

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΣΧΕΔΙΟ ΠΑΡΟΥΣΙΑΣΗΣ
ΑΤΟΜΙΚΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Αύγουστος 2021

Ατομική Διπλωματική Εργασία

BIO - CANCER LOGIC

Μάριος Γεωργίου

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Αύγουστος 2021

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Bio – Cancer Logic

Μάριος Γεωργίου

Επιβλέπων Καθηγητής

Αντώνης Κάκας

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των
απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του
Πανεπιστημίου Κύπρου

Αύγουστος 2021

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον καθηγητή του Τμήματος Πληροφορικής κύριο Αντώνη Κάκα που μου έδωσε την ευκαιρία να εμπλουτίσω τις γνώσεις μου σε καινούριους τομείς, όπως αυτόν της Βιοπληροφορικής. Επίσης, θα ήθελα να ευχαριστήσω την κυρία Χριστιάνα Νεοφύτου για την προσοχή και την βοήθεια στην κατανόηση των βιολογικού κομματιού της εργασίας. Ακόμη, θα ήθελα να ευχαριστήσω τον κύριο Αδάμο Κουμή για την πολύτιμη βοήθεια που μου έδωσε καθ' όλη τη διάρκεια της ανάπτυξης της εργασίας. Ο χρόνος και η προσοχή που αφιέρωσαν αποτέλεσαν πολύτιμη βοήθεια για την ολοκλήρωση της εργασίας. Τέλος, θα ήθελα να ευχαριστήσω την οικογένεια και τους φίλους μου για την υποστήριξη τους κατά τη διάρκεια των σπουδών μου.

Περίληψη

Η παρούσα διπλωματική εργασία έχει ως σκοπό την μοντελοποίηση των αποπτωτικών οδών σηματοδότησης και την δημιουργία ενός συστήματος το οποίο θα βοηθήσει στην προσομοίωση εργαστηριακών πειραμάτων και στη επαλήθευση των συμπερασμάτων που έχουμε από αυτά. Βασικός μας στόχος είναι οι ερευνητές του τμήματος της Βιολογίας να μπορούν με ευκολία και επιτυχία να διεξάγουν πειράματα *in silico*, δηλαδή με την βοήθεια του ηλεκτρονικού υπολογιστή, παρά με τον παραδοσιακό τρόπο στα εργαστήρια τους (*in vitro*).

Το σύστημα χρησιμοποιεί διάφορες τεχνολογίες διαδικτύου όπως HTML, CSS, PHP και JavaScript με σκοπό να δημιουργήσει ένα φιλικό προς τον χρήστη περιβάλλον που να είναι ευχάριστο και ενδιαφέρον.

Περιεχόμενα

Κεφάλαιο 1	Εισαγωγή.....	1
	1.1 Εισαγωγή στην Βιοπληροφορική	1
	1.2 Το σύστημα Bio_Cancer_Logic	2
	1.3 Μεθοδολογία	2
	1.4 Δομή	3
Κεφάλαιο 2	Απόπτωση και αποπτωτικές οδοί σηματοδότησης.....	5
	2.1 Εισαγωγή	5
	2.2 Το φαινόμενο της απόπτωσης	5
	2.3 Αποπτωτικές οδοί σηματοδότησης	7
	2.4 Διεξαγωγή ενός πειράματος στην Βιολογία	8
Κεφάλαιο 3	Λογική και Λογικός Προγραμματισμός.....	10
	3.1 Εισαγωγή	10
	3.2 Λογική	10
	3.3 Λογικός Προγραμματισμός – Prolog	12
Κεφάλαιο 4	Αναπαράσταση των Pathways και μετάδοση σήματος σε λογικό προγραμματισμό (Συνέπεια Υποθέσεων).....	18
	4.1 Εισαγωγή	18
	4.2 Αναπαράσταση γράφων και δικτύων	18
	4.3 Μετάδοση σήματος	19
Κεφάλαιο 5	Το σύστημα Bio_Cancer_Logic.....	24
	5.1 Εισαγωγή	24
	5.2 Υλοποίηση του συστήματος	24
	5.2.1 Home	25
	5.2.2 Simulation	26

5.2.3	How to Use	32
5.2.4	About	33
5.3	Αρχιτεκτονική Συστήματος	33
5.3.1	Εκτέλεση Προσομοίωσης Πειράματος	33
5.3.2	Εκτέλεση Προσομοίωσης Πειράματος με Υποθέσεις	35
Κεφάλαιο 6	Τελική Αξιολόγηση Συστήματος	37
6.1	Εισαγωγή	37
6.2	Αξιολόγηση βάση Πειραματικών Δεδομένων	37
Κεφάλαιο 7	Συμπεράσματα	40
7.1	Εισαγωγή	40
7.2	Συμπεράσματα	40
7.3	Μελλοντικές Αναβαθμίσεις	41
Βιβλιογραφία		42
Παράρτημα Α		Α-1
Παράρτημα Β		Β-1
Παράρτημα Γ		Γ-1
Παράρτημα Δ		Δ-1

Κεφάλαιο 1

Εισαγωγή

1.1 Εισαγωγή στην Βιοπληροφορική	1
1.2 Το σύστημα Bio_Cancer_Logic	2
1.3 Μεθοδολογία	2
1.4 Δομή	3

1.1 Εισαγωγή στην Βιοπληροφορική

Η πληροφορική στην πάροδο του χρόνου απέκτησε εφαρμογές σε διάφορους επιστημονικούς και μη κλάδους όπως τα μαθηματικά, η φυσική και τα χρηματοοικονομικά. Ένας από αυτούς τους κλάδους είναι και αυτός της Βιολογίας. Έτσι μπορούμε να ορίσουμε την Βιοπληροφορική ως ο επιστημονικός κλάδος ο οποίος προέκυψε από την συνεργασία των επιστημών της Πληροφορικής και της Βιολογίας. Η σύνδεση αυτών των δυο επιστημονικών κλάδων έχει καθαρά ερευνητικό χαρακτήρα. Αν και η επιστήμη της Βιοπληροφορικής ξεκίνησε με την μετατροπή της γενετικής πληροφορίας (DNA και RNA) σε ψηφιακή πληροφορία (0 και 1), πλέον έχει αποκτήσει τομείς όπως η προσομοίωση βιολογικών σημάτων.

Απεικόνιση βιολογικών σημάτων είναι η μοντελοποίηση φυσικών σημάτων ενός ανθρώπου με τη χρήση ειδικού λογισμικού, όπως η μοντελοποίηση των αποπτωτικών οδών σηματοδότησης. Αυτός ο τομέας της Βιοπληροφορικής είναι πολύ σημαντικός όχι μόνο για την έρευνα αλλά και για τη διάγνωση διάφορων παθήσεων. Απλό παράδειγμα απεικόνισης βιοσήματος είναι η τρισδιάστατη απεικόνιση της καρδιάς, κάτι το οποίο

έχει βοηθήσει στην διάγνωση παθήσεων τις οποίες οι γιατροί έπρεπε να “ακούσουν” με το στηθοσκόπιο έχοντας έτσι πολύ μεγάλα περιθώρια λάθους.

Θεωρώντας τα βιολογικά δεδομένα (DNA, RNA, πρωτεΐνες) ως ψηφιακή πληροφορία, εφαρμόζει αλγορίθμους για την επεξεργασία τους και την παραγωγή χρήσιμων συμπερασμάτων με αποδοτικό τρόπο.

1.2 Το σύστημα Bio_Cancer_Logic

Βασικός στόχος του συστήματος είναι να συνδυάσουμε τους 2 κλάδους της Πληροφορικής και της Βιολογίας. Το σύστημα που υλοποιήσαμε στα πλαίσια της παρούσας διπλωματικής έχει ως σκοπό την μοντελοποίηση του ενδοκυτταρικού περιβάλλοντος ούτως ώστε να βοηθήσουμε στην προσομοίωση εργαστηριακών πειραμάτων. Στόχος μας είναι οι ερευνητές του τμήματος της Βιολογίας να μπορούν με ευκολία και επιτυχία να διεξάγουν πειράματα *in silico*, δηλαδή με την βοήθεια του ηλεκτρονικού υπολογιστή, παρά με τον παραδοσιακό τρόπο στα εργαστήρια τους (*in vitro*).

Αυτό θα επιτευχθεί με την βοήθεια του ηλεκτρονικού υπολογιστή ο οποίος θα παίρνει τα βιολογικά δεδομένα και θα τα επεξεργάζεται, βγάζοντας αποτελέσματα σε μερικά δευτερόλεπτα, τα οποία υπό κανονικές συνθήκες θα χρειάζονταν πολλές ώρες και κόπο από ένα ερευνητή Βιολογίας.

1.3 Μεθοδολογία

Η μεθοδολογία που ακολουθήθηκε κατά την διάρκεια της διπλωματικής εργασίας αποτελείται από τα εξής βήματα:

Α. Έρευνα και κατανόηση διαφόρων όρων και εννοιών της Βιολογίας, και κατανόηση του προβλήματος. Σε αυτό το στάδιο έγινε μελέτη επιστημονικών άρθρων για την κατανόηση της λειτουργίας των αποπτωτικών οδών σηματοδότησης και άλλων όρων Βιολογίας.

Β. Μελέτη διαφόρων τρόπων υλοποίησης του δικτύου των αποπρωτικών οδών σηματοδότησης. Σε αυτό το στάδιο έγινε έρευνα για κάποια βιβλιοθήκη ή κάποιο άλλο εργαλείο το οποίο θα βοηθούσε σχετικά με τον τρόπο που θα μπορούσε να υλοποιηθεί το δίκτυο των αποπρωτικών οδών σηματοδότησης

Γ. Απόφαση σχετικά με το ποια εργαλεία θα χρησιμοποιηθούν. Σε αυτό το στάδιο πάρθηκαν οι αποφάσεις σχετικά με τα εργαλεία και τον τρόπο που θα χρησιμοποιηθούν.

Δ. Υλοποίηση συστήματος Σε αυτό το στάδιο γίνεται η υλοποίηση του συστήματος.

Ε. Αξιολόγηση τελικού συστήματος Σε αυτό το στάδιο γίνεται η αξιολόγηση του τελικού μας συστήματος. Η αξιολόγηση θα γίνει με 2 τρόπους : συγκρίνοντας πειραματικά δεδομένα και μέσω ενός online ερωτηματολογίου

1.4 Δομή

Η εργασία αυτή αποτελείται από επτά κεφάλαια. Το Κεφάλαιο 1 αποτελεί την εισαγωγή, όπου αρχικά βλέπουμε την έννοια της Βιοπληροφορικής και στη συνέχεια περιγράφεται το κίνητρο της εργασίας, καθώς και ο στόχος και η μεθοδολογία που ακολουθήθηκε γι' αυτήν.

Στο Κεφάλαιο 2 παρουσιάζεται η απαιτούμενη γνώση που χρειάζεται για την βέλτιστη κατανόηση της εργασίας, η έρευνα που έγινε σχετικά με την απόπτωση και τις αποπρωτικές οδούς σηματοδότησης., καθώς και περιγράφουμε πολύ συνοπτικά την διαδικασία που ακολουθούν οι ερευνητές της Βιολογίας για να διεξάγουν ένα πείραμα στα εργαστήρια τους.

Στο Κεφάλαιο 3 εξηγούμε λίγο την έννοια της λογικής και του λογικού προγραμματισμού και μελετούμε τις βασικές αρχές της prolog.

Στο Κεφάλαιο 4, βρίσκουμε πως μπορούμε να αναπαραστήσουμε δίκτυα και γράφους χρησιμοποιώντας την prolog , ενώ μετά επεκτείνουμε το ήδη υπάρχον αρχείο prolog έτσι ώστε να εκφράσουμε και τις σχέσεις ενεργοποίησης και απενεργοποίησης μεταξύ

των πρωτεϊνών στο δίκτυο και να δούμε πως λειτουργεί η μετάδοση του σήματος μέσα στο δίκτυο.

Στο κεφάλαιο 5, εξηγούμε το πως υλοποιήσαμε το σύστημα μας, βλέπουμε τι δυνατότητες μας παρέχει καθώς και πως μπορούμε να το χρησιμοποιήσουμε για να διεξάγουμε τα πειράματα που θέλουμε. Στο τέλος του κεφαλαίου 5, εξηγούμε συνοπτικά την αρχιτεκτονική του συστήματος και το πως λειτουργεί.

Συνεχίζοντας στο κεφάλαιο 6, βρίσκεται η τελική αξιολόγηση του τελικού συστήματος η οποία γίνεται μέσω σύγκρισης των δικών μας αποτελεσμάτων με τα πειραματικά δεδομένα που μας έδωσαν οι ερευνητές και προκύπτουν από τα πειράματα που έγιναν στα εργαστήρια

Στο τελευταίο κεφάλαιο, το Κεφάλαιο 7 ,καταγράφονται τα τελικά συμπεράσματα αυτής της εργασίας καθώς και κάποιες μελλοντικές αναβαθμίσεις που μπορούν να γίνουν στο σύστημά μας. Στο τέλος, ακολουθούν η Βιβλιογραφία και τα Παραρτήματα, τα οποία είναι ο σημαντικός κώδικας του συστήματος.

Κεφάλαιο 2

Απόπτωση και αποπτωτικές οδοί σηματοδότησης

2.1 Εισαγωγή	5
2.2 Το φαινόμενο της απόπτωσης	5
2.3 Αποπτωτικές οδοί σηματοδότησης	7
2.4 Διεξαγωγή ενός πειράματος στην Βιολογία	8

2.1 Εισαγωγή

Στην παρούσα διπλωματική εργασία είχαμε την ευκαιρία να συνεργαστούμε με το τμήμα Βιολογίας του Πανεπιστημίου μας και σε αυτό το κεφάλαιο θα εξηγήσουμε τι είναι το φαινόμενο της απόπτωσης και ποιες είναι οι αποπτωτικές οδοί σηματοδότησης, καθώς και την διαδικασία διεξαγωγής ενός πειράματος στα εργαστήρια της Βιολογίας.

2.2 Το φαινόμενο της απόπτωσης

Η λέξη απόπτωση, είναι μια από τις ευρύτερα χρησιμοποιημένες έννοιες στην Βιολογία και είναι λέξη ελληνικής ρίζας η οποία προέρχεται από τα 2 συνθετικά από και πτώση. Χρησιμοποιώντας το όρο απόπτωση, αναφερόμαστε στην διαδικασία, την οποία ακολουθούν όλα τα διάφορα οργανικά συστήματα του ανθρώπινου σώματος, κατά την οποία επιτυγχάνεται ο προγραμματισμένος θάνατος στα κύτταρα. Η απόπτωση προκαλείται από πολλούς εξωκυτταρικούς και ενδοκυτταρικούς παράγοντες. Κάποιοι από αυτούς τους παράγοντες είναι οι ιοί, η ακτινοβολία και η αδυναμία επιδιόρθωσης του DNA.

Πολύ συχνά η διαδικασία της απόπτωσης μπορεί να μπερδευτεί με μια άλλη διαδικασία την νέκρωση. Η νέκρωση είναι μια μορφή κυτταρικού τραυματισμού που ορίζεται ως ο μη ρυθμιζόμενος κυτταρικός θάνατος που προκύπτει από εσωτερικούς ή εξωτερικούς παράγοντες όπως τραυματισμοί, χημικοί παράγοντες ή παθογόνα. Η διαδικασία είναι συνήθως γρήγορη και οδηγεί σε διόγκωση των κυττάρων και έκρηξη του κυττάρου λόγω απώλειας ωσμωτικής πίεσης .

Αν και το γεγονός ότι τα κύτταρα πεθαίνουν κατά τη φυσιολογική ανάπτυξη αναγνωρίστηκε από τον Carl Vogt το 1842, περίπου πριν από 170 χρόνια δηλαδή, η απόπτωση είναι ίσως η σημαντικότερη διαδικασία του ανθρώπινου σώματος καθώς ο κυτταρικός θάνατος είναι απαραίτητος για τη ρύθμιση του κυτταρικού αριθμού και την πρόληψη της διάδοσης κατεστραμμένων ή επικίνδυνων κυττάρων. Αξιοσημείωτο είναι το γεγονός ότι ο μέσος ενήλικος άνθρωπος χάνει μεταξύ 50 και 70 δισεκατομμυρίων κυττάρων κάθε μέρα λόγω απόπτωσης, ενώ ένα μέσο παιδί ηλικίας 8 έως 14 ετών, χάνει περίπου 20-30 δισεκατομμύρια κύτταρα.

Για την επίτευξη της απόπτωσης, έχει αποδειχθεί ότι η ενεργοποίηση των κασπασών, οι οποίες είναι μια οικογένεια ενζύμων, παίζει πολύ καθοριστικό ρόλο. Οι κασπάσες παίζουν κεντρικό ρόλο στη μεταβίβαση των αποπτωτικών σημάτων και υπάρχουν 2 είδη κασπασών, οι κασπάσες εκκίνησης και οι κασπάσες καθοριστές. Αρχικά, η αποπτωτική σηματοδότηση ενεργοποιεί μια σειρά από κασπάσες τις -2, -8, -9 και -10, που ονομάζονται και κασπάσες εκκίνησης.. Στη συνέχεια, οι κασπάσες εκκίνησης με τη σειρά τους διασπώνται και ενεργοποιούν τις κασπάσες καθοριστές (κασπάσες -3, -6 και -7). Τέλος, οι κασπάσες καθοριστές εκτελούν απόπτωση διασπώντας κυτταρικές πρωτεΐνες.

Η απόπτωση έχει ως αποτέλεσμα σημαντικές μορφολογικές αλλαγές στο κύτταρο. Στις αρχικές φάσεις, ένα κύτταρο που υφίσταται απόπτωση χάνει τις κυτταρικές επαφές του και αλλάζει σχήμα. Η χρωματίνη, που είναι το υλικό που αποτελεί ένα χρωμόσωμα και αποτελείται από DNA και πρωτεΐνες, συμπυκνώνεται στον πυρήνα . Η συμπύκνωση αυτή ονομάζεται πυκνωση, και ξεκινά την αποικοδόμηση του DNA. Η απώλεια νερού οδηγεί σε σημαντική συρρίκνωση των κυττάρων και διόγκωση της μεμβράνης του πλάσματος, με τον πυρήνα του κυττάρου και το κυτταρόπλασμα να σπάζουν σε

αποπτωτικά σώματα. Τέλος, γίνεται απομάκρυνση των νεκρών κυττάρων ολοκληρώνοντας τη διαδικασία της απόπτωσης.

Η απομάκρυνση των νεκρών κυττάρων γίνεται από γειτονικά φαγοκύτταρα. Τα ετοιμοθάνατα κύτταρα που υπόκεινται τα τελευταία στάδια της απόπτωσης παρουσιάζουν στην επιφάνειά τους μόρια που αναγνωρίζονται από τα φαγοκύτταρα. Αυτά τα κύτταρα σηματοδοτούν τα κύτταρα για φαγοκυττάρωση από κύτταρα που κατέχουν κατάλληλους υποδοχείς, όπως τα μακροφάγα. Μετά την αναγνώριση, το φαγοκύτταρο αναδιοργανώνει τον κυτταροσκελετό του ώστε να ενδοκυτταρώσει (καταπιεί) το κύτταρο. Η απομάκρυνση των νεκρών κυττάρων από τα φαγοκύτταρα συμβαίνει χωρίς να προκληθεί φλεγμονώδης αντίδραση..

2.3 Αποπτωτικές οδοί σηματοδότησης

Υπάρχουν 2 γνωστοί τρόποι ενεργοποίησης των κασπασών στις αποπτωτικές οδούς σηματοδότησης, η ενδογενής οδός (intrinsic pathway) και η εξωγενής οδός (extrinsic pathway). Η ενδογενής και εξωγενής οδοί απόπτωσης είναι και οι δύο φυσικές διαδικασίες με τις οποίες ένα κύτταρο κατευθύνεται στον προγραμματισμένο κυτταρικό θάνατο.

Και οι δύο οδοί απόπτωσης ενεργοποιούν μια αλληλουχία σηματοδότησης κυττάρων που είναι απαραίτητο μέρος της ανάπτυξης και της λειτουργίας ενός οργανισμού. Επίσης, και στις δύο οδούς, η σηματοδότηση οδηγεί στην ενεργοποίηση μιας οικογένειας πρωτεασών Cys (Cysteine), που ονομάζονται κασπάσες, που έχουν στόχο την αποσυναρμολόγηση και την αφαίρεση του κυττάρου που πεθαίνει.

Η εξωγενής οδός απόπτωσης ξεκινά έξω από ένα κύτταρο, όταν οι συνθήκες στο εξωκυτταρικό περιβάλλον καθορίζουν ότι ένα κύτταρο πρέπει να πεθάνει. Η ενδογενής οδός της απόπτωσης ξεκινά όταν ένας τραυματισμός συμβαίνει μέσα στο κύτταρο και το στρες που προκύπτει ενεργοποιεί την αποπτωτική οδό.

Χρησιμοποιώντας τον όρο ενδογενής οδός, αναφερόμαστε στην έναρξη της αποπτωτικής οδού εντός του κυττάρου η οποία είναι το αποτέλεσμα διαφόρων εσωτερικών ερεθισμάτων, όπως, η γενετική βλάβη του DNA και το κυτταρικό στρες. Αυτά τα διάφορα ερεθίσματα οδηγούν τελικά σε αλλαγές στην ακεραιότητα της μιτοχονδριακής μεμβράνης, η οποία ρυθμίζεται από τις πρωτεΐνες της οικογένειας Bcl-2. Η ισορροπία μεταξύ των προ-αποπτωτικών και αντι-αποπτωτικών μελών της οικογένειας Bcl-2 καθορίζει εάν ένα κύτταρο θα υποβληθεί ή όχι την διαδικασία της απόπτωσης. Η πρωτεΐνη p53 είναι ένας αισθητήρας κυτταρικού στρες και είναι ίσως ο πιο κρίσιμος ενεργοποιητής της εσωτερικής οδού.

Συνεχίζοντας με την εξωγενή οδό ενεργοποίησης, η εξωγενής οδός απόπτωσης ξεκινά έξω από ένα κύτταρο, όταν οι συνθήκες στο εξωκυτταρικό περιβάλλον καθορίζουν ότι ένα κύτταρο πρέπει να πεθάνει. Η ενεργοποίηση των κασπασών επιτυγχάνεται με την σύνδεση των συνδετών θανάτου (ligands), με τους υποδοχείς θανάτου (death receptors). Οι υποδοχείς θανάτου αποτελούν μέρος της οικογένειας του γονιδίου του παράγοντα νέκρωσης όγκου (TNF) και παρέχουν μια γρήγορη και αποτελεσματική οδό για την απόπτωση.

Η απόπτωση είναι μια διαδικασία κατά την οποία τα κύτταρα υφίστανται προγραμματισμένο θάνατο και αντισταθμίζουν τον πολλαπλασιασμό. Είναι μορφολογικά διακριτή από τη νέκρωση και περιλαμβάνει συρρίκνωση και κατακερματισμό τόσο του πυρήνα όσο και του κυττάρου χωρίς ρήξη της κυτταρικής μεμβράνης. Αυτό αποτρέπει τη φλεγμονή του περιβάλλοντος ιστού. Η απόπτωση βασίζεται στην ενεργοποίηση διακριτών οδών σηματοδότησης που συχνά απορρυθμίζονται στον καρκίνο. Έτσι, μας δίνεται η ικανότητά να εκμεταλλευόμαστε αυτά τα μονοπάτια για να σχεδιάζουμε πιο αποτελεσματικές και μη τοξικές θεραπείες για τον καρκίνο.

2.4 Διεξαγωγή ενός πειράματος στην Βιολογία

Για την διεξαγωγή ενός πειράματος στα εργαστήρια της Βιολογίας, ο ερευνητής ουσιαστικά πραγματοποιεί 2 πειράματα. Ο ερευνητής, στο πρώτο πείραμα, επωάζει τα καρκινικά κύτταρα με ουσίες που προκαλούν απόπτωση. Παράλληλα, στο δεύτερο

πείραμα, έχει κύτταρα control τα οποία είναι είτε φυσιολογικά ή που δεν έχουν επωαστεί με κάτι. Στη συνέχεια αφήνει τις ουσίες για 48 περίπου ώρες και στη συνέχεια κάνει απομόνωση των πρωτεϊνών που θέλει να μελετήσει. Στη συνέχεια χρησιμοποιεί την τεχνική Western Blot έτσι ώστε να δει ημιποσοτικά τα επίπεδα των πρωτεϊνών. Δηλαδή με αυτή την τεχνική ο ερευνητής συγκρίνει τα επίπεδα των πρωτεϊνών στα 2 πειράματα για να δημιουργήσει συμπεράσματα για το πως επηρεάστηκε η πρωτεΐνη που θέλει να μελετήσει.

Κεφάλαιο 3

Λογική και Λογικός Προγραμματισμός

3.1 Εισαγωγή	10
3.2 Λογική	10
3.3 Λογικός Προγραμματισμός – Prolog	12

3.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα δούμε αναλυτικά την έννοια της Λογικής καθώς και τον λογικό προγραμματισμό. Τέλος, θα αναλύσουμε τις βασικές αρχές της γλώσσας λογικού προγραμματισμού Prolog.

3.2 Λογική

Υπάρχει μια αρχαία άποψη, που εξακολουθεί να θεωρείται σωστή, ότι αυτό που κάνει τα ανθρώπινα όντα ξεχωριστά, αυτό που μας ξεχωρίζει από τα ζώα, είναι ότι είμαστε λογικοί. Αυτή η συλλογιστική ικανότητα μπορεί να γίνει καλά και μπορεί να γίνει άσχημα. μπορεί να γίνει με σωστό τρόπο ή να γίνει με λάθος τρόπο. Δηλαδή, η λογική είναι η πειθαρχία που στοχεύει να διακρίνει τον έγκυρο συλλογισμό και συμπερασμό.

Ο καλός συλλογισμός όμως, δεν είναι απαραίτητα αποτελεσματικός συλλογισμός. Στην πραγματικότητα, ο κακός συλλογισμός είναι συχνά εξαιρετικά αποτελεσματικός - με την έννοια ότι οι άνθρωποι συχνά πείθονται από αυτό. Ένα πρόσφατο παράδειγμα

κακού συλλογισμού ήταν ο Χίτλερ, ο οποίος κατάφερε να πείσει ένα ολόκληρο έθνος να ακολουθήσει μια ποικιλία προτάσεων που δεν ήταν μόνο ψευδείς αλλά και εντελώς λάθος από ηθικής πλευράς.

Η λογική περιέχει αξιώματα ή δηλώσεις — δημιουργώντας και υποστηρίζοντάς τα με λόγους, καταγράφοντας τις συνέπειές τους. Οι προτάσεις είναι αυτά που ισχυριζόμαστε ή δηλώνουμε. Οι προτάσεις είναι τα είδη πραγμάτων που μπορεί να είναι αληθή ή ψευδή, και εκφράζονται με δηλωτικές προτάσεις. Χρησιμοποιούμε τέτοιες προτάσεις για να κάνουμε κάθε είδους ισχυρισμούς, όπως πραγματικά γεγονότα («η Γη περιστρέφεται γύρω από τον Ήλιο»), ακόμα και ισχυρισμούς σχετικά με την ηθική («Είναι λάθος να τρώμε κρέας»).

Επίσης σημαντικό είναι να διακρίνουμε προτάσεις σε δηλωτικό ύφος οι οποίες εκφράζουν προτάσεις, από προτάσεις σε άλλα ύφη, που δεν το κάνουν. Για παράδειγμα, υπάρχουν οι ανακριτικές προτάσεις οι οποίες θέτουν ερωτήσεις ("Βρέχει;") και οι επιτακτικές προτάσεις εκδίδουν εντολές ("Μην κάνετε φασαρία."). Δεν έχει νόημα να δούμε αν αυτά τα είδη προτάσεων είναι αλήθειες ή ψευδείς, οπότε δεν εκφράζουν προτάσεις.

Διακρίνουμε επίσης τις προτάσεις από τις προτάσεις που τις εκφράζουν, επειδή μια πρόταση μπορεί να εκφραστεί με διαφορετικές προτάσεις. Για παράδειγμα ας δούμε τις προτάσεις «Βρέχει» και «It's raining». Και οι δύο εκφράζουν την πρόταση ότι βρέχει. μια πρόταση το κάνει στα Ελληνικά και η άλλη στα Αγγλικά. Ένα άλλο παράδειγμα είναι οι προτάσεις «Ο Γιάννης διαβάζει» και «Ο Γιάννης κάνει τα μαθήματά του» οι οποίες εκφράζουν την ίδια πρόταση αλλά με διαφορετικό τρόπο διατύπωσης.

Το κυριότερο συστατικό στον συλλογισμό είναι το επιχείρημα. Στη λογική, ένα επιχείρημα είναι μια σειρά από δηλώσεις που χρησιμοποιούνται συνήθως για να πειστεί κάποιος ή να παρουσιαστούν οι λόγοι για την αποδοχή ενός συμπεράσματος. Αν συλλογιστούμε κάνοντας αξιώσεις και υποστηρίζοντας τους με λόγους, τότε ο ισχυρισμός που υποστηρίζεται είναι το συμπέρασμα ενός επιχειρήματος. οι λόγοι που δίνονται για να το υποστηρίξουν είναι το επιχείρημα. Αν συλλογιστούμε αντλώντας ένα συμπέρασμα από ένα σύνολο δηλώσεων, τότε το συμπέρασμα που συνάγουμε είναι το

συμπέρασμα ενός επιχειρήματος και οι δηλώσεις από τις οποίες συντάσσεται είναι οι προϋποθέσεις.

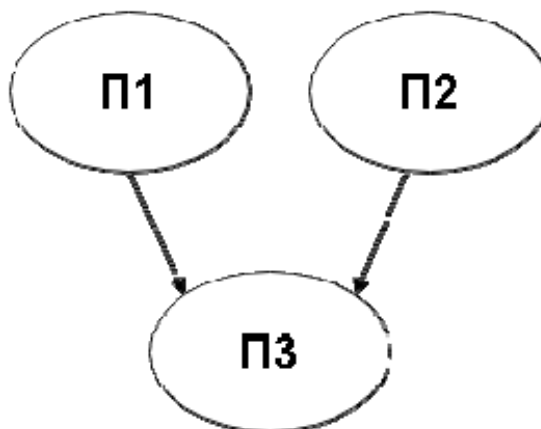
Το πιο γνωστό παράδειγμα είναι το σύνολο των προτάσεων που ακολουθεί :

Π1 :Οι άνθρωποι είναι θνητοί

Π2 :Ο Σωκράτης είναι άνθρωπος

Π3: Άρα, ο Σωκράτης είναι θνητός.

Όπου οι προτάσεις Π1 και Π2 είναι το σύνολο των δηλώσεων μας και η πρόταση Π3 είναι το συμπέρασμα που προκύπτει από τις προτάσεις Π1 και Π2.



Σχήμα 3.1 : Διάγραμμα προτάσεων και συμπεράσματος

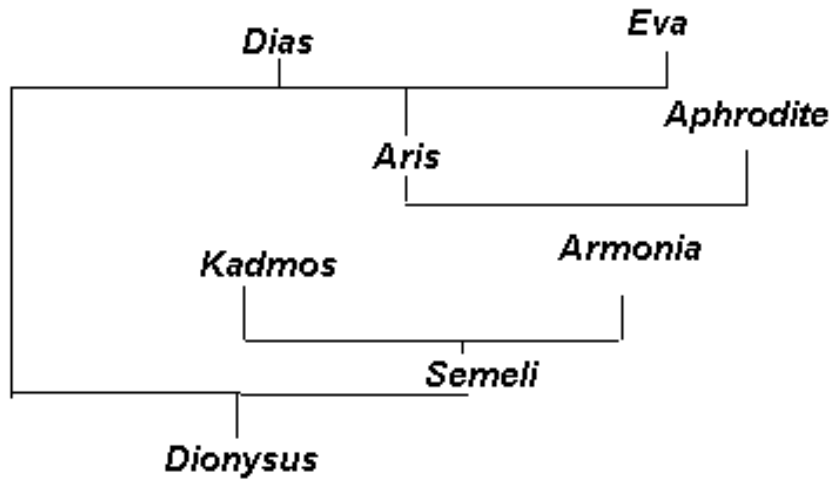
3.3 Λογικός Προγραμματισμός – Prolog

Ο λογικός προγραμματισμός είναι ένας τύπος προγραμματισμού που βασίζεται στη λογική. Αυτό σημαίνει ότι μια λογική γλώσσα προγραμματισμού έχει προτάσεις που ακολουθούν τη λογική έτσι ώστε να εκφράζουν γεγονότα και κανόνες. Ο υπολογισμός με λογικό προγραμματισμό γίνεται με τη δημιουργία λογικών συμπερασμάτων με βάση όλα τα διαθέσιμα δεδομένα. Προκειμένου τα προγράμματα υπολογιστών να κάνουν

χρήση του λογικού προγραμματισμού, πρέπει να υπάρχει μια βάση για την υπάρχουσα λογική, που ονομάζεται κατηγορική. Τα προθέματα χρησιμοποιούνται για την κατασκευή μεμονωμένων τύπων ή ατόμων, τα οποία δηλώνουν αληθινά γεγονότα. Τα πρόθεμα και τα άτομα χρησιμοποιούνται για τη δημιουργία τύπων και την εκτέλεση ερωτημάτων. Οι λογικές γλώσσες συχνά βασίζονται σε ερωτήματα για την εμφάνιση σχετικών δεδομένων. Αυτές οι ερωτήσεις μπορούν να αποτελέσουν μέρος της μηχανικής μάθησης, η οποία μπορεί να πραγματοποιηθεί χωρίς να απαιτείται χειρονακτική παρέμβαση.

Υπάρχουν πολλές διαφορετικές γλώσσες λογικού προγραμματισμού. Η πιο κοινή γλώσσα, η Prolog η οποία πήρε το όνομά της από το γαλλικό PROgrammation en LOGique, ή ο προγραμματισμός στη λογική, μπορεί επίσης να διασυνδεθεί με άλλες γλώσσες προγραμματισμού όπως η Java, κάτι που υλοποιήσαμε στα πλαίσια της παρούσας διπλωματικής εργασίας. Εκτός από τη δημοφιλέστερη λογική γλώσσα προγραμματισμού, η Prolog ήταν επίσης μία από τις πρώτες γλώσσες λογικού προγραμματισμού, αφού το πρώτο πρόγραμμα στην Prolog δημιουργήθηκε τη δεκαετία του 1970. Η Prolog αναπτύχθηκε χρησιμοποιώντας λογική πρώτης τάξης, που ονομάζεται επίσης λογική κατηγορίας, η οποία επιτρέπει τη χρήση μεταβλητών και όχι προτάσεων. Η Prolog χρησιμοποιεί τεχνητή νοημοσύνη (AI) για να βοηθήσει στη διαμόρφωση των συμπερασμάτων της και μπορεί να επεξεργαστεί γρήγορα μεγάλες ποσότητες δεδομένων. Η Prolog μπορεί να εκτελεστεί με ή χωρίς χειροκίνητες εισόδους, πράγμα που σημαίνει ότι μπορεί να προγραμματιστεί να εκτελείται αυτόματα ως μέρος της επεξεργασίας δεδομένων.

Για καλύτερη κατανόηση της γλώσσας λογικού προγραμματισμού Prolog θα μελετήσουμε το εξής παράδειγμα



Σχήμα 3.2 : Παράδειγμα Prolog

father (dias, aris).

father (dias, dionysus).

father (aris, armonia).

father (kadmos, semeli).

mother (eva, aris).

mother (armonia, semeli).

mother (semeli, dionysus).

mother (aphrodite, armonia).

woman (armonia).

woman (era).

woman (semeli).

woman (aphrodite).

man (aris).

man (dias).

man (dionysus).

man (kadmos).

Οι παραπάνω προτάσεις είναι τα γεγονότα του προγράμματος και βλέπουμε ότι αποτελούν την βάση δεδομένων για το πρόγραμμα της Prolog.

Ένα γεγονός είναι η απλούστερη μορφή πρότασης και εκφράζει μια ιδιότητα των ορισμάτων του.

Για παράδειγμα ας μελετήσουμε το γεγονός father (dias, aris). Παρατηρούμε ότι το γεγονός εκφράζει μια σχέση Father το οποίο ονομάζεται κατηγορημα (Predicate) μεταξύ των ορισμάτων του (Arguments) τα οποία είναι τα ονόματα dias και aris. Δηλαδή με λίγα λόγια αυτή η πρόταση μπορεί να αντιστοιχιστεί με την πρόταση ο dias είναι πατέρας του aris. Επίσης, μπορούμε να πούμε ότι επειδή υπάρχουν 2 ορίσματα η τάξη (arity) του κατηγορήματος father είναι 2.

Ένα άλλο διαφορετικό παράδειγμα που μπορούμε να μελετήσουμε είναι το man (dias). Παρατηρούμε ότι το γεγονός εκφράζει μια σχέση man στο όρισμα dias. Δηλαδή αυτή η πρόταση μπορεί να αντιστοιχιστεί με την πρόταση ο dias είναι άντρας. Επίσης, μπορούμε να πούμε ότι επειδή υπάρχει 1 όρισμα, η τάξη του κατηγορήματος man είναι 2.

Όπως είχα αναφέρει και πριν, οι λογικές γλώσσες συχνά βασίζονται σε ερωτήματα για την εμφάνιση σχετικών δεδομένων. Αυτό ισχύει και στην Prolog η οποία υποστηρίζει τις εξής τύπους ερωτήσεων :

- Απλές ερωτήσεις (Simple)

? father (dias, aris)

- Yes

Σε αυτή την ερώτηση η Prolog θα ψάξει στην βάση δεδομένων από τα γεγονότα για να βρει αν ισχύει. Αν την βρει απαντά θετικά, ενώ αν δεν την βρει απαντά αρνητικά.

- Ερωτήσεις με μεταβλητές (variables)

? father (dias , X)

- X = aris;

- X =dionysus

- Yes.

Σε αυτή την ερώτηση η Prolog θα ψάξει στην βάση δεδομένων από τα γεγονότα για να βρει αν μπορεί να αντιστοιχίσει την μεταβλητή X με κάποιο από τα ορίσματα και επιστρέφει όλα τα ορίσματα που αντιστοιχίζονται.

- Σύνθετες (Conjunctive Queries)

? father (aris, armonia), woman (armonia).

- Yes.

Σε αυτή την ερώτηση η Prolog θα ψάξει στην βάση δεδομένων από τα γεγονότα για να βρει αν ισχύουν και τα 2 γεγονότα. Αν ισχύουν και τα 2 απαντά θετικά, ενώ αν δεν ισχύουν απαντά αρνητικά.

- Ερωτήσεις με μεταβλητές (Queries with variables)

? mother (era ,X), man(Y).

- X = aris Y= aris;

- X = aris Y = dias;

- X = aris Y = dionysus;

- Yes.

Σε αυτή την ερώτηση η Prolog θα ψάξει στην βάση δεδομένων από τα γεγονότα για να βρει αν μπορεί να αντιστοιχίσει τις μεταβλητές X και Y με κάποιο από τα ορίσματα και επιστρέφει όλα τα ορίσματα που αντιστοιχίζονται.

? mother (era ,X), man(X).

- X = aris

- No.

Σε αυτή την ερώτηση η Prolog θα ψάξει στην βάση δεδομένων από τα γεγονότα για να βρει αν μπορεί να αντιστοιχίσει την μεταβλητή X με κάποιο από τα ορίσματα και επιστρέφει όλα τα ορίσματα που αντιστοιχίζονται. Αξίζει να σημειωθεί ότι σε αυτή την περίπτωση η μεταβλητή X πρέπει να είναι κοινή ανάμεσα στα 2 γεγονότα.

Όπως είδαμε, η Prolog υποστηρίζει ερωτήσεις με μεταβλητές. Οι μεταβλητές αντιπροσωπεύουν όρους οι οποίοι είναι άγνωστοι κατά τη στιγμή διατύπωσης της πρότασης και είναι τοπικές για κάθε πρόταση. Δεν έχουν κάποιο συγκεκριμένο αφού μπορούν να αντικατασταθούν με τις τιμές τύπου character, integer, και string. Σε κάθε χρονική στιγμή της εκτέλεσης μια μεταβλητή μπορεί να έχει τιμή (instantiated) ή να μην έχει (un-instantiated).

Για να δημιουργήσουμε κάτι πιο περίπλοκο, εκτός από ερωτήσεις, η γλώσσα λογικού προγραμματισμού Prolog υποστηρίζει και κανόνες (Rules).

Ένας κανόνας έχει την μορφή $A :- B_1, \dots, B_n$ όπου A είναι η κεφαλή (Head) ενώ το σύνολο $B_1, \dots, B_n$ αποτελεί το Σώμα (Body). Η πρόταση διαβάζεται με τον εξής τρόπο: Ισχύει το A αν ισχύει το σύνολο των B

Για παράδειγμα, οι πρότασεις $\text{parent}(X, Y) :- \text{father}(X, Y)$ και $\text{parent}(X, Y) :- \text{mother}(X, Y)$, σημαίνει ότι ο X είναι γονιός του Y αν ο X είναι πατέρας ή μητέρα του Y .

Έστω ότι έχουμε τις ακόλουθες 2 ερωτήσεις :

?father (aris, X), woman (X)

X = armonia

? daughter (X, aris)

X = armonia

Μπορούμε να διακρίνουμε ότι οι 2 αυτές ερωτήσεις είναι ισοδύναμες αφού η πρώτη θέτει τις ερωτήσεις ποιου είναι πατέρας ο aris και να είναι γυναίκα , ενώ η δεύτερη θέτει την ερώτηση ποια είναι κόρη του aris.

Έτσι, βλέπουμε ότι μπορούμε να δημιουργήσουμε τον κανόνα daughter :

$\text{daughter}(X, Y) :- \text{father}(Y, X), \text{woman}(X)$, όπου ουσιαστικά αποτυπώνουμε την σχέση κόρη.

Κεφάλαιο 4

Αναπαράσταση των Pathways και μετάδοση σήματος σε λογικό προγραμματισμό (Συνέπεια Υποθέσεων)

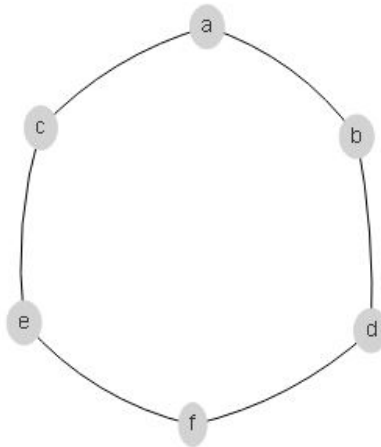
4.1 Εισαγωγή	18
4.2 Αναπαράσταση γράφων και δικτύων	18
4.3 Μετάδοση σήματος	19

4.1 Εισαγωγή

Στην παρούσα διπλωματική εργασία η αναπαράσταση των αποπτωτικών οδών σηματοδότησης βασίζεται στο σχήμα ενός κατευθυνόμενου γράφου, όπου οι κόμβοι του αντιπροσωπεύουν τις πρωτεΐνες και οι ακμές του την σχέση μεταξύ των πρωτεϊνών. Αρχικά θα μελετήσουμε ένα απλό παράδειγμα για το πως μπορούμε να αναπαραστήσουμε γράφους και δίκτυα χρησιμοποιώντας λογικό προγραμματισμό. Έπειτα, θα επεκτείνουμε το αρχείο μας για να δείξουμε την σχέση μεταξύ των πρωτεϊνών και την μετάδοση του σήματος ανάμεσα στις πρωτεΐνες.

4.2 Αναπαράσταση γράφων και δικτύων

Για να αναπαραστήσουμε γράφους και δίκτυα θα χρησιμοποιήσουμε το ακόλουθο μικρό παράδειγμα:



Σχήμα 4.1 : Παράδειγμα απλού γράφου

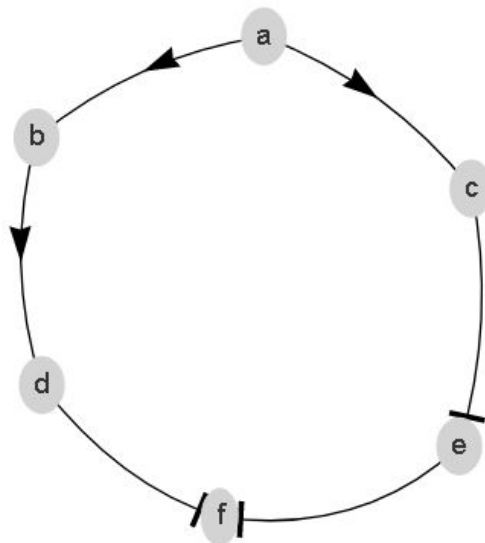
Για την αναπαράσταση του παραπάνω γράφου θα χρειαστεί να υλοποιήσουμε στην γλώσσα Prolog ένα μικρό και απλό πρόγραμμα στο οποίο θα έχουμε τα ακόλουθα κατηγορήματα(predicates):

```
connected(a,b).  
connected(b,d).  
connected(d,f).  
connected(a,c).  
connected(c,e).  
connected(e,f).
```

Όπου ουσιαστικά δηλώνουμε τις ενώσεις μεταξύ των κόμβων. Δηλαδή ότι ο κόμβος a είναι ενωμένος με τον κόμβο b, μετά ότι ο κόμβος b είναι ενωμένος με τον κόμβο d και ούτω καθεξής.

4.3 Μετάδοση σήματος

Για να αναπαραστήσουμε την μετάδοση του σήματος στο δίκτυο μας θα χρησιμοποιήσουμε το ακόλουθο μικρό παράδειγμα:



Σχήμα 4.2 : Παράδειγμα γράφου για μετάδοση σήματος

Για την αναπαράσταση του παραπάνω γράφου θα χρειαστεί να επεκτείνουμε το ήδη υπάρχον αρχείο στο οποίο έχουμε δηλώσει όλα τα κατηγορήματα τύπου `connected` με τα εξής κατηγορήματα :

`induce(a,b).`

`induce(a,c).`

`induce(b,d).`

`induce(d,f).`

`inhibit(c,e).`

`inhibit(e,f).`

Με τα κατηγορήματα τύπου `induce` δείχνουμε ενεργοποίηση των πρωτεϊνών, δηλαδή με την εισαγωγή του κανόνα `induce(a,b)`, δηλώνουμε την ενεργοποίηση της πρωτεΐνης `b` η οποία συνεχίζει την μετάδοση του σήματος στην επόμενη πρωτεΐνη.

Στην αντίθεση περίπτωση δηλώνοντας το κατηγορήμα `inhibit` μπορούμε να δείξουμε δείχνουμε την απενεργοποίηση των πρωτεϊνών, δηλαδή με την εισαγωγή του κανόνα `inhibit(c,e)`, δηλώνουμε την απενεργοποίηση της πρωτεΐνης `e` η οποία μπλοκάρει την μετάδοση του σήματος στην επόμενη πρωτεΐνη.

Συνεχίζοντας προσθέτουμε τα εξής κατηγορήματα

`not_signal(Y):-connected(X,Y),inhibit(X,Y).`

`signal(Y):-connected(X,Y),signal(X),induce(X,Y),not(not_signal(Y)).`

Εισάγοντας το γενικό κατηγορήμα `not_signal(Y)`, δηλώνουμε ότι μια πρωτεΐνη Y δεν έχει ενεργοποιημένο σήμα όταν ισχύουν τα κατηγορήματα

`connected(X,Y)` : η πρωτεΐνη Y να είναι ενωμένη με μια άλλη πρωτεΐνη X

`inhibit(X,Y)`, η πρωτεΐνη X μπλοκάρει την πρωτεΐνη Y

Στην αντίθετη περίπτωση, εισάγοντας το αναδρομικό κατηγορήμα `signal(Y)`, δηλώνουμε ότι μια πρωτεΐνη Y έχει ενεργοποιημένο σήμα όταν ισχύουν τα εξής κατηγορήματα :

`connected(X,Y)` : η πρωτεΐνη Y να είναι ενωμένη με μια άλλη πρωτεΐνη X

`signal(X)` : η πρωτεΐνη X πρέπει να έχει σήμα για να μεταδώσει

`induce(X,Y)` : η πρωτεΐνη Y ενεργοποιείται από την πρωτεΐνη X

`not(not_signal(Y))` : δεν υπάρχει κάποια άλλη πρωτεΐνη που να μπλοκάρει το σήμα στην πρωτεΐνη Y

Για περισσότερη επεξήγηση ας υποθέσουμε το πιο πάνω παράδειγμα στην εικόνα 3 :

Έστω ότι θέλουμε να ξεκινήσουμε το σήμα από την πρωτεΐνη a και θέλουμε να δούμε ποιες πρωτεΐνες θα ενεργοποιηθούν:

```
connected(a,b).
connected(b,d).
connected(d,f).
connected(a,c).
connected(c,e).
connected(e,f).

induce(a,b).
induce(a,c).
induce(b,d).
induce(d,f).

inhibit(c,e).
inhibit(e,f).

not_signal(Y):-connected(X,Y),inhibit(X,Y).

signal(Y):-connected(X,Y),signal(X),induce(X,Y),not(not_signal(Y)).
signal(a).
```

Σχήμα 4.3 : Αρχείο example.pl

Προσθέτουμε στον αρχείο μας το κατηγορήμα `signal(a)`. Βάση του αναδρομικού κανόνα `signal(Y)`, η `prolog` θα δοκιμάσει να ενεργοποιήσει την πρωτεΐνη `b`. Για να ισχύει το `signal(b)` θα πρέπει να ισχύουν οι κανόνες `connected(X,b)`, `signal(X)`, `induce(X,b)` και `not(not_signal(b))`. Για να ισχύει όμως ο κανόνας `not(not_signal(b))` θα πρέπει να ισχύουν οι κανόνες `connected(X,b)` και `inhibit(X,b)`. Αφού η `prolog` δεν βρίσκει κάποια μεταβλητή για την οποία ισχύουν οι κανόνες `connected(X,b)` και `inhibit(X,b)`, έχουμε το συμπέρασμα ότι δεν ισχύει το κατηγορήμα `not_signal(b)`, έτσι έχουμε το συμπέρασμα ότι ισχύει η άρνηση του, δηλαδή ότι ισχύει η συνθήκη `not(not_signal(b))`. Για τους κανόνες `connected(X,b)`, `signal(X)`, `induce(X,b)` η `prolog` ταυτίζει την μεταβλητή `X` με την πρωτεΐνη `a` αφού ισχύουν οι κανόνες `connected(a,b)`, `signal(a)`, `induce(a,b)`. Έτσι βλέπουμε ότι ισχύει ο κανόνας `signal(b)` και το σήμα μεταφέρεται και στην πρωτεΐνη `b`. Έπειτα η `prolog` θα επιχειρήσει να επαναλάβει την ίδια διαδικασία για τις υπόλοιπες πρωτεΐνες.

Τρέχουμε στην prolog την εντολή - ερώτηση `signal(X)`. και παίρνουμε ως αποτέλεσμα όλες τις μεταβλητές με τις οποίες ταυτίζεται το `X`, δηλαδή όλες τις πρωτεΐνες που έχουν ενεργοποιηθεί, όπως φαίνεται στο πιο κάτω σχήμα.

```
Welcome to SWI-Prolog (threaded, 64 bits, version 8.2.3)
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/HP/Desktop/example.pl compiled 0.00 sec, 15 clauses
?- signal(X).
X = b ;
X = d ;
X = c ;
X = a.
```

Σχήμα 4.4 : Αποτελέσματα εκτέλεσης αρχείου `example.pl`

Κεφάλαιο 5

Το σύστημα Bio_Cancer_Logic

5.1 Εισαγωγή	24
5.2 Υλοποίηση του συστήματος	24
5.2.1 Home	25
5.2.2 Simulation	26
5.2.3 How to Use	32
5.2.4 About	33
5.3 Αρχιτεκτονική Συστήματος	33
5.3.1 Εκτέλεση Προσομοίωσης Πειράματος	33
5.3.2 Εκτέλεση Προσομοίωσης Πειράματος με Υποθέσεις	35

5.1 Εισαγωγή

Στην παρούσα διπλωματική εργασία ο κύριος μας στόχος ήταν η ανάπτυξη ενός συστήματος που θα βοηθούσε τους επιστήμονες στο Τμήμα Βιολογίας να διεξάγουν πειράματα στον ηλεκτρονικό υπολογιστή. Σε αυτό το κεφάλαιο θα δούμε πιο αναλυτικά την υλοποίηση και αρχιτεκτονική του συστήματος μας καθώς και τι χρειάζεται να ξέρει ο χρήστης για να μπορεί να χρησιμοποιήσει το σύστημα.

5.2 Υλοποίηση του συστήματος

Η αρχική ιδέα ήταν να δημιουργήσουμε στον server του Πανεπιστημίου μια ιστοσελίδα μέσω της οποίας θα είχαν πρόσβαση οι επιστήμονες του τμήματος της Βιολογίας και στην οποία θα μπορούσαν να διεξάγουν ηλεκτρονικά τα πειράματά τους για δική τους ευκολία. Αυτό μπορεί να αποτελέσει ένα χρήσιμο εργαλείο για τους ερευνητές καθώς

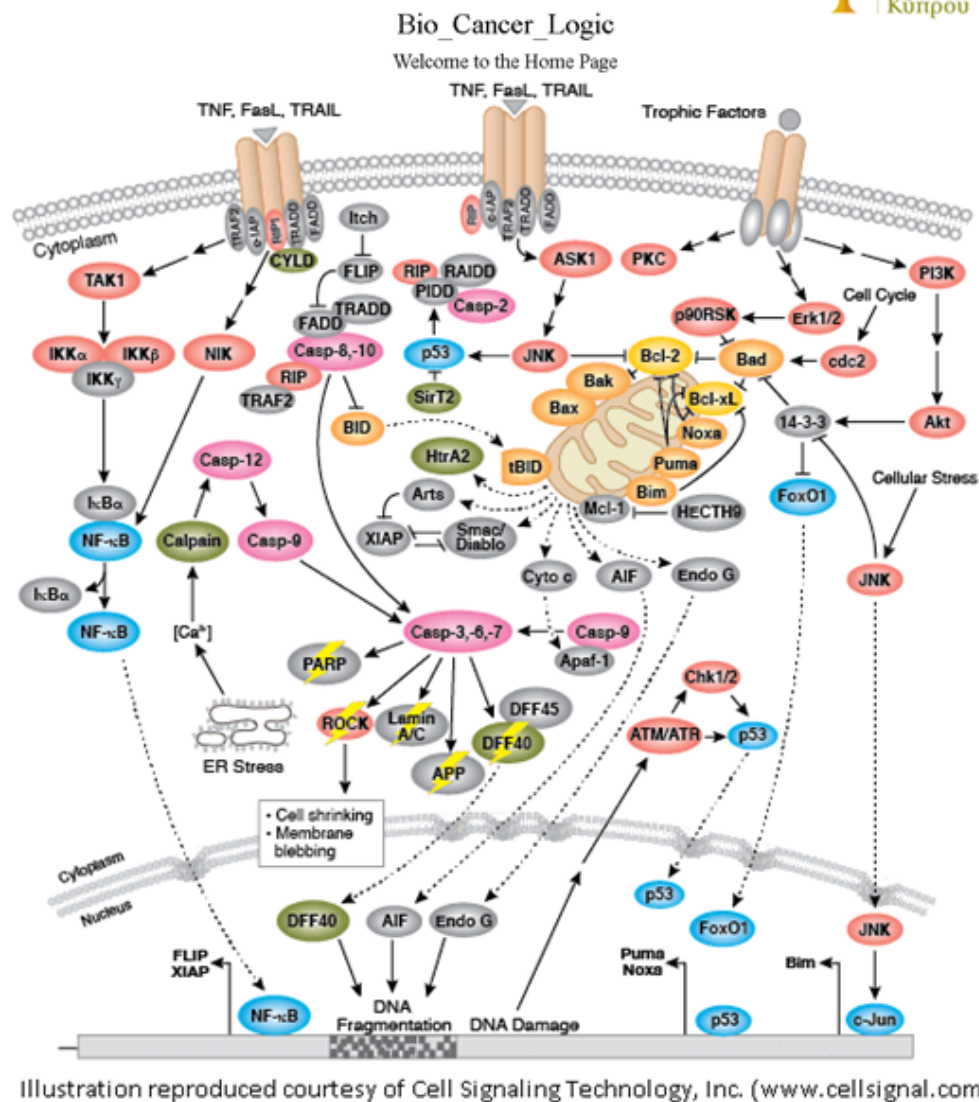
δεν χρειάζεται πλέον να διεξάγουν στο εργαστήριο τους χρονοβόρα πειράματα αλλά μπορούν να εκτελέσουν μια προσομοίωση του πειράματος στην ιστοσελίδα και να έχουν τα αποτελέσματα σε διάστημα μερικών δευτερολέπτων.

Ακολουθώντας τον σύνδεσμο [16], ο χρήστης παραπέμπεται στην αρχική σελίδα του συστήματος. Από εκεί μπορεί να επισκεφθεί όποια άλλη ιστοσελίδα επιθυμεί χρησιμοποιώντας το navigation bar το οποίο βρίσκεται στην πάνω αριστερά γωνιά της οθόνης. Οι διαθέσιμες ιστοσελίδες που μπορεί να επισκεφθεί ο χρήστης είναι :

- Home
- Simulation
- How to use
- About

5.2.1 Home

Αυτή η σελίδα αποτελεί την αρχική σελίδα του συστήματος. Η σελίδα αυτή (βλ. Σχήμα 5.1), περιέχει το σχήμα που αναπαριστά τις αποπτωτικές οδούς στις οποίες βασίστηκε η διπλωματική εργασία, το λογότυπο του Πανεπιστημίου Κύπρου καθώς και το navigation bar μέσω του οποίου ο χρήστης έχει πρόσβαση στις υπόλοιπες σελίδες του συστήματος.

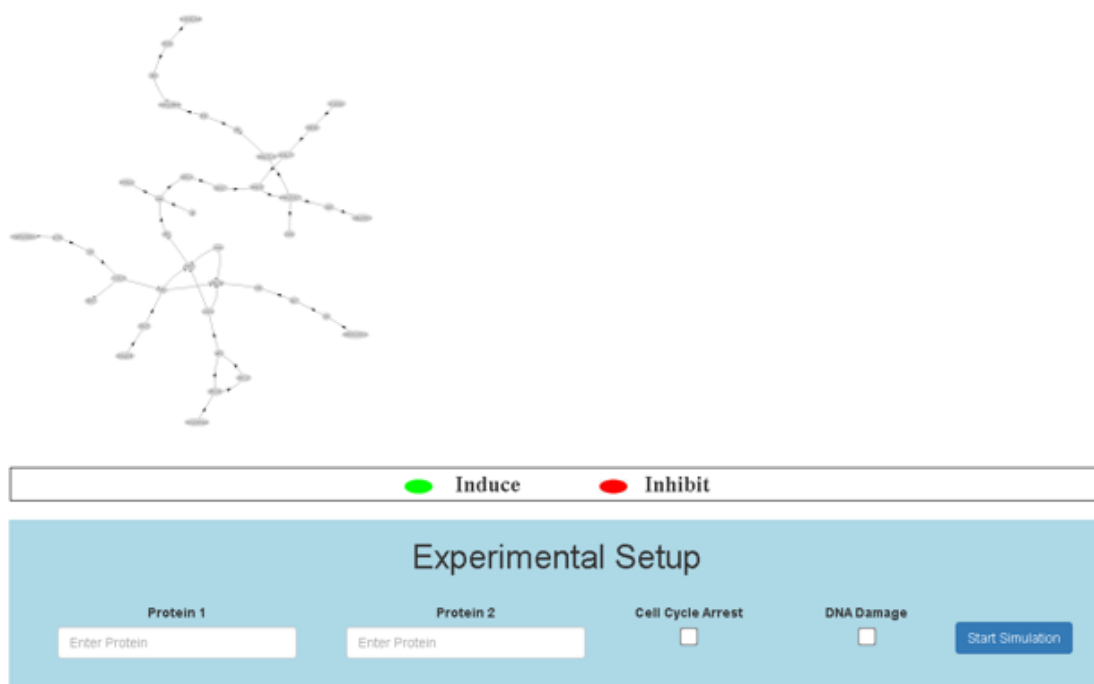


Σχήμα 5.1 : Αρχική Σελίδα

5.2.2 Simulation

Αυτή η σελίδα αποτελεί την κύρια σελίδα του συστήματος. Στην σελίδα αυτή ο χρήστης μπορεί να εκτελέσει προσομοιώσεις σε πειράματα (βλ. Σχήμα 5.2) καθώς και να δημιουργήσει παρατηρήσεις αναφορικά με τα πειράματα και να επιβεβαιώσει κατά πόσο είναι σωστές ή όχι (βλ. Σχήμα 5.4).

Regulation Of Apoptosis : Experiment Simulation

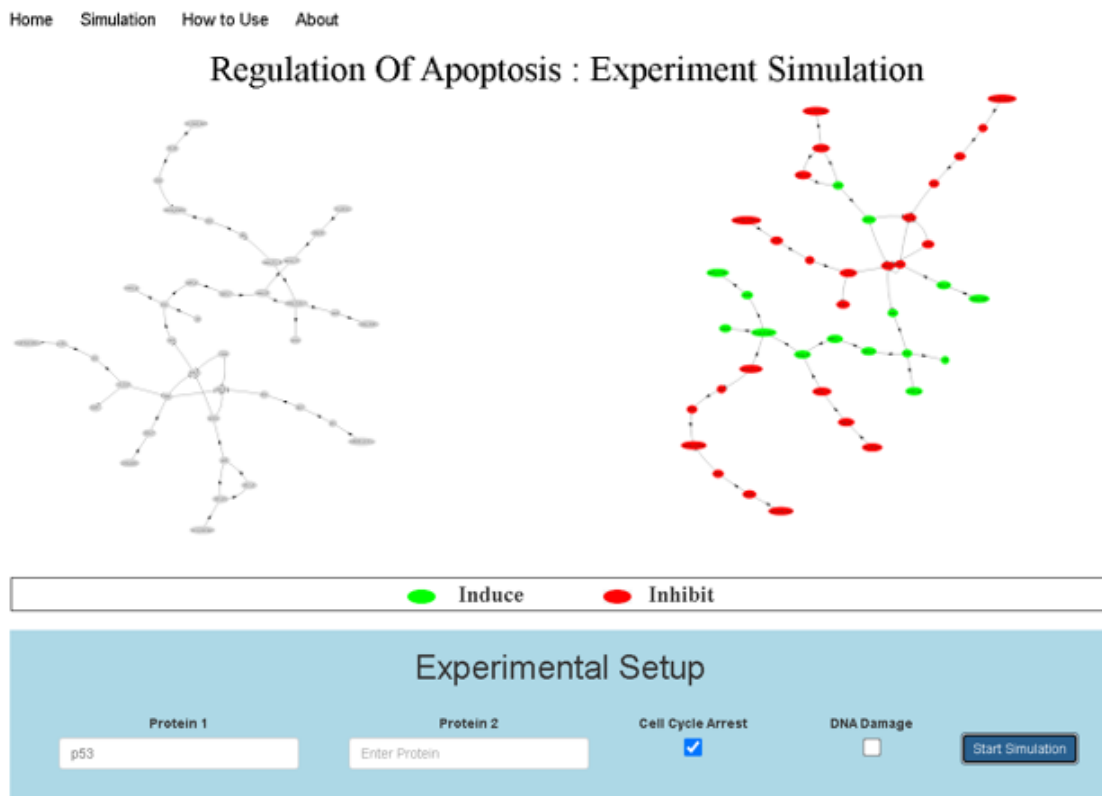


Σχήμα 5.2: Προσομοίωση πειράματος

Στο πιο πάνω σχήμα, μπορούμε να δούμε και πάλι το navigation bar με το οποίο ο χρήστης μπορεί να επισκεφθεί τις υπόλοιπες σελίδες του συστήματος. Βλέπουμε επίσης στα αριστερά το δίκτυο πρωτεϊνών του συστήματος που αποτελεί και την βάση δεδομένων των πρωτεϊνών και των μεταξύ τους συνδέσεων. Λόγω του μεγάλου όγκου δεδομένων στο δίκτυο, αυτό εμφανίζεται μικρό, αλλά ο χρήστης έχει την δυνατότητα να κάνει zoom in και zoom out χρησιμοποιώντας τον τροχό κύλισης στο ποντίκι καθώς και την δυνατότητα για να μετακινήσει τις πρωτεΐνες μέσα στο δίκτυο. Κάτω από το δίκτυο υπάρχει ένα υπόμνημα το οποίο βοηθά στην επεξήγηση του αποτελέσματος του πειράματος, το οποίο θα εμφανιστεί όταν ο χρήστης εκτελέσει το πείραμα. Στο αποτέλεσμα, όσες πρωτεΐνες ενεργοποιηθούν θα αλλάξουν χρώμα από γκριζο σε πράσινο, ενώ οι υπόλοιπες θα εμφανιστούν με κόκκινο χρώμα.

Για να εκτελέσουμε την προσομοίωση του πειράματος, ο χρήστης πρέπει να χρησιμοποιήσει το τμήμα του Experimental Setup. Χρειάζεται να γράψει ως είσοδο τουλάχιστον 1 πρωτεΐνη στο κουτί με όνομα Protein, ενώ μπορεί να δώσει ως είσοδο και μια δεύτερη πρωτεΐνη γράφοντας το όνομά της στο κουτί με όνομα Protein 2. Έχει επίσης, την επιλογή να ενεργοποιήσει κάποιο από τα 2 πλαίσια ελέγχου τα οποία

συμβολίζουν εξωτερική είσοδο στο σύστημα. Για παράδειγμα, αν ο χρήστης ενεργοποιήσει το πλαίσιο ελέγχου για DNA Damage, το σύστημα θα ενεργοποιήσει την πρωτεΐνη DNA_Damage. Αφού ο χρήστης τελειώσει με τις εισόδους του, πρέπει να πατήσει το κουμπί Start Simulation το οποίο θα εκτελέσει την προσομοίωση και θα εμφανίσει δεξιά από το αρχικό δίκτυο ένα καινούριο δίκτυο με τα αποτελέσματα του πειράματος.



Σχήμα 5.3 : Παράδειγμα εκτέλεσης μιας πειραματικής προσομοίωσης

Εκτός από την προσομοίωση ενός πειράματος, ο χρήστης έχει την δυνατότητα να δημιουργήσει παρατηρήσεις και να ελέγξει κατά πόσο είναι σωστές ή όχι. Για να εκτελέσει την προσομοίωση με τις παρατηρήσεις, ο χρήστης πρέπει αρχικά να μετακινηθεί στο τμήμα Observation Setup.

Observation

Marker	Hypothesis	Consistency Results
Please select	increase	
Please select	increase	
Please select	increase	
Please select	increase	
Please select	increase	
Please select	increase	
Please select	increase	
Please select	increase	
Please select	increase	
Please select	increase	

Observation Setup

Protein 1

Protein 2

Cell Cycle Arrest

☐

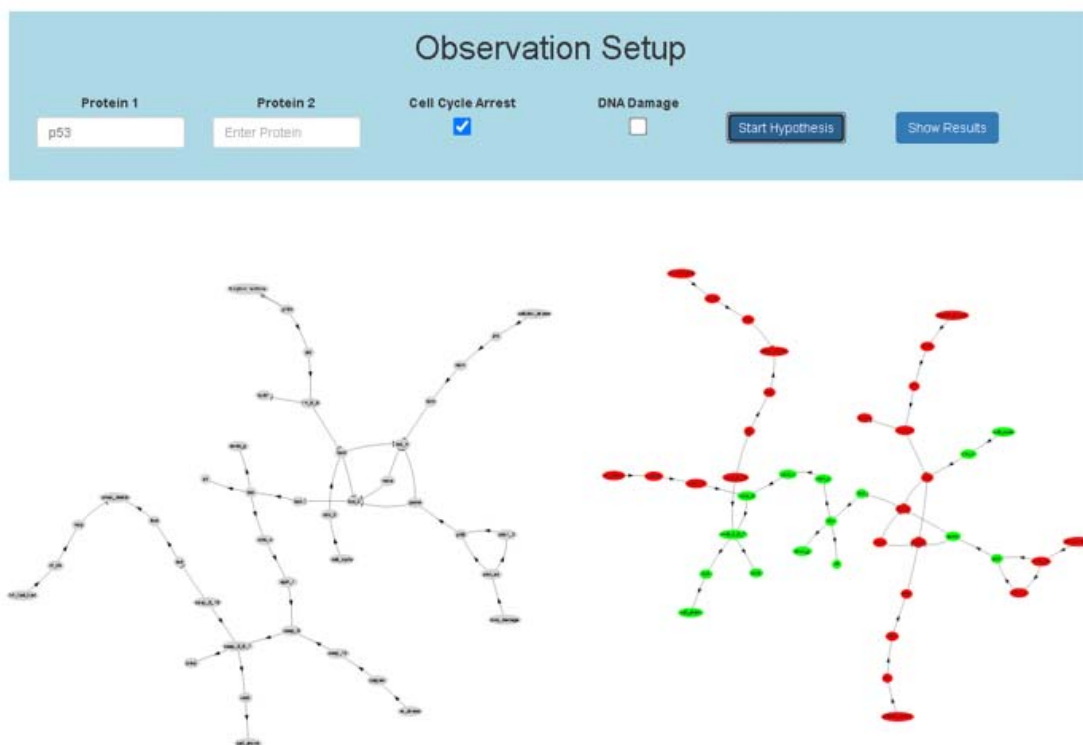
DNA Damage

☐

Σχήμα 5.4 : Διεξαγωγή προσομοίωσης πειράματος με υποθέσεις

Για να εκτελέσουμε την προσομοίωση του πειράματος, ο χρήστης πρέπει να χρησιμοποιήσει το τμήμα του Observation Setup. Όπως και πριν, ο χρήστης χρειάζεται να γράψει ως είσοδο τουλάχιστον 1 πρωτεΐνη στο κουτί με όνομα Protein, ενώ μπορεί να δώσει ως είσοδο και μια δεύτερη πρωτεΐνη γράφοντας το όνομά της στο κουτί με όνομα Protein 2. Έχει επίσης, την επιλογή να ενεργοποιήσει κάποιο από τα 2 πλαίσια ελέγχου τα οποία συμβολίζουν εξωτερική είσοδο στο σύστημα.. Αφού ο χρήστης τελειώσει με τις εισόδους του, πρέπει να πατήσει το κουμπί Start Hypothesis το οποίο

θα εκτελέσει την προσομοίωση και θα εμφανίσει δεξιά από το αρχικό δίκτυο του Observation ένα καινούριο δίκτυο με τα αποτελέσματα του πειράματος.



Σχήμα 5.5: Εκτέλεση προσομοίωσης πειράματος στο τμήμα Observation

Μετά από αυτό, ο χρήστης έχει την δυνατότητα να επιλέξει μέχρι 10 πρωτεΐνες στη στήλη Markers για τις οποίες θέλει να κάνει υποθέσεις. Διαλέγοντας μια πρωτεΐνη από το πτυσσόμενο μενού (βλ. Σχήμα 5.6), πρέπει να επιλέξει αν θέλει να υποθέσει την ενεργοποίηση της ή την απενεργοποίηση της στη στήλη Hypothesis (βλ. Σχήμα 5.7). Όταν ο χρήστης τελειώσει με την είσοδο των υποθέσεων, πρέπει να πατήσει το κουμπί Show Results το οποίο θα πραγματοποιήσει τους κατάλληλους ελέγχους και θα εμφανίσει στην κάθε υπόθεση το αποτέλεσμα της στην στήλη Consistency Results. Αν η υπόθεση είναι σωστή τότε θα εμφανιστεί το μήνυμα Correct ενώ αν είναι λάθος, θα εμφανιστεί το μήνυμα False με κόκκινα γράμματα.

Marker

Please select ▼

Please select

trophic_factors

p13k

akt

14_3_3

foxO1

bad

bcl_2

bcl_xl

puma

nox

bak

bax

bim

casp_8_10

bid

tbid

casp_3_6_7

parp

rock

Please select ▼

Please select ▼

Σχήμα 5.6 : Πτυσσόμενο μενού επιλογής πρωτεΐνης

Hypothesis

Increase ▼

Increase

Decrease

increase

Increase ▼

Increase ▼

Increase ▼

Increase ▼

Increase ▼

Increase ▼

Increase ▼

Σχήμα 5.7: Πτυσσόμενο μενού επιλογής ενεργοποίησης/απενεργοποίησης πρωτεΐνης

Observation

Marker	Hypothesis	Consistency Results
bcl_2	Increase	False
cyto_c	Decrease	False
bak	Increase	Correct
14_3_3	Decrease	Correct
puma	Increase	False

Σχήμα 5.8: Αποτελέσματα υποθέσεων στα Observations

5.2.3 How to Use

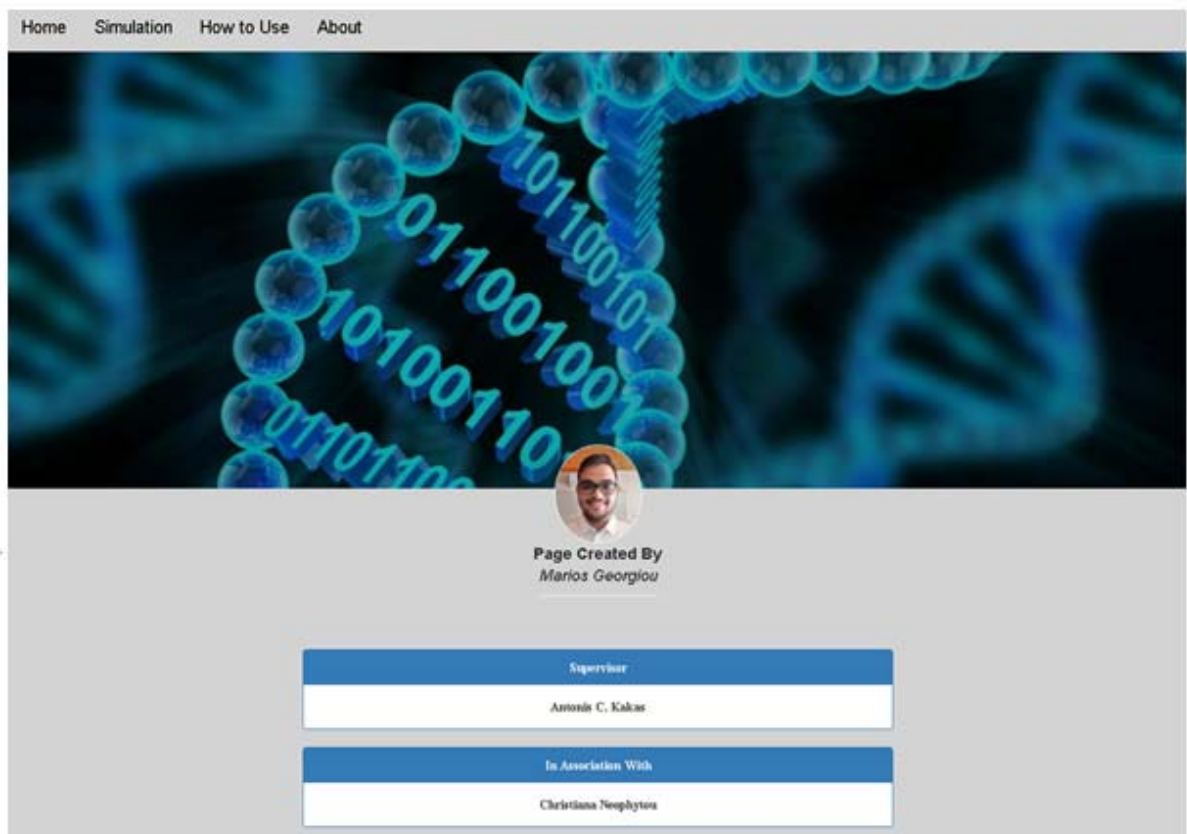
Αυτή η σελίδα αποτελεί τον οδηγό χρήσης του συστήματος στον χρήστη. Υπάρχει και πάλι το navigation bar για ευκολία μετακίνησης στις διάφορες σελίδες του συστήματος. Επίσης, παρέχει στον χρήστη όλες τις πληροφορίες που χρειάζονται για να εκτελέσει την προσομοίωση ενός πειράματος και την προσομοίωση ενός πειράματος με υποθέσεις αντίστοιχα.



Σχήμα 5.9: Οδηγός Χρήσης Συστήματος

5.2.3 About

Αυτή η σελίδα προσφέρει στον χρήστη κάποιες γενικές πληροφορίες αναφορικά με την ιστοσελίδα και το σύστημα.. Υπάρχει και πάλι το navigation bar για ευκολία μετακίνησης στις διάφορες σελίδες του συστήματος.



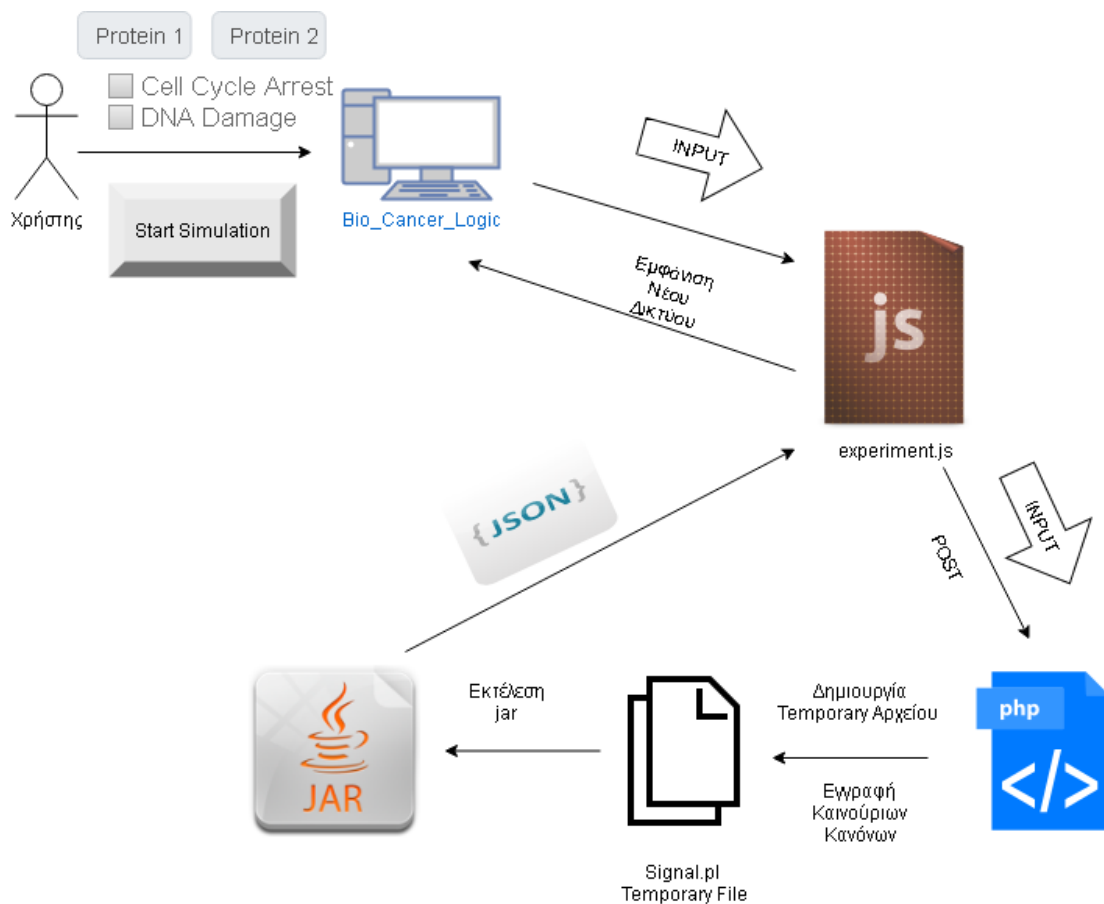
Σχήμα 5.10: Γενικές Πληροφορίες Συστήματος

5.3 Αρχιτεκτονική Συστήματος

5.3.1 Εκτέλεση Προσομοίωσης Πειράματος

Στο αρχείο sim.html χρησιμοποιώντας φόρμες παίρνουμε την είσοδο που επιθυμεί ο χρήστης. Αφού ο χρήστης τελειώσει με την είσοδο, πατά το κουμπί Start Simulation το οποίο καλεί την συνάρτηση experiment του αρχείου experiment.js, η οποία είναι υπεύθυνη για να στείλει την είσοδο με την μέθοδο POST στο αρχείο receive.php . Αυτό

το αρχείο είναι υπεύθυνο να διαβάσει την είσοδο του χρήστη με την μέθοδο GET. Αφού διαβάσει την είσοδο, δημιουργούμε ένα προσωρινό αντίγραφο του αρχείου με τους κανόνες μας μέσα στο οποίο προσθέτουμε καινούριους κανόνες ανάλογα με την είσοδο του χρήστη. Έπειτα μέσω της γραμμής εντολών εκτελείται το αρχείο ADE.jar το οποίο τρέχει με java το αρχείο στην prolog και επιστρέφει ένα αποτέλεσμα JSON. Τέλος, γίνεται μια επεξεργασία του αποτελέσματος και αλλάζουν τα χρώματα στις κατάλληλες πρωτεΐνες εμφανίζοντας και το δίκτυο.

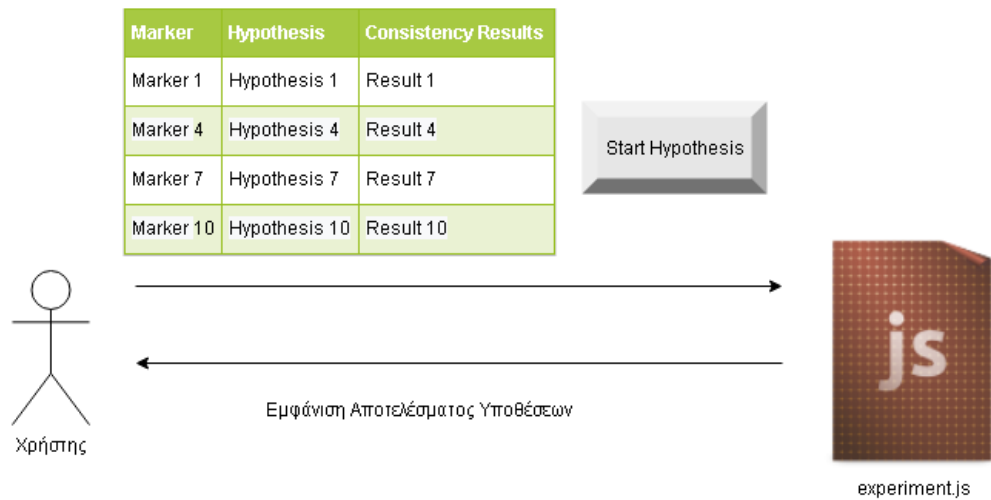
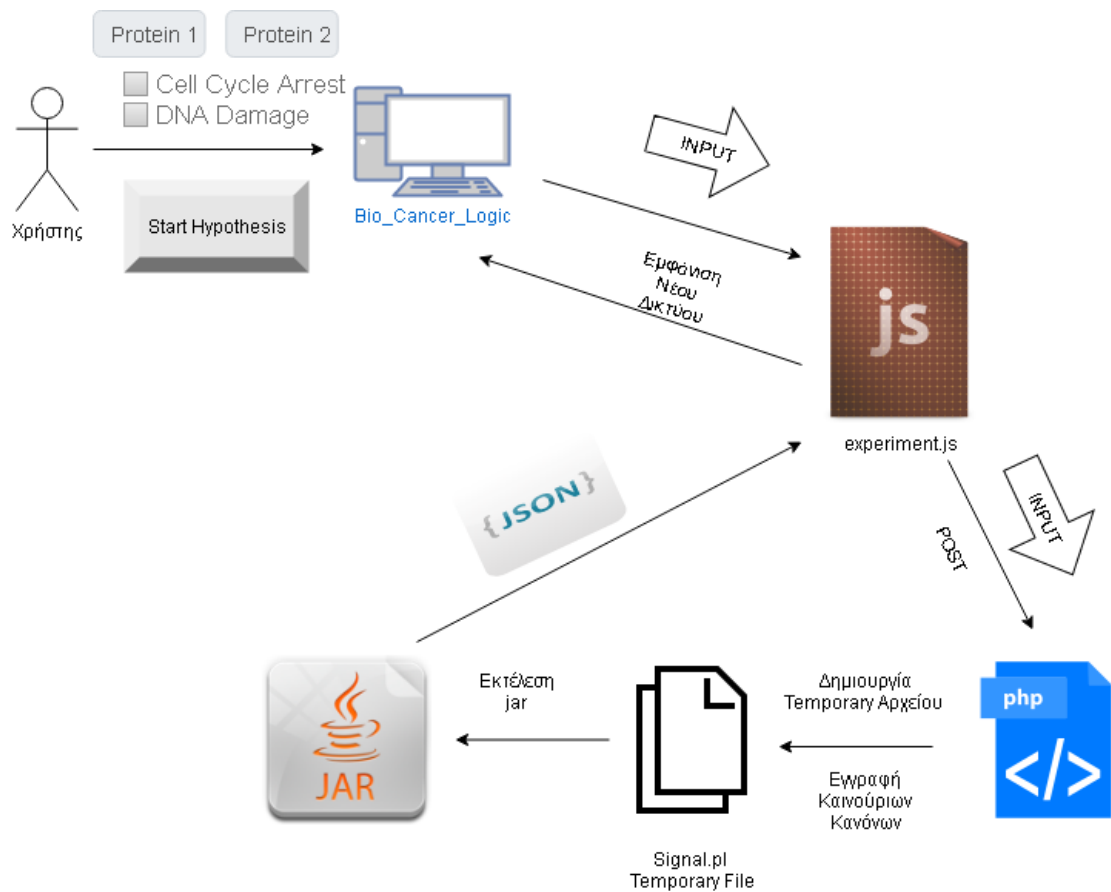


Σχήμα 5.11: Αρχιτεκτονική Συστήματος - Εκτέλεση Προσομοίωσης Πειράματος

5.3.2 Εκτέλεση Προσομοίωσης Πειράματος με Υποθέσεις

Ακολουθείται η ίδια αρχιτεκτονική με την εκτέλεση προσομοίωσης πειράματος για να πάρουμε το αποτέλεσμα JSON. Στο αρχείο `sim.html` χρησιμοποιώντας φόρμες παίρνουμε την είσοδο που επιθυμεί ο χρήστης. Αφού ο χρήστης τελειώσει με την είσοδο, πατά το κουμπί `Start Hypothesis` το οποίο καλεί την συνάρτηση `hypothesis` του αρχείου `experiment.js`, η οποία είναι υπεύθυνη για να στείλει την είσοδο με την μέθοδο `POST` στο αρχείο `receive.php`. Αυτό το αρχείο είναι υπεύθυνο να διαβάσει την είσοδο του χρήστη με την μέθοδο `GET`. Αφού διαβάσει την είσοδο, δημιουργούμε ένα προσωρινό αντίγραφο του αρχείου με τους κανόνες μας μέσα στο οποίο προσθέτουμε καινούριους κανόνες ανάλογα με την είσοδο του χρήστη. Έπειτα μέσω της γραμμής εντολών εκτελείται το αρχείο `ADE.jar` το οποίο τρέχει με `java` το αρχείο στην `prolog` και επιστρέφει ένα αποτέλεσμα JSON. Τέλος, γίνεται μια επεξεργασία του αποτελέσματος και αλλάζουν τα χρώματα στις κατάλληλες πρωτεΐνες εμφανίζοντας και το δίκτυο.

Όταν πάρουμε το αποτέλεσμα JSON μέσω της συνάρτησης `results` του αρχείου `test.js`, ελέγχουμε όλες τις υποθέσεις που έχει κάνει ο χρήστης και μετά από επεξεργασία εμφανίζεται το μήνυμα `Correct` αν η υπόθεση είναι σωστή ενώ αν είναι λάθος, θα εμφανιστεί το μήνυμα `False` με κόκκινα γράμματα.



Σχήμα 5.12: Αρχιτεκτονική Συστήματος - Εκτέλεση Προσομοίωσης Πειράματος με Υποθέσεις

Κεφάλαιο 6

Τελική Αξιολόγηση Συστήματος

6.1 Εισαγωγή	37
6.2 Αξιολόγηση βάση Πειραματικών Δεδομένων	37

6.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα δούμε την τελική αξιολόγηση του συστήματος μέσω των πειραματικών δεδομένων και ενός online ερωτηματολογίου.

6.2 Αξιολόγηση βάση Πειραματικών Δεδομένων

Για να μπορέσουμε να κάνουμε αξιολόγηση του συστήματος μας βάση πειραματικών δεδομένων, ζητήσαμε από τους ερευνητές της Βιολογίας να μας στείλουν κάποια πειραματικά δεδομένα από πειράματα που έχουν διεξαχθεί στα εργαστήρια.

Experiment 1: Effect of Gallic acid on breast cancer cells

Hypothesis: p53, Cell cycle arrest

Observation: bax increase, caspase 9,7 increase, parp increase, AIF increase

bax	Increase	Correct
casp_9	Increase	Correct
casp_3_6_7	Increase	Correct
parp	Increase	Correct
aif	Increase	Correct
Please select	Increase	
Please select	Increase	
Please select	Increase	

Observation Setup

Protein

Protein

Cell Cycle Arrest

☒

DNA Damage

☐

Start Hypothesis

Show Results

Σχήμα 6.1 : Αποτελέσματα πρώτου πειράματος με πειραματικά δεδομένα

Experiment 2: Effect of combination of Cisplatin and YM155 on lung cancer cells

Hypothesis: p53, DNA damage

Observation: Bcl-2 decrease, Bax increase, Endo G increase, p53 increase, parp increase, bak increase, bcl-xl decrease

bcl_2	Decrease	Correct
bax	Increase	Correct
endo_g	Increase	Correct
p53	Increase	Correct
parp	Increase	Correct
bak	Increase	Correct
bcl_xl	Decrease	Correct

Observation Setup

Protein

Protein

Cell Cycle Arrest

☐

DNA Damage

☒

Start Hypothesis

Show Results

Σχήμα 6.2 : Αποτελέσματα δεύτερου πειράματος με πειραματικά δεδομένα

Από τα 2 πειράματα και τα πειραματικά δεδομένα που μας δόθηκαν από τους ερευνητές της Βιολογίας, μετά από την εκτέλεση των ίδιων 2 πειραμάτων στο σύστημα μας, βλέπουμε ότι και τα δικά μας αποτελέσματα είναι τα ίδια και ότι είναι επιβεβαιώνονται όλες οι υποθέσεις που κάναμε.

Κεφάλαιο 7

Συμπεράσματα

7.1 Εισαγωγή	40
7.2 Συμπεράσματα	40
7.3 Μελλοντικές Αναβαθμίσεις	41

7.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα κάνουμε μια σύνοψη γράφοντας τα συμπεράσματά μας και αναφέροντας κάποιες πιθανές μελλοντικές αναβαθμίσεις στο σύστημά μας για να το κάνουμε καλύτερο.

7.2 Συμπεράσματα

Η παρούσα διπλωματική εργασία είχε σκοπό την μοντελοποίηση του ενδοκυτταρικού περιβάλλοντος ούτως ώστε να βοηθήσουμε στην προσομοίωση εργαστηριακών πειραμάτων. Καταφέραμε να αναπαραστήσουμε το δίκτυο πρωτεϊνών και βλέποντας και αναλύοντας την αξιολόγηση που έγινε στο πιο πάνω κεφάλαιο, μπορούμε να καταλήξουμε στο συμπέρασμα ότι οι ερευνητές του τμήματος της Βιολογίας, θα μπορούν επιτυχώς να διεξάγουν πειράματα *in silico*.

7.3 Μελλοντικές Αναβαθμίσεις

Αν και καταφέραμε να υλοποιήσουμε το σύστημα και να έχουμε τα επιθυμητά αποτελέσματα, μπορούμε να βρούμε κάποιες πιθανές μελλοντικές αναβαθμίσεις οι οποίες θα κάνουν το σύστημά μας ακόμα καλύτερο.

Αρχικά θα μπορούσαμε να υλοποιήσουμε και να ενώσουμε μια βάση δεδομένων στην οποία θα μπορούσαμε να αποθηκεύουμε τα αποτελέσματά από τα πειράματά μας ούτως ώστε να έχουμε ευκολότερη πρόσβαση στα αποτελέσματα χωρίς να χρειαστεί να τρέχουμε τα ίδια πειράματα περισσότερο από 1 φορές .

Τέλος, λόγω υπέρβασης του χώρου μνήμης στο stack χρειάστηκε να δώσαμε περισσότερη έμφαση σε κάποιες πρωτεΐνες και δεν καταφέραμε να τις συμπεριλάβουμε όλες. Πιστεύω ότι θα ήταν καλό να είχε βρεθεί μια λύση για το πρόβλημα αυτό ούτως ώστε να μπορούσαμε να συμπεριλάβουμε όλες τις πρωτεΐνες στο δίκτυο μας και να έχουμε ολοκληρωμένο το δίκτυο των πρωτεϊνών μας.

Βιβλιογραφία

- [1] About: Apoptosis
<https://dbpedia.org/page/Apoptosis>
- [2] Alireza Tamaddoni-Nezhad, Antonis Kakas, Stephen Muggleton, and Florencio Pazos “Modelling Inhibition in Metabolic Pathways Through Abduction and Induction”
- [3] An introduction to death receptors in apoptosis
<https://www.sciencedirect.com/science/article/pii/S1743919105000282#:~:text=Mechanisms%20of%20apoptosis&text=The%20extrinsic%20pathway%20involves%20the,turn%20initiates%20the%20caspase%20cascade.&text=Death%20receptors%20are%20part%20of,and%20efficient%20route%20to%20apoptosis.>
- [4] Blaz Zupan, Janez Demsar, Ivan Bratko, Peter Juhan, John A. Halter , Adam Kuspa and Gad Shaulsky “GenePath: a system for automated construction of genetic networks from mutant data”
- [5] Logic Programming
<https://www.virtusa.com/digital-themes/logic-programming>
- [6] Melody K. Morris, Julio Saez-Rodriguez, Peter K. Sorger and Douglas A. Lauffenburger “Logic-Based Models for the Analysis of Cell Signaling Networks”
- [7] Note on the origin and history of the term “apoptosis”
[https://anatomypubs.onlinelibrary.wiley.com/doi/full/10.1002/ar.b.20047#:~:text=Apoptosis%20is%20the%20term%20that,Navarro%E2%80%90Beltr%C3%A1n%2C%201999\).&text=Glucksmann%20initially%20discovered%20that%20cell,1997%3B%20Oppenheim%2C%201999\).](https://anatomypubs.onlinelibrary.wiley.com/doi/full/10.1002/ar.b.20047#:~:text=Apoptosis%20is%20the%20term%20that,Navarro%E2%80%90Beltr%C3%A1n%2C%201999).&text=Glucksmann%20initially%20discovered%20that%20cell,1997%3B%20Oppenheim%2C%201999).)

- [8] Oliver Ray, Antonis Kakas “ProLogICA: a practical system for Abductive Logic Programming”
- [9] The Modulation of Apoptotic Pathways by Gammaherpesviruses
<https://www.frontiersin.org/articles/10.3389/fmicb.2016.00585/full>
- [10] Understanding Apoptosis and Apoptotic Pathways Targeted Cancer Therapeutics
<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6664112/>
- [11] What is Logic?
<https://press.rebus.community/intro-to-phil-logic/chapter/chapter-1/>
- [12] What is the difference between necrosis and apoptosis?
<https://www.ptglab.com/news/blog/what-is-the-difference-between-necrosis-and-apoptosis/>
- [13] Wikipedia Website, Apoptosis
<https://en.wikipedia.org/wiki/Apoptosis>
- [14] Wikipedia Website, Bioinformatics
<https://en.wikipedia.org/wiki/Bioinformatics>
- [15] Wikipedia Website, Prolog
<https://en.wikipedia.org/wiki/Prolog>
- [16] Website for Bio_Cancer Logic
http://thesis.in.cs.ucy.ac.cy/Bio_Cancer_Logic/home.html

Παράρτημα Α

Ακολουθεί κομμάτι του κώδικα του αρχείου simulation.js, το οποίο λειτουργεί σαν η βάση δεδομένων για τις πρωτεΐνες και τις σχέσεις μεταξύ τους, και δημιουργεί το δίκτυο των πρωτεϊνών.

```
var nodes = [  
  {id: 1, label: 'trophic_factors', group: 'color'},  
  {id: 2, label: 'p13k', group: 'color'},  
  {id: 3, label: 'akt', group: 'color'},  
  {id: 4, label: '14_3_3', group: 'color'},  
  {id: 5, label: 'fox01', group: 'color'},  
  {id: 6, label: 'bad', group: 'color'},  
  {id: 7, label: 'bcl_2', group: 'color'},  
  {id: 8, label: 'bcl_xl', group: 'color'},  
  {id: 9, label: 'puma', group: 'color'},  
  {id: 10, label: 'nox', group: 'color'},  
  {id: 11, label: 'bak', group: 'color'},  
  {id: 12, label: 'bax', group: 'color'},  
  {id: 13, label: 'bim', group: 'color'},  
  {id: 14, label: 'casp_8_10', group: 'color'},  
  {id: 15, label: 'bid', group: 'color'},  
  {id: 16, label: 'tbid', group: 'color'},  
  {id: 17, label: 'casp_3_6_7', group: 'color'},  
  {id: 18, label: 'parp', group: 'color'},  
  {id: 19, label: 'rock', group: 'color'},  
  {id: 20, label: 'cell_shrink', group: 'color'},  
  {id: 21, label: 'er_stress', group: 'color'},  
  {id: 22, label: 'calpain', group: 'color'},  
  {id: 23, label: 'casp_12', group: 'color'},  
  {id: 24, label: 'casp_9', group: 'color'},  
  {id: 25, label: 'tnf_fas_l_trail', group: 'color'},
```

```

{id: 26, label: 'nf_kb', group: 'color'},
{id: 27, label: 'cellular_stress', group: 'color'},
{id: 28, label: 'jnk', group: 'color'},
{id: 29, label: 'cjun', group: 'color'},
{id: 30, label: 'dna_damage', group: 'color'},
{id: 31, label: 'atm_atr', group: 'color'},
{id: 32, label: 'chk1_2', group: 'color'},
{id: 33, label: 'p53', group: 'color'},
{id: 34, label: 'cyto_c', group: 'color'},
{id: 35, label: 'aif', group: 'color'},
{id: 36, label: 'endo_g', group: 'color'},
{id: 37, label: 'apaf_1', group: 'color'},
{id: 38, label: 'smac_diablo', group: 'color'},
{id: 39, label: 'xiap', group: 'color'},
{id: 40, label: 'cell_cycle', group: 'color'},
{id: 41, label: 'cdc_2', group: 'color'},

];

// create an array with edges
var edges = [
  {from: 1, to: 2, color: { color: 'black'},arrows: {middle: true}},
  {from: 2, to: 3, color: { color: 'black'},arrows: {middle: true}},
  {from: 3, to: 4, color: { color: 'black'},arrows: {middle: true}},
  {from: 11, to: 12, color: { color: 'black'},arrows: {middle: true}},
  {from: 15, to: 16, color: { color: 'black'},arrows: {middle: true}},
  {from: 14, to: 17, color: { color: 'black'},arrows: {middle: true}},
  {from: 17, to: 18, color: { color: 'black'},arrows: {middle: true}},
  {from: 17, to: 19, color: { color: 'black'},arrows: {middle: true}},
  {from: 19, to: 20, color: { color: 'black'},arrows: {middle: true}},
  {from: 21, to: 22, color: { color: 'black'},arrows: {middle: true}},
  {from: 22, to: 23, color: { color: 'black'},arrows: {middle: true}},
  {from: 23, to: 24, color: { color: 'black'},arrows: {middle: true}},

```

```

{from: 24, to: 17, color: { color: 'black'},arrows: {middle: true}},
{from: 25, to: 26, color: { color: 'black'},arrows: {middle: true}},
{from: 27, to: 28, color: { color: 'black'},arrows: {middle: true}},
{from: 28, to: 29, color: { color: 'black'},arrows: {middle: true}},
{from: 30, to: 31, color: { color: 'black'},arrows: {middle: true}},
{from: 31, to: 32, color: { color: 'black'},arrows: {middle: true}},
{from: 32, to: 33, color: { color: 'black'},arrows: {middle: true}},
{from: 31, to: 33, color: { color: 'black'},arrows: {middle: true}},
{from: 12, to: 34, color: { color: 'black'},arrows: {middle: true}},
{from: 12, to: 35, color: { color: 'black'},arrows: {middle: true}},
{from: 12, to: 36, color: { color: 'black'},arrows: {middle: true}},
{from: 34, to: 37, color: { color: 'black'},arrows: {middle: true}},
{from: 16, to: 38, color: { color: 'black'},arrows: {middle: true}},
{from: 37, to: 24, color: { color: 'black'},arrows: {middle: true}},
{from: 40, to: 41, color: { color: 'black'},arrows: {middle: true}},
{from: 41, to: 6, color: { color: 'black'},arrows: {middle: true}},
{from: 33, to: 9, color: { color: 'black'},arrows: {middle: true}},
{from: 29, to: 13, color: { color: 'black'},arrows: {middle: true}},
{from: 26, to: 39, color: { color: 'black'},arrows: {middle: true}},
{from: 4, to: 5, color: { color: 'black'},arrows: {to: true}},
{from: 4, to: 6, color: { color: 'black'},arrows: {to: true}},
{from: 6, to: 7, color: { color: 'black'},arrows: {to: true}},
{from: 6, to: 8, color: { color: 'black'},arrows: {to: true}},
{from: 9, to: 7, color: { color: 'black'},arrows: {to: true}},
{from: 10, to: 7, color: { color: 'black'},arrows: {to: true}},
{from: 7, to: 11, color: { color: 'black'},arrows: {to: true}},
{from: 9, to: 8, color: { color: 'black'},arrows: {to: true}},
{from: 10, to: 8, color: { color: 'black'},arrows: {to: true}},
{from: 13, to: 8, color: { color: 'black'},arrows: {to: true}},
{from: 14, to: 15, color: { color: 'black'},arrows: {to: true}},
{from: 39, to: 38, color: { color: 'black'},arrows: {to: true}},

```

];

```

// create a network
var container = document.getElementById('mynetwork');

var data= {
  nodes: nodes,
  edges: edges,
}
var options = {

  width: '670px',
  height: '600px',

  /*  nodes: {
    fixed:true,
    },*/ //this part makes the nodes fixed position

  edges:{
    arrows:{
      to:{
        type:'bar'
      }
    }
  },
  groups: {

    color: {
      color:'lightgrey',
    },
    color2:{
      color:'lime',
    },
    color3:{
      color:'red',

```

```
    },  
  },  
}  
var network = new vis.Network(container, data, options);
```

Παράρτημα Β

Ακολουθεί ο κώδικας του αρχείου receive.php, το οποίο παίζει σημαντικό ρόλο στην επιτυχή διεξαγωγή των πειραμάτων. Με αυτό το αρχείο γίνεται η σύνδεση της Java με την Prolog για να δημιουργηθεί το αποτέλεσμα.

```
<?php
error_reporting(E_ALL);

if(isset( $_REQUEST["protein"])&&!empty($_REQUEST["protein"])){
    $protein = $_REQUEST["protein"];
    $protein_bool = $_REQUEST["prot1_bool"];
    $protein2 = $_REQUEST["protein2"];
    $protein2_bool = $_REQUEST["prot2_bool"];
    $cca = $_REQUEST["cca"];
    $dna = $_REQUEST["dna"];

    $PrologFilePath = "./signal.pl";
    $PrologFile = file_get_contents($PrologFilePath);
    $tempFile = tempnam("./tempFiles", "temp");
    $handleStoryFile = fopen($tempFile, "w");
    fwrite($handleStoryFile, $PrologFile . PHP_EOL);

    if(($protein!="bcl_2")&&($protein2!="bcl_2")) {
        $input6 = "signal(bak).";
        fwrite($handleStoryFile, $input6 . PHP_EOL);
    }

    if($protein_bool) {
        $input = "signal(" . $protein . ").";
        fwrite($handleStoryFile, $input . PHP_EOL);
    }
}
```

```

        if($protein2_bool) {
            $input2 = "signal(" . $protein2 . ").";
            fwrite($handleStoryFile, $input2 . PHP_EOL);
        }

        if($cca) {
            $input3 = "signal(cell_cycle).";
            fwrite($handleStoryFile, $input3 . PHP_EOL);
        }

        if($dna) {
            $input4 = "signal(dna_damage).";
            fwrite($handleStoryFile, $input4 . PHP_EOL);
        }

        fclose($handleStoryFile);
        $newPrologFilePath = str_replace("\\", "/", $tempFile);
        $command='java -Djava.library.path=/usr/local/lib/swipl/lib/x86_64-linux -jar
ADE.jar '.escapeshellarg($newPrologFilePath);
        $output= shell_exec(escapeshellcmd($command));
        echo $output;
        unlink($tempFile);
    }
?>

```

Παράρτημα Γ

Ακολουθεί ο κώδικας του αρχείου signal.pl, το οποίο είναι το αρχείο της prolog στο οποίο θα γραφτούν οι νέοι κανόνες σύμφωνα με την είσοδο του χρήστη και θα τρέξει για να πάρουμε το αποτέλεσμα.

:- dynamic signal/1.

```
connected(trophic_factors,p13k).
connected(p13k,akt).
connected(akt,14_3_3).
connected(14_3_3,foxO1).
connected(14_3_3,bad).
connected(bad,bcl_2).
connected(bad,bcl_xl).
connected(puma,bcl_2).
connected(noxa,bcl_2).
connected(puma,bcl_xl).

connected(noxa,bcl_xl).
connected(bcl_2,bak).
connected(bak,bax).
connected(bim,bcl_xl).
connected(casp_8_10,bid).
connected(bid,tbid).
connected(casp_8_10,casp_3_6_7).
connected(casp_3_6_7,parp).
connected(casp_3_6_7,rock).
connected(rock,cell_shrink).
connected(er_stress,calpain).
connected(calpain,casp_12).
connected(casp_12,casp_9).
connected(casp_9,casp_3_6_7).
connected(tnf_fasl_trail,nf_kb).
connected(cellular_stress,jnk).
connected(jnk,c_jun).
connected(dna_damage,atm_atr).
connected(atm_atr,chk1_2).
connected(chk1_2,p53).
connected(atm_atr,p53).
connected(bax,cyto_c).
connected(bax,aif).
connected(bax,endo_g).
connected(cyto_c,apaf_1).
connected(tbid,smac_diablo).
```

connected(xiap,smac_diablo).
connected(apaf_1,casp_9).
connected(cell_cycle,cdc_2).
connected(cdc_2,bad).
connected(p53,puma).
connected(c_jun,bim).
connected(nf_kb,xiap).

induce(trophic_factors,p13k).
induce(p13k,akt).
induce(akt,14_3_3).
induce(bid,tbid).
induce(casp_8_10,casp_3_6_7).
induce(casp_3_6_7,parp).
induce(casp_3_6_7,rock).
induce(rock,cell_shrink).
induce(er_stress,calpain).
induce(calpain,casp_12).
induce(casp_12,casp_9).
induce(casp_9,casp_3_6_7).
induce(tnf_fasl_trail,nf_kb).
induce(cellular_stress,jnk).
induce(jnk,c_jun).
induce(dna_damage,atm_atr).
induce(atm_atr,chk1_2).
induce(chk1_2,p53).
induce(atm_atr,p53).
induce(bax,cyto_c).
induce(bax,aif).
induce(bax,endo_g).
induce(cyto_c,apaf_1).
induce(tbid,smac_diablo).
induce(apaf_1,casp_9).
induce(cell_cycle,cdc_2).
induce(cdc_2,bad).
induce(bak,bax).
induce(p53,puma).
induce(c_jun,bim).
induce(nf_kb,xiap).

inhibit(14_3_3,foxO1).
inhibit(14_3_3,bad).
inhibit(bad,bcl_2).
inhibit(bad,bcl_xl).
inhibit(puma,bcl_2).
inhibit(noxa,bcl_2).
inhibit(puma,bcl_xl).
inhibit(noxa,bcl_xl).
inhibit(bcl_2,bak).

```
inhibit(bim,bcl_xl).  
inhibit(casp_8_10,bid).  
inhibit(xiap,smac_diablo).
```

```
not_signal(Y):-connected(X,Y),inhibit(X,Y).
```

```
signal(Y):-connected(X,Y),signal(X),induce(X,Y),not(not_signal(Y)).
```

Παράρτημα Δ

Ακολουθεί ο κώδικας του αρχείου ADE2.java, με βάση το οποίο δημιουργούμε το ADE.jar. Σε αυτό το αρχείο παίρνουμε ως είσοδο ένα αρχείο prolog και το τρέχουμε για να πάρουμε τα αποτελέσματα.

```
import java.io.File;
import java.io.FileWriter;
import java.io.IOException;
import java.util.Map;
import org.jpl7.Query;
import org.jpl7.Term;
import org.json.simple.JSONObject;

public class ADE2 {
    @SuppressWarnings("unchecked")
    public static JSONObject executeProlog(String filename) {

        JSONObject resultJson = new JSONObject();

        String prologConsult = "consult(\"" + filename + "\").";
        Query q = new Query(prologConsult);

        if (q.hasSolution()) {
            String prologQuery = "signal(X)";
            Query q2 = new Query(prologQuery);
            if (q2.hasSolution()) {
                while (q2.hasNext()) {
                    Map<String, Term> result = q2.next();
                    String varX = result.get("X").toString();
                    resultJson.put(varX, varX);
                }
            }
        }
        return resultJson;
    }

    public static void main(String[] args) throws IOException {

        String myFile = args[0];
        final JSONObject resultJson = executeProlog(myFile);
        System.out.print(resultJson);
    }
}
```