CS 301 Theory of Automata

Serial No:

Final Exam Section II

Total Time: 2 Hours
Total Marks: 110 of Section II

Wednesday, Dec 14, 2016

Course Instructor

Dr Aftab Maroof, Dr Waseem Shehzad, Dr Labiba Fahad and Ms. Mehreen Alam

_____
Signature of Invigilator

_____ _____ _____ _____
Student Name Roll No Section Signature

DO NOT OPEN THE QUESTION BOOK OR START UNTIL INSTRUCTED.

Instructions:

1. This is Section II, the design part of the exam.
2. Attempt all of them. Read the question carefully, understand the question, and then attempt it.
3. No additional sheet will be provided for rough work. Use the back of the last page for rough work.
4. After asked to commence the exam, please verify that you have **twelve (12)** different printed pages including this title page. There are total of **11 questions**.
5. Use permanent ink pens only. Any part done using soft pencil will not be marked and cannot be claimed for rechecking.

	1	2	3	4	5	6	7	8	9	10	11	**Total**
Total Marks	10	10	10	10	10	10	10	10	10	10	10	**110**
Marks Obtained												

Vetted By: _____ Vetter Signature: _____

Q1. **[5+5 pts]** Write regular expression for the following languages:

1. defined over ∑ = {a,b} where no word contains the substring **abb**

   R.E. =

2. Defined over **∑ = { d, . }** to identify an IP address which could in the form **ddd.ddd.ddd.ddd**, where d represents a digit from 0-9 and IP address can hold hypothetical values like 999.999.999.999

   R.E. =

Q2. **[10 pts]** Construct a Context Free grammar over ∑ = {a,b} whose language is:

1. **EQUAL-EQUAL = { Δ, ab, ba, aabb, abab, baba, bbaa, abba, baab, aaabbb, aabbab, …….}**

2. **TRAILING-COUNT = { $wa^{length(w)}$ | w is a string defined over $(a+b)^*$ }**

Q3. **[10 pts]** Convert the following grammar to Chomsky Normal Form Grammar. Show all the intermediary steps clearly.

E → E + E | ET
E → E * E | TE
E → (E)
E → 7 | λ
T → 8T |TT
U → 7 | λ

Q4. **[10 pts]** Design a deterministic PDA for the language over ∑ = {a,b} such that

L = { **a^nS, where S starts with b and length(S)=n**}
= { **ab, aaba, aabb, aaabaa, aaabab, aaabba, aaabbb, aaaabaaa, …..** }

Q5. **[10 pts]** Use Pumping Lemma to prove that the following language is not Context-Free

$A = \{ 0^n 1^n 0^{2n} \mid n>0 \}$

Q6. **[10 pts]** Construct a pushdown automata for the set of all strings of the form $\mathbf{0^a1^b}$ such that **a >=b**.?

Q7. **[10 pts]** Design a Turing machine to reverse the input string. For example, on input aaaba and babaaabbb the outputs are abaaa and bbbaaabab respectively. Assume the tape head points in the start of the input while the tape head points at the start of the output when the computation is done. You are **allowed** to use the sub program **DELETE** only. An example for input aaaba is given below:

Status of tape on input is:

#	a	a	a	b	a	Δ	Δ	.	.
	↑								

Status of tape at the output is:

#	a	b	a	a	a	Δ	Δ	.	.
	↑								

Q8. **[10 pts]** For the language **L = { w | length(w)=2^n , where n>0 and w ∈ to (a+b)*}** , design a 2PDA.

Q9. **[10 pts]** Build a Post Machine that takes two natural number x and y in unary notation separated by a $ and leave on the Store (x - y) if x > y, and a '0' if y >= x.

Q10. **[10 pts]** Design a Turing machine that takes input two non-negative numbers and performs the **mod** operation on them, for example, **mod(7,3)=1**. Assume the input is in unary notation, the first number is always bigger than the second one, 0 is the separator, the tape head points in the start of the input while the tape head points at the start of the output when the computation is done. None of the numbers can be null. You are also **allowed** to use the sub programs of **INSERT** and **DELETE**. An example for mod(7,3)=1 is given below:

Status of tape on input is:

#	1	1	1	1	1	1	1	0	1	1	1	Δ	Δ
	↑												

Status of tape at the output is:

#	1	Δ	Δ	.	.	.
	↑					

Q11. **[10 pts]** Design a Turing machine that takes input a non-negative number x and performs the computable function **$f(x) = 2^x$**. Assume the input is in unary notation, the tape head points in the start of the input while the tape head points at the start of the output when the computation is done. Two examples are given below for your understanding:

Status of tape on input is:

#	1	1	Δ	Δ	Δ	.	.	.
	↑							

Status of tape at the output is:

#	1	1	1	1	Δ	Δ	.	.
	↑							

Status of tape on input is:

#	1	1	1	Δ	Δ	.	.	.
	↑							

Status of tape at the output is:

#	1	1	1	1	1	1	1	1	Δ	Δ	.	.
	↑											