



ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

**Σχεδίαση και Ανάπτυξη
ενός Γραφικού Εκπαιδευτικού Εργαλείου
για την Δημιουργία και Προσομοίωση Μηχανών Turing**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΠΑΡΑΣΚΕΥΑ ΘΕΟΦΑΝΗ ΚΟΝΤΟΥΛΗ

Εξεταστική Επιτροπή

Καθηγητής Μιχαήλ Γ. Λαγουδάκης

Καθηγητής Απόστολος Δόλλας

Καθηγήτρια Αικατερίνη Μανιά

Χανιά, Μάρτιος 2023

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 1η Μαρτίου 2023

Μ. Γ. Λαγουδάκης
Καθηγητής Π.Κ.

(Υπογραφή)

.....

Α. Δόλλας
Καθηγητής Π.Κ.

(Υπογραφή)

.....

Α. Μανιά
Καθηγήτρια Π.Κ.

(Υπογραφή)

.....

Χανιά, Μάρτιος 2023

ΠΑΡΑΣΚΕΥΑΣ ΘΕΟΦΑΝΗΣ ΚΟΝΤΟΥΛΗΣ

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Π.Κ

(Υπογραφή)

.....

Copyright © Παρασκευάς Θεοφάνης Κοντούλης, 2023

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Πολυτεχνείου Κρήτης.



Technical University Of Crete
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING
Division of Informatics

**Design and Development of a Graphical Educational Tool
for the Creation and Simulation of Turing Machines**

Diploma Thesis

by

Paraskevas Theofanis Kontoulis

Examination Committee

Professor Michail G. Lagoudakis

Professor Apostolos Dollas

Professor Aikaterini Mania

Chania, March 2023

Περίληψη

Η Μηχανή Turing είναι μια θεωρητική μηχανή που εφευρέθηκε από τον Alan Turing για χρήση ως μοντέλο σε μαθηματικούς υπολογισμούς. Θεωρείται ακρογωνιαίος λίθος της θεωρητικής επιστήμης των υπολογιστών και είχε βαθύ αντίκτυπο στην ανάπτυξη της επιστήμης των υπολογιστών και της τεχνητής νοημοσύνης. Πρόκειται για μια απλή μηχανή που μπορεί να εκτελέσει οποιονδήποτε υπολογισμό που μπορεί να εκφραστεί ως αλγόριθμος. Παρά την απλότητά της, έχει αποδειχθεί ότι η Μηχανή Turing είναι υπολογιστικά ισοδύναμη με οποιοδήποτε άλλο παρόμοιο μοντέλο υπολογισμού, καθιστώντας την ένα ισχυρό εργαλείο για την κατανόηση των θεμελίων του υπολογισμού. Τα τελευταία χρόνια, υπάρχει αυξανόμενο ενδιαφέρον για τη χρήση των Μηχανών Turing ως εργαλείο διδασκαλίας για την επιστήμη των υπολογιστών, τη θεωρία υπολογισμού και τα μαθηματικά. Μία από τις προκλήσεις της διδασκαλίας σχετικά με τις Μηχανές Turing είναι ότι παρουσιάζονται ως νοητά κουτιά, που καθιστούν δύσκολη την κατανόηση της λειτουργίας τους από τους φοιτητές. Για την αντιμετώπιση αυτής της πρόκλησης, υπήρξε μια αυξανόμενη ανάγκη για εκπαιδευτικά εργαλεία που διευκολύνουν την κατανόηση και τη διερεύνηση της συμπεριφοράς των Μηχανών Turing. Η παρούσα διπλωματική εργασία παρουσιάζει την ανάπτυξη μιας διαδικτυακής εφαρμογής που επιτρέπει στο χρήστη να σχεδιάζει και να προσομοιώνει Μηχανές Turing. Ο στόχος της εφαρμογής είναι να παρέχει σε φοιτητές και εκπαιδευτικούς έναν διαδραστικό και προσιτό τρόπο για να εξερευνήσουν τη συμπεριφορά των Μηχανών Turing και να κατανοήσουν τις θεμελιώδεις έννοιες της Θεωρίας Υπολογισμού. Η εφαρμογή επιτρέπει στους χρήστες να δημιουργούν και να προσομοιώνουν Μηχανές Turing, χρησιμοποιώντας την σημειογραφία και γραφική αναπαράσταση των Lewis και Παπαδημητρίου [1], να οπτικοποιούν τη συμπεριφορά της μηχανής και να πειραματίζονται με διαφορετικούς αλγόριθμους. Η εφαρμογή αναπτύχθηκε σε Typescript και έχει σχεδιαστεί για να είναι απλή και φιλική προς τον χρήστη, καθιστώντας την προσβάσιμη σε φοιτητές όλων των υποβάθρων και επιπέδων δεξιοτήτων. Παρέχοντας έναν διαδραστικό και προσιτό τρόπο εξερεύνησης των Μηχανών Turing, αυτή η εφαρμογή έχει τη δυνατότητα να βελτιώσει σημαντικά τον τρόπο με τον οποίο οι φοιτητές μαθαίνουν για την επιστήμη των υπολογιστών.

Abstract

A Turing Machine is a theoretical machine invented by Alan Turing for use as a model in mathematical calculations. It is considered a cornerstone of theoretical computer science and had a profound impact on the development of computer science and artificial intelligence. It is a simple machine that can perform any calculation that can be expressed as an algorithm. Despite its simplicity, the Turing Machine has been shown to be computationally equivalent to any other similar model of computation, making it a powerful tool for understanding the fundamentals of computation. In recent years, there has been growing interest in using Turing Machines as a teaching tool for computer science, theory of computation, and mathematics. One of the challenges of teaching about Turing Machines is that they are presented as theoretical boxes, making it difficult for students to understand how they work. To address this challenge, there has been a growing need for educational tools that facilitate understanding and investigating the behavior of Turing Machines. This thesis presents the development of a web application that allows the user to design and simulate Turing Machines. The goal of the app is to provide students and educators with an interactive and accessible way to explore the behavior of Turing Machines and understand the fundamental concepts of Theory of Computation. The application allows users to create and simulate Turing machines, using Lewis and Papadimitriou's notation and graphical representation, visualize the machine's behavior, and experiment with different algorithms. The application was developed in Typescript and is designed to be simple and user-friendly, making it accessible to students of all backgrounds and skill levels. By providing an interactive and accessible way to explore Turing machines, this app has the potential to significantly improve the way students learn about computer science.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω ιδιαίτερα τον Καθηγητή Μιχαήλ Λαγουδάκη και τον κύριο Νεκτάριο Μουμουτζή για τις πολύτιμες υποδείξεις και τη καθοδήγηση κατά τη διάρκεια της εκπόνησης της διπλωματικής μου εργασίας.

Ακόμη οφείλω ένα μεγάλο ευχαριστώ στους γονείς μου και στα αδέρφια μου για την αμέριστη υποστήριξη τους καθ' όλη την διάρκεια των σπουδών μου.

Πίνακας Περιεχομένων

Περίληψη	3
Abstract	4
Ευχαριστίες	5
Πίνακας περιεχομένων	6
Λίστα εικόνων	7
1. Εισαγωγή	9
1.1 Κίνητρα	9
1.2 Συνεισφορά	10
1.3 Περίγραμμα	10
2. Θεωρητικό Υπόβαθρο	12
2.1. Θεωρία Υπολογισμού	12
2.2. Μηχανές Turing	12
2.2.1. Ορισμός Μηχανής Turing	12
2.2.2. Γραφική Απεικόνιση Μηχανής Turing	15
2.3. Εργαλεία Ανάπτυξης Εφαρμογών	16
3. Περιγραφή Προβλήματος	20
3.1. Σχετικές Εργασίες	21
4. Προσέγγιση Σχεδιασμού και Υλοποίηση	25
4.1. Υλοποίηση Εφαρμογής	25
4.2. Προσέγγιση Σχεδιασμού	27
5. Αποτελέσματα	30
5.1 Προσομοίωση Μηχανής Turing	30
5.2 Δημιουργία Μηχανής Turing	34
6. Εμπειρία Χρηστών	49
6.1. Ερωτηματολόγιο	49
6.2. Αποτελέσματα	52
7. Συμπεράσματα	55
7.1. Συζήτηση	55
7.2. Μελλοντικές Αναβαθμίσεις	56
7.3. Μαθήματα	57
8. Βιβλιογραφία	58
ΠΑΡΑΡΤΗΜΑ Α'	60

Λίστα Εικόνων

<i>Εικόνα 2.1</i> Απεικόνιση Μηχανής Turing	14
<i>Εικόνα 2.2</i> Συνάρτηση μετάβασης παραδείγματος	15
<i>Εικόνα 2.3</i> Παράδειγμα γραφικής απεικόνισης Μηχανής Turing	15
<i>Εικόνα 3.1</i> Σχετική Εργασία Νο1	22
<i>Εικόνα 3.2</i> Σχετική Εργασία Νο2	23
<i>Εικόνα 3.3</i> Σχετική Εργασία Νο3	24
<i>Εικόνα 4.1</i> Η αρχιτεκτονική της εφαρμογής	29
<i>Εικόνα 5.1</i> Αρχική σελίδα εφαρμογής	30
<i>Εικόνα 5.2</i> Παράδειγμα έναρξης Μηχανής Turing	31
<i>Εικόνα 5.3</i> Παράδειγμα τερματισμού Μηχανής Turing	32
<i>Εικόνα 5.4</i> Παράδειγμα διαχείρισης ταινίας κεφαλής και αλφαβήτου Μηχανής Turing	32
<i>Εικόνα 5.5</i> Παράδειγμα διαχείρισης ενός node	33
<i>Εικόνα 5.6</i> Παράδειγμα δημιουργία μετάβασης	33
<i>Εικόνα 5.7</i> Αρχική οθόνη	34
<i>Εικόνα 5.8</i> Δημιουργία node	34
<i>Εικόνα 5.9</i> Μετατροπή σε αρχική βασική μηχανή	35
<i>Εικόνα 5.10</i> Προσθήκη μηχανής εύρεσης πρώτου κενού δεξιά	35
<i>Εικόνα 5.11</i> Πρώτο node	36
<i>Εικόνα 5.12</i> Δημιουργία δεύτερου node	36
<i>Εικόνα 5.13</i> Αποτέλεσμα στον καμβά	37
<i>Εικόνα 5.14</i> Προσθήκη συμβόλου 0	37
<i>Εικόνα 5.15</i> Προσθήκη συμβόλου 1	38
<i>Εικόνα 5.16</i> Αποτέλεσμα στο αλφάβητο	38
<i>Εικόνα 5.17</i> Προσθήκη τρίτου node	39
<i>Εικόνα 5.18</i> Προσθήκη τέταρτου node	39
<i>Εικόνα 5.19</i> Σύνολο των node	40
<i>Εικόνα 5.20</i> Προσθήκη μετάβασης	40
<i>Εικόνα 5.21</i> Επιλογή δεύτερου node	41

Εικόνα 5.22 Αποτέλεσμα προσθήκης	41
Εικόνα 5.23 Αποτέλεσμα στον καμβά μετά την προσθήκη	42
Εικόνα 5.24 Προσθήκη επιπλέον μετάβασης	42
Εικόνα 5.25 Επιλογή τέταρτου node	43
Εικόνα 5.26 Αποτέλεσμα στον καμβά μετά την προσθήκη των μεταβάσεων μέχρι τώρα	43
Εικόνα 5.27 Επιλογή δεύτερου node και προσθήκη μετάβασης	44
Εικόνα 5.28 Τελική Μηχανή Turing	44
Εικόνα 5.29 Δεδομένα node no1	45
Εικόνα 5.30 Δεδομένα node no3	45
Εικόνα 5.31 Δεδομένα node no4	46
Εικόνα 5.32 Εκκίνηση Μηχανής Turing	47
Εικόνα 5.33 Τερματισμός Μηχανής Turing	47
Εικόνα 6.1 Γραφικό διάγραμμα παραγόντων	50
Εικόνα 6.2 Ερωτηματολόγιο	51
Εικόνα 6.3 Αποτελέσματα αξιολόγησης	52
Εικόνα 6.4 Διάγραμμα αποτελεσμάτων Νο1	53
Εικόνα 6.5 Συνοπτικός πίνακας αποτελεσμάτων	54
Εικόνα 6.6 Αναλυτικός πίνακας αποτελεσμάτων	54
Εικόνα 6.7 Αναλυτικό διάγραμμα αποτελεσμάτων	54

1. *Εισαγωγή*

Η Θεωρία Υπολογισμού είναι ένας κλάδος της θεωρητικής επιστήμης των υπολογιστών που ασχολείται με τη μελέτη αλγορίθμων και την υπολογιστική πολυπλοκότητά τους. Διερευνά τα όρια του τι μπορεί να υπολογιστεί και πόσο αποτελεσματικά μπορεί να γίνει. Μία από τις κεντρικές έννοιες στη Θεωρία Υπολογισμού είναι η Μηχανή Turing, ένα μαθηματικό μοντέλο μιας υποθετικής μηχανής που μπορεί να εκτελέσει οποιονδήποτε υπολογισμό που είναι αλγοριθμικά διατυπωμένος. Η Μηχανή Turing εισήχθη για πρώτη φορά από τον Alan Turing το 1936 [0] και έκτοτε έχει γίνει η βάση της επιστήμης των υπολογιστών. Είναι ιδιαίτερα σημαντική και συμβάλει στην κατανόηση της έννοιας της υπολογισιμότητας και των ορίων της υπολογισιμότητας. Η μελέτη των Μηχανών Turing και η Θεωρία Υπολογισμού έχουν εκτεταμένες επιπτώσεις, συμπεριλαμβανομένων των θεμελίων της επιστήμης των υπολογιστών, στον σχεδιασμό αλγορίθμων και στην ανάλυση της υπολογιστικής πολυπλοκότητας. Η παρούσα εργασία έχει σκοπό να παρουσιάσει την υλοποίηση ενός γραφικού εργαλείου σχεδίασης και προσομοίωσης Μηχανών Turing στο διαδίκτυο και το σκεπτικό γύρω από την υλοποίηση αυτή.

1.1 Κίνητρα

Το κίνητρο για τη δημιουργία αυτής της διαδικτυακής εφαρμογής που προσομοιώνει Μηχανές Turing ήταν η έλλειψη διαθέσιμων προσομοιωτών στο Διαδίκτυο που είναι σε συμφωνία το ακαδημαϊκό σύγγραμμα που διδάσκεται στο πανεπιστήμιό μας [1].

Παρατηρήθηκε ότι οι υπάρχοντες προσομοιωτές δεν ευθυγραμμίζονται με τον συμβολισμό και την αναπαράσταση που χρησιμοποιούν οι Lewis & Papadimitriou στο βιβλίο τους [1] και δεν επαρκούσαν για σωστή κατανόηση των Μηχανών Turing στους φοιτητές. Η ανάπτυξη αυτού του νέου εργαλείου προσομοίωσης είχε ως στόχο να αντιμετωπίσει αυτό το ζήτημα παρέχοντας μια διαδραστική πλατφόρμα που ακολουθεί αυστηρά το διδακτέο βιβλίο. Η υλοποίηση αυτής της διαδικτυακής εφαρμογής όχι μόνο διευκολύνει το έργο του δασκάλου παρέχοντας ένα διαδραστικό εργαλείο για σκοπούς επίδειξης και εξήγησης των μηχανών, αλλά διασφαλίζει επίσης ότι οι φοιτητές έχουν καλύτερη κατανόηση αυτών. Αυτό οφείλεται στη φιλική προς το χρήστη διεπαφή και στην ικανότητα να απεικονίζει κανείς εύκολα τη λειτουργία μιας Μηχανής Turing. Είναι προφανές πως ένας προσομοιωτής Μηχανής Turing μπορεί να βοηθήσει τα άτομα να κατανοήσουν καλύτερα τον τρόπο λειτουργίας των Μηχανών Turing, παρέχοντας μια οπτική αναπαράσταση της λειτουργίας τους και επιτρέποντας στους χρήστες να περάσουν στη διαδικασία υπολογισμού.

Ένας προσομοιωτής Μηχανής Turing μπορεί να συμβάλει ως ένα χρήσιμο εργαλείο και για

καθηγητές που διδάσκουν Θεωρία Υπολογισμού και την έννοια των Μηχανών Turing. Μπορεί να προσφέρει μια πρακτική προσέγγιση στη μάθηση, επιτρέποντας στον καθηγητή να δώσει την δυνατότητα στους φοιτητές να πειραματιστούν εν ώρα μαθήματος με διαφορετικούς αλγόριθμους και να αποκτήσουν μια βαθύτερη κατανόηση της έννοιας. Μπορεί να βοηθήσει επίσης οποιονδήποτε επιθυμεί να μελετήσει τη συμπεριφορά των αλγορίθμων και να διερευνήσει τα όρια της υπολογισσιμότητας.

Ένα ακόμη σημαντικός λόγος για την δημιουργία του προσομοιωτή είναι η χρήση του για τον εντοπισμό και τη διόρθωση σφαλμάτων στην υλοποίηση από τον καθηγητή όταν χρειάζεται να διορθωθούν παραδοτέες ασκήσεις, αλλά και από τους φοιτητές όταν θα κληθούν να λύσουν τις ασκήσεις αυτές.

Συνολικά, ένας προσομοιωτής Μηχανής Turing μπορεί να είναι ένα πολύτιμο εργαλείο για όποιον ενδιαφέρεται να μάθει για τη Θεωρία Υπολογισμού και συγκεκριμένα για την έννοια των Μηχανών Turing, καθώς και για ερευνητές και εκπαιδευτικούς στον τομέα της επιστήμης των υπολογιστών.

1.2 Συνεισφορά

Η παρούσα διπλωματική εργασία παρουσιάζει την ανάπτυξη μιας διαδικτυακής εφαρμογής που επιτρέπει στο χρήστη να σχεδιάζει και να προσομοιώνει Μηχανές Turing. Ο στόχος της εφαρμογής είναι να παρέχει σε φοιτητές και εκπαιδευτικούς έναν διαδραστικό και προσιτό τρόπο για να εξερευνήσουν τη συμπεριφορά των Μηχανών Turing και να κατανοήσουν τις θεμελιώδεις έννοιες της Θεωρίας Υπολογισμού. Η εφαρμογή επιτρέπει στους χρήστες να δημιουργούν και να προσομοιώνουν Μηχανές Turing, χρησιμοποιώντας την σημειογραφία και γραφική αναπαράσταση των Lewis και Παπαδημητρίου [1], να οπτικοποιούν τη συμπεριφορά της μηχανής και να πειραματίζονται με διαφορετικούς αλγόριθμους. Η εφαρμογή αναπτύχθηκε σε Typescript και έχει σχεδιαστεί για να είναι απλή και φιλική προς τον χρήστη, καθιστώντας την προσβάσιμη σε φοιτητές όλων των υποβάθρων και επιπέδων δεξιοτήτων. Παρέχοντας έναν διαδραστικό και προσιτό τρόπο εξερεύνησης των Μηχανών Turing, αυτή η εφαρμογή έχει τη δυνατότητα να βελτιώσει σημαντικά τον τρόπο με τον οποίο οι φοιτητές μαθαίνουν για την επιστήμη των υπολογιστών.

1.3 Περίγραμμα

Στο Κεφάλαιο 2 δίνεται το απαραίτητο υπόβαθρο για την υπόλοιπη διπλωματική εργασία. Συγκεκριμένα παρουσιάζονται πληροφορίες για την Θεωρία Υπολογισμού και για την Μηχανή Turing. Όσον αφορά την τελευταία, δίνονται πληροφορίες για τον ορισμό της και την γραφική απεικόνισή της. Παρουσιάζονται επίσης τα εργαλεία που χρησιμοποιήθηκαν για την υλοποίηση του προσομοιωτή.

Στο Κεφάλαιο 3 δίνεται η περιγραφή του προβλήματος που μελετήσαμε και η αναφορά σχετικών εργασιών που είναι ηλεκτρονικά διαθέσιμες. Ειδικότερα, παρουσιάζεται το πιο βασικό κίνητρο για την δημιουργία του προσομοιωτή και στην συνέχεια παρουσιάζονται οι λόγοι για τους οποίους αυτές οι εργασίες δεν ικανοποιούσαν τις ανάγκες μας.

Το Κεφάλαιο 4 πραγματεύεται την υλοποίηση του προσομοιωτή και την προσέγγιση σχεδιασμού του. Περιγράφονται τα βήματα της διαδικασίας και η τελική δομή, συμπεριλαμβανομένου του σχήματος που απεικονίζει την αρχιτεκτονική της εφαρμογής.

Στο Κεφάλαιο 5 με άξονα την Μηχανή Turing περιγράφονται οι δυνατότητες που παρέχει η εφαρμογή μας για προσομοίωση και δημιουργία μηχανών.

Στο Κεφάλαιο 6 παρουσιάζεται το ερωτηματολόγιο προς συμπλήρωση από φοιτητές. Έπειτα ακολουθούν τα αποτελέσματα των ερωτηματολογίων, μαζί με σχετικό σχολιασμό και σύγκριση με αντίστοιχα αποτελέσματα άλλων εφαρμογών.

Το Κεφάλαιο 7 περιλαμβάνει τα συμπεράσματα που προκύπτουν από την δημιουργία της εφαρμογής και τις μελλοντικές αναβαθμίσεις που δύναται να βελτιώσουν σημαντικά τον προσομοιωτή.

Ακολουθεί η βιβλιογραφία στο Κεφάλαιο 8 και οι σύνδεσμοι που χρησιμοποιήθηκαν προς καθοδήγηση για τη δημιουργία του κώδικα της εφαρμογής.

Τέλος, στο Παράρτημα Α' δίνεται ένας Οδηγός Χρήσης της εφαρμογής, ο οποίος αναμένεται να διευκολύνει την καλύτερη κατανόηση της εφαρμογής.

2. *Θεωρητικό Υπόβαθρο*

Στο σημείο αυτό θα περιγράψουμε έννοιες της Θεωρίας Υπολογισμού οι οποίες είναι απαραίτητες για την κατανόηση της εφαρμογής. Επίσης, θα γίνει μία σύντομη περιγραφή των εργαλείων που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής.

2.1. *Θεωρία Υπολογισμού*

Με τον όρο Θεωρία Υπολογισμού [2] αναφερόμαστε στον κλάδο της θεωρητικής πληροφορικής ο οποίος ασχολείται με τα μαθηματικά των υπολογιστών. Στην ουσία, ασχολείται με το εάν και το πόσο αποδοτικά είναι δυνατόν να λυθεί ένα πρόβλημα, αξιοποιώντας ένα υπολογιστικό μοντέλο το οποίο εκτελεί κάποιον αλγόριθμο. Αποτελεί μια αφηρημένη μαθηματική έννοια, η οποία όμως είναι ορισμένη με αυστηρούς κανόνες.

Ο τομέας της Θεωρίας Υπολογισμού διαιρείται σε δύο βασικούς κλάδους: τη θεωρία πολυπλοκότητας και τη θεωρία υπολογισιμότητας. Και οι δύο εφαρμόζονται όμως σε μοντέλα υπολογισμού. Η πρώτη παρουσιάζει κάποιες ομοιότητες με την ανάλυση αλγορίθμων, έναν άλλο θεμελιώδη κλάδο της επιστήμης υπολογιστών. Η ανάλυση αλγορίθμων όμως ασχολείται με την εύρεση των ιδιοτήτων ενός δεδομένου αλγορίθμου που επιδιώκει την επίλυση κάποιου υπολογιστικού προβλήματος, σε αντίθεση με την θεωρία πολυπλοκότητας, η οποία επικεντρώνεται στις εγγενείς υπολογιστικές ιδιότητες του ίδιου του προβλήματος.

2.2. *Μηχανές Turing*

2.2.1. *Ορισμός Μηχανής Turing*

Η Μηχανή Turing εφευρέθηκε το 1936 από τον Alan Turing [0] και είναι μια υποθετική συσκευή, η οποία χειρίζεται σύμβολα σύμφωνα με ένα σύνολο κανόνων. Παρά την απλότητά της, μια τέτοια μηχανή μπορεί να προσαρμοστεί, ώστε να προσομοιώνει την λογική οποιουδήποτε αλγορίθμου, δίνοντας την δυνατότητα στους ερευνητές να καταλάβουν τα όρια του μηχανικού υπολογισμού.

Ο ρόλος της Μηχανής Turing [3] είναι να μοντελοποιήσει μαθηματικώς μια μηχανή που χειρίζεται μια ταινία μηχανικά. Η ταινία περιέχει μια σειρά από σύμβολα και χρησιμοποιεί μια κεφαλή με την οποία μπορεί να διαβάσει και να καταγράψει ένα σύμβολο κάθε φορά. Σε κάθε στιγμή, η μηχανή

βρίσκεται σε κάποια κατάσταση από ένα πεπερασμένο σύνολο καταστάσεων. Η λειτουργία της καθορίζεται από ένα πεπερασμένο σύνολο στοιχειωδών εντολών, όπως "στην κατάσταση 42, εάν το σύμβολο που εμφανίζεται στην ταινία είναι το 0, τότε γράψε 1 και πήγαινε στην κατάσταση 17 και στην κατάσταση 17, εάν το σύμβολο που εμφανίζεται είναι 1, γράψε 1 και πήγαινε στην κατάσταση 6 και στην κατάσταση 6, εάν το σύμβολο που εμφανίζεται είναι 1, μετακίνησε την κεφαλή αριστερά και μείνε στην κατάσταση 6 και στην κατάσταση 6, εάν το σύμβολο που εμφανίζεται είναι 1, γράψε 0, κοκ.

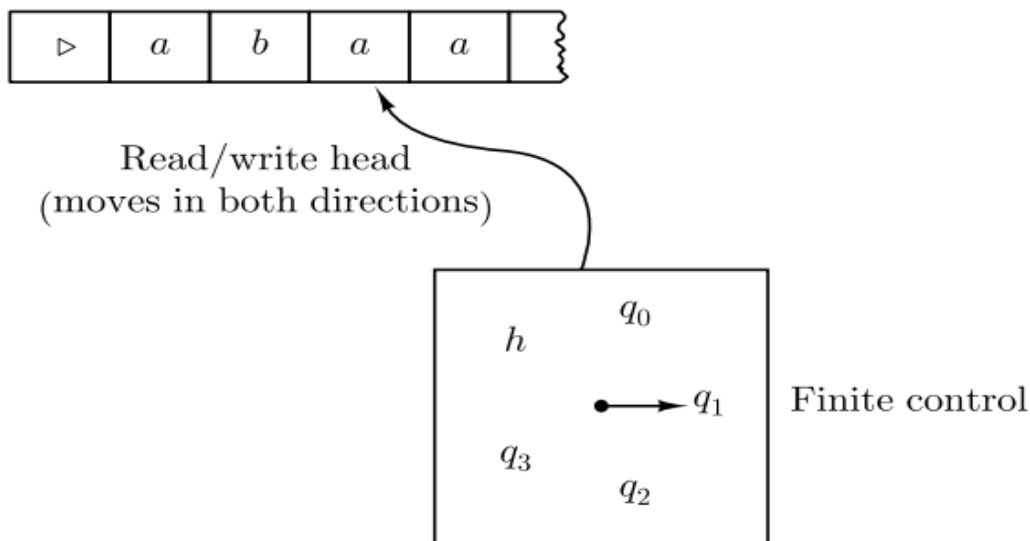
Πιο συγκεκριμένα, τα δομικά μέρη μιας Μηχανή Turing είναι:

1. Ένα αλφάβητο το οποίο περιέχει ένα ή παραπάνω σύμβολα, συμπεριλαμβανομένου του ειδικού συμβόλου κενού. Το πλήθος των συμβόλων του αλφαβήτου είναι πεπερασμένο.
2. Μία ταινία η οποία απαρτίζεται από κελιά, διατεταγμένα το ένα δίπλα στο άλλο. Κάθε κελί περιέχει ένα σύμβολο από το αλφάβητο. Λειτουργούμε υπό την υπόθεση ότι η ταινία δύναται να επεκταθεί στο άπειρο και προς τα αριστερά και προς τα δεξιά, δηλαδή δεν υπάρχει κανένας περιορισμός για μία Μηχανή Turing ως προς τις απαιτήσεις της για υπολογισμούς στην ταινία. Όποια κελιά δεν έχουν συμπληρωθεί, υποθέτουμε πως περιέχουν το σύμβολο κενού. Στα μοντέλα που θα ασχοληθούμε η ταινία έχει αριστερό άκρο, το οποίο περιέχει ένα ειδικό σύμβολο που απεικονίζεται με δεξί βέλος και η κεφαλή δεν μπορεί να κινηθεί πιο αριστερά από εκεί. Συνεχίζει να είναι όμως απείρως επεκτάσιμη η ταινία προς τα δεξιά.
3. Μία κεφαλή η οποία μπορεί να μετακινείται πάνω στην ταινία κατά ένα και μόνο ένα κελί κάθε φορά αριστερά ή δεξιά, όπως επίσης και να διαβάζει και να γράφει σύμβολα του αλφαβήτου πάνω στην ταινία.
4. Ένα σύνολο καταστάσεων με πεπερασμένο πλήθος που περιλαμβάνει και απεικονίζει την τρέχουσα κατάσταση της Μηχανής Turing. Ανάμεσα σε αυτές είναι η ειδική αρχική κατάσταση με την οποία ξεκινά η λειτουργία της μηχανής. Ένα υποσύνολο των καταστάσεων χαρακτηρίζεται ως τελικές καταστάσεις, όπου τερματίζεται η λειτουργία της μηχανής, όταν εισέρχεται σε μία εξ αυτών.
5. Έναν πεπερασμένο πίνακα, ο οποίος αποκαλείται συνάρτηση μετάβασης και περιέχει κανόνες, οι οποίοι προσδιορίζουν την ακριβή συμπεριφορά της μηχανής. Συγκεκριμένα, προσδιορίζει μονοσήμαντα ανάλογα με την τρέχουσα κατάσταση και το τρέχον σύμβολο, σε ποια κατάσταση θα μεταβεί η μηχανή και ποια ενέργεια θα κάνει με την κεφαλή.

Ανά πάσα στιγμή η μηχανή βρίσκεται σε κάποια κατάσταση και η κεφαλή σε κάποια θέση πάνω

στην ταινία. Συμβουλευόμενη την συνάρτηση μετάβασης, η μηχανή αποφασίζει σε ποια κατάσταση θα μεταβεί (μπορεί να μεταβεί και να παραμείνει ουσιαστικά στην κατάσταση που βρίσκεται ήδη) και τι θα κάνει με την κεφαλή (εγγραφή κάποιου συμβόλου στην θέση στην οποία βρίσκεται ή μετακίνηση αριστερά ή δεξιά κατά μία θέση). Ο κύκλος αυτός συνεχίζεται μέχρι η μηχανή να εισέλθει σε κάποια τερματική κατάσταση, οπότε και σταματάει η λειτουργία της.

Στην Εικόνα 1 απεικονίζονται τα δομικά μέρη της Μηχανής Turing.



Εικόνα 2.1. Απεικόνιση Μηχανής Turing

Οι Lewis και Παπαδημητρίου [1] ορίζουν τυπικά τη Μηχανή Turing ως μία πεντάδα $M = (K, \Sigma, \delta, s, H)$ όπου:

- K είναι ένα πεπερασμένο σύνολο καταστάσεων,
- Σ είναι ένα αλφάβητο που περιέχει το κενό σύμβολο και το σύμβολο αριστερού άκρου
- $s \in K$ η αρχική κατάσταση
- $H \subseteq K$ είναι το σύνολο των τελικών καταστάσεων
- δ η συνάρτηση μετάβασης $\delta: K \times \Sigma \rightarrow K \times \{\Sigma, \leftarrow, \rightarrow\}$

Ένα παράδειγμα μιας απλής Μηχανής Turing είναι η πεντάδα $M = (K, \Sigma, \delta, s, H)$ με :
 $K = \{q_0, h\}$, $\Sigma = \{a, \sqcup, \triangleright\}$, $s = q_0$, $H = \{h\}$ και δ όπως φαίνεται στην Εικόνα 2.2.

q	σ	$\delta(q, \sigma)$
q_0	a	(q_0, \leftarrow)
q_0	\sqcup	(h, \sqcup)
q_0	\triangleright	(q_0, \rightarrow)

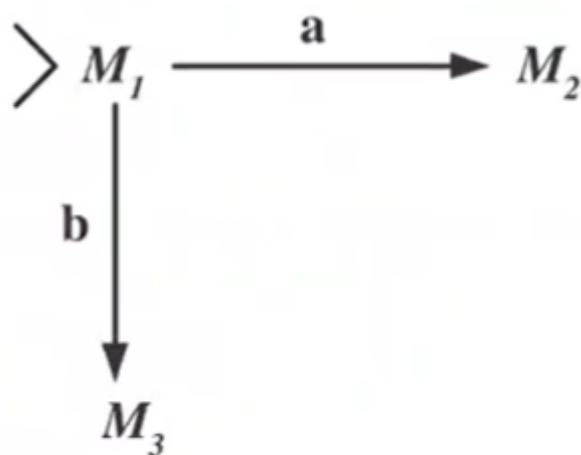
Εικόνα 2.2. Συνάρτηση μετάβασης απλής Μηχανής Turing

Στο παράδειγμα αυτό, έχουμε 2 καταστάσεις q_0 και h , με αρχική κατάσταση q_0 και h η κατάσταση τερματισμού και 3 σύμβολα ($a, \sqcup, \triangleright$).

2.2.2. Γραφική Απεικόνιση Μηχανής Turing

Στην πράξη, οι σύνθετες Μηχανές Turing δεν μπορούν να περιγραφούν με τον τυπικό τρόπο που αναφέρουν οι Lewis και Παπαδημητρίου [1], διότι είναι αρκετά δύσχρηστος. Για τον λόγο αυτό, καταφεύγουμε σε μια γραφική περιγραφή Μηχανών Turing, η οποία είναι πολύ πιο εύχρηστη. Η φιλοσοφία της αναπαράστασης αυτής βασίζεται στο ότι θα συνθέσουμε πιο πολύπλοκες μηχανές χρησιμοποιώντας απλούστερες. Τα δομικά μας στοιχεία είναι στοιχειώδεις Μηχανές Turing (μηχανή εγγραφής συμβόλου, μηχανές μετακίνησης κεφαλής κατά μία θέση) και ο τρόπος σύνθεσής τους βασίζεται στον κατευθυνόμενο γράφο (κόμβοι και ακμές με βέλη και σημάνσεις).

Η βασική αρχή είναι ότι οι κόμβοι αναπαριστούν τις βασικές μηχανές, οι ακμές είναι οι μεταβάσεις με βάση το τρέχον σύμβολο εισόδου (σ) και οι μεταβάσεις σημαίνουν τον τερματισμό της προηγούμενης μηχανής και επομένως την έναρξη της επόμενης. Ένα παράδειγμα μιας τέτοιας σύνθετης μηχανής M , φαίνεται στην Εικόνα 3.



Εικόνα 2.3. Παράδειγμα γραφικής απεικόνισης σύνθετης Μηχανής Turing M

- M1, M2, M3 : βασικές Μηχανές Turing
- Έναρξη υπολογισμού στην M1 (σύμβολο >)
- Η σύνθετη μηχανή M
 - ξεκινάει στην αρχική κατάσταση της M_1
 - αρχικά μιμείται την λειτουργία της M_1 μέχρι τερματισμού της M_1
 - μιμείται την M_2 ή την M_3 ανάλογα με το τρέχον σύμβολο τη στιγμή εκείνη

2.3. *Εργαλεία Ανάπτυξης Εφαρμογών*

JavaScript [15]: Η JavaScript είναι μια αντικειμενοστραφής γλώσσα προγραμματισμού και υποστηρίζει τον προγραμματισμό που βασίζεται σε συμβάντα, λειτουργικά και επιτακτικά. Χρησιμοποιείται κυρίως για την ανάπτυξη εφαρμογών στην πλευρά του πελάτη, που σημαίνει ότι ο κώδικας εκτελείται από το πρόγραμμα περιήγησης ιστού του χρήστη και όχι από τον διακομιστή. Αυτό επιτρέπει στη JavaScript να δημιουργεί διαδραστικά εφέ σε μια ιστοσελίδα, όπως ενημέρωση του περιεχομένου της σελίδας χωρίς επαναφόρτωση της σελίδας, επικύρωση φορμών, δημιουργία κινούμενων εικόνων και πολλά άλλα.

Είναι μια ευέλικτη γλώσσα και μπορεί να χρησιμοποιηθεί με πολλούς διαφορετικούς τρόπους, όπως δημιουργία εφαρμογών διακομιστή με χρήση Node.js, ανάπτυξη παιχνιδιών και δημιουργία εφαρμογών για desktop και mobile συσκευές χρησιμοποιώντας εργαλεία, όπως το Electron και το React Native.

Είναι μία από τις τρεις βασικές τεχνολογίες ανάπτυξης ιστοσελίδων, μαζί με την HTML (Hypertext Markup Language) και την CSS (Cascading Style Sheets) και αποτελεί βασικό συστατικό της σύγχρονης ανάπτυξης εφαρμογών. Η JavaScript είναι μια δημοφιλής γλώσσα και πολλές βιβλιοθήκες και πλαίσια έχουν αναπτυχθεί γι' αυτήν, διευκολύνοντας τους προγραμματιστές να δημιουργούν πολύπλοκες εφαρμογές και να αλληλεπιδρούν με άλλες τεχνολογίες.

Typescript [16]: Η TypeScript είναι ένα υπερσύνολο της JavaScript, που σημαίνει ότι όλος ο έγκυρος κώδικας JavaScript είναι επίσης έγκυρος κώδικας TypeScript. Προσθέτει προαιρετικούς, στατικού τύπου σχολιασμούς στη JavaScript, οι οποίοι μπορούν να βοηθήσουν να γίνουν οι μεγάλες, πολύπλοκες βάσεις κώδικα πιο διαχειρίσιμες και πιο κατανοητές. Περιλαμβάνει επίσης δυνατότητες από τις πιο πρόσφατες εκδόσεις JavaScript και μπορεί να χρησιμοποιηθεί για τη σύνταξη σύγχρονου κώδικα JavaScript, όπως async/wait και destructuring.

Παρέχει έλεγχο τύπων κατά τη στιγμή της μεταγλώττισης, ο οποίος μπορεί να βοηθήσει στη σύλληψη σφαλμάτων και στη βελτίωση της συνολικής ποιότητας του κώδικα. Αυτό μπορεί να είναι ιδιαίτερα χρήσιμο σε μεγαλύτερα έργα με πολλούς προγραμματιστές, καθώς μπορεί να βοηθήσει να διασφαλιστεί ότι όλοι χρησιμοποιούν τους σωστούς τύπους και να αποτρέψουν ορισμένους τύπους σφαλμάτων χρόνου εκτέλεσης.

Η TypeScript γίνεται όλο και πιο δημοφιλής και πολλά δημοφιλή πλαίσια front-end, όπως το Angular και το React, χρησιμοποιούν πλέον την TypeScript ως προεπιλεγμένη γλώσσα.

Συνοπτικά, η TypeScript είναι μια γλώσσα στατικών τύπων, εξαιρετικά ευανάγνωστη, που μπορεί να χρησιμοποιηθεί για την ανάπτυξη μεγάλων, πολύπλοκων εφαρμογών. Προσθέτοντας σχολιασμούς στατικού τύπου, μπορεί να βοηθήσει στον εντοπισμό σφαλμάτων και στη βελτίωση της συνολικής ποιότητας του κώδικα, διευκολύνοντας τη διατήρηση και την κλιμάκωση.

HTML [17]: Η HTML (Hypertext Markup Language) είναι η τυπική γλώσσα σήμανσης που χρησιμοποιείται για τη δημιουργία ιστοσελίδων. Είναι μια μορφή που βασίζεται σε κείμενο που παρέχει τη δομή και το περιεχόμενο των ιστοσελίδων, συμπεριλαμβανομένων επικεφαλίδων, παραγράφων, συνδέσμων, εικόνων και άλλων. Δεν είναι γλώσσα προγραμματισμού, αλλά χρησιμοποιείται για τον καθορισμό της δομής και του περιεχομένου των ιστοσελίδων και αποτελεί βασικό συστατικό της ανάπτυξης διαδικτυακών εφαρμογών.

Είναι μια γλώσσα σήμανσης, που σημαίνει ότι αποτελείται από ένα σύνολο ετικετών και χαρακτηριστικών, τα οποία χρησιμοποιούνται για να περιγράψουν το περιεχόμενο και τη δομή μιας ιστοσελίδας. Τα χαρακτηριστικά της μπορούν να χρησιμοποιηθούν για την παροχή πρόσθετων πληροφοριών σχετικά με μια ετικέτα, όπως η πηγή μιας εικόνας ή ο προορισμός ενός συνδέσμου.

CSS [18]: Η CSS (Cascading Style Sheets) είναι μια γλώσσα που χρησιμοποιείται για την περιγραφή της εμφάνισης και της μορφοποίησης ενός εγγράφου γραμμένου σε HTML. Χρησιμοποιείται για να διαχωρίσει το περιεχόμενο μιας ιστοσελίδας, η οποία ορίζεται με χρήση HTML, από την παρουσίασή της, η οποία ορίζεται με χρήση CSS.

Επιτρέπει στους προγραμματιστές να ελέγχουν τη διάταξη, το χρώμα, τη γραμματοσειρά και άλλα οπτικά στοιχεία μιας ιστοσελίδας, καθιστώντας δυνατή τη δημιουργία σελίδων με στυλ και οπτικά ελκυστικές. Οι κανόνες CSS ορίζονται χρησιμοποιώντας επιλογείς, οι οποίοι επιλέγουν τα στοιχεία HTML, στα οποία πρέπει να εφαρμοστούν τα στυλ, και ένα σύνολο δηλώσεων, που ορίζουν τα ίδια τα στυλ. Για παράδειγμα, ένας κανόνας CSS μπορεί να ορίζει ότι όλες οι επικεφαλίδες πρέπει να είναι μπλε και να έχουν συγκεκριμένο μέγεθος γραμματοσειράς.

JSON [19]: JSON (JavaScript Object Notation) ονομάζεται μια απλή μορφή ανταλλαγής δεδομένων που είναι εύκολο να διαβάσουν και να γράψουν οι προγραμματιστές και εύκολα να αναλύσουν και να δημιουργήσουν οι μηχανές. Είναι μια ανοιχτή τυπική μορφή που χρησιμοποιείται ευρέως για ανταλλαγή δεδομένων μεταξύ εφαρμογών, ειδικά για ανταλλαγή δεδομένων στο διαδίκτυο.

Βασίζεται σε ένα υποσύνολο της γλώσσας προγραμματισμού JavaScript και χρησιμοποιεί ζεύγη κλειδιών-τιμών για την αναπαράσταση δεδομένων. Τα κλειδιά είναι συμβολοσειρές και οι τιμές μπορεί να είναι συμβολοσειρές, αριθμοί, αντικείμενα (συλλογές ζευγών κλειδιών-τιμών), πίνακες (διατεταγμένες συλλογές τιμών), booleans (true ή false) ή null.

Ένα από τα πλεονεκτήματα είναι ότι η αναπαράσταση είναι ανεξάρτητη από τη γλώσσα, πράγμα που σημαίνει ότι μπορεί να χρησιμοποιηθεί με μια μεγάλη ποικιλία γλωσσών προγραμματισμού, συμπεριλαμβανομένων των JavaScript, Python, Ruby, PHP και πολλών άλλων. Είναι επίσης ελαφριά και συμπαγή, γεγονός που την καθιστά κατάλληλη για χρήση σε δίκτυα, ειδικά στο διαδίκτυο.

Η JSON μορφή έχει γίνει μια δημοφιλής εναλλακτική της XML (Extensible Markup Language), μια άλλη μορφή ανταλλαγής δεδομένων, λόγω της απλότητας, της συμπαγούς και ευκολίας χρήσης της. Χρησιμοποιείται συχνά για REST (Representational State Transfer) API, τα οποία χρησιμοποιούνται για την ανταλλαγή δεδομένων μεταξύ υπηρεσιών web, και χρησιμοποιείται επίσης ως μορφή δεδομένων για αποθήκευση και ανταλλαγή δεδομένων σε πολλές σύγχρονες εφαρμογές.

Angular [20]: Η Angular είναι μια πλατφόρμα και ένα πλαίσιο για τη δημιουργία εφαρμογών πελάτη μιας σελίδας χρησιμοποιώντας HTML και TypeScript. Η Angular είναι γραμμένη σε TypeScript. Εφαρμόζει βασική και προαιρετική λειτουργικότητα ως ένα σύνολο βιβλιοθηκών TypeScript που εισάγονται στις εφαρμογές. Η αρχιτεκτονική μιας εφαρμογής Angular βασίζεται σε ορισμένες θεμελιώδεις έννοιες, όπως τα components και τα services. Τα πρώτα αποτελούν βασικά δομικά στοιχεία της Angular και είναι οργανωμένα σε NgModules. Τα NgModules συλλέγουν σχετικό κώδικα σε λειτουργικά σύνολα και προσφέρουν μια μορφή ομαδοποίησης. Μια Angular εφαρμογή ορίζεται από ένα σύνολο NgModules, εκ των οποίων ένα παίζει πηγαίο ρόλο και επιτρέπει την εκκίνηση, καθώς συνήθως υπάρχουν διάφορες λειτουργικές μονάδες.

Visual Studio Code [21]: Το VSCode είναι ένα IDE περιβάλλον κατασκευασμένο και υποστηριζόμενο από την Microsoft. Παρέχει υποστήριξη για λειτουργίες ανάπτυξης, όπως εντοπισμός σφαλμάτων, εκτέλεση εργασιών και version control μέσω git. Μπορεί να χρησιμοποιηθεί για οποιαδήποτε γλώσσα προγραμματισμού, αλλά έχει ειδικά ενσωματωμένη υποστήριξη για την σύνταξη Typescript. Επίσης, υπάρχει η δυνατότητα εγκατάστασης επιπρόσθετων εφαρμογών για πιο εξειδικευμένες λειτουργίες, όπως π.χ. Angular snippets.

Docker [22]: Το Docker είναι μια πλατφόρμα για την ανάπτυξη, αποστολή και εκτέλεση εφαρμογών σε container. Παρέχει ένα σύνολο εργαλείων και υπηρεσιών για κατασκευή, αποστολή και λειτουργία container, καθώς και ένα πλούσιο οικοσύστημα πρόσθετων και επεκτάσεων για ενοποίηση με άλλα εργαλεία και υπηρεσίες. Το Docker διευκολύνει τη μετακίνηση εφαρμογών μεταξύ διαφορετικών περιβαλλόντων και παρέχει ένα συνεπές και αναπαραγώγιμο περιβάλλον για την εφαρμογή μας.

3.

Περιγραφή Προβλήματος

Η πρόθεση στην παρούσα διπλωματική εργασία είναι η ανάπτυξη ενός εργαλείου σχεδίασης και προσομοίωσης Μηχανών Turing, με γραφική διεπαφή χρήστη, δίνοντας ιδιαίτερη έμφαση στη χρήση του ως εκπαιδευτικό εργαλείο. Αυτό το διαδραστικό εργαλείο θα πρέπει να παρέχει στους φοιτητές οπτική ανατροφοδότηση σχετικά με τα υπολογιστικά βήματα μιας Μηχανής Turing που λειτουργεί και θα τους επιτρέπει να πειραματιστούν και να εξοικειωθούν με τις βασικές αρχές της θεωρίας των Μηχανών Turing ανεμπόδιστα. Ο πιο σημαντικός λόγος για την δημιουργία του προσομοιωτή, όπως θα δούμε και λίγο παρακάτω στις σχετικές εργασίες, είναι η έλλειψη μιας εφαρμογής Μηχανών Turing που να βασίζεται στην ορολογία, τις σημειώσεις και τα σύμβολα που χρησιμοποιούμε στο μάθημα, όπως διδάσκεται αυτό στο Πολυτεχνείο Κρήτης, σύμφωνα με το βιβλίο των Lewis & Papadimitriou, *Elements of the Theory of Computation* [1]. Επιπροσθέτως, ο προσομοιωτής μπορεί να αποτελέσει σημαντικό εργαλείο και για τον καθηγητή, του οποίου η δουλειά αφορά στη διόρθωση ασκήσεων πάνω σε Μηχανές Turing, που απαιτεί πολύ προσοχή, συγκέντρωση και χρόνο, καθώς και στην παρουσίαση και διδασκαλία στους φοιτητές, οι οποίοι μπορούν την ώρα του μαθήματος να έχουν μια διαδραστική απεικόνιση αυτών που μαθαίνουν. Για να επιτευχθούν τα προαναφερθέντα, απαιτείται προσεκτική ανάλυση των διαφόρων εννοιών και παραδειγμάτων που παρουσιάζονται στο βιβλίο, ώστε να χρησιμοποιηθούν για τον σχεδιασμό και την ανάπτυξη της διαδικτυακής εφαρμογής.

Ο προσομοιωτής που θα παρουσιαστεί παρακάτω καλύπτει πλήρως τις πρότυπες Μηχανές Turing, αλλά δεν υποστηρίζει την δημιουργία ή προσομοίωση επεκτάσεων Μηχανών Turing με πολλαπλές ταινίες ή πολλαπλές κεφαλές, ταινία δύο κατευθύνσεων, πολυδιάστατη ταινία, μνήμη τυχαίας προσπέλασης ή συνδυασμών των προηγούμενων. Είναι γνωστό όμως, ότι η υπολογιστική ισχύς των μηχανών αυτών είναι ίδια με αυτή της πρότυπης μηχανής.

Έτσι, ο προσομοιωτής θα πρέπει να ικανοποιεί τις παρακάτω προδιαγραφές:

- Να είναι δωρεάν και ανοιχτού κώδικα.
- Να είναι διαισθητικά εύκολος στη χρήση και φιλικός προς το χρήστη σε εμφάνιση και συμπεριφορά, προκειμένου να μην αποσπάται η προσοχή από τη μελέτη των μηχανών.
- Να αναπαριστά μία Μηχανή Turing μέσω γραφήματος, στην οποία οι καταστάσεις αντιπροσωπεύονται από κόμβους και οι μεταβάσεις με ακμές. Να απεικονίζει την εναρκτήρια μηχανή με ένα ειδικό σύμβολο. Ο χρήστης πρέπει επίσης να μπορεί να αλλάξει το αλφάβητο και τα περιεχόμενα της ταινίας εισαγωγής, η οποία πρέπει ομοίως να απεικονίζεται γραφικά.
- Να δίνει την δυνατότητα στον χρήστη να φτιάξει την δική του μηχανή, η οποία θα

αποτελείται από συνδυασμούς βασικών μηχανών, καθώς και μεταβάσεις.

- Να είναι διαδραστικός, όχι μόνο κατά τη σχεδίαση, αλλά και κατά τον υπολογισμό.
- Να υποστηρίζει μια αναπαραγωγή βήμα-βήμα της εκτέλεσης, καθώς επίσης και στιγμιαία παύση της λειτουργίας, όπως επίσης και έλεγχο της ταχύτητας λειτουργίας της μηχανής, ώστε να γίνεται ευκολότερα κατανοητή, όταν το επιθυμεί ο χρήστης.
- Να υποστηρίζει την δυνατότητα αποθήκευσης της μηχανής που δημιούργησε ο χρήστης τοπικά, αλλά και την δυνατότητα φόρτωσης στην εφαρμογή μιας μηχανής που είναι ήδη αποθηκευμένη τοπικά.
- Είναι σημαντικό επίσης, η εφαρμογή να έχει έτοιμα παραδείγματα μηχανών, τα οποία διδάσκονται ως πρώτα παραδείγματα κατά την διάρκεια του μαθήματος.

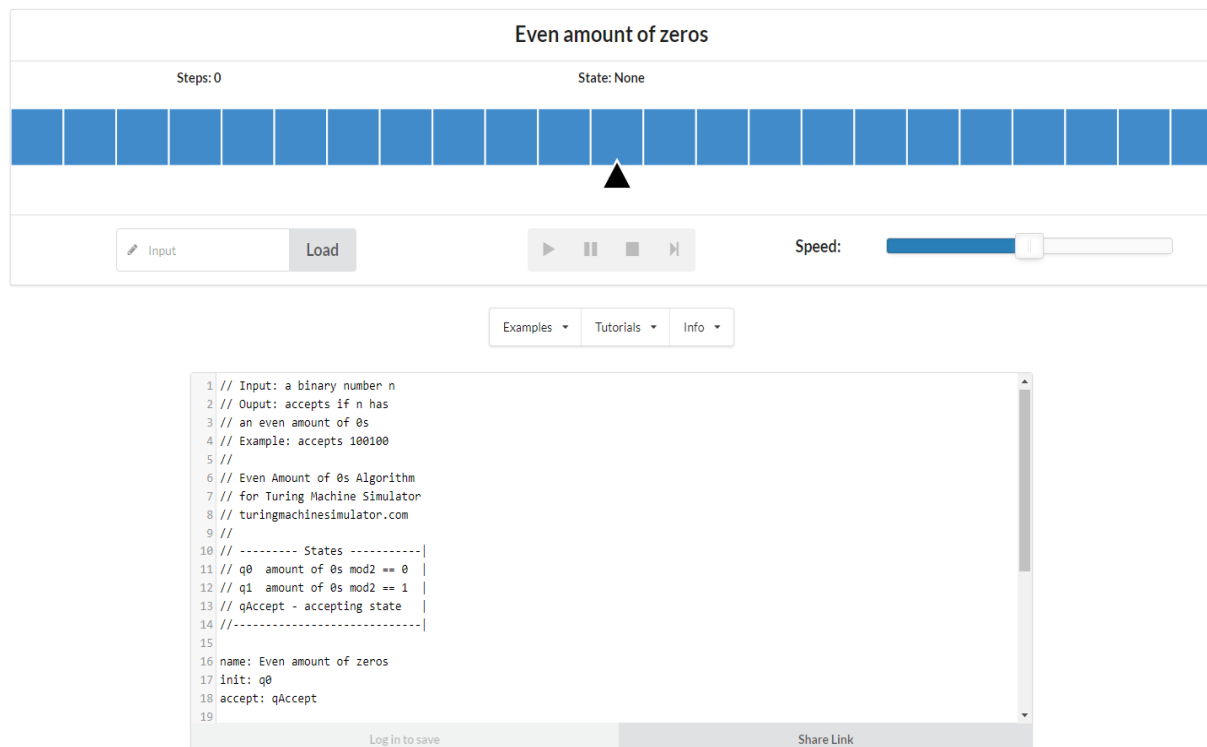
3.1. Σχετικές Εργασίες

Όπως αναφέρθηκε και νωρίτερα, στο διαδίκτυο υπάρχουν έτοιμες εφαρμογές προσομοίωσης Μηχανών Turing, οι οποίες περιλαμβάνουν αρκετά στοιχεία από τα παραπάνω. Δεν υπάρχει όμως, κανένας προσομοιωτής που να ακολουθεί τον συμβολισμό και το βιβλίο που διδάσκεται στο μάθημα του Πολυτεχνείου Κρήτης. Έτσι, η διευκόλυνση των φοιτητών και του καθηγητή στην εκμάθηση, κατανόηση, παραγωγή και διόρθωση μηχανών μέχρι σήμερα ήταν ανέφικτη. Ακολουθούν οι σύνδεσμοι για τους υπάρχοντες προσομοιωτές που έχουμε εντοπίσει:

1. <https://turingmachinesimulator.com/>
2. <https://turingmachine.io/>
3. <https://www.jflap.org/>

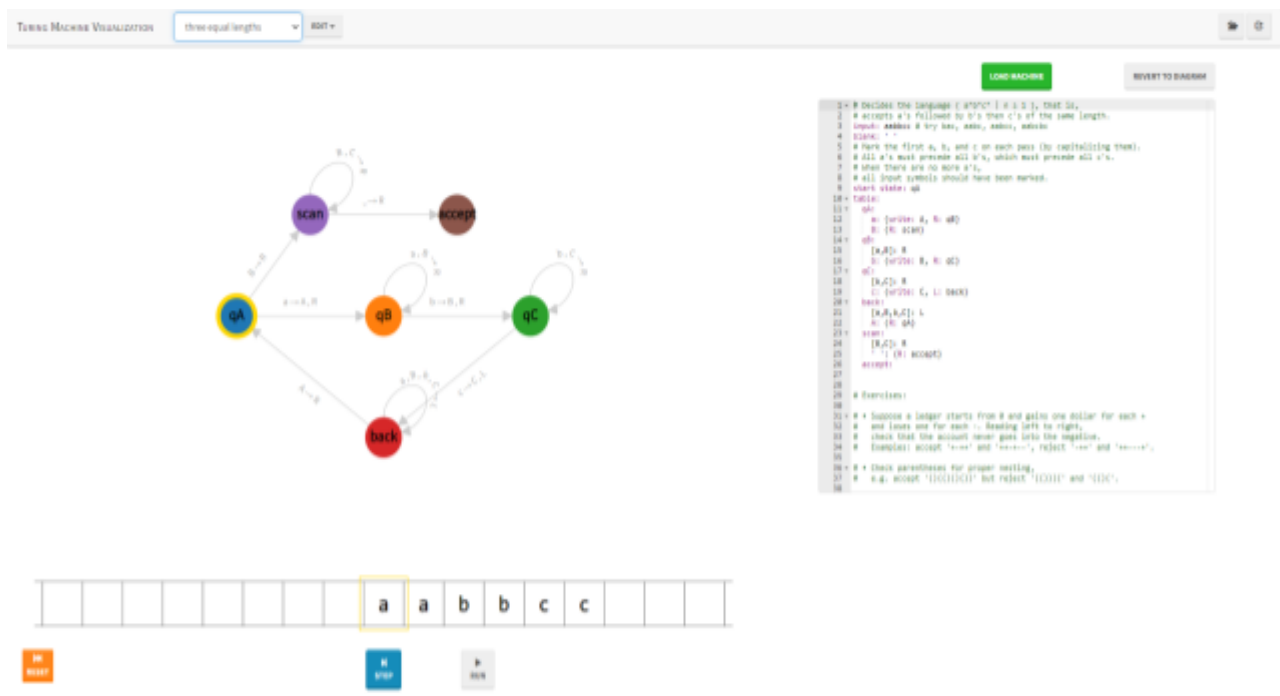
Εάν ο χρήστης της εφαρμογής έχει διδαχθεί την Θεωρία Υπολογισμού και τις Μηχανές Turing, όπως παρουσιάζονται στο μάθημα στο Πολυτεχνείο Κρήτης, τότε θα δυσκολευτεί να προσαρμοστεί στην αναπαράσταση, όπως την προσφέρουν οι τρεις εφαρμογές που αναφέρθηκαν. Παραδείγματα ακολουθούν στις παρακάτω εικόνες:

TURING MACHINE



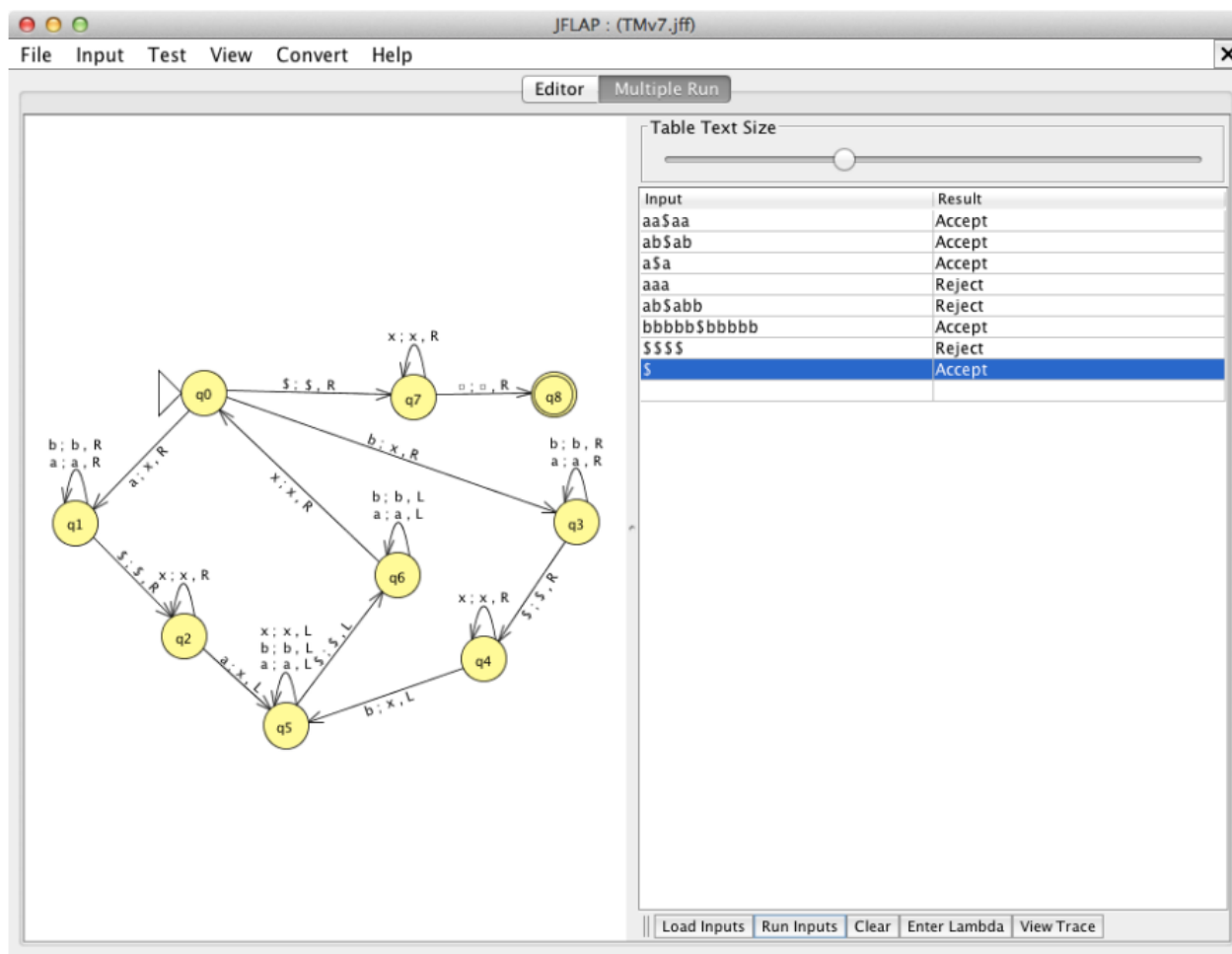
Εικόνα 3.1. Σχετική Εργασία Νο1 (<https://turingmachinesimulator.com/>)

Το Turing Machine Simulator του Martin Guarte δείχνει με πολύ όμορφη, αλλά ελλιπή, απεικόνιση τη λειτουργία μιας Μηχανής Turing, όπως φαίνεται στην Εικόνα 3.1. Εμφανής είναι μόνο η ταινία, ενώ η ενεργή κατάσταση απλά τυπώνεται ακριβώς από πάνω. Το γραφικό περιβάλλον δίνει την δυνατότητα εκκίνησης, παύσης, διακοπής και βηματικής προόδου, όπως επίσης και την επιλογή ταχύτητας και τέλος ένα πεδίο για εισαγωγή της συμβολοσειράς εισόδου. Η ταινία κινείται καθώς λειτουργεί η μηχανή και στο τέλος τυπώνεται το αποτέλεσμα. Η συγκεκριμένη προσέγγιση είναι πολύ μονομερής από την μεριά της ταινίας. Είναι πολύ δύσκολο για τον χρήστη, τόσο το να διαβάσει μια υπάρχουσα μηχανή, όσο και να δημιουργήσει μια καινούρια, καθώς αυτό γίνεται σε μορφή κειμένου, χρησιμοποιώντας ένα συντακτικό μη τυποποιημένο.



Εικόνα 3.2. Σχετική Εργασία Νο2 (<https://turingmachine.io/>)

Το turingmachine.io του Andy Li ακολουθεί διαφορετική προσέγγιση στην απεικόνιση, δίνοντας μεγαλύτερη βαρύτητα στην απεικόνιση των καταστάσεων, όπως φαίνεται στην Εικόνα 3.2. Αυτό είναι ιδιαίτερα πρακτικό για τον χρήστη που θέλει να κατανοήσει μια ήδη καταχωρημένη Μηχανή Turing, όμως δεν βοηθάει ιδιαίτερα στην δημιουργία ή τροποποίησή της. Και πάλι, εδώ ο ορισμός της Μηχανής Turing γίνεται μέσω κειμένου, αξιοποιώντας ένα μη τυποποιημένο συντακτικό, ορισμένο από τον δημιουργό της εφαρμογής. Αξίζει επίσης να σημειωθεί πως εδώ λείπει κάποια λειτουργικότητα, όπως η επιλογή ταχύτητας ή η δυνατότητα αλλαγής εισόδου μέσω της διεπαφής, αλλά συνοδεύεται από ένα πολύ πρακτικό τεχνικό εγχειρίδιο. Αυτή η προσέγγιση πλησιάζει περισσότερο τις δικές μας ανάγκες, αλλά δεν τις καλύπτει.



Εικόνα 3.3. Σχετική Εργασία Νο3 (<https://www.jflap.org/>)

Τέλος, η εφαρμογή jflap ακολουθεί επίσης διαφορετική προσέγγιση στην απεικόνιση. Δίνεται έμφαση στον ορισμό της Μηχανής Turing, όπως φαίνεται στην Εικόνα 3.3. Η σχεδίαση Μηχανών Turing με αυτόν τον τρόπο είναι πιο περίπλοκος και απαιτεί μεγαλύτερη προσοχή, με το ενδεχόμενο η σχεδίαση να είναι τελικά ελλιπής. Επίσης, η διεπαφή χρήστη θεωρείται ξεπερασμένη, καθώς δεν είναι ιδιαίτερα διαισθητική και φιλική προς τον χρήστη. Τέλος, για την λειτουργία του προσομοιωτή απαιτείται η εγκατάσταση του εργαλείου, ένα σημαντικό μειονέκτημα για την ευελιξία στην χρήση.

4. Προσέγγιση Σχεδιασμού και Υλοποίηση

4.1. Υλοποίηση Εφαρμογής

Στόχος της υλοποίησης ήταν η δημιουργία ενός εργαλείου, σύμφωνα με τους συγγραφείς Lewis & Papadimitriou, των οποίων το βιβλίο «Στοιχεία Θεωρίας Υπολογισμού» [1] διδάσκεται στο μάθημα «Θεωρία Υπολογισμού» του Πολυτεχνείου Κρήτης. Η έλλειψη του προσομοιωτή μηχανών Turing με τις προδιαγραφές του βιβλίου μειώνει σημαντικά την παραγωγικότητα του μαθήματος, τόσο για τον καθηγητή, όσο και για τους φοιτητές. Η Θεωρία Υπολογισμού είναι το θεμέλιο της επιστήμης των υπολογιστών και παρέχει τη θεωρητική βάση για την κατανόηση του τρόπου λειτουργίας των υπολογιστών διαχρονικά. Ένας προσομοιωτής είναι ένα πολύτιμο εργαλείο σε αυτό το μάθημα, καθώς επιτρέπει στον καθηγητή και στους φοιτητές να εφαρμόσουν τις έννοιες που έχουν μάθει και να τις δουν σε δράση.

Οι δύο βασικές αποφάσεις που πήραμε όσον αφορά την υλοποίηση ήταν η επιλογή μιας σύγχρονης διεπαφής χρήστη (UI) και η μη απαίτηση εγκατάστασης λογισμικού για την χρήση του προσομοιωτή. Με αυτόν τον τρόπο, η μαθησιακή διαδικασία θα μπορεί να βελτιωθεί και να γίνει περισσότερο αποτελεσματική, επιτρέποντας μεγαλύτερη εστίαση στο υλικό που διδάσκεται. Η σύγχρονη διεπαφή χρήστη σχεδιάστηκε για να παρέχει μια οπτικά ελκυστική και διαισθητική διεπαφή, διευκολύνοντας τους φοιτητές να κατανοήσουν και να ασχοληθούν με τις έννοιες που διδάσκονται. Καταργώντας την ανάγκη εγκατάστασης, εξαλείφονται τυχόν φραγμοί που μπορεί να εμποδίζουν τη συμμετοχή των φοιτητών.

Με βάση λοιπόν αυτά τα κριτήρια αποφασίσαμε ότι πρέπει να υλοποιήσουμε ένα εργαλείο με πρόσβαση σε φυλλομετρητή (browser) και όχι μια εφαρμογή που έχει αναπτυχθεί για μια συγκεκριμένη πλατφόρμα (native). Έτσι, οι επιλογές ήταν είτε σελίδα που υποστηρίζεται από κάποιον server, είτε stand-alone εφαρμογή, δηλαδή εφαρμογή που δεν απαιτεί κάποιο ξεχωριστό λογισμικό για να λειτουργήσει.

Η πρώτη επιλογή θα μας έδινε την δυνατότητα να χρησιμοποιήσουμε οποιαδήποτε γλώσσα προγραμματισμού για την λογική και την δυνατότητα αποθήκευσης δεδομένων για κάθε χρήστη ανάμεσα σε sessions, καθώς επίσης και την ζωντανή συνεργασία μεταξύ χρηστών ή και του καθηγητή. Θα είχαν την δυνατότητα δηλαδή, να συνεργαστούν για να σχεδιάσουν μαζί, ταυτόχρονα μια Μηχανή Turing. Όμως για λόγους απλότητας, στο πλαίσιο της διπλωματικής εργασίας, επιλέξαμε client stand-alone εφαρμογή, η οποία φορτώνει μια φορά και μετά μπορεί να χρησιμοποιηθεί, χωρίς να

απαιτείται συνεχόμενη σύνδεση στο διαδίκτυο. Το χαρακτηριστικό αυτό επιτυγχάνεται με τη βοήθεια του εργαλείου Electron που μας δίνει την επιλογή να “πακετάρουμε” την εφαρμογή και να μην απαιτείται σύνδεση. Το Electron παρέχει απλώς ένα εγγενές container για την διαδικτυακή μορφή μας, έτσι ώστε να μοιάζει με εφαρμογή desktop.

Στο πλαίσιο της standalone εφαρμογής ήμασταν υποχρεωμένοι να χρησιμοποιήσουμε Javascript ή Web Assembly. Ωστόσο, η δεύτερη επιλογή απορρίφθηκε, διότι δεν υπάρχει καλή υποστήριξη, καθώς βρίσκεται σε αρχικά στάδια ακόμη. Σε συνέχεια της επιλογής γλώσσας, προχωρήσαμε ένα μικρό βήμα πιο μπροστά και διαλέξαμε συγκεκριμένα την Typescript, η οποία προσφέρει καλύτερη δομή από την Javascript.

Εφόσον γνωρίζαμε τα παραπάνω και επιλέχθηκε και η γλώσσα προγραμματισμού, βασιζόμενοι στην Typescript επιλέχθηκε το framework Angular, το οποίο έχει πλήρη υποστήριξη για Typescript προσφέροντας μοντέρνο και εύχρηστο UI με την χρήση της Angular Material βιβλιοθήκης. Επιπροσθέτως, εκτός από τα προνόμια αυτά, η Angular μπορεί να χρησιμοποιήσει άριστα και την βιβλιοθήκη rxjs, η οποία περιλαμβάνει αντιδραστικά στοιχεία (reactive observables), υπεύθυνα για την ασύγχρονη ανανέωση των παρατηρήσιμων στοιχείων (asynchronous observables) που χρειαστήκαμε για την υλοποίησή μας.

Με την επιλογή της Angular, ακολουθήσαμε πιστά και την λογική της, δηλαδή τη δημιουργία συστατικών (components), ένα για κάθε λογική οντότητα. Με τον όρο Angular Components αναφερόμαστε στα δομικά στοιχεία μιας εφαρμογής που καθορίζουν διαφορετικές πτυχές της διεπαφής χρήστη. Είναι ένα directive με ένα πρότυπο που επιτρέπει την δημιουργία στοιχείων μιας διεπαφής χρήστη σε μια εφαρμογή Angular. Τα στοιχεία θα έχουν πάντα ένα πρότυπο, έναν επιλογέα και μπορεί να έχουν ή να μην έχουν ξεχωριστό ύφος και εμφάνιση. Κάθε Angular Component αποτελείται από:

- Ένα πρότυπο HTML που δηλώνει τι αποδίδεται στη σελίδα.
- Μια κλάση TypeScript που ορίζει τη συμπεριφορά.
- Έναν επιλογέα CSS που ορίζει πώς χρησιμοποιείται το στοιχείο σε ένα πρότυπο.
- Ένα αρχείο spec.ts για testing.

4.2. Προσέγγιση Σχεδιασμού

Με βάση τα παραπάνω, δημιουργήσαμε τα Angular Components που ακολουθούν:

App.component: Ορίζει την βάση της εφαρμογής, την σελίδα δηλαδή πάνω στην οποία θα εμφανίζονται όλα τα components και οι λειτουργίες που υλοποιήθηκαν ώστε να προσομοιωθούν οι Μηχανές Turing.

main-page.component: Χωρίζει την βασική σελίδα σε τρία μέρη. Την γραμμή εργαλείων, το πλαίσιο κειμένου και τον καμβά. Επίσης, αρχικοποιεί στην μνήμη του browser (local storage) την Μηχανή Turing που πρόκειται να υλοποιηθεί.

canvas.component: Εμφανίζει στον καμβά την Μηχανή Turing που έχει φορτωθεί, μαζί με την ταινία και την κεφαλή. Επίσης, κάτω αριστερά απεικονίζει το λογότυπο του Πολυτεχνείου Κρήτης, μαζί με το όνομα του δημιουργού της εφαρμογής.

toolbar.component: Περιέχει όλη την γραμμή εργαλείων και χρησιμοποιείται από το main-page.component για την απεικόνισή της. Περιλαμβάνει κουμπί για φόρτωση στον καμβά έτοιμης Μηχανής Turing από τα παραδείγματα που διδάσκονται στο μάθημα, επιλογή ενεργειών αποθήκευσης της μηχανής που έχει σχεδιάσει ο χρήστης, φόρτωση Μηχανής Turing από τον τοπικό δίσκο του χρήστη, ή αποθήκευση του καμβά της εφαρμογής με όσα απεικονίζονται εκείνη την στιγμή. Επίσης περιέχει τα απαραίτητα κουμπιά για την προσομοίωση της μηχανής, δηλαδή, εκκίνηση (Run) που ξεκινάει την μηχανή, σύμφωνα την αρχική κατάστασή της, και συνεχίζει την λειτουργία της εάν έχει προηγηθεί το κουμπί παύσης. Κουμπί παύσης (Pause) που σταματάει την μηχανή στιγμιαία. Κουμπί Step για την πρόοδο της λειτουργίας της μηχανής σταδιακά με μία αλλαγή κάθε φορά, καθώς επίσης και κουμπί για έλεγχο της ταχύτητας με την οποία θα προσομοιώνεται η λειτουργία της. Τέλος, περιλαμβάνει και δύο κουμπιά στην δεξιά άκρη της εργαλειοθήκης, το ένα για τη δημιουργία ενός νέου node βασικών Μηχανών Turing ή συνδυασμό αυτών, και το άλλο για τη δημιουργία νέας μετάβασης. Τα δύο τελευταία κουμπιά που αναφέρθηκαν ανοίγουν δύο παράθυρα για την εισαγωγή δεδομένων που απαιτούνται για την δημιουργία είτε του node είτε της μετάβασης.

tape.component: Απεικονίζει την ταινία της Μηχανής Turing μαζί με το αρχικό σύμβολο και την κεφαλή και χρησιμοποιείται από το main-page.component. Επίσης, υλοποιεί το χαρακτηριστικό της ταινίας, όταν αυτή γίνει αρκετά μεγάλη και η κεφαλή πρόκειται να ξεφύγει από τα ορατά σε εμάς κελιά της ταινίας, τότε με τη χρήση μιας μπάρας ολίσθησης ακολουθεί αυτόματα την κεφαλή και το κελί στο οποίο βρίσκεται εκείνη τη στιγμή χρωματίζοντας το με κόκκινο χρώμα. Η επιλογή απεικόνισης της κεφαλής με κόκκινο χρώμα έγινε για να είναι ευδιάκριτη στον χρήστη η θέση της κεφαλής ανά πάσα στιγμή.

tape-data.component: Απεικονίζει το παράθυρο που ανοίγει, όταν με τον κέρσορα διαλέξει ο χρήστης την ταινία. Στο παράθυρο αυτό εμπεριέχονται το αλφάβητο της μηχανής και η συμβολοσειρά της ταινίας, τα οποία μπορεί να επεξεργαστεί ο χρήστης με την βοήθεια των κουμπιών που του δίνονται δίπλα από το κάθε ένα. Επίσης, το παράθυρο περιλαμβάνει και τέσσερα κουμπιά για την

αλλαγή θέσης της κεφαλής.

state-card.component: Απεικονίζει το κάθε node της Μηχανής Turing, το οποίο περιλαμβάνει συνδυασμό βασικών και σύνθετων μηχανών, καθώς και το χαρακτηριστικό εάν το node είναι το αρχικό με το οποίο ξεκινάει η Μηχανή Turing, το οποίο ξεχωρίζει με την χρήση ενός χαρακτηριστικού συμβόλου στα αριστερά από το αρχικό node, όπως συμβολίζεται και στο βιβλίο των Lewis & Papadimitriou [1]. Επίσης, περιέχει και την ιδιότητα του node να έχει την δυνατότητα ο χρήστης να σύρει το κάθε ένα από αυτά όπου επιθυμεί πάνω στον καμβά. Η ιδιότητα αυτή είναι σημαντική για την ευδιάκριτη σχεδίαση της μηχανής, ώστε να είναι ευανάγνωστη και κατανοητή.

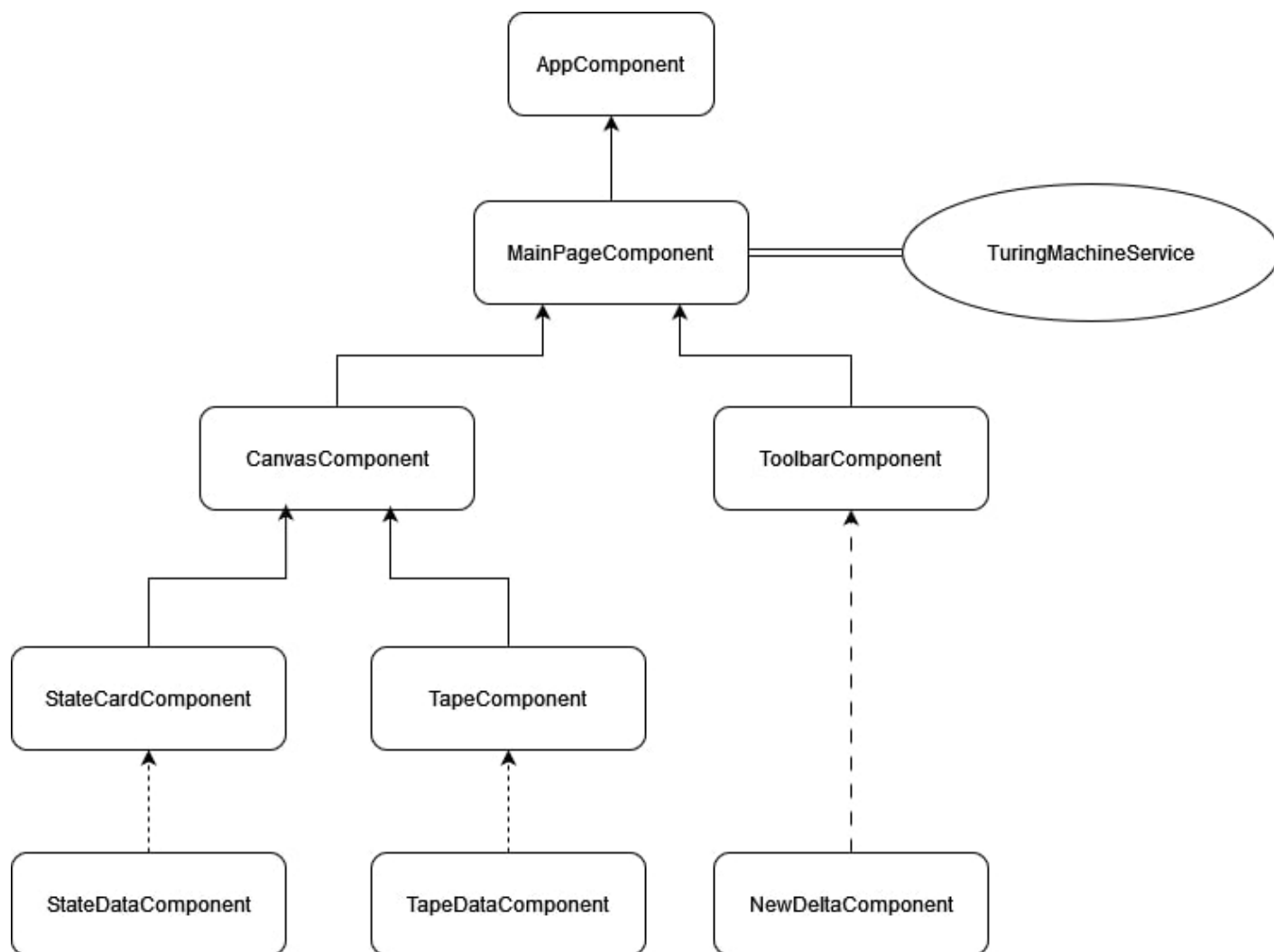
state-data.component: Απεικονίζει το παράθυρο που ανοίγει, όταν με τον κέρσορα διαλέξει ο χρήστης ένα υπάρχον node. Στο παράθυρο αυτό, βρίσκονται τα εργαλεία για την επεξεργασία του node. Αρχικά, απεικονίζεται το χαρακτηριστικό αναγνωριστικό id του node και ο τύπος του (αρχικός, ενδιάμεσος και τελικός). Ακολουθούν τα εργαλεία επεξεργασίας των βασικών και σύνθετων Μηχανών Turing που απαρτίζουν το node με λειτουργίες πρόσθεσης και αφαίρεσης μηχανών, καθώς και επεξεργασία των μεταβάσεων που ξεκινούν από το επιλεγμένο node και η δημιουργία νέας μετάβασης. Στο τέλος, υπάρχει κουμπί για την διαγραφή του επιλεγμένου node.

new-delta-data.component: Απεικονίζει το παράθυρο που ανοίγει, όταν επιλεχθεί το κουμπί που προαναφέρθηκε στο toolbar.component για δημιουργία νέας μετάβασης.

machine-examples.component: Το Component αυτό περιέχει μόνο κώδικα Typescript και πιο συγκεκριμένα πέντε αρχεία με Μηχανές Turing, όπως τις διαβάζει η εφαρμογή, σε μορφή JSON. Τα τρία αρχεία αφορούν έτοιμες Μηχανές Turing που διδάσκονται στο μάθημα (Copy Machine, Shift Right Machine, Deciding Machine), μια άδεια μηχανή που χρησιμοποιείται στην αρχικοποίηση και την μηχανή χωρίς αρχικά δεδομένα που προορίζεται για τον χρήστη, ο οποίος θα δημιουργήσει μια νέα, δική του Μηχανή Turing.

turing-machine.service: Το αρχείο service περιλαμβάνει όλη τη λογική της εφαρμογής σε γλώσσα Typescript. Σχεδόν όλες οι συναρτήσεις των components δεν κάνουν τίποτα περισσότερο από το να καλέσουν συναρτήσεις του service. Η μόνη τους δουλειά είναι να απεικονίζουν με HTML όσα περιγράψαμε και να επικοινωνούν με το service, ώστε να μπορούν να χρησιμοποιήσουν τις συναρτήσεις και τις μεθόδους του. Το αρχείο service έχει τον κώδικα για να γίνουν όλες οι ενέργειες που αναφέρθηκαν στα προηγούμενα components. Η σχεδίαση αυτή έγινε, ώστε σε μεταγενέστερο χρόνο να υπάρχει η δυνατότητα αντικατάστασης του αρχείου service με κλήσεις προς κάποιον server.

Μαζί, αυτά τα στοιχεία (components) που αναφέρονται παραπάνω, δημιουργούν ένα σύστημα που επιτρέπει στην εφαρμογή να λειτουργεί αποτελεσματικά και παρέχει στους χρήστες μια ολοκληρωμένη εμπειρία. Η αρχιτεκτονική που ακολουθούν παρουσιάζεται στην παρακάτω γραφική απεικόνιση:



Εικόνα 4.1: Η αρχιτεκτονική της εφαρμογής

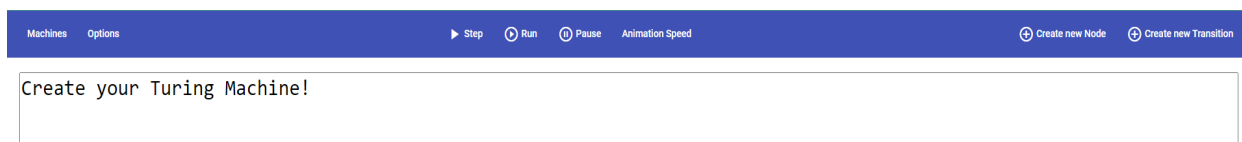
5.

Αποτελέσματα

Προκειμένου να υπάρξει μια πιο ολοκληρωμένη κατανόηση της εφαρμογής, θα ακολουθήσουν οπτικά βοηθήματα για να παρουσιαστούν τα βασικά χαρακτηριστικά και οι λειτουργίες της εφαρμογής, όπως περιγράφηκαν παραπάνω. Με άλλα λόγια, παρατίθενται εικόνες από την χρήση της εφαρμογής που δείχνουν το τελικό οπτικό αποτέλεσμα.

5.1 Προσομοίωση Μηχανής Turing

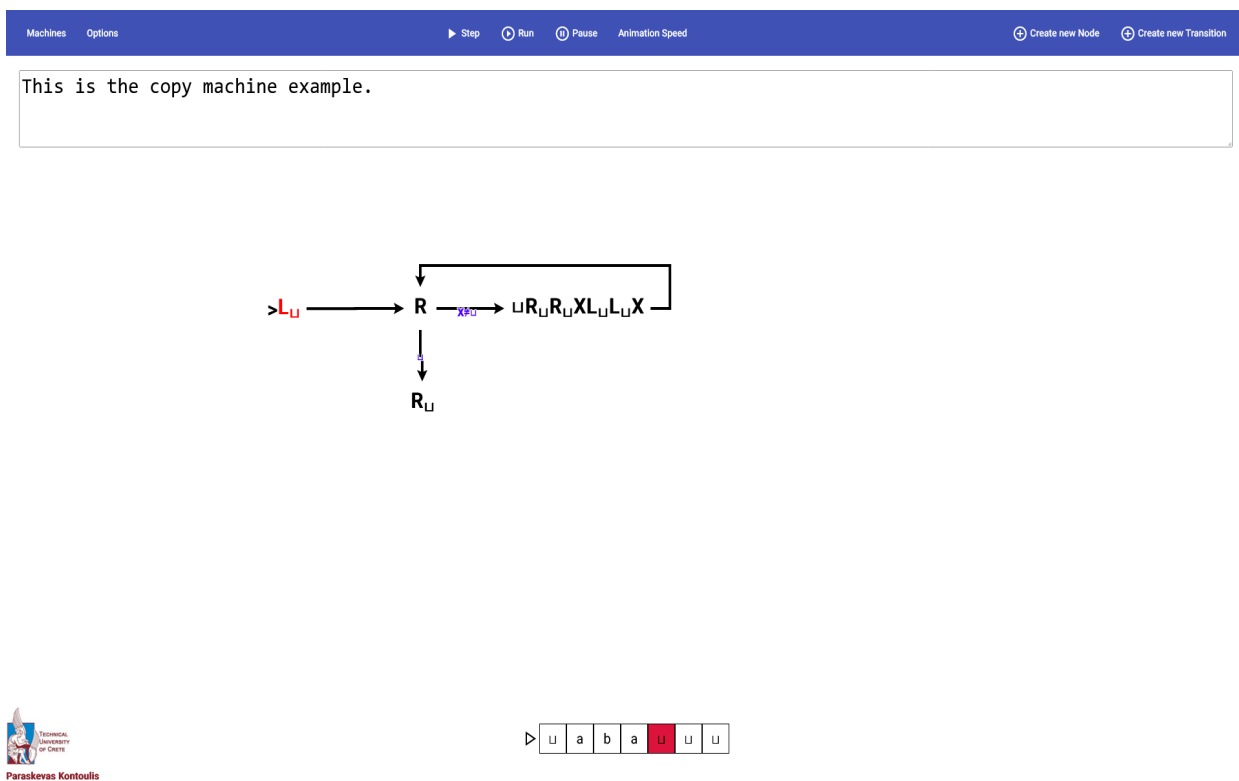
Αρχικά, στην Εικόνα 5.1 φαίνεται η σελίδα της εφαρμογής χωρίς κάποια υλοποίηση Μηχανής Turing. Παρατηρούνται όμως μερικά από τα components που αναφέρθηκαν νωρίτερα (ο διαχωρισμός σε τρία κομμάτια, η γραμμή εργαλείων, το πλαίσιο κειμένου, ο καμβάς, η ταινία με την κεφαλή χρώματος κόκκινου, το λογότυπο της σχολής και το όνομα του δημιουργού). Επίσης, παρατηρούνται και τα κουμπιά - εργαλεία της γραμμής εργαλείων με μωβ χρώμα με τα οποία ο χρήστης μπορεί να ξεκινήσει.



Εικόνα 5.1. Αρχική σελίδα εφαρμογής

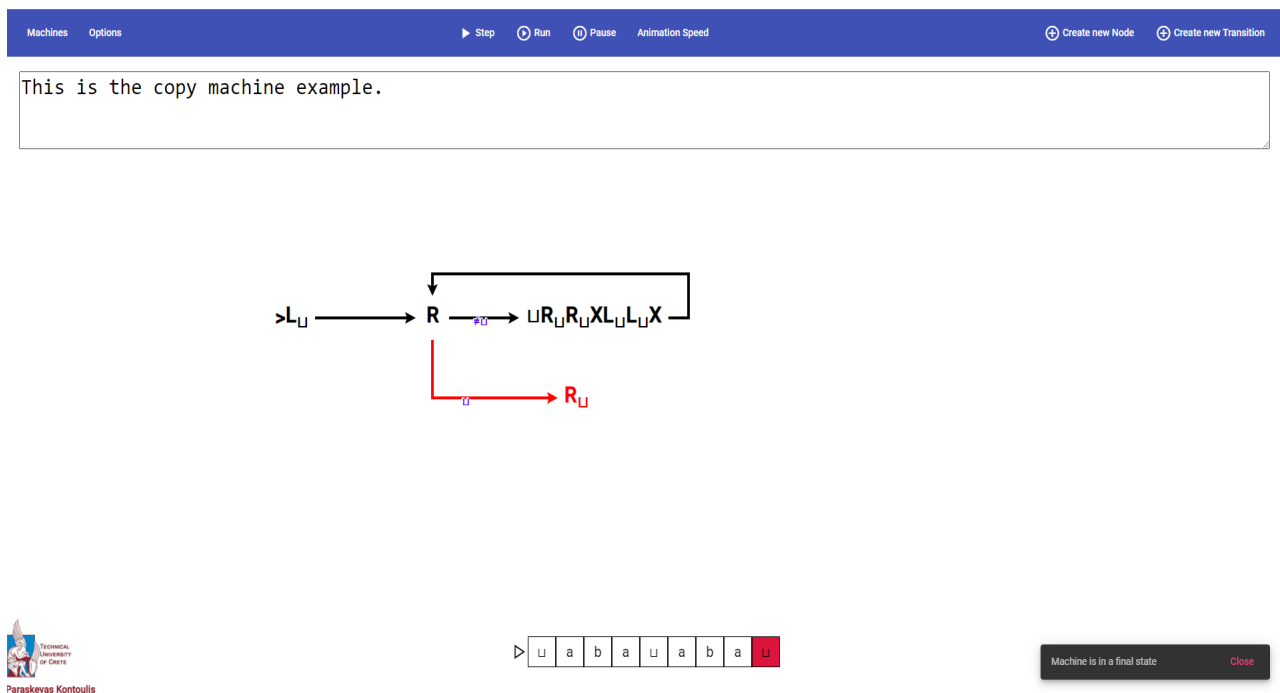
Στην Εικόνα 5.2 που ακολουθεί έχει γίνει φόρτωση έτοιμου παραδείγματος που υπάρχει διαθέσιμο για οποιονδήποτε χρήστη. Πρόκειται για την Μηχανή Turing αντιγραφής συμβολοσειράς, όπως αυτή

διδάσκεται ως παράδειγμα στο μάθημα της Θεωρίας Υπολογισμού. Είναι φανερό σε σύγκριση με την προηγούμενη εικόνα, όπου δεν είχε φορτωθεί κάποια μηχανή, ούτε είχε δημιουργηθεί νέα από τον χρήστη, ότι υπάρχουν αλλαγές στο κείμενο, τον καμβά και την ταινία μαζί με την κεφαλή. Παρατηρούνται τέσσερα καινούργια nodes, τα οποία περιλαμβάνουν απλές μηχανές ή συνδυασμό αυτών και ο συμβολισμός στο αριστερό κόκκινο node που δείχνει ότι πρόκειται για αρχικό node. Το κόκκινο χρώμα στο node αυτό συμβολίζει την μηχανή που θα εκτελεστεί μόλις γίνει χρήση του κουμπιού Run ή Step, δηλαδή το κόκκινο χρώμα σηματοδοτεί ανά πάσα στιγμή το τρέχον σημείο της εκτέλεσης της προσομοίωσης της μηχανής. Ένα ακόμα καινούργιο στοιχείο είναι οι μεταβάσεις μαζί με τις προϋποθέσεις - σύμβολα που οδηγούν σε επόμενα nodes. Τέλος, η ταινία έχει φορτωθεί με μία συμβολοσειρά και η κεφαλή έχει μετακινηθεί στο επιθυμητό σημείο.



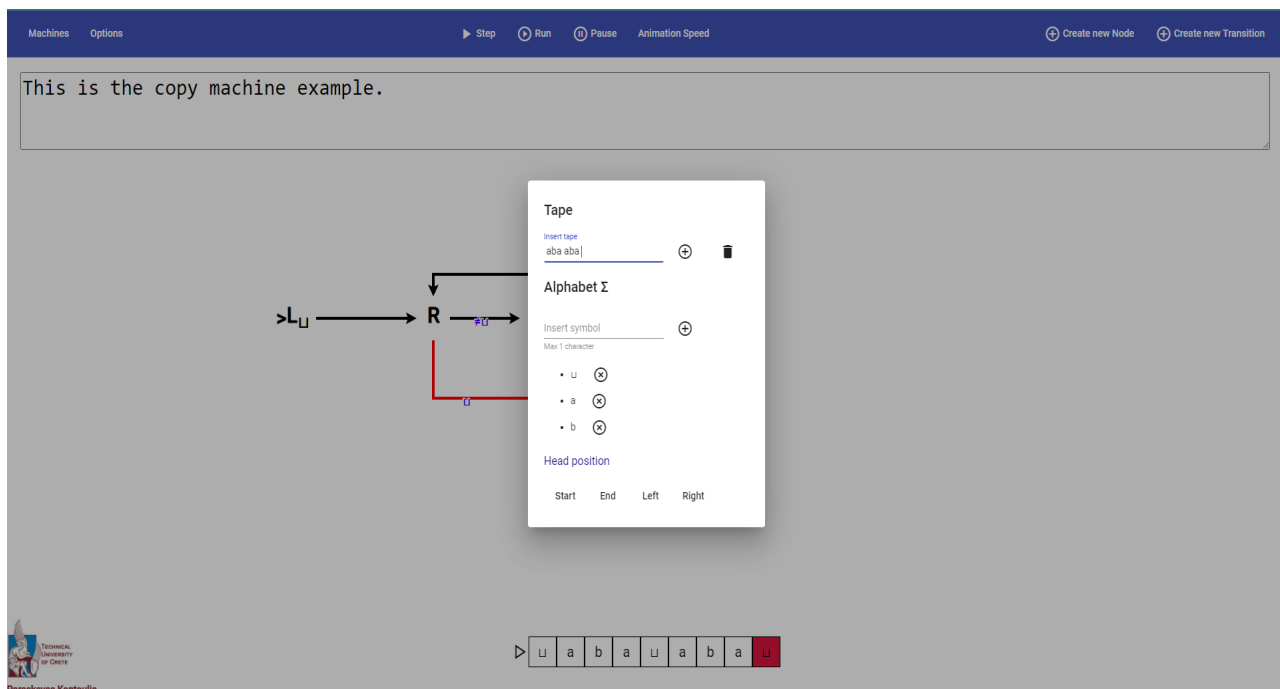
Εικόνα 5.2. Παράδειγμα έναρξης Μηχανής Turing

Παρακάτω (Εικόνα 5.3), φαίνεται το τέλος της λειτουργίας της μηχανής αντιγραφής συμβολοσειράς. Η μηχανή τελείωσε επιτυχώς, καθώς η αλλαγή στην ταινία το επιβεβαιώνει. Κάτω δεξιά υπάρχει και το αντίστοιχο μήνυμα από το σύστημα ότι η μηχανή έχει τελειώσει.



Εικόνα 5.3. Παράδειγμα τερματισμού Μηχανής Turing

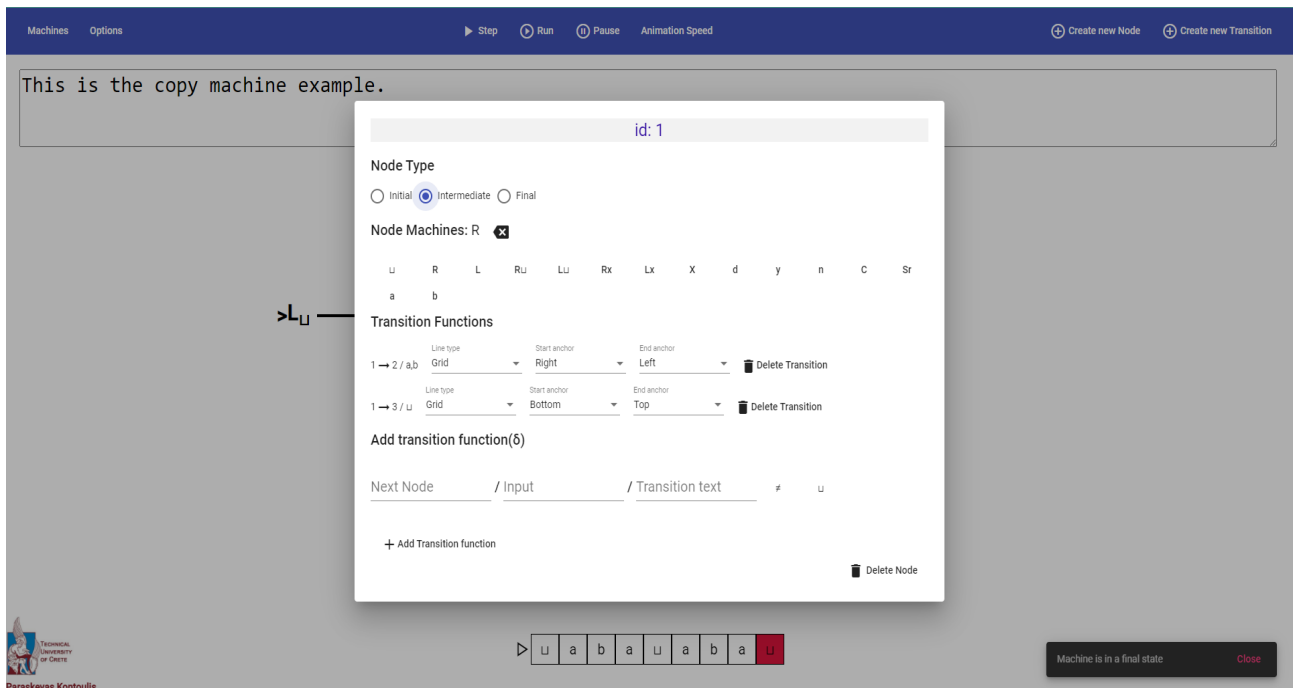
Στην Εικόνα 5.4 απεικονίζεται το tape-data.component, όπως αυτό περιγράφηκε σε προηγούμενο σημείο. Ο χρήστης μπορεί να διαχειριστεί στο component αυτό το αλφάβητο της μηχανής, την συμβολοσειρά εισόδου και την θέση της κεφαλής στην ταινία.



Εικόνα 5.4. Παράδειγμα διαχείρισης ταινίας κεφαλής και αλφαβήτου Μηχανής Turing

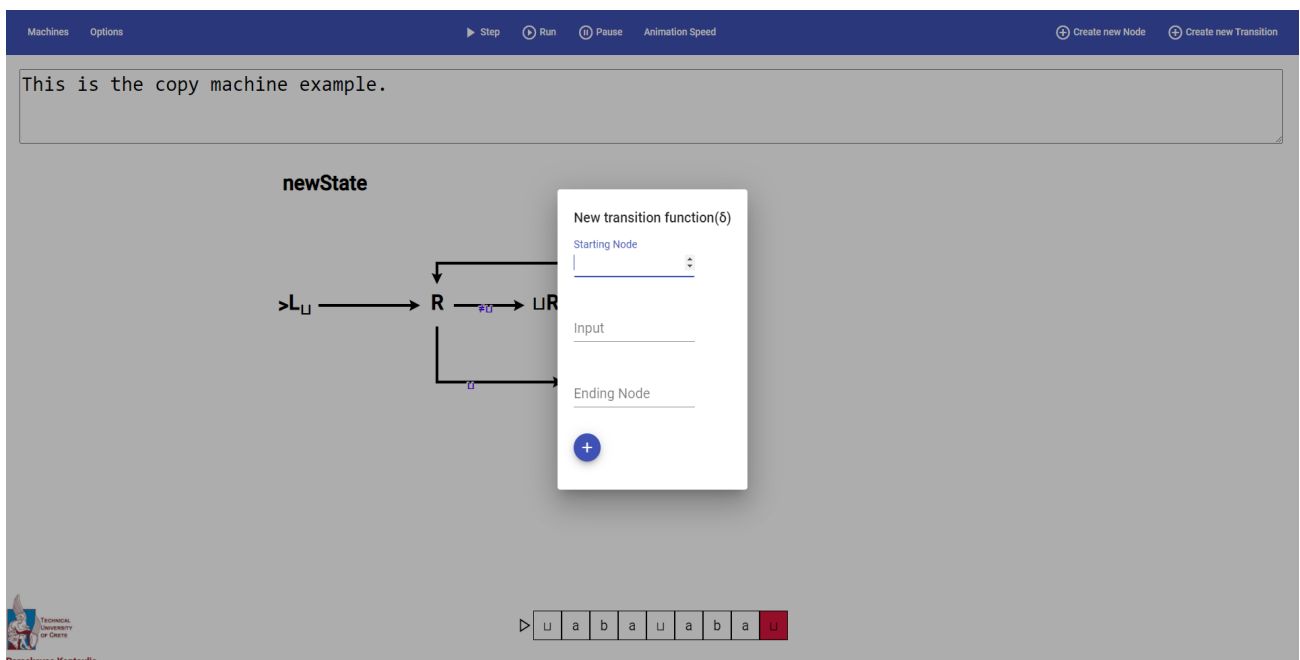
Στην Εικόνα 5.5 απεικονίζεται το state-data.component. Ο χρήστης μπορεί να διαχειριστεί τον τύπο του node (αρχικό, ενδιάμεσο, τελικό), ποιές βασικές μηχανές θα «τρέξουν», όταν φτάσει στο node αυτό, ποιές είναι οι μεταβάσεις που ξεκινούν από τον κόμβο που έχει επιλεγεί μαζί με επιλογές για

την καλύτερη σχεδίαση του σχήματος της Μηχανής Turing, καθώς και το εργαλείο για την δημιουργία μιας νέας μετάβασης. Κάτω δεξιά βρίσκεται η επιλογή διαγραφής του επιλεγμένου κόμβου.



Εικόνα 5.5. Παράδειγμα διαχείρισης ενός node

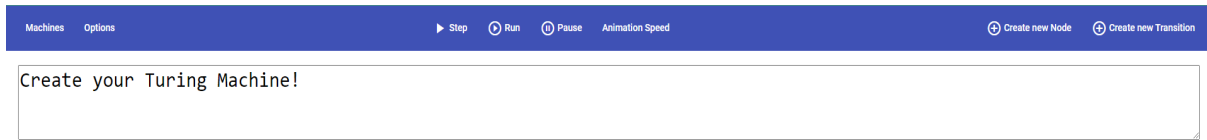
Στην Εικόνα 5.6 απεικονίζεται το new-delta-data.component, στο οποίο ο χρήστης μπορεί να σχεδιάσει μια καινούργια μετάβαση, συμπληρώνοντας τα απαιτούμενα δεδομένα.



Εικόνα 5.6. Παράδειγμα δημιουργίας μετάβασης

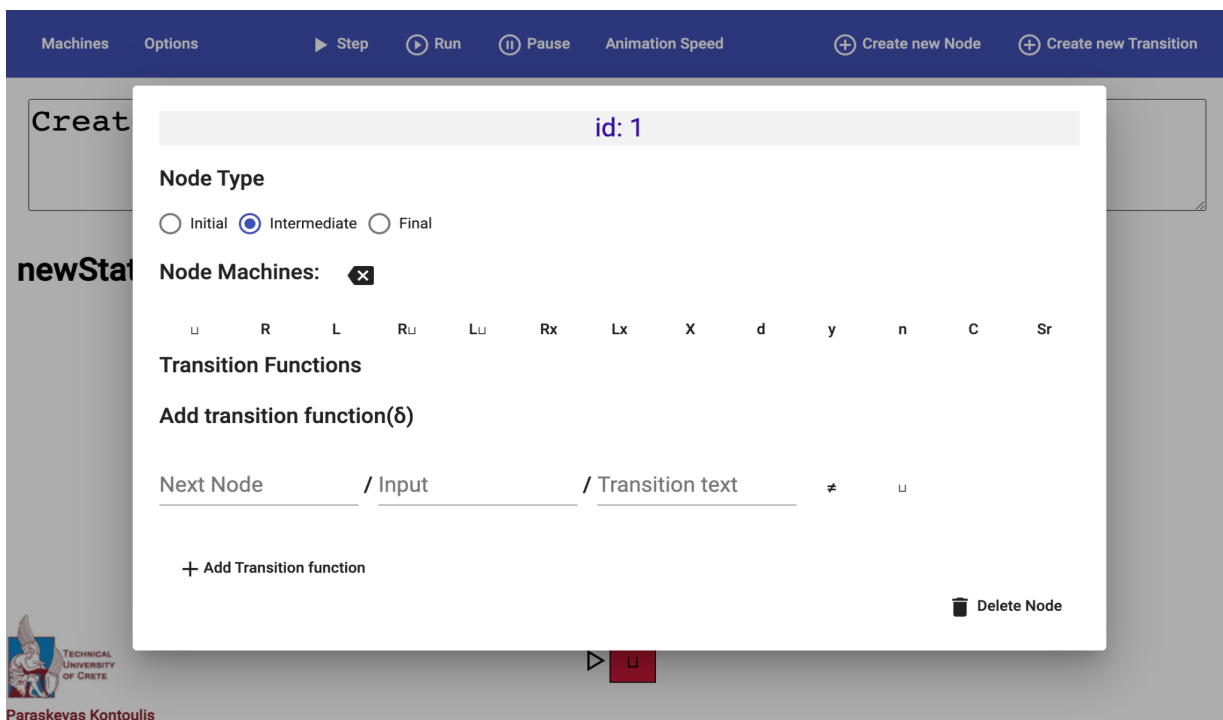
5.2 Δημιουργία Μηχανής Turing

- Για τη δημιουργία μιας νέας μηχανής, αρχικά ο καμβάς είναι κενός.



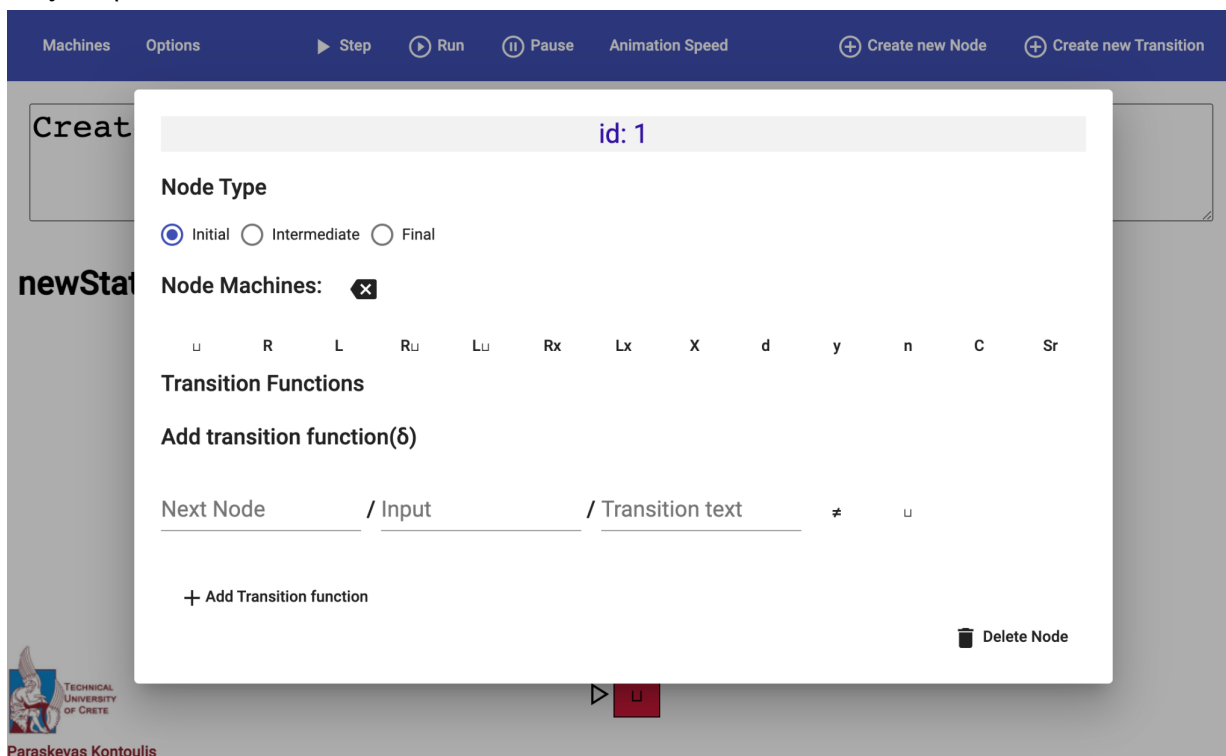
Εικόνα 5.7. Αρχική οθόνη

- Πατώντας το κουμπί “Create new Node” δημιουργούμε τη βασική μηχανή.



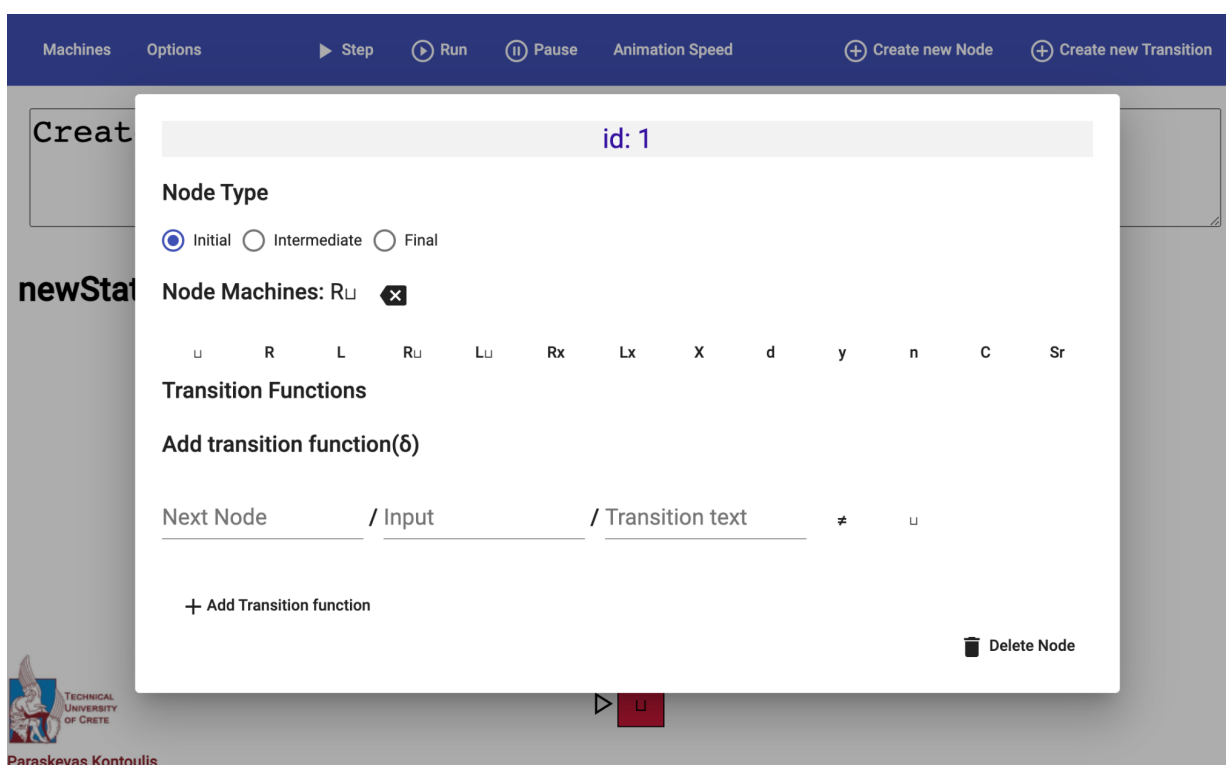
Εικόνα 5.8. Δημιουργία node

- Επιλέγουμε να είναι Initial η βασική μηχανή, ώστε να γνωρίζει ο προσομοιωτής από που θα ξεκινήσει.



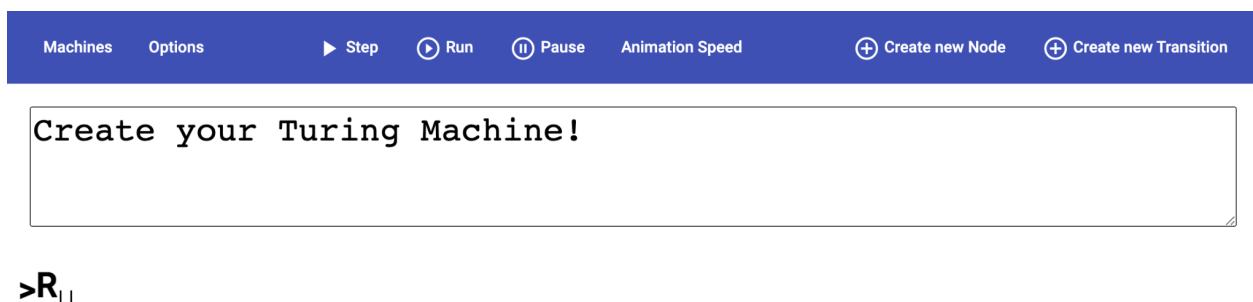
Εικόνα 5.9. Μετατροπή σε αρχική βασική μηχανή

- Θέλουμε να περιέχει την μηχανή ανεύρεσης πρώτου κενού δεξιά, οπότε στη λίστα με τα node machines επιλέγουμε τη μηχανή $R\sqcup$.



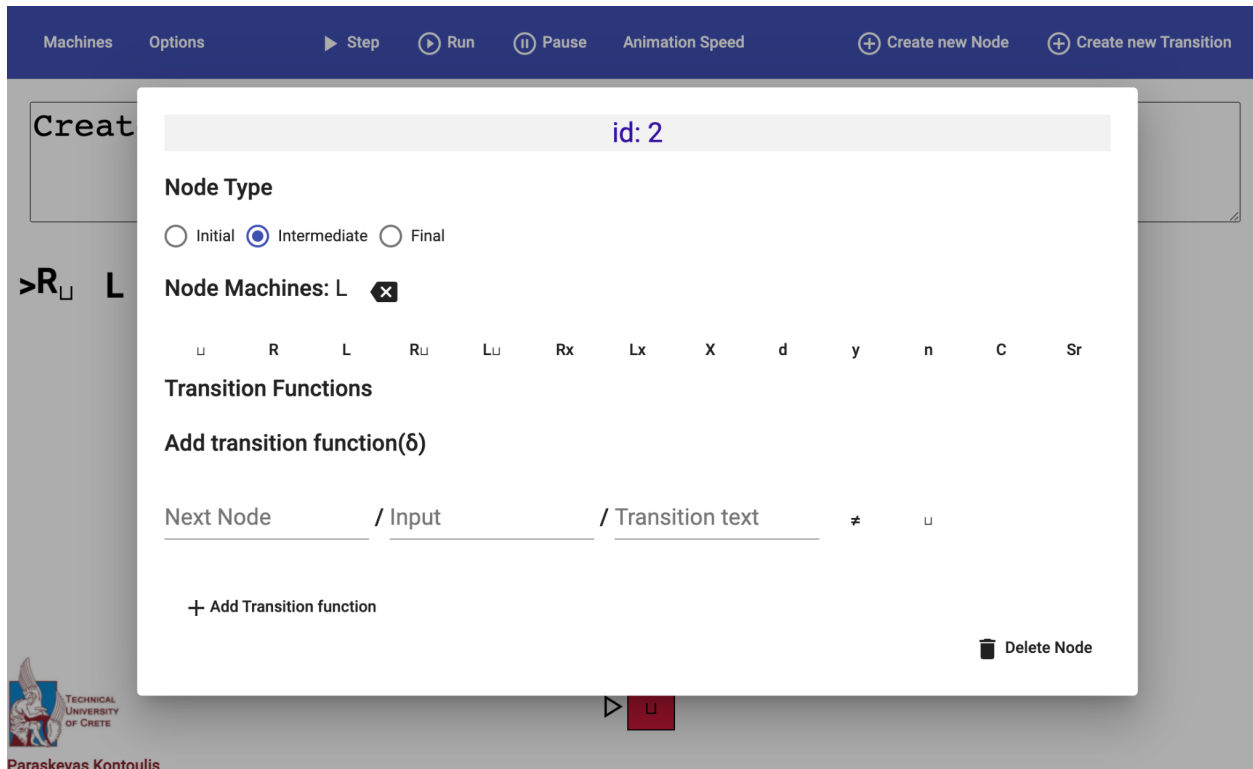
Εικόνα 5.10. Προσθήκη μηχανής εύρεσης πρώτου κενού δεξιά

- Όπως φαίνεται στην Εικόνα 5.10, έχουμε φτιάξει επιτυχώς τη βασική μηχανή.

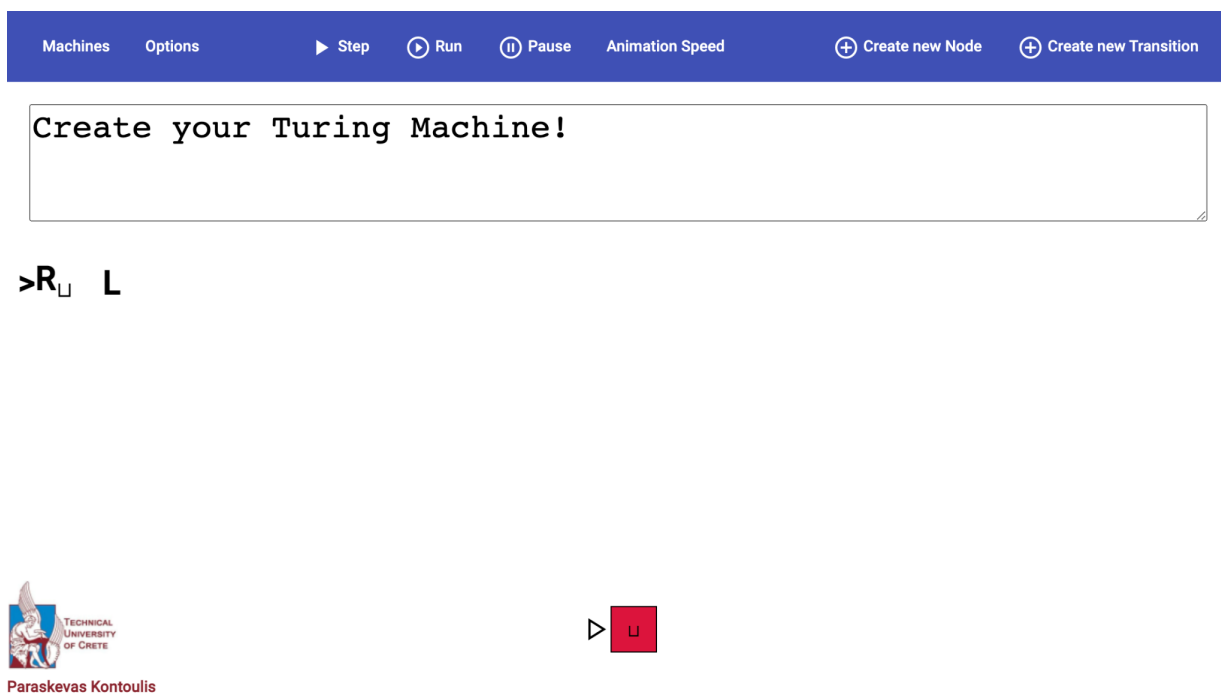


Εικόνα 5.11. Πρώτο node

- Με την ίδια διαδικασία δημιουργούμε και τη δεύτερη μηχανή.

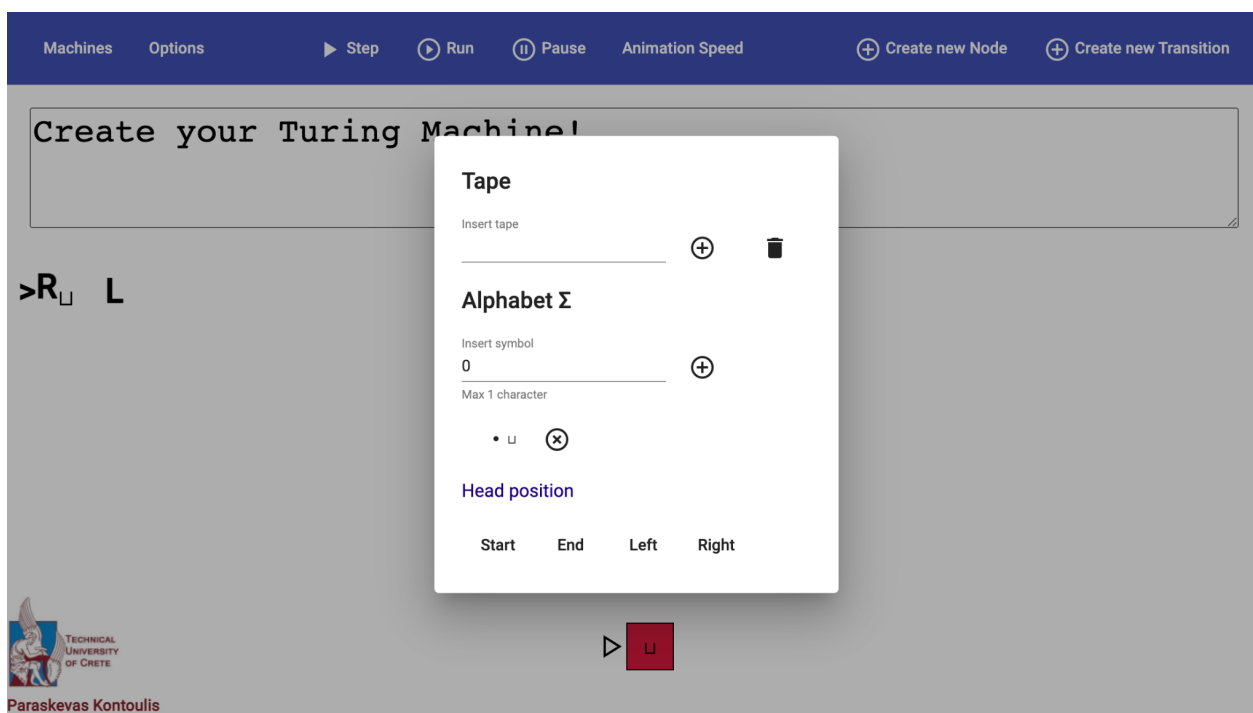


Εικόνα 5.12. Δημιουργία δεύτερου node

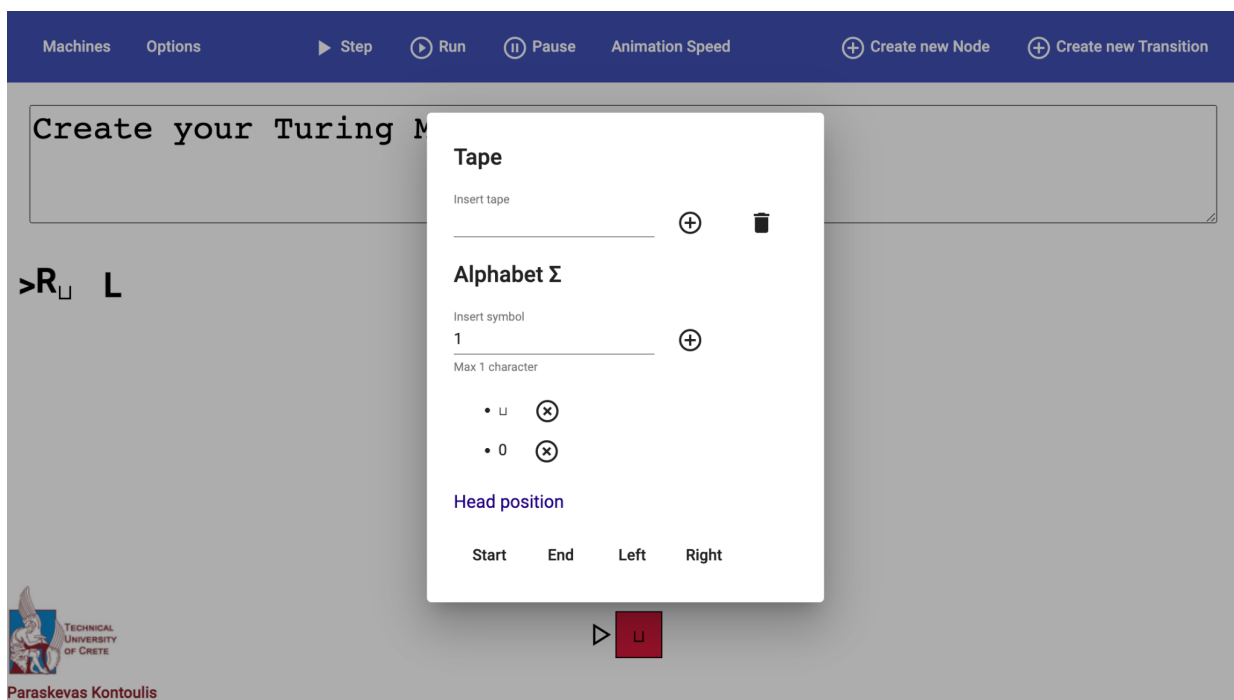


Εικόνα 5.13. Αποτέλεσμα στον καμβά

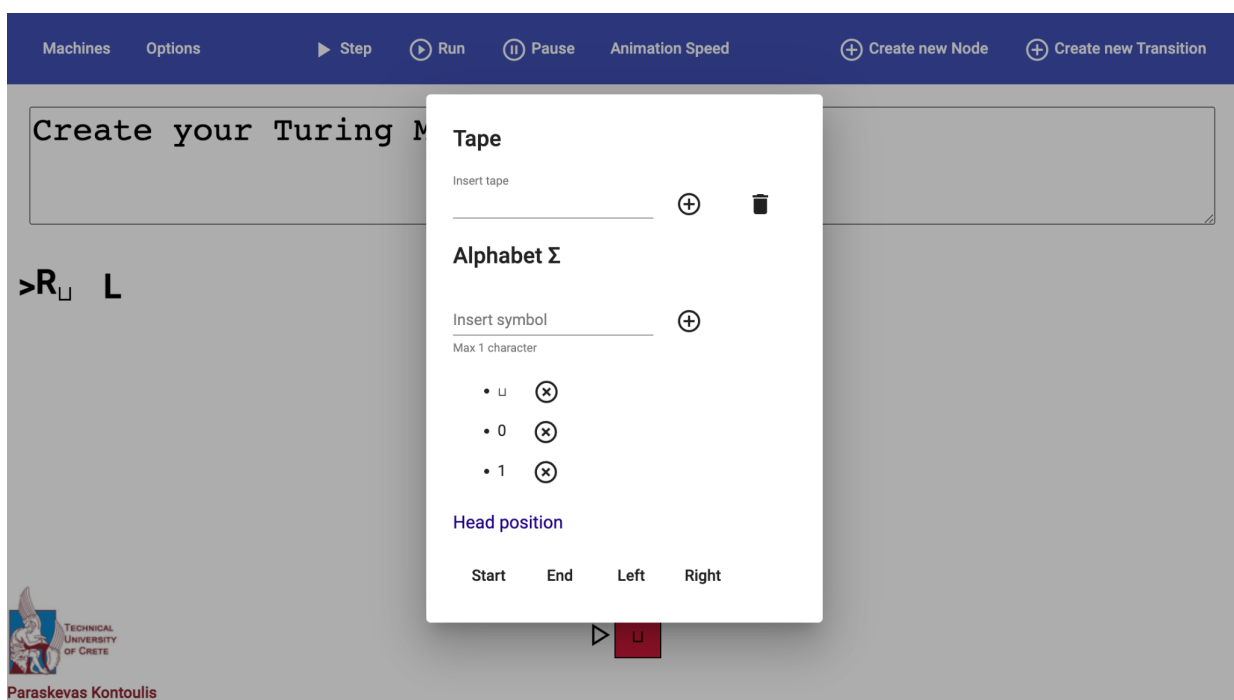
- Σε αυτό το σημείο θέλουμε να προσθέσουμε μηχανές καταγραφής των συμβόλων 0 και 1. Γι' αυτόν τον λόγο πρέπει να προσθέσουμε στο αλφάβητο τα σύμβολα 0 και 1, ώστε να εμφανιστεί η επιλογή των μηχανών αυτών στο παράθυρο επεξεργασίας κάθε Node. Για να γίνει αυτό, πρέπει να επιλέξουμε το component της ταινίας και να εμφανιστούν οι επιλογές.



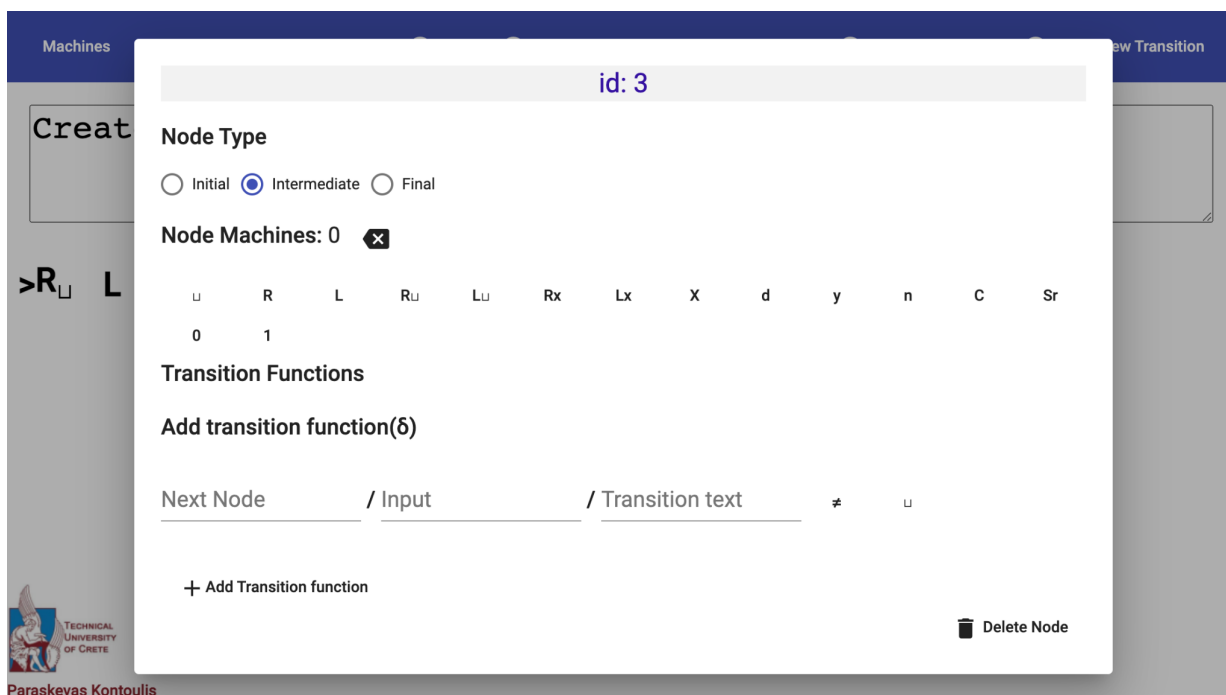
Εικόνα 5.14. Προσθήκη συμβόλου 0



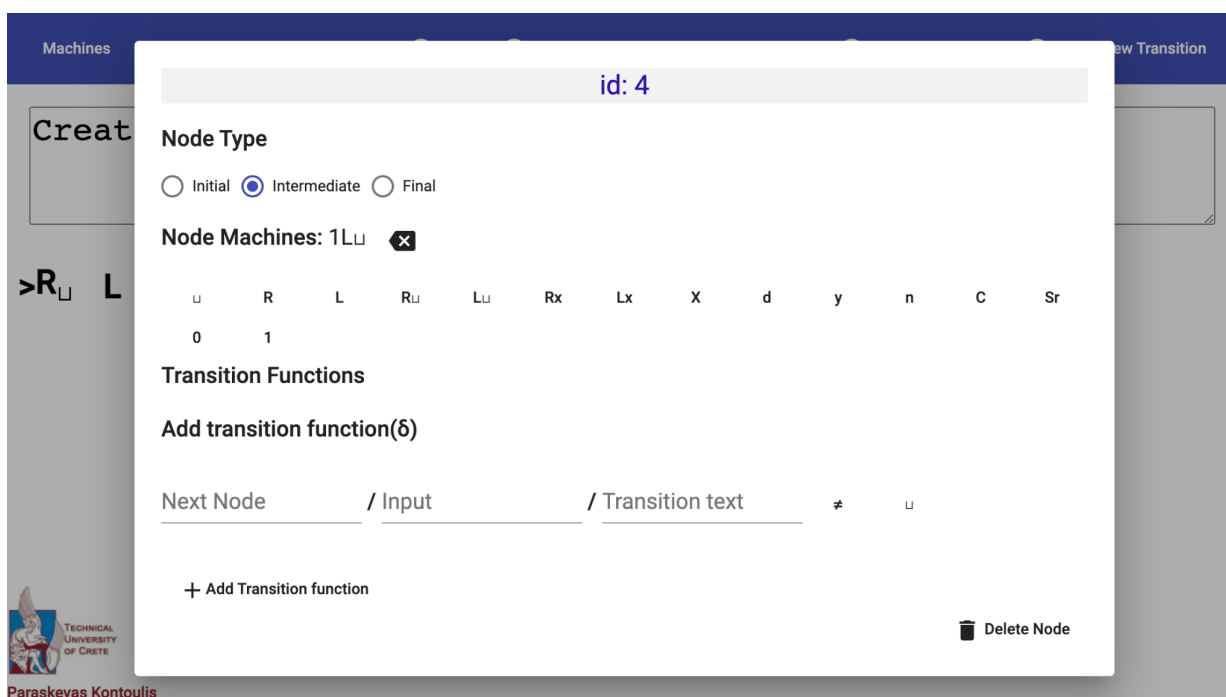
Εικόνα 5.15. Προσθήκη συμβόλου 1



Εικόνα 5.16. Αποτέλεσμα στο αλφάβητο

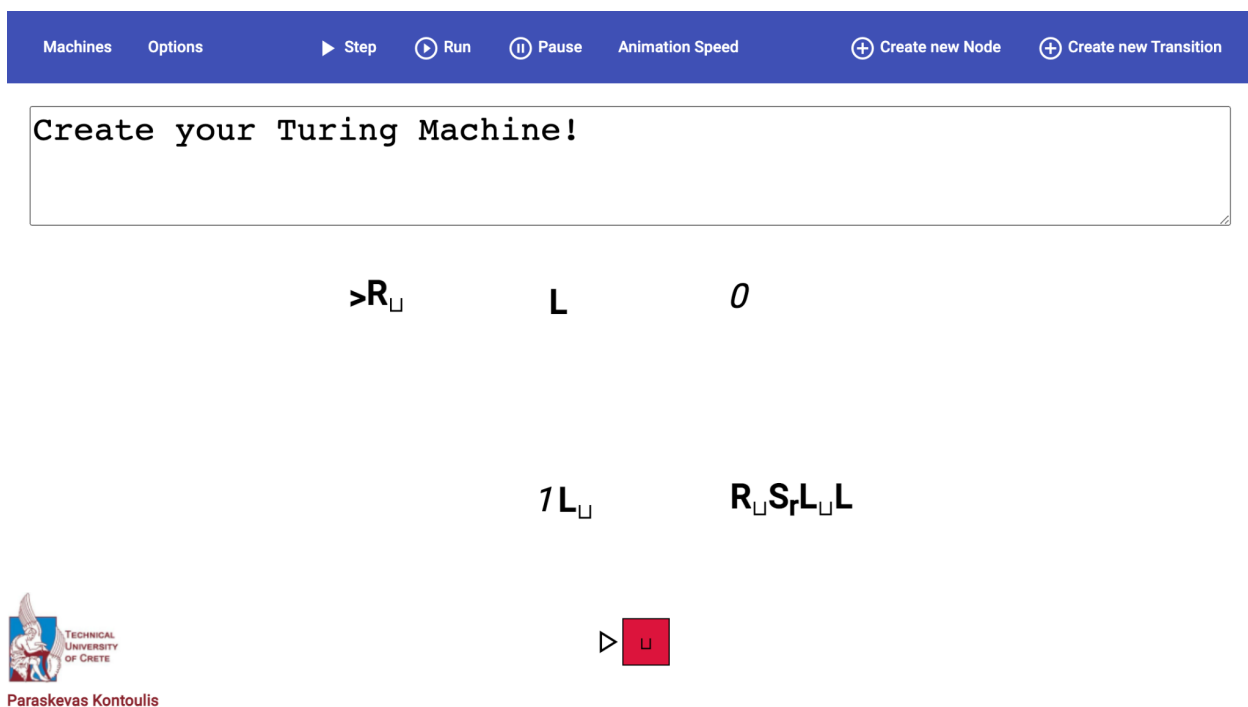


Εικόνα 5.17. Προσθήκη τρίτου node



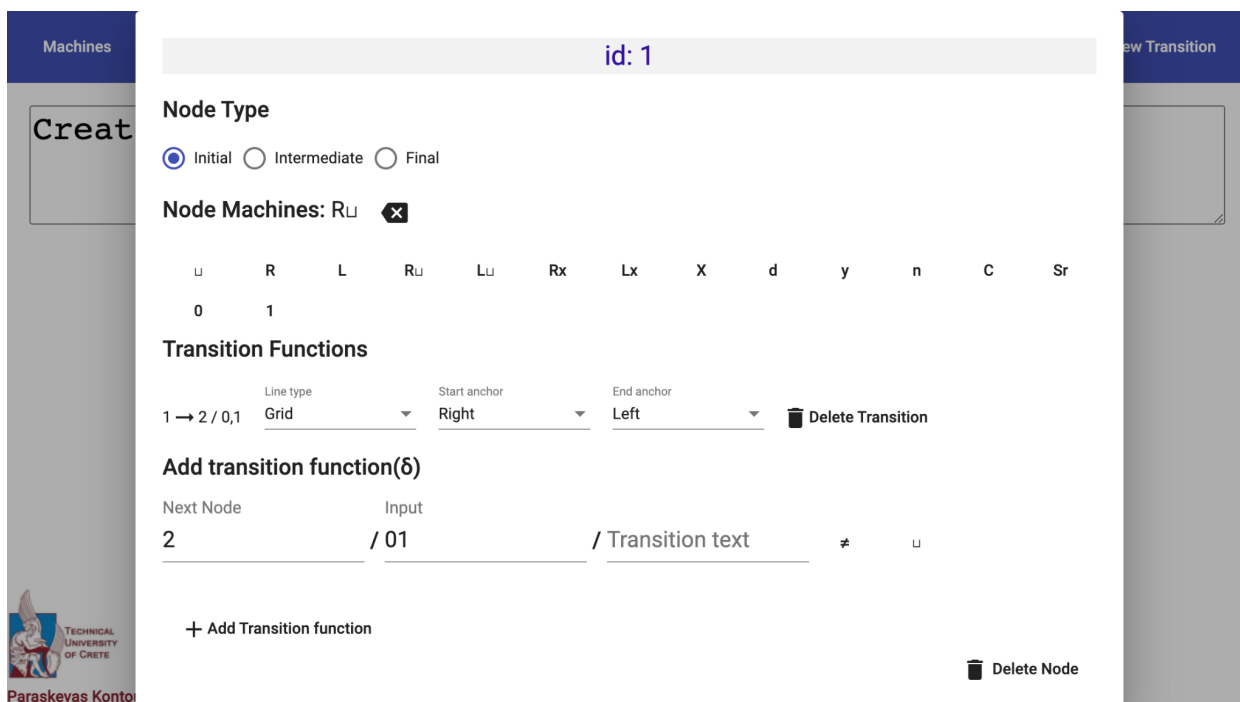
Εικόνα 5.18. Προσθήκη τέταρτου node

- Σε αυτό το σημείο και ακολουθώντας την διαδικασία που περιγράφηκε για τις πρώτες δύο μηχανές, δημιουργήσαμε όλα τα Nodes που θα χρειαστούμε για να υλοποιήσουμε την Μηχανή Turing.

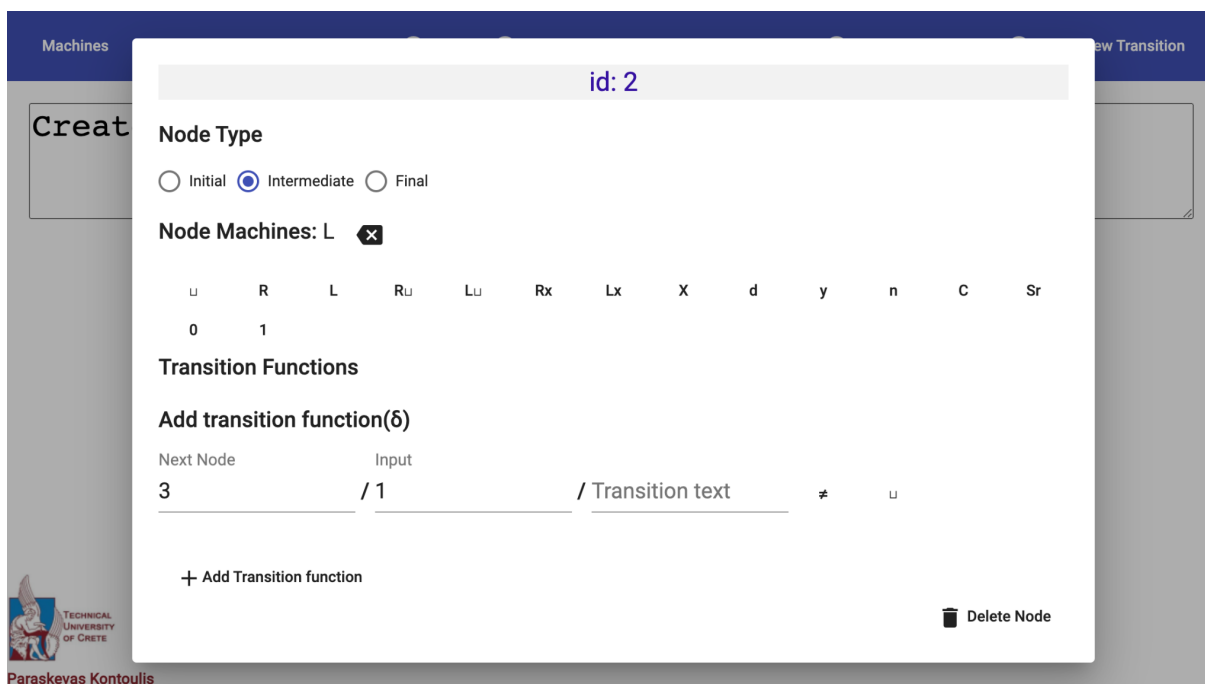


Εικόνα 5.19. Σύνολο των node

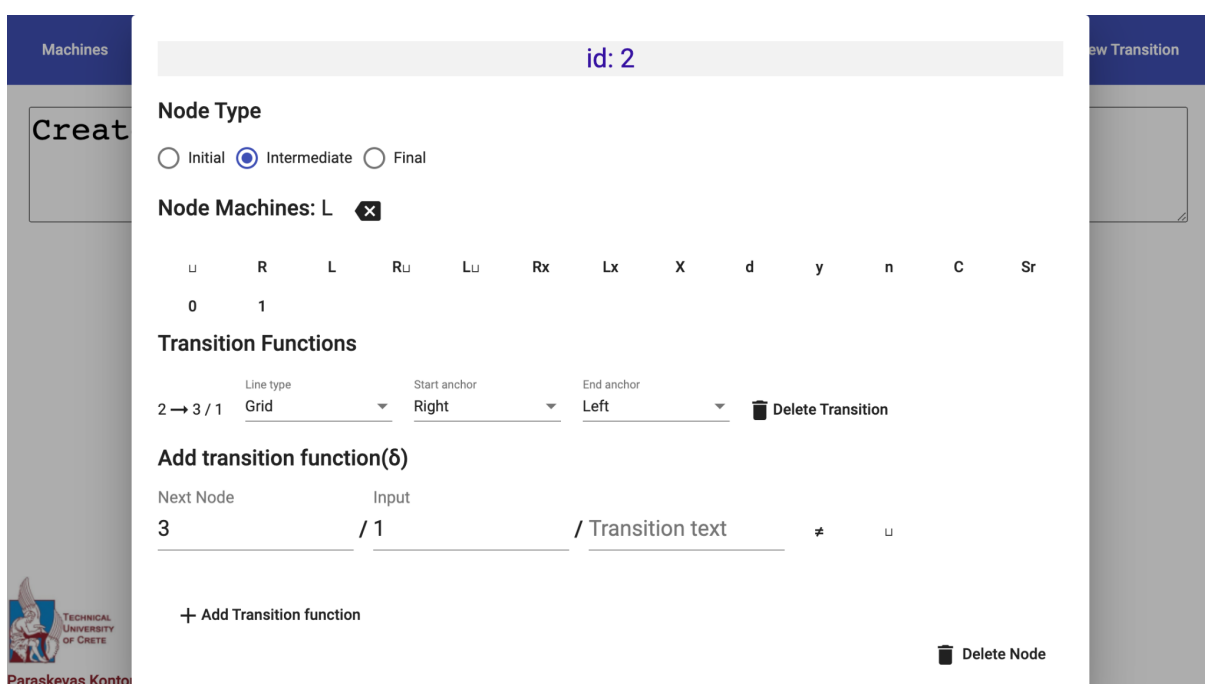
- Η μηχανή απαιτεί μεταβάσεις για να λειτουργήσει. Η δημιουργία αυτών επιτυγχάνεται επιλέγοντας το node από το οποίο θέλουμε να ξεκινάει η κάθε μετάβαση. Αφού το επιλέξουμε, θα ανοίξει το παράθυρο επεξεργασίας του Node. Στο πεδίο transition function μπορούμε να συμπληρώσουμε τα στοιχεία που είναι απαραίτητα για τη δημιουργία της κάθε μετάβασης, όπως φαίνεται παρακάτω.



Εικόνα 5.20. Προσθήκη μετάβασης



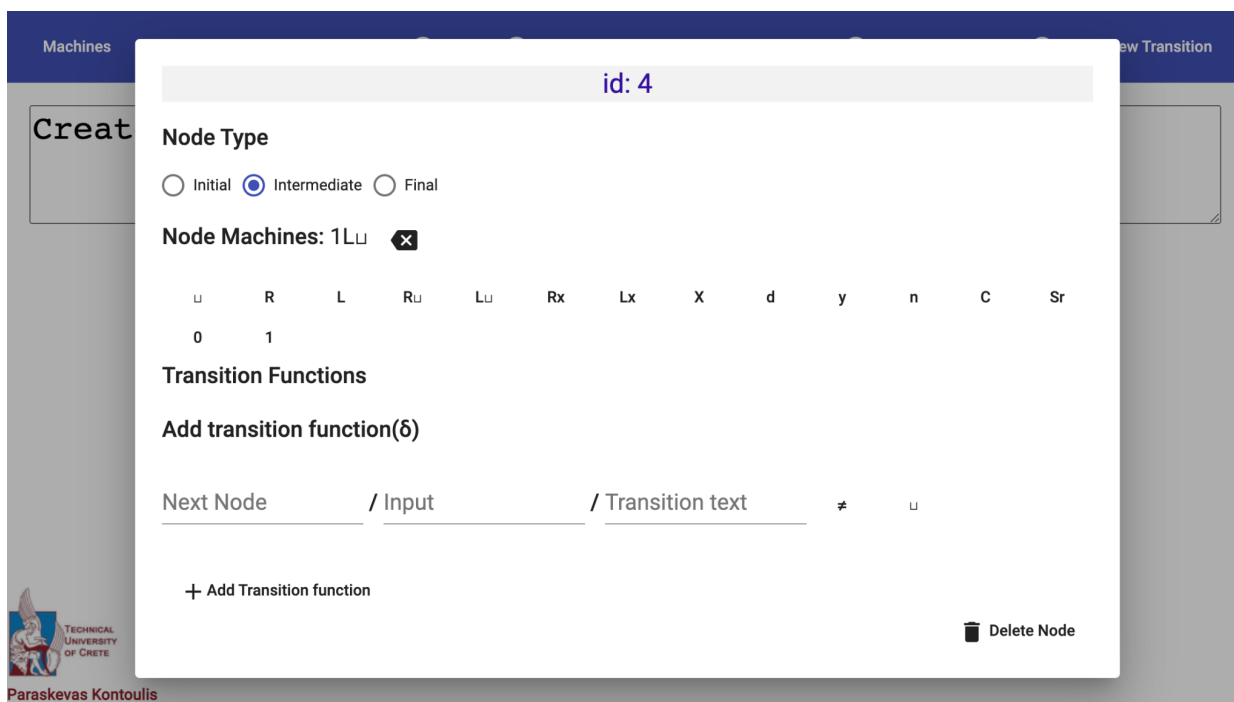
Εικόνα 5.21. Επιλογή δεύτερου node



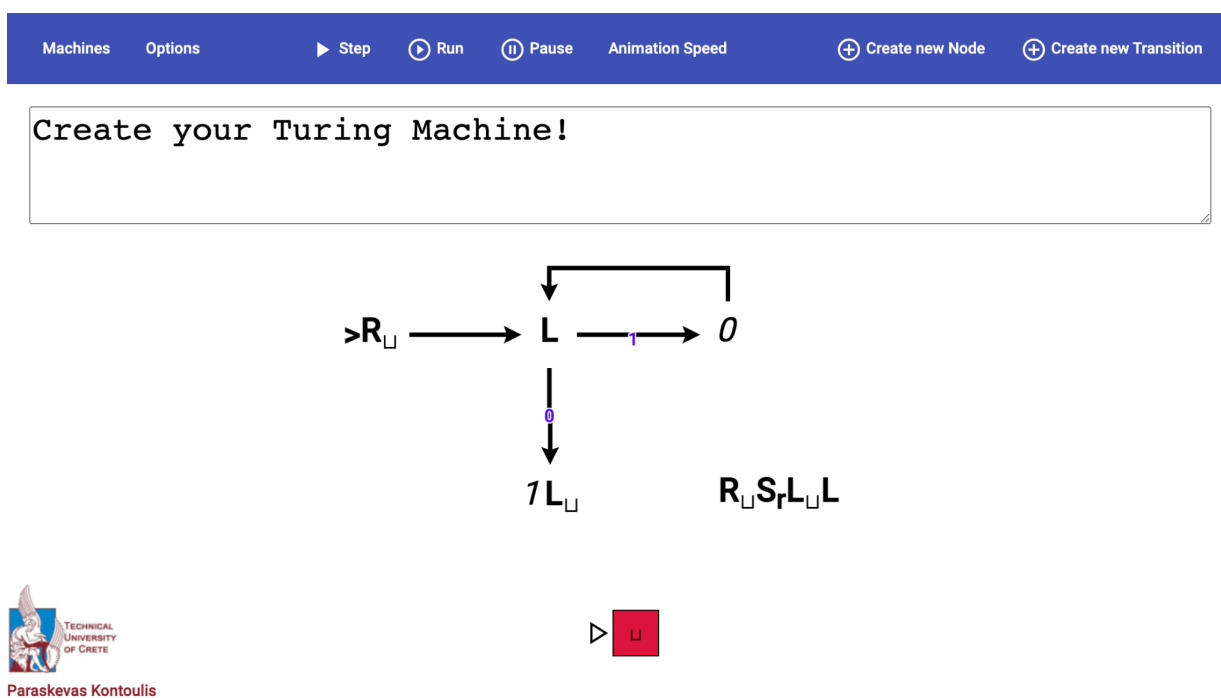
Εικόνα 5.22. Αποτέλεσμα προσθήκης

Εικόνα 5.23. Αποτέλεσμα στον καμβά μετά την προσθήκη

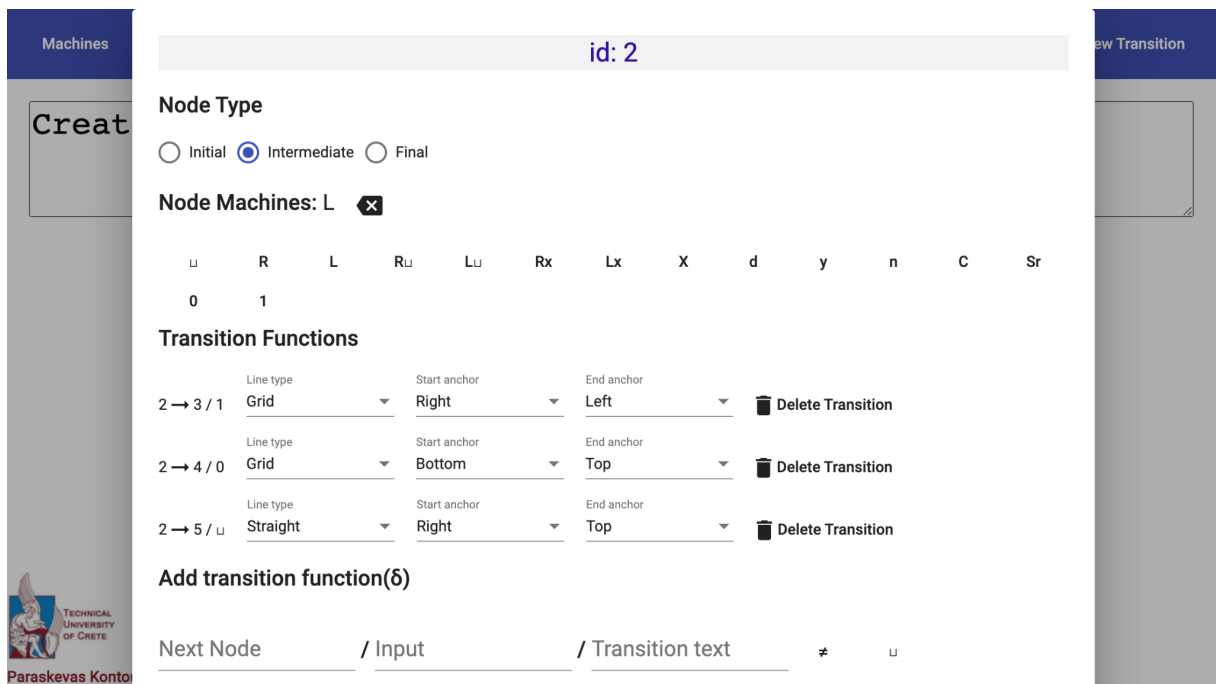
Εικόνα 5.24. Προσθήκη επιπλέον μετάβασης



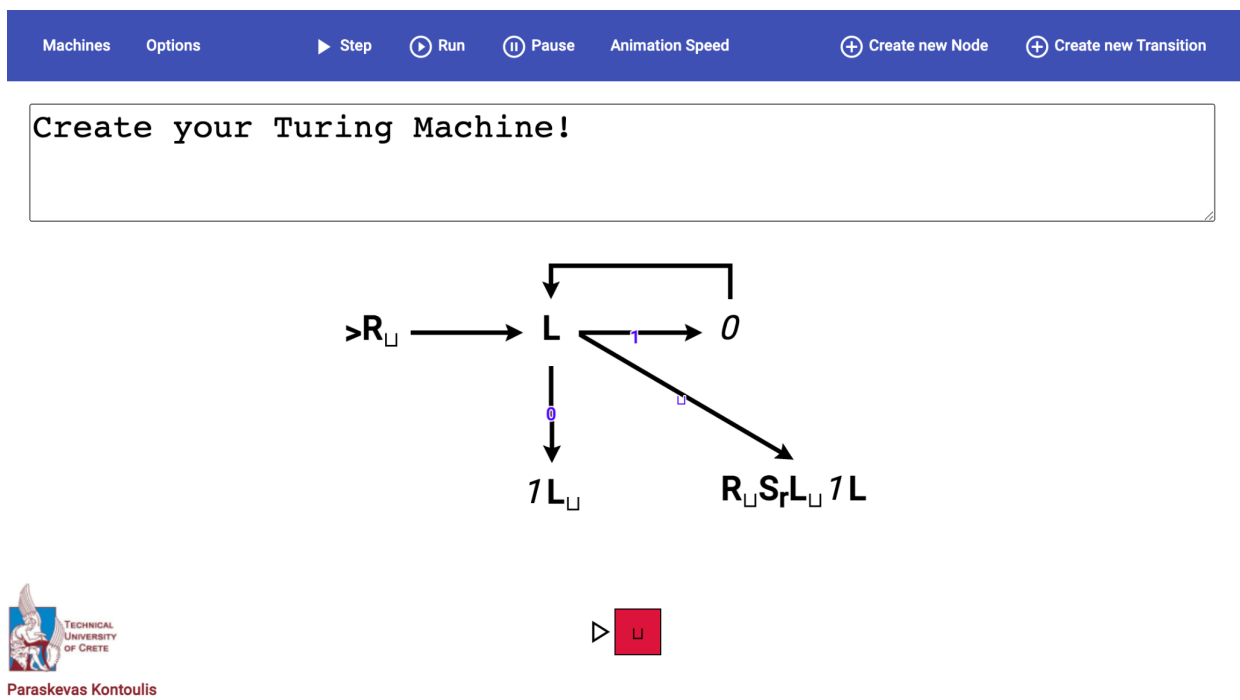
Εικόνα 5.25. Επιλογή τέταρτου node



Εικόνα 5.26. Αποτέλεσμα στον καμβά μετά την προσθήκη των μεταβάσεων μέχρι τώρα



Εικόνα 5.27. Επιλογή δεύτερου node και προσθήκη μετάβασης



Εικόνα 5.28. Τελική Μηχανή Turing

id: 4

Node Type

☐ Initial ☐ Intermediate ☒ Final

Node Machines: 1 ✕

u	R	L	Ru	Lu	Rx	Lx	X	d	y	n	C	Sr
0	1											

Transition Functions

Add transition function(δ)

Next Node / Input / Transition text = u

+ Add Transition function

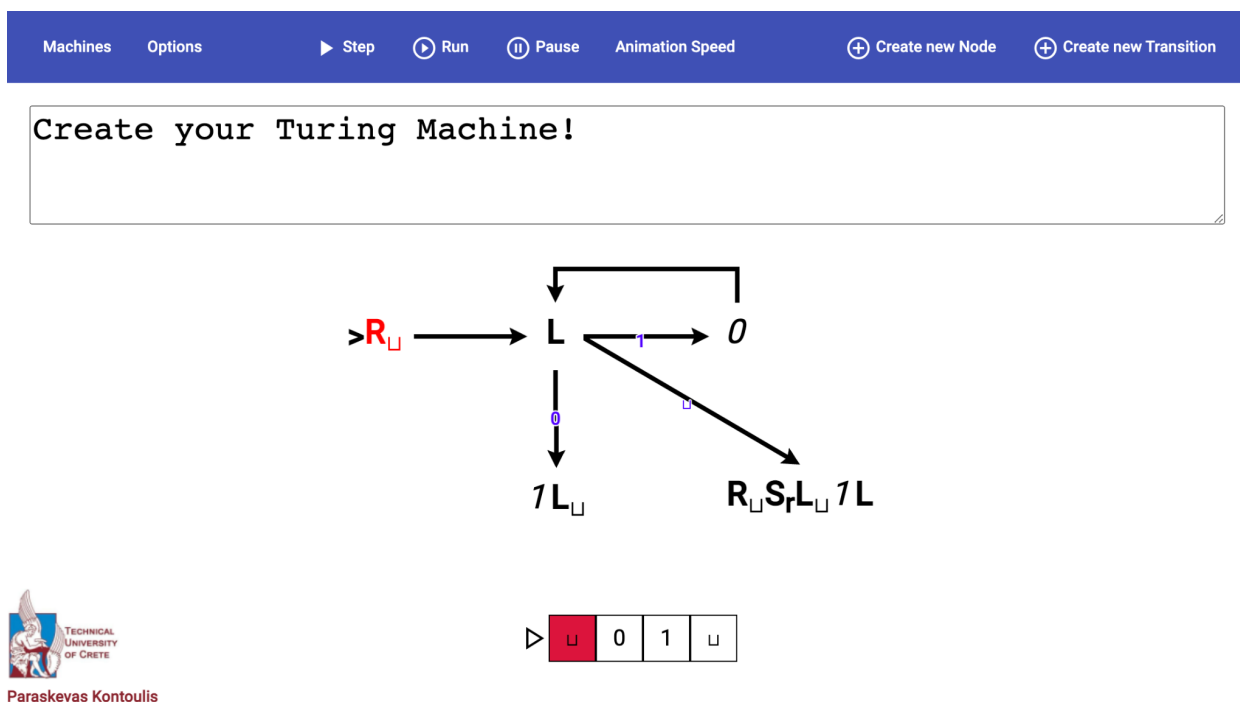
🗑 Delete Node

TECHNICAL UNIVERSITY OF CRETE

Paraskevas Kontoulis

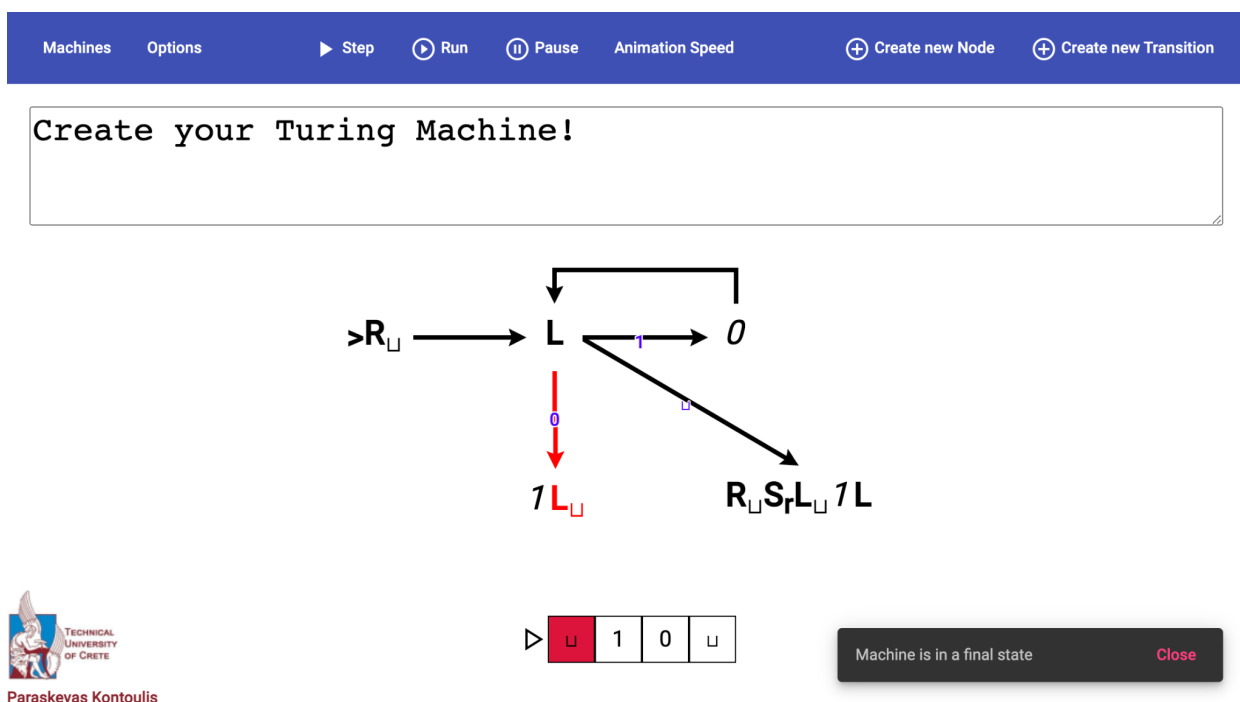
Εικόνα 5.31. Δεδομένα node no4

- Σε αυτό το σημείο, δημιουργήσαμε όλες τις μεταβάσεις που χρειαζόμασταν για τη Μηχανή Turing που θέλαμε να υλοποιήσουμε. Επίσης, θα προσθέσουμε επιλέγοντας την ταινία, την είσοδο που επιθυμούμε, όπως φαίνεται στην επόμενη εικόνα.



Εικόνα 5.32. Εκκίνηση Μηχανής Turing

- Αφού κατασκευάσαμε τα nodes και τις μεταβάσεις, προσθέσαμε στην είσοδο τη συμβολοσειρά που επιθυμούσαμε. Με τη χρήση των κουμπιών της μπάρας εργαλείων, έγινε έναρξη της Μηχανής Turing και στη τελευταία εικόνα φαίνεται το αποτέλεσμα. Η μηχανή αφορά στην αναδρομική συνάρτηση $\text{succ}(n) = n+1$.



Εικόνα 5.33. Τερματισμός Μηχανής Turing

Σε αυτήν την ενότητα, δείξαμε μια τυπική χρήση του εργαλείου σχεδίασης και προσομοίωσης που έχει δημιουργηθεί. Όπως είδαμε, η διεπαφή χρήστη είναι διαισθητική και εύκολη στην πλοήγηση, επιτρέποντας στους χρήστες να έχουν γρήγορη πρόσβαση στις διάφορες δυνατότητες και λειτουργίες της εφαρμογής. Ο προσομοιωτής παρέχει μια ρεαλιστική και ευχάριστη εμπειρία που επιτρέπει στους χρήστες να εξερευνήσουν τις δυνατότητες της εφαρμογής και να δοκιμάσουν διάφορα σενάρια. Οι απεικονίσεις είναι σαφείς και ενημερωτικές, δίνοντας στους χρήστες μια ολοκληρωμένη εικόνα των αποτελεσμάτων και των αποτελεσμάτων της εφαρμογής. Συνολικά, όλα όσα έχουμε δει σε αυτήν την ενότητα δείχνουν την αποτελεσματικότητα και τη φιλικότητα της εφαρμογής προς τον χρήστη, καθιστώντας την ένα ανεκτίμητο εργαλείο για μια ποικιλία περιπτώσεων.

6. *Εμπειρία Χρηστών*

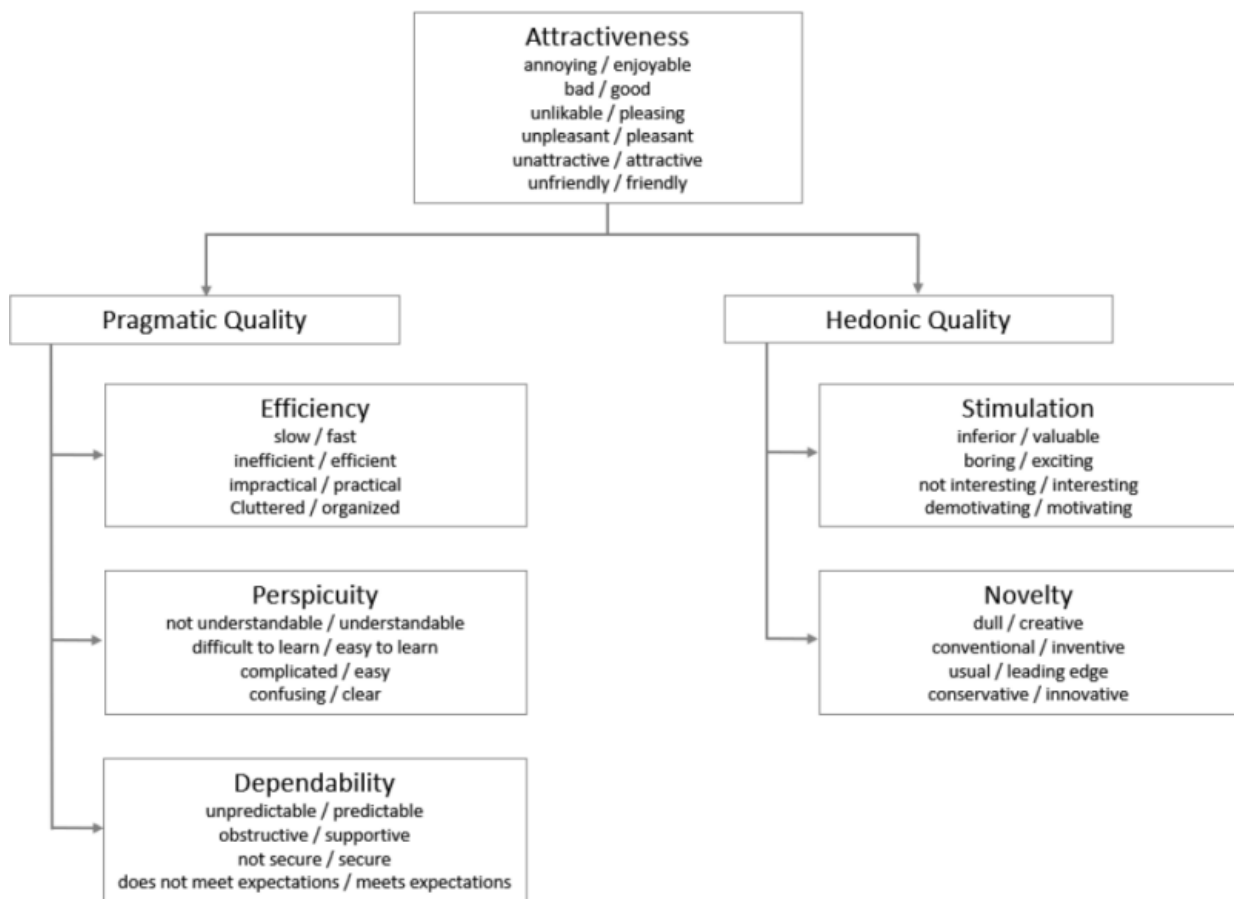
6.1. *Ερωτηματολόγιο*

Όσον αφορά στην ανάπτυξη μιας εφαρμογής, αξιολογούμενος είναι ο ρόλος του user evaluation μέσω χρήσης, αλλά και του user experience questionnaire μέσω ανάλυσης. Αναλυτικότερα, μια έρευνα User Experience (UX) συγκεντρώνει τις απαντήσεις των χρηστών σε συγκεκριμένες ερωτήσεις, συντείνοντας στην κατανόηση της συμπεριφοράς τους από τον προγραμματιστή. Δίνει στην ουσία στον προγραμματιστή μια συνολική εικόνα για την άποψη των χρηστών σχετικά με το πρόγραμμά του και την ποιότητά του.

Το εκάστοτε ερωτηματολόγιο απαρτίζεται από 26 ερωτήσεις, καταταγμένες σε κατηγορίες και οι οποίες πρόκειται να εμφανίσουν αποτελέσματα για τις εξής 6 διαφορετικές κατηγορίες:

- **Attractiveness:** Ευρύτερη γνώμη για το προϊόν, κατά ποσό είναι ελκυστικό στους χρήστες.
- **Perspicuity:** Ο παράγοντας που επαληθεύει εάν το πρόγραμμα θεωρείται ευχάριστο ή δύσχρηστο.
- **Efficiency:** Αποδοτικότητα του προγράμματος.
- **Dependability:** Σημαντικά θέματα σχετικά με την αλληλεπίδραση χρήστη και προγράμματος.
- **Stimulation:** Προτροπή των χρηστών για χρήση της εφαρμογής.
- **Novelty:** Ο βαθμός καινοτομίας-ευρηματικότητας του λογισμικού.

Ακολουθεί γραφικό διάγραμμα των παραπάνω παραγόντων, ώστε να δοθεί μια πιο επεξηγηματική εικόνα σχετικά με το τι εκφράζει η κάθε κλίμακα.



Εικόνα 6.1. Γραφικό διάγραμμα παραγόντων

Πιο συγκεκριμένα, σύμφωνα με το παραπάνω διάγραμμα (Εικόνα 39), καθίσταται εμφανές πως οι 26 ερωτήσεις διασπώνται σε κατηγορίες, καθώς επίσης και ότι η εκάστοτε κατηγορία αποδίδει ποσοστιαία ένα καθορισμένο scale. Επιπροσθέτως, η κλίμακα attractiveness αποτελείται από δύο υποκατηγορίες, την Pragmatic Quality (Efficiency, Perspicuity, Dependability) και την Hedonic (Stimulation and Novelty). Έτσι λοιπόν, το κάθε scale αντικατοπτρίζει ένα υποσύνολο ερωτήσεων που περιλαμβάνει το κάθε παραλληλόγραμμο.

	1	2	3	4	5	6	7		
ενοχλητικό	○	○	○	○	○	○	○	απολαυστικό	1
δυσνόητο	○	○	○	○	○	○	○	κατανοητό	2
δημιουργικό	○	○	○	○	○	○	○	αναποτελεσματικό	3
εύκολο στη μάθηση	○	○	○	○	○	○	○	δύσκολο στη μάθηση	4
πολύτιμο	○	○	○	○	○	○	○	υποδεέστερο	5
βαρετό	○	○	○	○	○	○	○	συναρπαστικό	6
αδιάφορο	○	○	○	○	○	○	○	ενδιαφέρον	7
απρόβλεπτο	○	○	○	○	○	○	○	προβλέψιμο	8
γρήγορο	○	○	○	○	○	○	○	αργό	9
εφευρετικό	○	○	○	○	○	○	○	συμβατικό	10
παρελκυστικό	○	○	○	○	○	○	○	υποστηρικτικό	11
καλό	○	○	○	○	○	○	○	κακό	12
περίπλοκο	○	○	○	○	○	○	○	εύκολο	13
αντιπαθητικό	○	○	○	○	○	○	○	συμπαθητικό	14
συνηθισμένο	○	○	○	○	○	○	○	πρωτοπόρο	15
δυσάρεστο	○	○	○	○	○	○	○	ευχάριστο	16
ασφαλές	○	○	○	○	○	○	○	ανασφαλές	17
ενθαρρυντικό	○	○	○	○	○	○	○	αποθαρρυντικό	18
ανταποκρίνεται στις προσδοκίες	○	○	○	○	○	○	○	δεν ανταποκρίνεται στις προσδοκίες	19
ανεπαρκές	○	○	○	○	○	○	○	επαρκές	20
σαφές	○	○	○	○	○	○	○	μπερδεμένο	21
μη πρακτικό	○	○	○	○	○	○	○	πρακτικό	22
οργανωμένο	○	○	○	○	○	○	○	ανοργάνωτο	23
ελκυστικό	○	○	○	○	○	○	○	απωθητικό	24
φιλικό	○	○	○	○	○	○	○	εχθρικό	25
συντηρητικό	○	○	○	○	○	○	○	καινοτόμο	26

Εικόνα 6.2. Ερωτηματολόγιο Αξιολόγησης

Αφού κατασκευάσαμε μία ηλεκτρονική φόρμα της Google, η οποία περιείχε το παραπάνω ερωτηματολόγιο, αυτή μοιράστηκε σε ένα σύνολο χρηστών, οι οποίοι είχαν την δυνατότητα να δοκιμάσουν την εφαρμογή. Εν τέλει, συγκεντρώθηκαν 27 δείγματα. Πάνω σε αυτά εφαρμόστηκε συγκεντρωτικά η ανάλυση των αποτελεσμάτων. Αξιοποιήθηκε το εργαλείο Data Analysis Tool, το οποίο παρέχεται από το USER-EXPERIENCE-QUESTIONNAIRE (UEQ) και είναι διαθέσιμο στο ευρύ κοινό. Το σύνολο των 27 χρηστών που συμμετείχαν ήταν φοιτητές της σχολής Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών (ΗΜΜΥ) του Πολυτεχνείου Κρήτης, ηλικίας 24-26 ετών, εκ των οποίων οι 23 ήταν άντρες και οι 4 ήταν γυναίκες.

6.2. Αποτελέσματα

Η χρήση του εργαλείου Data Analysis Tool συνεπάγεται μεγάλη διευκόλυνση στην ανάλυση δεδομένων UEQ. Εισάγοντας τα δεδομένα που έχουμε λάβει από το ερωτηματολόγιο σε μορφή ηλεκτρονικής φόρμας (η οποία μας παρέχεται από το λογιστικό φύλλο με τις απαντήσεις) μπορούμε αυτόματα να εξάγουμε όλα τα απαραίτητα στατιστικά στοιχεία που συμμετέχουν στην ερμηνεία των αποτελεσμάτων. Το εργαλείο αυτό περιλαμβάνει τα ακόλουθα Worksheets:

WorkSheet 1, Input Data

Εδώ έγινε η εισαγωγή των δεδομένων από το κάθε δείγμα δημιουργώντας πίνακα 27 γραμμών και 26 στηλών.

Worksheet 2, DT

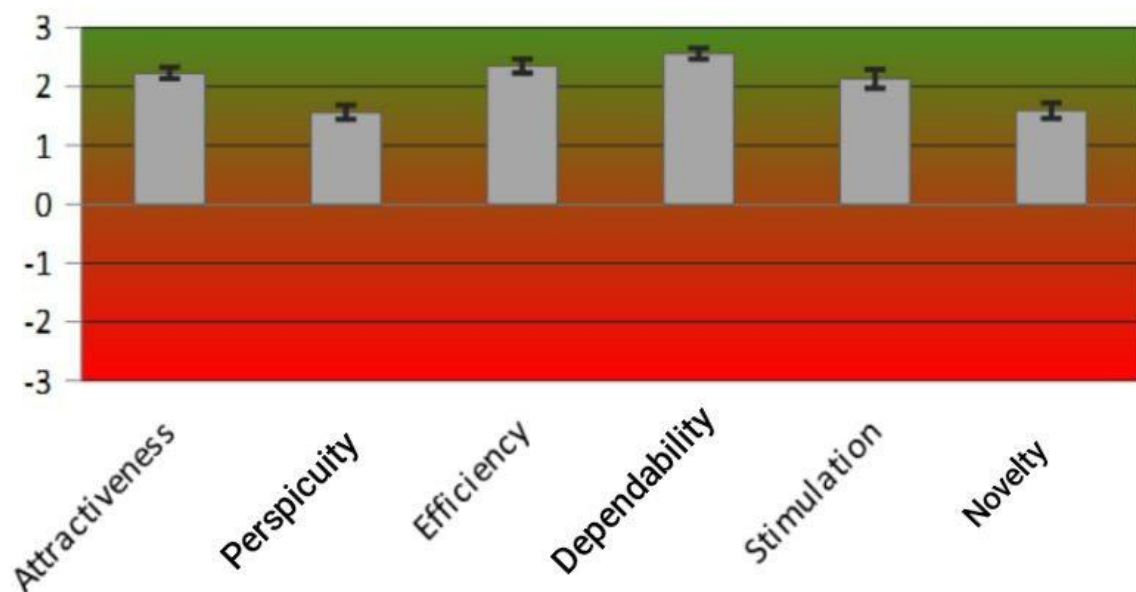
Είναι υπεύθυνο για την αντιστοίχιση των τιμών από τον πρώτο πίνακα με τις νέες ισορροποιημένες τιμές. Οι νέες τιμές που καταχωρούνται στον δεύτερο πίνακα βρίσκονται μεταξύ των τιμών -3 και +3 όπου το -3 εκπροσωπεί την πιο αρνητική απάντηση, ενώ το +3 την πιο θετική. Για παράδειγμα, αν η απάντηση σε μία ερώτηση ενός δείγματος ήταν ο αριθμός 7, τότε αυτή στον νέο πίνακα θα αντιστοιχούσε στον νέο αριθμό +3. Επιπροσθέτως, μετά το πέρας της προαναφερθείσας αντιστοίχισης, γίνεται εξαγωγή των αποτελεσμάτων για το scale-means-per-person. Πρόκειται για μια ομαδοποίηση των τιμών του δεύτερου πίνακα σε έναν νέο πίνακα 27x6, όπου οι 6 στήλες αντιστοιχούν στο σύνολο των scales που αναφέρθηκαν.

Worksheet 3, Results

Επίσης, το εργαλείο έχει την δυνατότητα να υπολογίσει την μέση τιμή (mean) των δεδομένων, καθώς και την διασπορά (variance) για κάθε μία από τις 26 ερωτήσεις, αλλά και για τα 27 διαφορετικά δείγματα. Έπειτα, με χρήση του νέου αυτού πίνακα που περιέχει τα στατιστικά στοιχεία μέσης τιμής και διασποράς, το εργαλείο μπορεί ακόμα αυτόματα να παράγει έναν γενικό πίνακα με τις αντίστοιχες μετρικές μέσης τιμής και διασποράς που όμως αντιστοιχούν ξεχωριστά στις 6 διαφορετικές κλίμακες. Ακολουθεί παρακάτω ο εν λόγω πίνακας:

UEQ Scales (Mean and Variance)		
Attractiveness	2.224	0.06
Perspicuity	1.558	0.10
Efficiency	2.346	0.10
Dependability	2.558	0.06
Stimulation	2.125	0.17
Novelty	1.587	0.12

Εικόνα 6.3. Αποτελέσματα αξιολόγησης



Εικόνα 6.4. Διάγραμμα αποτελεσμάτων Νο1

Εντός του εργαλείου ορίζεται ως ουδέτερο εύρος αυτό μεταξύ των τιμών -0.8 και +0.8. Οι τιμές που βρίσκονται πάνω από +0,8 αντιστοιχούν σε μία θετική αξιολόγηση και οι τιμές κάτω από -0,8 αντιστοιχούν σε μία αρνητική. Είναι επίσης εμφανές στο διάγραμμα ότι το εύρος τιμών στον κατακόρυφο άξονα, που αντιστοιχεί στις τιμές του mean, κυμαίνονται μεταξύ -3 και +3, όπως είναι αναμενόμενο. Όσο προσεγγίζεται ο αριθμός +3, πρόκειται για ένα πάρα πολύ καλό σύστημα, ενώ στην τιμή -3, για ένα πάρα πολύ κακό σύστημα. Ωστόσο, το data analysis tool αναφέρει για το προκειμένο σύστημα, πως λόγω του υπολογισμού του αριθμητικού μέσου για διαφορετικά δείγματα, δεν είναι δυνατόν να καλυφθεί ολόκληρο το εύρος.

Σε όλες τις κατηγορίες, οι τιμές ξεπερνούν το +0.8 και επομένως μπορούμε με ασφάλεια να συμπεράνουμε ότι πρόκειται για ένα σύστημα το οποίο θα χαίρει θετικής αντιμετώπισης από την πλειονότητα της πανεπιστημιακής κοινότητας.

Συνοπτικός υπολογισμός των τριών κύριων παραγόντων του συστήματος

Οι κλίμακες του UEQ μπορούν να ομαδοποιηθούν σε Pragmatic Quality (Perspicuity, Efficiency, Dependability) και Hedonic Quality (Stimulation, Originality).

Pragmatic and Hedonic Quality	
Attractiveness	2.22
Pragmatic Quality	2.15
Hedonic Quality	1.86

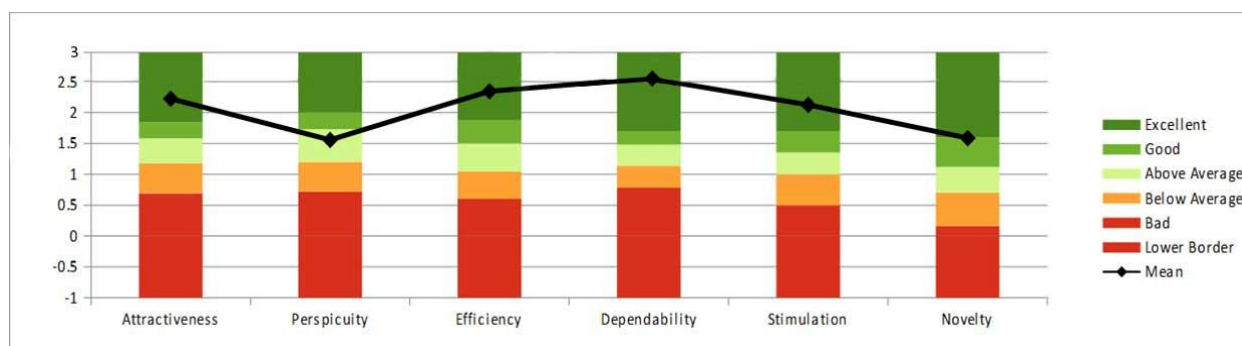
Εικόνα 6.5. Συνοπτικός πίνακας αποτελεσμάτων

Worksheet 4, Σύγκριση αποτελεσμάτων με Benchmark

Ένα ακόμα χρήσιμο εργαλείο που συμπεριλαμβάνεται είναι το Benchmark dataset. Συγκεκριμένα, το εργαλείο αυτό αξιοποιεί ένα σύνολο δεδομένων αναφοράς από 21175 άτομα σε 468 έρευνες που αφορούσαν διαφορετικά προϊόντα (ιστοσελίδες, επιχειρηματικό λογισμικό, κοινωνικά δίκτυα και καταστήματα ιστού) συγκριτικά με το αντικείμενο αξιολόγησης. Συνεπώς, μπορούμε να βελτιώσουμε την εμπιστοσύνη στην εξαγωγή ασφαλούς συμπεράσματος σχετικά με την ποιότητα του συστήματος που αξιολογούμε. Παραθέτουμε τις μέσες τιμές της εφαρμογής μας και το πως κατατάσσονται αυτές σχετικά με την ποιότητά τους σε κάθε scale, σύμφωνα με τους υπολογισμούς του benchmark.

Scale	Mean	Comparisson to benchmark	Interpretation
Attractiveness	2.22	Excellent	In the range of the 10% best results
Perspiciuity	1.56	Above Average	25% of results better, 50% of results worse
Efficiency	2.35	Excellent	In the range of the 10% best results
Dependability	2.56	Excellent	In the range of the 10% best results
Stimulation	2.13	Excellent	In the range of the 10% best results
Novelty	1.59	Good	10% of results better, 75% of results worse

Εικόνα 6.6. Αναλυτικός πίνακας αποτελεσμάτων



Εικόνα 6.7. Αναλυτικό διάγραμμα αποτελεσμάτων

Έπειτα από τη μελέτη των αποτελεσμάτων της ανάλυσης με βάση την διαδικασία όπως περιγράφηκε, μπορούμε να αποφανθούμε πως η ανταπόκριση ήταν εν γένει θετική. Επομένως, μπορούμε να θεωρήσουμε πως θα επιδιωχθεί η χρήση του εργαλείου από την πανεπιστημιακή κοινότητα με αποτέλεσμα την μακροχρόνια επιβίωση αυτού του έργου.

7. Συμπεράσματα

7.1. Συζήτηση

Συμπερασματικά, η παρούσα διπλωματική εργασία παρουσίασε την ανάπτυξη και υλοποίηση μιας διαδικτυακής εφαρμογής για την σχεδίαση και προσομοίωση Μηχανών Turing. Η εφαρμογή σχεδιάστηκε με μια φιλική προς το χρήστη διεπαφή για να την κάνει προσβάσιμη σε φοιτητές και ερευνητές στην επιστήμη των υπολογιστών, τα μαθηματικά και τη θεωρητική επιστήμη των υπολογιστών, σύμφωνα με το βιβλίο, τους κανόνες και τον συμβολισμό των Lewis & Papadimitriou [1]. Η εφαρμογή δημιουργήθηκε χρησιμοποιώντας σύγχρονες τεχνολογίες Ιστού, όπως HTML, CSS και JavaScript, οι οποίες επέτρεψαν την απρόσκοπτη ενσωμάτωση του αλγορίθμου προσομοίωσης και της διεπαφής χρήστη.

Η εφαρμογή δοκιμάστηκε και αξιολογήθηκε χρησιμοποιώντας μια ποικιλία Μηχανών Turing και τα αποτελέσματα έδειξαν ότι η εφαρμογή προσομοιώνει με ακρίβεια τη συμπεριφορά των Μηχανών Turing. Επιπλέον, η εφαρμογή περιλαμβάνει λειτουργίες, όπως οπτικοποίηση βήμα προς βήμα και δημιουργία ταινίας δυναμικής εισαγωγής, που βελτιώνει την κατανόηση του χρήστη για την ιδέα της Μηχανής Turing.

Συνολικά, η διαδικτυακή εφαρμογή σχεδίασης και προσομοίωσης Μηχανών Turing έχει τη δυνατότητα να γίνει ένα πολύτιμο εκπαιδευτικό εργαλείο για φοιτητές και ερευνητές στην επιστήμη των υπολογιστών και σε συναφείς τομείς. Παρέχει μια ευέλικτη και προσβάσιμη πλατφόρμα για εξερεύνηση και πειραματισμό με Μηχανές Turing και έχει τη δυνατότητα να επεκτείνει την κατανόηση και την εκτίμηση αυτής της θεμελιώδους ιδέας στη θεωρητική επιστήμη των υπολογιστών.

Εν κατακλείδι, αυτή η διπλωματική εργασία καταδεικνύει τη σκοπιμότητα και την αποτελεσματικότητα της ανάπτυξης διαδικτυακών εφαρμογών για την σχεδίαση και προσομοίωση Μηχανών Turing και θέτει τα θεμέλια για μελλοντικές εργασίες σε αυτόν τον τομέα.

7.2. *Μελλοντικές Αναβαθμίσεις*

Μία από τις πιο σημαντικές μελλοντικές βελτιώσεις για την εφαρμογή θα ήταν η ανάπτυξη ενός διακομιστή που θα επιτρέπει στους χρήστες να συνδέονται, να αποθηκεύουν την εργασία τους και να συνεργάζονται με άλλους χρήστες ταυτόχρονα. Η δυνατότητα συνεργασίας θα επέτρεπε στους χρήστες να συνεργαστούν στη δημιουργία Μηχανών Turing, κάτι που θα μπορούσε να οδηγήσει στη δημιουργία πιο πολύπλοκων και αποτελεσματικών μηχανών.

Σημαντική αναβάθμιση θα ήταν επίσης και η προσθήκη μνήμης στην εφαρμογή με τη χρήση κάποιας βάσης δεδομένων. Έτσι, όταν ο χρήστης δημιουργούσε την δική του μηχανή, θα μπορούσε το σύστημα να την αποθηκεύει στην βάση και με προσθήκη περισσότερης λογικής στην εφαρμογή, θα ήταν εφικτή η χρήση της καινούργιας μηχανής που δημιούργησε ο χρήστης ως δομικό στοιχείο, δηλαδή ως μηχανή που θα συμβολίζεται με κάποιο σύμβολο και θα μπορούσε να χρησιμοποιηθεί μέσα σε κάποιον κόμβο.

Μια άλλη πιθανή βελτίωση θα ήταν η προσθήκη επιπλέον παραδειγμάτων Μηχανών Turing στην εφαρμογή. Επεκτείνοντας τον αριθμό των διαθέσιμων παραδειγμάτων, οι φοιτητές θα είχαν μεγαλύτερη ποικιλία Μηχανών Turing και θα μπορούσαν να αποκτήσουν καλύτερη κατανόηση των δυνατοτήτων τους. Επιπλέον, η συμπερίληψη πιο περίπλοκων παραδειγμάτων θα μπορούσε να προκαλέσει τους χρήστες και να τους ενθαρρύνει να πειραματιστούν με διαφορετικές τεχνικές για την επίλυση προβλημάτων.

Τέλος, η προσθήκη επεκτάσεων Μηχανών Turing είναι μια πολύτιμη βελτίωση στην εφαρμογή. Αυτές οι επεκτάσεις θα επέτρεπαν στους φοιτητές να δουν όλη την θεωρία που διδάσκεται στο μάθημα, πρακτικά και διαδραστικά μέσω του προσομοιωτή. Με την ενσωμάτωση αυτών των επεκτάσεων στην εφαρμογή, οι φοιτητές θα έχουν πρόσβαση σε ένα ευρύτερο φάσμα υπολογιστικών εργαλείων και θα μπορούν να εξερευνούν πιο σύνθετα προβλήματα.

Συνολικά, αυτές οι βελτιώσεις θα κάνουν τον προσομοιωτή Μηχανής Turing πιο ισχυρό και φιλικό προς τον χρήστη, παρέχοντας στους χρήστες περισσότερες ευκαιρίες να μάθουν και να πειραματιστούν με αυτό το σημαντικό θεωρητικό κατασκεύασμα στην επιστήμη των υπολογιστών.

7.3. *Μαθήματα*

Η ανάπτυξη μιας διαδικτυακής εφαρμογής για την προσομοίωση Μηχανών Turing ήταν μια πολύτιμη εμπειρία μάθησης για μένα ως φοιτητή. Μέσω της διαδικασίας σχεδιασμού, κωδικοποίησης και δοκιμής της εφαρμογής, απέκτησα μια βαθύτερη κατανόηση της εσωτερικής λειτουργίας των Μηχανών Turing και της Θεωρίας Υπολογισμού στο σύνολό της.

Ένα από τα μαθήματα που πήρα μέσα από αυτή την ενασχόληση ήταν η σημασία της σαφούς και συνοπτικής επικοινωνίας στο σχεδιασμό μιας διαδικτυακής εφαρμογής. Διαπίστωσα ότι η δημιουργία μιας φιλικής προς τον χρήστη διεπαφής που επικοινωνεί με ακρίβεια και αποτελεσματικότητα με τη συμπεριφορά μιας Μηχανής Turing ήταν ζωτικής σημασίας για την επιτυχία του έργου. Εξετάζοντας προσεκτικά την εμπειρία του χρήστη, μπόρεσα να δημιουργήσω μια εφαρμογή που όχι μόνο προσομοιώνει με ακρίβεια Μηχανές Turing, αλλά είναι επίσης εύκολη στη χρήση και την κατανόηση.

Τέλος, έμαθα για την επαναληπτική φύση της ανάπτυξης λογισμικού. Διαπίστωσα ότι ήταν απαραίτητο να κάνω αλλαγές και βελτιώσεις στην εφαρμογή σε κάθε δοκιμή επίλυσης ασκήσεων του μαθήματος. Αυτή η διαδικασία μου δίδαξε τη σημασία του να είμαι ευέλικτος και ανοιχτός σε αλλαγές, καθώς και την αξία της δοκιμής και του εντοπισμού σφαλμάτων για τη διασφάλιση της ποιότητας και της αξιοπιστίας της εφαρμογής μας.

Συμπερασματικά, η ανάπτυξη μιας διαδικτυακής εφαρμογής για την σχεδίαση και προσομοίωση Μηχανών Turing ήταν μια προκλητική και ικανοποιητική εμπειρία που με δίδαξε σημαντικά μαθήματα σχετικά με την ανάπτυξη λογισμικού και τη Θεωρία Υπολογισμού. Η εφαρμογή όχι μόνο μου επέτρεψε να ενισχύσω την κατανόησή μου για τις Μηχανές Turing, αλλά μου παρείχε επίσης πολύτιμες δεξιότητες και εμπειρία που θα έχω την δυνατότητα να εφαρμόσω σε μελλοντικά έργα στην μετέπειτα πορεία μου.

8.

Βιβλιογραφία

- 0) Alan Turing, *On Computable Numbers, with an Application to the Entscheidungsproblem*. Proceedings of the London Mathematical Society 42 (1):230-265, 1936.
- 1) H. R. Lewis and C. H. Papadimitriou, *Elements of the Theory of Computation* (2nd edition, Prentice Hall, 1998).
- 2) «Θεωρία Υπολογισμού» Wikipedia, Wikimedia Foundation, 31 Αυγούστου 2022, https://el.wikipedia.org/wiki/Θεωρία_υπολογισμού
- 3) «Μηχανή Τούρινγκ» Wikipedia, Wikimedia Foundation, 31 Αυγούστου 2022, https://el.wikipedia.org/wiki/Μηχανή_Τούρινγκ
- 4) Anseki. *anseki/leader-line: Draw a leader line in your web page*. GitHub. Retrieved September 30, 2022, from <https://github.com/anseki/leader-line>
- 5) A. Bouchefra *Creating a Desktop App with Electron and Angular*. Buddy.Works. Retrieved October 2, 2022, from <https://buddy.works/tutorials/building-a-desktop-app-with-electron-and-angular>
- 6) *Drag and Drop (What Is JavaScript? - Learn Web Development | MDN)*. Angular Material. Retrieved June 3, 2022, from <https://material.angular.io/cdk/drag-drop/api>
- 7) *Groups and backreferences - JavaScript* MDN Web Docs. Retrieved September 21, 2022, from https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Regular_Expressions/Groups_and_Backreferences
- 8) *How to apply transform translate property to element using ngStyle in Angular2+*. Stack Overflow. Retrieved December 13, 2022, from <https://stackoverflow.com/questions/44364580/how-to-apply-transform-translate-property-to-element-using-ngstyle-in-angular2>
- 9) *How to create a .txt file using JavaScript / HTML5?* Stack Overflow. Retrieved November 14, 2022, from <https://stackoverflow.com/questions/31048215/how-to-create-txt-file-using-javascript-html5>
- 10) *Mat-Icon List : 900+ Angular Material Icons*. Angular Wiki. Retrieved December 13, 2022, from <https://www.angularjswiki.com/angular/angular-material-icons-list-mat-icon-list/>
- 11) *ngx-translate/core: The internationalization (i18n) library for Angular*. GitHub. Retrieved October 3, 2022, from <https://github.com/ngx-translate/core>
- 12) *QueryList*. Angular. Retrieved July 17, 2022, from <https://angular.io/api/core/QueryList>
- 13) S. Saseendran *Simple Way To Execute A Function In A Component From Another Component*. C# Corner. Retrieved July 17, 2022, from <https://www.c-sharpcorner.com/article/simple-way-to-execute-a-function-in-a-component-from-another-component/>

- 14) *Window.localStorage* - *Web APIs* MDN Web Docs. Retrieved December 4, 2022, from <https://developer.mozilla.org/en-US/docs/Web/API/Window/localStorage>
- 15) *What is JavaScript?* - *Learn web development* MDN Web Docs. Retrieved March 2, 2023, from https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/What_is_JavaScript
- 16) H. Agrawal *What is TypeScript and why should you use it?* Contentful. Retrieved March 2, 2023, from <https://www.contentful.com/blog/what-is-typescript-and-why-should-you-use-it/>
- 17) *What Is HTML? Hypertext Markup Language Basics Explained*. Hostinger. Retrieved March 2, 2023, from <https://www.hostinger.com/tutorials/what-is-html>
- 18) *CSS Introduction*. W3Schools. Retrieved March 2, 2023, from https://www.w3schools.com/css/css_intro.asp
- 19) *What is JSON?* W3Schools. Retrieved March 2, 2023, from https://www.w3schools.com/whatis/whatis_json.asp
- 20) *What is Angular?* Angular. Retrieved March 2, 2023, from <https://angular.io/guide/what-is-angular>
- 21) *Visual Studio Code Frequently Asked Questions*. Visual Studio Code. Retrieved March 2, 2023, from <https://code.visualstudio.com/docs/supporting/FAQ>
- 22) *Docker: Accelerated, Containerized Application Development*. Retrieved March 2, 2023, from <https://www.docker.com/>

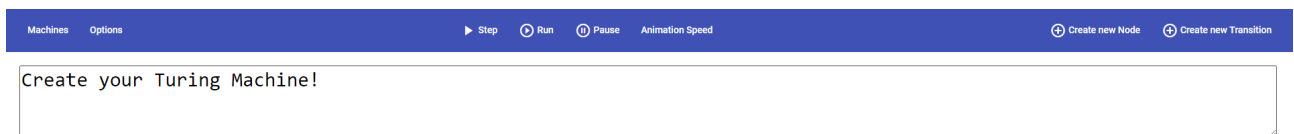
ΠΑΡΑΡΤΗΜΑ Α΄

Οδηγός Χρήσης

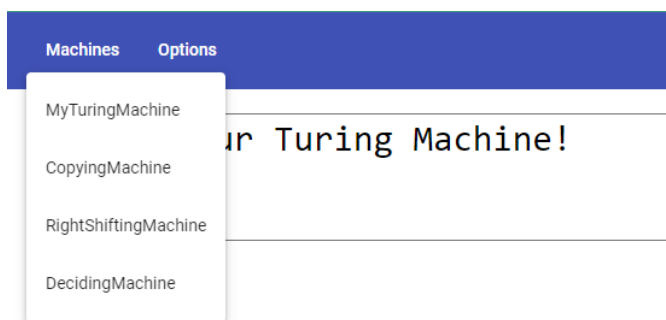
Το TUC-Turing-Machine-Simulator είναι ένα εργαλείο που δημιουργήθηκε για παιδαγωγικούς σκοπούς και προορίζεται να χρησιμοποιηθεί τόσο από φοιτητές, όσο και από καθηγητές. Οι φοιτητές μπορούν να το χρησιμοποιήσουν στο σπίτι ως βοηθό κατά τη μελέτη δύσκολων παραδειγμάτων ή την επίλυση ασκήσεων. Οι καθηγητές, από την άλλη, μπορούν να το χρησιμοποιήσουν για να απλοποιήσει και να οπτικοποιήσει περίπλοκες έννοιες των μηχανών την ώρα της διδασκαλίας. Ο παρακάτω σύνδεσμος (link) οδηγεί στην εφαρμογή, η οποία είναι ανεβασμένη στον server του εργαστηρίου Intelligent Systems Laboratory.

<http://alchimix1.intelligence.tuc.gr:4200/>

Ξεκινάμε, βλέποντας την αρχική σελίδα του προσομοιωτή. Όλες οι ενέργειες γίνονται στην ίδια σελίδα. Στην αρχή, δεν υπάρχει αρχικοποίηση και μπορούμε να ξεκινήσουμε.



Εάν ο χρήστης επιθυμεί να φορτώσει έτοιμη μηχανή από τα παραδείγματα που διδάσκονται στο μάθημα, τότε μπορεί με το κουμπί Machines που βρίσκεται πάνω αριστερά στην γραμμή εργαλείων να επιλέξει την μηχανή που επιθυμεί, όπως φαίνεται στην επόμενη εικόνα. Εάν επιθυμεί να δημιουργήσει μια νέα Μηχανή Turing, τότε μπορεί να διαλέξει την επιλογή MyTuringMachine.



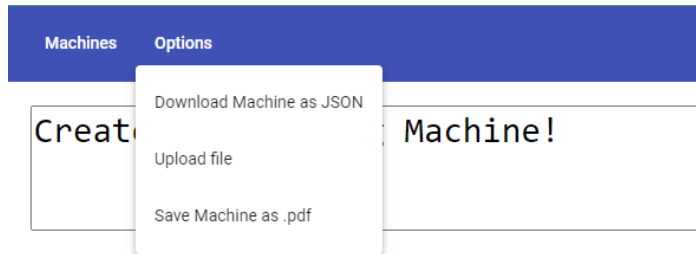
Paraskevas Kontoulis

Το κουμπί Options περιέχει μια λίστα με εργαλεία:

1. Να κατεβάσει ο χρήστης την μηχανή που έχει φορτωμένη στην σελίδα εκείνη τη στιγμή. Η μηχανή μπορεί να είναι έτοιμη ή ατελής.
2. Να ανεβάσει και να φορτώσει από τον τοπικό του δίσκο τη δική του μηχανή, η οποία θα πρέπει να βρίσκεται σε σωστή μορφή JSON όπως θα δείξουμε παρακάτω. Η μορφή που θα πρέπει να έχει το αρχείο JSON είναι η παρακάτω:

```
{
  "states": [
    { "id": 0, "type": "Initial", "actions": [ "L", "translateX": "690", "translateY": "120" },
    { "id": 1, "type": "Intermediate", "actions": [ "u", "R", "X", "L", "translateX": "866", "translateY": "116" },
    { "id": 2, "type": "Final", "actions": [ "Ru", "Ru", "translateX": "521", "translateY": "262" },
    { "id": 3, "type": "Intermediate", "actions": [ ] }
  ],
  "deltas": [
    { "prevStateId": 0, "input": [ "a", "b" ], "newStateId": 1, "text": "", "lineType": "grid", "startSocket": "right", "endSocket": "left" },
    { "prevStateId": 1, "input": [ "a", "b", " " ], "newStateId": 0, "text": "", "lineType": "grid", "startSocket": "top", "endSocket": "top" },
    { "prevStateId": 0, "input": [ " " ], "newStateId": 2, "text": "", "lineType": "grid", "startSocket": "bottom", "endSocket": "top" }
  ],
  "alphabet": [ " ", "a", "b" ],
  "tape": [ " ", "a", "a", "b", "a", "b", " " ],
  "head": 6,
  "text": "This is the shift right machine example."
}
```

3. Να αποθηκεύσει την μηχανή που έχει φορτωμένη στην σελίδα εκείνη τη στιγμή ως pdf αρχείο. Η επιλογή αυτή δόθηκε, ώστε ο φοιτητής να μπορεί να παραδώσει την εργασία του ως λύση σε άσκηση που θα έχει δοθεί προς επίλυση.

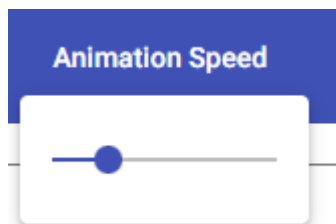
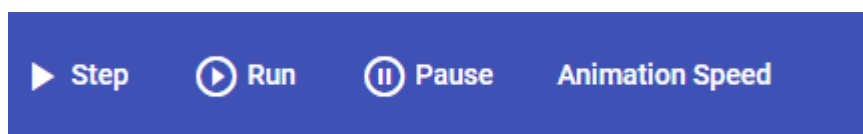


Paraskevas Kontoulis

Στις παρακάτω εικόνες, βλέπουμε τα τέσσερα κουμπιά προσομοίωσης της μηχανής που έχουμε φορτώσει στην σελίδα μας.

1. Step: Οποιαδήποτε στιγμή στην οποία η μηχανή είναι σε παύση (έχει πατηθεί το κουμπί Pause) μπορεί να γίνει η χρήση του κουμπιού Step ώστε η μηχανή να “τρέξει” για μία μετάβαση. Εάν χρησιμοποιηθεί αρκετές φορές ώστε να καταλήξει η μηχανή θα έχει το ίδιο ακριβώς αποτέλεσμα με την χρήση μόνο του κουμπιού Run.
2. Run: Αρκεί να πατηθεί μία φορά για να ξεκινήσει η προσομοίωση. Ύστερα ο έλεγχος για την ροή της προσομοίωσης γίνεται από το κουμπί Pause.
3. Pause: Σταματάει την ροή της προσομοίωσης και όταν πατηθεί αλλάζει σε Resume. Όταν ξαναχρησιμοποιηθεί και αναγράφει Resume τότε θα γίνει εκκίνηση της μηχανής από το σημείο που είχε σταματήσει.

4. Animation Speed: όταν πατηθεί ανοίγει μια μπάρα στην οποία όσο πάμε πιο δεξιά τόσο πιο γρήγορα γίνεται η προσομοίωση ενώ αντίθετα στα αριστερά γίνεται πιο αργή.



Η παρακάτω εικόνα δείχνει τα δύο κουμπιά με τα οποία χρήστης μπορεί να δημιουργήσει νέους κόμβους και νέες μεταβάσεις. Στις εικόνες που ακολουθούν φαίνεται τι θα συμβεί, εάν πατήσει ο χρήστης το κάθε κουμπί αντίστοιχα.



Στην παρακάτω εικόνα βλέπουμε αρχικά το αναγνωριστικό του κόμβου, το οποίο είναι απαραίτητο, όταν θα θέλουμε να φτιάξουμε μια μετάβαση. Για να γνωρίζει η εφαρμογή που θα πάει η καινούργια μετάβαση που δημιουργήσε ο χρήστης, θα πρέπει να χρησιμοποιήσει το αναγνωριστικό αυτό.

Ακολουθεί ο τύπος κόμβου. Όταν θελήσει ο χρήστης να φτιάξει έναν κόμβο από τον οποίο θα ξεκινάει η μηχανή του, τότε θα πρέπει να διαλέξει την επιλογή Initial. Όταν θα θελει να σχεδιάσει κόμβο που δεν θα οδηγεί σε άλλο κόμβο με κάποια μετάβαση, τότε θα επιλέξει την επιλογή Final. Σε οποιαδήποτε άλλη περίπτωση, αρκεί να διαλέξει την επιλογή Intermediate.

Στην επιλογή Node Machines που ακολουθεί, ο χρήστης μπορεί να διαλέξει από τις επιλογές που υπάρχουν με την βοήθεια κουμπιών να προσθέσει όποια βασική ή σύνθετη μηχανή επιθυμεί ο ίδιος στον κόμβο που έχει διαλέξει. Για να αφαιρέσει κάτι, αρκεί να πατήσει το κουμπί δεξιά από την αναπαράσταση όλων των μηχανών που έχει ήδη επιλέξει. Οι μηχανές που διατίθενται είναι με την σειρά:

1. Μηχανή εγγραφής κενού συμβόλου στην θέση όπου βρίσκεται η κεφαλή εκείνη τη στιγμή.
2. Μηχανή μετακίνησης της κεφαλής μία θέση δεξιά.

3. Μηχανή μετακίνησης της κεφαλής μία θέση αριστερά.
4. Μηχανή εύρεσης του πρώτου κενού στα δεξιά.
5. Μηχανή εύρεσης του πρώτου κενού στα αριστερά.
6. Μηχανή εύρεσης του πρώτου συμβόλου στα δεξιά που είναι ίδιο με το σύμβολο που διάβασε η μηχανή ως είσοδο όταν έφτασε στον κόμβο.
7. Μηχανή εύρεσης του πρώτου συμβόλου στα αριστερά που είναι ίδιο με το σύμβολο που διάβασε η μηχανή ως είσοδο όταν έφτασε στον κόμβο.
8. Μηχανή καταγραφής του συμβόλου που διάβασε η μηχανή ως είσοδο όταν έφτασε στον κόμβο, στην θέση που δείχνει η κεφαλή εκείνη τη στιγμή.
9. Μηχανή καταγραφής του συμβόλου 'd' στην θέση που δείχνει η κεφαλή (χρησιμοποιείται ως σημάδι σε συγκεκριμένες ασκήσεις)
10. Μηχανή τερματισμού με μήνυμα 'yes'
11. Μηχανή τερματισμού με μήνυμα 'no'
12. Μηχανή αντιγραφής της συμβολοσειράς που έχει η ταινία εκείνη την στιγμή (απαιτείται η κεφαλή να βρίσκεται στο πρώτο κενό δεξιά από το τέλος της συμβολοσειράς με προαπαιτησι η ταινία να περιέχει κενό σύμβολο ακολουθούμενο από την συμβολοσειρά και στο τέλος κενό σύμβολο
13. Μηχανή ολίσθησης της συμβολοσειράς μία θέση δεξιά.
14. Στην δεύτερη σειρά θα εμφανίζονται τόσες μηχανές καταγραφής συγκεκριμένου συμβόλου όσα και τα διαφορετικά σύμβολα του αλφαβήτου της μηχανής.

Ακολουθούν οι μεταβάσεις. Για κάθε μετάβαση, αριστερά βλέπουμε τα δεδομένα και δεξιά ακολουθούν επιλογές για τον τύπο σχεδίασης των ακμών, καθώς επίσης από που θα ξεκινάει η ακμή στον επιλεγμένο κόμβο και που θα τελειώνει η ακμή στον κόμβο που επιθυμεί ο χρήστης. Υπάρχει και κουμπί διαγραφής για την κάθε ακμή.

Για να προσθέσει μια νέα μετάβαση, ο χρήστης θα πρέπει να συμπληρώσει στο πλαίσιο Next Node το id του node που επιθυμεί, στο πλαίσιο input τους χαρακτήρες με τους οποίους όταν τους διαβάζει η κεφαλή θα επιλέξει να μεταβεί εκεί και στο πλαίσιο transition text το κείμενο που επιθυμεί να φαίνεται πάνω από την ακμή. Εάν το αφήσει κενό, τότε αυτόματα θα δείχνει τα σύμβολα που έχει δώσει ο χρήστης στο πλαίσιο input.

Κάτω δεξιά υπάρχει και κουμπί διαγραφής του επιλεγμένου Node.

id: 2

Node Type

☐ Initial
 ☒ Intermediate
 ☐ Final

Node Machines: R

U
a

R
b

L
c

RU

LU

Rx

Lx

X

d

y

n

C

Sr

Transition Functions

2 → 2 / a,d	Line type Grid	Start anchor Right	End anchor Left	Delete Transition
2 → 3 / b	Line type Grid	Start anchor Right	End anchor Left	Delete Transition
2 → 9 / c,U	Line type Straight	Start anchor Bottom	End anchor Top	Delete Transition

Add transition function(δ)

Next Node / Input / Transition text ≠

+ Add Transition function

 Delete Node

Όταν επιλεγθεί το κουμπί Create New Transition, τότε ο χρήστης θα πρέπει να συμπληρώσει στο πλαίσιο Starting Node το id του κόμβου που επιθυμεί να ξεκινήσει η ακμή, στο πλαίσιο input τα σύμβολα με τα οποία θα γίνεται η μετάβαση και στο πλαίσιο Ending Node το id του κόμβου που επιθυμεί για να καταλήξει η ακμή.

New transition function(δ)

Starting Node

Input

Ending Node

+

Για να επεξεργαστεί ο χρήστης το αλφάβητο, την ταινία και την κεφαλή θα πρέπει να κάνει κλικ στην ταινία. Τότε, όπως φαίνεται στην παρακάτω εικόνα, θα έχει την επιλογή να δώσει την συμβολοσειρά που επιθυμεί (θα πρέπει η συμβολοσειρά να διαθέτει μόνο σύμβολα που είναι δηλωμένα στο αλφάβητο της μηχανής), να προσθέσει τα σύμβολα που επιθυμεί στο αλφάβητο και να μετακινήσει την κεφαλή στην θέση που επιθυμεί με την βοήθεια των τεσσάρων κουμπιών:

1. Start: Μετακινεί την κεφαλή στην αρχή της ταινίας.
2. End: Μετακινεί την κεφαλή στο τέλος της συμβολοσειράς.
3. Left: Μετακινεί την κεφαλή μία θέση αριστερά.
4. Right: Μετακινεί την κεφαλή μία θέση δεξιά.

The image shows a user interface for a Turing Machine simulation. It is divided into three main sections:

- Tape:** Contains a text input field with the string "aabab" and a cursor at the end. To the right of the input are a plus sign (+) button and a trash can icon.
- Alphabet Σ :** Contains a text input field labeled "Insert symbol" with a plus sign (+) button to its right. Below the input is the text "Max 1 character". There is a list of symbols: "• \sqcup ", "• a", and "• b". Each symbol is followed by a circle containing an 'x' (✕), indicating they are available for selection.
- Head position:** Contains four buttons labeled "Start", "End", "Left", and "Right".