Theory of Computation
Chapter 3:Recursive enumerable Language

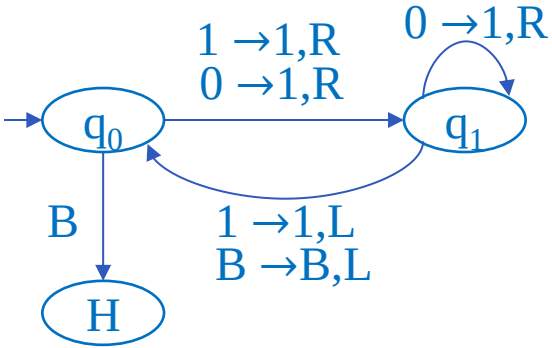
GATE CS Previous year Questions
Chapter wise Solved By

Monalisa Pradhan

- GATE CS 2003,Q53: A single tape Turing Machine M has two states q_0 and q_1, of which q_0 is the starting state. The tape alphabet of M is {0,1,B}and its input alphabet is {0,1}. The symbol B is the blank symbol used to indicate end of an input string. The transition function of M is described in the following table.

	0	**1**	**B**
q_0	q_1,1,R	q_1,1,R	Halt
q_1	q_1,1,R	q_0,1,L	q_0,B,L

- The table is interpreted as illustrated below.
- The entry (q_1,1,R) in row q_0 and column 1 signifies that if M is in state q_0 and reads 1 on the current page square, then it writes 1 on the same tape square, moves its tape head one position to the right and transitions to state q_1.
- Which of the following statements is true about M?
- (A)M does not halt on any string in $(0+1)^+$
- (B)M does not halt on any string in (00+1)*
- (C)M halts on all strings ending in a 0
- (D)M halts on all strings ending in a 1
- {ϵ,0,1,00,01,10,11,…..}
- It halt ϵ,
- For any string except ϵ it will be infinite loop never halt .
- Ans : (A)M does not halt on any string in $(0+1)^+$



- GATE CS 2010,Q17: Let L1 be a recursive language. Let L2 and L3 be languages that are recursively enumerable but not recursive. Which of the following statements is not necessarily true?
- (A) L2 – L1 is recursively enumerable.
- (B) L1 – L3 is recursively enumerable
- (C) L2 ∩ L1 is recursively enumerable
- (D) L2 ∪ L1 is recursively enumerable
- (A)$L2 - L1 = L2 \cap \overline{L1}$, L1 = recursive ⇒ $\overline{L1}$ =recursive.[Recursive is closed under complement]
- So $L2 \cap \overline{L1}$ =RES ∩ RS=RES , L2 − L1 = RES. True
- (B)$L1 - L3 = L1 \cap \overline{L3}$,L3 = Recursive Enumerable ⇒ $\overline{L3}$ ≠ Recursive Enumerable
- [Recursive enumerable is not closed under complement]
- RS ∩ NRES ≠ RES,L1 − L3 ≠ RES .False
- (C) L2 ∩ L1=Recursive enumerable. True.[RES is closed under intersection]
- (D)L2 ∪ L1 =Recursive enumerable. True.[RES is closed under Union]
- **Ans :**(B) L1 – L3 is recursively enumerable

- GATE CS 2013,Q17 : Which of the following statements is/are FALSE?
- 1. For every non-deterministic Turing machine, there exists an equivalent deterministic Turing machine.
- 2. Turing recognizable languages are closed under union and complementation.
- 3. Turing decidable languages are closed under intersection and complementation.
- 4. Turing recognizable languages are closed under union and intersection.
- **(A)** 1 and 4 only **(B)** 1 and 3 only **(C)** 2 only **(D)** 3 only
- 1.True
- 2.Turing recognizable=Recursively enumerable languages which is closed under UNION but they are not closed under complementation. False.
- 3.Turing decidable= Recursive languages which is closed under Intersection and complementation. True
- 4.Turing recognizable=Recursively enumerable languages which is closed under UNION and INTERSECTION. True
- **Ans : (C)** 2 only

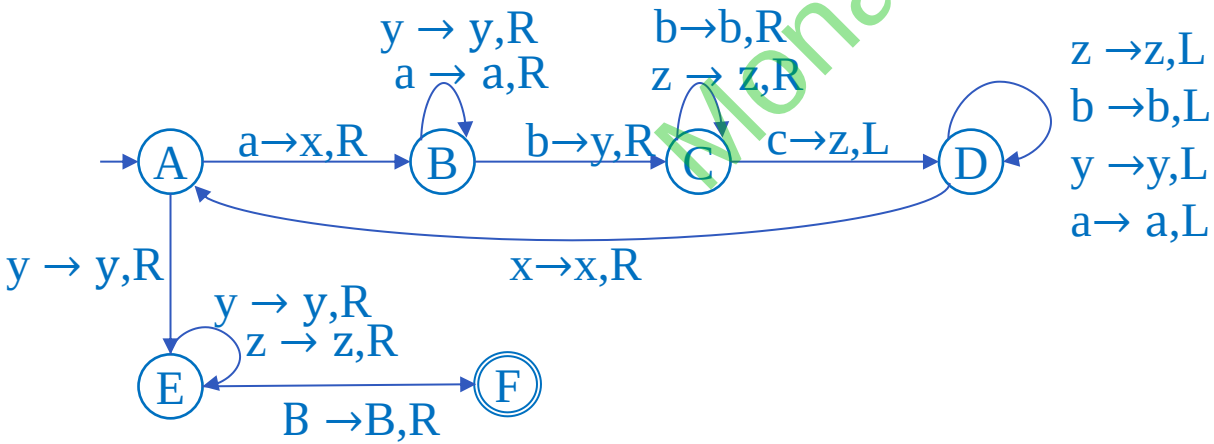
- GATE CS 2014,Set-1,Q35:Let L be a language and $\bar{L}$ be its complement. Which one of the following is NOT a viable possibility?
- (A) Neither L nor $\bar{L}$ is recursively enumerable (r.e.).
- (B) One of L and $\bar{L}$ is r.e. but not recursive; the other is not r.e.
- (C) Both L and $\bar{L}$ are r.e. but not recursive.
- (D) Both L and $\bar{L}$ are recursive
- A) If L is not RE. Then $\bar{L}$ will also not be RE. Possible
- B) If language is RE but not recursive and its complement is NOT RE. Possible
- C) This is not possible as RE is not closed under complement.
- both L and $\bar{L}$ are RE is not possible
- D) It is possible as recursive language is closed under complement .
- Both L and $\bar{L}$ are recursive is possible
- Ans (C) Both L and $\bar{L}$ are r.e. but not recursive.

- GATE CS 2015,Set-1,Q3: For any two languages L_1 and L_2 such that L_1 is context-free and L_2 is recursively enumerable but not recursive, which of the following is/are necessarily true?
- I.$\overline{L_1}$ (Compliment of L_1) is recursive
- II.$\overline{L_2}$(Compliment of L_2) is recursive
- III. $\overline{L_1}$ is context-free
- IV. $\overline{L_1} \cup L_2$ is recursively enumerable
- (A) I only (B)III only (C)III and IV only (D)I and IV only
- L_1=Context-Free , L_2=Recursive Enumerable but not Recursive
- I. $\overline{L_1}$ is not Context Free as CFL is not closed under Compliment but it can be CSL.
- CSL is subset of Recursive , True.
- II. $\overline{L_2}$ is not Recursive Enumerable as REL is not closed under complement, False.
- III. $\overline{L_1}$ is not context free as CFL is not closed under complement ,False
- IV. $\overline{L_1} \cup L_2$ is Recursive Enumerable ,True
- $\overline{L_1}$ ⇒ Recursive ⇒Recursive Enumerable.
- $\overline{L_1} \cup L_2$ ⇒ Recursive enumerable as REL is closed under union.
- Ans: (D)I and IV only

- GATE CS 2016,Set-2,Q18: Consider the following types of languages:L_1 :Regular, L_2: Context -free, L_3: Recursive, L_4: Recursively enumerable. Which of the following is/are TRUE?
- I.$\overline{L_3} \cup L_4$ is recursively enumerable
- II. $\bar{L}_2 \cup L_3$ is recursive
- III. $L_1* \cap L_2$ is context-free
- IV. $L_1 \cup \overline{L_2}$ is context-free
- (A) I only (B) I and III only (C) I and IV only (D) I, II and III only
- I. $\overline{L_3}$ = recursive as recursive languages are closed under complement.
- Recursive ⊂recursive enumerable .So $\overline{L_3}$ is recursive enumerable
- $\overline{L_3} \cup L_4$ is recursive enumerable as REL are closed under union ,True.
- II. $\bar{L}_2 \neq$ CFL as CFL is not closed under compliment but can be CSL.
- CSL ⊂recursive, so $\bar{L}_2$ can be recursive.
- $\bar{L}_2 \cup L_3$ is recursive as recursive languages are closed under union , True.
- III. L_1* is regular as regular languages are closed under kleene closure.
- $L_1 * \cap L_2$ is context free as CFL is closed under regular intersection, True.
- IV. $\overline{L_2} \neq$context free as CFL are not closed under complement.
- $L_1 \cup \overline{L_2}$ may not be context free ,False.
- Ans:(D) I, II and III only

- GATE CS 2016,Set-2,Q44: Consider the following languages.
- L_1 = {⟨M⟩|M takes at least 2016 steps on some input},
  L_2 = {⟨M⟩ | M takes at least 2016 steps on all inputs} and
  L_3 = {⟨M⟩|M accepts ε},
- Where for each Turing machine M, ⟨M⟩ denotes a specific encoding of M. Which one of the following is **TRUE**?
- **(A)** L_1 is recursive and L2, L3 are not recursive **(B)** L2 is recursive and L1, L3 are not recursive
  (C) L1, L2 are recursive and L3 is not recursive **(D)** L1, L2, L3 are recursive
- L_1 is recursive as counting any number of steps can be always decided.
- We can design a TM whether it takes more than 2016 steps on some input string 'w'.
- |w|≤2016.It may halt on accepting state or rejecting state.
- L_2 is recursive.
- We can design TM whether it takes more than 2016 steps on each input string 'w' ,|w|≤2016.
- It may halt on accepting state or rejecting state.
- If L(M) contains ϵ is a non-trivial property and undecidable as per Rice's theorem.
- L_3 is not recursive as emptiness of TM is undecidable.
- Ans: **(C)** L1, L2 are recursive and L3 is not recursive

- $L_1=\{a^nb^nc^n|n\geq 1\}$
- $=\{abc,aabbcc,aaabbbccc,\ldots..\}$
- L_1 = {⟨M⟩|M takes at least 6 steps on some input},
- L_2 = {⟨M⟩ | M takes at least 6 steps on all inputs} and
- L_3 = {⟨M⟩|M accepts ε},



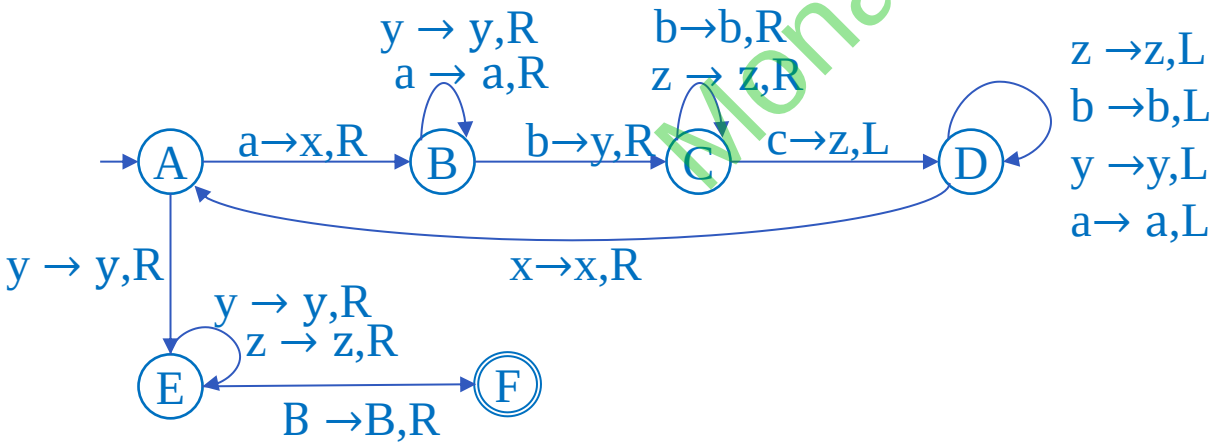
- GATE CS 2017,Set-1,Q39 : Let A and B be finite alphabets and let # be a symbol outside both A and B. Let f be a total functional from A* to B* .We say f is *computable* if there exists a Turning machine M which given an input x in A*, always halts with *f(x)* on its tape. Let L_f denotes the language $\{x\#f(x)|x \in A^*\}$. Which of the following statements is true?
  (A) *f* if computable if and only if L_f is recursive.
  (B) *f* if computable if and only if L_f is recursive enumerable.
  (C) if *f* is computable then L_f is recursive, but not conversely.
  (D) if *f* is computable then L_f is recursively enumerable, but not conversely.
- If *f* is *total* computable function,*f* is computable if and only if L_f is recursive
- If *f* is a partial computable function, f is computable if and only if L_f is r.e.
- Every recursive language is computable, but converse may not be true.
- Ans:**(A)** f if computable if and only if L_f is recursive.

- GATE CS 2018,Q7: The set of all recursively enumerable languages is
- (A) closed under complementation.
- (B) closed under intersection.
- (C) a subset of the set of all recursive languages.
- (D) an uncountable set.
- A. REL are not closed under Complementation.
- B. REL are closed under intersection.
- C. Recursive languages are subset of REL.
- D. REL are a countable set.
- Ans: (B) closed under intersection.

- GATE CS 2019,Q34: Consider the following sets:
- S_1: Set of all recursively enumerable languages over the alphabet {0, 1}.
- S_2: Set of all syntactically valid C programs.
- S_3: Set of all languages over the alphabet {0, 1}.
- S_4: Set of all non-regular languages over the alphabet {0, 1}.
- Which of the above sets are uncountable?
- **(A)** S_1 and S_2 **(B)** S_3 and S_4 **(C)** S_1 and S_4 **(D)** S_2 and S_3
- S_1:Set of all REL are countable.
- S_2:Syntactically valid C program can be represented with CFG. CFG generates CFL, CFL is countable.
- S_3: All languages over {0, 1}= 2^{Σ^*} is uncountable.
- S_4: Set of regular languages are countable, non-regular languages is uncountable.
- Ans: **(B)** S_3 and S_4

- GATE CS 2021,Set-1,Q39: For a Turing machine M, ⟨M⟩ denotes an encoding of M. Consider the following two languages.
- L_1={⟨M⟩ | M takes more than 2021 steps on all inputs}
- L_2={⟨M⟩ | M takes more than 2021 steps on some inputs}
- Which one of the following options is correct?
- (A) Both L_1 and L_2 are decidable.
- (B) L_1 is decidable and L_2 is undecidable .
- (C) L_1 is undecidable and L_2 is decidable .
- (D) Both L_1 and L_2 are undecidable.
- L_1 and L_2 are recursive as counting any number of steps can be always decided.
- L_1 :We can design TM whether it takes more than 2021 steps on each input string ‘w’ ,$|w| \leq 2021$. It may halt on accepting state or rejecting state.
- L_2 :We can design a TM whether it takes more than 2021 steps on some input string ‘w’.$|w| \leq 2021$.It may halt on accepting state or rejecting state.
- Ans:(A) Both L_1 and L_2 are decidable.

- $L=\{a^n b^n c^n | n \geq 1\}$
- $=\{abc, aabbcc, \ldots..\}$
- $L_1=\{\langle M \rangle \mid M \text{ takes more than x steps on all inputs}\}$
- $L_2=\{\langle M \rangle \mid M \text{ takes more than x steps on some inputs}\}$



- GATE CS 2022 | Question: 13 **MSQ**

Which of the following statements is/are TRUE?

- (A)Every subset of a recursively enumerable language is recursive.
- (B)If a language L and its complement $\bar{L}$ are both recursively enumerable, then L must be recursive.
- (C)Complement of a context-free language must be recursive.
- (D)If L_1 and L_2 are regular, then $L_1 \cap L_2$ must be deterministic context-free.
- (A)Every subset of a recursively enumerable language need not be recursive . False.
- (B)Recursive language are closed under complement while recursively enumerable language are not closed under complement .
- Both L & $\bar{L}$ can be Recursive while both L & $\bar{L}$ can not be Recursive Enumerable .
- As L and its complement $\bar{L}$ are both recursively enumerable, so L must be recursive . True.
- (C) $\overline{CFL} \in$ CSL $\subset$ Recursive .True.
- (D)Regular language are closed under intersection
- RL $\cap$ RL=RL $\subset$ DCFL .True.
- Ans: B,C,D

- **GATE CS 2023 | Question: 14 MSQ**
- Which of the following statements is/are CORRECT?

A. The intersection of two regular languages is regular.

B. The intersection of two context-free languages is context-free.

C. The intersection of two recursive languages is recursive.

D. The intersection of two recursively enumerable languages is recursively enumerable.

- (A) Regular ∩ Regular= Regular. So a is correct.
- (B) Context free languages are not closed under intersection and complement.
- So, b is incorrect.
- (C) Recursive ∩ Recursive= Recursive . So, c is correct.
- (D) REL ∩REL=REL , So d is also correct .
- Ans : A,C,D