Section 7.1: Measuring Complexity

These exercises reflect material from our text, *Introduction to the Theory of Computation*, by Michael Sipser, PWS Publishing Co., 1997.

Definitions

Define each of the following concepts:

(a) Time complexity of a deterministic Turing machine, M

(b) Time complexity of a nondeterministic Turing machine, N

(c) $f(n) \in \mathcal{O}\big(g(n)\big)$

(d) $f(n) \in o\big(g(n)\big)$

(e) $TIME(n^k)$

Time Complexity

Prove or disprove:
If a language L is decided by a TM with time complexity $2^{\mathcal{O}(lg\ n)}$, then $L \in TIME(n^k)$ for some $k \geq 0$.

Verify the time classes in Sipser's Example 7.3.

Let $A = \{0^k 1^k : k \geq 0\}$.

Show that $A \in TIME(n^2)$.

Show that $A \in TIME(n \log n)$.

Show that A can be decided in time $\mathcal{O}(n)$ on a 2-tape Turing machine.

State a theorem which establishes the time complexity of the simulation of a 2-tape Turing machine by a single-tape Turing machine.

State a theorem which establishes the time complexity of the simulation of a nondeterministic Turing machine by a single-tape deterministic Turing machine.

Let $f(n) = nb^n$, for some constant $b > 1$. Show that $f(n) \in 2^{\mathcal{O}(n)}$.