Theory of Computation '24 Quiz 4
Time : 40 minutes

Name : **Roll :** **Marks :**

Problem 1. Consider the following classes numbered (1)-(2) as follows

(1) Decidable (2) Recognizable (3) Non-recognizable

For each of the following languages specify the lowest-numbered class to which it surely belongs. For example, for a recognizable language L that is decidable, the right number is 1 but if L is not decidable but recognizable, the right number is 2. (No explanation needed)

a. $L_1 = \{\langle M\rangle \mid M$ is a TM and $L(M)$ is finite $\}$ **(3)** Done in tutorial

b. $L_2 = \{\langle M\rangle \mid M$ is a TM and $L(M)$ is countable $\}$ **(1)** This is the trivially the set of all Turing machines since every language is countable

c. $L_3 = \{\langle M_1, M_2\rangle \mid M_1, M_2$ are TMs and $\varepsilon \in L(M_1) \cap L(M_2)\}$ **(2)** It is easy to see the language is recognizable since you can build a new TM which runs the two TMs in parallel and accepts when both accepts.

   We can show this is undecidable by a mapping reduction from $HALT_{TM}$. Given an instance $\langle M, w\rangle$ of $HALT_{TM}$, construct a single TM M' : "On input x, ignore x and run M on w. Accept if M halts ". Output $\langle M', M'\rangle$.

   Check the rest of the proof by yourself.

d. $L_4 = \{\langle M\rangle \mid \exists x, y \in \Sigma^\star$ such that either $x \in L(M)$ or $y \notin L(M)\}$ **(1)** Again, this the language of all Turing machines trivially and hence decidable.

e. $L_5 = \{\langle M\rangle \mid$ both $L(M), \overline{L(M)}$ are infinite $\}$ **(3)** This is the only tricky one. You can do this by a mapping reduction from $\overline{A_{TM}}$ as follows. Given an instance $\langle M, w\rangle$, construct a new TM M' which works as follows : " On input x, if $|x|$ is odd, accept. Else if $|x|$ is even, simulate $\langle M\rangle$ on w, accept if M accepts w. "

   Let us prove the correctness of the above. Suppose $\langle M, w\rangle \in \overline{A_{TM}}$. This means that M does not accept w and hence this means M' accepts only odd length string. Thus $L(M)$ is infinite (all odd length strings) and $\overline{L(M)}$ is also infinite (all even length strings). Thus $M' \in L_4$

   On the other hand, suppose $\langle M, w\rangle \notin \overline{A_{TM}}$: This means M will accept w. Hence M' will accept all strings and thus $L(M)$ is infinite. But $\overline{L(M)} = \emptyset$ is finite. Hence $M' \notin L_4$.

f. $L_6 = \{\langle M\rangle \mid M$ is a TM and there exists an input on which M halts in less than $\mid \langle M\rangle \mid$ steps$\}$ **(1)**. Here is a decider Turing machine for the language L_6 : "On input $\langle M\rangle$: Simulate M on all possible strings of length at most $|\langle M\rangle|$ for at most $|\langle M\rangle|$ steps. Accept if M accepts on any of the strings." Convince yourself that this works.

Problem 2. State True/False(**T/F**) for each of the following statements (No explanation needed)

a. Any decidable language is mapping reducible to $0^\star 1^\star$ **T**. In fact, any decidable language is mapping reducible to any language L as long as L is neither $\emptyset$ nor $\Sigma^\star$. Languages $\Sigma^\star$ and $\emptyset$ will map to only "YES" and "NO" instances respectively, which are definitely present in $0^\star 1^\star$.

b. There exists some non-trivial decidable subset of every non-empty recognizable language. (A subset is non-trivial if there is some element in the subset and some element not in the subset.) **F** The fact is untrue for a language that contains just one string.

c. Every subset of any language in P is decidable. **F** Consider the language $\Sigma^\star$. This language is clearly in P since the TM for this language simply accepts all strings. But clearly there are subsets of $\Sigma^\star$ which are undecidable.

d. Let $L = \{0^k 1^k \mid k \geq 0\}$. Define another language $L' = \{\langle M \rangle \mid L(M) = L\}$. L' is decidable.**F** The point is the language L itself is decidable (as we have seen several times). But, testing whether a TM accepts this language is still undecidable. The easiest way to see this is to observe that this is a non-trivial property of a TM and hence you can just apply Rice Theorem (those who did not attend tuts can read up Rice Theorem in their own time).

e. Recall the polynomial time reduction from $3SAT$ to $CLIQUE$ done in the lectures. The reduction works just the same for SAT. **T**

f. If $A \leq_p B$ and B is in NP, then A is also in NP **T**