राष्ट्रीय प्रौद्योगिकी संस्थान जमशेदपुर

NATIONAL INSTITUTE OF TECHNOLOGY JAMSHEDPUR

(An Institution of National Importance under MHRD, Government of India)

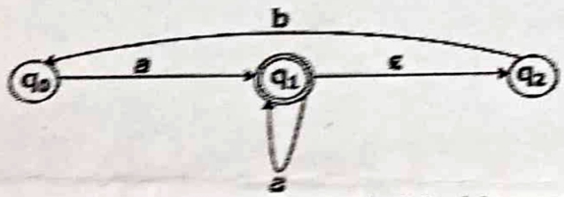
Department of Computer Science and Engineering

Semester: Spring 2023-24
Course Title: Formal Language and Automata Theory
Full Marks: 30
Course Instructor: Dr. Mayukh Sarkar

Examination: Mid Semester
Course Code: CA3405
Duration: 2 Hours
Semester: MCA 4th Semester

(Answer All Questions)

1. **(a)** Given an ϵ-NFA $M_1 = (Q, \Sigma, \delta, q_0, F)$ where $Q = \{q_0, q_1, q_2\}$, $\Sigma = \{a, b, \epsilon\}$, $F = \{q1\}$, and δ as shown in the accompanying image, determine and draw the DFA that accepts the same language as M_1. **[3]**

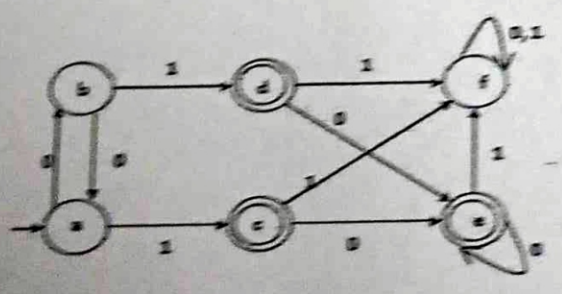


(b) Demonstrate the mathematical definitions of Mealy Machine and Moore Machine with suitable examples. **[2]**

2. **(a)** Given the alphabet $\Sigma = \{a, b\}$, determine and draw an ϵ-NFA, step by step following Kleene construction, that will accept the regular expression $(a+b)^*ab$. **[3]**

(b) Using pumping lemma for regular languages, show that the language $L = \left\{0^{2^i} \middle| \, i \text{ is an integer}, i \geq 1\right\}$ over alphabet $\Sigma = \{0\}$, is not regular. **[2]**

3. **(a)** Consider the following DFA. The language accepted by this DFA is union of some equivalence classes of an equivalence relation. Find all the equivalence classes of the relation and determine the classes whose union will be the language. **[3]**



(b) Given any two finite automata $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ and $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$. How will you construct a finite automaton that will accept the intersection of the two languages accepted by M_1 and M_2 respectively. **[2]**

(In all questions below, small letters represent terminals whereas capital letters represent variables)

4. (a) Convert the following grammar into Chomsky Normal Form. [3]

S → baAB
A → bAB | ϵ
B → BAa | A | ϵ

(b) Suppose, a context-free grammar G=(V,T,P,S) has already been given in Greibach Normal Form. Briefly elaborate how will you convert G into an equivalent grammar with each production of the form A → a, A →aB, and A → aBC only. [2]

5. (a) Given the grammar G = ({S}, {a, b}, {$S \rightarrow aSbb \mid a$}, S), design a pushdown automata M that will accept the same language generated by G, by emptying its stack. With the example of string *aabb*, show that G generates the string and M accepts the same string by emptying its stack (demonstrate this by showing sequence of Instantaneous Descriptions). [3]

(b) Given a pushdown automata M1 that accepts some language L by emptying its stack. How will you construct a pushdown automata M2 that will accept the same language L by entering some final state? Elaborate with a suitable example. [2]

6. (a) State the pumping lemma for context-free languages. Using the lemma, show that the language L = {$a^i b^i c^i \mid i \geq 1$} cannot be a context-free language. [3]

(b) Construct a finite automaton that will accept the same language generated by the following grammar G = ({S,A,B}, {a,b}, P, S) where P is demonstrated below. [2]

S → abS | A
A → baB
B → aA | bb

राष्ट्रीय प्रौद्योगिकी संस्थान जमशेदपुर

NATIONAL INSTITUTE OF TECHNOLOGY JAMSHEDPUR

(An Institution of National Importance under MHRD, Government of India)

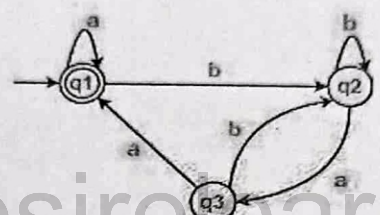
Department of Computer Science and Engineering

Semester: Spring 2023-24
Course Title: Formal Language and Automata Theory
Full Marks: 50
Course Instructor: Dr. Mayukh Sarkar

Examination: End Semester
Course Code: CA3405
Duration: 3 Hours
Semester: MCA 4th Semester

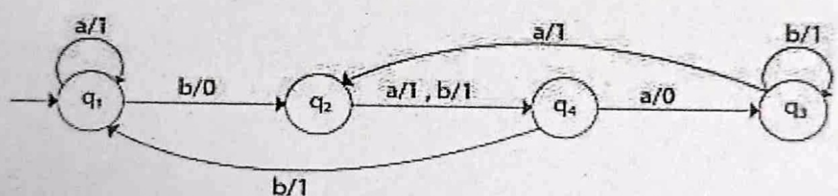
(Answer All Questions)

1. (a) Given the following DFA, determine the language accepted by the automata, using Arden's Lemma. [3]



(b) Show that, determining whether two finite automata are equivalent or not, is decidable. [2]

2. (a) Convert the following Mealy machine into an equivalent Moore machine. [3]



(b) Given any two finite automata $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ and $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$. How will you construct a finite automaton that will accept the union of the two languages accepted by M_1 and M_2 respectively. [2]

3. (a) Given, $L = \{w | w \in 0^* \text{and } |w| \text{ is divisible by } 3 \text{ or } 5\}$. Show that, either L is not regular, or if L is regular, design the finite automata accepting L. [3]
(b) Let G be the grammar ({S}, {a,b}, {S → aS | aSbS | ϵ}, S). Show that the language accepted by grammar is L = {x | each prefix of x has at least as many a's as b's}. [2]

4. (a) Given the grammar G = ({S}, {~, [,],&,p,q}, {S → ~S | [S & S] | p | q}, S), obtain its equivalent Chomsky Normal Form. [2]
(b) State Ogden's Lemma. "If a language can be proved as non-CFL using Ogden's lemma, that can be proved as non-CFL using pumping lemma, but the vice-versa is not true" – Disprove or justify the statement. [3]

5. (a) State the mathematical definition of Pushdown Automata with suitable example. [2]
(b) Consider the grammar ({S,A,B,C}, {a,b}, {$S \rightarrow AB \mid BC$, $A \rightarrow BA \mid a$, $B \rightarrow CC \mid b$, $C \rightarrow AB \mid a$}, S). Determine the membership of the string *baaba* using Cocke-Younger-Kasami algorithm. [3]

6. (a) Given a deterministic Turing Machine M = (Q, Σ, Γ, δ, q, B, F) that solves a computational problem in T steps. Show that, restricting the tape alphabet to {0,1} will allow the TM to solve the same problem in $O(T \log |\Gamma|)$ steps. [2]
(b) Show that, a two-way infinite tape Turing Machine is equivalent to a one-way infinite tape Turing Machine. [3]

7. Show that, the set of infinite binary sequence is uncountably infinite. Using this fact, show that some languages are not Turing-recognizable. [2+3]

8. (a) Show that the language L = {<M, w> | M is a Turing Machine, w is a string in $\{0,1\}^*$ and M accepts w} is Turing-undecidable. [3]
(b) Show that a language L is Turing-decidable if and only if L and L' are Turing-recognizable, where L' represents the complement of L. [2]

9. (a) Define polynomial time reducibility. Using reducibility show that the Halting problem is Turing-undecidable. [3]

(b) Demonstrate SAT problem with a suitable example. [2]

10. (a) Show that, a problem is in NP if and only if it can be verified by a deterministic Turing Machine in polynomial time. polynomial time [3]
(b) Show that, if an NP-hard problem A is reducible to problem B, then B is NP-hard. [2]