Exercise #4

1. (a) Assume $A, B \in NL$, i.e., there exist NDTMs M_A, M_B deciding A, B respectively in logarithmic space. We describe a NDTM deciding $A \cup B$ in logarithmic space:

Algorithm 1 DECIDE-A-OR-B(x)

```
guess m ∈ {0,1}
if m = 0 then
    return M_A(x)
else
    return M_B(x)
```

Correctness:
($\Rightarrow$) If $x \in A \cup B$, then either $x \in A$ or $x \in B$. If $x \in A$, there's an execution path of M_A that returns *true*, so by guessing $m = 0$ and following that path, DECIDE-A-OR-B returns *true*. The case $x \in B$ is analogous.
($\Leftarrow$) If DECIDE-A-OR-B returned *true*, it has found an execution path in either M_A or M_B that returns *true*, hence $x \in A$ or $x \in B$. In any case, $x \in A \cup B$.
Space Complexity: Only constant space more than $M_A, M_B \Rightarrow$ logarithmic space.

(b) Using $NL = \text{co-}NL$ and Question 1a,

$$A, B \in NL \Rightarrow \overline{A}, \overline{B} \in NL \Rightarrow \overline{A} \cup \overline{B} \in NL \Rightarrow A \cap B = \overline{\overline{A} \cup \overline{B}} \in NL$$

2. (a) SAME-SCC$(G, u, v) =$ (S,T)-CON$(G, u, v) \wedge$ (S,T)-CON(G, v, u). Since (S,T)-CON $\in NL$, a reasoning similar to the one used in Question 1 leads to SAME-SCC $\in NL$.

(b) $NL = \text{co-}NL$, hence DIFFERENT-SCC = $\overline{\text{SAME-SCC}} \in \text{co-}NL$. For some fixed k, let $L_{\geq k} = \{\langle G \rangle |$ the directed graph G contains $\geq k$ SCCs$\}$. Consider the following algorithm:

Algorithm 2 AT-LEAST-k-SCCS(G)

```
for i = 1 to k do
    guess v_i ∈ V
    for j = 1 to i − 1 do
        if DIFFERENT-SCC(v_i, v_j) = false then
            return false
return true
```

Correctness:
($\Rightarrow$) If G has $< k$ SCCs, then for every choice of $v_1, \ldots, v_k$, there exist $j < i$ such that v_i, v_j are in the same SCC, and for that pair, every execution path of DIFFERENT-SCC returns $false$. Thus, all execution paths return $false$.
($\Leftarrow$) If G has $\geq k$ SCCs, then when choosing $v_1, \ldots, v_k$ from distinct SCCs, for all $1 \leq j < i \leq k$ some execution path of DIFFERENT-SCC(v_i, v_j) returns *true*.Thus, there exists an execution path that returns *true*.
Space Complexity:
AT-LEAST-k-SCCS has to remember k vertices ($k \log n$) and DIFFERENT-SCC uses logarithmic space $\Rightarrow$ logarithmic space.

AT-LEAST-k-SCCS accepts $L_{\geq k}$ in logarithmic space, so $L_{\geq k} \in NL$ for all $k \in \mathbb{N}$, esp. for $k = 2006, L_{\geq 2006} \in NL$.

(c) Using $NL = \text{co-}NL$, Question 1b and Question 2b,

$$L_{=k} = L_{\geq k} \cap L_{<(k+1)} = L_{\geq k} \cap \overline{L_{\geq (k+1)}} \in NL$$

3. Clearly DS $\in NP$ since we can check whether $S \subset V$ dominates G in $O(|V|^4)$. We show that DS is NP-hard by reducing VERTEX-COVER (VC) to it.

   Let $(G, k), G = (V, E)$ be an instance of VC. Let $I \subset V$ be the set of isolated vertices in G. Define $\varphi(G, k) = (G', k)$ where $G' = (V', E')$, $V' = V \cup E - I$, $E' = E \cup \{(v, e) | v \in e \in E\}$.

   Correctness:

   ($\Rightarrow$) Let $S \subseteq V, |S| = k$ be a vertex cover of G and let $S' = S - I$. For $v \in V - I$, there exists some edge $(u, v) \in E$ such that either $u \in S'$ or $v \in S'$; for $e = (u, v) \in E$, either $u \in S'$ or $v \in S'$. Hence, V' is dominated by S' of size $\leq k$.

   ($\Leftarrow$) Let $S \subseteq V', |S| \leq k$ be a dominating set of G' with minimal $|S \cap E|$. We claim that $S \subseteq V$: if some $e = (u, v) \in S \cap E$ exists, we could take either u or v to S instead of e, reducing $|S \cap E|$. The set remains a dominating set since $\{u, v, e\}$ is a triangle in G'. Now, for $e = (u, v) \in E \subseteq V'$ we have some $v \in S$ dominating e, so S is a vertex cover of size $\leq k$ in G.

   Space Complexity:
   Logarithmic, as we remember only indices $\leq n$.

4. (a) We can solve 3-SAT in NL^*: given a formula φ and a witness y, we check that y encodes a valid truth assignment for φ and that each clause of φ is satisfied by y. We only need logarithmic space for indices, and obviously $\varphi \in$ 3-SAT if and only if we accept φ, y.

   Since any $L \in NP$ can be logarithmically reduced to 3-SAT, $NP \subseteq NL^*$.

   (b) We have $NL \subseteq P \subseteq NP \subseteq NL^*$. If $P \neq NP$, this containment is strict, so $NL \neq NL^*$.

5. Similar to what we do when composing log-space reductions, we simulate M_f's run on $g(x)$ and compute each bit of $g(x)$ when M_f tries to access it by the following method:
   Simulate M_g on x. Whenever M_g writes a bit, advance a counter; when the requested bit arrives, remember it. Continue running until either M_g accepts (and then return the requested bit of $g(x)$ to M_f) or rejects (and then reject as well).

   Correctness:
   We never return a wrong bit of $g(x)$ and there's always an execution path that of M_g that returns the correct bit. Therefore, there's always an execution path of M_f that returns the correct answer.

   Space Complexity:
   Simulating M_g takes $O(\log |x|)$, remembering which bit of $g(x)$ is needed takes $O(\log |g(x)|)$ and simulating M_f takes $O(\log |g(x)|)$ as well. We have $|g(x)| = poly(|x|)$ since M_g uses log-space, hence $O(\log |g(x)|) = O(\log |x|)$.