Mid II **Theory of Computation** **Total: 30 Marks**

Name and Roll No.: ____________________

1. Define $L_t = \{(M, w, t)$: M is a turing machine that accepts input w in less than t steps $\}$. Is L_t Turing decidable for each positive integer $t > 0$? Is $L = L_1 \cup L_2 \cup .....$ Turing decidable? Justify your answer. $\boxed{2+2}$

   Soln: L_t is decidable for each positive integer t because a UTM can simulate M on w for t steps and accept if M accepts, reject otherwise. However $L = L_1 \cup L_2 \cup ..... = L_u$, is the unversal language which is undecidable (proved in class).

2. Show that $L = \{a^i b^j c^j d^i : i \geq 0, j \geq 0\}$ satisfies the Chomsky Schutzenberger Theorem conditions and hence is context free. (Specify parenthesis language, regular language and homomorphism). $\boxed{2}$

   Soln: $G : S \longrightarrow (S)|[S]|SS|\epsilon$. $h(L(G) \cap ({}^*[{}^*]^*)^*) = L$ where $h(() = a, h([) = b, h(]) = c, h()) = d$.

3. Let M_1 and M_2 be Turing machines that accept languages L_1 and L_2. Consider the algorithm A that on input w simulates (using UTM strategy) M_1 on w first, and then simulates M_2 on w. A accepts w if at least one of the simulations is accepting. What is the language accepted by A? $\boxed{2}$

   Soln: $L(A) = L(M_1) \cup L(M_2) \setminus \{w : M_1$ does not halt on $w\}$.

4. Let $L_{\overline{u}} = \{(M, w)$: M is a Turing Machine that does not accept input $w\}$. Let $L_\emptyset = \{M : L(M) = \emptyset\}$. Show that $L_{\overline{u}} \preceq_m L_\emptyset\}$. What can you conclude about the deciability and acceptability of $L_\emptyset$? Justify. $\boxed{2+2}$

   Soln: The reduction from L_u to (L_ϵ and L_{Σ^*}) discussed in class serves as a reduction from $L_{\overline{u}}$ to L_ϕ where $L_{\overline{u}} = \{(M, w)$: M does not accept $w\}$. Since $L_{\overline{u}}$ is not Turing acceptable, so is L_ϕ.

5. An instance of the set cover problem (SC) takes a triple (S, T, k) where S is a finite set, T is a collection of subsets of S and k is a positive integer. A YES instance of the problem has the property that you can pick k subsets from T whose union is S. Give a reduction algorithm from the Vertex Cover problem to SC. $\boxed{2}$

   Soln: Given a graph $G = (V, E)$ and integer k as instance of the vertex cover problem, Let $N(v) = \{e \in E : e$ is incident on $v\}$. The reduction algorithm outputs $S = E$ and $T = \{N(v) : v \in V\}$ and k. (For each vertex in V, there is a set in T that corresponds to the edges covered by the vertex. The requirement is to cover all edges by picking k such sets). Prove that G has a vertex cover S of size k if and only if $\{N(v) : v \in S\}$ covers E.

6. Is the language $L = a^*b^*c^* - a^nb^nc^n$ context free? If so, provide a CFG and explain why the CFG generates the language. If not, give a proof that the language is not context free. $\boxed{2}$

   Soln: $L = \{a^ib^jc^k : i \neq j$ OR $j \neq k\}$ We can split L as $L = \{a^*b^ic^j : i > j\} \cup \{a^*b^ic^j : i < j\} \cup \{a^ib^jc^* : i > j\} \cup \{a^ib^jc^* : i < j\}$. Now it is easy to write a CFG for $\{a^*b^ic^j : i > j\} = \{a^*bb^*b^jc^j\}$ ($S \longrightarrow ABT, A \longrightarrow aA|\epsilon, B \longrightarrow bB|b, T \longrightarrow bTc|\epsilon$) etc.