Birla Institute of Technology & Science-Pilani
Hyderabad Campus
1st semester 2016-17
Theory of Computation (CSF 351) Test-1(Regular)

Exam Dt: 09.09.2016 Weightage: 30% Time: 60 Mins Type: Close Book

SOLUTIONS

Q.1 Draw a DFA for the language over {*a*,*b*}, where each ***a*** is followed by at least one ***b***. (7)

Q.2 Brief on Moore machine, with full and appropriate formal (mathematical) description. (8)

Q.3 Explain the procedure for computing the intersection of two Regular languages (in terms of union and difference). (7)

Q.4 Construct DFA for the language represented by the RE- ***(00)*(11)**** (11)

Q.5 Convert the following NDFA to DFA. (Complete working is required). (17)

Set of transitions for the NDFA:
{ (q1,e)→q2; (q1,1)→q3; (q3,0)→q3; (q3,0)→q2; (q3,1)→q2; (q2, 0)→q1 }

Q.6 Give the Regular expression for the language accepted by the following FA. (10)

{ (q0,1)→q1; (q0,1)→q2; (q0,0)→q3; (q1,0)→q4; (q2,1)→q3; (q3,0)→q3;
(q3,1)→q4; (q4, e)→q5 }

Birla Institute of Technology & Science-Pilani
Hyderabad Campus
1st semester 2016-17
Theory of Computation (CSF 351) Test-2(Regular)

Dt: 24.10.2016 Weightage: 30% Time: 60 Mins Type: Open Book
Note: No additional sheets will be supplied.

Q.1 Construct CFG for the following languages

(i) Language over ∑= {a,b,c,d} and any string is of the form $\{a^n b^n c^m d^m \mid n, m \geq 1\}$

(ii) Language over ∑= {a, b} and any string contains exactly two ***a***s.

(7+8=15)

Q.2 Prove that following CFGs are ambiguous.

(i) S→0Y | 01
Y→XY1 | 0
X→ 0XY | 0

(ii) S→SBS | a
B→bB | b (7+8=15)

Q.3 For the following CFG give two derivations D1 and D2, for the string ***baaaaab*** such that D1 precedes D2. (7)

S→a | Sa | SSb | bSS | SbS

Q.4 Construct a Deterministic PDA for the following CFG.

S→ aAS | bS | c
A→ d | e (15)

Now, using the standard tabular representation show how the string ***adabc*** is accepted.

Note: Use the guidelines for mapping CFG to PDA.
The symbol 'e' stands for null.

Q.5 For the following CFG, with stack position after each operation, show how bottom-up parsing works in the process of accepting the string- ***aabbccb***. Do not give transitions of PDA, not required, just depict the process using stacks. (8)

S→ AB
A→ ab | aAB
B→ b |cB

Note: For all the above CFGs, S is the starting NT, and the symbol '|' stands for alternation.
Lower-case is used for terminals and upper-case for NTs.

***

Birla Institute of Technology & Science-Pilani
Hyderabad Campus
1st semester 2016-17
Theory of Computation (CS F351)
Comprehensive Examination (Regular)

Date: 03.12.2016 (AN) Time: 3 Hrs. Type: Closed Book Total marks: 80

Note : (i) Follow the notations used in the Text book/class.
(ii) Answer all questions of a PART (A/B) together in the same order, else not evaluated.
(iii) For answering each question (not subparts), go to next fresh page.
(iv) Perfection in presentation of solution/concept, and neatness carry some weightage.

PART-A (40 Marks)

1. (a) Give a DFA for the language over {a,b} where each string has at least one occurrence of ***ab*** or ***bba*** (could be both).
   (b) Give a Regular Expression for strings over {0,1} containing odd number of ***1***s (ex: 1, 3, 5..)..
   (c) Convert the following NDFA to DFA.

   { (q0,0)→q0; (q0,e)→q1; (q1,0)→q2; (q1,e)→q3; (q2,1)→q1; (q3,0)→q3; }

   [4+3+5=12]

2. (a) Give CFG for the language given by RE ***abba (baa)* aab (aabb)****
   (b) Give a Finite Automata for the CFG {S→baS | aA; A→ bbA | bb}.
   Note: follow a systematic way.
   (c) Give the symbols (very much formal) that can be possibly used in a basic Regular Expression. Also the recursive definitions (which define what can be a RE), used to obtain Regular Expressions.

   [3+4+4=11]

3. (a) What is the meaning of saying- "the states q_1 and q_4 are ***3-equivalent***", in a DFA?
   (b) Give the complete formal description of a PDA (with notations) and also give its formal definitions (mathematical expressions) for configuration and transition.
   (c) Give Pumping Theorem for CFLs and show that $a^n b^n c^n$ is not a CFL.

   [2+4+4=10]

4. Brief on the following.
   (i) The *configuration* of a DFA with formal expression.
   (ii) Working of a *Mealy* Machine.
   (iii) *Shift-reduce conflict* in Bottom-up parsers, and heuristics for resolving it.
   (iv) Give operations under which CFLs and RLs are not closed, (write separately)

   [2+2+2+2=8]

==========

PART-B (40 Marks)

==========

1. (a) Give a one-tape combined TM (diagram) for deciding $L=\{=\{ a^n b^{2n} c^n$ where $n \geq 1\}$. [8]

   (b) Construct a standard 1-tape TM (give formal description with set of transitions, not diagram) that writes alternate symbols (2, 4, 6,..) of the input string, after the first blank appears after the string. Assume no blanks in the input string. Initial position of the RW head is at the blank that is in the immediate cell to the right of the left-end marker. The string follows the first blank. Do not alter initial input string. [7]

2. (a) Give a combined two-tape TM (diagram) to decide a language over {a,b} with equal number of ***a***s and ***b***s. Assume no blanks in the input string. Initial position of the RW head is at the blank that is in the immediate cell to the right of the left-end marker. The string follows the first blank.

[6]

(b) Give a formal description of a 2-tape TM with mathematical expression for its configuration.

[4]

3. Using the given (below) set of instructions, write a program for a RATM M, that computes the sum of the content of tape locations staring from index 10 to 20, which contain non-zero positive integers. The final result (sum) needs to be written to location with index 50.
Give the formal description of your Machine.

[7]

Instruction	Operand	Semantics
read	j	$R_0 := T[R_j]$
write	j	$T[R_j] := R_0$
store	j	$R_j := R_0$
load	j	$R_0 := R_j$
load	$= c$	$R_0 := c$
add	j	$R_0 := R_0 + R_j$
add	$= c$	$R_0 := R_0 + c$
sub	j	$R_0 := \max\{R_0 - R_j, 0\}$
sub	$= c$	$R_0 := \max\{R_0 - c, 0\}$
half		$R_0 := \lfloor \frac{R_0}{2} \rfloor$
jump	s	$\kappa := s$
jpos	s	if $R_0 > 0$ then $\kappa := s$
jzero	s	if $R_0 = 0$ then $\kappa := s$
halt		$\kappa := 0$

4. Brief on the following.
(i) *Syntax analysis* step in compilation process
(ii) *Greibach Normal Form*
(iii) Reason for appending a ' $ ' symbol to the input-string in parsing using PDA
(iv) Give formal description (with transitions) for left- head moving TM ***L***.

[2X4=8]

**** *THE END*****