

# ΠΑΝΕΠΙΣΤΗΜΙΟ ΜΑΚΕΔΟΝΙΑΣ

## ΤΜΗΜΑ ΕΦΑΡΜΟΣΜΕΝΗΣ

## ΠΛΗΡΟΦΟΡΙΚΗΣ

### ΘΕΩΡΙΑ ΥΠΟΛΟΓΙΣΜΩΝ ΚΑΙ ΑΥΤΟΜΑΤΩΝ

#### 1<sup>η</sup> Εργασία

##### 1. Τεκμηρίωση Κώδικα

- Δομή και ανάγνωση αρχείου περιγραφής αυτομάτου

Η δομή του αρχείου είναι παρόμοια με αυτήν που ορίστηκε από την εκφώνηση της άσκησης, με 1 διευκρίνιση προς τον χρήστη. Για αυτόματα άνω των 9 καταστάσεων απαιτείται από τον χρήστη να προσθέτει ένα κόμμα πριν και μετά από κάθε κατάσταση όταν η γραμμή περιέχει κατάσταση της οποίας το αντίστοιχο νούμερο έχει περισσότερα από ένα ψηφιά. Η διευκρίνιση αυτή ισχύει για τις γραμμές του αρχείου που αφορούν μεταβάσεις και τελικές καταστάσεις. Για παράδειγμα:

Μεταβάσεις:

4 @ 5

,5, b ,12,

,12, @ ,2,

Τελικές καταστάσεις:

,1,11,12,

Η διευκρίνηση αυτή θα μπορούσε να γίνει κανονικά και με κενά ( ' ') αναμεσά στα σύμβολα. Το πρόγραμμα όμως σχεδιάστηκε εξ αρχής να λειτουργεί αφαιρώντας πρώτα τα κενά από κάθε γραμμή γιατί συχνά προθέτονται παραπάνω κενά εκ 'παραδρομής δημιουργώντας σφάλματα. Απαιτούνται επομένως τα κόμματα για να 'δηλώνουν' καταστάσεις δυο ή παραπάνω ψηφίων.

Κατά την έναρξη της εκτέλεσης του προγράμματος ζητάτε από τον χρήστη να δηλώσει το path του αρχείου στον υπολογιστή του. Η ανάγνωση του αρχείου γίνεται εντός while-loop το οποίο τερματίζει όταν εντοπίσει μια κενή γραμμή.

Η δομή του αρχείου πρέπει να τηρείται αυστηρά για την σωστή εκτέλεση του προγράμματος καθώς για παράδειγμα κενές γραμμές εντός της περιγραφής θα δημιουργήσουν προβλήματα. Επιπλέον στην περίπτωση μη αιτιοκρατικών αυτομάτων όπου ορισμένες καταστάσεις έχουν μεταβάσεις για το ίδιο γράμμα σε παραπάνω από μια καταστάσεις, θα πρέπει, τα βελάκια αυτά, να δηλώνονται στο αρχείο σε ξεχωριστές γραμμές (π.χ. 1 α 2

1 α 3 K.O.K.)

- Δομές δεδομένων

Κατά την ανάγνωση του αρχείου τα δεδομένα αποθηκεύονται σε λίστες μεγέθους ανάλογου του είδους των καταστάσεων που αποθηκεύει (current states, accepting states κ.α.). Ιδιαίτερη περίπτωση αποτελεί η αποθήκευση των transitions αναμεσά στις καταστάσεις όπου κάνοντας χρήση

εμφωλευμένων λιστών δημιουργείται μια  $T \times 3$  λίστα, όπου  $T$  το πλήθος των transitions. Στην πρώτη στήλη αποθηκεύεται το νούμερο της κατάστασης από πού ξεκινά το βελάκι, στην δεύτερη το γράμμα που απαιτείται για την μετάβαση και τέλος στην τρίτη το νούμερο της κατάστασης στην οποία δείχνει το βελάκι παρόμοια με την δομή του αρχείου. Παρακάτω περιγράφονται και κάποιες άλλες δισδιάστατες λίστες που χρησιμοποιούνται από το πρόγραμμα. Τέλος όπου χρειάζεται αποθήκευση του αλφάβητου της γλώσσας του αυτομάτου γίνεται χρήση συνόλων.

- Συνάρτηση Μεταβάσεων - transitionFunction

Είσοδος: Το υπό εξέταση γράμμα, λίστα του αρχικού state, λίστα των current states, λίστα των τελικών states, λίστα των transitions, πλήθος των transactions.

Έξοδος: Το updated current list.

Υστέρα από την ανάγνωση του αρχείου και την ενημέρωση των λιστών, ορίζεται η Transition Function με τα παραπάνω ορίσματα. Η συνάρτηση αυτή κάνει χρήση ενός εμφωλευμένου loop το οποίο για κάθε current κατάσταση ελέγχει την λίστα των transitions με βάση το υπό εξέταση γράμμα έτσι ώστε να ενημερώσει καταλληλά μια βοηθητική λίστα. Η βοηθητική λίστα αυτή στη συνέχεια θα γίνει η νέα λίστα των current states.

Μια ακόμη αρμοδιότητα της συνάρτησης είναι να εντοπίζει περιπτώσεις όπου το αυτόματο λαμβάνει κενή λέξη (η οποία για το πρόγραμμα ορίζεται ως το ' ') και ανάλογα με τη λίστα του αρχικού state και των τελικών states να κρίνει αν το αυτόματο θα αποδεχτεί την λέξη. Επιπλέον εντοπίζει περιπτώσεις όπου διαβάζεται από το αυτόματο, γράμμα για το

οποίο τα current states δεν διαθέτουν μετάβαση σε άλλο state και η λέξη δεν γίνεται αποδεκτή από το αυτόματο.

- Μετατροπή αυτομάτου με e-transitions σε Automaton χωρίς e-transitions.

Η τεχνική που επιλέχθηκε για την αντιμετώπιση των e-transitions είναι η αφαίρεση τους μετατρέποντας το αυτόματο στην κατάλληλη μορφή. Η μετατροπή γίνεται μέσω της συνάρτησης `convertToNoEpsilon`. Το αυτόματο που προκύπτει από την μετατροπή αυτή μπορεί να είναι είτε Αιτιοκρατικό είτε Μη Αιτιοκρατικό.

Είσοδος: Λίστα των τελικών states, λίστα των transitions, πλήθος των transactions.

Έξοδος: Λίστα των transitions χωρίς e-transitions, ανανεωμένη λίστα των τελικών states, ανανεωμένο πλήθος transitions.

Η λειτουργία της συνάρτησης αποτελείται από 5 βήματα:

#### 1. Δημιουργία του E Transition Matrix

Ουσιαστικά πρόκειται για τη συνάρτηση μετάβασης του αυτομάτου στην μορφή πίνακα. Οι γραμμές του πίνακα έχουν πλήθος ίσο με το σύνολο των καταστάσεων του αυτομάτου και οι στήλες πλήθος ίσο με τα γράμματα του αλφάβητου της γλώσσας του αυτομάτου με μια επιπλέον στήλη για τα e-transitions. Μέσω ενός τριπλού for-loop η συνάρτηση μετατρέπει την λίστα των transitions σε μορφή πίνακα.

#### 2. Δημιουργία του E Closure Matrix

Πρόκειται για έναν τετραγωνικό πίνακα με πλήθος γραμμών και στηλών ίσο με το σύνολο των καταστάσεων του αυτομάτου. Κάθε κατάσταση αντιπροσωπεύεται από μια

γραμμή του πίνακα όπου σημειώνονται όλες οι καταστάσεις στις οποίες είναι δυνατό να μεταβεί κάνοντας χρήση e-transition συμπεριλαμβανομένου του εαυτού της. Η δημιουργία του πίνακα αυτού γίνεται κάνοντας χρήση της στήλης του E Transition Matrix που περιέχει τα e-transition κάθε κατάστασης.

### 3. Δημιουργία του No E Transition Matrix

Όπως και στην περίπτωση του E Transition Matrix, ο πίνακας αυτός διαθέτει μια γραμμή για κάθε κατάσταση του αυτομάτου και μια στήλη για κάθε γράμμα του αλφάβητου της γλώσσας του αυτομάτου. Δεν διαθέτει όμως στήλη για τα e-transitions καθώς πλέον γίνεται να αφαιρεθούν. Κάνοντας χρήση του E Closure Matrix η συνάρτηση γνωρίζει ποιες γραμμές του E Transition Matrix να προσθέσει ώστε να δημιουργήσει τον No E Transition Matrix.

### 4. Δημιουργία της νέας λίστας των μεταβάσεων χωρίς e-transitions

Διαβάζοντας κάθε γραμμή του No E Transition Matrix η συνάρτηση ενημερώνει καταλληλά τη λίστα των μεταβάσεων η οποία πλέον δεν περιέχει e-transitions και είναι δυνατό να χρησιμοποιηθεί από transitionFunction. Επιπλέον αντίστοιχα ενημερώνονται το πλήθος των transitions όπως και η λίστα των τελικών states καθώς οποία κατάσταση μπορεί μέσω e-transitions να μεταβεί σε τελική κατάσταση γίνεται και η ίδια τελική κατάσταση. Παρακάτω παρουσιάζονται αναλυτικά παραδείγματα της εκτέλεσης της συνάρτησης αυτής.

- Εκτύπωση αποτελέσματος

Στο τελευταίο κομμάτι του προγράμματος γίνεται interaction με τον user προκειμένου να δοθεί μια λέξη ως είσοδος. Κάθε γράμμα της λέξης αυτής γίνεται στοιχείο μιας λίστας και στην συνέχεια για κάθε στοιχείο της λίστας αυτής καλείται η Transition Function. Εκτυπώνεται το αποτέλεσμα και ο χρήστης μπορεί να δώσει νέα λέξη ή να τερματίσει την εκτέλεση του προγράμματος.

## 2. Παραδείγματα εκτέλεσης του προγράμματος

- Αιτιοκρατικά Ντετερμινιστικά Αυτόματα

**Παράδειγμα 1: Αυτόματο που δέχεται λέξεις που τελειώνουν με τουλάχιστον δυο α (εκφώνηση) με  $\Sigma=\{0,1\}$ .**

Αρχείο: Automaton.txt

```
F:\workspace\Finite Automaton Simulation>py FAS.py
Please enter the file path (e.g. "C:\workspace\Automaton.txt").

F:\workspace\Finite Automaton Simulation\Automaton.txt
Please enter a word.

aa
The word is accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

aba
The word is not accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

abbbbaa
The word is accepted by the automaton
Do you want to test more words? [Y,N]
```

Οι λέξεις aa, abbbbaa γίνονται σωστά αποδέκτες καθώς τελειώνουν με δυο α ενώ η λέξη aba σωστά απορρίπτεται.

**Παράδειγμα 2: Αυτόματο που δέχεται λέξεις που τελειώνουν ή ξεκινούν με 01,  $\Sigma=\{0,1\}$ .**

Αρχείο: Automaton 2.txt

```
F:\workspace\Finite Automaton Simulation>py FAS.py
Please enter the file path (e.g. "C:\workspace\Automaton.txt").

F:\workspace\Finite Automaton Simulation\Automaton 2.txt
Please enter a word.

01
The word is accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

1001
The word is accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

100
The word is not accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

111
The word is not accepted by the automaton
Do you want to test more words? [Y,N]

n
```

Οι λέξεις 01, 1001 γίνονται σωστά αποδέκτες από το αυτόματο καθώς ξεκινούν/τελειώνουν σε 01 ενώ οι λέξεις 100, 111 σωστά απορρίπτονται.

**Παράδειγμα 3: Αυτόματο που δέχεται λέξεις που περιέχουν τουλάχιστον ένα a και τουλάχιστον ένα b με  $\Sigma=\{a,b,c\}$ .**

Αρχείο: Automaton 3.txt

```
F:\workspace\Finite Automaton Simulation>py FAS.py
Please enter the file path (e.g. "C:\workspace\Automaton.txt").

F:\workspace\Finite Automaton Simulation\Automaton 3.txt
Please enter a word.

ab
The word is accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

aaabaac
The word is accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

ca
The word is not accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

bcb
The word is not accepted by the automaton
Do you want to test more words? [Y,N]

n
```

Οι λέξεις ab, aaabaac γίνονται σωστά αποδέκτες από το αυτόματο καθώς περιέχουν ένα ή περισσότερα a και b. Οι λέξεις ca, bcb σωστά απορρίπτονται.



- Μη Αιτιοκρατικά Ντετερμινιστικά Αυτόματα

**Παράδειγμα 1: Αυτόματο που δέχεται λέξεις για τις οποίες υπάρχουν 2 μηδενικά χωρισμένα από έναν αριθμό ψηφίων πολλαπλάσιο του 4, με  $\Sigma=\{0,1\}$ .**

Αρχείο: Non Deterministic Automaton.txt

```
F:\workspace\Finite Automaton Simulation>py FAS.py
Please enter the file path (e.g. "C:\workspace\Automaton.txt").

F:\workspace\Finite Automaton Simulation\Non Deterministic Automaton.txt
Please enter a word.

1001
The word is accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

1010010
The word is accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

10101
The word is not accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

10111110
The word is not accepted by the automaton
Do you want to test more words? [Y,N]

n
```

Οι λέξεις 1001, 1010010 σωστά γίνονται αποδέκτες από το αυτόματο καθώς περιέχουν δυο μηδενικά που χωρίζονται από 0 και 4 ψηφιά. Παράλληλα οι λέξεις 10101, 10111110 σωστά απορρίπτονται γιατί περιέχουν μηδενικά που χωρίζονται από 1 και 5 ψηφιά αντίστοιχα.

## Παράδειγμα 2: Αυτόματο που δέχεται λέξεις της γλώσσας $L=(ab + aba)^*$ .

Αρχείο: Non Deterministic Automaton 2.txt

```
F:\workspace\Finite Automaton Simulation>py FAS.py
Please enter the file path (e.g. "C:\workspace\Automaton.txt").

F:\workspace\Finite Automaton Simulation\Non Deterministic Automaton 2.txt
Please enter a word.

The word is accepted by the automaton
Do you want to test more words? [Y,N]
y
Please enter a word.

ab
The word is accepted by the automaton
Do you want to test more words? [Y,N]
y
Please enter a word.

aba
The word is accepted by the automaton
Do you want to test more words? [Y,N]
y
Please enter a word.

aab
The word is not accepted by the automaton
Do you want to test more words? [Y,N]
n
```

Το αυτόματο σωστά αποδέχεται την κενή λέξη καθώς και τις λέξεις  $ab$ ,  $aba$  καθώς ανήκουν στην γλώσσα του αυτομάτου και σωστά απορρίπτει την  $aab$ .

- Μη Αιτιοκρατικά Ντετερμινιστικά Αυτόματα με  $\epsilon$ -transitions.

**Παράδειγμα 1: Αυτόματο που δέχεται το σύνολο των λέξεων που αποτελούνται από 0 ή περισσότερα a, ακολουθούμενα από 0 ή περισσότερα b και ακολουθούμενα από 0 ή περισσότερα c.**

Αρχείο: NDA with e-transitions.txt

```
F:\workspace\Finite Automaton Simulation>py FAS.py
Please enter the file path (e.g. "C:\workspace\Automaton.txt").

F:\workspace\Finite Automaton Simulation\NDA with e-transitions.txt
Please enter a word.

abc
Transitions List: [['1', 'a', '1'], ['1', '@', '2'], ['2', 'b', '2'], ['2', '@', '3'], ['3', 'c', '3']]

E Transition Matrix: [['1', '1', ' ', ' ', ' ', '2'], [' ', ' ', '2', ' ', ' ', '3'], [' ', ' ', ' ', '3', ' ', ' ']]

E_Closure Matrix: [['1', '2', '3'], ['2', '3', ' '], ['3', ' ', ' ']]

No E Transition Matrix: [['1', '1', '2', '3'], [' ', '2', '3'], [' ', ' ', '3']]

New Transition List: [['1', 'a', '1'], ['1', 'b', '2'], ['1', 'c', '3'], ['2', 'b', '2'], ['2', 'c', '3'], ['3', 'c', '3'], [' ', ' ', ' '], [' ', ' ', ' '], [' ', ' ', ' ']]

Transition Amount before: 5 Transitions Amount now: 6

Accept List before: ['3'] New Accept State List: ['3', '1', '2']

The word is accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

abbcc
The word is accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

c
The word is accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

The word is accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

cba
The word is not accepted by the automaton
Do you want to test more words? [Y,N]

n
```

Στην παραπάνω εικόνα φαίνεται πως η συνάρτηση με τα βήματα που περιεγράφηκαν προηγουμένως, σωστά μετατρέπει την λίστα των transitions με e-transitions στην κατάλληλη λίστα των transitions χωρίς e-transitions, δημιουργώντας τους κατάλληλους πίνακες. Παρατηρώντας το New Transition List βλέπουμε ότι το αυτόματο που προκύπτει στο παράδειγμα αυτό είναι αιτιοκρατικό. Επίσης εκτυπώνονται οι αλλαγές που πραγματοποιεί η συνάρτηση στην λίστα των τελικών states. Το αυτόματο σωστά αποδέχεται τις λέξεις abc, abbcc, c καθώς και την κενή λέξη. Επιπλέον απορρίπτει σωστά την λέξη cba καθώς μετά το διάβασμα του γράμματος c δεν επιτρέπονται τα γράμματα b και a.

Αρχείο: NDA with e-transitions 2.txt

```
F:\workspace\Finite Automaton Simulation\NDA with e-transitions 2.txt
Please enter a word.

ab
Transitions List: [['1', '@', '2'], ['2', '@', '3'], ['2', '@', '6'], ['3', 'a', '4'], ['4', '@', '5'], ['5', 'b', '12'], ['12', '@', '2'], ['6', 'a', '7'], ['7', '@', '8'], ['8', 'a', '9'], ['9', '@', '10'], ['10', 'b', '11'], ['11', '@', '2']]

E Transition Matrix: [[' ', ' ', '2'], [' ', ' ', '3,6'], ['4', ' ', ''], [' ', ' ', '5'], [' ', '12', ''], ['7', ' ', ''], [' ', ' ', '8'], ['9', ' ', ''], [' ', ' ', '10'], [' ', ' ', '11'], [' ', ' ', ' ', '2'], [' ', ' ', ' ', '2']]

E Closure Matrix: [['1', '2', '3', '6', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '], ['2', '3', '6', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '], [' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '], ['4', '5', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '], [' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '], ['6', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '], [' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '], ['7', '8', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '], [' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '], ['9', '10', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '], [' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '], [' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '], ['11', '2', '3', '6', ' ', ' ', ' ', ' ', ' ', ' ', ' '], ['10', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' '], ['12', '2', '3', '6', ' ', ' ', ' ', ' ', ' ', ' ', ' ']]

No E Transition Matrix: [['7 4 ', ' ', ''], ['7 4 ', ' ', ''], ['4', ' ', ''], [' ', '12'], [' ', '12'], ['7 ', ''], ['9 ', ''], ['9', ''], [' ', '11'], [' ', '11'], ['7 4 ', ' ', ''], ['7 4 ', ' ', '']]

New Transition List: [['1', 'a', '7'], ['1', 'a', '4'], ['2', 'a', '7'], ['2', 'a', '4'], ['3', 'a', '4'], ['4', 'b', '12'], ['5', 'b', '12'], ['6', 'a', '7'], ['7', 'a', '9'], ['8', 'a', '9'], ['9', 'b', '11'], ['10', 'b', '11'], ['11', 'a', '7'], ['11', 'a', '4'], ['12', 'a', '7'], ['12', 'a', '4'], [' ', ' ', ' ', ''], [' ', ' ', ' ', ''], [' ', ' ', ' ', ''], [' ', ' ', ' ', ''], [' ', ' ', ' ', ''], [' ', ' ', ' ', '']]

Transition Amount before: 13 Transitions Amount now: 16

Accept List before: ['1', '11', '12'] New Accept State List: ['1', '11', '12']

The word is accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

The word is accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

abaabababab
The word is accepted by the automaton
Do you want to test more words? [Y,N]

y
Please enter a word.

bab
The word is not accepted by the automaton
Do you want to test more words? [Y,N]

n
```

Αν και η ανάγνωση των στοιχείων κάθε πίνακα είναι σχετικά δύσκολη, καθώς πρόκειται για αυτόματο με 12 καταστάσεις, η συνάρτηση προσομοιώνει μια σωστή μετατροπή 'με το χέρι' του αυτομάτου με e-transitions στο αντίστοιχο αυτόματο χωρίς e-transitions. Από το New Transition List φαίνεται ότι το αυτόματο που προέκυψε είναι μη αιτιοκρατικό καθώς για παράδειγμα, η κατάσταση 1 έχει δυο μεταβάσεις για το γράμμα 'α' προς τις καταστάσεις 4 και 7. Οι λέξεις ab, abaabababab καθώς και η κενή λέξη σωστά γίνονται αποδέκτες από το αυτόματο, ενώ η bab απορρίπτεται καθώς δεν ανήκει στην γλώσσα του αυτομάτου.

### 3. Εκτέλεση προγράμματος

Το πρόγραμμα μπορεί να εκτελεστεί μέσω κάποιου IDE (π.χ. Eclipse) ή της γραμμής εντολών (μέσω της εντολής `py FAS.py`).

### 4. Λογισμικά, γλώσσα προγραμματισμού και modules

Η συνάρτηση γράφτηκε με Python 3.8 μέσω του Eclipse IDE.

Εσωτερικά στο πρόγραμμα έγινε χρήση των εξής modules:

- a) `Linecache`: για διάβασμα γραμμή προς γραμμή του αρχείου εισόδου.
- b) `Sys`: για τον τερματισμό του προγράμματος.
- c) `Copy`: για την δημιουργία deep copy λίστας.