Vol. 3 No. 1 July 2015 ISSN: 2321 – 788X

# AN EFFICIENT ALGORITHM TO DESIGN DFA THAT ACCEPT STRINGS OVER THE INPUT SYMBOL a, b, c HAVING ATMOST X NUMBER OF a, Y NUMBER OF b & Z NUMBER OF c

**S.Shanmugavadivoo**
Faculty in Mathematics, Madurai Kamaraj University College, Aundipatti, Theni
**Dr. M.Kamaraj**
Associate Professor of Mathematics, Government Arts & Science College, Sivakasi

***Abstract***

*Automata theory has played an important role in modeling behavior of systems In this paper we propose an algorithm to construct a DFA that accept strings over input symbols a, b, c having accept strings over input symbols a, b, c having atmost x number of a, y number of b & z number of c.*
***Keywords***: *DFA, Automata, strings, Implementation, symbol*

## Introduction

Automata theory has proved to be a counterstone of theoretical computer science. In search of simplest models to capture the finite state machines, MC Culloch and Pitts were among the first researchers to introduce a concept similar to finite automaton in 1943 (1)

Automata theory has become a basis in theoretical computer science because of its various applications(2). Danish Ather and others develop an efficient algorithm to design DFA that accept strings over input symbol a, b having at most $x$ number of a & y number of b (3). We are motivated by this to develop an Algorithm to design DFA over three input symbols a, b, c that accept strings having atmost $x$ number of a, y number of b & z number of c. We use the following definition of DFA & acceptance of strings in this paper.

## Definition

Finite Automata (M) is defined as a five tuple ( Q, Σ, $\delta$, $q_0$, F) Where

Q - a finite, non empty set of states

Σ - a finite, non empty set of inputs

$\delta$ - Q X Σ → Q is the state – transition function

$q_0 \in Q$ is the initial state

$F \subseteq Q$ is the set of final states

## Definition

A string $w \in \Sigma^*$ is said to be accepted by a DFA M if $\delta(q_0, w) \in F$ **2.Algorithm :**

By Applying this Algorithm we can construct Deterministic Finite Automate that accept strings over input symbol a, b, c having atmost $x$ number of a, y number of b & Z number of c.

Algorithm to draw Transition Graph Deterministic Finite Automata

M - ( Q, Σ, $\delta$, $q_{0,}$ F) where

Q = $\{q_{ojk}\}$ U $\{q_{i0k}\}$ U $\{q_{ij0}\}$ where i = 0 to x, j = 0 to y, k = 0 to z

Σ = $\{a, b, c\}$

$\delta$ - Q X Σ → Q (Represented by Transition Graph)

$q_0$ = q $i_{jk \text{ Where}}$ $i_{= j = k = o \text{ ie}}$ $q_{000,}$

F = Q

Let Q be the set of states in Deterministic Finite Automata such that

Q - $\{q_{000}, q_{011}, q_{021}, q_{ojk}, q_{101}, q_{201}, \ldots\ldots q_{i0k}, q_{110}, \ldots . qi_{jo}\}$

Where i - 0 to x

j - 0 to $y$

k - 0 to $z$

Input symbol Σ = $\{a, b, c\}$

$q_{000}$ is the initial state

Design a directed transition graph having

(x+1) (y+1) + (y+1) (z+1) + (x+1) (z+1) – (x+y+z+2)

states and mark all states as final states label each node as $q_{000}$, $q_{110}$, $q_{011}$, $q_{021}$, ……. $q_{ojk}$, $q_{010}$, $q_{110}$, …… q $i_{jo}$, $q_{001}$, $q_{101}$, $q_{201}$/…… $qi_{ok}$, Where i = 0 to x, j = 0 to y, k = 0 to z

For i = 0 to $x$

do

For j - O to $y$

do

For k - O to $z$

do

if $_{I = j = k = 0}$ then $q_{000,} \in Q_0$ (initial state)

else there exist a edge E such that

$\delta$ (q $i_{jo, a}$) = q $_{(}i_{+1) jo}$ ; $\delta$ ($q_{(}i_{ok, a}$ ) = q $_{(}i_{+1) ok}$ ;

$\delta$ (q $_{ojk, b)}$ = $q_{o (j+1) k}$ ; $\delta$ ($q_{(}i_{jo, b}$ ) = $qi_{(j+1) o}$ ;

$\delta$ ($q_{ojk, c)}$ = $q_{oj (K+1)}$ ; $\delta$ ($q_{(}i_{ok, c}$ ) = $qi_{o (k+1)}$

done inner loop

done outer loop

DFA “M’’ will accept strings over input symbol a, b, c having atmost $x$ number of a, y number of b & number of c.

**Implementation**

Design a DFA that accept strings over Input symbol a, b, c having atmost three a’s; three b’s of three c’s

Let the resultant DFA is M - ( Q, Σ, $\delta$, $q_{0,}$ F) where

$\{q_{ojk}\}$ U $\{q_{i0k}\}$ U $\{q_{ij0}\}$ where i = 0 to 3, j = 0 to 3, k = 0 to 3

$\Sigma$ = $\{a, b, c\}$

$Q_0$ - $q_{000}$,

F = Q

$\delta$ - $Q \times \Sigma \rightarrow Q$ is given by the transition graph in Figure1

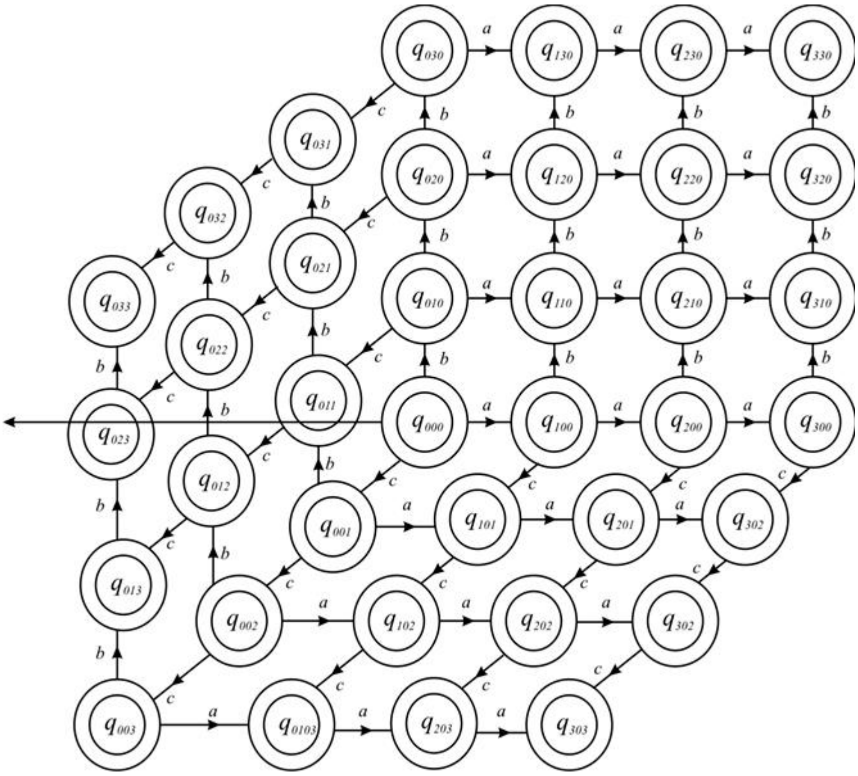


**Figure 1 TG of DFA that accept strings over input symbol a, b, c having atmost 3, a's, 3 b 's, 3 c 's**

**Design a DFA that accept strings over input symbol a,b,c having atmost only one a, two b & two C**

Let the resultant DFA is M = ( Q, $\Sigma$, $\delta$, $q_0$, F) where

Q = $\{q_{000}, q_{100}, q_{010}, q_{110}, q_{020}, q_{120}, q_{001}, q_{101}, q_{002}, q_{102}, q_{022}, q_{021}, q_{012}, q_{011}\}$

$\Sigma$ = $\{a, b, c\}$

$\delta$ - $Q \times \Sigma \rightarrow Q$ is given by the transition graph in Fig2

$Q_0$ - $q_{000}$,

F = Q

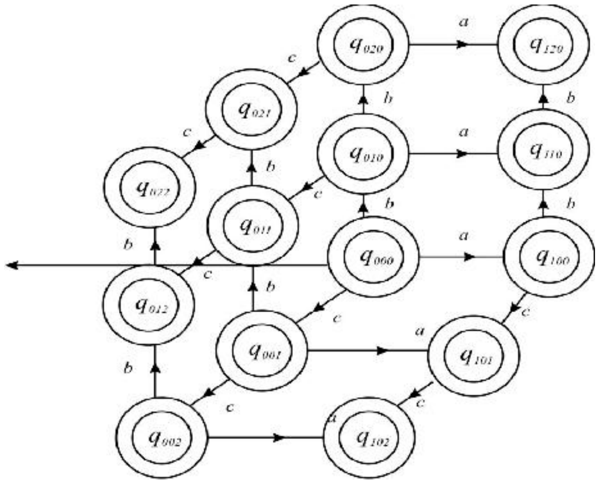


**Figure 2 TG of DFA that accept strings over Input symbol a, b, c having atmost one a, two b's & two c's**

**Design a DFA that accept strings over input symbol a,b,c having atmost two a's two b's & two c's**

Let the resultant DFA is

M = ( Q, Σ, $\delta$, $q_0$, F)

Q = $\{q_{ojk}\}$ U $\{q_{i0k}\}$ U $\{q_{ij0}\}$ where i = 0 to 2, j = 0 to 2, k = 0 to 2

Σ = $\{a, b, c\}$

$Q_0$ - $q_{000}$,

$\delta$ - Q X Σ → Q is given by the transition graph in Fig2

F = Q

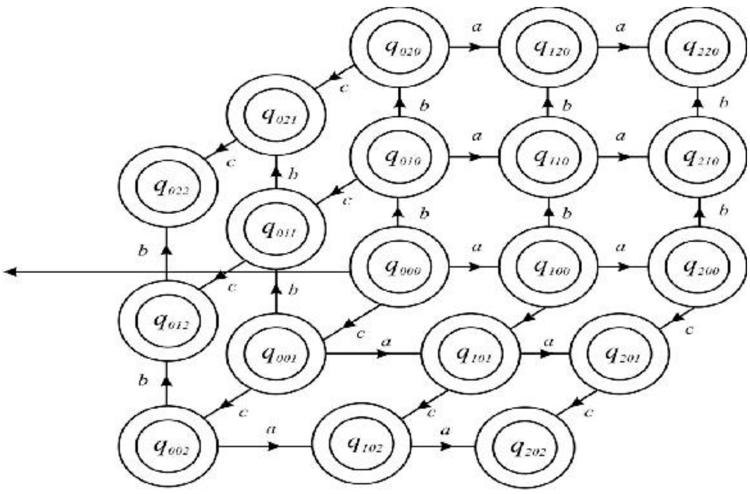


**Figure 3 TG of DFA that accept strings over input symbol a, b, c having atmost two a's, two b's & two c's**

**Analysis of Acceptance of Strings**

4.1 In the DFA given in Fig1, the following strings are checked whether they are accepted or rejected and the result is given below.

| Strings | Result |
|---|---|
| a a b | Accepted |
| a a b c b b b | Rejected |
| a b c a b c a a | Rejected |
| b c a b a c a | Accepted |
| a a b c c c | Accepted |
| c a a b a a b | Rejected |

4.2 In the DFA given in Fig 2 the following strings are verified whether they are accepted or rejected and the result is given below.

| Strings | Result |
|---|---|
| a a b c | Rejected |
| a a a c | Rejected |
| a b b c c | Accepted |
| b b c c | Accepted |
| a b b c c | Accepted |
| a b c | Accepted |
| a b a b a b | Rejected |

4.3 In the DFA given in Fig 3 the following strings are verified whether they are accepted or rejected and the result is given below.

| Strings | Result |
|---|---|
| a b a c c b | Accepted |
| a a b b c c | Accepted |
| c a b b b c | Rejected |
| a b c c | Accepted |
| a b c b c | Accepted |
| b a c | Accepted |

**Conclusion**

Thus in this paper, we give an efficient algorithm to design a DFA that accept strings over input symbol a, b, c having atmost $x$ number of a, y number of b & z number of c. In the next research, we will give an algorithm over more than 3 input symbols.

## References


1. Lawson, Mark V (2004) Finite automata, Chapman and Hall / CRC. ISBN 1 – 58488 – 255 -7 – zbi 1086 – 68074
2. J.A Anderson, Automata theory with modern applications, Cambridge University Press – ISBN : 9780511223013, 2006.
3. Danish Ather, Raghuraj Singh, Vinodani Katiyar "An Efficient Algorithm to Design DFA that Accept strings over Input symbol a, b having atmost $x$ number of a, y number of b Journal of Nature Inspired Computing (JNIC) Vol1, No2, 2013.
4. Hopcroft, John Motwani Rajeev, Ulman, Jeffrey D.Introduction to Automata Theory, languages and computation (2nd edition) Adolision Wesely. ISBN No – 201 – 44124-1 Retrieved 19 November 2012
5. Rabin M.O. Scott.D (1959)
   "Finite Automata' and their decision problems, IBMJ Res.Develop 114 -125.