LAPORAN TUGAS BESAR I IF2220

TEORI BAHASA FORMAL DAN AUTOMATA APLIKASI PERMAINAN "TIC-TAC-TOE" DENGAN FINITE AUTOMATA



DIBUAT OLEH: ADITYA PUTRA SANTOSA / 13517013

PROGRAM STUDI TEKNIK INFORMATIKA SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA INSTITUT TEKNOLOGI BANDUNG BANDUNG

2018

A. Deskripsi Persoalan

Tic-tac-toe adalah sebuah permainan untuk dua orang yang secara bergiliran saling membuat huruf X dan O didalam sebuah kotak 3 x 3. Pemenang dari permainan ini adalah pemain pertama yang berhasil membuat tiga tanda "X" atau tanda "O" berurutan di dalam suatu permainan.

Buat sebuah aplikasi permainan tic-tac-toe sederhana, dimana permainan ini akan dimainkan oleh komputer dan player. Program harus bisa memastikan bahwa **komputer tidak mungkin kalah didalam permainan**. Aplikasi akan membuka file yang berisi informasi mengenai daftar state, daftar simbol, state awal, state akhir, dan transition function. Informasi dari file tersebut akan digunakan untuk mengecek masukan dari pengguna. Perlu diperhatikan bahwa program diwajibkan untuk membaca konfigurasi dari file eksternal, dan logika state machine tidak boleh di-hardcode ke program secara langsung. Batasan masalah : Pada langkah pertama permainan, player/CPU dipastikan meletakkan tanda "X" atau "O" di bagian tengah papan. Setelah itu pastikan bahwa **CPU tidak pernah kalah** di dalam permainan ini. Petak pada sudut kiri atas adalah petak 1, 2, dan 3, kemudian baris selanjutnya dilanjutkan dengan petak 4, 5, 6, dan baris terakhir adalah petak 7, 8, dan 9.

B. DFA

I. Komputer pertama

1. State

State untuk DFA Komputer Pertama sebanyak 119 state dengan rincian sebagai berikut :

\mathcal{C}	
(001)5	(020)5,1,2,8,4,6,3,7,9
(002)5,1,2	(021)5,1,2,8,4,6,3,9,7
(003)5,2,1	(022)5,2,1,3,9
(004)5,3,1	(023)5,2,1,4,9
(005)5,4,1	(024)5,2,1,6,9
(006)5,6,1	(025)5,2,1,7,9
(007)5,7,1	(026)5,2,1,8,9
(008)5,8,1	(027)5,2,1,9,4
(009)5,9,1	(028)5,2,1,9,4,3,6
(010)5,1,2,3,8	(029)5,2,1,9,4,6,7
(011)5,1,2,4,8	(030)5,2,1,9,4,7,6
(012)5,1,2,6,8	(031)5,2,1,9,4,8,6
(013)5,1,2,7,8	(032)5,3,1,2,9
(014)5,1,2,8,4	(033)5,3,1,4,9
(015)5,1,2,9,8	(034)5,3,1,6,9
(016)5,1,2,8,4,3,6	(035)5,3,1,7,9
(017)5,1,2,8,4,6,3	(036)5,3,1,8,9
(018)5,1,2,8,4,7,6	(037)5,3,1,9,6
(019)5,1,2,8,4,9,6	(038)5,3,1,9,6,2,4

(039)5,3,1,9,6,4,2	(080)5,8,1,7,9
(040)5,3,1,9,6,7,4	(081)5,8,1,9,7
(041)5,3,1,9,6,8,4	(082)5,8,1,9,7,2,3
(042)5,3,1,9,6,4,2,7,8	(083)5,8,1,9,7,3,4
(043)5,3,1,9,6,4,2,8,7	(084)5,8,1,9,7,4,3
(044)5,4,1,2,9	(085)5,8,1,9,7,6,3
(045)5,4,1,3,9	(086)5,9,1,2,4
(046)5,4,1,6,9	(087)5,9,1,3,6
(047)5,4,1,7,9	(088)5,9,1,4,2
(048)5,4,1,8,9	(089)5,9,1,6,3
(049)5,4,1,9,2	(090)5,9,1,7,8
(050)5,4,1,9,2,3,8	(091)5,9,1,8,7
(051)5,4,1,9,2,6,3	(092)5,9,1,2,4,3,6
(052)5,4,1,9,2,7,3	(093)5,9,1,2,4,6,7
(053)5,4,1,9,2,8,3	(094)5,9,1,2,4,7,6
(054)5,6,1,2,9	(095)5,9,1,2,4,8,6
(055)5,6,1,3,9	(096)5,9,1,3,6,2,4
(056)5,6,1,4,9	(097)5,9,1,3,6,4,2
(057)5,6,1,7,9	(098)5,9,1,3,6,7,4
(058)5,6,1,8,9	(099)5,9,1,3,6,8,4
(059)5,6,1,9,3	(100)5,9,1,3,6,4,2,7,8
(060)5,6,1,9,3,2,7	(101)5,9,1,3,6,4,2,8,7
(061)5,6,1,9,3,4,2	(102)5,9,1,4,2,3,8
(062)5,6,1,9,3,7,2	(103)5,9,1,4,2,6,3
(063)5,6,1,9,3,8,2	(104)5,9,1,4,2,7,3
(064)5,7,1,2,9	(105)5,9,1,4,2,8,3
(065)5,7,1,3,9	(106)5,9,1,6,3,2,7
(066)5,7,1,4,9	(107)5,9,1,6,3,4,2
(067)5,7,1,6,9	(108)5,9,1,6,3,7,2
(068)5,7,1,8,9	(109)5,9,1,6,3,8,2
(069)5,7,1,9,8	(110)5,9,1,7,8,2,3
(070)5,7,1,9,8,2,3	(111)5,9,1,7,8,3,2
(071)5,7,1,9,8,3,2	(112)5,9,1,7,8,4,2
(072)5,7,1,9,8,4,2	(113)5,9,1,7,8,6,2
(073)5,7,1,9,8,6,2	(114)5,9,1,7,8,2,3,4,6
(074)5,7,1,9,8,2,3,4,6	(115)5,9,1,7,8,2,3,6,4
(075)5,7,1,9,8,2,3,6,4	(116)5,9,1,8,7,2,3
(076)5,8,1,2,9	(117)5,9,1,8,7,3,4
(077)5,8,1,3,9	(118)5,9,1,8,7,4,3
(078)5,8,1,4,9	(119)5,9,1,8,7,6,3
(079)5,8,1,6,9	

Dimana 3 digit di awal adalah nomor state dan sisanya adalah nama statenya.

2. Simbol

Simbol yang valid bagi DFA ini adalah angka dari 1 s/d 9 yang menyatakan petak petak pada papan permainan Tic-Tac-Toe.

3. State Awal

State awal DFA ini adalah (001)5 yang menyatakan CPU menaruh "O" di petak nomor 5.

4. State Akhir

Semua state yang merupakan final state sebanyak	92 State dengan rincian:
(010)5,1,2,3,8	(050)5,4,1,9,2,3,8
(011)5,1,2,4,8	(051)5,4,1,9,2,6,3
(012)5,1,2,6,8	(052)5,4,1,9,2,7,3
(013)5,1,2,7,8	(053)5,4,1,9,2,8,3
(015)5,1,2,9,8	(054)5,6,1,2,9
(016)5,1,2,8,4,3,6	(055)5,6,1,3,9
(018)5,1,2,8,4,7,6	(056)5,6,1,4,9
(019)5,1,2,8,4,9,6	(057)5,6,1,7,9
(020)5,1,2,8,4,6,3,7,9	(058)5,6,1,8,9
(021)5,1,2,8,4,6,3,9,7	(060)5,6,1,9,3,2,7
(022)5,2,1,3,9	(061)5,6,1,9,3,4,2
(023)5,2,1,4,9	(062)5,6,1,9,3,7,2
(024)5,2,1,6,9	(063)5,6,1,9,3,8,2
(025)5,2,1,7,9	(064)5,7,1,2,9
(026)5,2,1,8,9	(065)5,7,1,3,9
(028)5,2,1,9,4,3,6	(066)5,7,1,4,9
(029)5,2,1,9,4,6,7	(067)5,7,1,6,9
(030)5,2,1,9,4,7,6	(068)5,7,1,8,9
(031)5,2,1,9,4,8,6	(071)5,7,1,9,8,3,2
(032)5,3,1,2,9	(072)5,7,1,9,8,4,2
(033)5,3,1,4,9	(073)5,7,1,9,8,6,2
(034)5,3,1,6,9	(074)5,7,1,9,8,2,3,4,6
(035)5,3,1,7,9	(075)5,7,1,9,8,2,3,6,4
(036)5,3,1,8,9	(076)5,8,1,2,9
(038)5,3,1,9,6,2,4	(077)5,8,1,3,9
(040)5,3,1,9,6,7,4	(078)5,8,1,4,9
(041)5,3,1,9,6,8,4	(079)5,8,1,6,9
(042)5,3,1,9,6,4,2,7,8	(080)5,8,1,7,9
(043)5,3,1,9,6,4,2,8,7	(082)5,8,1,9,7,2,3
(044)5,4,1,2,9	(083)5,8,1,9,7,3,4
(045)5,4,1,3,9	(084)5,8,1,9,7,4,3
(046)5,4,1,6,9	(085)5,8,1,9,7,6,3
(047)5,4,1,7,9	(092)5,9,1,2,4,3,6
(048)5,4,1,8,9	(093)5,9,1,2,4,6,7

(094)5,9,1,2,4,7,6	(107)5,9,1,6,3,4,2
(095)5,9,1,2,4,8,6	(108)5,9,1,6,3,7,2
(096)5,9,1,3,6,2,4	(109)5,9,1,6,3,8,2
(098)5,9,1,3,6,7,4	(111)5,9,1,7,8,3,2
(099)5,9,1,3,6,8,4	(112)5,9,1,7,8,4,2
(100)5,9,1,3,6,4,2,7,8	(113)5,9,1,7,8,6,2
(101)5,9,1,3,6,4,2,8,7	(114)5,9,1,7,8,2,3,4,6
(102)5,9,1,4,2,3,8	(115)5,9,1,7,8,2,3,6,4
(103)5,9,1,4,2,6,3	(116)5,9,1,8,7,2,3
(104)5,9,1,4,2,7,3	(117)5,9,1,8,7,3,4
(105)5,9,1,4,2,8,3	(118)5,9,1,8,7,4,3
(106)5,9,1,6,3,2,7	(119)5,9,1,8,7,6,3

5. Fungsi Transisi

Fungsi Transisi untuk DFA ini adalah :

Nama State	1	2	3	4	5	6	7	8	9
(001)5	(002)5,1,2	(003)5,2,1		(005)5,4,1		(006)5,6,1	(007)5,7,1	(008)5,8,1	(009)5,9,1
(002)5,1,2	-	-		(011)5,1,2,4,8	-	(012)5,1,2,6,8	(013)5,1,2,7,8	(014)5,1,2,8,4	(015)5,1,2,9,8
(003)5,2,1	-	-	(022)5,2,1,3,9	(023)5,2,1,4,9	-	(024)5,2,1,6,9	(025)5,2,1,7,9	(026)5,2,1,8,9	(027)5,2,1,9,4
(004)5,3,1	-	(032)5,3,1,2,9	-	(033)5,3,1,4,9	-	(034)5,3,1,6,9	(035)5,3,1,7,9	(036)5,3,1,8,9	(037)5,3,1,9,6
(005)5,4,1	-		(045)5,4,1,3,9	-	-	(046)5,4,1,6,9	(047)5,4,1,7,9	(048)5,4,1,8,9	(049)5,4,1,9,2
(006)5,6,1	-			(056)5,6,1,4,9	-	-	(057)5,6,1,7,9	(058)5,6,1,8,9	(059)5,6,1,9,3
(007)5,7,1	-	(064)5,7,1,2,9		(066)5,7,1,4,9	-	(067)5,7,1,6,9	-	(068)5,7,1,8,9	(069)5,7,1,9,8
(008)5,8,1	-	(076)5,8,1,2,9	(077)5,8,1,3,9	(078)5,8,1,4,9		(079)5,8,1,6,9	(080)5,8,1,7,9		(081)5,8,1,9,7
(009)5,9,1	-	(086)5,9,1,2,4	(087)5,9,1,3,6	(088)5,9,1,4,2	-	(089)5,9,1,6,3	(090)5,9,1,7,8	(091)5,9,1,8,7	-
(010)5,1,2,3,8	-	-	-	-	-	-	-	-	-
(011)5,1,2,4,8	-	-	-	-	-	-	-	-	-
(012)5,1,2,6,8	-		-		-	-		-	-
	-	-	-	-	-	-	-	-	-
	-	-	(016)5,1,2,8,4,3,6	-	-	(017)5,1,2,8,4,6,3	(018)5,1,2,8,4,7,6	-	(019)5,1,2,8,4,9,6
(015)5,1,2,9,8	-	-	-	-	-	-	-	-	-
(016)5,1,2,8,4,3,6	-	-	-		-	-		-	
	-		-		-	-	(020)5,1,2,8,4,6,3,7,9	-	(021)5,1,2,8,4,6,3,9,7
	-	-	-	-	-	-	-	-	-
(019)5,1,2,8,4,9,6	-	-	-	-	-	-	-	-	-
(020)5,1,2,8,4,6,3,7,9	-	-	-	-	-	-	-	-	-
(021)5,1,2,8,4,6,3,9,7	-	-	-		-	-		-	
	-	-	-	-	-	-		-	-
(023)5,2,1,4,9	-		-		-	-		-	-
(024)5,2,1,6,9	-	-	-	-	-	-	-	-	-
(025)5,2,1,7,9	-	-	-	-	-	-	-	-	-
(026)5,2,1,8,9	-		-		-	-		-	
	-	-	(028)5,2,1,9,4,3,6	-	-	(029)5,2,1,9,4,6,7	(030)5,2,1,9,4,7,6	(031)5,2,1,9,4,8,6	-
	-	-	-	-	-	-	-	-	-
(029)5,2,1,9,4,6,7	-	-	-	-	-	-	-	-	-
(030)5,2,1,9,4,7,6	-	-	-	-	-	-		-	-
(031)5,2,1,9,4,8,6	-		-		-	-		-	
	-		-		-	-		-	-
	-	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-	-
(036)5,3,1,8,9	-				-	-		-	-
	-	(038)5,3,1,9,6,2,4		(039)5,3,1,9,6,4,2	-	-	(040)5,3,1,9,6,7,4	(041)5,3,1,9,6,8,4	-
	-	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	(042)5,3,1,9,6,4,2,7,8	(043)5,3,1,9,6,4,2,8,7	-
(040)5,3,1,9,6,7,4	-		-	-	-	-	-	-	

[0415,3,1,9,6,4,2,8,7]	3
[043]5,3,1,9,6,4,2,8,7	
[043]5,3,1,9,6,4,2,8,7	
[044]5,4,1,2,9	
[045]6,41,3.9	
[049]6,41,6.9	
[047]5,4,17,9	-
[049]5,4,1,9,2	
[049]5,4,1,9,2 -	- - - - - - - -
[049]5,4,1,9,2 -	3
[050]5, 4, 19, 2, 3, 8	- - - -
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	- - - - -
[062]6,4,1,9,2,7,3	-
[053]5,4,1,9,2,8,3	- - -
[054]5,6,12,9	-
[055], 6, 13, 9	-
[055], 6, 13, 9	-
[056]5,6,1,4,9	
[057]5.6,17,9	_
[056]5,6,1,8,9	
[059)5,6,1,9,3 - (060)5,6,1,9,3,2,7 - (061)5,6,1,9,3,4,2 - (062)5,6,1,9,3,7,2 (063)5,6,1,9,3,8,	
(069)5,6,1,9,3,4,2 - (062)5,6,1,9,3,7,2 (063)5,6,1,9,3,8,3 - (062)5,6,1,9,3,7,2 (063)5,6,1,9,3,8,3 - (062)5,6,1,9,3,7,2 (063)5,6,1,9,3,8,3 - (063)5,6,1,9,3,8,3 - (063)5,6,1,9,3,4,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,6,1,9,1,2 - (063)5,1,9,1,2 - (063)5,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1	-
	-
(060)5,6,1,9,3,2,7	-
(061)5,6,1,9,3,4,2	_
(062)5,6,1,9,3,7,2	-
(063)6.13.3.2	
(064)5,7,1,2,9	-
[065)5,7,1,3,9	-
[066]5,7,1,4,9	-
(067)5,7,1,6,9	-
(068)5,7,1,8,9	-
[0695,7,1,9,8] - (070)5,7,1,9,8,2,3 (071)5,7,1,9,8,3,2 (072)5,7,1,9,8,4,2 - (073)5,7,1,9,8,6,2	
[070]5,7,1,9,8,2,3 - [075]5,7,1,9,8,2,3,4,6 - [075]5,7,1,9,8,2,3,6,4	-
[071)5,7,1,9,8,3,2 - - - - - - - -	-
[072]5,7,1,9,8,4,2	-
(073)5,7,1,9,8,6,2	-
074)5,7,1,9,8,2,3,4,6	-
075)5,7,1,9,8,2,3,6,4	
(076)5.8,1,2,9	
(0776) 8.13.9	
	-
[078)5.8,1,4,9	-
[079]5,8,1,6,9 - - - - - - -	-
(080)5,8,1,7,9	-
(081)5,8,1,9,7 - (082)5,8,1,9,7,2,3 (083)5,8,1,9,7,3,4 (084)5,8,1,9,7,4,3 - (085)5,8,1,9,7,6,3 -	-
(082)5,8,1,9,7,2,3	-
(083)5, 8, 1, 9, 7, 3, 4	_
(084)5,3,1,9,7,4,3	
(085),8,1,9,7,6,3	
(000)5,0,1,3,7,0,5	
(086)5,9,1,2,4 - (092)5,9,1,2,4,3,6 - (093)5,9,1,2,4,7,6 (094)5,9,1,2,4,7,6 (095)5,9,1,2,4,8,6	
(087)5,9,1,3,6 - (096)5,9,1,3,6,2,4 - (097)5,9,1,3,6,4,2 - (098)5,9,1,3,6,7,4 (099)5,9,1,3,6,8,	
(088)5,9,1,4,2 - (102)5,9,1,4,2,3,8 - (103)5,9,1,4,2,7,3 (104)5,9,1,4,2,7,3 (105)5,9,1,4,2,8,	š -
(089)5,9,1,6,3 - (108)5,9,1,6,3,2,7 - (107)5,9,1,6,3,4,2 (108)5,9,1,6,3,7,2 (109)5,9,1,6,3,8,	2 -
(090)5,9,1,7,8 - (110)5,9,1,7,8,2,3 (111)5,9,1,7,8,3,2 (112)5,9,1,7,8,4,2 - (113)5,9,1,7,8,6,2 -	
(091)5,9,1,8,7 - (116)5,9,1,8,7,2,3 (117)5,9,1,8,7,3,4 (118)5,9,1,8,7,4,3 - (119)5,9,1,8,7,6,3	
(092)5 9 1 2 4 3 6	
000000000000000000000000000000000000000	-
(093)5,9,1,2,4,6,7 - - - - -	
(093)5,9,1,2,4,6,7	
(093)5, 9, 1, 2, 4, 6, 7	-
(093)5,9,1,2,4,6,7	-
(093)5,9,1,2,4,6,7	
(993)5, 9, 1, 2, 4, 6, 7	- - - - 2,8,7
109316_9,1,2,4,6,7	- - - - 2,8,7 - -
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	2,8.7 -
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	- - 2,8,7 - - -
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	2.8.7
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	2.6.7
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	2.8.7
(093)5, 91, 24, 67	2.8.7
(093)5, 91, 24, 6.7	287
(993)6, 91, 24, 67	
(093)5, 9, 1, 2, 4, 6, 7	
(093)5, 91, 24, 67	
(093)5, 91, 24, 6.7	
(093)5, 91, 24, 67	
(093)5, 91, 24, 67	
(093)5, 91, 24, 67	
109315, 91, 12, 14, 6.7	
109316, 91, 24, 67	
1093 5, 9, 1, 2, 4, 6, 7	
109316, 91, 12, 14, 16, 7	
109315, 91, 24, 67	
109316, 91, 12, 14, 16, 7	
1093 5,9,1,2,4,6,7	
109315, 91, 24, 67	
109315, 91, 24, 67	

II. Player pertama

1. State

State untuk DFA Player Pertama sebanyak 117 state dengan rincian sebagai berikut :

(001)5,1	(010)5,1,2,8,4,6
(002)5,1,2,8	(011)5,1,2,8,6,4
(003)5,1,3,7	(012)5,1,2,8,7,3
(004)5,1,4,6	(013)5,1,2,8,9,3
(005)5,1,6,4	(014)5,1,2,8,3,7,4,9
(006)5,1,7,3	(015)5,1,2,8,3,7,6,4
(007)5,1,8,2	(016)5,1,2,8,3,7,9,4
(008)5,1,9,3	(017)5,1,2,8,4,6,3,7
(009)5,1,2,8,3,7	(018)5,1,2,8,4,6,7,3

(019)5,1,2,8,4,6,9,3	(062)5,1,4,6,3,7,8,2
(020)5,1,2,8,4,6,3,7,9	(063)5,1,4,6,3,7,9,2
(021)5,1,2,8,4,6,7,3,9	(064)5,1,4,6,3,7,2,8,9
	(065)5,1,4,6,3,7,8,2,9
(022)5,1,2,8,4,6,9,3,7	(, , , , , , , , , , , , , , , , , , ,
(023)5,1,2,8,6,4,3,7	(066)5,1,4,6,3,7,9,2,8
(024)5,1,2,8,6,4,7,3	(067)5,1,4,6,7,3,2,9
(025)5,1,2,8,6,4,9,7	(068)5,1,4,6,7,3,8,2
(026)5,1,2,8,6,4,7,3,9	(069)5,1,4,6,7,3,9,2
(027)5,1,2,8,7,3,4,6	(070)5,1,4,6,8,2,3,7
(028)5,1,2,8,7,3,6,4	(071)5,1,4,6,8,2,7,3
(029)5,1,2,8,7,3,9,4	(072)5,1,4,6,8,2,9,3
(030)5,1,2,8,7,3,4,6,9	(073)5,1,4,6,8,2,3,7,9
(031)5,1,2,8,7,3,6,4,9	(074)5,1,4,6,9,2,3,7
(032)5,1,2,8,7,3,9,4,6	(075)5,1,4,6,9,2,7,3
(033)5,1,2,8,9,3,4,6	(076)5,1,4,6,9,2,8,3
(034)5,1,2,8,9,3,6,4	(077)5,1,4,6,9,2,3,7,8
(035)5,1,2,8,9,3,7,4	(078)5,1,6,4,2,7
(036)5,1,2,8,9,3,4,6,7	(079)5,1,6,4,3,7
(037)5,1,2,8,9,3,6,4,7	(080)5,1,6,4,7,3
(038)5,1,2,8,9,3,7,4,6	(081)5,1,6,4,8,7
(039)5,1,3,7,2,4	(082)5,1,6,4,9,7
(040)5,1,3,7,4,6	(083)5,1,6,4,7,3,2,8
(041)5,1,3,7,6,4	(084)5,1,6,4,7,3,8,2
(042)5,1,3,7,8,4	(085)5,1,6,4,7,3,9,2
(043)5,1,3,7,9,4	(086)5,1,6,4,7,3,2,8,9
(044)5,1,3,7,4,6,2,8	(087)5,1,7,3,2,8
(045)5,1,3,7,4,6,8,2	(088)5,1,7,3,4,2
(046)5,1,3,7,4,6,9,2	(089)5,1,7,3,6,2
(047)5,1,3,7,4,6,2,8,9	(090)5,1,7,3,8,2
(048)5,1,3,7,4,6,8,2,9	(091)5,1,7,3,9,2
(049)5,1,3,7,4,6,9,2,8	(092)5,1,7,3,2,8,4,6
(050)5,1,4,6,2,8	(093)5,1,7,3,2,8,6,4
(051)5,1,4,6,3,7	(094)5,1,7,3,2,8,9,4
(052)5,1,4,6,7,3	(095)5,1,7,3,2,8,4,6,9
(053)5,1,4,6,8,2	(096)5,1,7,3,2,8,6,4,9
(054)5,1,4,6,9,2	(097)5,1,7,3,2,8,9,4,6
(055)5,1,4,6,2,8,3,7	(098)5,1,8,2,3,7
(056)5,1,4,6,2,8,7,3	(099)5,1,8,2,4,3
(057)5,1,4,6,2,8,9,3	(100)5,1,8,2,6,3
(058)5,1,4,6,2,8,3,7,9	(101)5,1,8,2,7,3
(059)5,1,4,6,2,8,7,3,9	(102)5,1,8,2,9,3
(060)5,1,4,6,2,8,9,3,7	(103)5,1,8,2,3,7,4,6
(061)5,1,4,6,3,7,2,8	(104)5,1,8,2,3,7,6,4

(105)5,1,8,2,3,7,9,4	(112)5,1,9,3,2,8,4,6
(106)5,1,8,2,3,7,4,6,9	(113)5,1,9,3,2,8,6,4
(107)5,1,9,3,2,8	(114)5,1,9,3,2,8,7,4
(108)5,1,9,3,4,2	(115)5,1,9,3,2,8,4,6,7
(109)5,1,9,3,6,2	(116)5,1,9,3,2,8,6,4,7
(110)5,1,9,3,7,2	(117)5,1,9,3,2,8,7,4,6
(111)5,1,9,3,8,2	

2. Simbol

Simbol yang valid bagi DFA ini adalah angka dari 1 s/d 9 yang menyatakan petak petak pada papan permainan Tic-Tac-Toe.

3. State Awal

State awal DFA ini adalah (001)5,1 yang menyatakan Player menaruh "X" di petak nomor 1 lalu CPU menaruh "O" di petak nomor 5.

4. State Akhir

Semua state yang merupakan final state sebanyak 65 State dengan rincian :

Semua state yang merupakan imai state sebai	nyak 05 State dengan mician.
(014)5,1,2,8,3,7,4,9	(066)5,1,4,6,3,7,9,2,8
(015)5,1,2,8,3,7,6,4	(067)5,1,4,6,7,3,2,9
(016)5,1,2,8,3,7,9,4	(068)5,1,4,6,7,3,8,2
(020)5,1,2,8,4,6,3,7,9	(069)5,1,4,6,7,3,9,2
(021)5,1,2,8,4,6,7,3,9	(071)5,1,4,6,8,2,7,3
(022)5,1,2,8,4,6,9,3,7	(072)5,1,4,6,8,2,9,3
(023)5,1,2,8,6,4,3,7	(073)5,1,4,6,8,2,3,7,9
(025)5,1,2,8,6,4,9,7	(075)5,1,4,6,9,2,7,3
(026)5,1,2,8,6,4,7,3,9	(076)5,1,4,6,9,2,8,3
(030)5,1,2,8,7,3,4,6,9	(077)5,1,4,6,9,2,3,7,8
(031)5,1,2,8,7,3,6,4,9	(078)5,1,6,4,2,7
(032)5,1,2,8,7,3,9,4,6	(079)5,1,6,4,3,7
(036)5,1,2,8,9,3,4,6,7	(081)5,1,6,4,8,7
(037)5,1,2,8,9,3,6,4,7	(082)5,1,6,4,9,7
(038)5,1,2,8,9,3,7,4,6	(084)5,1,6,4,7,3,8,2
(039)5,1,3,7,2,4	(085)5,1,6,4,7,3,9,2
(041)5,1,3,7,6,4	(086)5,1,6,4,7,3,2,8,9
(042)5,1,3,7,8,4	(088)5,1,7,3,4,2
(043)5,1,3,7,9,4	(089)5,1,7,3,6,2
(047)5,1,3,7,4,6,2,8,9	(090)5,1,7,3,8,2
(048)5,1,3,7,4,6,8,2,9	(091)5,1,7,3,9,2
(049)5,1,3,7,4,6,9,2,8	(095)5,1,7,3,2,8,4,6,9
(058)5,1,4,6,2,8,3,7,9	(096)5,1,7,3,2,8,6,4,9
(059)5,1,4,6,2,8,7,3,9	(097)5,1,7,3,2,8,9,4,6
(060)5,1,4,6,2,8,9,3,7	(099)5,1,8,2,4,3
(064)5,1,4,6,3,7,2,8,9	(100)5,1,8,2,6,3
(065)5,1,4,6,3,7,8,2,9	(101)5,1,8,2,7,3

(102)5,1,8,2,9,3 (104)5,1,8,2,3,7,6,4 (105)5,1,8,2,3,7,9,4 (106)5,1,8,2,3,7,4,6,9 (108)5,1,9,3,4,2 (109)5,1,9,3,6,2 (110)5,1,9,3,7,2 (111)5,1,9,3,8,2 (115)5,1,9,3,2,8,4,6,7 (116)5,1,9,3,2,8,6,4,7 (117)5,1,9,3,2,8,7,4,6

5. Fungsi Transisi

Fungsi Transisi untuk DFA ini adalah:

)5,1	1	(002)5,1,2,8	(003)5,1,3,7	4 (004)5,1,4,6	5	(005)5,1,6,4	7 (006)5,1,7,3	8 (007)5,1,8,2	(008)5,1,9,3
)5,1,2,8	-	(002)5, 1,2,6	(003)5,1,3,7	(010)5,1,2,8,4,6	-	(011)5,1,2,8,6,4	(012)5,1,7,3	(007)5, 1, 6, 2	(013)5,1,2,8,9,3
)5,1,2,0		(039)5,1,3,7,2,4	(003)3, 1,2,0,3,1	(040)5,1,3,7,4,6		(041)5,1,3,7,6,4	(012)5,1,2,0,7,5	(042)5,1,3,7,8,4	(043)5,1,3,7,9,4
)5,1,4,6		(050)5,1,4,6,2,8	(051)5,1,4,6,3,7	(040)5, 1,3,7,4,0	•	(041)0,1,3,7,0,4	(052)5,1,4,6,7,3	(053)5,1,4,6,8,2	(054)5,1,4,6,9,2
)5,1,6,4	-	(070)5 1 6 4 2 7	(079)5,1,6,4,3,7		-	•	(080)5,1,6,4,7,3	(081)5,1,6,4,8,7	(082)5,1,6,4,9,7
)5,1,7,3	-	(070)5,1,0,4,2,7	(0/3)3, 1,0,4,3,7	(000)E 4 7 2 4 2	-	/000\E 4 7 2 C 2	(000)5, 1, 6, 4, 7, 3		
15,1,8,2	•	(087)5,1,7,3,2,8	(098)5,1,8,2,3,7	(088)5,1,7,3,4,2		(089)5,1,7,3,6,2	(404)(-4.0.2.7.2	(090)5,1,7,3,8,2	(091)5,1,7,3,9,2
	•	(107)5,1,9,3,2,8	(090)5, 1,0,2,3,7	(099)5,1,8,2,4,3 (108)5,1,9,3,4,2		(100)5,1,8,2,6,3 (109)5,1,9,3,6,2	(101)5,1,8,2,7,3 (110)5,1,9,3,7,2	(444)5 4 0 2 0 2	(102)5,1,8,2,9,3
)5,1,9,3)5,1,2,8,3,7	•	(107)5, 1,5,3,2,6	•	(100)5, 1,5,5,4,2	-	(103)5,1,3,5,0,2	(110)5, 1,5,5,7,2	(111)5,1,9,3,8,2	(046)6 4 2 9 2 7 (
	-	-	(047) 5 4 0 0 4 6 0 7	(014)5,1,2,8,3,7,4,9	•	(015)5,1,2,8,3,7,6,4	(040)5 4 0 0 4 6 7 3	+	(016)5,1,2,8,3,7,
5,1,2,8,4,6		-	(017)5,1,2,8,4,6,3,7				(018)5,1,2,8,4,6,7,3	+	(019)5,1,2,8,4,6,
5,1,2,8,6,4		-	(023)5,1,2,8,6,4,3,7	(007)5 4 0 0 7 0 4 6	•	(000)5 4 0 0 7 0 6 4	(024)5,1,2,8,6,4,7,3	<u>-</u>	(025)5,1,2,8,6,4,
5,1,2,8,7,3	-	-	-	(027)5,1,2,8,7,3,4,6	-	(028)5,1,2,8,7,3,6,4		+	(029)5,1,2,8,7,3,
5,1,2,8,9,3	-	-	-	(033)5,1,2,8,9,3,4,6	-	(034)5,1,2,8,9,3,6,4	(035)5,1,2,8,9,3,7,4	-	-
5,1,2,8,3,7,4,9	-	-			-		-	-	-
5,1,2,8,3,7,6,4		-	-	-	-	-	-	-	-
5,1,2,8,3,7,9,4	-	-	-		-	-	-	-	-
5,1,2,8,4,6,3,7	-	-	-	-	-	-	-	-	(020)5,1,2,8,4,6,
5,1,2,8,4,6,7,3	-	-	-	-	-	-	-	-	(021)5,1,2,8,4,6,
5,1,2,8,4,6,9,3	-	-			-		(022)5,1,2,8,4,6,9,3,7	4-	-
5,1,2,8,4,6,3,7,9		-	-	-	-	-	-	-	-
5,1,2,8,4,6,7,3,9	-	-	-	-	-	-	-	-	-
5,1,2,8,4,6,9,3,7	-	-	-	-	-	-	-	-	-
5,1,2,8,6,4,3,7									-
5,1,2,8,6,4,7,3									(026)5, 1, 2, 8, 6, 4,
5,1,2,8,6,4,9,7	-		-		-	-	-	-	-
5,1,2,8,6,4,7,3,9	_	-	_		_	_	_		-
5,1,2,8,7,3,4,6	_	-	_	-	_	_	_	-	(030)5,1,2,8,7,3,
5,1,2,8,7,3,6,4								-	(031)5,1,2,8,7,3,
5,1,2,8,7,3,9,4	-					(032)6 1 2 8 7 3 0 4 6		-	(001)0, 1,2,0,7,3,
	-	-	-	-	-	(032)5,1,2,8,7,3,9,4,6		-	-
5,1,2,8,7,3,4,6,9	•	-	-	•	-	-	-	-	-
5,1,2,8,7,3,6,4,9	-	-	-	-	-	-	-	-	-
5,1,2,8,7,3,9,4,6	•	-	-	-	-	-		-	-
5,1,2,8,9,3,4,6		-					(036)5,1,2,8,9,3,4,6,7	-	-
5,1,2,8,9,3,6,4	•	-	-	-	-	-	(037)5,1,2,8,9,3,6,4,7	4-	-
5,1,2,8,9,3,7,4	-	-	-	-	-	(038)5,1,2,8,9,3,7,4,6	-	-	-
5,1,2,8,9,3,4,6,7	-	-		-	-	-	-	-	-
5,1,2,8,9,3,6,4,7	-	-	-	-	-	-	-	-	-
5,1,2,8,9,3,7,4,6	-	-	-	-	-	-	-	-	-
5,1,3,7,2,4	-	-	-	-	-	-	-	-	-
5,1,3,7,4,6	-	(044)5,1,3,7,4,6,2,8	-	-	-	-	-	(045)5,1,3,7,4,6,8,2	(046)5,1,3,7,4,6,
5,1,3,7,6,4		-			-	-	-	1-	-
5,1,3,7,8,4		-		-	-	-	-	-	-
5,1,3,7,9,4		-			-	-	-	-	-
5,1,3,7,4,6,2,8		-	-	-	-	-	-	-	(047)5,1,3,7,4,6,2
5,1,3,7,4,6,8,2		-	-	-	-	-	_	-	(048)5,1,3,7,4,6,8
5,1,3,7,4,6,9,2		-	-		_	_	-	(049)5,1,3,7,4,6,9,2,8	-
5,1,3,7,4,6,2,8,9		-	-			-	-	-	
5,1,3,7,4,6,8,2,9		_	_	_	-	_	_	_	
5,1,3,7,4,6,9,2,8		-	_	_	-	_	-	_	
5,1,4,6,2,8			(055)5,1,4,6,2,8,3,7				(056)5,1,4,6,2,8,7,3		(057)5,1,4,6,2,8,
5,1,4,6,3,7		(061)5,1,4,6,3,7,2,8	(033)3, 1,4,0,2,0,3,7	-		_	(030)3,1,4,0,2,0,7,3	(062)5,1,4,6,3,7,8,2	(063)5,1,4,6,3,7,
5,1,4,6,7,3	-	(067)5,1,4,6,7,3,2,9	-	-	_			(068)5,1,4,6,7,3,8,2	(069)5,1,4,6,7,3,
5,1,4,0,7,5	•	(061)5, 1,4,6,1,3,2,5	(070)E 4 4 C 9 2 2 7		-	-	(074)6 4 4 6 9 2 7 2	(000)5, 1,4,6,7,5,6,2	(003)5,1,4,0,7,3,
5,1,4,6,8,2	-	-	(070)5,1,4,6,8,2,3,7	-	•	•	(071)5,1,4,6,8,2,7,3	(07C)E 1 4 C 0 2 C 2	(072)5,1,4,6,8,2,
5,1,4,6,9,2		-	(074)5,1,4,6,9,2,3,7		-	-	(075)5,1,4,6,9,2,7,3	(076)5,1,4,6,9,2,8,3	(0E0)E 1 4 C 0 0
5,1,4,6,2,8,3,7		-			-	-	•	-	(058)5,1,4,6,2,8,3
5,1,4,6,2,8,7,3		-	-	-	-	-	(0C0)C 4 4 C 0 C C C	*	(059)5,1,4,6,2,8,
5,1,4,6,2,8,9,3	-	•	-	-	-	-	(060)5,1,4,6,2,8,9,3,7	<u> </u>	-
5,1,4,6,2,8,3,7,9	-	-	-	-	-	-	-	<u>-</u>	-
5,1,4,6,2,8,7,3,9	-	-	-	-	-	-	-	-	-
5,1,4,6,2,8,9,3,7	-	-	-	-	-	-	-	-	-
5,1,4,6,3,7,2,8	-	-	-	-	-	-	-	-	(064)5,1,4,6,3,7,2
5,1,4,6,3,7,8,2			-		-	-	-	-	(065)5,1,4,6,3,7,8
5,1,4,6,3,7,9,2	-	-	-	-	-	-	-	(066)5,1,4,6,3,7,9,2,8	-
	-	-	-	-	-	-	-	-	-
5,1,4,6,3,7,2,8,9	-		-	-	-	-	-	-	-
5,1,4,6,3,7,2,8,9 5,1,4,6,3,7,8,2,9					-	-	-	-	-
5,1,4,6,3,7,2,8,9 5,1,4,6,3,7,8,2,9 5,1,4,6,3,7,9,2,8					-	-	-	-	-
5,1,4,6,3,7,2,8,9 5,1,4,6,3,7,8,2,9 5,1,4,6,3,7,9,2,8	-				-	-	_	_	
5,1,4,6,3,7,2,8,9 5,1,4,6,3,7,8,2,9 5,1,4,6,3,7,9,2,8 5,1,4,6,7,3,2,9 5,1,4,6,7,3,8,2	- - -	-	-	-					
5,1,4,6,3,7,2,8,9 5,1,4,6,3,7,8,2,9 5,1,4,6,3,7,9,2,8 5,1,4,6,7,3,2,9 5,1,4,6,7,3,8,2	- - -	-	-	-	-	-	-	-	-
5,1,4,6,3,7,2,8,9 5,1,4,6,3,7,8,2,9 5,1,4,6,3,7,9,2,8 6,1,4,6,7,3,2,9 5,1,4,6,7,3,8,2 5,1,4,6,7,3,9,2	- - -	-	-	-	-	-	-	-	(073)5 1 4 6.8 2
5,1,4,6,3,7,2,8,9 5,1,4,6,3,7,9,2,8 6,1,4,6,7,3,2,9 6,1,4,6,7,3,2,9 6,1,4,6,7,3,9,2 5,1,4,6,7,3,9,2 5,1,4,6,8,2,3,7	- - - -	-	-	- - -	-	-	-	-	(073)5,1,4,6,8,2,3
5,1,4,6,3,7,2,8,9 5,1,4,6,3,7,9,2,8 5,1,4,6,7,3,2,9 5,1,4,6,7,3,2,9 5,1,4,6,7,3,8,2 5,1,4,6,7,3,9,2 5,1,4,6,8,2,3,7 5,1,4,6,8,2,7,3	-	-	-	- - - -	-	-	-		(073)5,1,4,6,8,2,:
5.1,4,6,3,7,2,8,9 5.1,4,6,3,7,8,2,9 5.1,4,6,7,3,2,9 5.1,4,6,7,3,2,9 5.1,4,6,7,3,9,2 5.1,4,6,7,3,9,2 5.1,4,6,8,2,7,3 5,1,4,6,8,2,7,3 5,1,4,6,8,2,9,3	- - - - -	-	-	-	-	-	-	-	(073)5,1,4,6,8,2,3
5.1,4,6,3,7,2,8,9 5.1,4,6,3,7,8,2,9 6.1,4,6,7,3,2,9 6.1,4,6,7,3,8,2 6.1,4,6,7,3,8,2 6.1,4,6,7,3,9,2 5.1,4,6,8,2,3,7 6.1,4,6,8,2,7,3 6.1,4,6,8,2,9,3 6.1,4,6,8,2,3,7,9	- - - - -	-	-	-	-	-	-		-
5.1.4.6.3.7.2.8.9 5.1.4.6.3.7.9.2.8 5.1.4.6.3.7.9.2.8 5.1.4.6.7.3.2.9 5.1.4.6.7.3.8.2 5.1.4.6.7.3.9.2 5.1.4.6.8.2.3.7 5.1.4.6.8.2.3.7 5.1.4.6.8.2.3.7 5.1.4.6.8.2.3.7.9 5.1.4.6.8.2.3.7.9	- - - - - -	-	-	-	-	- - - -	-	- - - - - (077)5,1,4,6,9,2,3,7,8	-
5.14.63.7.2.8.9 5.14.63.7.9.2.8 5.14.63.7.9.2.8 5.14.67.3.2.9 5.14.67.3.9.2 5.14.67.3.9.2 5.14.6.8.2.3.7 5.14.6.8.2.7.3 5.14.6.8.2.7.3 5.14.6.8.2.3.7 5.14.6.8.2.3.7 5.14.6.8.2.3.7.9 5.14.6.9.2.3.7 5.14.6.9.2.3.7	-	-		-	-	-	-	- - - - - - (077)5,1,4,6,9,2,3,7,8	-
5.1.4.6.3.7.2.8.9 5.1.4.6.3.7.9.2.8 5.1.4.6.7.3.2.9 5.1.4.6.7.3.8.2 5.1.4.6.7.3.9.2 5.1.4.6.7.3.9.2 5.1.4.6.8.2.3.7 5.1.4.6.8.2.7.3 5.1.4.6.8.2.3.7 5.1.4.6.9.2.7.3 5.1.4.6.9.2.3.7 5.1.4.6.9.2.3.7		-				- - - - -	-	- - - - - (077)5,1,4,6,9,2,3,7,8	-
5.1.4.6.3.7.2.8.9 5.1.4.6.3.7.8.2.9 5.1.4.6.7.3.2.9 5.1.4.6.7.3.2.9 5.1.4.6.7.3.9.2 5.1.4.6.8.2.3.7 5.1.4.6.8.2.3.7 5.1.4.6.8.2.3.7,9 5.1.4.6.8.2.3.7,9 5.1.4.6.8.2.3.7,9 5.1.4.6.9.2.3.7	-	-					-	(077)5,1,4,6,9,2,3,7,8	-
5.1.4.6.3.7.2.8.9 5.1.4.6.3.7.9.2.8 5.1.4.6.7.3.2.9 5.1.4.6.7.3.8.2 5.1.4.6.7.3.9.2 5.1.4.6.7.3.9.2 5.1.4.6.8.2.3.7 5.1.4.6.8.2.3.7 5.1.4.6.8.2.3.7,9 5.1.4.6.9.2.3.7,5 5.1.4.6.9.2.3.7,5 5.1.4.6.9.2.3.7,5 5.1.4.6.9.2.3.7,5 5.1.4.6.9.2.3.7,5 5.1.4.6.9.2.3.7,5		-	-			-	-	- - - - (077)5,1,4,6,9,2,3,7,8 - -	-
5.1.4.6.3.7.2.8.9 5.1.4.6.3.7.8.2.9 5.1.4.6.7.3.2.9 5.1.4.6.7.3.2.9 5.1.4.6.7.3.9.2 5.1.4.6.8.2.3.7 5.1.4.6.8.2.3.7 5.1.4.6.8.2.3.7,9 5.1.4.6.8.2.3.7,9 5.1.4.6.8.2.3.7,9 5.1.4.6.9.2.3.7					-	- - - - - - - -	-	(077)5,1,4,6,9,2,3,7,8 	-

082)5,1,6,4,9,7	-	-	-	-	-	-		-	-
083)5,1,6,4,7,3,2,8	-	-	-	-	-	-		-	(086)5, 1, 6, 4, 7, 3, 2, 8, 9
084)5,1,6,4,7,3,8,2	-	-	-	-	-	-	-	-	-
085)5,1,6,4,7,3,9,2	-		-	-		-		-	-
086)5,1,6,4,7,3,2,8,9	-			-		-			-
087)5,1,7,3,2,8	-		-	(092)5,1,7,3,2,8,4,6		(093)5,1,7,3,2,8,6,4		-	(094)5,1,7,3,2,8,9,4
088)5,1,7,3,4,2	-	-	-	-	-	-		-	-
089)5,1,7,3,6,2	-	-	-	-	-	-		-	-
090)5,1,7,3,8,2	-	-	-	-	-	-		-	-
091)5,1,7,3,9,2	-		-	-		-		-	-
092)5,1,7,3,2,8,4,6	-			-		-			(095)5,1,7,3,2,8,4,6,9
093)5,1,7,3,2,8,6,4	-	-	-	-	-	-		-	(096)5,1,7,3,2,8,6,4,9
094)5,1,7,3,2,8,9,4	-			-	-	(097)5,1,7,3,2,8,9,4,6			-
095)5,1,7,3,2,8,4,6,9	-	-	-	-	-	-	-	-	-
096)5,1,7,3,2,8,6,4,9	-		-	-	-	-		-	-
097)5,1,7,3,2,8,9,4,6	-			-		-			-
098)5,1,8,2,3,7	-			(103)5,1,8,2,3,7,4,6		(104)5,1,8,2,3,7,6,4			(105)5,1,8,2,3,7,9,4
099)5,1,8,2,4,3	-	-	-	-	-	-		-	-
100)5,1,8,2,6,3	-			-	-	-			-
101)5,1,8,2,7,3	-	-	-	-	-	-	-	-	-
102)5,1,8,2,9,3	-		-	-		-		-	-
103)5,1,8,2,3,7,4,6	-			-		-			(106)5,1,8,2,3,7,4,6,9
104)5,1,8,2,3,7,6,4	-			-		-			-
105)5,1,8,2,3,7,9,4	-	-	-	-	-	-		-	-
106)5,1,8,2,3,7,4,6,9	-			-	-	-			-
107)5,1,9,3,2,8	-	-	-	(112)5,1,9,3,2,8,4,6	-	(113)5,1,9,3,2,8,6,4	(114)5,1,9,3,2,8,7,4	-	-
108)5,1,9,3,4,2	-		-	-		-	-	-	-
109)5,1,9,3,6,2	-			-		-			-
110)5,1,9,3,7,2	-			-	-	-			-
111)5,1,9,3,8,2	-	-		-	-	-			-
112)5,1,9,3,2,8,4,6	-	-	-	-	-	-	(115)5,1,9,3,2,8,4,6,7	-	-
113)5,1,9,3,2,8,6,4		_	-	-	-	-	(116)5,1,9,3,2,8,6,4,7	-	-
		-		-	-	(117)5,1,9,3,2,8,7,4,6	-	-	-
114)5,1,9,3,2,8,7,4 115)5,1,9,3,2,8,4,6,7			-		-	(117)5,1,9,3,2,8,7,4,6	-	-	
114)5,1,9,3,2,8,7,4	-	-	-	-	-	(117)5,1,9,3,2,8,7,4,6	-	-	-

C. Penjelasan State dan Aksi

State yang berada di file external berupa (###)A,B,C,D,... dengan ### menyatakan nomor statenya dan A,B,C,D menyatakan langkah langkah yang terlah terjadi. Jadi secara tidak langsung nama state tersebut juga telah mengencode apa saja yang terjadi selama permainan berlangsung. Contohnya adalah State di DFA Player (097)5,1,7,3,2,8,9,4,6 player pada awalnya menaruh X di petak 5, lalu cpu menaruh O di petak 1, lalu player di petak 7, lalu cpu di petak 3, dan seterusnya hingga akhirnya player menaruh X di petak 6 dan permainan berakhir seri karena tidak ada lagi petak yang bisa ditempatkan. Berikut adalah bentuk board yang dihasilkan dari contoh state tersebut :



Cara komputer menentukan dimanakah dia harus menaruh O adalah dengan mengambil angka terakhir pada nama state. Contoh: Pada awalnya DFA mulai di state (001)5,1 komputer mengambil angka paling belakang yaitu 1 dan menaruh O di 1. Lalu player menginput 7 sehingga DFA menuju state sesuai tabel transisi yaitu (006)5,1,7,3 komputer mengambil angka paling belakang yaitu 3 dan menaruh O di petak nomor 3, begitu seterusnya selama permainan belum berakhir (Game Over / salah satu menang).

D. Source Code

dfa.h

```
1. #ifndef DFA H
2. #define DFA H
3.
4. #include <stdbool.h>
5. #include "papan.h"
6.
7. #define banyakState 120 //Maksimal banyak state
8. #define banyakTipeInput 9
9.
10. typedef struct
11. {
       int tabelTransisi[banyakState][banyakTipeInput];
12.
13.
       char daftarState[banyakState][23];
14.
       int start;
       bool final[banyakState];
15.
16.
       int curState;
17.
       char stateDilewati[7][23];
18.
       int daftarSimbol[banyakTipeInput];
19.
       int turn;
20. } DFA;
21.
22. //Selektor
23. #define State(D, S, I) (D).tabelTransisi[(S)][(I)]
24. #define Daftar(D, I) (D).daftarState[(I)]
25. #define Final(D,I) (D).final[(I)]
26. #define Current(D) (D).curState
27. #define Start(D) (D).start
28. #define isFinal(D) (D).final[Current(D)]
29. #define StateDilewati(D, I) (D).stateDilewati[(I)]
30. #define Simbol(D,I) (D).daftarSimbol[(I)]
31. #define Turn(D) (D).turn
32.
33. //fungsi
34. int nomorState(char *namaState);
35. //mendapat nomor state dari nama state yang berformat (nomor)namaState
36.
37. void initDFA(DFA *D,char *namaDaftarSimbol, char *namaDaftarState,char *namaStateAwal,
   char *namaTabel, char *namaFinishState,int turn);
38. //Load DFA dengan daftar state dan tabel transisinya, serta mengisi pula finish stateny
   a
39.
40. void next(DFA *D, int X);
41. //DFA pindah ke state setelah diberi input X
42.
43. int getMove(DFA D);
44. //Mendapat move dari current state
46. bool inputValid(DFA D, int input, PAPAN P);
47. //Mengecek apakah input valid
48. #endif
```

```
    #include <stdbool.h>

2. #include <stdio.h>
3. #include <string.h>
4. #include "dfa.h"
5. //fungsi
6.
7. int nomorState(char* namaState)
8. //mendapat nomor state dari nama state yang berformat (nomor)namaState
9. {
10.
        //"001" = 0*100 + 0 * 10 + 1
        //"012" = 0*100 + 1 * 10 + 2
11.
        int ratusan = (int)namaState[1] - 48;
12.
        int puluhan = (int)namaState[2] - 48;
13.
       int satuan = (int)namaState[3] - 48;
14.
15.
        return 100*ratusan+10*puluhan+satuan;
16. }
17. void initDFA(DFA *D, char *namaDaftarSimbol, char *namaDaftarState, char *namaStateAwal,
   char *namaTabel, char *namaFinishState,int turn)
18. //Load DFA dengan daftar state dan tabel transisinya, serta mengisi pula finish stateny
   а
19. {
20.
       bool baca;
21.
        //baca daftar state
        FILE *fileDaftar = fopen(namaDaftarState, "r");
22.
23.
        baca = true;
24.
        int nomor = 0;
       while (baca)
25.
26.
27.
            char dummy[23] = "";
28.
            if (fscanf(fileDaftar, "%s", dummy) != EOF)
29.
            {
30.
                //Belum eof
31.
                strcpy(Daftar(*D, nomor), dummy);
32.
                nomor++;
33.
            }
34.
            else
35.
            {
36.
                //EOF
37.
                baca = false;
38.
39.
40.
        fclose(fileDaftar);
41.
        //baca tabel transisi
42.
        FILE *fileTabel = fopen(namaTabel, "r");
43.
        baca = true;
44.
        int bar = 0;
45.
       while (baca)
46.
47.
            char temp[9][23];
            if (fscanf(fileTabel, "%s %s %s %s %s %s %s %s %s", temp[0], temp[1], temp[2],
48.
   temp[3], temp[4], temp[5], temp[6], temp[7], temp[8]) != EOF)
49.
50.
                //0 jika tidak pindah state
51.
                for (int i = 0; i < 9; i++)</pre>
52.
53.
                    //cek apakah yang dibaca merupakan - atau nama state
54.
                    if (temp[i][0] == '-')
55.
```

```
56.
                         //jika - maka tabel transisinya mengacu balik ke stateawal nya
57.
                        State(*D, bar, i) = bar;
58.
                    }
59.
                    else
60.
61.
                         State(*D, bar, i) = nomorState(temp[i]);
62.
63.
                bar += 1;
64.
65.
            }
66.
            else
67.
            {
68.
                //EOF
69.
                baca = false;
70.
71.
72.
        fclose(fileTabel);
73.
        //baca finish state
74.
        FILE *fileFinish = fopen(namaFinishState, "r");
75.
        baca = true:
76.
        //assign semua false
        for (int i = 0; i < banyakState; i++)</pre>
77.
78.
79.
            Final(*D, i) = false;
80.
81.
       while (baca)
82.
83.
            char dummy[23];
84.
            if (fscanf(fileDaftar, "%s", dummy) != EOF)
85.
            {
86.
                //Belum eof
87.
                //Dapatkan nomor state
88.
                int nomor = nomorState(dummy);
89.
                Final(*D, nomor) = true;
90.
91.
            else
92.
93.
                //EOF
94.
                baca = false;
95.
            }
96.
97.
        fclose(fileFinish);
98.
        //Baca start State dari file external
99.
        FILE *fileStart = fopen(namaStateAwal, "r");
100.
               fscanf(fileStart, "%d",&Start(*D));
101.
               fclose(fileStart);
102.
               Current(*D) = Start(*D);
103.
               //Baca daftar simbol
104.
               FILE *fileSimbol = fopen(namaDaftarSimbol, "r");
105.
               for(int i = 0;i< banyakTipeInput;i++)</pre>
106.
107.
                    fscanf(fileSimbol,"%d",&Simbol(*D,i));
108.
109.
               fclose(fileSimbol);
110.
111.
               Turn(*D) = turn;
112.
               for(int i = 0; i < 7; i++){
                    strcpy(StateDilewati(*D,i),"");
113.
114.
           }
115.
116.
```

```
117.
           void next(DFA *D, int X)
118.
           //DFA pindah ke state setelah diberi input X
119.
120.
               //Prekondisi : Input valid
121.
               //Final : DFA pindah ke state yang sesuai setelah diberi input X
122.
               Current(*D) = State(*D, Current(*D)-1, X-1); //-1 karena array dari 0
123.
           }
124.
125.
           int getMove(DFA D)
126.
           //Mendapat move dari current state
127.
128.
               char state[23];
129.
               strcpy(state, Daftar(D, Current(D)-1)); //-1 karena array dari 0
130.
               return (int)state[strlen(state) - 1] - 48; //minus 48 karena '1' = 49
131.
           }
132.
133.
           bool inputValid(DFA D,int input,PAPAN P)
           //Mengecek apakah input valid
134.
135.
136.
               //cek apakah berada di range 1-9
137.
               bool valid = false;
138.
               for(int i = 0;i<banyakTipeInput;i++)</pre>
139.
140.
                   if(Simbol(D,i) == input){
141.
                       valid = true;
142.
                       break;
143.
                   }
144.
145.
               if (valid)
146.
147.
                   //cek apakah diletakan di tempat yang kosong
148.
                   int bar = (input - 1) / 3, kol = (input - 1) % 3;
149.
                   if (isi(P, bar, kol) == '_')
150.
151.
                       return true;
152.
                   }
153.
                   else
154.
155.
                       return false;
156.
157.
               }
158.
               else
159.
160.
                   return false;
161.
               }
162.
```

game.h

```
1. #ifndef GAME_H
2. #define GAME_H
3.
4. #include "dfa.h"
5. #include "papan.h"
6.
7. void startGame(DFA* D, char start);
8. //Memulai game dengan dfa D dan karakter awal start (X atau 0)
9.
10. #endif
```

game.c

```
    #include <stdio.h>

2. #include <string.h>
3. #include "dfa.h"
4. #include "game.h"
5. #include "papan.h"
6.
7. void startGame(DFA *D, char start)
8. //Memulai game dengan dfa D dan karakter awal start (X atau O)
9. {
       PAPAN game;
10.
11.
        int pilihan, noStateDilewati = 0;
12.
       if (start == 'X')
13.
14.
           printf("Game dimulai, player harus menaruh di tengah\n");
15.
       }
16.
       else
17.
18.
           printf("Game dimulai, cpu harus menaruh di tengah\n");
19.
20.
       //Catat state awal
       strcpy(StateDilewati(*D,noStateDilewati),Daftar(*D,Start(*D)-1)); //Isi -
21.
   1 karena nomor State dimulai dari 1 dan indeksnya 0
22.
       noStateDilewati += 1;
        initPapan(&game, start);
23.
24.
       printBoard(game);
                                     //Ganti turn
25.
       Turn(*D) += 1;
26.
       while (cekMenang(game) == 0) //Berhenti saat game telah selesai
27.
28.
           if (Turn(*D) % 2)
29.
30.
                //jika 1, maka if akan true, menandakan giliran player skrng
31.
                printf("Sekarang giliranmu, input angka 1-
   9 yang menandakan tempat di papan : ");
                scanf("%d", &pilihan);
32.
33.
                //Validasi input
34.
               while (!inputValid(*D, pilihan, game))
35.
36.
                    printf("Input tidak valid, masukkan 1-
   9 dan di tempat yang kosong : ");
                    scanf("%d", &pilihan);
37.
38.
39.
                //Sudah valid
40.
                //pasang gerakan disana
41.
                int bar = (pilihan - 1) / 3, kol = (pilihan - 1) % 3;
```

```
42.
                isi(game, bar, kol) = 'X';
43.
                //next state dfa
44.
                next(D, pilihan);
45.
                //Catat perubahan state
46.
                strcpy(StateDilewati(*D,noStateDilewati),Daftar(*D,Current(*D)-1));
47.
                //Current(*D)-
   1 karena nomor state dimulai dari 1, dan di indeks dimulai dari 0
48.
                noStateDilewati += 1;
49.
                //print papan ke layar
50.
                printBoard(game);
51.
            }
           else
52.
53.
            {
                //jika 0, maka if akan false, menandakan giliran komp skrng
54.
55.
                printf("Sekarang giliran cpu\n");
                //Dapatkan move paling optimal
56.
                int optimal = getMove(*D);
57.
                int bar = (optimal - 1) / 3, kol = (optimal - 1) % 3;
58.
                isi(game, bar, kol) = '0';
59.
                //print papan ke layar
60.
61.
                printBoard(game);
62.
63.
            Turn(*D) += 1; //Ganti turn
64.
65.
        //Game selesai, output hasil permainan
66.
       int kondisiAkhir = cekMenang(game);
        if (kondisiAkhir == 1)
67.
68.
        {
69.
           printf("CPU MENANG\n");
70.
71.
        else if (kondisiAkhir == 2)
72.
73.
           printf("PLAYER MENANG\n");
74.
75.
       else if (kondisiAkhir == 3)
76.
77.
            printf("DRAW\n");
78.
79.
        //Print semua state yang dilewati
       printf("State yang telah dilewati : \n");
        for (int i = 0; i < noStateDilewati; i++)</pre>
                                                      //Loop dari 0 sampai Turn(*D)-
   1 karena nilai Turn saat selesai adalah banyak state yang dilewati+1
82.
83.
            printf("%s\n",StateDilewati(*D,i));
84.
85. }
```

papan.h

```
1. #ifndef BOARD H
2. #define BOARD H
3.
4. #include<stdbool.h>
5.
6. typedef struct{
7.
       char data[3][3];
8. }PAPAN;
9.
10. #define isi(P,R,C) (P).data[(R)][(C)]
11. #define giliran(P) (P).giliran
12.
13. void initPapan(PAPAN* P,char start);
14. //Init papan dengan 0 semua, giliran = true
15. void printBoard(PAPAN P);
16. //Print papan ke layar
17. int cekMenang(PAPAN P);
18. //return 0 jika belum ada yang menang,1 jika 0 menang, 2 jika X menang, 3 jika draw
19. #endif
```

papan.c

```
    #include <stdbool.h>

2. #include <stdio.h>
3. #include "papan.h"
4.
5. void initPapan(PAPAN *P, char start)
6. //Init papan dengan 0 semua, giliran = true
7. {
8.
        for (int i = 0; i < 3; i++)</pre>
9.
10.
            for (int j = 0; j < 3; j++)
11.
12.
                isi(*P, i, j) = '__';
13.
14.
15.
        isi(*P, 1, 1) = start;
16. }
17. void printBoard(PAPAN P)
18. //Print papan ke layar
19. {
        printf("\n");
20.
21.
        for (int i = 0; i < 3; i++)</pre>
22.
23.
            for (int j = 0; j < 3; j++)
24.
25.
                printf("%c ", isi(P, i, j));
26.
27.
            printf("\n");
28.
29.
        printf("\n");
30.}
31. int cekMenang(PAPAN P)
32. //return 0 jika belum ada yang menang,1 jika O menang, 2 jika X menang, 3 jika draw
33. {
34.
        //Cek baris
35.
        for (int i = 0; i < 3; i++)</pre>
```

```
36. {
37.
           if ((isi(P, i, 0) == isi(P, i, 1)) && (isi(P, i, 0) == isi(P, i, 2)))
38.
39.
                char cek = isi(P, i, 0);
               if (cek == '0')
40.
41.
42.
                  return 1;
43.
44.
               else if (cek == 'X')
45.
46.
                  return 2;
47.
                }
48.
49.
       //Cek kolom
50.
51.
       for (int i = 0; i < 3; i++)</pre>
52.
53.
           if ((isi(P, 0, i) == isi(P, 1, i)) && (isi(P, 0, i) == isi(P, 2, i)))
54.
55.
                char cek = isi(P, 0, i);
56.
               if (cek == '0')
57.
58.
                  return 1;
59.
60.
               else if (cek == 'X')
61.
62.
                  return 2;
63.
                }
64.
65.
66.
       //Cek diagonal
67.
       if ((isi(P, 0, 0) == isi(P, 1, 1)) && (isi(P, 0, 0) == isi(P, 2, 2)))
68.
69.
            char cek = isi(P, 0, 0);
70.
           if (cek == '0')
71.
72.
              return 1;
73.
74.
           else if (cek == 'X')
75.
76.
               return 2;
77.
            }
78.
79.
        //Cari apakah ada yang kosong
80.
       for (int i = 0; i < 3; i++)
81.
82.
           for (int j = 0; j < 3; j++)
83.
               if (isi(P, i, j) == '_')
84.
85.
86.
                  return 0;
87.
                }
88.
89.
       //Jika sampai sini, berarti draw
90.
91.
       return 3;
92.}
```

main.c

```
1. #include<stdio.h>
2. #include "dfa.h"
3. #include "game.h"
4.
5. int main()
6. {
      int pilihan;
7.
8.
      printf("Selamat datang di permainan \n");
      9.
10.
      printf("... ##.::'##... ##:::'## ##:::'##... ##::::'##... ##:::\n");
      printf("::: ##:::: ##:: ##::: ##::: ##::: ##::: ##::: ##::: ##::: ##::: ##::: ##::: ##::: ##::: h");
11.
12.
      printf("::: ##:::: ##:: ##:::: ##::: ##::: ##::: ##::: ##::: ##::: ##::: ##::: ##::: ##::: ##::: |n");
      printf("::: ##:::: ##:: ##::: ##::: ##::: ##::: ##:::: ##::: ##::: ##::: ##::: ##::: ##::: |n");
13.
14.
      printf("::: ##:::: ##::: ##::: ##:::: ##:::: ##::: ##:::: ##::: ##::: ##:::: ##::: ##:::: h");
15.
      16.
      printf(":::..:\n");
      printf("Input 0 jika ingin cpu duluan, input 1 jika anda ingin mulai duluan\n");
18.
      printf("Pilihanmu : ");
19.
      scanf("%d",&pilihan);
20.
      if(pilihan==0){
21.
          DFA cpuFirst;
22.
          initDFA(&cpuFirst, "State/daftarSimbol.txt", "State/daftarStates.txt", "State/stat
  eAwal.txt", "State/states.txt", "State/finishState.txt",0);
23.
          startGame(&cpuFirst, '0');
24.
      }else{
25.
          DFA playerFirst;
          initDFA(&playerFirst, "State/daftarSimbol.txt", "State/daftarStatesPlayer.txt", "S
  tate/stateAwalPlayer.txt","State/statesPlayer.txt","State/finishStatePlayer.txt",1);
27.
          startGame(&playerFirst,'X');
28.
29.
      return 0;
30.}
```

E. Contoh Input dan Output Program

Tampilan awal:

Cpu mulai duluan :

```
Pilihanmu : 0
Game dimulai, cpu harus menaruh di tengah
_ 0 _
_ - - -
Sekarang giliranmu, input angka 1-9 yang menandakan tempat di papan :
```

Game Over (Draw) dan state yang telah dilewati :

```
Sekarang giliran cpu

0 0 X

X X 0
0 X

Sekarang giliranmu, input angka 1-9 yang menandakan tempat di papan : 9

0 0 X

X X 0
0 X X

X O
0 X X

DRAW

State yang telah dilewati :
(001)5,1
(003)5,1,3,7
(040)5,1,3,7,4,6
(045)5,1,3,7,4,6,8,2
(048)5,1,3,7,4,6,8,2,9
```

Game Over(CPU menang) dan state yang telah dilewati: Sekarang giliran cpu o x _ 0 _ Sekarang giliranmu, input angka 1-9 yang menandakan tempat di papan : 3 OXX 0 _ _ _ _ Sekarang giliran cpu OXX _ 0 _ _ _ o CPU MENANG State yang telah dilewati : (001)5 (003)5,2,1 (022)5,2,1,3,9 Input tidak valid (bukan 1-9): Game dimulai, cpu harus menaruh di tengah _ 0 _ Sekarang giliranmu, input angka 1-9 yang menandakan tempat di papan : 10 Input tidak valid, masukkan 1-9 dan di tempat yang kosong : Input tidak valid (bukan petak kosong): Sekarang giliran cpu o x _ 0 _ Sekarang giliranmu, input angka 1-9 yang menandakan tempat di papan : 1 Input tidak valid, masukkan 1-9 dan di tempat yang kosong :