Bahria University, Islamabad Campus
Department of Computer Science
Final Assessment
Class/Section: BSCS 4A,B MSCS 0A
(Spring 2020 Semester)

Course:	**Theory of Automata**	Date/Time Assigned: 9-7-2020/ 10:00 AM
Course Code:	CSC-315	Submission Date/Time: 9-7-2020/ 6:00 PM
Faculty's Name:	Dr. Sabina Akhtar	Max Marks: 50

INSTRUCTIONS:

I. **Your answers must be handwritten. You can take the picture of your workings and paste them in this document.**

Student Name: ____________________ Enrollment No. ____________

Question # 1: Design a Deterministic Finite Automata for the machine that identifies whether the given string contains "sme" as a substring or not. If it contains, the machine outputs accept otherwise rejects. For example, the word "assessment" contains "sme" as a substring, therefore the machine should accept.

The alphabet Σ is {a, b, c, ... z}. Provide transition diagram along with the formal specifications. Transition table should be used to show the transition function.

(marks: 8+4 = 12)

Question # 2

The if-then-else is a standard programming language construct that is used to check for different options. However, it's context free grammar can become ambiguous because of the dangling else problem.

Enrollment Number: ______________________________

stmt → if (expr) stmt | if (expr) stmt else stmt | S

S → i=0;

Start variable for the above grammar is stmt. The alphabet Σ is {if , else , (,) , i=0; , expr}.

Prove that the above grammar is ambiguous by finding out a string for which the grammar can generate at least two parse trees. Your solution must clearly show both the parse trees along with its two left-most derivations and two right-most derivations.

(marks: 4+4+4 = 12)

Question # 3 Design Non-deterministic Pushdown Automata for the following language

$$L = \{w \mid w \text{ is an odd length palindrome}\}$$

Only show the transition diagram.

(marks: 10)

Question # 4 **Let's describe a Turing machine (TM) M that accepts the strings belonging to L = $\{0^{2^n} \mid n \geq 0\}$, the language consisting of all strings of 0s whose length is a power of 2.**

(marks: 14+2 = 16)

M = ($\{q_1, q_2, q_3, q_4, q_5, q_6, q_7\}$, {0}, {0, x, B}, δ, q_1, B, $\{q_7\}$)

State	Symbol 0	Symbol x	Symbol B
-> q_1	(q_2, B, R)	(q_7, x, R)	(q_7, B, R)
q_2	(q_3, x, R)	(q_2, x, R)	(q_7, B, R)
q_3	(q_4, 0, R)	(q_3, x, R)	(q_5, B, L)
q_4	(q_3, x, R)	(q_4, x, R)	(q_7, B, R)
q_5	(q_5, 0, L)	(q_5, x, L)	(q_2, B, R)
q_6	-	-	-
*q_7	-	-	-

Simulate the word 00000 starting from the first configuration ***$q_1$00000*** **till the final configuration.** ***Is the word accepted by the turing machine?***