμοί, προϋποθέσεις, πρόγονοι, ενώ αυτά που ακολουθούν μετά το THEN μπορούν να ονομάζονται συμπεράσματα.
 - ◆ IF A (ισχυρισμοί)
 - ◆ and B (πρόγονοι,προηγούμενα)
 - ◆ and ... (προϋποθέσεις)
 - ◆ THEN X (συμπεράσματα)
 - ◆ and Y (συνεπαγόμενα)
 - ◆ and ... (ενέργειες)

Ισότητα και Ενοποίηση: Δύο όροι ενοποιούνται εάν είναι ο ίδιος όρος ή εάν περιέχουν μεταβλητές που μπορούν να δημιουργηθούν ομοιόμορφα με όρους με τέτοιο τρόπο ώστε οι όροι που προκύπτουν να είναι ίσοι.

?- X=Y.

Σε άλλες γλώσσες προγραμματισμού όπως C, C++, αυτό σημαίνει πως X είναι ίσο με το Y. Στη Prolog σημαίνει πως μπορούν να ενοποιηθούν. Δηλαδή, εάν το X και το Y είναι της ίδιας μορφής μπορούν να ενοποιηθούν. Πιο απλά, ό,τι υπάρχει στην αριστερή πλευρά είναι ίσο με ό,τι υπάρχει στην δεξιά πλευρά μόνο εάν είναι στην ίδια μορφή ή εάν είναι μη εγκατεστημένες μεταβλητές και η αριστερή και η δεξιά πλευρά. (uninstatiated variables).

i. X=Y.

Y=10.

Τότε X=10.

Το X και το Y επιτρέπεται να είναι μη εγκατεστημένες μεταβλητές (uninstatiated variables). Δηλαδή, εάν το ένα μέρος είναι μεταβλητή τότε γίνεται και το άλλο.

ii. Οι ακέραιοι και τα άτομα είναι πάντα ίσα με τους εαυτούς τους.

police=police. -->true

police=pen. -->>false

155=155. -->true

155=2. -->>false

iii. Δύο σύνθετοι όροι είναι ίσοι όταν:

- a) Έχουν τους ίδιους συντελεστές
- b) Έχουν το ίδιο πλήθος στοιχείων
- c) Τα αντίστοιχα στοιχεία είναι ίσα

Παράδειγμα 1:

?- a(b,c(d,e))=a(X,c(Y,e)).

X=b

Y=d

Εάν όμως ήταν: a(b,c(d,e))=a(X,u(Y,e)). τότε θα εμφάνιζε false διότι το c δεν είναι ίσο με το u (βλ. κανόνα ii) [5].

Παράδειγμα 2:

?- letter(c)=word(letter). -->false . Παραβιάζει τους δύο πρώτους κανόνες.

Αποκοπή: Το αυτόματο backtracking (οπισθοδρόμηση) είναι ένα από τα πιο δυνατά χαρακτηριστικά της Prolog. Η οπισθοδρόμηση όμως μπορεί να οδηγήσει σε αναποτελεσματικότητα. Μερικές φορές η Prolog μπορεί να σπαταλήσει χρόνο εξερευνώντας δυνατότητες που δεν οδηγούν πουθενά. Μέχρι στιγμής έχουμε δει μόνο δύο τρόπους για να το κάνουμε αυτό: αλλαγή σειράς κανόνων και αλλαγή σειράς στόχων. Υπάρχει όμως και άλλος τρόπος. Υπάρχει ένα ενσωματωμένο κατηγορημα ! (το θαυμαστικό), που ονομάζεται cut, το οποίο προσφέρει έναν πιο άμεσο τρόπο άσκησης ελέγχου στον τρόπο με τον οποίο η Prolog αναζητά λύσεις. Η λειτουργία του είναι να μειώνει το δένδρο αναζήτησης κλαδεύοντας δυναμικά, δηλαδή κατά την ώρα της εκτέλεσης, κλαδιά (μονοπάτια αναζήτησης) τα οποία γνωρίζουμε ότι δεν οδηγούν σε λύση. Στην ουσία, το cut δε σε αφήνει να γυρίσεις προς τα πίσω χωρίς λόγο και να ψάχνεις τα υπόλοιπα μονοπάτια. Αν περάσεις το cut ή τα cuts τότε θα πρέπει να δώσεις ως απάντηση αυτό που βρήκες. Όταν ο μηχανισμός αναζήτησης της Prolog συναντήσει μία αποκοπή αγνοούνται όλες οι προτάσεις που ανήκουν στη διαδικασία και έπονται της πρότασης που περιέχει την αποκοπή και αγνοούνται όλες οι εναλλακτικές λύσεις των ατομικών τύπων που βρίσκονται πριν από την αποκοπή μέσα στο σώμα του κανόνα που εμφανίζεται. Το όφελος είναι μεγάλο καθώς ο υπολογιστής μας επεξεργάζεται όλο αυτό που του δίνουμε, ενώ με τα cuts τον βοηθάμε να μη χρειάζεται να τα επεξεργαστεί όλα, να γλιτώσει άδικη επεξεργασία στοιχείων, άδικο χρόνο και θα σώσουμε και αρκετή μνήμη RAM [13].

1.3. Παράδειγμα προγράμματος σε Prolog

Ένα πρόγραμμα στη γλώσσα Prolog όπως προαναφέρθηκε χρησιμοποιεί τα δύο βασικά στοιχεία, τα γεγονότα και τους κανόνες. Ακολουθεί ένα παράδειγμα από το πλέον αναπόσπαστο κομμάτι της καθημερινότητας των ανθρώπων, τα μέσα κοινωνικής δικτύωσης (social media), ένα κοινωνικό δίκτυο με χρήστες που συνδέονται με σχέσεις φιλίας.

Knowledge Base:

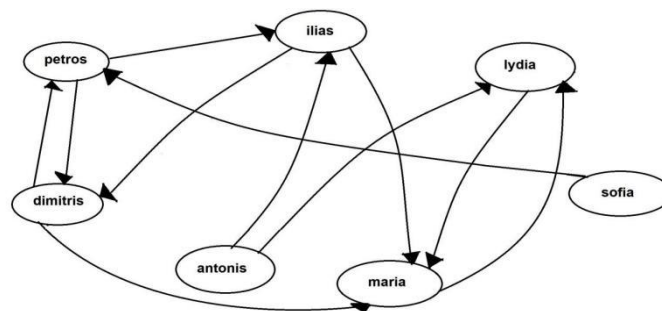
male(petros).

male(ilias).

male(dimitris).

male(antonis).

female(maria).
 female(lydia).
 female(sofia).
 follows(petros,ilias).
 follows(petros,dimitris).
 follows(ilias,dimitris).
 follows(antonis,ilias).
 follows(dimitris,petros).
 follows(dimitris,maria).
 follows(ilias,maria).
 follows(sofia,petros).
 follows(maria,lydia).
 follows(lydia,maria).
 follows(lydia,sofia).
 follows(antonis,lydia).



Εικόνα 7: Σχέδιο UML Κοινωνικής δικτύωσης Social Media

Στο κοινωνικό δίκτυο το ζητούμενο είναι να εξάγουμε χρήσιμη πληροφορία χρησιμοποιώντας γεγονότα που αναπαριστούν το δίκτυο και κανόνες που ορίζουν πιο πολύπλοκες σχέσεις, βασισμένοι σε απλούστερες σχέσεις. Μια τέτοια πολύπλοκη σχέση είναι για παράδειγμα δύο χρήστες που είναι φίλοι (friends) όταν ο ένας ακολουθεί τον άλλον ή ότι ο χρήστης έχει φίλους του ίδιου φύλου (friend_same_gender). Στο πρόγραμμα μας μπορούμε να δούμε: follows(petros,ilias). --> ο petros ακολουθεί τον ilias , και, follows(petros,dimitris). --> ο petros ακολουθεί τον dimitris, κ.ο.κ. Δύο χρήστες X και Y είναι “φίλοι” όταν ακολουθεί ο ένας τον

άλλον, δηλαδή πρέπει να ισχύει ταυτόχρονα ο X να ακολουθεί τον Y και ο Y τον X. Οπότε, επεκτείνεται έτσι το πρόγραμμα προσθέτοντας τον εξής κανόνα: friends(X,Y):- follows(X,Y), follows(Y,X).

Ο κανόνας που μας δείχνει εάν ο χρήστης έχει φίλους του ίδιου φύλου θα πρέπει να εκφραστεί σε δύο εναλλακτικές περιπτώσεις, μια για κάθε φύλο:

- friend_same_gender(X,Y):-male(X), male(Y), friends(X,Y).
- friend_same_gender(X,Y):-female(X), female(Y), friends(X,Y).

Η πρώτη πρόταση εκφράζει πως ο X είναι φίλος του ίδιου φύλου με τον Y μόνο εάν ο X είναι άρρεν και ο Y είναι άρρεν και ο X με τον Y είναι φίλοι. Η δεύτερη πρόταση εκφράζει πως η X είναι φίλη του ίδιου φύλου με την Y μόνο εάν η X είναι θηλύ και η Y είναι θηλύ και η X με την Y είναι φίλες [7].

1.4. SWI-Prolog

Το SWI-Prolog είναι μια ευέλικτη εφαρμογή της γλώσσας Prolog. Αν και το SWI Prolog κέρδισε τη δημοτικότητά του κυρίως στην εκπαίδευση, η ανάπτυξή του οφείλεται κυρίως στις ανάγκες ανάπτυξης εφαρμογών. Αυτό διευκολύνεται από μια πλούσια διεπαφή με άλλα στοιχεία πληροφορικής υποστηρίζοντας πολλούς τύπους εγγράφων και πρωτόκολλα (δικτύου), καθώς και μια ολοκληρωμένη διεπαφή χαμηλού επιπέδου με το C που αποτελεί τη βάση για διεπαφές υψηλού επιπέδου σε C++, Java (ομαδοποιημένα), C#, Python κ.λπ. Οι επεκτάσεις τύπου δεδομένων, όπως οι υπαγορεύσεις και οι συμβολοσειρές, καθώς και η πλήρης υποστήριξη για το Unicode και τους μη δεσμευμένους ακέραιους απλοποιούν την ομαλή ανταλλαγή δεδομένων με άλλα στοιχεία. Παράλληλα η SWI-Prolog στοχεύει στην επεκτασιμότητα. Η ισχυρή υποστήριξή της για πολυνηματικό προγραμματισμό (multi-threading) εκμεταλλεύεται αποτελεσματικά το υλικό πολλαπλών πυρήνων και απλοποιεί την ενσωμάτωση της σε ταυτόχρονες εφαρμογές. Το Just In Time Indexing (JITI) παρέχει διαφανή και αποτελεσματική υποστήριξη για κατηγορήματα που περιλαμβάνουν εκατομμύρια όρους. Επιπλέον η SWI Prolog ενοποιεί πολλές επεκτάσεις της βασικής γλώσσας που έχουν αναπτυχθεί στην κοινότητα της Prolog, όπως καταλογογράφηση (tabling), περιορισμοί (constraints), καθολικές (global) μεταβλητές, καταστροφική ανάθεση (destructive assignment) και interactors. Τέλος η SWI-Prolog προσφέρει μια ποικιλία εργαλείων ανάπτυξης, τα περισσότερα από τα οποία μπορούν να συνδυαστούν κατά βούληση. Το εγγενές

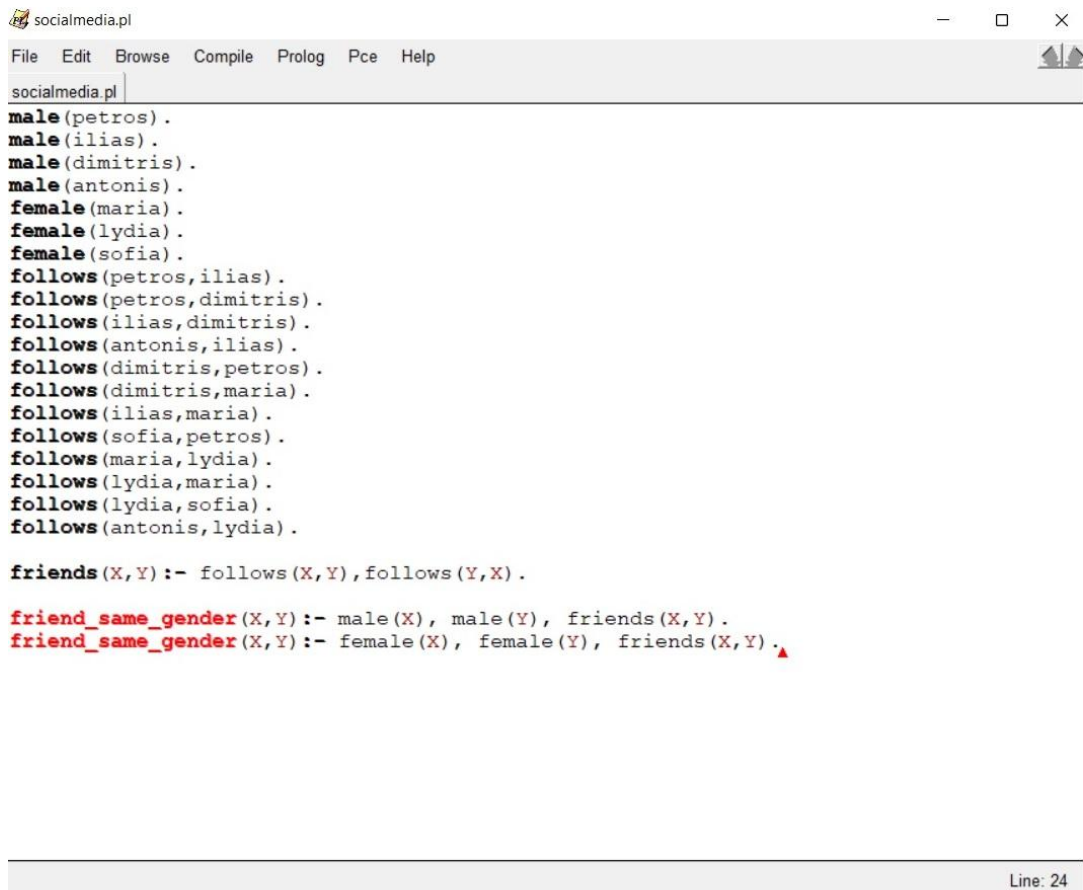
σύστημα παρέχει έναν editor γραμμένο σε Prolog που αποτελεί στενό κλώνο του Emacs. Επίσης παρέχει σημασιολογική επισήμανση βασισμένη σε ανάλυση που γίνεται σε πραγματικό χρόνο από το ίδιο το σύστημα της Prolog. Τα συμπληρωματικά εργαλεία περιλαμβάνουν ένα γραφικό debugger, profiler, και ένα cross-referencer. Εναλλακτικά, υπάρχει και η επιλογή για το GNU-Emacs, το Eclipse plugin που ονομάζεται PDT, και ένα VSC plugin. Παρόλο που η Prolog αναγνωρίζεται ευρέως ως γλώσσα ειδικού σκοπού για εργασίες όπως η αξιολόγηση κανόνων, θεωρούμε ότι είναι κυρίως μια πλατφόρμα που είναι κατάλληλη για χρήση ως κόλλα μεταξύ διαφόρων στοιχείων. Ο κύριος λόγος για αυτό είναι ότι τα δεδομένα βρίσκονται στον πυρήνα πολλών σύγχρονων εφαρμογών ενώ υπάρχει μια μεγάλη ποικιλία στην οποία δομούνται και αποθηκεύονται δεδομένα. Οι κλασσικές γλώσσες ερωτημάτων όπως SQL, SPARQL, XPATH, κ.λπ. μπορούν να ασχοληθούν με μια τέτοια μορφή μόνο, ενώ η Prolog μπορεί να παρέχει μια συνοπτική και φυσική γλώσσα ερωτημάτων για καθεμία από αυτές τις μορφές που μπορεί είτε να εκτελεστεί απευθείας είτε να μεταγλωττιστεί σε ειδική γλώσσα ερωτημάτων εκφράσεις. Το σχεσιακό παράδειγμα της Prolog ταιριάζει καλά με δεδομένα σε πίνακα (RDBMS), ενώ η βελτιστοποιημένη υποστήριξη για αναδρομικό κώδικα ταιριάζει καλά με δεδομένα σε σχήμα δέντρου και γραφήματος (RDF). Στο πλαίσιο της πτυχιακής εργασίας η SWI-Prolog χρησιμοποιήθηκε για την δημιουργία μιας εφαρμογής εύρεσης διαδρομών [16].



Εικόνα 8: Παράθυρο εκτέλεσης εντολών SWI Prolog.

1.5. Εκτέλεση παραδείγματος social media στη SWI-Prolog

Στο σημειωματάριο (editor) του SWI-Prolog δημιουργούμε ένα αρχείο με κατάληξη .pl όπου γράφουμε όλο το πρόγραμμα που είδαμε στο προηγούμενο παράδειγμα ώστε να μπορέσουμε να εκτελέσουμε ερωτήματα.



```
socialmedia.pl
File Edit Browse Compile Prolog Pce Help
socialmedia.pl
male(petros).
male(ilias).
male(dimitris).
male(antonis).
female(maria).
female(lydia).
female(sofia).
follows(petros,ilias).
follows(petros,dimitris).
follows(ilias,dimitris).
follows(antonis,ilias).
follows(dimitris,petros).
follows(dimitris,maria).
follows(ilias,maria).
follows(sofia,petros).
follows(maria,lydia).
follows(lydia,maria).
follows(lydia,sofia).
follows(antonis,lydia).

friends(X,Y):- follows(X,Y), follows(Y,X).

friend_same_gender(X,Y):- male(X), male(Y), friends(X,Y).
friend_same_gender(X,Y):- female(X), female(Y), friends(X,Y).
```

Line: 24

Εικόνα 8: Δημιουργία αρχείου .pl με το πρόγραμμα social media.

Στη συνέχεια φορτώνουμε το πρόγραμμα μας κάνοντας consult το αρχείο socialmedia.pl που δημιουργήσαμε. Κατόπιν μπορούμε να θέσουμε κάποια ερωτήματα (queries) στη βάση γνώσεων του προγράμματος με σκοπό την εμφάνιση των “ φίλων ” που υπάρχουν, εάν είναι του ίδιου φύλου κλπ.



```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.1)
File Edit Settings Run Debug Help
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic) or ?- apropos(Word).

?-
?- c:/Users/nusb/Documents/Prolog/socialmedia.pl compiled 0.02 sec. --2 clauses
?-
|
```

Εικόνα 9: Κάνοντας consult το αρχείο socialmedia.pl .


```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.1)
File Edit Settings Run Debug Help
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?-
% c:/Users/numb2/Documents/Prolog/socialmedia.pl compiled 0.00 sec, 22 clauses
?-
| follows(petros,ilias).
| true .
?- follows(antonis,ilias).
| true .
?- follows(dimitris,lydia).
| false .
?- follows(maria,lydia),follows(lydia,maria).
| true .
?- █
```

Εικόνα 10: Στιγμιότυπο ερωτήσεων (α).

Όπως βλέπουμε στο παραπάνω στιγμιότυπο έχουμε τέσσερα ερωτήματα στα οποία η prolog επιστρέφει true εάν βρει αντιστοιχία με τα γεγονότα που έχουν δοθεί ειδάλλως επιστρέφει false.

- `follows(petros,ilias)`. Εξήγηση: Ακολουθεί ο petros τον ilias? Σύμφωνα με τη βάση γνώσεων η Prolog μας επιστρέφει true.
- `follows(antonis,ilias)`. Εξήγηση: Ακολουθεί ο antonis τον ilias? Σύμφωνα με τη βάση γνώσεων η Prolog μας επιστρέφει true.
- `follows(dimitris,lydia)`. Εξήγηση: Ακολουθεί ο dimitris την lydia? Σύμφωνα με τη βάση γνώσεων η Prolog μας επιστρέφει false.
- `follows(maria,lydia),follows(lydia,maria)`. Εξήγηση: Ακολουθεί η maria την lydia και η lydia την maria? Σύμφωνα με τη βάση γνώσεων η Prolog μας επιστρέφει true.

```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.1)
File Edit Settings Run Debug Help
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?-
% c:/Users/numb2/Documents/Prolog/socialmedia.pl compiled 0.00 sec, 22 clauses
?-
| follows(petros,ilias).
true .

?- follows(antonis,ilias).
true .

?- follows(dimitris,lydia).
false.

?- follows(maria,lydia),follows(lydia,maria).
true .

?- follows(sofia,X).
X = petros.

?- follows(sofia,_).
true.

?- follows(lydia,X).
X = maria ;
X = sofia.

?- █
```

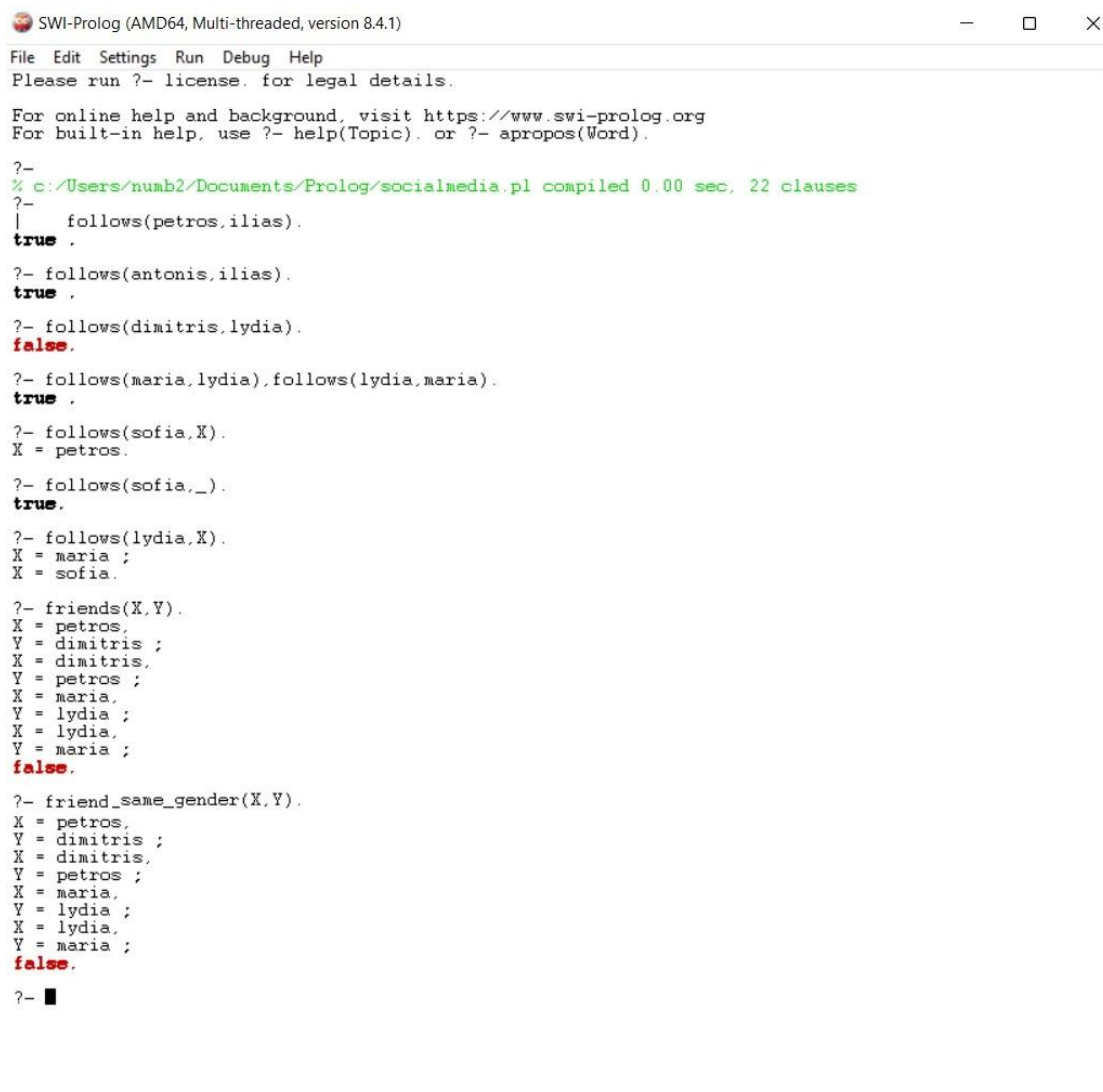
Εικόνα 11: Εκτέλεση ερωτημάτων με τη χρήση μίας μεταβλητής.

Στο στιγμιότυπο αυτό μπορούμε να δούμε τρία ερωτήματα που χρησιμοποιούν μια μεταβλητή:

- `follows(sofia,X)`. Εξήγηση: Η εισαγωγή της μεταβλητής στην ερώτηση, διατυπώνει ουσιαστικά το ερώτημα “υπάρχει X, για το οποίο η λογική σχέση `follows(sofia,X)` να είναι αληθής;” και η Prolog απαντά δίνοντας (ενοποιώντας) την μεταβλητή X με την σταθερά `petros`.
- `follows(sofia,_)`. Εξήγηση: Η εισαγωγή της μεταβλητής “_” στην ερώτηση, διατυπώνει ουσιαστικά το ερώτημα “υπάρχει X, για το οποίο η λογική σχέση `follows(sofia,_)` να είναι αληθής; Εάν υπάρχει δεν θέλω να μάθω το όνομα, μόνο εάν ισχύει”.
- `follows(lydia,X)`. Εξήγηση: Η εισαγωγή της μεταβλητής στην ερώτηση, διατυπώνει ουσιαστικά το ερώτημα “υπάρχει X, για το οποίο η λογική σχέση

follows(lydia,X) να είναι αληθής;” και η Prolog απαντά δίνοντας (ενοποιώντας) την μεταβλητή X με την σταθερά maria.

Μια γρήγορη ανάγνωση του προγράμματος, δείχνει ότι υπάρχουν και άλλα γεγονότα στο πρόγραμμα τα οποία θα μπορούσαν να αποδώσουν στο X τιμές ώστε η σχέση να αληθεύει. Αν ζητηθεί από την Prolog να παράξει όλες τις εναλλακτικές λύσεις (πατώντας το πλήκτρο “;”) θα εμφανίσει όσα ονόματα υπάρχουν ως απάντηση. Στην ουσία εδώ η Prolog κάνει οπισθοδρόμηση (backtracking) για να επιστρέψει εναλλακτικές λύσεις.



```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.1)
File Edit Settings Run Debug Help
Please run ?- license. for legal details.
For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?-
% c:/Users/numb2/Documents/Prolog/socialmedia.pl compiled 0.00 sec, 22 clauses
?-
| follows(petros,ilias).
true .
?- follows(antonis,ilias).
true .
?- follows(dimitris,lydia).
false .
?- follows(maria,lydia),follows(lydia,maria).
true .
?- follows(sofia,X).
X = petros.
?- follows(sofia,_).
true .
?- follows(lydia,X).
X = maria ;
X = sofia.
?- friends(X,Y).
X = petros,
Y = dimitris ;
X = dimitris,
Y = petros ;
X = maria,
Y = lydia ;
X = lydia,
Y = maria ;
false .
?- friend_same_gender(X,Y).
X = petros,
Y = dimitris ;
X = dimitris,
Y = petros ;
X = maria,
Y = lydia ;
X = lydia,
Y = maria ;
false .
?- ■
```

Εικόνα 12: Εκτέλεση ερωτημάτων με τη χρήση δύο μεταβλητών.

- friends(X,Y). Εξήγηση: Ποιοι είναι φίλοι ? Εδώ χρησιμοποιείται ο κανόνας friends(X,Y):- follows(X,Y), follows (Y,X). Η prolog εκτελεί τον κανόνα και αντιστοιχίζει στις δύο μεταβλητές αυτούς που είναι φίλοι μεταξύ τους, δηλαδή ακολουθούν ο ένας τον άλλον [16].

- `friend_same_gender(X,Y)`. Εξήγηση: Ποιοί είναι φίλοι και είναι του ίδιου φύλου? Εδώ χρησιμοποιούνται οι κανόνες: `friend_same_gender(X,Y):-male(X), male(Y), friends(X,Y)` και `friend_same_gender(X,Y):-female(X), female(Y), friends(X,Y)`. Η Prolog εκτελεί τους κανόνες και αντιστοιχίζει στις μεταβλητές αυτούς που είναι φίλοι και είναι του ίδιου φύλου [17].

1.6. Visual Prolog

Η Visual Prolog είναι μια γλώσσα προγραμματισμού πολλαπλών παραδειγμάτων που βασίζεται στη λογική γλώσσα Prolog. Ο στόχος του Visual Prolog είναι να διευκολύνει τις προγραμματικές λύσεις σύνθετων προβλημάτων με έμφαση στη γνώση. Η Visual Prolog είναι μια ισχυρή και ασφαλής γλώσσα προγραμματισμού υψηλού επιπέδου που συνδυάζει τα καλύτερα χαρακτηριστικά των λογικών, λειτουργικών και αντικειμενοστραφών παραδειγμάτων προγραμματισμού με συνεπή και κομψό τρόπο. Η Visual Prolog μπορεί να δημιουργήσει εφαρμογές για τις πλατφόρμες Microsoft Windows 32/64. Υποστηρίζει προηγμένες λύσεις πελάτη-διακομιστή και τριών επιπέδων και είναι ιδιαίτερα κατάλληλο για την αντιμετώπιση σύνθετων γνώσεων. Χρησιμοποιώντας το ισχυρό σύστημα αντικειμένων, μπορείτε να αρχιτεκτονήσετε την εφαρμογή πολύ άκαμπτα και ταυτόχρονα να επωφεληθείτε από την πολύ χαλαρή σύζευξη. Αυτό επιτρέπει τη μείωση του κόστους ανάπτυξης και ακόμη περισσότερο το κόστος συντήρησης. Η Visual Prolog υπερέχει στην αναπαράσταση και τη διαχείριση δομημένης, επίσημης γνώσης. Οι προγραμματιστές μπορούν να κωδικοποιήσουν τη γνώση, συμπεριλαμβανομένων γεγονότων, κανόνων και σχέσεων, χρησιμοποιώντας μια λογική και δηλωτική σύνταξη. Οι βάσεις γνώσεων που δημιουργούνται στο Visual Prolog επιτρέπουν την αποθήκευση πληροφοριών που μπορούν να υποβληθούν σε αναζήτηση και να χειριστούν, διευκολύνοντας την εξαγωγή πληροφοριών και λύσεων. Η γλώσσα αξιοποιεί το συλλογισμό βασισμένο σε κανόνες για λογικά συμπεράσματα, εξαγωγή συμπερασμάτων και λήψη αποφάσεων με βάση τη διαθέσιμη γνώση. Αυτή η ποιότητα καθιστά την Visual Prolog μια εξαιρετική επιλογή για την ανάπτυξη εφαρμογών που βασίζονται στη γνώση. Η Visual Prolog απλοποιεί την ανταλλαγή γνώσης μεταξύ ανθρώπων και μηχανών, προσφέροντας μια ευανάγνωστη και ερμηνεύσιμη σύνταξη για την έκφραση της γνώσης. Αυτή η δυνατότητα ενθαρρύνει

την αποτελεσματική συνεργασία με ειδικούς του τομέα και ενδιαφερόμενα μέρη, επιτρέποντας την αποτύπωση της τεχνογνωσίας τους. Επιπλέον, καθώς η γνώση εξελίσσεται με την πάροδο του χρόνου, η Visual Prolog παρέχει μηχανισμούς για την ενημέρωση και τη διατήρηση των βάσεων γνώσης, διασφαλίζοντας την ακρίβεια και τη νομισματική μονάδα των έμπειρων συστημάτων και των συστημάτων υποστήριξης αποφάσεων. Συνοπτικά, η λογική βάση της Visual Prolog επιτρέπει τον αποτελεσματικό χειρισμό της γνώσης, προσφέροντας ένα ισχυρό πλαίσιο για την αναπαράσταση, τη συλλογιστική και τη διαχείριση της γνώσης. Αποδεικνύεται ανεκτίμητο στη δημιουργία έξυπνων και εντατικών εφαρμογών γνώσης, όπως διαγνωστικά συστήματα, συστήματα υποστήριξης αποφάσεων, μηχανές συστάσεων, έμπειρα συστήματα και εφαρμογές κατανόησης φυσικής γλώσσας [11].

Κεφάλαιο 2ο: Σχεδίαση εφαρμογής GPS “WayToHome” - Βάση γνώσεων - UML

Περίληψη

Η σχεδίαση της εφαρμογής GPS “WayToHome” πραγματοποιήθηκε στη γλώσσα προγραμματισμού Prolog. Η εφαρμογή αυτή απευθύνεται κυρίως σε παιδιά που τώρα ξεκινάνε να μαθαίνουν να επιστρέφουν μόνα τους στο σπίτι αφού σχολάσουν από το σχολείο. Το ζητούμενο της συγκεκριμένης εφαρμογής είναι η επίδειξη της διαδρομής που πρέπει να ακολουθήσει ο χρήστης από ένα αρχικό σημείο A (σχολείο) προς ένα τελικό σημείο B (σπίτι).

2.1. Βάση γνώσης (Knowledge base)

Η βάση γνώσης της εφαρμογής GPS “WayToHome” περιλαμβάνει 24 σύνθετους όρους εκ των οποίων οι 15 είναι η δήλωση των ονομάτων των οδών που μας ενδιαφέρουν όπως πχ. `street(glinou)`, ενώ οι 3 από τους 18 σύνθετους όρους είναι η δήλωση της κατεύθυνσης πχ. `way(aristera)`. Ακολουθούν 14 ακόμη σύνθετοι όροι οι οποίοι δηλώνουν ποια κατεύθυνση πρέπει να ακολουθήσει κάποιος από μια οδό προς μία άλλη πχ. `directWay(doiranis,papafi,eftheia)`.

Βάση γνώσης:

`street(glinou).`

`street(panteli_horn).`

`street(larisis).`

`street(grigoriou_E).`

`street(leof_karamanli).`

`street(papanastasiou).`

`street(hrisostomou).`

`street(perdika).`

`street(doiranis).`

`street(papafi).`

`street(katsimidi).`

`street(paraskevopoulou).`

`street(delfwn).`

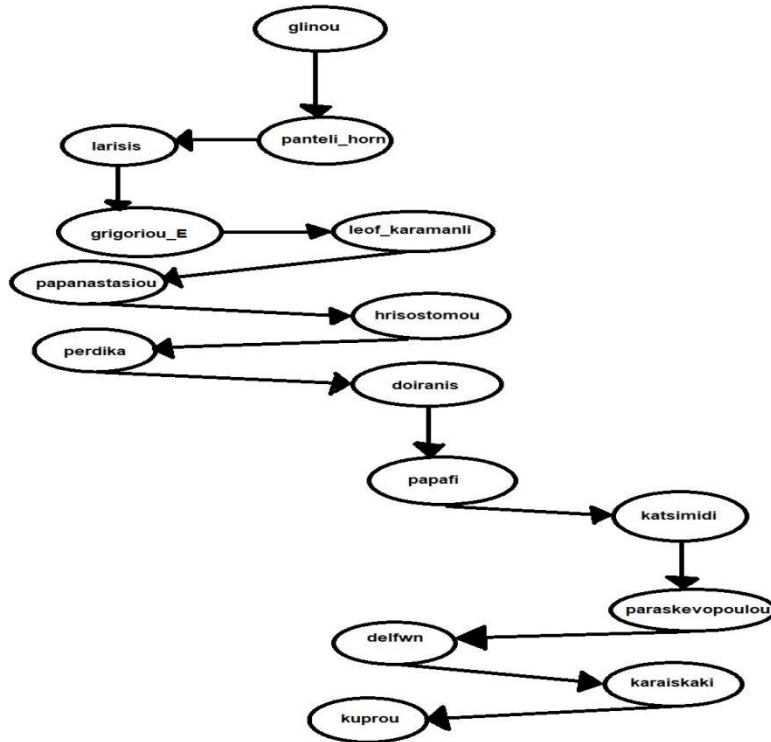
`street(karaiskaki).`

`street(kuprou).`

way(aristera).
way(deksia).
way(eftheia).
directWay(glinou,panteli_horn,eftheia).
directWay(panteli_horn,larisis,aristera).
directWay(larisis,grigoriou_E,eftheia).
directWay(grigoriou_E,leof_karamanli,deksia).
directWay(leof_karamanli,papanastasiou,aristera).
directWay(papanastasiou,hrisostomou,deksia).
directWay(hrisostomou,perdika,aristera).
directWay(perdika,doiranis,deksia).
directWay(doiranis,papafi,eftheia).
directWay(papafi,katsimidi,deksia).
directWay(katsimidi,paraskevopoulou,eftheia).
directWay(paraskevopoulou,delfwn,aristera).
directWay(delfwn,karaiskaki,deksia).
directWay(karaiskaki,kuprou,aristera).

2.2. UML σχέδιο εφαρμογής GPS “WayToHome”

Στο παρακάτω UML σχέδιο απεικονίζονται οι όροι και οι σχέσεις μεταξύ τους όπως περιγράφονται μέσα στη βάση γνώσης της εφαρμογής GPS “WayToHome”.



Εικόνα 13: UML σχέδιο εφαρμογής GPS “WayToHome”.

2.3. Ανάλυση βάσης γνώσης

Στη βάση γνώσης μας μπορούμε να δούμε:

- `street(glinou)`, `street(panteli_horn)`, `street(Iarisis)`, `street(grigoriou_E)`. Άτομα που δηλώνουν τα ονόματα των οδών.
- `way(aristera)`, `way(deksia)`, `way(eftheia)`. Άτομα που δηλώνουν κατεύθυνση για αριστερά, δεξιά, ευθεία.
- `directWay(glinou,panteli_horn,eftheia)`. Σύνθετος όρος: υπάρχει διαδρομή από την οδό “glinou” με κατεύθυνση ευθεία προς την οδό “panteli_horn”.
- `directWay(panteli_horn,Iarisis,aristera)`. Σύνθετος όρος: υπάρχει διαδρομή από την οδό “panteli_horn” με κατεύθυνση αριστερά προς την οδό “Iarisis”.
- `directWay(Iarisis,grigoriou_E,eftheia)`. Σύνθετος όρος: υπάρχει διαδρομή από την οδό “Iarisis” με κατεύθυνση ευθεία προς την οδό “grigoriou_E”.
- `directWay(grigoriou_E,leof_karamanli,deksia)`. Σύνθετος όρος: υπάρχει διαδρομή από την οδό “grigoriou_E” με κατεύθυνση δεξιά προς την οδό “leof_karamanli”.

- `directWay(leof_karamanli,papanastasiou,aristera)`. Σύνθετος όρος: υπάρχει διαδρομή από την οδό “leof_karamanli” με κατεύθυνση αριστερά προς την οδό “papanastasiou”.
- `directWay(papanastasiou,hristostomou,deksia)`. Σύνθετος όρος: υπάρχει διαδρομή από την οδό “papanastasiou” με κατεύθυνση δεξιά προς την οδό “hristostomou”.
- `directWay(hristostomou,perdika,aristera)`. Σύνθετος όρος: υπάρχει διαδρομή από την οδό “hristostomou” με κατεύθυνση αριστερά προς την οδό “perdika”.
- `directWay(perdika,doiranis,deksia)`. Σύνθετος όρος: υπάρχει διαδρομή από την οδό “perdika” με κατεύθυνση δεξιά προς την οδό “doiranis”.
- `directWay(doiranis,parafi,eftheia)`. Σύνθετος όρος: υπάρχει διαδρομή από την οδό “doiranis” με κατεύθυνση ευθεία προς την οδό “parafi”.
- `directWay(parafi,katsimidi,deksia)`. Σύνθετος όρος: υπάρχει διαδρομή από την οδό “parafi” με κατεύθυνση δεξιά προς την οδό “katsimidi”.
- `directWay(katsimidi,paraskevouroulou,eftheia)`. Σύνθετος όρος: υπάρχει διαδρομή από την οδό “katsimidi” προς την οδό “paraskevouroulou”.
- `directWay(paraskevouroulou,delfwn,aristera)`. Σύνθετος όρος: υπάρχει διαδρομή από την οδό “paraskevouroulou” με κατεύθυνση αριστερά προς την οδό “delfwn”.
- `directWay(delfwn,karaiskaki,deksia)`. Σύνθετος όρος: υπάρχει διαδρομή από την οδό “delfwn” με κατεύθυνση δεξιά προς την οδό “karaiskaki”.
- `directWay(karaiskaki,kuprou,aristera)`. Σύνθετος όρος: υπάρχει διαδρομή από την οδό “karaiskaki” με κατεύθυνση αριστερά προς την οδό “kuprou”.

Κεφάλαιο 3ο: Παρουσίαση κώδικα εφαρμογής GPS “WayToHome”

Περίληψη

Η σχεδίαση της εφαρμογής GPS “WayToHome” πραγματοποιήθηκε στη γλώσσα προγραμματισμού Prolog και συγκεκριμένα στο περιβάλλον SWI-Prolog [4].

3.1. Κώδικας εφαρμογής GPS “WayToHome”

Ο κώδικας της εφαρμογής GPS “WayToHome” περιλαμβάνει την βάση γνώσης που περιγράφηκε στο κεφάλαιο 2 η οποία ακολουθείται από 2 κανόνες ονόματι “walkBetween” οι οποίοι ορίζουν την τελική απάντηση στις ερωτήσεις που θα πραγματοποιήσει ο τελικός χρήστης για το ποια διαδρομή και ποια κατεύθυνση θα πρέπει να ακολουθήσει από ένα σημείο προς ένα άλλο.

Κώδικας:

street(glinou).

street(panteli_horn).

street(larisis).

street(grigoriou_E).

street(leof_karamanli).

street(papanastasiou).

street(hrisostomou).

street(perdika).

street(doiranis).

street(papafi).

street(katsimidi).

street(paraskevopoulou).

street(delfwn).

street(karaiskaki).

street(kuprou).

way(aristera).

way(deksia).

way(eftheia).

directWay(glinou,panteli_horn,eftheia).

```

directWay(panteli_horn,larisis,aristera).
directWay(larisis,grigoriou_E,eftheia).
directWay(grigoriou_E,leof_karamanli,deksia).
directWay(leof_karamanli,papanastasiou,aristera).
directWay(papanastasiou,hrisostomou,deksia).
directWay(hrisostomou,perdika,aristera).
directWay(perdika,doiranis,deksia).
directWay(doiranis,papafi,eftheia).
directWay(papafi,katsimidi,deksia).
directWay(katsimidi,paraskevopoulou,eftheia).
directWay(paraskevopoulou,delfwn,aristera).
directWay(delfwn,karaiskaki,deksia).
directWay(karaiskaki,kuprou,aristera).

walkBetween(X,Y,M):- directWay(X,Y,M),write(X),write('-->'),write(M),
write('-->'),write(Y).

walkBetween(X,Y,A):- directWay(X,Z,M),write(X),write('-->'),write(M),
write('-->'), write(Z),
    write('-->'),walkBetween(Z,Y,A).

```

3.2. Ανάλυση κώδικα εφαρμογής GPS “WayToHome”

Στο προηγούμενο κεφάλαιο αναλύθηκε η βάση γνώσεων της εφαρμογής. Για να ολοκληρωθεί ένας κώδικας εφαρμογής χρειάζεται και κανόνες.

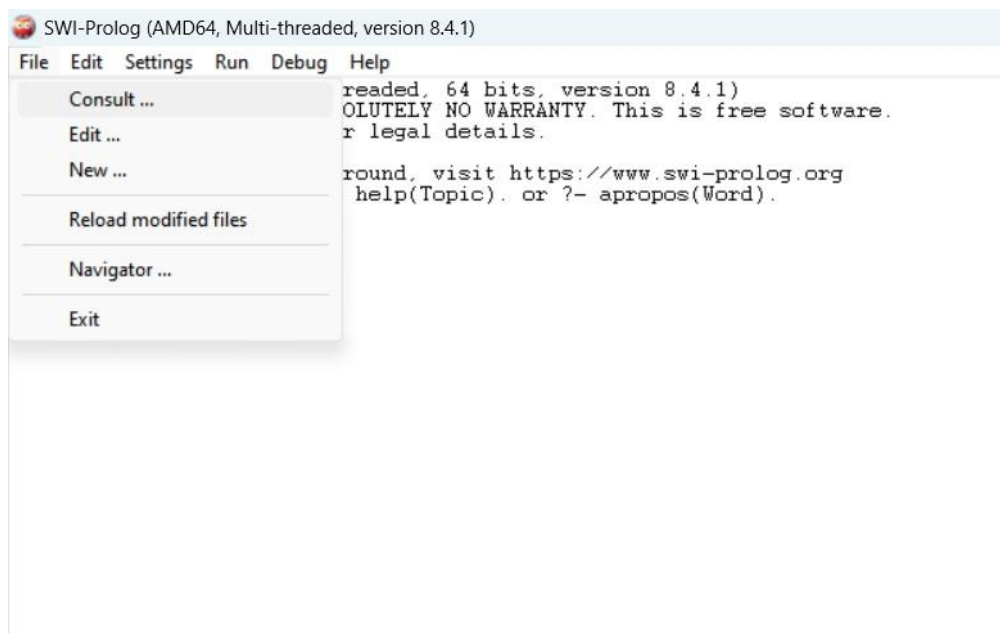
Πρώτα χρησιμοποιείται ο κανόνας: `walkBetween(X,Y,M):-directWay(X,Y,M),write(X),write('-->'),write(M),write('-->'),write(Y)`. όπου X και Y δύο σημεία και M η κατεύθυνση. Ο κανόνας αυτός σημαίνει πως εάν υπάρχει ο σύνθετος όρος `directWay` μεταξύ δύο οδών τότε θα πρέπει να εμφανιστεί η διαδρομή και η κατεύθυνση μεταξύ τους με βελάκια από το ένα σημείο στο άλλο. Έπειτα, προστίθεται ένας πιο πολύπλοκος κανόνας: `walkBetween(X,Y,A):-directWay(X,Z,M),write(X),write('-->'),write(M),write('-->'), write(Z),write('-->'),walkBetween(Z,Y,A)`. όπου X και Y δύο σημεία και M η κατεύθυνση. Ο κανόνας αυτός σημαίνει πως εάν υπάρχει ο σύνθετος όρος `directWay` μεταξύ δύο οδών και πληρείται ο πρώτος κανόνας τότε θα πρέπει να εμφανιστεί η διαδρομή και η κατεύθυνση μεταξύ τους με βελάκια από το ένα σημείο στο άλλο.

3.3. Εκτέλεση του κώδικα εφαρμογής GPS “WayToHome” με παραδείγματα

Για την εκτέλεση της εφαρμογής πρέπει να δημιουργηθεί και να αποθηκευτεί αρχείο με κατάληξη .pl και να περιέχει τον κώδικα. Κατόπιν πρέπει να γίνει consult στο περιβάλλον της SWI-Prolog [3].

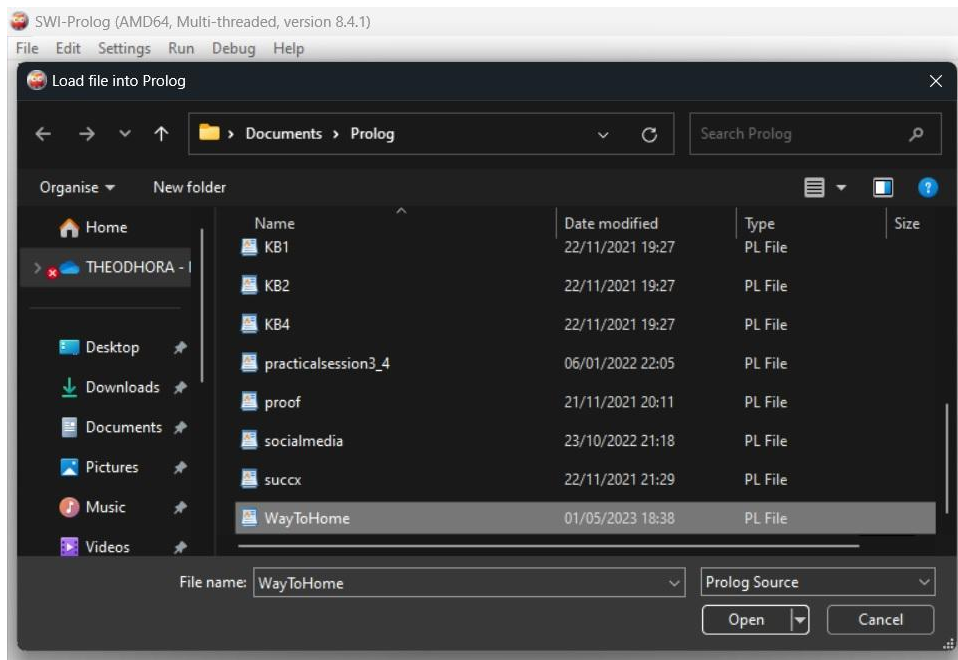
3.3.1. Βήματα εκτέλεσης της εφαρμογής

Στο 1ο βήμα ο χρήστης ανοίγει το SWI-Prolog και επιλέγει File -> Consult.



Εικόνα 14: 1ο βήμα εκτέλεσης εφαρμογής.

Στο 2ο βήμα ο χρήστης επιλέγει το αρχείο και το ανοίγει.



Εικόνα 15: 2ο βήμα εκτέλεσης εφαρμογής.

Αμέσως μετά θα εμφανιστεί το εξής:

```

SWI-Prolog (AMD64, Multi-threaded, version 8.4.1)
File Edit Settings Run Debug Help
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?-
% c:/Users/numb2/Documents/Prolog/WayToHome.pl compiled 0.00 sec, -2 clauses

```

Εικόνα 16: Εμφάνιση παραθύρου κατόπιν 2ου βήματος εκτέλεσης εφαρμογής.

3.3.2. Παραδείγματα εκτέλεσης της εφαρμογής

Στο 1ο παράδειγμα θα δούμε την εκτέλεση μιας απλής εντολής. Στην εντολή αυτή ο χρήστης ζητά από το πρόγραμμα να εμφανίσει τις πιθανές οδούς που έχει η βάση γνώσης.

```

SWI-Prolog (AMD64, Multi-threaded, version 8.4.1)
File Edit Settings Run Debug Help
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?-
% c:/Users/numb2/Documents/Prolog/WayToHome.pl compiled 0.00 sec, 34 clauses
?-
|   street(X).
X = glinou ;
X = panteli_horn ;
X = laris ;
X = grigoriou_E ;
X = leof_karamanli ;
X = papanastasiou ;
X = hrisostomou ;
X = perdika ;
X = doiranis ;
X = papafi ;
X = katsimidi ;
X = paraskevopoulou ;
X = delfwn ;
X = karaiskaki ;
X = kuprou.

?- █

```

Εικόνα 17: 1ο παράδειγμα - Εμφάνιση πιθανών οδών.

Στο 2ο παράδειγμα θα δούμε την εκτέλεση μιας ακόμη απλής εντολής. Στην εντολή αυτή ο χρήστης ζητά να μάθει εάν ισχύει πως υπάρχει συγκεκριμένη διαδρομή. Η απάντηση θα δοθεί με ένα “true” ή ένα “false”.

```

SWI-Prolog (AMD64, Multi-threaded, version 8.4.1)
File Edit Settings Run Debug Help
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?-
% c:/Users/numb2/Documents/Prolog/WayToHome.pl compiled 0.00 sec, 34 clauses
?-
|   street(X).
X = glinou ;
X = panteli_horn ;
X = laris ;
X = grigoriou_E ;
X = leof_karamanli ;
X = papanastasiou ;
X = hrisostomou ;
X = perdika ;
X = doiranis ;
X = papafi ;
X = katsimidi ;
X = paraskevopoulou ;
X = delfwn ;
X = karaiskaki ;
X = kuprou.

?- directWay(glinou,panteli_horn,eftheia).
true.

?- █

```

Εικόνα 18: 2ο παράδειγμα - Επιβεβαίωση υπάρχουσας διαδρομής.

Στο 3ο παράδειγμα ο χρήστης ζητά να μάθει τη κατεύθυνση από τη μια οδό προς δεύτερη οδό. Η απάντηση θα πρέπει να είναι περιγραφική με βελάκια σύμφωνα με τους κανόνες του κώδικα και θα πρέπει να δοθεί ως απάντηση “aristera” ή “deksia” ή “eftheia”.

```

SWI-Prolog (AMD64, Multi-threaded, version 8.4.1)
File Edit Settings Run Debug Help
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?-
% c:/Users/numb2/Documents/Prolog/WayToHome.pl compiled 0.00 sec, -2 clauses
?-
|   walkBetween(glinou,panteli_horn,M).
glinou-->eftheia-->panteli_horn
M = eftheia .

?- █

```

Εικόνα 19: 3ο παράδειγμα - Ερώτηση κατεύθυνσης.

Στο 4ο παράδειγμα ο χρήστης ζητά να μάθει τη διαδρομή και την κατεύθυνση από μία οδό που γνωρίζει ήδη στην επόμενη που δε γνωρίζει ποια είναι. Η απάντηση θα πρέπει να είναι περιγραφική με βελάκια σύμφωνα με τους κανόνες του κώδικα και θα πρέπει να δοθεί ως απάντηση το όνομα της επόμενης οδού και η κατεύθυνση “aristera” ή “deksia” ή “eftheia” [10].

```

SWI-Prolog (AMD64, Multi-threaded, version 8.4.1)
File Edit Settings Run Debug Help
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?-
% c:/Users/numb2/Documents/Prolog/WayToHome.pl compiled 0.00 sec, -2 clauses
?-
|   walkBetween(glinou,panteli_horn,M).
glinou-->eftheia-->panteli_horn
M = eftheia .

?- walkBetween(glinou,Y,M).
glinou-->eftheia-->panteli_horn
Y = panteli_horn,
M = eftheia .

?-

```

Εικόνα 20: 4ο παράδειγμα - Ερώτηση διαδρομής και κατεύθυνσης.

Στο 5ο παράδειγμα ο χρήστης ζητά να μάθει τις πιθανές διαδρομές και τη κατεύθυνση από μια οδό που γνωρίζει ήδη. Η απάντηση θα πρέπει να είναι περιγραφική με βελάκια σύμφωνα με τους κανόνες του κώδικα και θα πρέπει να δοθούν ως απαντήσεις οι επόμενες οδοί όπως και η κατεύθυνση προς κάθε μία. Όταν δεν βρεθούν άλλες πιθανές διαδρομές η απάντηση θα είναι “false” [17].

```

SWI-Prolog (AMD64, Multi-threaded, version 8.4.1)
File Edit Settings Run Debug Help

?- walkBetween(larisis,Y,M).
larisis-->eftheia-->grigoriou_E
Y = grigoriou_E.
M = eftheia ;
larisis-->eftheia-->grigoriou_E-->grigoriou_E-->deksia-->leof_karaaanli
Y = leof_karaaanli.
M = deksia ;
grigoriou_E-->deksia-->leof_karaaanli-->leof_karaaanli-->aristera-->papanastasiou
Y = papanastasiou.
M = aristera ;
leof_karaaanli-->aristera-->papanastasiou-->papanastasiou-->deksia-->hrisostomou
Y = hrisostomou.
M = deksia ;
papanastasiou-->deksia-->hrisostomou-->hrisostomou-->aristera-->perdika
Y = perdika.
M = aristera ;
hrisostomou-->aristera-->perdika-->perdika-->deksia-->doiranis
Y = doiranis.
M = deksia ;
perdika-->deksia-->doiranis-->doiranis-->eftheia-->papafi
Y = papafi.
M = eftheia ;
doiranis-->eftheia-->papafi-->papafi-->deksia-->katsiaini
Y = katsiaini.
M = deksia ;
papafi-->deksia-->katsiaini-->katsiaini-->eftheia-->paraskevopoulou
Y = paraskevopoulou.
M = eftheia ;
katsiaini-->eftheia-->paraskevopoulou-->paraskevopoulou-->aristera-->delfvn
Y = delfvn.
M = aristera ;
paraskevopoulou-->aristera-->delfvn-->delfvn-->deksia-->karaiskaki
Y = karaiskaki.
M = deksia ;
delfvn-->deksia-->karaiskaki-->karaiskaki-->aristera-->kuprou
Y = kuprou.
M = aristera ;
karaiskaki-->aristera-->kuprou-->
false.
?-

```

Εικόνα 21: 5ο παράδειγμα - Ερώτηση πιθανών διαδρομών και κατευθύνσεων.

Στο 6ο παράδειγμα ο χρήστης ζητά να μάθει όλες τις πιθανές διαδρομές που υπάρχουν και τις κατευθύνσεις τους. Η απάντηση θα πρέπει να είναι περιγραφική με βελάκια σύμφωνα με τους κανόνες του κώδικα και θα πρέπει να δοθούν ως απαντήσεις όλα τα ονόματα οδών οι οποίες έχουν συσχέτιση μεταξύ τους, η διαδρομή μεταξύ τους και η κατεύθυνση. Για να γίνει αυτό θα πρέπει ο χρήστης να μην πατήσει “enter” αλλά το σύμβολο “;” [8].


```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.1)
File Edit Settings Run Debug Help
glinou-->eftheia-->panteli_horn
Y = panteli_horn,
M = eftheia ,

?- walkBetween(X,Y,M).
glinou-->eftheia-->panteli_horn
X = glinou,
Y = panteli_horn,
M = eftheia .

?- walkBetween(X,Y,M).
glinou-->eftheia-->panteli_horn
X = glinou,
Y = panteli_horn,
M = eftheia ;
panteli_horn-->aristera-->larisis
X = panteli_horn,
Y = larisis,
M = aristera ;
larisis-->eftheia-->grigoriou_E
X = larisis,
Y = grigoriou_E,
M = eftheia ;
grigoriou_E-->deksia-->leof_karamanli
X = grigoriou_E,
Y = leof_karamanli,
M = deksia ;
leof_karamanli-->aristera-->papanastasiou
X = leof_karamanli,
Y = papanastasiou,
M = aristera ;
papanastasiou-->deksia-->hrisostomou
X = papanastasiou,
Y = hrisostomou,
M = deksia ;
hrisostomou-->aristera-->perdika
X = hrisostomou,
Y = perdika,
M = aristera ;
perdika-->deksia-->doiranis
X = perdika,
Y = doiranis,
M = deksia ;
doiranis-->eftheia-->papafi
X = doiranis,
Y = papafi,
M = eftheia ;
papafi-->deksia-->katsimidi
X = papafi,
Y = katsimidi,
M = deksia ;
katsimidi-->eftheia-->paraskevopoulou
X = katsimidi,
Y = paraskevopoulou,
M = eftheia ;
paraskevopoulou-->aristera-->delfwn
X = paraskevopoulou,
Y = delfwn,
M = aristera
```

Εικόνα 22: 6ο παράδειγμα - Ερώτηση πιθανών διαδρομών και κατευθύνσεων.

Στο 7ο και τελευταίο παράδειγμα ο χρήστης ζητά τη διαδρομή από την αρχική οδό έως τη τελική. Η απάντηση θα πρέπει να είναι περιγραφική με βελάκια σύμφωνα με τους κανόνες του κώδικα και θα πρέπει να δοθούν ως απαντήσεις όλα τα ονόματα οδών οι οποίες έχουν συσχέτιση μεταξύ τους, η διαδρομή μεταξύ τους και η κατεύθυνση με τελικό προορισμό την οδό “kyprou” [15].

```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.1)
File Edit Settings Run Debug Help

?- walkBetween(glinou, kuprou, M).
glinou-->eftheia-->panteli_horn-->panteli_horn-->aristera-->larisis-->larisis-->eftheia-->grigoriou_
E-->grigoriou_E-->deksia-->leof_karamanli-->leof_karamanli-->aristera-->papanastasiou-->papanastasio
u-->deksia-->hrisostomou-->hrisostomou-->aristera-->perdika-->perdika-->deksia-->doiranis-->doiranis
-->eftheia-->papafi-->papafi-->deksia-->katsimidi-->katsimidi-->eftheia-->paraskevopoulou-->paraskev
opoulou-->aristera-->delfwn-->delfwn-->deksia-->karaiskaki-->karaiskaki-->aristera-->kuprou
M = aristera .

?-
```

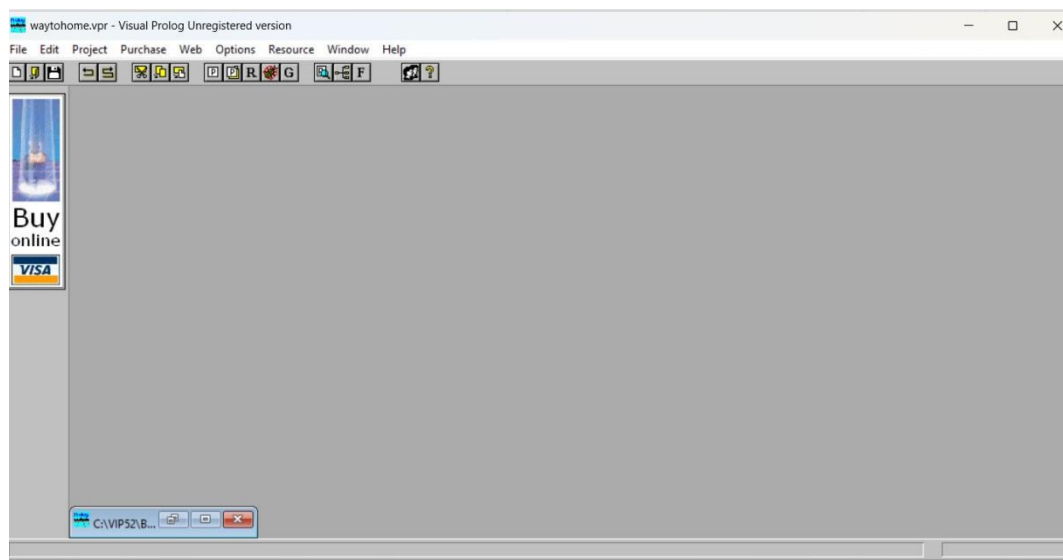
Εικόνα 23: 7ο παράδειγμα - Ερώτηση διαδρομής από αρχική έως τελική οδό.

Κεφάλαιο 4ο: Εκτέλεση του κώδικα εφαρμογής GPS “WayToHome” στο περιβάλλον Visual Prolog.

Περίληψη

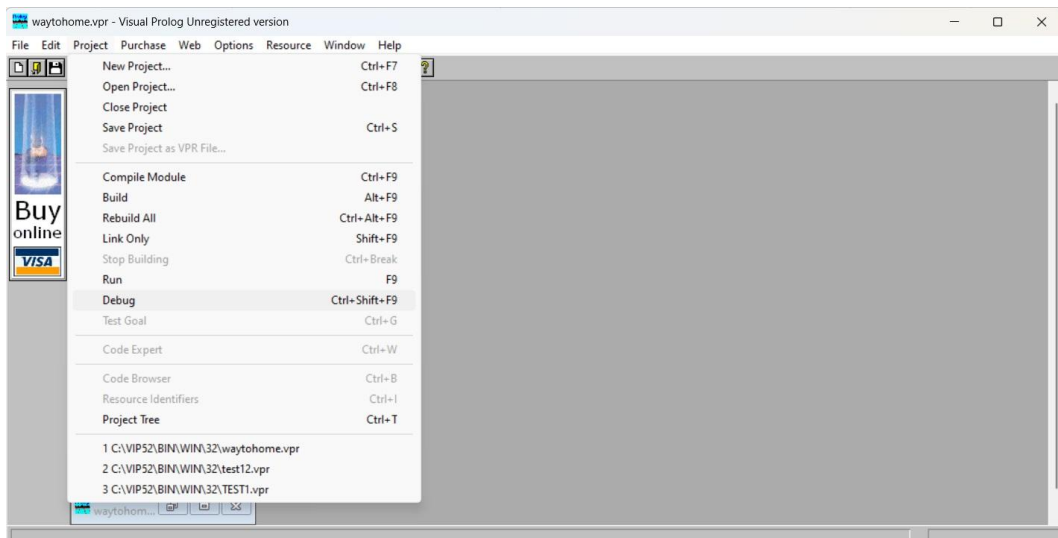
Η σχεδίαση της εφαρμογής GPS “WayToHome” πραγματοποιήθηκε στη γλώσσα προγραμματισμού Prolog ενώ για την εκτέλεση της εφαρμογής χρησιμοποιήθηκε το περιβάλλον της Visual Prolog το οποίο μπορεί κάποιος να κατεβάσει δωρεάν από το διαδίκτυο [11].

4.1. Βήματα εκτέλεσης του κώδικα εφαρμογής GPS “WayToHome”



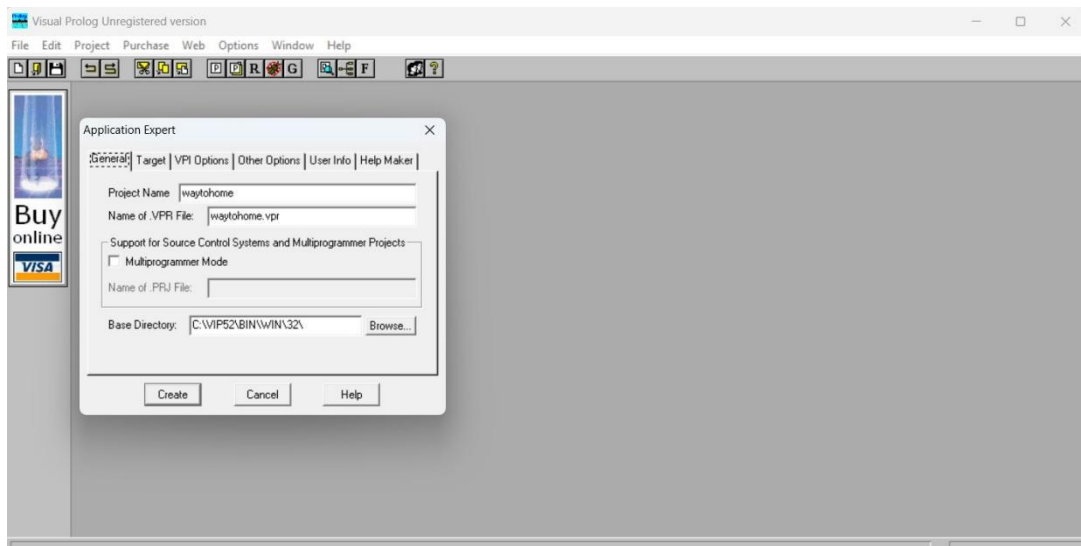
Εικόνα 24: Περιβάλλον Visual Prolog

Η επόμενη κίνηση είναι να δημιουργήσουμε μια νέα εργασία ή αλλιώς “new project”.



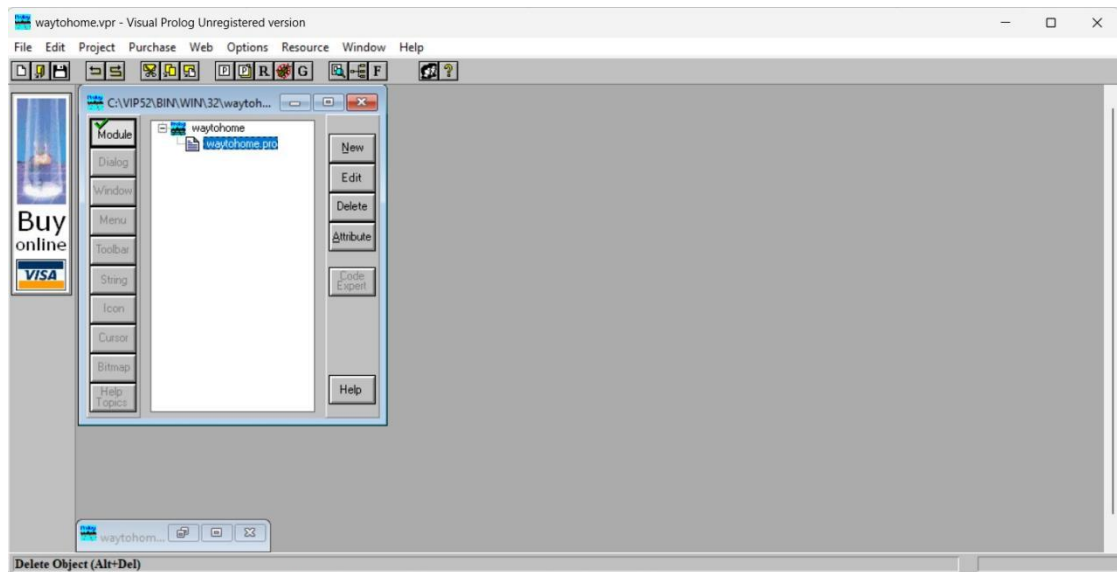
Εικόνα 25: Δημιουργία new project

Επιλέγουμε το “new project” και το ονομάζουμε “waytohome”.



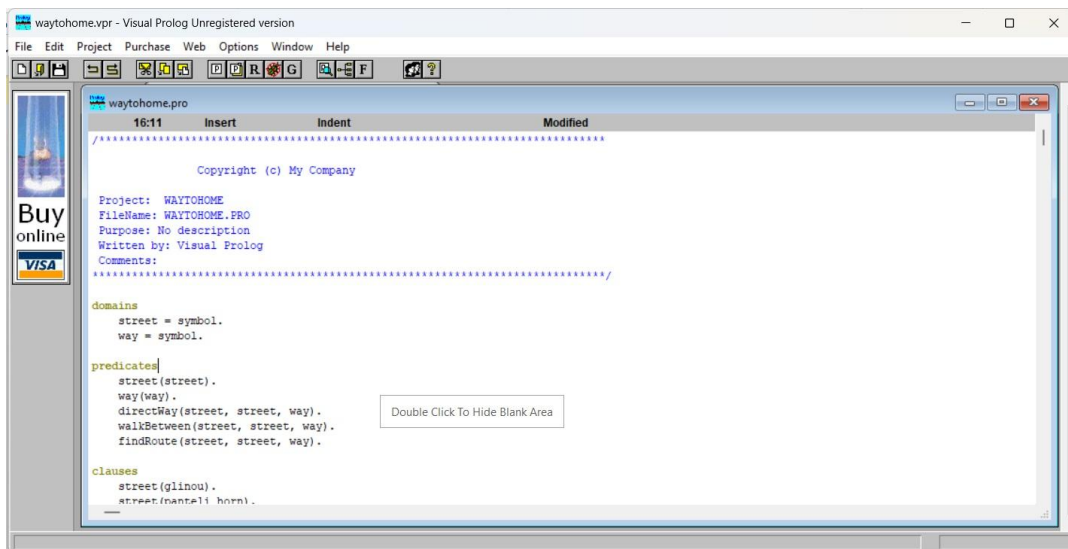
Εικόνα 26: Project name “waytohome”

Επιλέγουμε την επιλογή “create” και δημιουργείται το project.



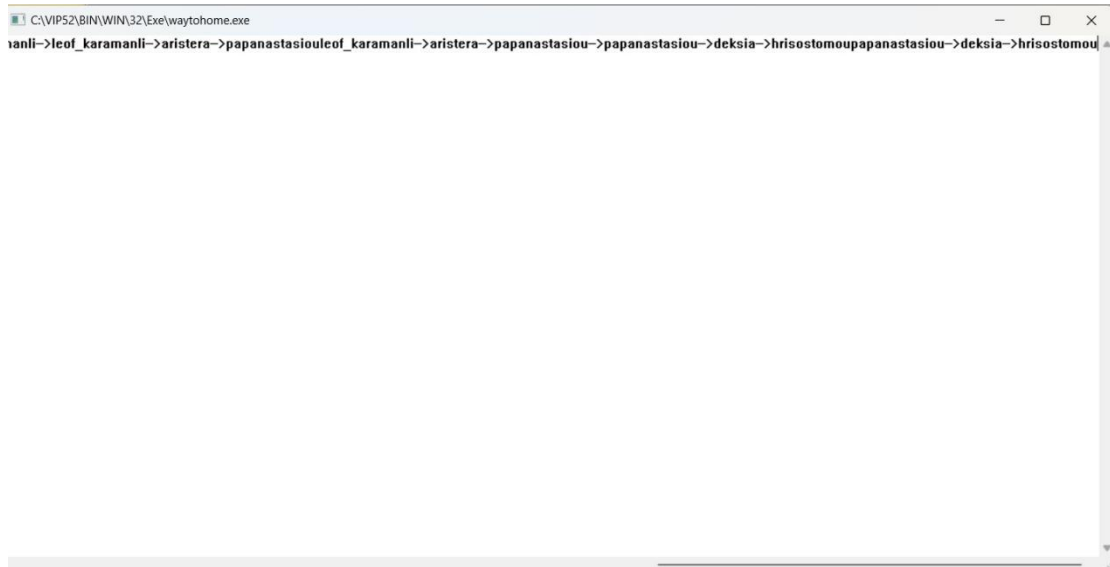
Εικόνα 27: Δημιουργία project “waytohome”

Επιλέγουμε και ανοίγουμε το αρχείο “waytohome.pro” και γράφουμε τον κώδικα μας.



Εικόνα 28: Παράθεση κώδικα “waytohome”

Τέλος, εκτελούμε τον κώδικα μας από την επιλογή “project” επιλέγοντας το “run”.



Εικόνα 29: Εκτέλεση κώδικα “waytohome”

4.2. Κώδικας εφαρμογής “WayToHome” στο περιβάλλον Visual Prolog.

domains

street = symbol.

way = symbol.

predicates

street(street).

way(way).

directWay(street, street, way).

walkBetween(street, street, way).

findRoute(street, street, way).

clauses

street(glinou).

street(panteli_horn).

street(larisis).

street(grigoriou_E).

street(leof_karamanli).

street(papanastasiou).

street(hrisostomou).

street(perdika).

street(doiranis).

street(papafi).

street(katsimidi).

street(paraskevopoulou).

street(delfwn).

street(karaiskaki).

street(kuprou).

way(aristera).

way(deksia).

way(eftheia).

directWay(glinou, panteli_horn, eftheia).

directWay(panteli_horn, laris, aristera).

directWay(laris, grigoriou_E, eftheia).

directWay(grigoriou_E, leof_karamanli, deksia).

directWay(leof_karamanli, papanastasiou, aristera).

directWay(papanastasiou, hrisostomou, deksia).

directWay(hrisostomou, perdika, aristera).

directWay(perdika, doiranis, deksia).

directWay(doiranis, papafi, eftheia).

directWay(papafi, katsimidi, deksia).

directWay(katsimidi, paraskevopoulou, eftheia).

directWay(paraskevopoulou, delfwn, aristera).

directWay(delfwn, karaiskaki, deksia).

directWay(karaiskaki, kuprou, aristera).

walkBetween(X, Y, M) :- directWay(X, Y, M),

write(X), write("-->"), write(M), write("-->"), write(Y).

```
findRoute(X, Y, A) :- walkBetween(X, Z, M),  
    write(X), write("-->"), write(M), write("-->"), write(Z),  
    write("-->"), findRoute(Z, Y, A).
```

goal

```
findRoute(glinou, kuprou, A),  
    write("Path: "), write(A), write("%!\n").
```


Συμπεράσματα

Οι εφαρμογές GPS έχουν αποδειχθεί ιδιαίτερα σημαντικές για τον άνθρωπο καθώς χρησιμοποιούνται σε όλους τους τομείς του σημερινού κόσμου. Εκτός από την καίρια στρατιωτική σημασία της χρήσης τους, από όπου και ξεκίνησαν, οι εφαρμογές GPS μπορούν να βοηθήσουν στην ανάλυση δεδομένων ανάλογα από την περιοχή στην οποία βρίσκεται ο κάθε χρήστης ανά πάσα στιγμή, πληροφορίες που χρησιμοποιούνται ακόμη και στα τμήματα μάρκετινγκ μεγάλων εταιριών με σκοπό την αύξηση των διαφημίσεων τους με αποτέλεσμα την αύξηση κερδών. Για τον απλό κόσμο όμως, μια εφαρμογή GPS έχει απελευθερώσει όσους ήθελαν να εξερευνήσουν νέα μέρη και μονοπάτια με τον φόβο πως θα αποπροσανατολιζόνταν και θα έβρισκαν αδιέξοδο ή γονείς που λόγω εργασίας αναγκάζονται να αφήνουν τα παιδιά τους να πηγαινοέρχονται στις δραστηριότητες μόνα τους με κίνδυνο πάντα να χαθούν.

Βιβλιογραφία

- [1] Μαρακάκης Μ., «Prolog: Προγραμματισμός σε Λογική για Τεχνητή Νοημοσύνη». Εκδόσεις Νέων Τεχνολογιών, NewTech Pub, 2019.
- [2] Μαυρομιχάλη Ε. Κ. , «Εισαγωγή στη Γλώσσα Προγραμματισμού Prolog,» 2006. Available: <https://icsdweb.aegean.gr/stamatatos/courses/Logic/Prolog/>.
- [3] Ντάκος Γ., Μεταπτυχιακή διατριβή με τίτλο: «Εφαρμογή Μεταβλητών Καλλιεργητικών Φροντίδων σε Οπωρώνες», Σχολή Γεωπονικών Επιστημών, Πανεπιστήμιο Θεσσαλίας, 2015. Available: [13826.pdf \(uth.gr\)](#) .
- [4] Σακελλαρίου Η., Βασιλειάδης Ν., Κεφαλάς Π., Σταμάτης Δ., « Τεχνικές Λογικού Προγραμματισμού: Η γλώσσα prolog.», Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών, 2015. Available: [Τεχνικές Λογικού Προγραμματισμού \(ihu.gr\)](#) .
- [5] Britannica, «PROLOG computer language». 2019, Available: [PROLOG | computer language | Britannica](#) .
- [6] GEOAWESOME, «Learning GIS programming: An overview», 18.08.2014, Available: [Learning GIS programming: An overview - Geoawesomeness](#) .
- [7] «Learn Prolog Now!» by Patrick Blackburn, Johan Bos, and Kristina Striegnitz. 15.06.2006, Available: [Learn Prolog Now! \(rug.nl\)](#) .
- [8] Michael J.Dunn, DISL, DAF, Technical Director, «Global Positioning Systems Directorate - Systems Engineering & Integration. Interface Specification IS-GPS-200. Navstar GPS Space Segment/Navigation User Interfaces» . 21.10.2014. Available: [IS-GPS-200H.pdf](#) .
- [9] Sixt Magazine, «20 Best free GPS Navigation Apps & Traffic Apps for 2023». 03.02.2023, Available: [Top 20 Free GPS Navigation Apps & Traffic Apps in 2023 | Sixt](#) .
- [10] «SWI-Prolog's features», 30.11.2022, Available: [SWI-Prolog's features](#) .
- [11] Visual Prolog, 2022, Available: [Visual Prolog \(visual-prolog.com\)](#) .
- [12] Wikipedia, «Apple Maps». 22.02.2022, Available: [Apple Maps - Wikipedia](#) .
- [13] Wikipedia, «Cut (logic programming)», 08.12.2022, Available: [Cut \(logic programming\) - Wikipedia](#) .
- [14] Wikipedia, «Global Positioning System». 04.05.2023, Available: [Global Positioning System - Βικιπαίδεια \(wikipedia.org\)](#) .
- [15] Wikipedia, «Logic programming», 05.12.2022 [Logic programming - Wikipedia](#) .

[16] Wikipedia, «Prolog», 10.02.2022, Available: [Prolog - Βικιπαίδεια \(wikipedia.org\)](#) .

[17] Wikipedia, «Prolog», 20.11.2022, Available: [Prolog - Wikipedia](#) .

[18] Wikipedia, «Visual Prolog», 10.12.2022, Available: [Visual Prolog - Wikipedia](#) .