Name ______________________ Student No. ______________________

No aids allowed. Answer all questions on test paper. Use backs of sheets for scratch work. You may use any results presented in lecture.

Total Marks: 40

1. Let $\Sigma = \{0, 1, \#\}$ be a finite alphabet, and let Σ^* be the set of all strings over Σ (as usual in this course). Let $L, L_1, L_2 \subseteq \Sigma^*$ be three languages.

[5] (a) Show that if L is in P, then so is $\overline{L} = \Sigma^* - L$.

If $L \in$ P, then there is a polytime TM M such that $L = L(M)$. Let M' be M except that we interchange the states q_{accept} and q_{reject}. Then $\overline{L} = L(M')$, and since M' is also polytime, $\overline{L} \in$ P.

[5] (b) Suppose that L_1 is in P, and suppose that there are two strings $w_1, w_2 \in \Sigma^*$ such that $w_1 \in L_2$ and $w_2 \notin L_2$. Show that $L_1 \leq_p L_2$.

$L_1 \leq_p L_2$ iff there exists a polytime function f such that $x \in L_1 \iff f(x) \in L_2$, for all $x \in \Sigma^*$. Since $L_1 \in$ P, we can decide in polytime $x \overset{?}{\in} L_1$, and therefore f defined as follows satisfies the above conditions:

$$f(x) = \begin{cases} w_1 & x \in L_1 \\ w_2 & \text{otherwise} \end{cases}$$

2. Recall that the decision problem **Independent Set** (IS) is given as follows:
Input: Graph $G = (V, E)$, positive integer B.
Question: Does G have an independent set of size B?

Also recall that $V_0 \subseteq V$ is an *independent set* of G if no two vertices in V_0 are joined by an edge in E. Let the problem **Odd Independent Set** (OIS) be the same as IS, except that the parameter B must be an odd integer.

Show that IS $\leq_p$ OIS by following the outline below:

[5] (a) Give the reduction.

Let $G = (V, E), B$ be an input instance of IS. We define $G' = (V', E'), B'$, an input instance of OIS as follows:
If B is odd, then just let $G' = G$ (i.e., $V' = V$, $E' = E$) and $B' = B$.
If B is even, then:

$$\begin{aligned} V' &= V \cup \{v_{\text{new}}\} \\ E' &= E \\ B' &= B + 1 \end{aligned}$$

[5] (b) State explicitly the correctness condition of your reduction.

$G = (V, E), B$ is a YES instance of IS iff $G' = (V', E'), B'$ is a YES instance of OIS.

[10] (c) Prove the correctness condition.

("$\Longleftarrow$") Suppose $G = (V, E), B$ is a YES instance of IS. Then there exists a subset $V_0 \subseteq V$ such that $|V_0| = B$, and no two vertices in V_0 are connected by an edge in E. If B is odd, then let $V_0' = V_0$. Otherwise, let $V_0' = V_0 \cup \{v_{\text{new}}\}$. Then V_0' is also an independent set, and $|V_0'| = B + 1 = B'$.

("$\Longrightarrow$") Suppose $G' = (V', E'), B'$ is a YES instance of IS. Let V_0' be an independent set in G' of size B'. We want to show that G has an independent set V_0 of size B. If B is odd, then $B' = B$, so just let $V_0 = V_0'$. If B is even, then V_0' might contain v_{new}, but $B = B' - 1$, so we take v_{new} out of V_0' if it was there, and some other vertex if it wasn't, and let V_0 be the result (since $0 < B$, B is a positive integer, $B' = B + 1 > 1$, so V_0' has at least three vertices—it cannot have two as it has an odd number of vertices).

[10] 3. Recall that the problem **Circuit SAT** is defined as follows:
Input: $\langle C \rangle$ where C is a circuit on n inputs.
Question: Does C output 1 on some $a_1 a_2 \ldots a_n \in \{0,1\}^n$?

Define the related search problem, **Circuit SAT Search**, as follows:
Input: $\langle C \rangle$ where C is a circuit on n inputs.
Output: A string $a_1 a_2 \ldots a_n \in \{0,1\}^n$ such that $C(a_1 a_2 \ldots a_n) = 1$.

Show that Circuit SAT Search $\xrightarrow{p}$ Circuit SAT.

(Hint: you should use the following notation: $C(x_1 \leftarrow a_1, \ldots, x_i \leftarrow a_i)$ to express the fact that inputs $x_1, \ldots, x_i$ have been set to $a_1 \ldots a_i \in \{0,1\}^i$).

We present a polytime algorithm which has access to a solver for CircuitSAT:

```
if CircuitSAT(C) = NO then
  output "no such input"
else
  for i : 1..n
      if CircuitSAT(x_1 ← a_1, ..., x_{i-1} ← a_{i-1}, x_i ← 0) = YES then
        a_i ← 0
      else
        a_i ← 1
      end if
  end for
  output a_1 a_2 ... a_n
end if
```

Name ______________________ Student No. ______________________

No aids allowed. Answer all questions on test paper. Use backs of sheets for scratch work. You may use any results presented in lecture.

Total Marks: 40

1. Let $\Sigma = \{0, 1, \#\}$ be a finite alphabet, and let Σ^* be the set of all strings over Σ (as usual in this course). Let $L_1, L_2, L_3 \subseteq \Sigma^*$ be three languages.

[5] (a) Show that if $L_1 \leq_p L_2$, then $\overline{L_1} \leq_p \overline{L_2}$. (Here $\overline{L}$ denotes the complement of L, that is, $\overline{L} = \Sigma^* - L$).

$L_1 \leq_p L_2$ iff there is a polytime function f such that $x \in L_1 \iff f(x) \in L_2$. Therefore, $x \notin L_1 \iff f(x) \notin L_2$, so $x \in \overline{L_1} \iff f(x) \in \overline{L_2}$, so by definition, $\overline{L_1} \leq_p \overline{L_2}$.

[5] (b) Suppose that L_1 is NP-complete, $L_2 \neq \Sigma^*$, L_2 is in NP, L_3 is in P, and $L_1 = L_2 \cap L_3$. Show that then L_2 is also NP-complete

Since $L_2 \in$ NP, to show that L_2 is NP-complete, it is enough if we show that $L_1 \leq_p L_2$. Define f as follows:

$$f(x) = \begin{cases} x & x \in L_3 \\ y_0 & \text{otherwise} \end{cases}$$

where y_0 is some fixed string in Σ^* such that $y_0 \notin L_2$ (we know that such a string exists as $L_2 \neq \Sigma^*$). Since L_3 is in P, $x \overset{?}{\in} L_3$ can be determined in polytime, so f is polytime.

We now show that $x \in L_1 \iff f(x) \in L_2$. If $x \in L_1$, then $x \in L_3$, so $f(x) = x \in L_2$. If $f(x) \in L_2$, then $f(x) = x$ (since $y_0 \notin L_2$), so $x \in L_3$, so $x \in L_2 \cap L_3 = L_1$.

2. Recall that the decision problem **Independent Set** (IS) is given as follows:
Input: Graph $G = (V, E)$, positive integer B.
Question: Does G have an independent set of size B?

Also recall that $V_0 \subseteq V$ is an *independent set* of G if no two vertices in V_0 are joined by an edge in E. Let the problem **Majority Independent Set** (MIS) be the same as IS, except that the parameter B must satisfy $B \geq \frac{1}{2}|V|$.

Show that IS $\leq_p$ MIS by following the outline below:

[5] (a) Give the reduction.

Suppose that $G = (V, E), B$ is an input instance of IS. We now design an input instance $G' = (V', E'), B'$ to MIS as follows:
If $B \geq \frac{1}{2}|V|$, then $V' = V, E' = E, B' = B$.
Otherwise, assume that $V = \{v_1, \dots, v_n\}$ and let $V' = V \cup \{v'_1, \dots, v'_n\}$, i.e., we added n new vertices to V to obtain V'. Also let $E' = E$ and let $B' = B + |V| = B + n$.

[5] (b) State explicitly the correctness condition of your reduction.

$G = (V, E), B$ is a YES input instance of IS iff $G' = (V', E'), B'$ is a YES input instance of MIS.

[10] (c) Prove the correctness condition.

("$\Longrightarrow$") Suppose that $G = (V, E), B$ is a YES input instance of IS. Then, there is a subset $V_0 \subseteq V$ such that $|V_0| = B$ and V_0 is an independent set. If $B \geq \frac{1}{2}|V|$, then we are done. Otherwise, $V' = V \cup \{v'_1, \ldots, v'_n\}$, so let $V'_0 = V_0 \cup \{v'_1, \ldots, v'_n\}$; clearly $|V'_0| = |V_0| + n = B + n = B'$, and $B + n > n = \frac{1}{2}|V'|$. Furthermore, V'_0 is an independent set.

("$\Longleftarrow$") Suppose that $G' = (V', E'), B'$ is a YES input instance of MIS. If $B' = B$, we are done. Otherwise, there is an independent set in G', call it V'_0, such that $|V'_0| = B' = B + n$. Take all the v_i's out of V'_0 (there are at most n of them), and maybe some v_i's (so that we take out exactly n vertices in total), to obtain V_0 such that $|V_0| = B$. Since B is a positive integer, $B' = B + n > 1 + n$, so we *can* take n vertices out of V'_0. Then V_0 is an independent set in G of size B as required.

[10] 3. Let **Independent Set Search** be defined as follows:
Input: Graph $G = (V, E)$, positive integer B.
Output: An independent set V_0 of G such that $|V_0| = B$ (and "no such set" if it does not exist).

Show that ISS $\xrightarrow{p}$ IS.

We use the standard vertex by vertex search; our algorithm is a polytime algorithm that uses a solver for IS:

$V = \{1, 2, \ldots, n\}$
If IS($G = (V, E), B$) = NO then output "no such set" and halt
for $i : 1..n$
    if IS($G = (V - \{i\}, E - \{e| \ e$ is attached to $i\ \}), B$) = YES then
        $V \longleftarrow V - \{i\}$
        $E \longleftarrow E - \{e| \ e$ is attached to $i\ \}$
    end if
end for
output V