Περιεχόμενα

[Εργασία - Θέμα 1 1](#_Toc63981125)

[Εκφώνηση 1](#_Toc63981126)

[Dependencies, τρόπος εκτέλεσης προγράμματος και τρόπος εκκίνησης εκτελέσιμου 1](#_Toc63981127)

[Παράδειγμα εκτέλεσης 2](#_Toc63981128)

[3](#_Toc63981129)

[Ντετερμινιστικό Αυτόματο Στοίβας 4](#_Toc63981130)

[Περιγραφή αρχείων πηγαίου κώδικα 4](#_Toc63981131)

[Τρόπος λειτουργίας αυτόματου – Αλγοριθμική ανάπτυξη της προγραμματιστικής υλοποίησης 5](#_Toc63981132)

[Βιβλιογραφία 6](#_Toc63981133)

# Εργασία - Θέμα 1

## Εκφώνηση

Υλοποιήστε Ντετερμινιστικό Αυτόματο Στοίβας (ΝΑΣ) που αναγνωρίζει εκφράσεις αποτελούμενες από τους χαρακτήρες ‘x’ και ‘y’, κατά τρόπον ώστε:

- όσοι χαρακτήρες ‘x’ εμφανίζονται συνολικά, άλλοι τόσοι χαρακτήρες ‘y’ εμφανίζονται συνολικά

- κοιτάζοντας την έκφραση από αριστερά προς τα δεξιά οι χαρακτήρες ‘y’ δεν είναι ποτέ περισσότεροι από τους χαρακτήρες ‘x’. Να τυπώνεται η αλληλουχία βημάτων που οδήγησαν στην αναγνώριση (ή στην απόρριψη) της έκφρασης.

## Dependencies, τρόπος εκτέλεσης προγράμματος και τρόπος εκκίνησης εκτελέσιμου

Το πρόγραμμα υλοποιήθηκε σε C++, σε Linux. Χρησιμοποιήθηκαν οι βιβλιοθήκες της C++ <stack> για χρήση στοίβας, <string> για χρήση string, <queue> για χρήση ουράς, <iostream> για την εκτύπωση μηνυμάτων στον χρήστη.

Σε περιβάλλον τερματικού (terminal):

* Εκτέλεση προγράμματος **g++ Subject1.cpp -o Subject1.out**
* Εκκίνηση εκτελέσιμου αρχείου (εφόσον το working directory είναι το directory στο οποίο δημιουργήθηκε το εκτελέσιμο με την παραπάνω εντολή) **./Subject1.out inputString**

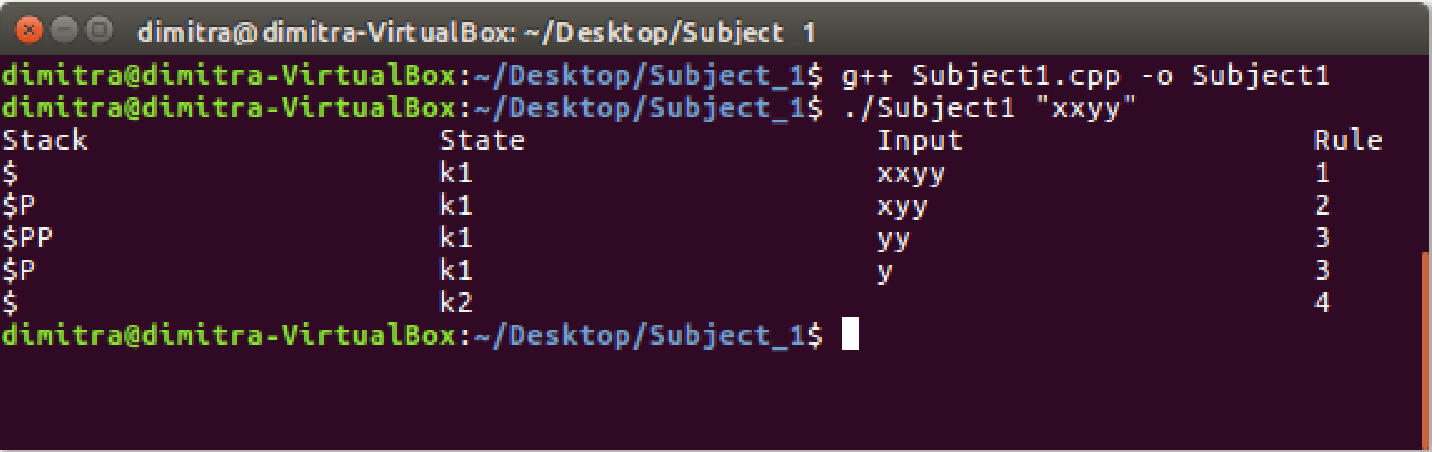
Όπου inputString = String με χαρακτήρες «x» και «y», η είσοδος που δίνει ο χρήστης για αναγνώριση από το Ντετερμινιστικό Αυτόματο Στοίβας. (Προσοχή! Η συμβολοσειρά εισόδου πρέπει να εσωκλειστεί σε “ ”). Επίσης ενδείκνυται ο χρήστης να χρησιμοποιεί λατινικούς χαρακτήρες στην είσοδο. Τέλος το extension του εκτελέσιμου μπορεί και να παραληφθεί επειδή εργαζόμαστε σε Linux.

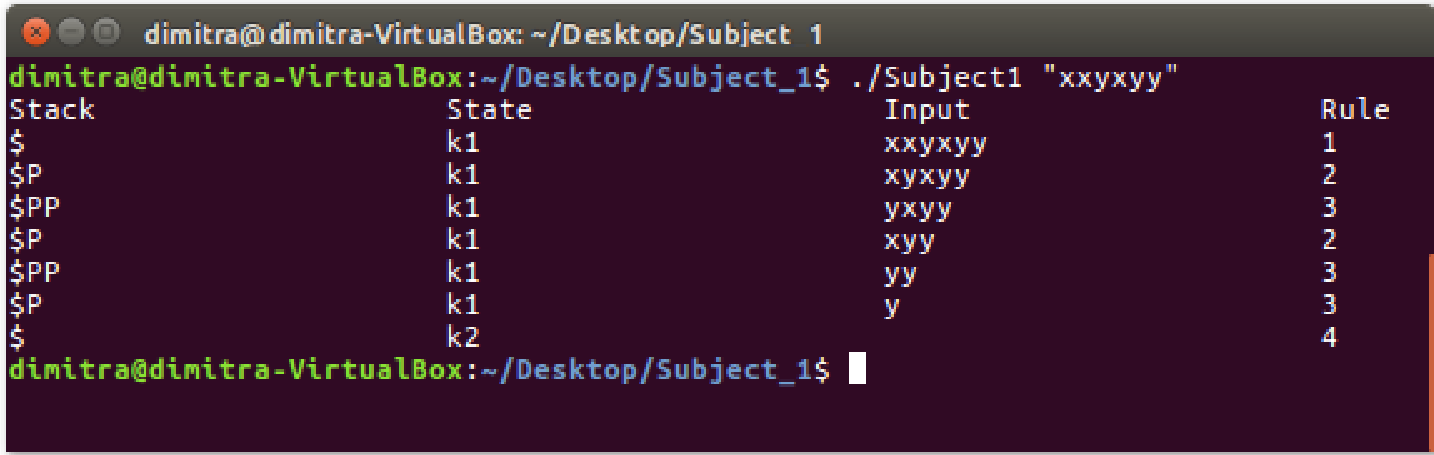
## Παράδειγμα εκτέλεσης

Σε περιβάλλον τερματικού καλούμε το εκτελέσιμο αρχείο (Subject1) και δίνουμε σαν όρισμα τη συμβολοσειρά εισόδου που επιθυμούμε. (παραδείγματος χάριν command 🡪 ./Subject1 “xxyy”) Παρακάτω επισυνάπτουμε ενδεικτικά screenshots τριών παραδειγμάτων εκτέλεσης.

Στην 1η στήλη εμφανίζονται τα περιεχόμενα (ή αλλιώς βάθος) στοίβας, στη 2η στήλη εμφανίζεται η τρέχουσα κατάσταση στην οποία βρίσκεται το αυτόματο, στη 3η στήλη εμφανίζεται η συμβολοσειρά εισόδου (διαγράφεται κάθε φορά το σύμβολο που αναγνωρίστηκε έπειτα και από την εφαρμογή κατάλληλου κανόνα) και τέλος στην 4η στήλη εμφανίζεται ο κανόνας που χρησιμοποιήθηκε (η αρίθμηση των κανόνων είναι ίδια με αυτή που διατυπώνεται παρακάτω).

Εικόνα 1 Επιτυχής αναγνώριση συμβολοσειράς "xxyy"





Εικόνα 2 Επιτυχής αναγνώριση συμβολοσειράς "xxyxyy"

## 

Εικόνα 3 Ανεπιτυχής αναγνώριση συμβολοσειράς "xxyx", η συμβολοσειρά δεν αναγνωρίστηκε

## Ντετερμινιστικό Αυτόματο Στοίβας

Το ζητούμενο Ντετερμινιστικό Αυτόματο Στοίβας χαρακτηρίζεται από την επτάδα M = (K, T, V, p, k1, $, {k2}) όπου:

* K το σύνολο καταστάσεων, Κ = {k1, k2}
* T το αλφάβητο εισόδου, T = {“x”, “y”, e} όπου e συμβολίζει το κενό
* V το αλφάβητο συμβόλων στοίβας, V = {P, $}
* Το p αποτελεί τη συνάρτηση στοίβας με τους εξής κανόνες:

1. p(k1, $, “x”) = (k1, $P)
2. p(k1, P, “x”) = (k1, PP)
3. p(k1, P, “y”) = (k1, e)
4. p(k1, e, “y”) = (k2, e)

* η k1 αποτελεί την αρχική κατάσταση
* το $ αποτελεί το αρχικό σύμβολο στοίβας
* το {k2} αποτελεί το σύνολο των τελικών καταστάσεων, με μία αποδεκτή τελική κατάσταση την k2

Πίνακας ελέγχου αυτομάτου

|  |  |  |  |
| --- | --- | --- | --- |
| **V/T** | **x** | **y** | **e** |
| **P** | Push(P) | Pop |  |
| **$** | Push(P) |  | k2 |

## Περιγραφή αρχείων πηγαίου κώδικα

**Subject1.cpp** περιέχει την Main συνάρτηση της λύσης, δέχεται ως όρισμα ένα string που περιέχει τους χαρακτήρες «x» και «y», τους τοποθετεί σε μια δομή σειράς (Queue) και ξεκινά την διαδικασία αναγνώρισης.

**Automaton.h** header file που περιέχει την υλοποίηση του Ντετερμινιστικού Αυτόματου Στοίβας, ο τρόπος λειτουργίας θα αναπτυχθεί λεπτομερώς παρακάτω.

**State.h** header file που περιέχει την υλοποίηση της κατάστασης που χρησιμοποιείται από το αυτόματο (περιλαμβάνει βασικά attributes που καθορίζουν έναν κανόνα και απαιτούνται για την επιλογή του κατάλληλου κανόνα από το αυτόματο, όπως τρέχουσα κατάσταση, κατάσταση μετάβασης, χαρακτήρας που υπάρχει στην κορυφή της στοίβας κ. α.) .

**PrettyPrint.h** header file που χρησιμοποιείται καθαρά και μόνο για την εκτύπωση του πίνακα καταστάσεων, με όσο το δυνατό πιο ευανάγνωστο τρόπο, στον χρήστη.

## Τρόπος λειτουργίας αυτόματου – Αλγοριθμική ανάπτυξη της προγραμματιστικής υλοποίησης

Ο τρόπος λειτουργίας του αυτόματου είναι αρκετά απλός. Μέσω της μεθόδου Start περνάμε τη σειρά χαρακτήρων που εισήγαγε ο χρήστης, αφού πρώτα έχουμε τοποθετήσει τη συμβολοσειρά εισόδου σε ένα queue (χρησιμοποιήθηκε ουρά σαν δομή, διότι κάθε φορά το αυτόματο επιχειρεί να αναγνωρίσει τον 1ο από τα αριστερά χαρακτήρα της συμβολοσειράς, και εφόσον τον αναγνωρίσει τον διαγράφει από την ουρά, αξιοποιήθηκε η ιδιότητα της δομής queue First-In-First-Out).

Στη συνέχεια, καλώντας κάθε φόρα τη μέθοδο Feed και τροφοδοτώντας τη με ένα χαρακτήρα, το αυτόματο ελέγχει αν υπάρχει κανόνας που να ικανοποιεί την καινούργια είσοδο. Για την εύρεση του κατάλληλου κανόνα, ελέγχεται:

1. η κατάσταση στην οποία βρίσκεται το αυτόματο
2. το σύμβολο που υπάρχει στην κορυφή της στοίβας
3. καθώς και το σύμβολο εισόδου (κάθε φορά δίνεται σαν όρισμα στη Feed() ο πρώτος χαρακτήρας που βρίσκεται στην ουρά που περιλαμβάνει τη συμβολοσειρά εισόδου

Αν υπάρχει λοιπόν κανόνας που οι παραπάνω τρεις προδιαγραφές του ταιριάζουν με τα δεδομένα της στοίβας, του αυτομάτου και της συμβολοσειράς τη συγκεκριμένη χρονική στιγμή, τότε εφαρμόζεται ο κανόνας ως εξής:

1. Ανάλογα με τον κανόνα που πρέπει να εφαρμοστεί το αυτόματο είτε παραμένει στην ίδια κατάσταση ή μεταβαίνει σε καινούρια
2. Στη συνέχεια είτε τοποθετείται νέο σύμβολο στην κορυφή της στοίβας, είτε αφαιρείται το σύμβολο που υπάρχει στην κορυφή της στοίβας, αντικαθίσταται δηλαδή με το κενό
3. Και τέλος αφαιρείται από την ουρά το 1ο σύμβολο, δηλαδή ο χαρακτήρας που μόλις αναγνωρίστηκε.

Το αυτόματο συνεχίζει κατά τον ίδιο τρόπο μέχρι το πέρας της σειράς χαρακτήρων που έδωσε ο χρήστης, όπου ολοκληρώνεται και η διαδικασία αναγνώρισης. Συγκεκριμένα, η διαδικασία της αναγνώρισης ολοκληρώνεται όταν:

* η στοίβα περιέχει μόνο το αρχικό της σύμβολο, δηλαδή το $
* έχουν καταναλωθεί (δηλαδή επεξεργαστεί και αναγνωριστεί) όλοι οι χαρακτήρες από τη συμβολοσειρά εισόδου
* όταν κατάσταση του αυτόματου είναι η τελική κατάσταση, δηλαδή η k2

Μόνο εφόσον ισχύουν και οι 3 παραπάνω προϋποθέσεις, η συμβολοσειρά εισόδου έχει αναγνωριστεί επιτυχώς. Αν όμως κατά τη διάρκεια αναγνώρισης δεν υπάρχει κανόνας που να ικανοποιεί την είσοδο, δηλαδή τα δεδομένα της στοίβας, του αυτομάτου και της συμβολοσειράς για τη συγκεκριμένη χρονική στιγμή δεν ταιριάζουν με κανενός κανόνα τις προδιαγραφές, τότε το αυτόματο επιστρέφει μήνυμα σφάλματος στον χρήστη και η διαδικασία αναγνώρισης σταματά.

# Βιβλιογραφία

Χρυσαφιάδη, Δ. Κ. (χ.χ.). *Παραδείγματα για Αυτόματα Στοίβας.* Ανάκτηση από gunet2: https://gunet2.cs.unipi.gr/modules/document/file.php/TMB100/%ce%a3%cf%85%ce%bc%cf%80%ce%bb%ce%b7%cf%81%cf%89%ce%bc%ce%b1%cf%84%ce%b9%ce%ba%ce%ad%cf%82%20%ce%94%ce%b9%ce%b1%cf%86%ce%ac%ce%bd%ce%b5%ce%b9%ce%b5%cf%82%202017-2018/%ce%a0%ce%b1%cf%81%ce%b1%ce%b