United International University
Department of Computer Science and Engineering

CSI 233: Theory of Computing Mid-term Examination : Spring 2019

Total Marks: 30 Time: 1 hour 45 minutes

Answer all the 5 questions. Numbers to the right of the questions denote their marks.

1. (a) Draw the state diagram of a DFA that accepts the strings which contain at least two 1's. (Accepted: 101100, 01100; Rejected: 0100, 10) Given, $\sum$={0,1}. Write the *start state, accept states* and *transitions* of the DFA besides showing the state diagram. [2+2]

   (b) Design a DFA that accepts the following language:
   L = {*The length of each string will be divisible by 3*}.
   (Accepted: empty string, aba, aabbab, Rejected: ab,a, aabb). Given, $\sum$= {a, b}. Draw only the state diagram. [2]

2. (a) Draw the state diagram of an NFA / ϵ-NFA which accepts strings those do not contain substring "main". Here, $\sum$= {a,b,c,d,. . . ,z} [2]

   (b) Draw the state diagram of an NFA / ϵ-NFA which accepts binary strings which has even values.
   Here, $\sum$= {0, 1} (Accepted: 01010, Not accepted : 10101) [2]

   (c) Draw the state diagram of an NFA / ϵ-NFA which recognizes FIFA World Cup years in 4 digits. Assume World Cup occurs every 4 years starting from 2002. Here, $\sum$ = {0,1,2,3,4,5,6,7,8,9}. [2]

3. (a) Convert the following Regular Expression to ϵ-NFA over the alphabet, $\sum$={1,c,a,0,Z,2,3}
   (ac(10))* (2+ϵ+Z1*)3a* [2]

   (b) Write a regular expression over the alphabet $\sum$ = {a, b} for the language where *no string ends with aa.* [2]

   (c) Write a regular expression over the alphabet $\sum$ = {0, 1} *for the language where all strings start with 0 and have odd length, or start with 1 and have even length.* [2]

4. Consider the following ϵ-NFA:

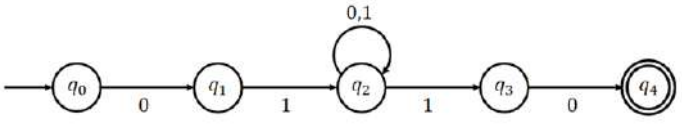
Input states	ϵ	a	b
→Q1	{Q2}	{Q3}	ϕ
*Q2	ϕ	{Q1}	ϕ
Q3	ϕ	{Q2}	{Q2,Q3}

   (a) Compute the ϵ-closure of each state. [2]

   (b) Convert the ϵ-NFA to equivalent DFA. Show both transition table and state diagram of DFA. [4]

5. (a) Describe the languages of the following regular expressions: [1.5+1.5]
   i. *(ab+ba)aa(a+bc)**
   ii. *(ϵ+aa)(bb*a)**

   (b) Find out if the following strings are accepted by the given NFA: [1.5+1.5]



   i. 11010010
   ii. 01101110