BITS Pilani - Dubai Campus
Knowledge Village, Dubai
I – Semester 2005-2006

Course Number	:	CS UC 351
Course Name	:	Theory of Computation
Nature of Component	:	**Comprehensive Examination**
Weightage	:	40%
Max. Marks	:	40 Marks
Duration	:	180 minutes
Date of Examination	:	04.01.2006

Number of Pages: 3

Number of Questions: 11

Note: 1) Please follow all the instructions to candidates given on the cover page of the answer book.
2) All parts of the question should be answered consecutively. Each answer should start from a fresh page.

1. Give the Non-Deterministic Finite Automata design for the following language
$L = (ab \cup aba)^*$ (4 Marks)

2. Give the transition diagram and the transition equation for the Transition table given below:

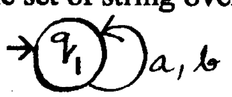
State / Sigma	0	1
→ [q0, q4]	[q1, q7]	[q3, q5]
[q1, q7]	[q6]	[q2]
* [q2]	[q0, q4]	[q2]
[q3, q5]	[q2]	[q6]
[q6]	[q6]	[q0, q4]

Where s = [q0, q4], and F = [q2]. (3 Marks)

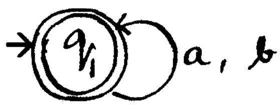
3. Show that the Moore Finite Automata to accept the words "cat", "cob", "cab", "cot" but no others from the alphabet L = { a,b,c,t,o }. (3 Marks)

4. Prove by mathematical induction that $6^{n+2} + 7^{2n+1}$ is divisible by 43 for each positive integer n. (3 Marks)

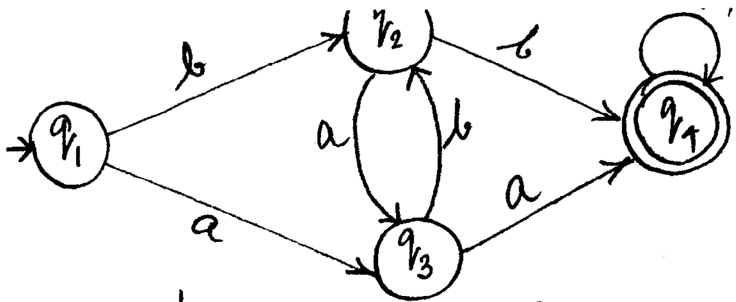
5. Find the set of string over S={a,b} recognized by the transition systems shown below:

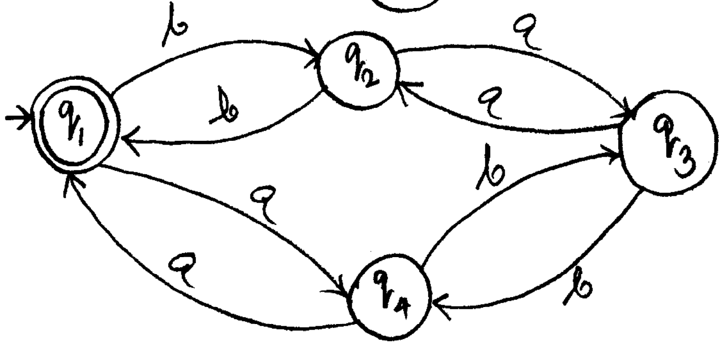
(a) 

(½ Mark)

(b) 

(½ Mark)

(c)  (1 ½ Mark)

(d)  (2 Marks)

6. State that the below give statement are true or false. (2 ½ Marks)
   (a) If Composite = {(N) | N is a composite integer} is not NP-complete, then P=/ NP.
   (b) If a function f: A→B is a mapping reduction from A to B and is one-to-one, then B is mapping-reducible to A as well.
   (c) If A and B are in NP, then so is A∪B.
   (d) If A and B are NP-Complete, then so is A∩B.
   (e) The set of problem to which A_{TM} is mapping-reducible is uncountable (note that such problems are necessarily un-decidable).
7. Consider a grammar, G = (V, S, R, E), where V, S, and R are as follows. (5 marks)

   V = { x, 1, 2, +, *, (,), T, F, E}

   S = { x, 1, 2, +, *, (,) }

   R = { E→E+T,
   E→T,
   T→T*F,
   T→F,
   F→(E)
   F→x1,
   F→x2 }.

   Check whether the grammar G generates the string given below: (x1*x2+x1)*(x1+x2).
8. Construct a Turing Machine M, that accept the Language L={w∈Σ_0^* : w has equal number of a's , b's and c's} and Σ_0={a,b,c}. (5 Marks)

9. If M be the Turing Machine, M=(K, Σ, δ, s), where K={q_0, q_1, q_2, q_3, q_4}, Σ={a,b,#}, s=q_0 and δ is given by the following table: (3 Marks)

q	σ	δ(q,p)
q_0	a	(q_2, R)
q_0	b	(q_3, a)
q_0	#	(h, #)
q_1	a	(q_2, R)
q_1	b	(q_2, R)
q_1	#	(q_2, R)
q_2	a	(q_1, b)
q_2	b	(q_3, a)
q_2	#	(h, #)
q_3	a	(q_4, R)
q_3	b	(q_4, R)
q_3	#	(q_4, R)
q_4	a	(q_2, R)
q_4	b	(q_4, R)
q_4	#	(h, #)

Start from the configuration (q_0, $\underline{a}$aabbbaa).

10. Construct a FA with reduced states equivalent to the regular expression:

10+ (0+11)0*1 (3 Marks)

11. Design a Finite State Machine for the following problem definition: Consider the behavior and associated events of a gate at a train crossing. Suppose that there is a road crossing railway track, and a gate that opens and closes over the road. When a train approaches the crossing, the gate should close. More than one train can be crossing area at once, for example, a convoy of trains, each with a single engine and no cars. When the last train has left and the area is empty of trains, the gate should open. The gate could be in one of four states: **open, closed, opening,** and **closing**. The relevant events are: **cg** and **og** which are commands to close and open the gate, respectively; and **o-o** and **c-c** indicating from sensor input that the gate has completed opening (and thus changed from the opening to the opened state), and that the gate has completed closing, respectively. The initial state is closed and there are no stop states. The Finite state machine accepts the **cg** and **og** commands in all its states; thus, for example, the gate may be commanded to close while it is opening. (4 Marks)

BITS, PILANI-DUBAI CAMPUS, DUBAI.

FIRST SEMESTER 2005-2006

TEST – II

III Year (Computer Science)

(Only Prescribed Text Book and Class notes are allowed)

Course Name & Number	:	**CS UC 351 – Theory of Computation**
Nature of Component	:	***Open Book***
Max. Marks	:	20
Weightage	:	20%
Duration	:	50 mins
Date	:	**27.11.2005,** Sun.

Note: 1) Please follow all the instructions to candidates given on the cover page of the answer book.
2) All parts of the question should be answered consecutively. Each answer should start from a fresh page.

1. Give the Top-Down pda parsing for the arithmetic expression **n3*(m2),** if the CF grammar rules are : E→E+T, E→ T, T → T * F, T → F, F→ (E), F→ n3, F→ m2; and the pda transition obtained from the above grammar is R1: (p, z0, e)= { (p, z0)},
R2: (p, e, T+E)={ (p, E)}, R3: (p, e, T)={(p, E)}, R4: (p, e, F*T)={(p, T)},
R5: (p, e, F)={ (p, T)}, R6: (p, e,)E( ) ={(p, F)}, R7: (p, e, 3n)= { (p, F)},
R8: (p, e, 2m) = {(p, F)}, R9: (p, e, E) = {(q, e)}. (5 marks)

2. Let L = { $a^m b^n$ | n<m }. Construct (i) a context-free grammar accepting L, (ii) a pda accepting L by empty store. (5 marks)

3. Construct a pda from the CFG production rules: S→SS | (S) | Λ (where Λ represents empty or null value) and check whether the generated pda accept the given string **(( ))( )** (5 marks)

4. Construct a pda accepting L(G), if the CFG is G=({S, A, B}, {a, b}, P, S) where P consist of S→aAB, S→bBA, A→bS, A→a, B→aS, B→b. check whether the string **abbbab** is accepted by the CFG, and pda ? (5 marks)

---oOo---

A

BITS, PILANI – DUBAI CAMPUS
KNOWLEDGE VILLAGE, DUBAI
CS UC 351, THEORY OF COMPUTATION
QUIZ (CLOSED BOOK)

MAXIMUM MARKS: 10 | **I – Semester 2005-2006** | **DATE: 20.10.2005**

Time: 30 MINUTES

Name: | **Id. No.:**

ANSWER ALL QUESTIONS:

1. We denote the length of a string *w* by |w|; thus |101| = _____ and | e | = ___________.

   a) 1 and 1, b) 2 and 1 c) 3 and 0 d) 3 and 1 (½ mark)

2. Draw Non-deterministic Finite Automata with the specified number of states to accept. The language 0*1*0*0 (3 states). (1 ½ marks)

3. Let M = (K, Σ, Δ,s, F) be an NFA , where
   K = {q0,q1,q2}, Σ = {a,b,c,d},
   Δ ={(q0,a,q0),(q0,e,q1),(q1,b,q1), (q1,c,q1),(q1,e,q2),(q2,d,q2)}
   S = q0, F = {q0,q1,q2}.
   The regular language corresponds to the language accepted by M is : (1 mark)

   i) (abcd) ii)a*b*c*d* iii)(a b u c d)* iv) a*(b c)*d* v) none of above

4. A deterministic finite automaton is a simple Language Recognition Device

   i) Yes ii) No (½ mark)

5. The tabular representation of the transition function can be represented in a graphical model or diagram, called as: (½ mark)

   i) Device Diagram ii) Machine Diagram iii) State Diagram iv) Standard diagram.

6. A Non-Deterministic Finite automaton can be much more convenient device to design than a Deterministic finite automata. (½ mark)

   i) Yes ii) No

7. L (((a U b) * a)) = { w ∈ { a, b}* : w ends with a }. (True / False)

(½ mark)

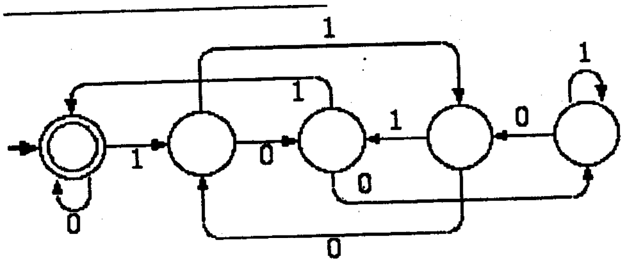
8. Finite automata M1 and M2 are said to be equivalent if and only if ________. (½ mark)

a) L(M1) = A, b) L(M1)={w | w ends in a 1}, c) L(M1) ∩ λ(M1), d) L(M1) = L(M2)

9. A context-free grammar G is (V,Σ, P, S) , where P (the set of rules) is a finite subset of (V-Σ) x V*. The members of (V-Σ) are called ________________. (½ mark)

a) Variables, b) start variables c) non-terminals, d) terminals.

10. The transition diagram given below is a DFA accepting string of ________________ ________________ (1 mark)



11. Let the alphabet S be the standard 26 letters {a,b,..,z}. If A = { good, bad } and B = { boy, girl }, then , what is **A o B** = ________________________

(½ mark)

12. Draw a generic One-State Automaton for a regular expression denoting the strings that it accepts is R*. (½ mark)

13. If E and F are regular expressions, then E + F is a regular expression denoting the union of L(E) and L(F). (True / False) (½ mark)

14. Test whether 001100, 001010, and 01010 are in the language generated by the grammar S -> 0S1 | 0A | 0 | 1B | 1, A -> 0A | 0, B -> 1B | 1. (1 ½ marks)

...o0o...

B

, PILANI – DUBAI CAMPUS
KNOWLEDGE VILLAGE, DUBAI
CS UC 351, THEORY OF COMPUTATION
QUIZ (CLOSED BOOK)
I – Semester 2005-2006

MAXIMUM MARKS: 10 **DATE: 20.10.2005**
Time: 30 MINUTES
Name: **Id. No.:**

ANSWER ALL QUESTIONS:

1. Let the alphabet S be the standard 26 letters {a,b,...,z}. If A = { good, bad } and B = { boy, girl }, then , what is **A o B** = ______________________ (½ mark)

2. If E and F are regular expressions, then E + F is a regular expression denoting the union of L(E) and L(F). (True / False) (½ mark)

3. Draw a generic One-State Automaton for a regular expression denoting the strings that it accepts is R*. (½ mark)

4. Test whether 001100, 001010, and 01010 are in the language generated by the grammar S -> 0S1 | 0A | 0 | 1B | 1, A -> 0A | 0, B -> 1B | 1. (1 ½ marks)

5. L (((a U b) * a)) = { w ∈ { a, b}* : w ends with a }. (True / False) (½ mark)

6. A context-free grammar G is (V,Σ, P, S) , where P (the set of rules) is a finite subset of (V-Σ) x V*. The members of (V-Σ) are called ______________________. (½ mark)

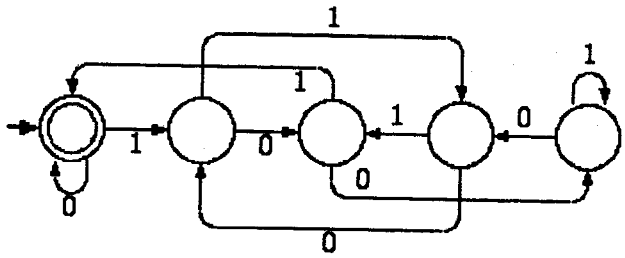
   a) Variables, b) start variables c) non-terminals, d) terminals.

7. Finite automata M1 and M2 are said to be equivalent if and only if _______. (½ mark)

a) L(M1) = A, b) L(M1)={w | w ends in a 1}, c) L(M1) ∩ λ(M1), d) L(M1) = L(M2)

8. The transition diagram given below is a DFA accepting string of ____________________

____________________ (1 mark)



9. We denote the length of a string *w* by |w|; thus |101| = _____ and | e | = __________.

a) 1 and 1, b) 2 and 1 c) 3 and 0 d) 3 and 1 (½ mark)

10. Let M = (K, Σ, Δ,s, F) be an NFA, where
K = {q0,q1,q2}, Σ = {a,b,c,d},
Δ ={(q0,a,q0),(q0,e,q1),(q1,b,q1), (q1,c,q1),(q1,e,q2),(q2,d,q2)}
S = q0, F = {q0,q1,q2}.
The regular language corresponds to the language accepted by M is : (1 mark)

i) (abcd) ii)a*b*c*d* iii)(a b u c d)* iv) a*(b c)*d* v) none of above

11. Draw Non-deterministic Finite Automata with the specified number of states to accept. The language 0*1*0*0 (3 states). (1 ½ marks)

12. A deterministic finite automaton is a simple Language Recognition Device

i) Yes ii) No (½ mark)

13. A Non-Deterministic Finite automaton can be much more convenient device to design than a Deterministic finite automata. (½ mark)

i) Yes ii) No

14. The tabular representation of the transition function can be represented in a graphical model or diagram, called as: (½ mark)

i) Device Diagram ii) Machine Diagram iii) State Diagram iv) Standard diagram.

BITS, Pilani – Dubai Campus
Knowledge Village, Dubai.
I – Semester 2005-2006
TEST- I

Course: **CS UC 351 Theory of Computation** Date: 09.10.2005
Duration: 50 mins **III Year – Computer Science**
Weightage : 20 % Max. Marks: 20

Closed Book

Note:- 1) Please follow all the instructions to candidates given on the cover page of the answer book.
2) All parts of the question should be answered consecutively. Each answer should start from a fresh page.

1. If R is the set of ordered pair of elements, R = { (1,2), (2,3), (1,4), (4,2), (3,4) }. Find R^+ and R^*? (2)
2. Draw the transition diagram and the transition table for the following problem definition given below: Suppose we have a vending machine that dispenses a can of soda when we deposit 25 cents. The coins we may insert are nickels, dimes and quarters, and the machine will count to see when we reach 25 cents. In our simple model of this machine, no change will be given, and the machine will keep any extra money. You can omitted the arrows for coin combinations that total more than 25 cents, but they could be added for each state, with all of them terminating in the accept state. Also, we could add arrows for inputs other than nickels, dimes, and quarters (call them all slugs). The arrows for such input in each state would loop back to the same state. The formal definition of this machine would be given as the 5-tuple (Q, Σ, δ, 0, {25})
where Q = {0,5,10,15,20,25}
Σ = {n,d,q,s}, where
(n = nickels(its value is 5), d = dimes(its value is 10), q = quarter(its value is 25 and s = slug (other than 5, 10, 25)) (4)
3. Design a DFA that accepts the language over {a, b, c} that has three consecutive b's. As we think about this, we realize we need to count the number of **b**'s that we see in a row. We could have none, one, two, or three. In any of these cases, seeing a b takes us to the next state (but not beyond three). Seeing either **a** or **c** causes us to reset to the none state. Seeing anything once we have seen three **b**'s doesn't change anything since we only care about finding at least three consecutive **b**'s. Also give the transition table for the above problem. (3)
4. If G = ({S, C}, {a,b}, P, S), where P c consists of S→aCa, C→aCa | b. Find L(G)? (2)
5. Construct a grammar G generating $\{a^n b^n c^n \mid n>=1\}$? (3)
6. Give the transition system equivalent to regular expression (3)
$(0 + 1)^* (00 + 11) (0 + 1)^*$
7. Consider the following production S→aB | bA, A→aS | bAA | a, B→bS | aBB | b
For the string aaabbabbba, find (3)
   (i) the leftmost derivation
   (ii) the rightmost derivation
   (iii) Parse tree.

()