Om Shete
TE C3
2103163

19/10/23

Assignment No. 6

Q.1 Write short note on : Recursive and Recursively Enumerable languages.

⇒

- There is a difference between recursively enumerable (Turing acceptable) and recursive (Turing Decidable) language.
- Following statements are equivalent :
  1) The language L is Turing acceptable.
  2) The language L is recursively enumerable
- Following statements are equivalent :
  1) The language L is Turing decidable.
  2) The language L is recursive.
  3) There is an algorithm for recognizing L.
- Every turing decidable language is turing acceptable.
- Every turing acceptable language need not be turing decidable.

• <u>Turning Acceptable Language</u> :

- A language $L \subseteq \Sigma^*$ is said to be a Turing language if there is a turing machine M which halts on every $w \in L$ with an answer 'YES'.
- However, if $w \notin L$, then M may not halt.

• Turing Decidable Language:

- A language $L \subseteq \Sigma^*$ is said to be turing being decidable if there is a turing machine M which always halts on every $w \in \Sigma^*$.
- If $w \in L$ then ~~sto~~ M halts, with answer 'YES', and if $w \notin L$ then M halts with answer 'NO'.

- A set of solutions for any problem defines a language.
- A problem P is said to be decidable / solvable if the language $L \subseteq \Sigma^*$ representing the problem is turing decidable.
- If P is solvable / decidable then there is an algorithm for recognizing L, representing the problem. It may be noted that an algorithm terminates on all inputs.
- Following statements are equivalent:

1) The language L is Turing decidable.
2) the language L is recursive.
3) There is an algorithm for recognizing L.

THADOMAL SHAHANI TSEC ENGINEERING COLLEGE

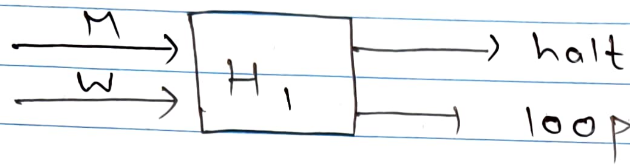
Q.2 What is Halting Problem ? Explain in detail.
⇒

- The halting problem of a turing machine states :
Given a Turing Machine M and an input ω to the machine M, determine if the machine M will eventually halt when it is given input ω.
- Halting problem of a turing machine is unsolvable.
- Proof :

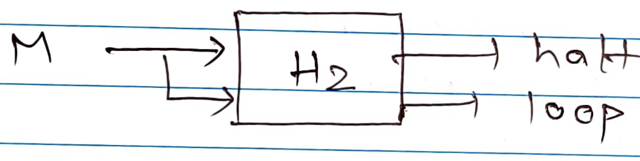
- Moves of a turing machine can be represented using a binary number. Thus, a turing machine can be represented using a string over Σ^* (0, 1). This concept has already been explained in the chapter.
- Insolvability of halting problem of a turing machine can be proved through the method of contradiction.

• Step 1 : Let us assume that the halting problem of a Turing machine is solvable. There exists a machine H_1 (say). H_1 takes two inputs :
  1) A String describing M.
  2) An input ω for machine M.

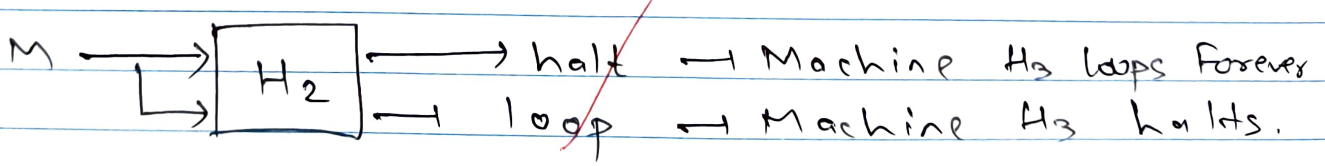
- H_1 generates an output "halt" if H_1 determines that M stops on input w; otherwise H outputs "loop". Working of the machine H_1 is shown below.



- <u>Step 2</u>: Let us revise the machine H_1 as H_2 to take both inputs and H_2 should be able to determine if M will halt on M as its input.

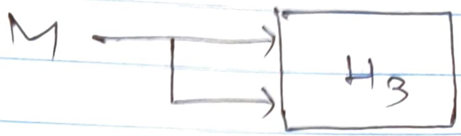


- <u>Step 3</u>: Let us construct a new turing machine H_3 that takes output of H_2 as input and does the following:
  1) If the output of H_2 is "loop" than H_3 halts.
  2) If the output of H_2 is "halt" than H_3 will loop forever.



H_3 will do opposite of output of H_2.

<u>Step 4:</u> Let us give H_3 itself as inputs to H_3.



- If H_3 halts on H_3 as input then H_3 would loop.
- If H_3 loops forever on H_3 as input H_3 halts.
- In either case, the result is wrong.
- Hence, H_3 does not exist.
- If H_3 does not exist then H_2 does not exist.
- If H_2 does not exist then H_1 does not exist.

THADOMAL SHAHANI
TSEC
ENGINEERING COLLEGE

Q.3 Explain detailed explanation on Rice's theorem
⇒
- Every property that is satisfied by some but not all recursively enumerable language is un-decidable.
- Any property that is satisfied by some recursively enumerable language but not all is known as nontrivial property.
- We have seen many properties of R.E languages that are un-decidable. These properties include:

1) Given a TM M, is L(M) nonempty?
2) Given a TM M, is L(M) finite?
3) Given a TM M, is L(M) regular?
4) Given a TM M, is L(M) recursive?

- The rice's theorem can be proved by reducing some other unsolvable problem to nontrivial property of recursively enumerable language.
- Non-trivial Property:
⇒ A property is considered "non-trivial" if it holds for some turing machines but not for others.
- For eg., the property "contains at least one palindrome" is non-trivial because some turing machines recognizes languages with palindromes, while other do not.

THADOMAL SHAHANI
TSEC
ENGINEERING COLLEGE

- No Algorithm

⇒ Rice's theorem asserts that there's no general algorithm that can determine whether an arbitory turing machine recognizes a language with non-specific trivial property.

- Implications

⇒ This theorem has significant consequences. It shows that many interesting question about turing machines' languages are undecidable.

- Limations

⇒ Rice's theorem doesn't simplify that all questions about turing machines are undecidable.

Q.4 Define post correspondance problem. Prove that PCP with two lists $x = \{b, bab^3, ba\}$ and $y = \{b^3, ba, a\}$ have a solution.

⇒

- Let A and B be two non-empty lists of strings over Σ A and B are given as below:

$$A = \{x_1, x_2, x_3, \ldots, x_k\}$$
$$B = \{y_1, y_2, y_3, \ldots, y_k\}$$

- We say, there is a post correspondence between A and B if there is sequence of one or more integers $i, j, k, \ldots m$ such that: The string $x_i x_j \ldots x_m$ is equal to $y_i y_j \ldots y_m$.

THADOMAL SHAHANI TSEC ENGINEERING COLLEGE

- $X = \{b, bab^3, ba)$
  $Y = \{b^3, ba, a)$

$\Rightarrow$

We will have to find a sequence using which when the elements of A and B are listed, will produce identical string.

$\therefore$ The required string is $(2, 1, 1, 3)$.

$\therefore$ $\cancel{X}_2 X_1 X_1 X_3$
$= bab^3 \cdot b \cdot b \cdot ba$
$= b^7 a^2$

$\therefore$ $Y_2 Y_1 Y_1 Y_3$
$= ba \cdot b^3 \cdot b^3 \cdot a$
$= b^7 \cdot a^2$

Thus, the PCP has solution.