

RGPV 2015 PYQ

Q. State and explain the properties of transition functions ?

Ans. A transition function is defined on every state for every input symbol.

Transition Function (δ) is defined as $\delta = Q \times \Sigma \rightarrow Q$.

Where,

- Q is set of all states.*
- Σ is set of input symbols.*

Properties of transition functions:

1. Domain and Range

- Domain: The transition function is a function that maps each state in the set Q and each input symbol in the set Σ to a state in the set Q . In other words, the domain of the transition function is $Q \times \Sigma$.
- Range: The range of the transition function is Q .

2. Determinism

The transition function must be deterministic. This means that for each state and input symbol, the transition function must specify a unique next state. In other words, there cannot be two different next states for the same state and input symbol.

3. Totality

The transition function must be total. This means that for every state and every input symbol, the transition function must specify a next state. In other words, there cannot be any state and input symbol pairs that are not defined in the transition function.

4. Consistency with Accepting States

The transition function must be consistent with the set of accepting states. This means that if the DFA is in an accepting state and it reads an input symbol, then it must transition to another accepting state. In other words, it is not possible for the DFA to transition from an accepting state to a non-accepting state.

5. Unique Next State

The transition function must always lead to a unique next state for any given state and input symbol combination. This ensures that the behavior of the DFA is predictable and consistent.

Property 1: $\delta(q, \Lambda) = q$. It means the state of a system can be changed by an input symbol.

Property 2: For all strings w and input symbol a ,

$$\delta(q, aw) = \delta(\delta(q, a), w)$$

$$\delta(q, wa) = \delta(\delta(q, w), a)$$

It means the state after the automaton consumes or reads the first symbol of a string aw and the state after the automaton consumes a prefix of the string wa .

References:

- Introduction to the Theory of Computation” by Michael Sipser.

Related posts:

1. Definition of Deterministic Finite Automata
2. Notations for DFA
3. How do a DFA Process Strings?
4. DFA solved examples
5. Definition Non Deterministic Finite Automata
6. Moore machine
7. Mealy Machine
8. Regular Expression Examples
9. Regular expression
10. Arden’s Law
11. NFA with ϵ -Moves
12. NFA with ϵ to DFA Indirect Method
13. Define Mealy and Moore Machine
14. What is Trap state ?
15. Equivalent of DFA and NFA
16. Mealy to Moore Machine
17. Moore to Mealy machine
18. Difference between Mealy and Moore machine
19. Pushdown Automata
20. Remove ϵ transitions from NFA
21. TOC 1
22. Difference between Mealy and Moore machine
23. RGPV TOC What do you understand by DFA how to represent it

24. What is Regular Expression
25. What is Regular Set in TOC
26. RGPV short note on automata
27. RGPV TOC properties of transition functions
28. RGPV TOC What is Trap state
29. DFA which accept 00 and 11 at the end of a string
30. CFL are not closed under intersection
31. NFA to DFA | RGPV TOC
32. Moore to Mealy | RGPV TOC PYQ
33. DFA accept even 0 and even 1 |RGPV TOC PYQ
34. Short note on automata | RGPV TOC PYQ
35. DFA ending with 00 start with 0 no epsilon | RGPV TOC PYQ
36. DFA ending with 101 | RGPV TOC PYQ
37. Construct DFA for a power n , $n \geq 0$ || RGPV TOC
38. Construct FA divisible by 3 | RGPV TOC PYQ
39. Construct DFA equivalent to NFA | RGPV TOC PYQ
40. RGPV Define Mealy and Moore Machine
41. RGPV TOC Short note on equivalent of DFA and NFA
42. RGPV notes Write short note on NFA
43. Minimization of DFA
44. Construct NFA without ϵ
45. CNF from $S \rightarrow aAD; A \rightarrow aB/bAB; B \rightarrow b, D \rightarrow d$.
46. NFA accepting two consecutive a's or two consecutive b's.
47. Regular expression to CFG
48. Regular expression to Regular grammar
49. Grammar is ambiguous. $S \rightarrow aSbS|bSaS|\epsilon$
50. leftmost and rightmost derivations

51. Construct Moore machine for Mealy machine
52. RGPV TOC PYQs
53. Introduction to Automata Theory