

ΠΑΝΕΠΙΣΤΗΜΙΟ ΜΑΚΕΔΟΝΙΑΣ
ΤΜΗΜΑ ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΘΕΩΡΙΑ ΥΠΟΛΟΓΙΣΜΩΝ ΚΑΙ ΑΥΤΟΜΑΤΩΝ

Ονοματεπώνυμο: ΠΑΠΑΓΕΩΡΓΙΟΥ ΓΕΩΡΓΙΟΣ

Αριθμός Μητρώου: dai17233

1η Εργασία

Το αρχείο *Nondeterministic Finite Automata.exe* είναι ένα πρόγραμμα στην γλώσσα προγραμματισμού C, το οποίο δέχεται ως είσοδο από **αρχείο κειμένου (*.txt)** την **περιγραφή** ενός **πεπερασμένου αυτομάτου**, έπειτα **δέχεται** διάφορες **λέξεις** από το πληκτρολόγιο και **απαντά** εάν κάθε λέξη **ανήκει ή όχι** στη γλώσσα του αυτομάτου. Το πρόγραμμα λειτουργεί για **μη-ντετερμινιστικά αυτόματα χωρίς ε-μεταβάσεις**. Ο **πηγαίος κώδικας** του προγράμματος βρίσκεται στο αρχείο *Nondeterministic Finite Automata.c*.

Επεξήγηση κώδικα:

- 1) Αρχικά δημιουργήσα μια **δομή (struct)**, με το όνομα **name**, στην οποία υπάρχει ένας πίνακας χαρακτήρων **char name[]**.

Μετά έφτιαξα **πίνακες** τύπου **name** για την **αποθήκευση** των **καταστάσεων**, των **λέξεων/χαρακτήρων που εισάγει ο χρήστης** και των **γραμμάτων του αλφάβητου**:

- name states[] – Αποθήκευση ενδιάμεσων καταστάσεων
- name alphabets[] – Αποθήκευση κάθε γράμματος που χρησιμοποιεί το αλφάβητο
- name initialStates[] – Αποθήκευση αρχικών καταστάσεων
- name finalStates[] – Αποθήκευση τελικών καταστάσεων
- name Buf[] – Αποθήκευση των λέξεων/χαρακτήρων που εισάγει ο χρήστης

Επιπλέον, δημιουργήσα μεταβλητές τύπου **int** για την **αποθήκευση** του **αριθμού** των **καταστάσεων** και των **γραμμάτων του αλφάβητου** που έχει το αυτόματό μου:

- numberOfStates – Αριθμός ενδιάμεσων καταστάσεων
- numberOfAlphabets – Αριθμός γραμμάτων του αλφάβητου
- numberOfInitialStates – Αριθμός αρχικών καταστάσεων
- numberOfFinalStates – Αριθμός τελικών καταστάσεων

- 2) Δημιούργησα μια **δομή(struct)**, με όνομα **edge**, η οποία έχει **τρεις πίνακες χαρακτήρων**.
Αυτή η δομή αναλαμβάνει την **αποθήκευση της διαδρομής** του αυτομάτου, από **κατάσταση σε κατάσταση**, μέσω **ακμών**.

- `char from[]` – Αποθήκευση της κατάστασης από όπου ξεκίνησε η μετάβαση(βελάκι)
- `char to[]` – Αποθήκευση της κατάστασης που πήγε η μετάβαση(βελάκι)
- `char alphabet[]` – Αποθήκευση του χαρακτήρα με τον οποίο έγινε η αλλαγή της κατάστασης

Επιπλέον, χρησιμοποιώ μια μεταβλητή τύπου `int`, στην οποία αποθηκεύεται ο αριθμός των μεταβάσεων(βελάκια).

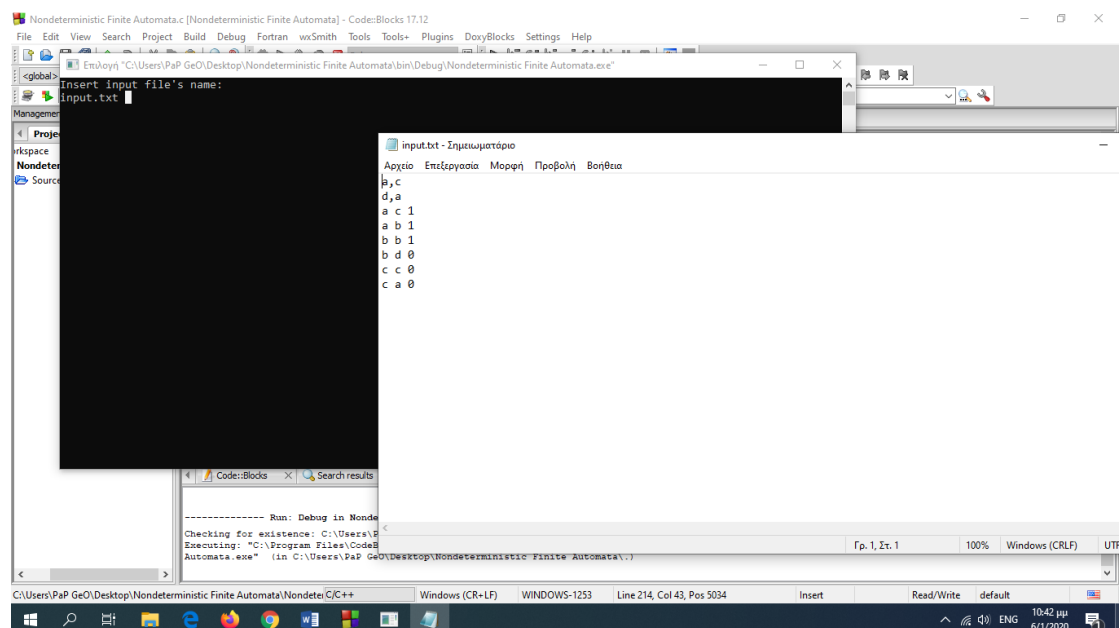
- 3) Το πρόγραμμα χρησιμοποιεί τις παρακάτω **συναρτήσεις/διαδικασίες**:

- **`void GetInput()`**
Πρώτα ζητάει το **όνομα του αρχείου** από το οποίο θα **διαβάσει την περιγραφή του αυτομάτου**.
Αναλαμβάνει να **διαβάσει την περιγραφή του αυτομάτου** από το **αρχείο κειμένου(*.txt)** που έδωσε ο χρήστης και να **αποθηκεύσει τις προδιαγραφές(αριθμός καταστάσεων, αρχικές καταστάσεις, τελικές καταστάσεις, αριθμός μεταβάσεων, περιγραφή των μεταβάσεων)** του αυτομάτου στις **κατάλληλες μεταβλητές της δομής(struct) name**.
- **`void printEdges()`**
Αυτή η συνάρτηση **τυπώνει τον αριθμό όλων των μεταβάσεων του αυτομάτου και την αναπαράστασή τους**.
Συγκεκριμένα, τυπώνει **κάθε μετάβαση με τη μορφή: (1) ---(2)---> (a)**, Όπου (1) = κατάσταση έναρξης, (2) = κατάσταση στην οποία μετέβη, (a) = ο χαρακτήρας που χρησιμοποιήθηκε για να γίνει αυτή η μετάβαση.
- **`void getAutomaton()`**
Δημιουργία του αυτομάτου, με τις **κατάλληλες καταστάσεις, μεταβάσεις και αλφάβητο**, ανάλογα με τις **προδιαγραφές** που είναι **αποθηκευμένες στις μεταβλητές τύπου name: states[], alphabets[], initialStates[] και finalStates[] και τύπου int: numberOfEdges, numberOfStates, numberOfAlphabets, numberOfInitialStates, numberOfFinalStates**.
- **`void insertCharacters()`**
Η συγκεκριμένη συνάρτηση **διαβάσει μια σειρά χαρακτήρων ή μια λέξη από τον χρήστη και έπειτα καλεί την `void check()` για να ελέγξει αν το αυτόματο τερμάτισε σε τελική ή μη-τελική κατάσταση**.
Μετά τον έλεγχο γίνεται σχετική ερώτηση και ανάλογα με την απάντηση, η διαδικασία **επαναλαμβάνεται ή τερματίζεται η εκτέλεση του προγράμματος**.

- **void check()**
Αρχικά παίρνει τους **χαρακτήρες**(ή την **λέξη**) που εισήγαγε ο χρήστης μέσω της συνάρτησης **void insertCharacters()** και έπειτα **ελέγχει** εάν το αυτόματο **τερμάτισε** σε **τελική** ή **μη-τελική κατάσταση** και **εμφανίζει** ανάλογο **μήνυμα** στην οθόνη.
- **int isInitial(char name[])**
Δέχεται μια **κατάσταση** και **ελέγχει** εάν είναι **αρχική** ή όχι.
Αν επιστρέψει 1, η κατάσταση είναι αρχική.
Αν επιστρέψει 0, η κατάσταση δεν είναι αρχική.
- **int isFinal(char name[])**
Είναι παρόμοια με την **int isInitial(char name[])**, αλλά **ελέγχει** αν η **κατάσταση** που δέχεται ως είσοδο είναι **τελική** ή όχι.
Αν επιστρέψει 1, η κατάσταση είναι τελική.
Αν επιστρέψει 0, η κατάσταση δεν είναι τελική.
- **void getAndPrintAutomaton()**
Καλεί τις συναρτήσεις **getInput()**, **getAutomaton()** και **printEdges()** με την σειρά για την **κατασκευή** και την **εκτύπωση** του **αυτομάτου**. Έπειτα **εκτυπώνει** τις **αρχικές**, **τελικές** και **ενδιάμεσες καταστάσεις** του αυτομάτου και τον **αριθμό των χαρακτήρων του αλφάβητου**.
Επιπλέον, **καλεί** την **void insertCharacters()**, για την **εισαγωγή χαρακτήρων/λέξεων** από τον **χρήστη**.

Περιγραφή εκτέλεσης:

Αρχικά το πρόγραμμα ζητάει το όνομα του αρχείου που περιέχει την περιγραφή του αυτόματου (έστω "input.txt").



Μετά την ορθή εισαγωγή του ονόματος του αρχείου, το πρόγραμμα συνεχίζει την εκτέλεση, δημιουργεί το αυτόματο και τυπώνει τον αριθμό των μεταβάσεων και όλες τις μεταβάσεις, τον αριθμό των καταστάσεων και την κατηγορία της κάθε μίας(αρχική, ενδιάμεση, τελική) και τους χαρακτήρες που περιέχει το αλφάβητο του αυτομάτου.

Στην συγκεκριμένη περίπτωση, το αυτόματο έχει 5 μεταβάσεις(edges), 4 καταστάσεις(states), αρχικές καταστάσεις (a, c), ενδιάμεσες καταστάσεις (b), τελικές καταστάσεις (a, d) και το αλφάβητο περιέχει 2 χαρακτήρες (1, 0).

Έπειτα, εμφανίζει σχετικό μήνυμα(“Insert characters”) και περιμένει την εισαγωγή χαρακτήρων/λέξεων από τον χρήστη.

```
Automation Edges:
Number of Edges: 5
(a) ---(c)---> (1)
(a) ---(b)---> (1)
(b) ---(b)---> (1)
(b) ---(d)---> (0)
(c) ---(c)---> (0)
(c) ---(a)---> (0)

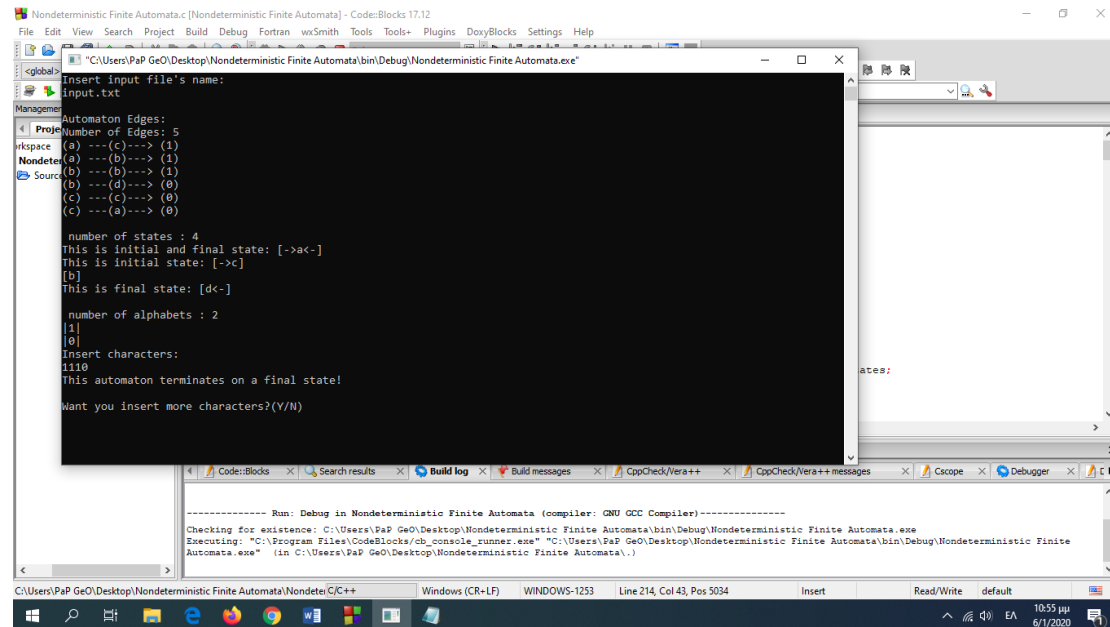
number of states : 4
This is initial and final state: [->a<-]
This is initial state: [->c]
[b]
This is final state: [d<-]

number of alphabets : 2
[1]
[0]
Insert characters:

----- Run: Debug in Nondeterministic Finite Automata (compiler: GNU GCC Compiler)-----
Checking for existence: C:\Users\PaP GeO\Desktop\Nondeterministic Finite Automata\bin\Debug\Nondeterministic Finite Automata.exe
Executing: "C:\Program Files\CodeBlocks\cb_console_runner.exe" "C:\Users\PaP GeO\Desktop\Nondeterministic Finite Automata\bin\Debug\Nondeterministic Finite Automata.exe" (in C:\Users\PaP GeO\Desktop\Nondeterministic Finite Automata\.)
```

Αφού ο χρήστης εισάγει μια σειρά χαρακτήρων, το πρόγραμμα ελέγχει εάν το αυτόματο τερμάτισε σε τελική ή μη-τελική θέση και εμφανίζει κατάλληλο μήνυμα.

Μετά γίνεται ερώτηση για το αν ο χρήστης θέλει να εισάγει άλλους χαρακτήρες(ή λέξεις) και η διαδικασία επαναλαμβάνεται, με την χρήση του ίδιου αυτομάτου.



```
Non-deterministic Finite Automata.c [Non-deterministic Finite Automata] - Code::Blocks 17.12
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help

C:\Users\PaP GeO\Desktop\Non-deterministic Finite Automata\bin\Debug\Non-deterministic Finite Automata.exe
Insert input file's name:
input.txt

Automaton Edges:
Number of Edges: 5
(a) ---(c)---> (1)
(b) ---(b)---> (1)
(b) ---(d)---> (0)
(c) ---(c)---> (0)
(c) ---(a)---> (0)

number of states : 4
This is initial and final state: [->a<-]
This is initial state: [->c]
[b]
This is final state: [d<-]

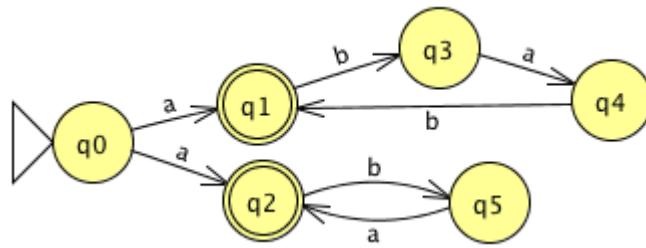
number of alphabets : 2
{1}
{0}
Insert characters:
1110
This automaton terminates on a final state!
Want you insert more characters?(Y/N)

----- Run: Debug in Non-deterministic Finite Automata (compiler: GNU GCC Compiler)-----
Checking for existence: C:\Users\PaP GeO\Desktop\Non-deterministic Finite Automata\bin\Debug\Non-deterministic Finite Automata.exe
Executing: "C:\Program Files\CodeBlocks\cb_console_runner.exe" "C:\Users\PaP GeO\Desktop\Non-deterministic Finite Automata\bin\Debug\Non-deterministic Finite Automata.exe" (in C:\Users\PaP GeO\Desktop\Non-deterministic Finite Automata\.)

C:\Users\PaP GeO\Desktop\Non-deterministic Finite Automata\Nondete\C\C++ Windows (CR+LF) WINDOWS-1253 Line 214, Col 43, Pos 5034 Insert Read/Write default 10:55 μμ 6/1/2020
```

Εκτελέσεις με διάφορα αυτόματα:

1. Στο αρχείο *auto1.txt* περιγράφεται το παρακάτω αυτόματο:



Το πρόγραμμά μου δίνει τα παρακάτω αποτελέσματα:

```
Insert input file's name:
auto1.txt

Automaton Edges:
Number of Edges: 6
(q0 --- (q1)---> (a)
(q1 --- (q3)---> (b)
(q3 --- (q4)---> (a)
(q4 --- (q1)---> (b)
(q0 --- (q2)---> (a)
(q2 --- (q5)---> (b)
(q5 --- (q2)---> (a)

number of states : 6
This is initial state: [->q0]
This is final state: [q1<-]
[q3]
[q4]
This is final state: [q2<-]
[q5]

number of alphabets : 2
[a]
[b]
Insert characters:

auto1.txt - Σημειωματάριο
Αρχείο  Επεξεργασία  Μορφή  Προβολή  Βοήθεια
q0
q1,q2
q0 q1 a
q1 q3 b
q3 q4 a
q4 q1 b
q0 q2 a
q2 q5 b
q5 q2 a

----- Run: Debug in Nondeterministic Finite Automata
Checking for existence: C:\Users\PaP\Program Files\CodeBlocks\bin\Debug\Nondeterministic Finite Automata.exe
Executing: "C:\Program Files\CodeBlocks\bin\Debug\Nondeterministic Finite Automata.exe" (in C:\Users\PaP\Program Files\CodeBlocks\bin\Debug)
```

Αριθμός μεταβάσεων: 6

Αριθμός καταστάσεων: 6

Αρχικές καταστάσεις: q0

Ενδιάμεσες καταστάσεις: q3, q4, q5

Τελικές καταστάσεις: q1, q2

Αλφάβητο: a, b

-Με την εισαγωγή των χαρακτήρων “ababbab” το αυτόματο τερματίζει σε τελική κατάσταση(q1):

The screenshot shows the NFA application interface. The main window displays the automaton's configuration after processing the input string "ababbab". The output indicates that the automaton terminates on a final state, specifically q1. The interface includes a menu bar, a toolbar, and a status bar. The console window at the bottom shows the execution details, including the compiler used (GNU GCC Compiler) and the path to the executable file.

```
Insert input file's name:
Malauto1.txt

Automaton Edges:
Number of Edges: 6
(q0) --- (q1) --> (a)
(q1) --- (q3) --> (b)
(q3) --- (q4) --> (a)
(q4) --- (q1) --> (b)
(q0) --- (q2) --> (b)
(q2) --- (q5) --> (b)
(q5) --- (q2) --> (a)

number of states : 6
This is initial state: [->q0]
This is final state: [q1<-]
[q3]
[q4]
This is final state: [q2<-]
[q5]

number of alphabets : 2
[a]
[b]
Insert characters:
ababbab
This automaton terminates on a final state!
Want you insert more characters?(Y/N) _
```

-Με την εισαγωγή των χαρακτήρων “abb” το αυτόματο τερματίζει σε μη-τελική κατάσταση(q3 ή q5):

The screenshot shows the NFA application interface. The main window displays the automaton's configuration after processing the input string "abb". The output indicates that the automaton terminates on a non-final state, specifically q3. The interface includes a menu bar, a toolbar, and a status bar. The console window at the bottom shows the execution details, including the compiler used (GNU GCC Compiler) and the path to the executable file.

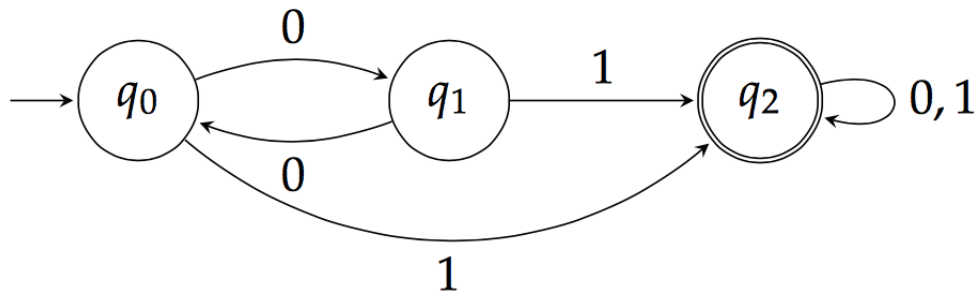
```
Insert input file's name:
auto1.txt

Automaton Edges:
Number of Edges: 6
(q0) --- (q1) --> (a)
(q1) --- (q3) --> (b)
(q3) --- (q4) --> (a)
(q4) --- (q1) --> (b)
(q0) --- (q2) --> (b)
(q2) --- (q5) --> (b)
(q5) --- (q2) --> (a)

number of states : 6
This is initial state: [->q0]
This is final state: [q1<-]
[q3]
[q4]
This is final state: [q2<-]
[q5]

number of alphabets : 2
[a]
[b]
Insert characters:
abb
This automaton terminates on a non-final state!
Want you insert more characters?(Y/N)
```

2. Στο αρχείο *auto2.txt* περιγράφεται το παρακάτω αυτόματο:



Το πρόγραμμά μου δίνει τα παρακάτω αποτελέσματα:

```
Automaton Edges:
Number of Edges: 5
(q0) --(q1)--> (0)
(q1) --(q0)--> (0)
(q1) --(q2)--> (1)
(q2) --(q2)--> (0)
(q2) --(q2)--> (1)
(q0) --(q2)--> (0)

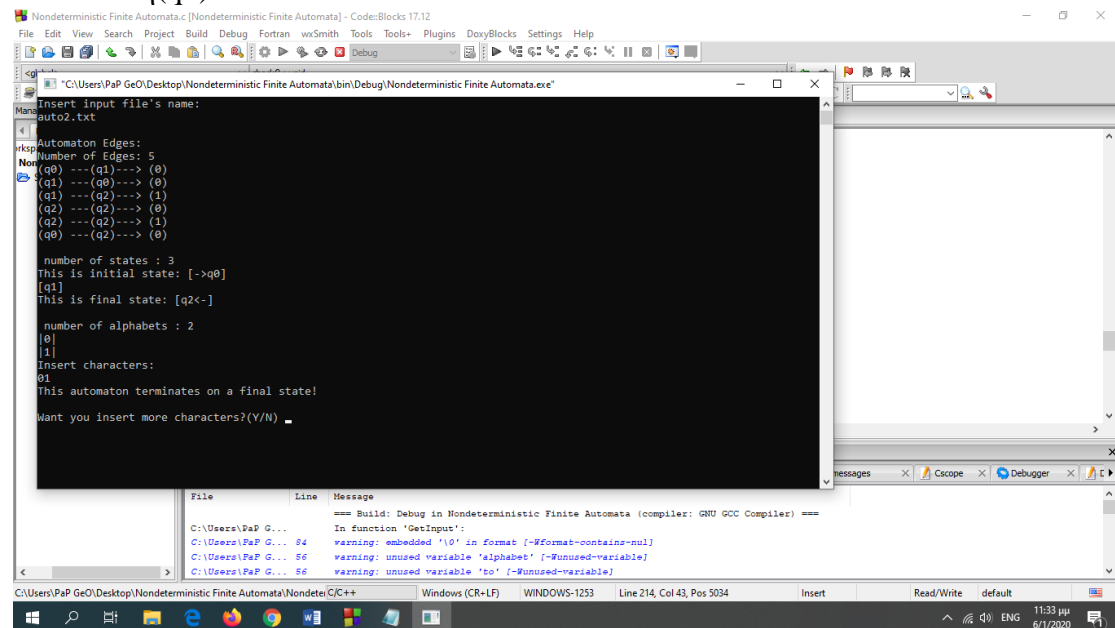
number of states : 3
This is initial state: [->q0]
This is final state: [q2<-]

number of alphabets : 2
[0]
[1]
Insert characters:
```

```
auto2.txt - Σημειώσεις
Αρχείο  Επεξεργασία  Μορφή  Προβολή  Βοήθεια
q0
q2
q0 q1 0
q1 q0 0
q1 q2 1
q2 q2 0
q2 q2 1
q0 q2 0
```

Αριθμός μεταβάσεων: 5
Αριθμός καταστάσεων: 3
Αρχικές καταστάσεις: q0
Ενδιάμεσες καταστάσεις: q1
Τελικές καταστάσεις: q2
Αλφάβητο: a, b

-Με την εισαγωγή των χαρακτήρων “01” το αυτόματο τερματίζει σε τελική κατάσταση(q2):



The screenshot shows the application window with the following text in the console:

```
Insert input file's name:
auto2.txt

Automaton Edges:
Number of Edges: 5
(q0) --(q1)--> (0)
(q1) --(q0)--> (0)
(q1) --(q2)--> (1)
(q2) --(q2)--> (0)
(q2) --(q2)--> (1)
(q0) --(q2)--> (0)

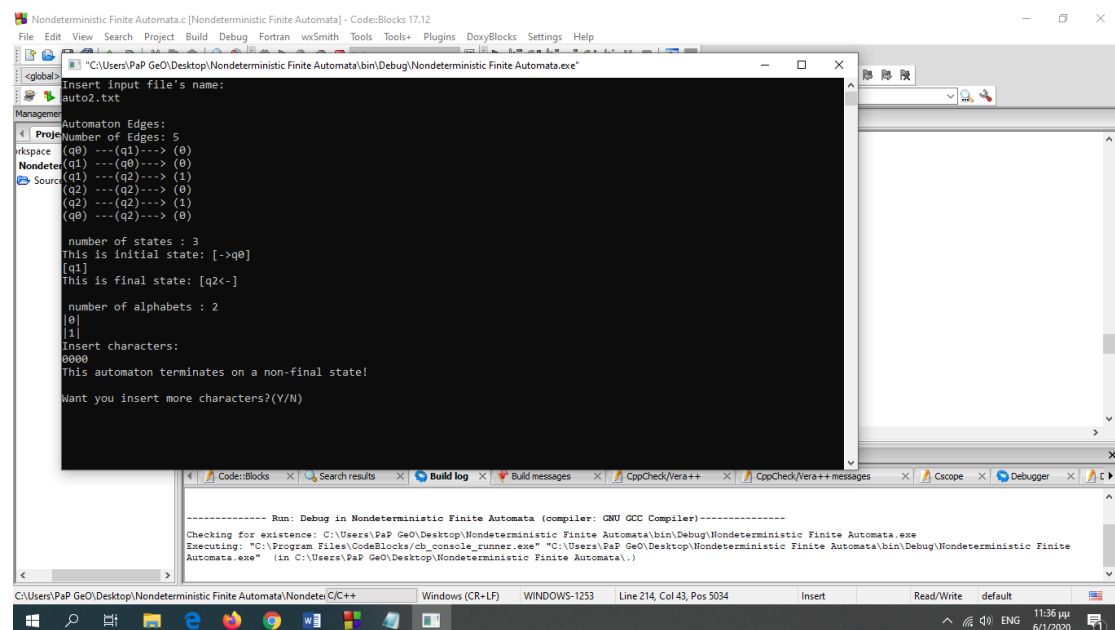
number of states : 3
This is initial state: [->q0]
[q1]
This is final state: [q2<-]

number of alphabets : 2
[0]
[1]
Insert characters:
q1
This automaton terminates on a final state!
Want you insert more characters?(Y/N)
```

The bottom panel shows the build output:

```
==== Build: Debug in Nondeterministic Finite Automata (compiler: GNU GCC Compiler) ====
In function 'GetInput':
C:\Users\PaP G... 84 warning: embedded '\0' in format [-Wformat-contains-nul]
C:\Users\PaP G... 86 warning: unused variable 'alphabet' [-Wunused-variable]
C:\Users\PaP G... 86 warning: unused variable 'to' [-Wunused-variable]
```

-Με την εισαγωγή των χαρακτήρων “0000” το αυτόματο τερματίζει σε μη-τελική κατάσταση(q1):



The screenshot shows the application window with the following text in the console:

```
Insert input file's name:
auto2.txt

Automaton Edges:
Number of Edges: 5
(q0) --(q1)--> (0)
(q1) --(q0)--> (0)
(q1) --(q2)--> (1)
(q2) --(q2)--> (0)
(q2) --(q2)--> (1)
(q0) --(q2)--> (0)

number of states : 3
This is initial state: [->q0]
[q1]
This is final state: [q2<-]

number of alphabets : 2
[0]
[1]
Insert characters:
0000
This automaton terminates on a non-final state!
Want you insert more characters?(Y/N)
```

The bottom panel shows the build output:

```
----- Run: Debug in Nondeterministic Finite Automata (compiler: GNU GCC Compiler)-----
Checking for existence: C:\Users\PaP GeO\Desktop\Nondeterministic Finite Automata\bin\Debug\Nondeterministic Finite Automata.exe
Executing: "C:\Program Files\CodeBlocks\cb_console_runner.exe" "C:\Users\PaP GeO\Desktop\Nondeterministic Finite Automata\bin\Debug\Nondeterministic Finite Automata.exe" (in C:\Users\PaP GeO\Desktop\Nondeterministic Finite Automata\.)
```