Theory of Computation
Chapter 4: Undecidability

GATE CS Previous year Questions
Chapter wise Solved By
Monalisa Pradhan

- GATE CS 2012,Q24: Which of the following problems are decidable?
- 1.Does a given program ever produce an output?
- 2.If L is a context-free language, then, is $\overline{L}$ also context-free?
- 3.If L is a regular language, then, is $\overline{L}$ also regular?
- 4.If L is a recursive language, then, is $\overline{L}$ also recursive?
- **(A)** 1, 2, 3, 4 **(B)** 1, 2 **(C)** 2, 3, 4 **(D)** 3, 4
- 1."Does a given program ever produce an output?" = "Does a Turing Machine will halt for any arbitrary string?" = "halting problem of Turing Machine",undecidable.
- 2.Context free languages are not closed under complement.
- If L is CFL then $\overline{L}$ may or may not be CFL,undecidable.
- 3.Regular languages is closed under complement.
- If L is a regular language, then, $\overline{L}$ must also be regular, decidable.
- 4.Recursive languages are closed under complement.
- If L is a recursive language then $\overline{L}$ must also be recursive,decidable.
- **Ans :(D) 3, 4**

- GATE CS 2013,Q41: Which of the following is/are undecidable?
  1. G is a CFG. Is $L(G) = \Phi$?
  2. G is a CFG. Is $L(G) = \sum^*$?
  3. M is a Turing machine. Is $L(M)$ regular?
  4. A is a DFA and N is an NFA. Is $L(A) = L(N)$?

  (A) 3 only (B) 3 and 4 only (C) 1, 2 and 3 only (D) 2 and 3 only
- 1.Emptiness problem for context free language is decidable.
- 2.Completeness problem for context free language is undecidable.
- 3. Regularity in Turing machine is undecidable.
- 4.Language accepted by an NFA and by a DFA is equivalent is decidable.
- We can convert NFA to DFA.
- **Ans :**(D) 2 and 3 only

- GATE CS 2014,Set-2,Q16: Let $A \leq_m B$ denotes that language A is mapping reducible (also known as many-to-one reducible) to language B. Which one of the following is FALSE?
- (A) If $A \leq_m B$ and B is recursive then A is recursive.

  (B) If $A \leq_m B$ and A is undecidable then B is undecidable.

  (C) If $A \leq_m B$ and B is recursively enumerable then A is recursively enumerable.

  (D) If $A \leq_m B$ and B is not recursively enumerable then A is not recursively enumerable.
- A. If B is recursive then A is recursive ,True.
- B. If A is undecidable then B is undecidable, True.
- C. If B is recursively enumerable then A is recursively enumerable,True
- D. B is not recursively enumerable then A is not recursively enumerable.False
- Rule :If A is not recursively enumerable then B is not recursively enumerable.
- Ans: (D) If $A \leq_m B$ and B is not recursively enumerable then A is not recursively enumerable.

- GATE CS 2014,Set-2,Q35: Let <M> be the encoding of a Turing machine as a string over ∑= {0, 1}. Let *L* = { *<M>* |*M is a Turing machine that accepts a string of length 2014* }. Then, L is
- (A) decidable and recursively enumerable
  (B) undecidable but recursively enumerable
  (C) undecidable and not recursively enumerable
  (D) decidable but not recursively enumerable
- There are 2^{2014} number of strings of length '2014'. Which is finite.
- TM may halt in final state.
- But, if TM is unable to accept the input string then it will halt in non-final state or go in an infinite loop and never halt.
- L is undecidable and recursively enumerable .
- Finiteness property of TM is undecidable.
- Ans: (B) undecidable but recursively enumerable

- GATE CS 2014,Set-3,Q35: Which of the following problems is undecidable?

  (A) Deciding if a given context-free grammar is ambiguous.

  (B) Deciding if a given string is generated by a given context-free grammar.

  (C) Deciding if the language generated by a given context-free grammar is empty.

  (D) Deciding if the language generated by a given context-free grammar is finite.
- (A) Ambiguity of CFG is undecidable.
- (B)Membership is decidable.
- By CYK algorithm we can decides whether a given string is generated by a given CFG or not.
- (C)Emptiness is decidable.
- Whether the language generated by a given CFG is empty or not.
- (D)Finiteness is decidable
- Whether the language generated by a given CFG is finite or not.
- Ans: **(A)** Deciding if a given context-free grammar is ambiguous.

- GATE CS 2016,Set-1,Q17: Which of the following decision problems are undecidable?
- I. Given NFAs N_1 and N_2, is $L(N_1) \cap L(N_2) = \Phi$?
  II. Given a CFG $G = (N,\Sigma,P,S)$ and a string $x \in \Sigma^*$, does $x \in L(G)$?
  III. Given CFGs G_1 and G_2, is $L(G_1) = L(G_2)$?
  IV. Given a TM M, is $L(M) = \Phi$?
- (A) I and IV only (B) II and III only (C) III and IV only (D) II and IV only
- I. Design NFA for $L(N_1) \cap L(N_2)$ and decide whether the resulting NFA is for empty language or not. Decidable.
- II. Membership is decidable.
- III. Equality of CFG Undecidable.
- IV. Emptiness property of TM undecidable.
- Ans: (C) III and IV only

- GATE CS 2016,Set-1,Q44: Let X be a recursive language and Y be a recursively enumerable but not recursive language. Let W and Z be two languages such that $\bar{Y}$ reduces to W, and Z reduces to $\bar{X}$ (reduction means the standard many-one reduction). Which one of the following statements is **TRUE**?

  (A) W can be recursively enumerable and Z is recursive.
  (B) W can be recursive and Z is recursively enumerable.
  (C) W is not recursively enumerable and Z is recursive.
  (D) W is not recursively enumerable and Z is not recursive.
- The rules are:If $A \leq B$
- Rule 1: If B is recursive then A is recursive.
- Rule 2: If B is recursively enumerable then A is recursively enumerable.
- Rule 3: If A is not recursively enumerable then B is not recursively enumerable.
- $\bar{Y} \leq W$, $\bar{Y}$ is not REL as REL is not closed under complement.
- W is not recursively enumerable.[rule 3]
- $Z \leq \bar{X}$, $\bar{X}$ is recursive as recursive language is closed under compliment.
- Z is also recursive.[rule 1]
- Ans: (C) W is not recursively enumerable and Z is recursive.

- GATE CS 2017,Set-2,Q41 : Let L(R) be the language represented by regular expression R. Let L(G) be the language generated by a context free grammar G. Let L(M) be the language accepted by a Turing machine M.
- Which of the following decision problems are undecidable?
- I. Given a regular expression R and a string w, is $w \in L(R)$?
  II. Given a context-free grammar G, is $L(G) = \emptyset$?
  III. Given a context-free grammar G, is $L(G) = \Sigma^*$ for some alphabet Σ?
  IV. Given a Turing machine M and a string w, is $w \in L(M)$?
- **(A)** I and IV only **(B)** II and III only **(C)** II, III and IV only **(D)** III and IV only
- I. Membership problem for regular language is decidable.
- II. Emptiness problem for Context free language is decidable.
- III. Completeness problem is undecidable.
- IV. Membership problem for TM is undecidable.
- Ans:**(D)** III and IV only

- GATE CS 2018,Q36 :Consider the following problems. L(G) denotes the language generated by a grammar G. L(M) denotes the language accepted by a machine M.
- (I) For an unrestricted grammar G and a string w, whether w∈L(G)
  (II) Given a Turing machine M, whether L(M) is regular
  (III) Given two grammar G_1 and G_2, whether $L(G_1) = L(G_2)$
  (IV) Given an NFA N, whether there is a deterministic PDA P such that N and P accept the same language
- Which one of the following statement is correct?
- **(A)** Only I and II are undecidable **(B)** Only II is undecidable
  (C) Only II and IV are undecidable **(D)** Only I, II and III are undecidable
- I. Unrestricted grammar=Type 0 grammar.
- Membership property of TM is undecidable .
- II. Regularity of TM is undecidable.
- III. Equality of CFG is Undecidable.
- IV. RL⊂CFL.
- For every regular language we can have a NFA & DPDA
- N & P can accept same language ,Decidable.
- Ans: **(D)** Only I, II and III are undecidable

- GATE CS 2020,Q26: Which of the following languages are undecidable? Note that ⟨M⟩ indicates encoding of the Turing machine M.
- L_1 = { ⟨M⟩ | L(M)=∅ }
- L_2 = { ⟨M,w,q⟩ | M on input w reaches state q in exactly 100 steps }
- L_3 = { ⟨M⟩ | L(M) is not recursive }
- L_4 = { ⟨M⟩ | L(M) contains at least 21 members }
- **(A)** L_1, L_3, and L_4 only **(B)** L_1 and L_3 only **(C)** L_2 and L_3 only **(D)** L_2, L_3, and L_4 only
- L_1:Emptiness problem of TM is undecidable.
- L_2:counting number of steps is decidable as we can run the TM for 100 steps and see if it reaches state q.
- L_3 :Recursiveness is undecidable since there is no algorithm to check whether a given TM accept recursive language.
- L_4 :Membership is undecidable. It may or may not halt. May go to infinite loop.
- Ans : (A) L_1,L_3,L_4 only

- GATE CS 2021,Set-1,Q12: Let ⟨M⟩ denote an encoding of an automaton M. Suppose that $\Sigma=\{0,1\}$. Which of the following languages is/are NOT recursive?
  (A) L = {⟨M⟩ | M is a DFA such that L(M)=∅}
  (B) L = {⟨M⟩ | M is a DFA such that L(M)=Σ*}
  (C) L = {⟨M⟩ | M is a PDA such that L(M)=∅}
  (D) L = {⟨M⟩ | M is a PDA such that L(M)=Σ*}
- Not decidable.
- (A) Emptiness problem of DFA is decidable .
- (B) Completeness problem of DFA is decidable .
- (C) Emptiness problem of PDA is decidable.
- (D) Completeness problem of PDA is undecidable .
- Ans :**(D)** L = {⟨M⟩ | M is a PDA such that L(M)=Σ*}

- GATE CS 2022 | Question: 36 **MSQ**
  Which of the following is/are undecidable?
- (A)Given two Turing machines M_1 and M_2 decide if $L(M_1)=L(M_2)$.
- (B)Given a Turing machine M, decide if $L(M)$ is regular.
- (C)Given a Turing machine M , decide if M accepts all strings.
- (D)Given a Turing machine M, decide if M takes more than 1073 steps on every string.
- Turing machines are undecidable for
- Is $L_1=L_2$?[Equality problem]
- Is L is regular[Regularity problem]
- Is L=Σ* [Completeness problem]
- (A) Equality problem
- (B) Regularity problem
- (C) Completeness problem
- (D) Counting number of steps is decidable as we can run the TM for 1073 steps and see if it reaches final state .
- To monitor TM for 1073+1 steps for all possible inputs of size <=1073.
- If for at least one of the candidate strings, Input TM is halting in <=1073 steps, then we(algorithm) can answer “No” for input M .
- Ans : A, B ,C