Introduction to the Theory of Computation, 3e
by Michael Sipser
$\not\sim$
Theoremata, Lemmata, & Corollaries (Selected)

theorem (n.) - *A mathematical statement proven to be true.*

lemma (n.) - *A mathematical statement proven to be true, but only relevant in the context of proving another statement.*

corollary (n.) - *A conclusion that follows logically from a theorem easily enough that it does not warrant its own proof.*

This document includes only those theorems, lemmas, and corollaries found in sections of the textbook covered by Professor Jody Paul's Introduction to the Theory of Computation course during the Fall 2020 semester. Definitions and proofs are intentionally excluded, as this is meant to serve as a quick way to answer the question, "Which theorem was the one about [...] ?"

The numbering system given here is drawn from *Introduction to the Theory of Computation, Third Edition*, written by Michael Sipser, published in 2013 in Boston, MA, USA by Cengage Learning. This index was compiled and prepared in LaTeXby Nate Roberts in December 2020,

Chapter 0
Introduction

THEOREM 0.20 For any two sets A and B, $\overline{A \cup B} = \overline{A} \cap \overline{B}$.

THEOREM 0.21 For every graph G, the sum of the degrees of all the nodes in G is an even number.

THEOREM 0.22 For each even number n greater than 2, there exists a 3-regular graph with n nodes.

THEOREM 0.24 $\sqrt{2}$ is irrational.

THEOREM 0.25 For each $t \geq 0$,

$$P_t = PM^t - Y\left(\frac{M^t - 1}{M - 1}\right)$$

Chapter 1

Regular Languages

THEOREM 1.25 The class of regular languages is closed under the union operation.

`Repeated as Theorem 1.45.`

THEOREM 1.26 The class of regular languages is closed under the concatenation operation.

`Repeated as Theorem 1.47.`

THEOREM 1.39 Every nondeterministic finite automaton has an equivalent deterministic finite automaton.

COROLLARY 1.40 A language is regular if, and only if, some nondeterministic finite automaton recognizes it.

THEOREM 1.49 The class of regular languages is closed under the [Kleene] star operation.

THEOREM 1.54 A language is regular if, and only if, some regular expression describes it.

LEMMA 1.55 If a language is described by a regular expression, then it is regular.

LEMMA 1.60 If a language is regular, then it is described by a regular expression.

THEOREM 1.70 **The Pumping Lemma for Regular Languages**

Chapter 2

Context-Free Languages

THEOREM 2.9 Any context-free language is generated by a context-free grammar in Chomsky Normal Form.

THEOREM 2.20 A language is context-free if, and only if, some pushdown automaton recognizes it.

LEMMA 2.21 If a language is context-free, then some pushdown automaton recognizes it.

LEMMA 2.27 If a pushdown automaton recognizes some language, then it is context-free.

COROLLARY 2.32 Every regular language is context-free.

THEOREM 2.34 **The Pumping Lemma for Context-Free Languages**

Chapter 3

The Church-Turing Thesis

THEOREM 3.13 Every multitape Turing machine has an equivalent single-tape Turing machine.

COROLLARY 3.15 A language is Turing-recognizable if, and only if, some multitape Turing machine recognizes it.

THEOREM 3.16 Every nondeterministic Turing machine has an equivalent deterministic Turing machine.

COROLLARY 3.18 A language is Turing-recognizable if, and only if, some nondeterministic Turing machine recognizes it.

COROLLARY 3.19 A language is decidable if, and only if, some nondeterministic Turing machine decides it.

THEOREM 3.21 A language is Turing-recognizable if, and only if, some enumerator enumerates it.

Chapter 4

Decidability

THEOREM 4.1 A_{DFA} is a decidable language.

THEOREM 4.2 A_{NFA} is a decidable language.

THEOREM 4.3 A_{REX} is a decidable language.

THEOREM 4.4 E_{DFA} is a decidable language.

THEOREM 4.5 EQ_{DFA} is a decidable language.

THEOREM 4.7 A_{CFG} is a decidable language.

THEOREM 4.8 E_{CFG} is a decidable language.

THEOREM 4.9 Every context-free language is decidable.

THEOREM 4.11 A_{TM} is undecidable.

THEOREM 4.17 $\mathbb{R}$ is uncountable.

COROLLARY 4.18 Some languages are not Turing-recognizable.

THEOREM 4.22 A language is decidable if, and only if, it is Turing-recognizable and co-Turing-recognizable.

COROLLARY 4.23 $\overline{A_{TM}}$ is not Turing-recognizable.

Chapter 5

Reducibility

THEOREM 5.1 $HALT_{TM}$ is undecidable.

THEOREM 5.2 E_{TM} is undecidable.

THEOREM 5.3 $REGULAR_{TM}$ is undecidable.

THEOREM 5.4 EQ_{TM} is undecidable.

THEOREM 5.13 ALL_{CFG} is undecidable.

THEOREM 5.22 If $A \leq_m B$ and B is decidable, then A is decidable.

COROLLARY 5.23 If $A \leq_m B$ and A is undecidable, then B is undecidable.

THEOREM 5.28 If $A \leq_m B$ and B is Turing-recognizable, then A is Turing-recognizable.

COROLLARY 5.29 If $A \leq_m B$ and A is not Turing-recognizable, then B is not Turing-recognizable.

THEOREM 5.30 EQ_{TM} is neither Turing-recognizable nor co-Turing-recognizable.

Chapter 7

Time Complexity

Theorem 7.8 Let $t(n)$ be a function, where $t(n) \geq n$. Then every $t(n)$-time [$O(t(n))$] multitape Turing machine has an equivalent $t^2(n)$-time [$O(t^2(n))$] single-tape Turing machine.

Theorem 7.11 Let $t(n)$ be a function, where $t(n) \geq n$. Then every $t(n)$-time [$O(t(n))$] nondeterministic single-tape Turing machine has an equivalent $2^{(t(n)}$-time [$O(2^{t(n)})$] deterministic single-tape Turing machine.

Theorem 7.14 $PATH \in \mathrm{P}$.

Theorem 7.15 $RELPRIME \in \mathrm{P}$.

Theorem 7.16 Every context-free language is a member of P.

Theorem 7.20 A language is in NP if, and only if, it is decided by some nondeterministic polynomial-time Turing machine.

Corollary 7.22 $\mathrm{NP} = \cup_k \mathrm{NTIME}(n^k)$.

Theorem 7.24 $CLIQUE$ is in NP.

Theorem 7.25 $SUBSET - SUM$ is in NP.

Theorem 7.27 $SAT \in \mathrm{P}$ if, and only if, $\mathrm{P} = \mathrm{NP}$.

Theorem 7.31 If $A \leq_p B$ and $B \in \mathrm{P}$, then $A \in \mathrm{P}$.

Theorem 7.32 $3SAT$ is polynomial-time reducible to $CLIQUE$.

Theorem 7.35 If B is NP-complete and $B \in \mathrm{P}$, then $\mathrm{P} = \mathrm{NP}$.

THEOREM 7.36 If B is NP-complete and $B \leq_p C$ for C in NP, then C is NP-complete.

THEOREM 7.37 SAT is NP-complete.

COROLLARY 7.42 $3SAT$ is NP-complete.

COROLLARY 7.43 $CLIQUE$ is NP-complete.

THEOREM 7.44 $VERTEX-COVER$ is NP-complete.

THEOREM 7.46 $HAMPATH$ is NP-complete.

THEOREM 7.55 $UHAMPATH$ is NP-complete.

THEOREM 7.56 $SUBSET-SUM$ is NP-complete.