Formal Languages and Automata Recitation-1

Tacettin Ayar (ayart@itu.edu.tr)

Example #1

- Build a DFA for the following language:
  L = { w | w is a binary string that has even number of 1s and even number of 0s}
- ?

Example #1 (Solution)

C0C1 = Even number of 0s

Even number of 1s

C0T1 = Even number of 0s

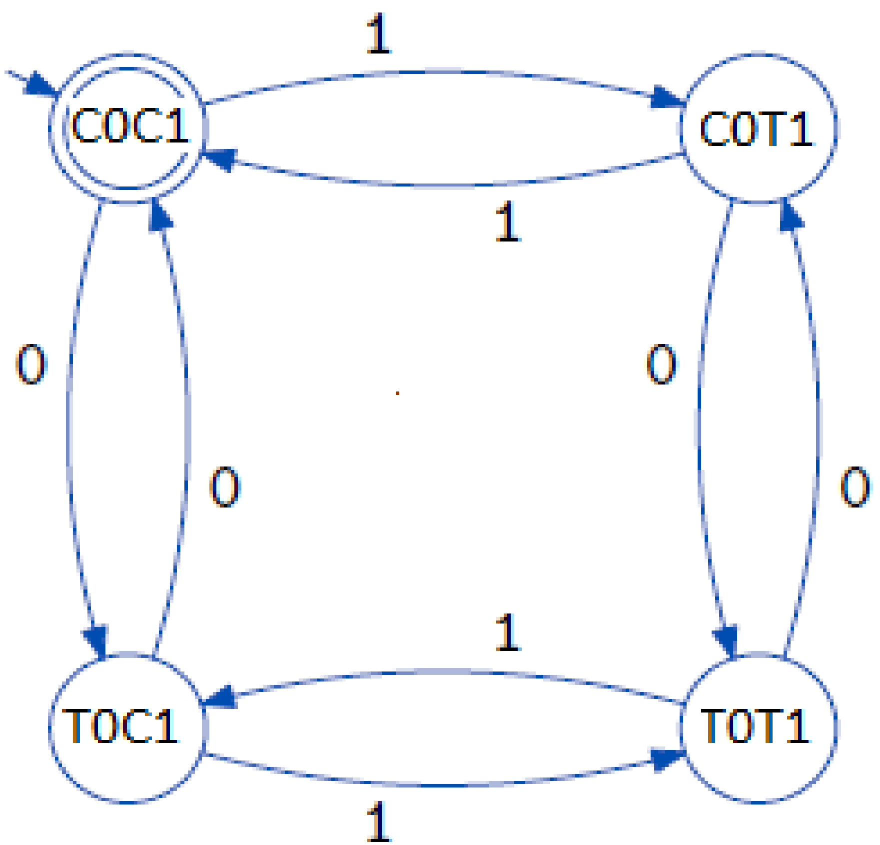
Odd number of 1s

T0T1 = Odd number of 0s

Odd number of 1s

T0C1 = Even number of 0s

Even number of 1s



Example #2

- $\hat{\delta}(q,wa) = \delta(\hat{\delta}(q,w), a)$
- Work out example #1 using the input sequence w=10010, a=1:
  - $\hat{\delta}(q_0,wa) = ?$

Example #2 (Solution)

$\hat{\delta}(C0C1,100101) = \delta(\hat{\delta}(C0C1,10010), 1)$

$\hat{\delta}(C0C1,10010) = \delta(\hat{\delta}(C0C1,1001), 0)$

$\hat{\delta}(C0C1,1001) = \delta(\hat{\delta}(C0C1,100), 1)$

$\hat{\delta}(C0C1,100) = \delta(\hat{\delta}(C0C1,10), 0)$

$\hat{\delta}(C0C1,10) = \delta(\hat{\delta}(C0C1,1), 0)$

$\hat{\delta}(C0C1,1) = C0T1$

$\hat{\delta}(C0C1,10) = \delta(C0T1, 0) = T0T1$

$\hat{\delta}(C0C1,100) = \delta(T0T1, 0) = C0T1$

$\hat{\delta}(C0C1,1001) = \delta(C0T1, 1) = C0C1$

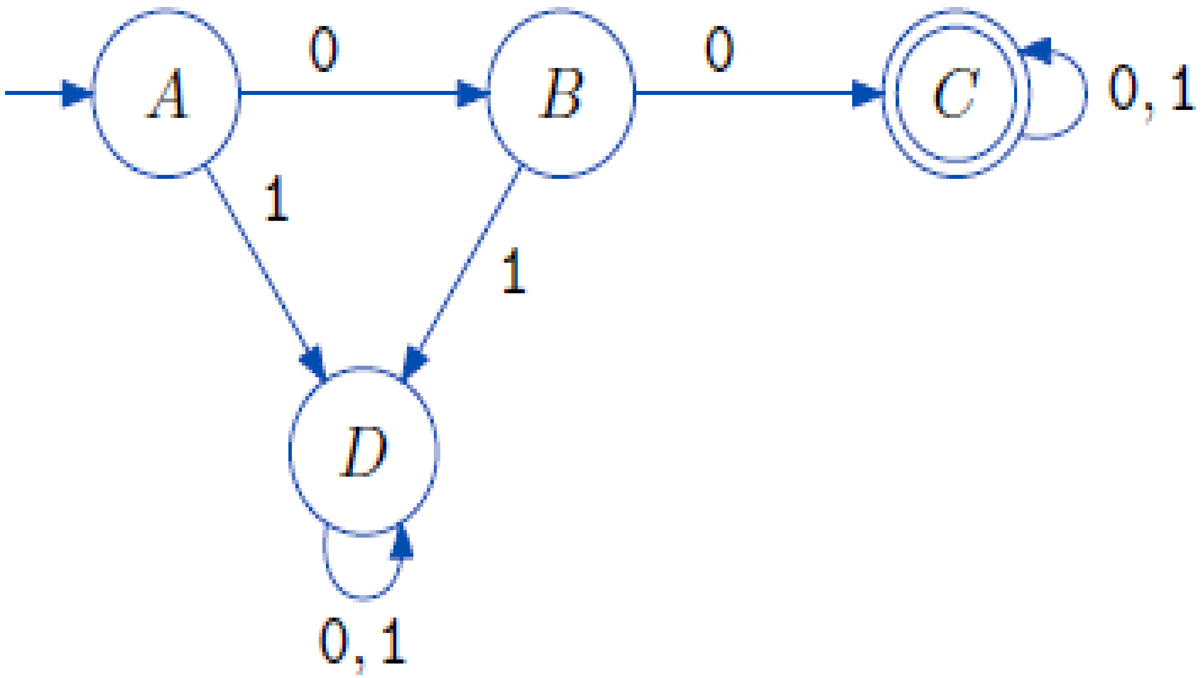
$\hat{\delta}(C0C1,10010) = \delta(C0C1, 0) = T0C1$

$\hat{\delta}(C0C1,100101) = \delta(T0C1, 1) = T0T1$

Example #3

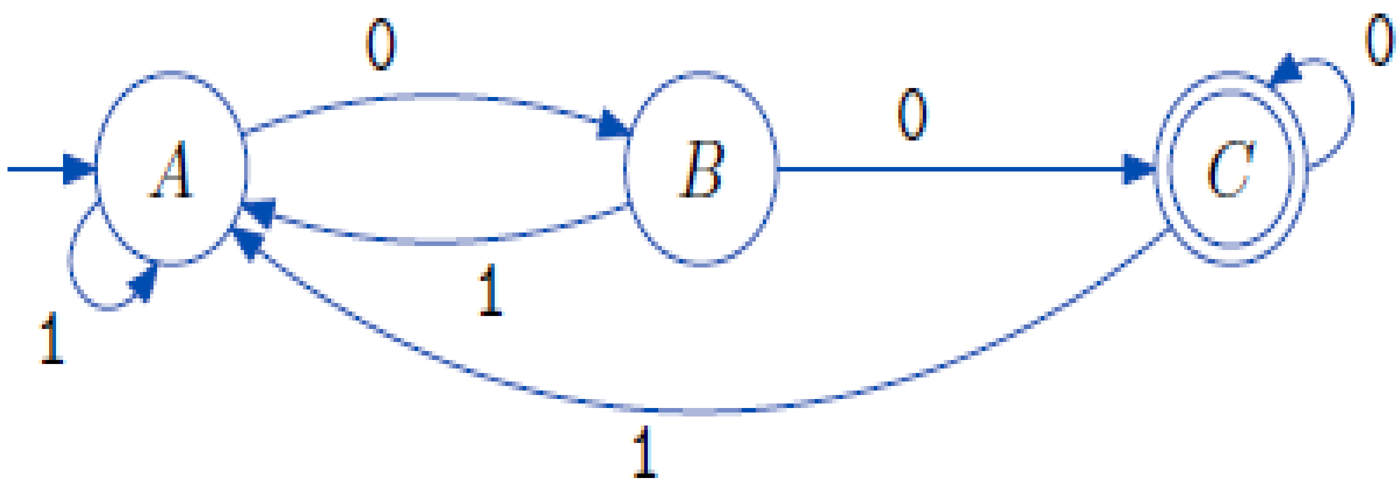
- Build a DFA for the following language:

  L = { w | w starts with 00}



Example #4

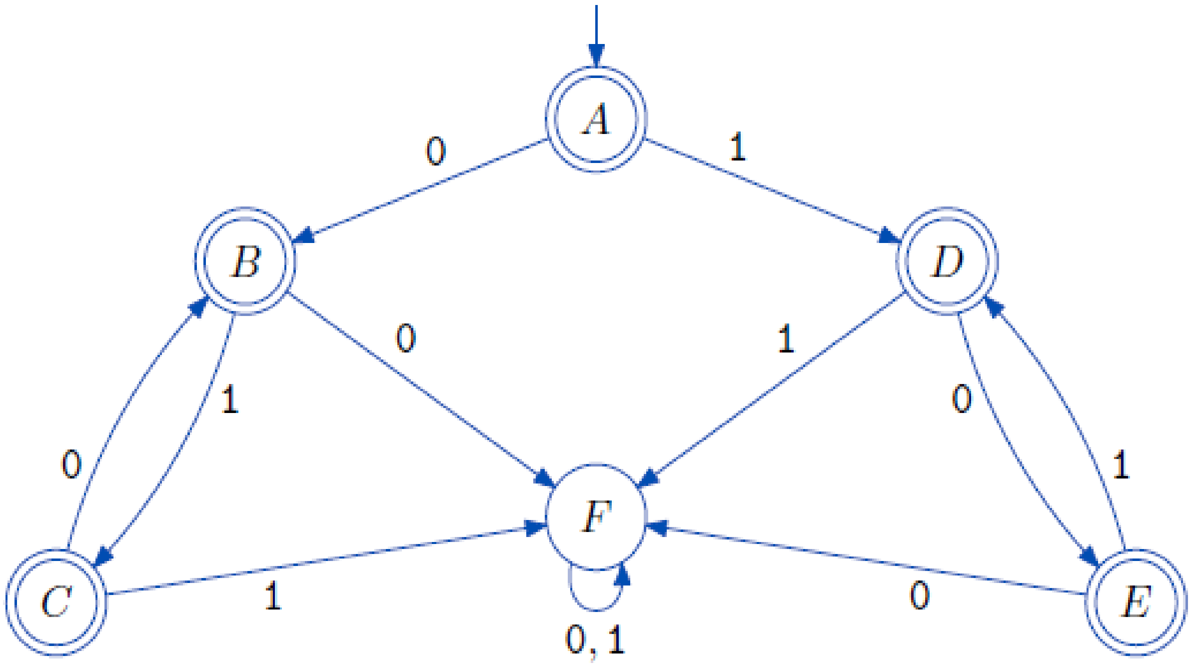
- Build a DFA for the following language:
  L = { w | w ends with 00}



Example #5 (1)

- Build a DFA for the following language:

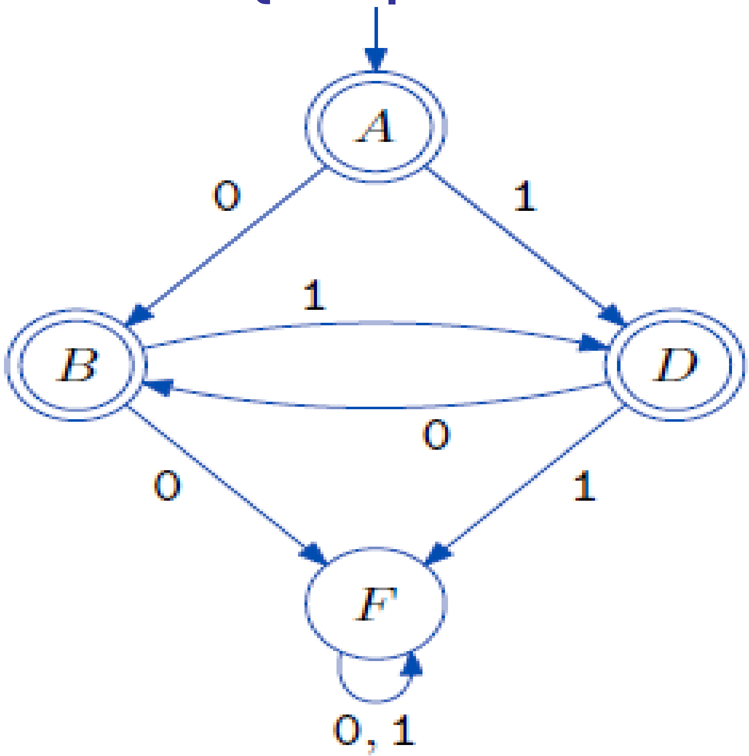
  L = { w | w consists of alternating 0's and 1's}



Example #5 (2)

- Build a DFA (again) for the following language:

  L = { w | w consists of alternating 0's and 1's}

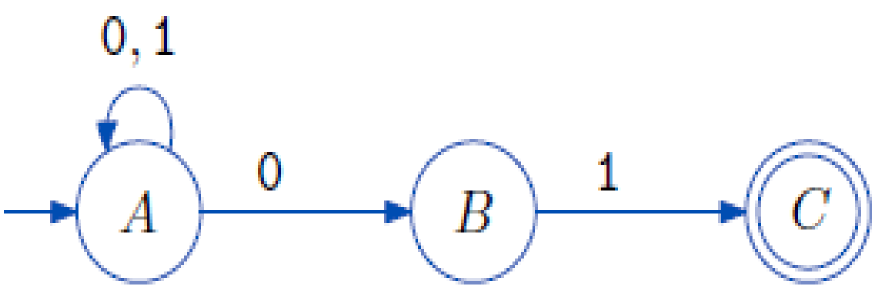


RESULT:

Although a given DFA corresponds to only one language, a given language can have many DFAs that accept it

Example #6

- Build an NFA for the following language:
  L = { w | w ends in 01}



REFERENCES

- “Introduction to Automata Theory,Languages, and Computation“

  by J. E. Hopcroft, R. Motwani, and J. D.Ullman

- “Introducing The Theory Of Computation”

  by W. Goddard