CSI 4105 Midterm 2001 *(Format was compressed to fit 2 pages)*

1. Turing Machines (9 points)

Consider a Turing machine M with input alphabet $\Sigma = \{0, 1\}$ and machine alphabet $\Gamma = \{0, 1, \sqcup\}$, and transition function depicted in the following diagram:

Part A (6 points) Give L, the language *accepted* by M.
Part B (3 points) Does M decide L? Justify.

2. Short answers (21 points) For each of the statements below choose TRUE or FALSE and briefly justify your answer. Note: You may use any result shown in class or in homework without proving it.

- $\mathbf{NP} \cap \mathbf{co\text{-}NP} = \emptyset$
- if Ham-Cycle $\in \mathbf{P}$ then Vertex-Cover $\in \mathbf{P}$
- if Sat $\notin \mathbf{P}$ then Clique $\notin \mathbf{P}$
- Shortest-Path $\in \mathbf{P}$
- if Shortest-Path $\in \mathbf{NPC}$ then $\mathbf{P} = \mathbf{NP}$.
- It is possible that there exists a RAM program that decides a language L_1 in polynomial time, but every Turing machine that decides L_1 runs in exponential time.
- If there exists a polynomial-time algorithm that **accepts** a language L_2, then there exists a polynomial-time algorithm that **decides** L_2.

3. Polynomial-time reducibility (10 points)

Prove that $\leq_P$ relation is a transitive relation on languages. That is, prove that:
if $L_1 \leq_P L_2$ and $L_2 \leq_P L_3$ then $L_1 \leq_P L_3$.
Hint: Use the definition of $\leq_P$.

4. NP-completeness reductions (20 points)

In this question, you are asked to apply some reductions discussed in class to the examples given. For language L_1 and L_2 shown in each part below, we have seen a reduction algorithm that shows that $L_1 \leq_P L_2$; you should apply the algorithms discussed in class.

Part A (5 points) Circuit-Sat $\leq_P$ Sat
Consider the following instance for the circuit satisfiability problem (Circuit-Sat):
circuit C_1:

- Give the corresponding instance for the formula satisfiability problem (Sat), in accordance with the reduction algorithm for Circuit-Sat $\leq_P$ Sat.

- Does $< C_1 > \in$ Circuit-Sat ? Justify.

Part B (5 points) Ham-Cycle $\leq_P$ TSP
Consider the following instance for the hamiltonian cycle problem (Ham-Cycle):

G_1:

- Give the corresponding instance for the traveling salesman problem (TSP), in accordance with the reduction algorithm for Ham-Cycle $\leq_P$ TSP.
- Does $< G_1 > \in$ Ham-Cycle ? Justify.

Part C. (10 points) Vertex-Cover $\leq_P$ Ham-Cycle
Consider the following instance for the vertex-cover problem (Vertex-Cover):

$k = 2$, G_2:

- Give the corresponding instance for the hamiltonian cycle problem (Ham-Cycle), in accordance with the reduction algorithm for Vertex-Cover $\leq_P$ Ham-Cycle.
- Outline the hamiltonian cycle (on the graph you have drawn above) that corresponds to the vertex cover $C = \{v_1, v_2\}$ of G_2.

5.(40 points) **Longest simple cycle is in NPC**

The **longest-simple-cycle** problem is the problem of determining a simple cycle (no repeated vertices) of **maximum** length in a graph. For example:

G_1:	G_2:
longest simple cycle in G_1: `(1,2,3,6)` length of the longest simple cycle: 4	longest simple cycle in G_2: `(1,2,3,5,4)` length of the longest simple cycle: 5

Part A.(2) State the **decision problem** associated with the longest-simple-cycle problem:
Part B. (3) Give the language corresponding to the decision problem given in part A: LSC =
Part C. (35) Prove that LSC is NP-complete, by proving that:

- (10) LSC $\in$ **NP**.
- (25) LSC is NP-hard. (Hint: reduce from Ham-Cycle)