First Internal Test-Answer Key

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Sub: | Automata Theory and Computability | | | | | | | Code: | **15CS54** |
| Date: | 08 / 09 / 2018 | Duration: | 90 mins | Max Marks: | 50 | Sem: | V | Branch: | **ISE** |
| Answer ALL Questions | | | | | | | | | |

| | | Marks | OBE CO | OBE RBT |
|---|---|---|---|---|
| 1 (a) | List any four applications of Automata Theory. | [04*1=4 marks] | CO1 | L1 |

- Languages enable Machine/machine and person/machine communication
  - Network protocols, HTML, Semantic Web (RDF,OWL), Music
- Programming languages, compilers, & context-free grammars.
- FSMs (finite state machines) for parity checkers, vending machines, communication protocols, & building security devices.
- Interactive games as nondeterministic FSMs.

(b) Find the following Languages are closed or not. Give the reason if it is No. [02 marks for each answer=4marks] CO1 L2

(i) The odd length strings over the alphabet {a, b} under Kleene star.

Not closed because, if two odd length strings are concatenated, the result is of even length. The closure is the set of all nonempty strings drawn from the alphabet {a, b}.

(ii) $L = \{w \in \{a, b\}^*: w$ ends in a$\}$ under concatenation. CO1 L2

Closed.

(c) Explain with examples about the concatenation of languages. [02]

Definition: 1 mark
Example:1 mark

If $L_1$ and $L_2$ are languages over Σ:

$L_1L_2 = \{w : \exists s \in L_1 \ \& \ \exists t \in L_2 \ni w = st \}$

Examples:

$L_1$ = {cat, dog}

$L_2$ = {apple, pear}

$L_1 L_2$ = {catapple, catpear, dogapple, dogpear}

$L_2 L_1$ ={applecat,appledog,pearcat,peardog}

2 (a) For each of the following statements say it is true or false. Prove your answer [04] CO1 L3

Answer: 1 mark
Derivation:1 mark

(i) $(\varnothing \cup \varnothing^*) \cap (\neg\varnothing - (\varnothing\varnothing^*)) = \varnothing$ (where $\neg\varnothing$ is the complement of $\varnothing$).

False. The left hand side equals {ε}, which is not equal to ∅.

$\varnothing=\{\ \}$, $\varnothing^*=\{\varepsilon\}$,

$(\varnothing \cup \varnothing^*)=\{\varepsilon\}$,

$\neg\varnothing = \Sigma^*$,

$\varnothing\varnothing^*=\varnothing$

$(\neg\varnothing - (\varnothing\varnothing^*))=\{\varepsilon\}$,

$(\varnothing \cup \varnothing^*) \cap (\neg\varnothing - (\varnothing\varnothing^*)) =\{\varepsilon\}$ not equal to ∅

(ii) $\forall L\ (\varnothing L^* = \{\varepsilon\})$.

False. For any L, and thus for any L*, ∅L = ∅.

(b) Compare DFSM with NDFSM [04] CO2 L2 CO1 L2

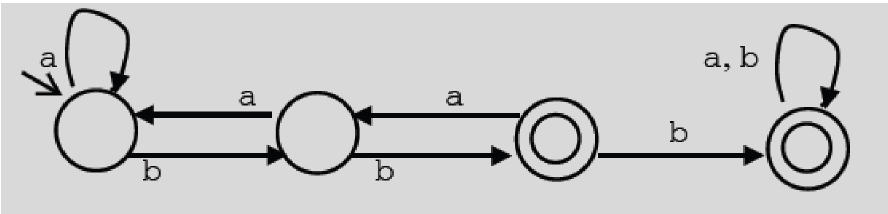
Each point 1 mark

| DFSM | NDFSM |
|---|---|
| • One transition for each input symbol at any state | • Multiple transitions |

| ε transition is not here | ε transition is there |
|---|---|
| Transition function KXΣ -> K states | Transition relation<br>KXΣ -> $2^K$ states |
| Single path for a string to accept | There exists multiple path for a string. Any path leads to accept is accepted by NDFSM |

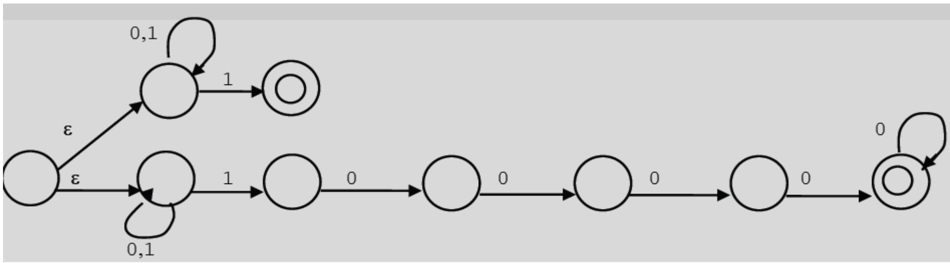
(c) Explain Non determinism with real time example.
Travel plan from source destination by multiple ways: explanation should be given. [02]

3 (a) Design DFSM for L = {w ∈ {a, b}* : w contains at least two b's that are not immediately followed by a's}. [05] CO3 L2



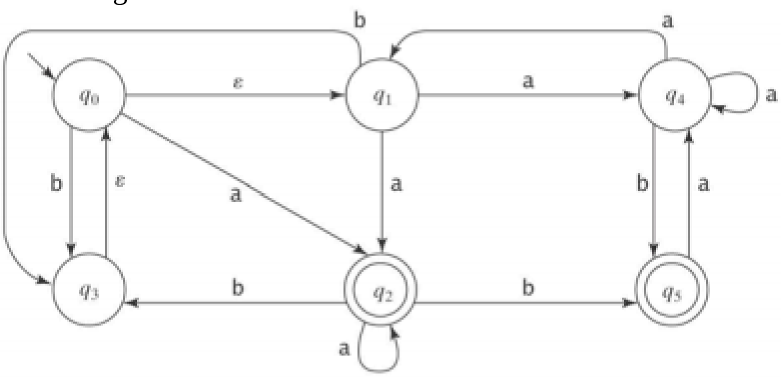

States correct: 2 marks
Transition Correct: 3 marks

(b) Design NDFSM for the L={w ∈ {0, 1}* : w corresponds to the binary encoding of a positive integer that is divisible by 16 or is odd}. [05] CO3 L2




States correct: 2 marks
Transition Correct: 3 marks

4 (a) Convert the given NDFSM to DFSM. [10] CO2 L3




Epsilon closure for all states=3 marks
Transition function= 4 marks
Final correct DFSM=3 marks

| *s* | ***eps(s)*** |
|---|---|
| $q_0$ | $\{q_0, q_1\}$ |
| $q_1$ | $\{q_1\}$ |
| $q_2$ | $\{q_2\}$ |
| $q_3$ | $\{q_3, q_0, q_1\}$ |
| $q_4$ | $\{q_4\}$ |
| $q_5$ | $\{q_5\}$ |

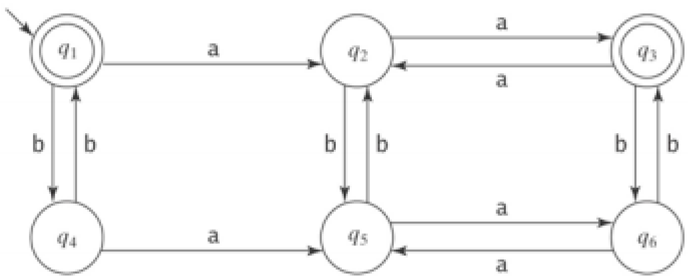
| | | |
|---|---|---|
| $\{q_0, q_1\}$ | a | $\{q_2, q_4\}$ |
| | b | $\{q_0, q_1, q_3\}$ |
| $\{q_2, q_4\}$ | a | $\{q_1, q_2, q_4\}$ |
| | b | $\{q_0, q_1, q_3, q_5\}$ |
| $\{q_0, q_1, q_3\}$ | a | $\{q_2, q_4,\}$ |
| | b | $\{q_0, q_1, q_3\}$ |
| $\{q_1, q_2, q_4\}$ | a | $\{q_1, q_2, q_4\}$ |
| | b | $\{q_0, q_1, q_3, q_5\}$ |
| $\{q_0, q_1, q_3, q_5\}$ | a | $\{q_2, q_4\}$ |
| | b | $\{q_0, q_1, q_3\}$ |

Accepting state is $\{q_0, q_1, q_3, q_5\}$. 1

Diagram should be drawn.

5 (a) Let M be the following DFSM. Minimize M and draw the resultant minimized DFSM M' [7] CO1 L3

Step 1: 4 marks
Step 2: 3 marks




CO1 L1

Initially, *classes* = {[1, 3], [2, 4, 5, 6]}.

At step 1:

((1, a), [2, 4, 5, 6]) ((3, a), [2, 4, 5, 6]) No splitting required here.
((1, b), [2, 4, 5, 6]) ((3, b), [2, 4, 5, 6])

((2, a), [1, 3]) ((4, a), [2, 4, 5, 6]) ((5, a), [2, 4, 5, 6]) ((6, a), [2, 4, 5, 6])
((2, b), [2, 4, 5, 6]) ((4, b), [1, 3]) ((5, b), [2, 4, 5, 6]) ((6, b), [1, 3])

These split into three groups: [2], [4, 6], and [5]. So classes is now {[1, 3], [2], [4, 6], [5]}.

At step 2, we must consider [4, 6]:

((4, a), [5]) ((6, a), [5])
((4, b), [1]) ((6, b), [1])

No further splitting is required. The minimal machine has the states: {[1, 3], [2], [4, 6], [5]}, with transitions as shown above.

(b) What is a Mealy machine? Give an example. [3]

Mealy Machine: Finite state transducer in which the output depends both on the current state and the current input.

Definition: 1 mark
Example: 2 marks

six-tuple $(K, \Sigma, O, \delta, s, A)$, where:

- $K$ is a finite set of states,
- $\Sigma$ is an input alphabet,
- $O$ is an output alphabet,
- $s \in K$ is the start state,
- $A \subseteq$ is the set of accepting states, and
- $\delta$ is the transition function. It is a function from $(K \times \Sigma)$ to $(K \times O^*)$.

A Mealy machine $M$ computes a function $f(w)$ iff, when it reads the input string $w$, its output sequence is $f(w)$.

**Example: Odd parity Bit after reading every 4 bit:**