ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΜΟΝΤΕΛΟΠΟΙΗΣΗ ΤΩΝ ΚΑΝΟΝΩΝ ΠΑΙΧΝΙΔΙΟΥ BLACKJACK ΜΕ ΤΥΠΙΚΕΣ ΜΕΘΟΔΟΥΣ

<<Κρίτωνας Στίγγας>>

Α.Μ. 171119

Εισηγητής: << Κωνσταντίνος Μπάρλας, Ε.ΔΙ.Π>>

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΜΟΝΤΕΛΟΠΟΙΗΣΗ ΤΩΝ ΚΑΝΟΝΩΝ ΠΑΙΧΝΙΔΙΟΥ BLACKJACK ΜΕ ΤΥΠΙΚΕΣ ΜΕΘΟΔΟΥΣ

Κρίτωνας Στίγγας

Α.Μ. <<171119>>

Εισηγητής:

<< Κωνσταντίνος Μπάρλας, Ε.ΔΙ.Π >> Konstantinos Barlas Digitally signed by Konstantinos Barlas Date: 2024.04.12 18:22:40 +03'00'

Εξεταστική Επιτροπή:

<< Σταύρος Φατούρος, Αναπληρωτής Καθηγητής>> Stavros Fatouros Digitally signed by Stavros Fatouros Date: 2024.04.12 14:21:00 +03'00'

<< Χρήστος Τρούσσας, Επίκουρος Καθηγητής>> Christos Troussas Digitally signed by Christos Troussas Date: 2024.04.12 16:56:09 +03'00'

Ημερομηνία εξέτασης:

21/03/2024

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΜΕΤΑΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Ο κάτωθι υπογεγραμμένος Στίγγας Κρίτωνας του Νεόφυτου, με αριθμό μητρώου 171119 φοιτητής του Πανεπιστημίου Δυτικής Αττικής της Σχολής Μηχανικών του Τμήματος Μηχανικών Πληροφορικής και Υπολογιστών, δηλώνω υπεύθυνα ότι:

«Βεβαιώνω ότι είμαι συγγραφέας αυτής της Διπλωματικής εργασίας και κάθε βοήθεια την οποία είχα για την προετοιμασία της, είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο.

Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από μένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του Ιδρύματος. Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου μου».

Ο Δηλών

ΕΥΧΑΡΙΣΤΙΕΣ

Με την ολοκλήρωση της διπλωματικής μου εργασίας, θα ήθελα να ευχαριστήσω θερμά τον καθηγητή του Πανεπιστημίου Δυτικής Αττικής και επιβλέποντα της διπλωματικής μου εργασίας κ. Κωνσταντίνο Μπάρλα για την καθοδήγηση και υποστήριξη της προσπάθειας μου. Επίσης θα ήθελα να ευχαριστήσω την οικογένεια μου και τους φίλους μου για τη στήριξη και συμπαράσταση τους όλα αυτά τα χρόνια.

ΠΕΡΙΛΗΨΗ

Αντικείμενο τις διπλωματικής εργασίας αποτελεί η εφαρμογή των Τυπικών Μεθόδων για τη μοντελοποίηση και επαλήθευση των κανόνων παιχνιδιού BlackJack χρησιμοποιώντας αλγεβρικές προδιαγραφές. Οι αλγεβρικές προδιαγραφές οι οποίες είναι γλώσσες, τεχνικές και εργαλεία βασισμένες στα μαθηματικά, παρέχουν τη δυνατότητα να αναλύσουμε και να επαληθεύσουμε τις ιδιότητες του συστήματος, περιγράφοντας το μέσω μιας αυστηρά μαθηματικά ορισμένης προδιαγραφής. Μέσω τις εκτελέσιμης γλώσσας αλγεβρικών προδιαγραφών CafeOBJ και με την ενσωμάτωση συμπεριφοριακών προδιαγραφών, το παραγόμενο μοντέλο περιγράφει σε μορφή αλγεβρικής οντότητας το παιχνίδι και επιτρέπει τη μελέτη της οντότητας σε κάθε πιθανή κατάσταση θωρακίζοντας το από κακό σχεδιασμό κανόνων.

Λέξεις κλειδιά: Τυπικές Μέθοδοι, Αλγεβρικές Γλώσσες Προδιαγραφών, CafeOBJ, Τυπική Επαλήθευση, Απόδειξη Θεωρήματος, Μέθοδος coinduction

ABSTRACT

The subject of the thesis is the application of formal methods for modeling and verifying BlackJack game rules using algebraic specifications. Algebraic specifications which are languages, techniques and tools based on mathematics, provide the possibility to analyze and verify the properties of the system by describing it through a rigorously mathematically defined specification. Through the executable algebraic specification language CafeOBJ and the integration of behavioral specifications, the generated model describes the game in the form of an algebraic entity and allows the entity to be studied in every possible situation, protecting it from bad rule design.

Keywords: Formal Methods, Algebraic Specification Languages, CafeOBJ, Formal Verification, Theorem Proof, Behavioral Specifications, coinduction method

ΠΕΡΙΕΧΟΜΕΝΑ

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

ΣΥΝΤΟΜΟΓΡΑΦΊΕΣ

Εισαγωγή

Η αυξανόμενη πολυπλοκότητα των συστημάτων λογισμικού και η εμφάνιση νέων τεχνολογιών, προκάλεσε την ανάγκη για αποτελεσματικές και αξιόπιστες μεθόδους ανάπτυξης λογισμικού. Μια τέτοια μέθοδος είναι η χρήση των Τυπικών Μεθόδων, οι οποίες λόγω των ικανοτήτων τους κατάφεραν να εξασφαλίσουν την αξιοπιστία και ορθότητα των συστημάτων λογισμικού καταφέρανε να ελκύσουν την προσοχή. Με τη χρήση μαθηματικών προτάσεων για την περιγραφή της λειτουργικότητας και της συμπεριφοράς ενός συστήματος λογισμικού με τρόπο που είναι σαφής, ακριβής και αναγνώσιμος από τη μηχανή, οι Τυπικές Μέθοδοι χρησιμοποιούνται σε ένα σύστημα για την μοντελοποίηση και επαλήθευση των απαιτήσεων, τον εντοπισμό σφαλμάτων και άλλων ενδεχόμενων προβλημάτων, νωρίς στη διαδικασία ανάπτυξης, καθώς οι παραδοσιακές τεχνικές επαλήθευσης και δοκιμών ενδέχεται να μην είναι επαρκείς για να εξασφαλίσουν την αξιοπιστία και την ορθότητα του συστήματος. Επιτρέποντας μια αυστηρά αναλυτική περιγραφή της λειτουργικότητας και της συμπεριφοράς ενός συστήματος, οι Τυπικές Μέθοδοι είναι ικανές να υποστηρίξουν σημαντικά τους μηχανικούς λογισμικών στον εντοπισμό πιθανών σφαλμάτων και προβλημάτων, παρέχοντας μια ακριβή βάση για επαλήθευση και δοκιμές.

1.1 Σκοπός Της Διπλωματικής Εργασίας

Στο παρόν έγγραφο, οι Τυπικές Μέθοδοι εφαρμόζονται σε ένα υποσύνολο του παιχνιδιού Blackjack. Η εργασία αυτή περιέχει την ανάπτυξη μιας τυπικής προδιαγραφής του συστήματος και τη χρήση εργαλείων και τεχνικών για την αξιολόγηση και επικύρωση των προδιαγραφών. Μέσω αυτής της διαδικασίας, εξετάζεται η πρακτική αξία των Τυπικών Προδιαγραφών στην μοντελοποίηση και ανάπτυξη υψηλής ποιότητας, αξιόπιστων συστημάτων λογισμικού. Η διπλωματική εργασία έχει ως στόχο να συνεισφέρει στον κλάδο της μηχανικής λογισμικού υποστηρίζοντας την αποτελεσματικότητα των τυπικών προδιαγραφών σε πολύπλοκα συστήματα, χρησιμοποιώντας το παιχνίδι Blackjack ως περιπτωσιολογική μελέτη.

1.2 Δομή της Διπλωματικής Εργασίας

Στα επόμενα κεφάλαια περιγράφονται τα συστατικά στοιχεία του παιχνιδιού Blackjack, οι αρχές και τα χαρακτηριστικά της τυπικής μεθοδολογίας και οι τεχνικές που εφαρμόστηκαν για τη δημιουργία των τυπικών προδιαγραφών. Ακολούθως, παρουσιάζεται η εφαρμογή των προδιαγραφών, καθώς και τα αποτελέσματα και τα συμπεράσματα που προέκυψαν κατά τη διάρκεια της μελέτης.

2 Επεξήγηση Παιχνιδιού BlackJack

Στο κεφάλαιο αυτό γίνεται μια εκτενής περιγραφή των χαρακτηριστικών και της δομής του παιχνιδιού Blackjack, παρουσιάζοντας τους κανόνες και τις δυνατές ενέργειες που μπορούν να συμβούν στα αντικείμενα που το απαρτίζουν. Προσφέρει αναλυτικές πληροφορίες παρέχοντας μια εις βάθος επισκόπηση του παιχνιδιού που είναι απαραίτητες για το σχεδιασμό και την ανάλυση των Τυπικών Προδιαγραφών που παρουσιάζονται στην παρούσα εργασία.

2.1 Τι είναι το παιχνίδι Blackjack - περιγραφή

Το Blackjack είναι ένα στρατηγικό παιχνίδι καρτών τράπουλας, παγκόσμια δημοφιλές στα καζίνο, όπου οι παίκτες δεν ανταγωνίζονται μεταξύ τους αλλά ενάντια στο καζίνο το οποίο εκπροσωπείται από ένα dealer [1]. Συνήθως ένα τραπέζι Blackjack μπορεί να φιλοξενήσει το μέγιστο οκτώ παίκτες, και μπορεί να αποτελείται από μία έως οκτώ τράπουλες όπου η καθεμία περιέχει 52 κάρτες [1], η απόφαση των παραμέτρων καθορίζεται από το εκάστοτε καζίνο. Ο σκοπός του παιχνιδιού είναι οι παίκτες να κερδίσουν τον dealer, έχοντας άθροισμα αξίας καρτών χωρίς να υπερβούν το εικοσιένα που να είναι μεγαλύτερο του αθροίσματος του dealer. Η αξία των καρτών ορίζεται σε δέκα για τις κάρτες των βαλέδων, νταμών και ρηγάδων, σε ένα ή έντεκα για τις κάρτες των άσσων και οι υπόλοιπες κάρτες από το δύο έως το δέκα παίρνουν την αξία που αναγράφεται στην κάρτα τους, τα σύμβολα των καρτών δεν έχουν κανένα ρόλο και δεν επηρεάζουν την κατάσταση του παιχνιδιού [1]. Για να ξεκινήσει ένας γύρος παιχνιδιού, οι παίκτες υποχρεούνται να στοιχηματίσουν χρησιμοποιώντας μάρκες στην ειδικά διαμορφωμένη περιοχή του τραπεζιού και έπειτα ο dealer διαμοιράζει στους παίκτες δύο κάρτες που είναι ορατές σε όλους στο τραπέζι και άλλες δύο για τον εαυτό του με την μία να είναι κρυφή. Οι παίκτες με τη σειρά που βρίσκονται στο τραπέζι επιλέγουν της ενέργειες του χεριού τους μέχρι να σταματήσουν να τραβάνε κάρτες ή να υπερβούν το ανώτατο όριο της συνολικής αξίας και με την ολοκλήρωση τους συνεχίζει ο dealer. Όταν ολοκληρώσει τις κινήσεις του ο dealer προκύπτουν οι νικητές.

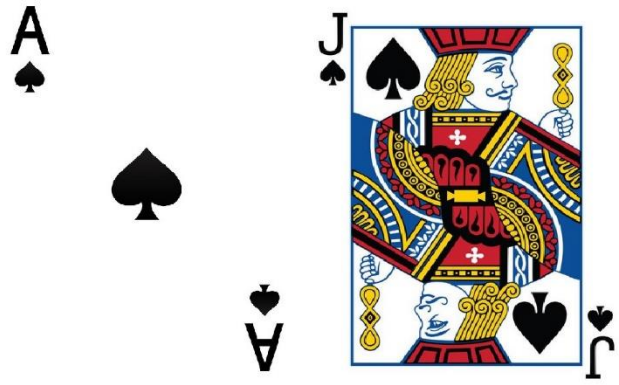
2.2 Κανόνες παιχνιδιού

Το παιχνίδι αποτελείται από τους εξής κανόνες και ενέργειες:

1. Blackjack
2. Ασφάλεια (Insurance)
3. Διακανονισμός
4. Τράβηγμα κάρτας – Διατήρηση καρτών (Hit – Stand)
5. Διάσπαση καρτών (Pair Splitting)
6. Διπλασιασμός στοιχήματος (Double Down)

2.2.1 Blackjack

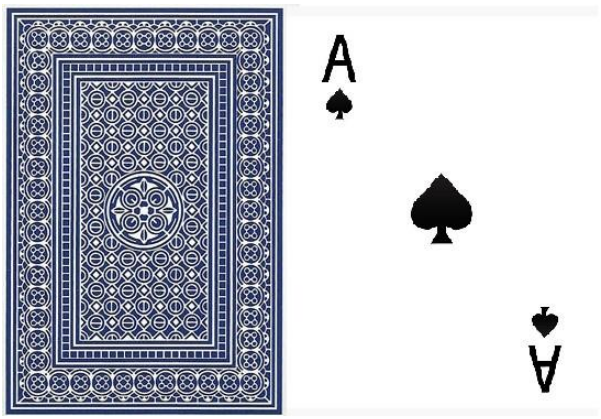
Blackjack νοείται η αξία του αθροίσματος των δύο αρχικών καρτών ισούται με εικοσιένα. Στην περίπτωση που η ορατή κάρτα του dealer αποτιμάται σε δέκα, τότε είναι υποχρεωμένος να δείξει και την κρυφή κάρτα πριν παίξουν οι παίκτες, καθώς στη περίπτωση που έχει Blackjack, δηλαδή κάρτα αξίας δέκα και άσσο, οι παίκτες είναι αδύνατο να κερδίσουν αλλά στην καλύτερη περίπτωση να εξέλθουν ισόπαλοι, σε αντίθετη περίπτωση που ο παίκτης έχει blackjack και δεν έχει ο dealer τότε η αξία του στοιχήματος που κερδίζει πολλαπλασιάζεται [1].



Εικόνα 2.1 : Δομή blackjack, συνολική αξία καρτών εικοσιένα

2.2.2 Ασφάλεια (Insurance)

Όταν η ορατή κάρτα του dealer είναι άσσος, τότε οι παίκτες δικαιούνται να προσθέσουν ένα επιπλέον ασφαλιστικό στοίχημα πριν ο dealer δείξει την κρυφή κάρτα, ούτως ώστε εάν ο dealer επιτύχει blackjack να επιστραφεί στον παίκτη ένα μέρος του αρχικού του στοιχήματος [1].



Εικόνα 2.2 : Δομή καρτών dealer, δυνατότητα ασφάλειας στον παίκτη

2.2.3 Διακανονισμός

Όταν κανένας από τους παίκτες ή τον dealer δεν έχει blackjack, τότε ο κάθε παίκτης με τη σειρά που βρίσκονται στο τραπέζι, κατασκευάζει το ‘χέρι’ αποφασίζοντας τι κινήσεις θα κάνει και τελευταίος κατασκευάζει το χέρι του ο dealer [1].

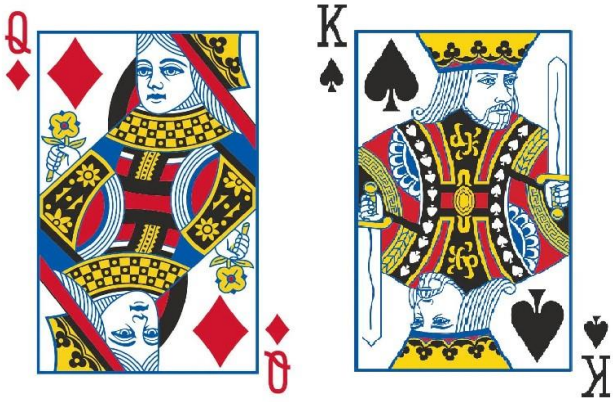
1. Όταν η συνολική αξία του χεριού ενός παίκτη υπερβεί το εικοσιένα, τότε ο παίκτης ‘καίγεται’ και ο dealer κερδίζει, ανεξάρτητα αν και αυτός μετέπειτα ‘καεί’.
2. Εάν ο παίκτης διατηρήσει την αξία των καρτών του μικρότερη ή ίση του εικοσιένα και ο dealer ‘καεί’, τότε ο παίκτης κερδίζει.
3. Εάν η συνολική αξία των καρτών του παίκτη ισούται με την αξία των καρτών του dealer, τότε υπάρχει ισοπαλία και το στοίχημα επιστρέφεται στον παίκτη.
4. Σε οποιαδήποτε άλλη περίπτωση που οι παίκτες δεν έχουν υπερβεί το όριο της αξίας των καρτών, ο νικητής είναι αυτός με τη μεγαλύτερη αξία.

2.2.4 Τράβηγμα κάρτας – Διατήρηση καρτών (Hit – Stand)

Ο παίκτης έχει τη δυνατότητα να διατηρήσει τις κάρτες του εάν τον ικανοποιεί η αξία των καρτών του ή να τραβήξει ακόμα μία κάρτα προσθέτοντας την αξία της νέας κάρτας στο συνολικό άθροισμα των υπολοίπων, μέχρις ότου θελήσει να σταματήσει ή να ‘καεί’ . Ο dealer είναι υποχρεωμένος να σταματήσει όταν η αξία των καρτών του υπερβαίνει το δεκαεπτά και αντίστοιχα να τραβήξει όταν το άθροισμα δεν ξεπερνά το δεκαεπτά.

2.2.5 Διάσπαση καρτών (Pair Splitting)

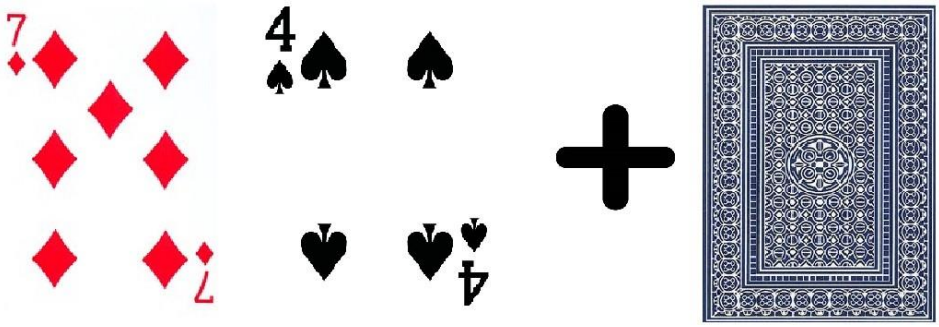
Το δικαίωμα της διάσπασης των καρτών παρέχεται μόνο στους παίκτες. Ένας παίκτης δικαιούται να διαχωρίσει τις κάρτες του σε δύο διαφορετικά ‘χέρια’ μόνο όταν η αξία των αρχικών καρτών που δόθηκαν από τον dealer είναι ίδια. Εάν ο παίκτης αποφασίσει να διασπάσει τις κάρτες του τότε τοποθετεί άλλο ένα στοίχημα που ισούται με το αρχικό και ο dealer συμπληρώνει τα δύο ‘χέρια’ του παίκτη με άλλη μία κάρτα από τη τράπουλα.



Εικόνα 2.3 : Δομή αρχικών καρτών, δυνατότητα διάσπασης καρτών

2.2.6 Διπλασιασμός στοιχήματος (Double)

Ο κάθε παίκτης, έχοντας τις αρχικές δύο κάρτες που διαμοιράστηκαν από τον dealer, έχει τη δυνατότητα να διπλασιάσει το αρχικό του στοίχημα και να ολοκληρώσει τις κινήσεις του προσθέτοντας ακριβώς μια κάρτα στο ‘χέρι’.



Εικόνα 2.4 : Δομή τελικού χεριού μετά από διπλασιασμό

3 Τυπικές Μέθοδοι

Λόγω της δυνατότητας τους να εξασφαλίζουν την ακρίβεια, την ασφάλεια και ορθότητα των λογισμικών συστημάτων, οι Τυπικές Μέθοδοι καθίστανται όλο και πιο δημοφιλείς σε ένα κύκλο ζωής λογισμικού. Αυτό το κεφάλαιο εξερευνά τις πρακτικές και τις αρχές των Τυπικών Μεθόδων, μέσω των τυπικών γλωσσών και μεθόδων για τον καθορισμό προδιαγραφών , την επαλήθευση και την επικύρωση λογισμικών συστημάτων. Επιπλέον περιγράφει τα πλεονεκτήματα των Τυπικών Μεθόδων, όπως η μείωση σφαλμάτων, η εμπιστοσύνη στην ορθότητα και η αναβαθμισμένη συντήρηση λογισμικών συστημάτων καθώς και διάφορα εργαλεία και τεχνικές που χρησιμοποιούνται σε Τυπικές Μεθόδους, όπως είναι η απόδειξη θεωρημάτων και ο έλεγχος μοντέλου. Συνοψίζοντας το κεφάλαιο αυτό παραθέτει μια εκτενή περιγραφή των Τυπικών Μεθόδων στη μηχανική λογισμικού και τονίζει τη σημασία τους για τη δημιουργία λογισμικών εξαιρετικής ποιότητας και αξιοπιστίας.

3.1 Επισκόπηση Τυπικών Μεθόδων

Οι Τυπικές Μέθοδοι στη μηχανική λογισμικού, αποτελούνται από ένα σύνολο τεχνικών που αποσκοπούν στη βελτίωση της ορθότητας και αξιοπιστίας των λογισμικών συστημάτων. Ένα σύστημα με Τυπικές Μεθόδους περιγράφεται χρησιμοποιώντας μια τυπικά καθορισμένη γλώσσα, της οποίας το λεξιλόγιο, το συντακτικό, και η σημασιολογία βασίζονται στα μαθηματικά[2]. Το πεδίο των μαθηματικών που εφαρμόζεται για τον ορισμό αυτών των γλωσσών είναι τα διακριτά μαθηματικά, και οι μαθηματικές έννοιες της λογικής, της θεωρίας συνόλων και της άλγεβρας. Η εφαρμογή των Τυπικών Μεθόδων προσφέρει μια ακριβέστερη σημασιολογία για το σύστημα σχετικά με την αφαιρετικότητα της αρχιτεκτονικής παρέχοντας τη δυνατότητα να εφαρμοστούν σε διάφορα επίπεδα της ανάπτυξης λογισμικού, όπως είναι π.χ ο καθορισμός απαιτήσεων, ο σχεδιασμός συστήματος, η υλοποίηση και η δοκιμή του. Αυτό επιτρέπει μια αυστηρή και αιτιολογημένη ανάλυση των βασικών ιδιοτήτων και διασφαλίζει την ορθότητα του συστήματος καθ' όλη τη διάρκεια του κύκλου ζωής του λογισμικού [3], [4].

3.2 Τυπικές Γλώσσες

Όπως προαναφέρθηκε, οι Τυπικές Μεθόδοι εφαρμόζουν μία τυπικά ορισμένη γλώσσα[1] της οποίας η σημασιολογία, το λεξιλόγιο και το συντακτικό της βασίζεται στα μαθηματικά. Μια τυπική γλώσσα η οποία στηρίζεται στα μαθηματικά, σε συνδυασμό με τα μέσα που χρησιμοποιούνται για την επίδειξη της συνέπειας και της εκτελεσιμότητας του συστήματος προσφέρει μια πιο σαφής και αναλυτική περιγραφή. Αυτό προκύπτει από το γεγονός ότι τα μαθηματικά προσφέρουν ακρίβεια, σαφήνεια παρέχοντας πειστικά επιχειρήματα για να αιτιολογηθούν ότι οι λύσεις και η απόδειξη της εφαρμογής είναι εφικτή[5].

[1] αναφέρεται και σαν επίσημη/τυπική γλώσσα προδιαγραφής.

3.2.1 Στοιχεία Τυπικής Γλώσσας

Σύμφωνα με τον ορισμό του Wing[4], μια τυπική γλώσσα απαρτίζεται από τα παρακάτω στοιχεία:

- **Syn :** Ο όρος 'Syn' αναφέρεται στο σύνολο των κανόνων σύνταξης (syntax) που ισχύουν για την τυπική γλώσσα. Αυτό το σύνολο περιλαμβάνει μια σειρά από σύμβολα, όπως γράμματα και σύμβολα λειτουργιών, και μια συλλογή γραμματικών κανόνων που καθορίζουν πώς αυτά τα σύμβολα μπορούν να χρησιμοποιηθούν για τη δημιουργία τυπικών προτάσεων.

- **Sem :** Ο όρος 'Sem' αναφέρεται στο σύνολο των σημασιολογικών απαιτήσεων (semantics) της γλώσσας. Αυτό το σύνολο περιλαμβάνει έννοιες που παρέχουν νόημα σύμφωνα με την κατηγορία της γλώσσας (π.χ. γλώσσες με βάση το μοντέλο, αλγεβρικές γλώσσες).

- **Sat**: Ο όρος 'Sat' αναφέρεται στην επιτυχή ικανοποίηση της σχέσης μεταξύ των σημασιολογικών απαιτήσεων (Semantics) και των συντακτικών κανόνων (Syntax). Μέσω αυτής της ικανοποίησης, προσδίδεται η σχέση μεταξύ των στοιχείων που συνθέτουν μια τυπική πρόταση.

Ορισμός κατα Wing : Μία τυπική γλώσσα < Syn, Sem, Sat > και $if\ Sat(syn, sem)$ τότε το syn είναι μια προδιαγραφή του sem και το sem ένα ειδικό και του syn.

Παράδειγμα

Τυπική Γλώσσα: Προτασιακή λογική

Τυπικά καθορισμένη πρόταση: (p ⋁ q) → r

Σύνταξη (Syntax): Το σύμβολο ⋁ αντιπροσωπεύει την διάζευξη (λογικό τελεστή OR), το σύμβολο → αντιπροσωπεύει τον τελεστή λογικής συνεπαγωγής(δηλαδή, εάν τότε) και τα p, q και r είναι λογικές προτασιακές μεταβλητές (δηλαδή, μεταβλητές Boolean που μπορούν να χαρακτηριστούν είτε true είτε false).

Σημασιολογία (Semantics): Η πρόταση επιβεβαιώνει ότι αν η υπόθεση (p) ή η υπόθεση (q) είναι αληθές, τότε και το συμπέρασμα (r) θα πρέπει να είναι αληθές.

Ικανοποίηση (Satisfaction): Η πρόταση θεωρείται ικανοποιημένη μόνο εάν η λογική τιμή τις πρότασης του συμπεράσματος (r) προκύπτει από τις λογικές τιμές των προτάσεων-υποθέσεων (p, q). Παραδείγματος χάριν, εάν το p είναι ψευδές, το q είναι ψευδές και το r είναι ψευδές, τότε η πρόταση δεν είναι ικανοποιημένη. Εν τούτοις, αν το p είναι ψευδές, το q είναι αληθές και το r είναι αληθές, τότε η πρόταση ικανοποιείται.

Εξήγηση/αποσαφήνιση χρήσης της πρότασης:

Έστω ότι ένα λογισμικό που πρέπει να επαληθεύσει εάν ένα email είναι έγκυρο για να το αποδεχτεί. Χρησιμοποιώντας την πρόταση (p ∨ q) → r για τον καθορισμό των κριτηρίων, όπου το p αντιπροσωπεύει “το email περιέχει έγκυρη διεύθυνση αποστολέα”, το q αντιπροσωπεύει “το περιεχόμενο του email είναι έγκυρο” και το r αντιπροσωπεύει “ το πρόγραμμα αποδέχεται το email” τότε η πρόταση (p ∨ q) → r σημαίνει ότι εάν το email περιέχει έγκυρη διεύθυνση αποστολέα ή το περιεχόμενο του email είναι έγκυρο, τότε το πρόγραμμα θα αποδεχτεί το email. Η πρόταση μπορεί να είναι χρήσιμη για τον έλεγχο εάν ένα email ανταποκρίνεται στις απαιτήσεις ούτως ώστε να γίνει αποδεκτό.

3.2.2 Κατηγορίες Τυπικών Γλωσσών

Η ύπαρξη διαφορετικών τυπικών γλωσσών, με την κάθε γλώσσα να έχει τα δικά της χαρακτηριστικά, καθιστά την επιλογή κατάλληλης γλώσσας για μια συγκεκριμένη εφαρμογή ως ένα κρίσιμο βήμα. Ανάλογα με τα χαρακτηριστικά και τις ιδιότητες της κάθε γλώσσας, διαμορφώνονται οι εξής κατηγορίες[5].

- **Γλώσσες με βάση το μοντέλο :** Οι γλώσσες με βάση το μοντέλο είναι ένας τύπος γλώσσας στον οποίο οι προδιαγραφές αναπαρίστανται ως μοντέλο με διαφορετικές καταστάσεις. Ορίζοντας τον τρόπο που οι λειτουργίες επιδρούν στην κατάσταση του μοντέλου μέσω μαθηματικών εννοιών όπως σχέσεις, σύνολα, συναρτήσεις, ακολουθίες, δίνονται οι λειτουργίες που το περιγράφουν. Επιπλέον τα κατηγορήματα που καθορίζονται ως προς τις προηγούμενες και τις επόμενες συνθήκες του συστήματος, περιγράφουν τις λειτουργίες του μοντέλου.
- **Γλώσσες Αλγεβρικών Προδιαγραφών :** Οι αλγεβρικές γλώσσες προδιαγραφών, προσδιορίζουν ένα σύστημα βάση των σχέσεων των λειτουργιών του και τα αξιώματα που χαρακτηρίζουν τις επιθυμητές ιδιότητες του συστήματος, περιγράφοντας όλες τις πιθανές καταστάσεις των δεδομένων και όλες τις πιθανές μεταβάσεις της κατάστασης. Χρησιμοποιεί μαθηματικές τεχνικές όπως την εξισωτική λογική για την περιγραφή του συστήματος και άλγεβρα και θεωρία κατηγοριών για την σημασιολογία.
- **Γλώσσες προσανατολισμένες στη διαδικασία :** Οι γλώσσες προσανατολισμένες στις διαδικασίες είναι βασισμένες στις διεργασίες του συστήματος που παράγονται μέσω εκφράσεων.
- **Υβριδικές Γλώσσες :** Οι υβριδικές γλώσσες εφαρμόζονται για την περιγραφή συστημάτων με ψηφιακά και αναλογικά στοιχεία. Απαιτούνται γλώσσες που περιέχουν στοιχεία τόσο από τα συνεχή μαθηματικά όσο και τα διακριτά μαθηματικά για την περιγραφή του συστήματος.

3.3 Οι Πρακτικές Των Τυπικών Μεθόδων

Οι Τυπικές Μέθοδοι παρέχουν μια σειρά πρακτικών που έχουν την δυνατότητα να εφαρμοστούν για την περιγραφή και προτυποποίηση ενός λογισμικού συστήματος. Συνδυάζοντας τις πρακτικές αυτές προσφέρεται μια πιο λεπτομερής και αναλυτική εικόνα του εν λόγω συστήματος.

3.3.1 Τυπικές προδιαγραφές

Οι τυπικές προδιαγραφές(formal specifications) ορίζουν τη συμπεριφορά και τις ιδιότητες ενός συστήματος χρησιμοποιώντας μια τυπική γλώσσα με σαφώς καθορισμένη σύνταξη και σημασιολογία [2], [3]. Οι ιδιότητες του συστήματος συμπεριλαμβάνουν την εσωτερική δομή τη λειτουργική συμπεριφορά, και τη χρονική συμπεριφορά. Οι τυπικές προδιαγραφές μπορούν επίσης να αποτυπώσουν λεπτές απαιτήσεις και περιορισμούς και να αποφύγουν την ασάφεια εφαρμόζοντας την ακριβή σύνταξη και σημασιολογία που προσφέρουν οι τυπικές γλώσσες [6].

3.3.2 Τυπική Επαλήθευση

Η τυπική επαλήθευση (formal verification), είναι μια μέθοδος που χρησιμοποιείται για να αναλύσει και να επιβεβαιώσει ότι οι ιδιότητες του μοντέλου θα έχουν την επιθυμητή συμπεριφορά, χρησιμοποιώντας αυτοματοποιημένα εργαλεία για τον έλεγχο της προδιαγραφής.

Μέσω αυτού του αναλυτικού ελέγχου μπορούν να ανιχνευθούν λάθη όπως παραβιάσεις των ιδιοτήτων ασφαλείας ή αδιέξοδα(deadlocks)[3] και να τα επιλύσουν, χρησιμοποιώντας τεχνικές όπως η απόδειξη θεωρήματος (Theorem proving) και ο έλεγχος μοντέλου (Model checking)[6].

- **Έλεγχος Μοντέλου:** Η τεχνική αυτή είναι βασισμένη στη κατασκευή ενός πεπερασμένου μοντέλου του συστήματος και στον εξαντλητικό έλεγχο της ύπαρξης επιθυμητών χαρακτηριστικών στις καταστάσεις που μπορεί να περιέλθει και τις μεταβάσεις αυτών των καταστάσεων μέσω αλγορίθμων.
- **Απόδειξη Θεωρημάτων:** Η τεχνική της απόδειξης του θεωρήματος είναι βασισμένη στην διατύπωση των επιθυμητών ιδιοτήτων του συστήματος ως μαθηματικών τύπων. Η απόδειξη αρχίζει από το σύνολο των αξιωμάτων και τους τύπους του μοντέλου και είναι βασισμένη στους συμπερασματικούς κανόνες, καταλήγοντας στις επιθυμητές ιδιότητες.

3.3.3 Τυπικός Συλλογισμός

Ο τυπικός συλλογισμός (formal reasoning), είναι μια τεχνική εύρεσης καταστάσεων ώστε να επαληθεύσουμε ότι το σύστημα ικανοποιεί κάποιες επιθυμητές ιδιότητες του μοντέλου. Μέσω αυτής της διαδικασίας μπορούν να εξαχθούν συμπεράσματα σχετικά με την ασφάλεια και τη λειτουργικότητα του συστήματος σε όλες τις πιθανές καταστάσεις [3], [6].

Οι δύο βασικές κατηγορίες ιδιοτήτων που εξετάζονται είναι:

- Η πρώτη κατηγορία είναι τα αμετάβλητα κατηγορήματα κατάστασης (invariants), τα οποία συμβολίζουν τις ιδιότητες του συστήματος που παραμένουν αναλλοίωτες σε κάθε πιθανή κατάσταση του συστήματος[12].

- Η δεύτερη κατηγορία είναι οι προσβάσιμες ιδιότητες τις καταστάσεις (reachable), οι οποίες είναι ένα αποτέλεσμα ενεργειών που συμβαίνουν στο μοντέλο και συμβολίζουν τις αλλαγές που θα γίνουν στο σύστημα σε κάποιο σημείο σε πεπερασμένο χρόνο[12]. Συνοπτικά, οι αναλλοίωτες ιδιότητες διασφαλίζουν ότι δεν θα συμβούν ανεπιθύμητες αλλαγές στο σύστημα, ενώ οι προσβάσιμες ιδιότητες διασφαλίζουν ότι θα γίνουν μόνο επιτρεπτές αλλαγές, εγγυώντας έτσι την ορθότητα του μοντέλου.

3.4 Πλεονεκτήματα Τυπικών Μεθόδων

Η ορθή καταγραφή και κατανόηση των απαιτήσεων του συστήματος που πρέπει να υλοποιηθεί, αποτελεί ένα από τα κυριότερα προβλήματα που καλούνται να αντιμετωπίσουν οι μηχανικοί λογισμικού. Η εφαρμογή απλών μη τυπικών γλωσσών όπως για παράδειγμα η αγγλική γλώσσα, δεν επιτρέπει την περιγραφή του συστήματος με τον ίδιο κατανοητό και ενδελεχή τρόπο όπως κατορθώνει μια τυπική γλώσσα. Οι Τυπικές Μέθοδοι επιτρέπουν στους μηχανικούς λογισμικού να χρησιμοποιήσουν τις τυπικές γλώσσες για να ορίσουν αυστηρά και αναλυτικά τις ιδιότητες του συστήματος και να παράγουν τυπικές προδιαγραφές. Μέσω του τυπικού συλλογισμού και του τυπικού έλεγχου επιτυγχάνεται ο εντοπισμός των λαθών που μπορεί να προκύψουν και να διασφαλίσουν ότι οι ιδιότητες και οι απαιτήσεις πληρούν τις απαιτήσεις του συστήματος πριν από το στάδιο της υλοποίησης. Σύμφωνα με μια μελέτη των Batra et al [7] που δημοσιεύθηκε τον Μάιο του 2013, οι συγγραφείς αναφέρουν τα ακόλουθα πλεονεκτήματα:

1. **Μέτρο ορθότητας:** Η χρήση Τυπικών Μεθόδων παρέχει ένα μέσο ελέγχου της ορθότητας και ποιότητας ενός συστήματος, αντιθετικά με τα υφιστάμενα μέτρα ελέγχου του συστήματος[7].
2. **Πρώιμη ανίχνευση ελαττωμάτων:** Με τη χρήση Τυπικών Μεθόδων σε πρώιμο στάδιο της διαδικασίας σχεδιασμού, τα ελαττώματα ανιχνεύονται, εντοπίζονται και διορθώνονται ταχύτερα και αμεσότερα[7].
3. **Διασφάλιση της ορθότητας:** Μέσω της τυπικής επαλήθευσης, ο ελεγκτής μοντέλων εκτελεί το σύστημα με όλους τους δυνατούς τρόπους ώστε να εντοπιστούν όλα τα πιθανά σφάλματα, παρέχοντας ολοκληρωμένα επίπεδα κάλυψης[7].
4. **Λιγότερο επιρρεπείς σε σφάλματα:** Οι Τυπικές Μέθοδοι είναι πλήρεις. Αυτό σημαίνει ότι μπορούν να καλυφθούν όλες οι πτυχές του συστήματος. Το αποτέλεσμα αυτού είναι η συμβολή στην ελαχιστοποίηση των σφαλμάτων που συμβαίνουν κατά τη διάρκεια ή μετά τη συγγραφή του πηγαίου κώδικα, καθώς οι μηχανικοί πρέπει να εξετάζουν άλλες πτυχές του συστήματος πριν από τη διαδικασία υλοποίησης στο στάδιο της περιγραφής και

καταγραφής των απαιτήσεων και των λειτουργιών των τυπικών προδιαγραφών[7].

5. **Αφαίρετικότητα :** Μια τυπική προδιαγραφή είναι μια αφηρημένη, ακριβής και μερικές φορές πλήρης περιγραφή, βοηθόντας τον αναγνώστη να κατανοήσει ευκολότερα τον γενικότερο σχεδιασμό του λογισμικού σε αντίθεση με ένα εκτεταμένο, πολύπλοκο πρόγραμμα[7].
6. **Αυστηρή ανάλυση:** Η τυποποίηση της περιγραφής του συστήματος με τη χρήση τυποποιημένων μεθόδων επιτρέπει μια πιο ακριβή ανάλυση του συστήματος, καθιστώντας ευκολότερη την κρίση της ορθότητας των απαιτήσεων υψηλού επιπέδου και των σχεδιαστικών προτάσεων[7].
7. **Αξιοπιστία:** Οι Τυπικές Μέθοδοι διαθέτουν τα αποδεικτικά στοιχεία που χρειάζονται σε κλάδους όπου απαιτούνται αυστηρές ρυθμίσεις (π.χ. διαχείριση της λειτουργίας των τρένων) και ειδικούς λόγους για την επιλογή ενός τυπικά καθορισμένου συστήματος[7].
8. **Αποτελεσματικές περιπτώσεις δοκιμών:** στο πλαίσιο μιας τυπικής προδιαγραφής, ορισμένες αποτελεσματικές περιπτώσεις δοκιμών μπορούν να προκύψουν συστηματικά απευθείας από την ίδια την προδιαγραφή, παρέχοντας έτσι ένα οικονομικά αποτελεσματικό τρόπο παραγωγής περιπτώσεων δοκιμής[7].

4 Εργαλεία και μεθοδολογίες

Όπως προαναφέρθηκε, οι Τυπικές Μέθοδοι αποτελούν ένα σύνολο από πρακτικές που εφαρμόζονται στην μηχανική λογισμικού. Σε αυτό το κεφάλαιο παρουσιάζεται η γλώσσα αλγεβρικών προδιαγραφών που χρησιμοποιήθηκε στην παρούσα διπλωματική εργασία καθώς και κατάλληλες πληροφορίες για τις τεχνικές που εφαρμόστηκαν για την εξέταση της προδιαγραφής, Στο τέλος του κεφαλαίου ο αναγνώστης θα μπορεί να κατανοήσει τις ιδιότητες και τις προδιαγραφές που παρουσιάζονται σε αυτή την εργασία.

4.1 Γλώσσα CafeOBJ

Η γλώσσα τυπικών προδιαγραφών που χρησιμοποιήθηκε είναι η CafeOBJ[8] η οποία αναπτύχθηκε στην Ιαπωνία από τον Kokichi Futatsugi και είναι μια εκτελέσιμη γλώσσα αλγεβρικών προδιαγραφών που υποστηρίζει τεχνικές απόδειξης θεωρημάτων (theorem prover) όπως τη συμπεριφοριακή προδιαγραφή και coinduction για την απόδειξη των ιδιοτήτων. Είναι βασισμένη στην εξισωτική λογική με διατεταγμένους τύπους (sorts and subsorts) συνδυασμένη με τη λογική της αναγραφής και την εξισωτική λογική με κρυμμένους τύπους (behavorial coinduction).

4.1.1 Λογικό υπόβαθρο

Λογική της αναγραφής	Χρησιμοποιείται για τη μοντελοποίηση και τον έλεγχο των συστημάτων με βάση τις εξισώσεις και τις σχέσεις μετάβασης, ορίζοντας εξισώσεις που περιγράφουν τις σχέσεις μεταξύ των στοιχείων ή καταστάσεων[9]. Επιτρέπει τη μετάβαση από μια κατάσταση σε άλλη χρησιμοποιώντας τις καθορισμένες συνθήκες και κανόνες.
Λογική με διατεταγμένους τύπους	Επιτρέπει τη διάταξη τύπων με τρόπο που ένας τύπος μπορεί να θεωρηθεί υποσύνολο ενός άλλου τύπου. Υποστηρίζει την κληρονομικότητα των τελεστών, έτσι ώστε ένας τελεστής που έχει οριστεί για έναν τύπο να μπορεί να χρησιμοποιηθεί από οποιοδήποτε υπότυπο αυτού. Για παράδειγμα ένας τελεστής που έχει οριστεί για κλασματικούς αριθμούς μπορεί να χρησιμοποιηθεί και από τους φυσικούς. Επίσης μπορούν να οριστούν τελεστές που να εφαρμόζονται μόνο σε ένα υποσύνολο του τύπου, καθώς και να διαχειριστούν εξαιρέσεις που προκύπτουν κατά την εκτέλεση των τελεστών[10].

Συμπεριφοριακή προδιαγραφή	Γενίκευση της αλγεβρικής προδιαγραφής. Τεχνική απόδειξης θεωρήματος βάση της άλγεβρας με κρυμμένους τύπους για την επαλήθευση συστημάτων. Δεν περιγράφει τον τρόπο υλοποίησης του συστήματος και των αντικειμένων αλλά την συμπεριφορά τους.
Κρυμμένοι τύποι	Λογικό υπόβαθρο των συμπεριφοριακών προδιαγραφών παρέχοντας σημασιολογική μοντελοποίηση, χρησιμοποιεί την συμπεριφοριακή ισοδυναμία για να καθορίσει την ισοδυναμία μεταξύ δύο κρυμμένων τύπων βάση της συμπεριφοράς τους, δηλαδή να συμπεριφέρονται όμοια σε όλες τις καταστάσεις που παρατηρούνται, εκτελώντας την ίδια λειτουργία[11] .
Τεχνική απόδειξης συμπεριφοριακών ιδιοτήτων - Coinduction	Η επαγωγική μέθοδος coinduction, χρησιμοποιείται για την επαλήθευση των συμπεριφοριακών προδιαγραφών, αποδεικνύοντας την συμπεριφοριακή ισοδυναμία

Πίνακας 4.1.1: Λογικό υπόβαθρο CafeOBJ

4.1.2 Συντακτικό και γραμματική

Για την κατανόηση των προδιαγραφών που θα παρουσιαστούν στην παρούσα διπλωματική εργασία είναι απαραίτητο να παρουσιαστεί το συντακτικό που χρησιμοποιείται από την γλώσσα. Μια προδιαγραφή της CafeOBJ αποτελείται από τα τεμάχια **modules** στα οποία δηλώνονται οι τύποι, οι τελεστές και οι εξισώσεις. Η γλώσσα CafeOBJ υποστηρίζει δύο είδη τύπων **(sorts)**. Ο πρώτος είναι οι ορατοί **(visible)** οι οποίοι απεικονίζουν τους αφηρημένους τύπους δεδομένων και ο δεύτερος είναι οι κρυμμένοι **(hidden)** που αναπαριστούν τις καταστάσεις. Ένας τύπος μπορεί να χρησιμοποιηθεί σαν υπότυπος ενός άλλου **[Nat < Int]** με το σύμβολο < και >, καθορίζοντας έτσι την διάταξη και μερικές συναρτήσεις. Με το σύμβολο - - δηλώνονται τα σχόλια τα οποία δεν λαμβάνονται υπόψη από τον μεταφραστή της γλώσσας. Οι τελεστές ορίζονται με τη λέξη **op** και οι τελεστές με κρυμμένους τύπους στο πεδίο ορισμού τους ορίζονται με τη λέξη **bop** (behavorial operators) και χωρίζονται σε δράσεις όταν επιστρέφουν ένα κρυμμένο τύπο και σε παρατηρήσεις όταν επιστρέφουν ένα ορατό.

Η αρχή της προδιαγραφής δηλώνεται από τη λέξη **mod** (module) και έπειτα από το όνομα της προδιαγραφής που θα δώσουμε. Στη συνέχεια ανάμεσα από **[Visible_Sort]** ορίζονται τα ονόματα των ορατών τύπων και σε ***[Hidden_Sort]*** οι κρυμμένοι τύποι. Η CafeOBJ παρέχει ενσωματωμένους (built-in) τύπους **(NAT, INT, BOOL)** οι οποίοι μπορούν να κληθούν και να χρησιμοποιηθούν στην προδιαγραφή από τον χρήστη. Επίσης υποστηρίζει την εισαγωγή τεμαχίων αλλά και την επιτρεπόμενη χρήση αυτών με τρείς τρόπους. Με την ενσωμάτωση τεμαχίων σαν

protecting δεν επιτρέπεται η επέκταση ή η ελάττωση του, με **extending** επιτρέπετε μόνο η επέκταση και με **using** είναι επιτρεπτά και τα δύο. Με τη λέξη **op operator_Name** γίνεται η δήλωση των τελεστών. Στη συνέχεια του ονόματος του τελεστή μετά από μια άνω και κάτω τελεία ορίζεται το πεδίο ορισμού (ή αλληλουχία τύπων) και με το σύμβολο → ορίζεται ένας ενιαίος τύπος επιστροφής (πχ op divInt : Int Int → Nat .) . Οι συμπεριφοριακοί τελεστές ορίζονται με τη λέξη **bop** και πρέπει να έχουν ένα κρυμμένο τύπο στο πεδίο ορισμού τους. Οι εξισώσεις καθορίζονται χρησιμοποιώντας τη λέξη **eq** και οι υπό-συνθήκη με τη λέξη **ceq** ακολουθούμενη από το τελεστή ισότητας **=** (πχ. eq όρος1 = όρος2 .) και ολοκληρώνονται με μια **.** . Η δήλωση των μεταβλητών επιτυγχάνεται μέσω της λέξης **var** ακολουθούμενη από το όνομα της, άνω και κάτω τελεία και έπειτα τον τύπο της (πχ, var S1 : Int .)

5 Υλοποίηση διαδικασίας δημιουργίας κανόνων Blackjack

Στο παρόν κεφάλαιο περιγράφεται και αναλύεται η υλοποίηση της τυπικής προδιαγραφής του υποσυνόλου του παιχνιδιού blackjack χρησιμοποιώντας τη γλώσσα αλγεβρικών προδιαγραφών CafeOBJ, με σκοπό να σχεδιαστεί μια ορθή και σαφής προδιαγραφή ώστε να εξεταστεί η συμπεριφορά του συστήματος και να αποφευχθούν κρίσιμα σχεδιαστικά λάθη. Επίσης παρουσιάζεται η ανάλυση των δομικών στοιχείων που συμπεριλήφθηκαν στη προδιαγραφή και έπειτα παρουσιάζεται η τυπική προδιαγραφή. Τέλος παρουσιάζονται τεχνικές απόδειξης, μοντελοποιώντας την προδιαγραφή σε συμπεριφοριακή μέσω της κρυφής άλγεβρας και επαληθεύονται οι ιδιότητες του συστήματος μέσα από τη συμπεριφοριακή συνεπαγωγή(coinduction).

5.1 Δομικά στοιχεία προδιαγραφής παιχνιδιού Blackjack

Για την υλοποίηση της τυπικής προδιαγραφής χρησιμοποιήθηκε ένα υποσύνολο των κανόνων και αντικειμένων μοντελοποιώντας τα διακριτά βήματα του παιχνιδιού. Η τυπική προδιαγραφή που κατασκευάστηκε υποστηρίζει τις εξής οντότητες αντικειμένων και λειτουργίες:

- **Τραπέζι (Table):** Το τραπέζι υλοποιήθηκε ως κρυμμένος τύπος δηλώνοντας και παρατηρώντας τη συμπεριφορά της κατάστασης του παιχνιδιού.

- **Κάρτες (Cards):** Ο αφηρημένος ορατός τύπος των καρτών αναπαριστά τις κάρτες της τράπουλας και υποστηρίζουν λειτουργίες για το χειρισμό και τη κατασκευή των καρτών καθορίζοντας την αξία της κάθε κάρτας.

- **Χέρι (Hand):** Ο αφηρημένος ορατός τύπος του χεριού χρησιμοποιείται για την αναπαράσταση και κατασκευή των χεριών για τους παίκτες και τον dealer, υποστηρίζοντας λειτουργίες όπως η αρχικοποίηση χεριών, η προσθήκη κάρτας σε ένα αρχικό χέρι, το άθροισμα της αξίας των χεριών, τη καταμέτρηση άσσων σε ένα χέρι, το συμβολισμό ενός άδειου χεριού. Επίσης προδιαγράφηκαν λογικοί τελεστές για την υπόδειξη των χεριών που είναι blackjack και των χεριών που έχουν υπερβεί το ανώτατο όριο και έχουν ‘καεί’.

- **Παίκτης (Player):** Ο αφηρημένος ορατός τύπος του παίκτη υλοποιήθηκε σαν υπότυπος των χεριών και υποστηρίζει λειτουργίες όπως η αρχικοποίηση των χεριών του παίκτη, τη δυνατότητα να σταματήσουν ή να προσθέσουν κάρτες στο χέρι και λογικούς τελεστές για την υπόδειξη της δυνατότητας των προηγούμενων λειτουργιών. Επίσης προδιαγράφηκαν λογικοί τελεστές για τον καθορισμό των συνθηκών νίκης, ήττας και ισοπαλίας.

- **Dealer:** Ο dealer προδιαγράφηκε με τον ίδιο τρόπο με το τύπο του παίκτη και

υποστηρίζει τις ίδιες λειτουργίες, με μόνη διαφορά να βρίσκεται στους κανόνες που ορίζουν τις λειτουργίες αυτές.

5.2 Τμήμα υπογραφής – signature part

Οι αλγεβρικές προδιαγραφές αποτελούνται από δύο μέρη, το τμήμα της υπογραφής (signature part) και το τμήμα των αξιωμάτων (axioms part). Το τμήμα υπογραφής περιγράφει τους κανόνες σύνταξης των τύπων, τα ονόματα των τελεστών, τις παραμέτρους του πεδίου ορισμού και τους επιστρεφόμενους τύπους[2]. Παρακάτω παρουσιάζεται το πρώτο τμήμα της προδιαγραφής, στο οποίο ορίζονται οι κανόνες σύνταξης των αφηρημένων τύπων μέσω των τελεστών.

Περιγραφή τελεστή	**Κώδικας CafeOBJ**
Κάρτες βαλέ, ντάμας, ρήγα	$op\ FaceCard : \rightarrow\ Card\ .$
Κάρτες δύο – δέκα	$op\ NumericCard : Int\ \rightarrow\ Card\ .$
Κάρτα άσσου	$op\ AceCard : \rightarrow\ Card\ .$
Αριθμητική αξία κάρτας	$op\ cardValue : Card\ \rightarrow\ Int\ .$
Άδειο χέρι, χωρίς κάρτες	$op\ emptyHand : \rightarrow\ Hand\ .$
Προσθήκη κάρτας στο χέρι	$op\ addCard : Card\ Hand\ \rightarrow\ Hand\ .$
Μέτρηση άσσων σε ένα χέρι	$op\ countAces : Hand\ \rightarrow\ Int\ .$
Αρχικοποίηση χεριού με δύο κάρτες	$op\ initHand : Card\ Card\ \rightarrow\ Hand\ .$
Αριθμητική αξία χεριού	$op\ handValue : Hand\ \rightarrow\ Int\ .$
Χέρι παίκτη	$op\ playerHand : Hand\ \rightarrow\ Player\ .$
Χέρι dealer	$op\ dealerHand : Hand\ \rightarrow\ Dealer\ .$
Ένα χέρι είναι blackjack	$op\ isBlackJack? : Hand\ \rightarrow\ Bool\ .$
Ένα χέρι έχει ‘καεί’	$op\ isBusted? : Hand\ \rightarrow\ Bool\ .$
Ο παίκτης δικαιούται να προσθέσει κάρτα στο χέρι	$op\ playerCanDraw? : Player\ \rightarrow\ Bool\ .$
Ο παίκτης δικαιούται να σταματήσει	$op\ playerCanStand? : Player\ \rightarrow\ Bool\ .$

Ο dealer υποχρεούται να προσθέσει κάρτα	$op\ dealerShouldDraw?:\ Dealer \rightarrow\ Bool\ .$
Ο dealer υποχρεούται να σταματήσει	$op\ dealerShouldStand?:\ Dealer \rightarrow\ Bool\ .$
Συνθήκες νικών παίκτη	$op\ PlayerWins:\ Player\ Dealer \rightarrow\ Bool\ .$
Συνθήκες νικών dealer	$op\ DealerWins:\ Player\ Dealer \rightarrow Bool\ .$
Συνθήκη ισοπαλίας	$op\ Draw:\ Player\ Dealer \rightarrow\ Bool\ .$
Ο παίκτης προσθέτει κάρτα στο χέρι	$op\ Hit:\ Card\ Player \rightarrow\ Player\ .$
Ο παίκτης σταματά	$op\ Stand:\ Player \rightarrow\ Player\ .$
Ο dealer προσθέτει κάρτα στο χέρι	$op\ dealerHit:\ Card\ Dealer \rightarrow\ Dealer\ .$
Ο dealer σταματά	$op\ dealerStand:\ Dealer \rightarrow\ Dealer\ .$

Πίνακας 5.2: Τελεστές ορατών τύπων, τμήμα υπογραφής αλγεβρικής προδιαγραφής Blackjack.

Οι τελεστές $FaceCard:\ ->\ Card\ .$ και $AceCard:\ ->\ Card\ .$ είναι σταθερές που αντιστοιχούν στις κάρτες των βαλέδων, ντάμων, ρηγάδων και των άσσων αντίστοιχα. Ο τελεστής $NumericCard:\ Int\ ->\ Card\ .$ κατασκευάζει τις νουμερικές κάρτες δύο έως δέκα, παίρνοντας σαν όρισμα ένα αριθμό Int που αντιστοιχεί στον αριθμό της κάρτας. Ο τελεστής $cardValue:\ Card\ ->\ Int\ .$ επιστρέφει την αριθμητική αξία των καρτών. Ο σταθερός τελεστής $emptyHand:\ ->\ Hand\ .$ συμβολίζει τα άδεια χέρια, δηλαδή χέρια που δεν έχουν κάρτες. Ο τελεστής $addCard:\ Card\ Hand\ ->\ Hand\ .$ παίρνει σαν όρισμα μια κάρτα και ένα χέρι και επιστρέφει το νέο χέρι προσθέτοντας μία κάρτα σε αυτό. Ο τελεστής $countAes:\ Hand\ ->\ Int\ .$ παίρνει σαν όρισμα ένα χέρι και επιστρέφει τον αριθμό των άσσων που βρίσκονται σε ένα χέρι. Ο τελεστής $initHand:\ Card\ Card\ ->\ Hand\ .$ παίρνει σαν όρισμα δύο κάρτες και κατασκευάζει το αρχικό χέρι. Ο τελεστής $handValue:\ Hand\ ->\ Int\ .$ παίρνει σαν όρισμα ένα χέρι και επιστρέφει τη συνολική αριθμητική αξία του χεριού. Χρησιμοποιώντας τον τελεστή $playerHand:\ Hand\ ->\ Player\ .$ και $dealerHand:\ Hand\ ->\ Dealer\ .$ ανατίθεται ένα αρχικό χέρι σε παίκτη και αντίστοιχα στον dealer. Ο τελεστής $isBlackJack?:\ Hand\ ->\ Bool\ .$ παίρνει σαν όρισμα ένα χέρι και επιστρέφει τη τιμή true εάν είναι blackjack. Ο τελεστής $isBusted?:\ Hand\ ->\ Bool\ .$ επιστρέφει true εάν ένα χέρι έχει υπερβεί το όριο της επιτρεπτής αριθμιτικής αξίας. Οι τελεστές $playerCanDraw?:\ Player\ ->\ Bool\ .$ και $dealerShouldDraw?:\ Dealer\ ->\ Bool\ .$

επιστρέφουν την τιμή true εάν ο παίκτης έχει το δικαίωμα να τραβήξει κάρτα και αντίστοιχα εάν ο dealer υποχρεούται να τραβήξει κάρτα. Οι τελεστές $playerCanStand? : Player -> Bool$. και $dealerShouldStand? : Dealer -> Bool$. επιστρέφουν την τιμή true εάν ο παίκτης έχει το δικαίωμα να σταματήσει και αντίστοιχα εάν ο dealer υποχρεούται να σταματήσει. Ο τελεστής $Hit : Card\ Player -> Player$. δηλώνει την απόφαση του παίκτη να τραβήξει κάρτα και κατασκευάζει το νέο χέρι αφου προστεθεί η κάρτα. Ο τελεστής $Stand : Player -> Player$. δηλώνει την απόφαση του παίκτη να σταματήσει και να διατηρήσει τις κάρτες του. Ο τελεστής $dealerHit : Card\ Dealer -> Dealer$. και $dealerStand : Dealer -> Dealer$. μοντελοποιεί την υποχρέωση του dealer να τραβήξει κάρτα και αντίστοιχα να σταματήσει. Οι τελεστές $PlayerWins\ Player\ Dealer -> Bool$. , $DealerWins\ Player\ Dealer -> Bool$., $Draw\ Player\ Dealer -> Bool$. παίρνουν σαν όρισμα το χέρι του παίκτη και του dealer, περιγράφουν τις συνθήκες νίκης για τον παίκτη και τον dealer αλλά και την ισοπαλία επιστρέφοντας την τιμή true.

5.3 Τμήμα αξιωμάτων – axioms part

Το δεύτερο μέρος μιας αλγεβρικής προδιαγραφής αποτελείται από το τμήμα αξιωμάτων. Το τμήμα των αξιωμάτων χρησιμοποιώντας τους τελεστές του τμήματος υπογραφής, παρέχει τις σημασιολογικές απαιτήσεις των τελεστών δηλώνοντας ένα σύνολο αξιωμάτων που χαρακτηρίζουν τη συμπεριφορά των τύπων, συσχετίζοντας λειτουργίες κατασκευής οντοτήτων και λειτουργίες που χρησιμοποιούνται για παρατηρήσεις[2]. Παρακάτω παρουσιάζονται οι εξισώσεις που υλοποιήθηκαν.

- Ο τελεστής cardValue χρησιμοποιήθηκε για να καθοριστεί η αριθμητική αξία των καρτών. Η αξία των βαλέδων, νταμών, ρηγάδων ορίστηκε σε δέκα, η αξία των άσσων ορίστηκε σε έντεκα και η αξία των υπολοίπων καρτών ορίστηκε ίση με τη μεταβλητή N, αφού πρώτα επαληθευθεί η συνθήκη ότι N μεγαλύτερο του ένα και μικρότερο του έντεκα.

$$(cardValue)\ -\ axioms:$$

$$ceq\ cardValue(NumericCard(N))\ =\ N$$

$$if\ ((N\ >\ 1)\ and\ (N\ <\ 11))$$

$$eq\ cardValue(FaceCard)\ =\ 10$$

$$eq\ cardValue(AceCard)\ =\ 11$$

- Ο τελεστής countAces χρησιμοποιήθηκε για την καταμέτρηση των άσσων σε ένα χέρι. Ο αριθμός των άσσων σε ένα άδειο χέρι έχει ορισθεί σε 0. Όταν προστίθεται μία κάρτα σε ένα χέρι, ελέγχεται εάν η κάρτα είναι άσσος, το σύνολο αυξάνεται κατά ένα.

$$countAces - axioms:$$

$$eq\ countAces(emptyHand) = 0$$

$$eq\ countAces(addCard(C1, H)) = (if\ (C1 = = AceCard)\ then\ (1 + countAces(H))\ else\ countAces(H)\ fi)$$

- Ο τελεστής handValue χρησιμοποιήθηκε για το άθροισμα τις αξίας των καρτών ενός χεριού. Η αριθμητική αξία ενός άδειου χεριού ορίστηκε σε μηδέν. Η handValue όποτε προσθέτετε μια κάρτα, υπολογίζει την αριθμητική αξία της κάρτας και την προσθέτει στην συνολική αξία του χεριού. Σε περίπτωση που υπάρχει κάρτα άσσου στο χέρι και η συνολική αξία υπερβαίνει το εικοσιένα, αφαιρούνται δέκα αριθμητικές μονάδες από την συνολική αξία, καθορίζοντας έτσι την αξία του άσσου σε ένα.

$$handValue - axioms:$$

$$eq\ handValue(emptyHand) = 0$$

$$ceq\ handValue(addCard(C1, H)) = (cardValue(C1) + handValue(H))$$

$$if\ (countAces(H) = 0)$$

$$eq\ handValue(addCard(C1, H)) = (if\ ((countAces(H) = 1)\ and\ ((cardValue(C1) + handValue(H)) > 21))\ then\ (cardValue(C1) + (handValue(H) - 10))\ else\ (cardValue(C1) + handValue(H))\ fi)$$

- Η εξίσωση του playerHand όπως και του dealerHand χρησιμοποιήθηκε για την ανάθεση των αρχικών χεριών(initHand) σε παίκτη και dealer, προσθέτοντας μέσω του τελεστή addCard τις δύο κάρτες σε ένα άδειο χέρι(emptyHand).

$$playerHand, dealerHand - axioms:$$

$$eq\ playerHand(initHand(C1, C2)) = addCard(C1, addCard(C2, emptyHand))$$

$$eq\ dealerHand(initHand(C1, C2)) = addCard(C1, addCard(C2, emptyHand))$$

- Οι εξισώσεις isBlackJack? και isBusted? καθορίζουν πότε ένα χέρι (παίκτη ή dealer) είναι blackjack ή αν έχει καεί ελέγχοντας τη συνολική αξία ενός χεριού.

$$isBlackJack?, isBusted? - axioms:$$

$$ceq\ isBlackJack?(H) = true\ if\ (handValue(H) = 21)$$

$$ceq\ isBusted?\ (H)\ =\ true\ if\ (handValue(H)\ >\ 21)$$

- Οι εξισώσεις playerCanDraw? και playerCanStand?, επιστρέφουν την τιμή true σε περίπτωση που ο παίκτης έχει το δικαίωμα να τραβήξει κάρτα ή να σταματήσει εφόσον η συνολική αξία του χεριού δεν έχει υπερβεί το ανώτατο όριο.

$$playerCanDraw?, playerCanStand?\ -axioms:$$

$$ceq\ playerCanDraw?\ (PH)\ =\ true\ if\ (handValue(PH)\ <\ 21)$$

$$ceq\ playerCanStand?\ (PH)\ =\ true\ if\ ((handValue(PH)\ <\ 21)\ or\ isBlackJack?\ (PH))$$

- Οι κανόνες για το υποχρεωτικό τράβηγμα κάρτας και των υποχρεωτικό σταματημό δηλώθηκαν μέσω των τελεστών dealerShouldDraw? και dealerShouldStand?, ελέγχοντας το συνολικό άθρισμα της αξίας των καρτών του χεριού τους.

$$dealerShouldDraw?\ dealerShouldStand?\ -axioms:$$

$$ceq\ dealerShouldDraw?\ (DH)\ =\ true\ if\ (handValue(DH)\ <\ 17)$$

$$ceq\ dealerShouldStand?\ (DH)\ =\ true\ if\ (handValue(DH)\ >\ 16)$$

- Οι συνθήκες νίκης για τον παίκτη ορίστηκαν μέσω του PlayerWins. Ο παίκτης κερδίζει όταν το χέρι του είναι blackjack και του dealer δεν είναι, όταν δεν έχει καεί και έχει καεί ο dealer, όταν το χέρι του δεν έχει καεί και η αξία του χεριού του είναι μεγαλύτερη από αυτήν του dealer. Οι συνθήκες νίκης για τον dealer είναι ίδιες με του παίκτη. Ο παίκτης και dealer έρχονται ισόπαλοι σε περίπτωση που έχουν και οι δύο blackjack, η σε περίπτωση που και οι δύο δεν έχουν καεί και η συνολική αξία του χεριού τους είναι ίδια.

$$PlayerWins, DealerWins, Draw\ -\ axioms:$$

$$ceq\ PlayerWins(PH, DH)\ =\ true$$

$$if\ (((handValue(PH)\ <\ 22)\ and\ ((handValue(PH)\ >\ handValue(DH))\ or\ isBusted?\ (DH)))\ or\ (isBlackJack?\ (PH)\ and\ (isBlackJack?\ (DH)\ =\ false)))$$

$$ceq\ DealerWins(PH, DH)\ =\ true\ if\ (((handValue(DH)\ <\ 22)\ and\ ((handValue(DH)\ >\ handValue(PH))\ or\ isBusted?\ (PH)))\ or\ (isBlackJack?\ (DH)\ and\ (isBlackJack?\ (PH)\ =\ false)))$$

$$ceq\ Draw(PH, DH)\ =\ true\ if\ ((handValue(PH)\ =\ handValue(DH))\ or\ (isBlackJack?\ (PH)\ and\ isBlackJack?\ (DH)))$$

- Ο παίκτης έχει το δικαίωμα να ταβήξει μία κάρτα και να προσθέσει την αξία της στο χέρι, χρησιμοποιώντας τον τελεστές Hit εφόσον ικανοποιείτε η συνθήκη playerCanDraw?. Αντίστοιχα μπορεί να σταματήσει χρησιμοποιώντας τον τελεστή Stand εφόσον ικανοποιείται η συνθήκη playerCanStand?, ελέγχοντας αν η συνολική αξία του χεριού είναι κάτω από είκοσιένα.

$$Hit, Stand\ -\ axioms:$$

$$eq\ Hit(C1, PH)\ =\ (if\ playerCanDraw?\ (PH)\ then\ addCard(C1, PH)\ else\ PH\ fi)$$

$$ceq\ Stand(PH)\ =\ PH\ if\ playerCanStand?\ (PH)$$

- Αντίστοιχα ο dealer υποχρεούται να τραβήξει μέσω του τελεστή dealerHit εφόσον ικανοποιείται η συνθήκη dealerShouldDraw? και να σταματήσει μέσω του τελεστή dealerHit εφόσον ικανοποιείται η dealerShouldStand?

$$dealerStand, dealerHit\ -\ axioms:$$

$$ceq\ dealerHit(C1, DH)\ =\ addCard(C1, DH)\ if\ dealerShouldDraw?\ (DH)$$

$$ceq\ dealerStand(DH)\ =\ DH\ if\ dealerShouldStand?\ (DH)$$

5.4 Αλγεβρική προδιαγραφή παιχνιδιού Blackjack, κώδικας CafeOBJ

Το τεμάχιο της προδιαγραφής ονομάστηκε BLACKJACK. Ο κρυμμένος τύπος Table συμβολίζει το χώρο καταστάσεων του συστήματος. Οι ορατοί τύποι Card και Hand, συμβολίζουν τους αφηρημένους τύπους των καρτών και χεριών. Οι τύποι Player και Dealer συμβολίζουν τον παίκτη και τον dealer και είναι υπότυποι του τύπου των χεριών. Στο τεμάχιο ενσωματώθηκαν οι ήδη κατασκευασμένοι τύποι από το σύστημα της CafeOBJ, Int και Bool και δηλώθηκαν μεταβλητές.

$$mod\ BLACKJACK\ \{$$
$$*\ [Table]\ *$$
$$[Player\ Dealer\ <\ Hand]$$
$$[Card]$$
$$pr(INT)$$
$$pr(BOOL)$$

$$op\ FaceCard : \rightarrow\ Card\ .$$

$$op\ NumericCard : Int \rightarrow Card .$$
$$op\ AceCard : \rightarrow Card .$$
$$op\ cardValue : Card \rightarrow Int .$$
$$op\ emptyHand : \rightarrow Hand .$$
$$op\ addCard : Card\ Hand \rightarrow Hand .$$
$$op\ countAces : Hand \rightarrow Int .$$
$$op\ initHand : Card\ Card \rightarrow Hand .$$
$$op\ handValue : Hand \rightarrow Int .$$
$$op\ playerHand : Hand \rightarrow Player .$$
$$op\ dealerHand : Hand \rightarrow Dealer .$$
$$op\ isBlackJack? : Hand \rightarrow Bool .$$
$$op\ isBusted? : Hand \rightarrow Bool .$$
$$op\ playerCanDraw? : Player \rightarrow Bool .$$
$$op\ playerCanStand? : Player \rightarrow Bool .$$
$$op\ dealerShouldDraw? : Dealer \rightarrow Bool .$$
$$op\ dealerShouldStand? : Dealer \rightarrow Bool .$$
$$op\ PlayerWins : Player\ Dealer \rightarrow Bool .$$
$$op\ DealerWins : Player\ Dealer \rightarrow Bool .$$
$$op\ Draw : Player\ Dealer \rightarrow Bool .$$
$$op\ Hit : Card\ Player \rightarrow Player .$$
$$op\ Stand : Player \rightarrow Player .$$
$$op\ dealerHit : Card\ Dealer \rightarrow Dealer .$$
$$op\ dealerStand : Dealer \rightarrow Dealer .$$

$$var\ N : Int .$$
$$vars\ C1\ C2 : Card .$$
$$var\ H : Hand .$$
$$var\ PH : Player .$$
$$var\ DH : Dealer .$$
$$var\ T : Table .$$

$$ceq\ cardValue(NumericCard(N)) = N\ if\ N > 1\ and\ N < 11 .$$
$$eq\ cardValue(FaceCard) = 10 .$$
$$eq\ cardValue(AceCard) = 11 .$$
$$eq\ countAces(emptyHand) = 0 .$$
$$eq\ countAces(addCard(C1, H)) = if\ C1 == AceCard$$
$$then\ 1 + countAces(H)$$

$$else\ countAces(H)\ fi\ .$$
$$eq\ playerHand(initHand(C1, C2))\ =\ addCard(C1, addCard(C2, emptyHand))\ .$$
$$eq\ dealerHand(initHand(C1, C2))\ =\ addCard(C1, addCard(C2, emptyHand))\ .$$
$$eq\ handValue(emptyHand)\ =\ 0\ .$$
$$ceq\ handValue(addCard(C1, H))$$
$$=\ cardValue(C1)\ +\ handValue(H)\ if\ countAces(H)\ =\ 0\ .$$
$$eq\ handValue(addCard(C1, H))\ =\ if\ countAces(H)$$
$$=\ 1\ and\ cardValue(C1)\ +\ handValue(H)\ >\ 21$$
$$then\ cardValue(C1)\ +\ handValue(H)\ -\ 10$$
$$else\ cardValue(C1)\ +\ handValue(H)\ fi\ .$$
$$ceq\ isBlackJack?\ (H)\ =\ true\ if\ handValue(H)\ =\ 21\ .$$
$$ceq\ isBusted?\ (H)\ =\ true\ if\ handValue(H)\ >\ 21\ .$$
$$ceq\ playerCanDraw?\ (PH)\ =\ true\ if\ handValue(PH)\ <\ 21\ .$$
$$ceq\ playerCanStand?\ (PH)\ =\ true\ if\ handValue(PH)$$
$$<\ 21\ or\ isBlackJack?\ (PH)\ .$$
$$ceq\ dealerShouldDraw?\ (DH)\ =\ true\ if\ handValue(DH)\ <\ 17\ .$$
$$ceq\ dealerShouldStand?\ (DH)\ =\ true\ if\ handValue(DH)\ >\ 16\ .$$
$$ceq\ PlayerWins(PH, DH)\ =\ true\ if\ (handValue(PH)\ <\ 22\ and\ (handValue(PH)$$
$$>\ handValue(DH)\ or\ isBusted?\ (DH)))$$
$$or\ (isBlackJack?\ (PH)\ and\ isBlackJack?\ (DH)\ =\ false)\ .$$
$$ceq\ DealerWins(PH, DH)\ =\ true\ if\ handValue(DH)\ <\ 22\ and\ (handValue(DH)$$
$$>\ handValue(PH)\ or\ isBusted?\ (PH))$$
$$or\ (isBlackJack?\ (DH)\ and\ isBlackJack?\ (PH)\ =\ false)\ .$$
$$ceq\ Draw(PH, DH)\ =\ true\ if\ handValue(PH)\ =\ handValue(DH)$$
$$or\ isBlackJack?\ (PH)\ and\ isBlackJack?\ (DH)\ .$$

$$eq\ Hit(C1, PH)\ =\ if\ playerCanDraw?\ (PH)$$
$$then\ addCard(C1, PH)$$
$$else\ PH\ fi\ .$$
$$ceq\ Stand(PH)\ =\ PH\ if\ playerCanStand?\ (PH)\ .$$
$$ceq\ dealerHit(C1, DH)\ =\ addCard(C1, DH)\ if\ dealerShouldDraw?\ (DH)\ .$$
$$ceq\ dealerStand(DH)\ =\ DH\ if\ dealerShouldStand?\ (DH)\ .$$
}

Πίνακας 5.4: Κώδικας CafeOBJ, αλγεβρική προδιαγραφή Blackjack.

5.5 Απόδειξη συμπεριφοριακών ιδιοτήτων - τεχνική coinduction

Για την απόδειξη συμπεριφοριακών ιδιοτήτων χρησιμοποιήθηκε η επαγωγική μέθοδος (coinduction). Δύο στοιχεία είναι συμπεριφοριακά ισοδύναμα εφόσον αποδειχθεί ότι είναι

όμοια(congruent), δηλαδή εξισώσεις τύπου $(\forall X)T = T'$. Πρώτα ορίζεται μια σχέση ισοδυναμίας κρυμμένων τύπων T=*=T', έπειτα αποδεικνύεται η ομοιότητα(congruence) και στο τέλος αποδεικνύεται ότι T =*= T'. Για την απόδειξη της συμπεριφοριακής ισοδυναμίας προδιαγράφηκαν οι παρακάτω συμπεριφοριακοί τελεστές. Οι τελεστές που επιστρέφουν ένα ορατό τύπο αντιστοιχούν στις παρατηρήσεις (χαρακτηριστικά) και οι τελεστές που επιστρέφουν ένα κρυμμένο στις δράσεις(μεθόδους). Παρακάτω παρουσιάζετε η συμπεριφοριακή μοντελοποίηση για την απόδειξη της συμπεριφοριακής ισοδυναμίας της ιδιότητας για την αριθμητική αξία ενός αρχικού χεριού(initH).

$-- \ Actions$

$bop\ table : \ Hand\ Hand \ \rightarrow \ Table\ .$ $\quad -- construct\ Table\ action$

$bop\ initH : \ Card\ Card \ \rightarrow \ Table\ .$ $\quad -- init\ Hand\ action$

$bop\ addC : \ Card \ \rightarrow \ Table\ .$ $\quad -- add\ Card\ action$

$bop\ tableVals : \ Int\ Int \rightarrow \ Table\ .$ $\quad --Helper\ action\ for\ coinduction\ T =*= T'$

$-- \ Observations$

$bop\ cardV : \ Table \ \rightarrow \ Int\ .$ $\quad -- cardValue\ observation$

$bop\ handV : \ Table \ \rightarrow \ Int\ .$ $\quad -- handValue\ observation$

$op\ init : \rightarrow Table\ .$ $\quad -- init\ constant\ of\ table$

$op\ emptyH : \rightarrow \ Table\ .$ $\quad -- cosntant\ emptyHand$

$-- \ Axioms$

$eq\ init \ = \ table(emptyHand, emptyHand)\ .$ $\quad -- inital\ state\ of\ table$

$eq\ cardV(addC(FaceCard)) \ = \ 10\ .$

$eq\ cardV(addC(AceCard)) \ = \ 11\ .$

$eq\ handV(initH(C1, C2)) = \ cardV(addC(C1)) \ + \ cardV(addC(C2)) \ + \ handV(emptyH)\ .$

$eq\ handV(emptyH) = \ 0\ .$

Ο χώρος καταστάσεων του παιχνιδιού περιγράφετε από τον κρυμμένο τύπο Table. Ο τελεστής δράσης table παίρνει ως όρισμα δύο χέρια και συμβολίζει το τραπέζι. Η σταθερά init συμβολίζει την αρχική κατάσταση, στην οποία το τραπέζι έχει δύο άδεια χέρια, του παίκτη και του dealer και περιγράφετε μέσω της εξίσωσης $eq\ init \ = \ table(emptyHand, emptyHand)$ . . Οι τελεστές

δράσης addC και initH αλλάζουν την κατάσταση του τραπεζιού, προσθέτοντας κάρτες και κατασκευάζοντας τα αρχικά χέρια, ενώ οι παρατηρήσεις cardV και handV επιστρέφουν την αξία μιας κάρτας και την αξία του αρχικού χεριού αντίστοιχα. Ο τελεστής emptyH είναι μία σταθερά που συμβολίζει ένα άδειο χέρι. Η αξία ενός άδειου χεριού ορίστηκε σε μηδέν μέσω της εξίσωσης $eq\ handV(emptyH)\ =\ 0$. Οι εξισώσεις για την αξία των καρτών ορίστηκαν με τις εξισώσεις, $eq\ cardV(addC(FaceCard))\ =\ 10$. και $eq\ cardV(addC(AceCard))\ =\ 11$. Οι εξισώσεις για την συνολική αξία των καρτών ενός αρχικού χεριού προδιαγράφηκαν ως $eq\ handV(initH(C1, C2))\ =\ cardV(addC(C1))\ +\ cardV(addC(C2))\ +handV(emptyH)$. Το κατηγόρημα της υπόθεσης δύο κρυμμένων τύπων μέσω του δυαδικού τελεστή T=*=T' υποδηλώνει ότι δύο εξισώσεις είναι όμοιες(congruent), εάν method(x) == method(x') για οποιαδήποτε μέθοδο κρυμμένου τύπου και για οποιοδήποτε x, τότε σύστημα αυτόματα δηλώνει ένα αξίωμα . Το αξίωμα που δηλώθηκε για την προδιαγραφή μας είναι το εξής:

$**\ system\ already\ proved\ "\ =*=\ "\ is\ a\ congruence\ of\ BLACKJACK$

$ceq\ (hs1{:}Table\ =*=\ hs2{:}Table)\ =\ true$

$if\ ((handV(hs1)\ ==\ handV(hs2))\ and\ (cardV(hs1)\ ==\ cardV(hs2)))$

Μέσω του τελεστή tableVal και της εντολής reduce της CafeOBJ η εξίσωση
$reduce\ tableVals(handV(initH(C1, C2)), handV(initH(C1, C2)))\ =*=$
$tableVals(handV(initH(C2, C1)), handV(initH(C2, C1)))$. επέστρεψε τη λογική τιμή true αποδεικνύοντας την ομοιότητα και τη συμπεριφοριακή ισοδυναμία.

Έξοδος συστήματος:

$[15(cond)]{:}\ (true\ and\ true){:}\ Bool$

$-->\ (true){:}\ Bool$

$[16]{:}\ (tableVals((cardV(addC(C2))\ +\ cardV(addC(C1))), (cardV(addC(C2))\ +\ cardV(addC(C1))))\ =*\ =\ tableVals((cardV(addC(C1))\ +\ cardV(addC(C2))), (cardV(addC(C1))\ +\ cardV(addC(C2)))){:}\ Bool$

$--->\ (true){:}\ Bool$

$(true){:}\ Bool$

ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα διπλωματική εργασία παρουσιάστηκε η εφαρμογή των Τυπικών Μεθόδων και συγκεκριμένα των αλγεβρικών προδιαγραφών, χρησιμοποιώντας ένα υποσύνολο των κανόνων του παιχνιδιού blackjack ως περιπτωσιολογική μελέτη. Δημιουργήθηκε μια ακριβής και ορθή προδιαγραφή εκφράζοντας τα χαρακτηριστικά του συστήματος βάση των λειτουργιών και των σχέσεων τους, χρησιμοποιώντας τη γλώσσα αλγεβρικών προδιαγραφών CafeOBJ. Μέσω τεχνικών επαλήθευσης και μεθόδων απόδειξης θεωρημάτων, χρησιμοποιώντας την άλγεβρα κρυφών τύπων και συγκεκριμένα της επαγωγικής μεθόδου συμπεριφοριακής ισοδυναμίας, εξετάστηκε η συμπεριφορά του συστήματος, επιβεβαιώθηκε η ορθή λειτουργία και αποφευχθήκαν σχεδιαστικά λάθη νωρίς στη διαδικασία ανάπτυξης. Από την έρευνα αποκτήθηκε γνώση για τη εφαρμογή των Τυπικών Μεθόδων και για τη συνεισφορά τους στον σχεδιασμό για ασφαλή και αξιόπιστα λογισμικά συστήματα.

Βιβλιογραφία

[1]Peter A. Griffin. The Theory of Blackjack. Huntington Press, Las Vegas, NV, 1988.

[2] 'Ch_27_Formal_spec.pdf'. Ημερομηνία πρόσβασης: 3 Απρίλιος 2023. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: https://ifs.host.cs.st-andrews.ac.uk/Books/SE9/WebChapters/PDF/Ch_27_Formal_spec. pdf

[3] J.-R. Abrial, 'Formal Methods: Theory Becoming Practice', *Formal Methods*. [4]J. M. Wing, 'A specifier's introduction to formal methods', *Computer*, τ. 23, τχ. 9, σσ. 8–22, Σεπτεμβρίου 1990, doi: 10.1109/2.58215.

[5] M. V. Ruhela, 'Z Formal Specification Language - An Overview', *International Journal of Engineering Research*, τ. 1, τχ. 6, 2012.

[6]'Formal methods'. https://dl.acm.org/doi/epdf/10.1145/242223.242257 (ημερομηνία πρόσβασης 6 Απρίλιος 2023).

[7]M. Batra, A. Malik, και D. M. Dave, 'FORMAL METHODS: BENEFITS, CHALLENGES AND FUTURE DIRECTION', 2010.

[8]R. Diaconescu, K. Futatsugi, CafeOBJ Report - The Language, Proof Techniques, and Methodologies for Object-Oriented Algebraic Specification, AMAST Series in Computing, vol.6, World Scientific, 1998.

[9] J. Meseguer, Conditional rewriting logic: Deduction, models and concurrency, in: Proc. 2nd International CTRS Workshop, LNCS 516, pp. 64-91, 1991.

[10] J. Goguen, J. Meseguer, Order-sorted algebra i: Equational deduction for multiple inheritance, polymorphism, overloading and partial operations, Technical Report SRI-CSL89-10, SRI International, 1989.

[11] J. Goguen, G. Malcolm, A hidden agenda, Technical Report CS97-538, University of California at San Diego, 1997.

[12] Kazuhiro Ogata, Kokichi Futatsugi, Simulation-based Verification for Invariant Properties in the OTS/CafeOBJ Method, School of Information Science Japan Advanced Institute of Science and Technolog 1-1 Asahidai, Nomi, Ishikawa 923-1290