NEWGIZA UNIVERSITY

CSAI 335 Theory of Computing (Fall 2022)
Dr. Muhammad Rushdi
Mrs. Pansy Yusuf

Academic Year:	2022/2023	Semester:	Fall
School:	Information Technology	Course Code:	CSAI 335
Date:	January 22nd, 2023	Course Title:	Theory of Computing
Time:	3 hours	Full Mark:	100

Final Exam

Answer ALL Questions. You can have three A4-sized cheat sheets.

Question A: True/False Statements [10 marks]

State whether the following statements are **True** or **False** (Don't correct the false ones).

	Statement
A.1	A finite automaton (FA) can exactly one accept state.
A.2	A deterministic FA can have exactly one transition exiting every state for each possible input symbol.
A.3	The empty language contains the empty string.
A.4	If the start state of a FA is an accept state, the empty string ε is not accepted.
A.5	The intersection operation is a binary operation over regular languages.
A.6	Every context-free language is regular.
A.7	A string w is derived ambiguously in context-free grammar G if it has a unique rightmost derivation.
A.8	The pumping lemmas for the regular and context-free languages are based on the pigeonhole principle.

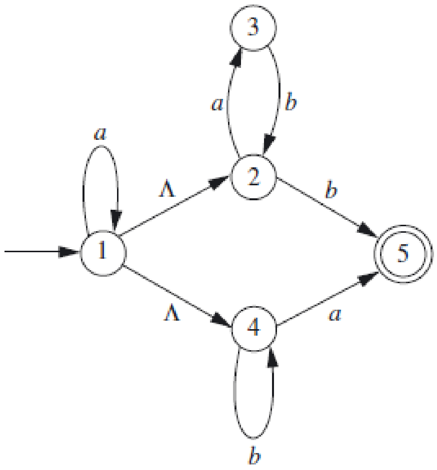
A.9	The language $\{0^i 1^j \mid 0 \leq i \leq j\}$ is non-regular.
A.10	The CFG variables are the symbols that appear on the left-hand side of the rules and the terminals are the remaining symbols.
A.11	In a Turing machine, the tape alphabet contains the input alphabet.
A.12	A non-deterministic single-tape Turing machine is more powerful than a deterministic one.
A.13	Any Turing-recognizable language is decidable.
A.14	The set of negative rational numbers is countable.
A.15	There are more Turing machines than languages.
A.16	The Church-Turing Thesis indicates that computability is dependent on the model of computation.
A.17	NP is the set of languages where we can test membership quickly.
A.18	The CLIQUE problem is NP-complete.
A.19	A problem that is polynomial-time reducible to a problem in *P* is not in *P*.
A.20	A problem with a polynomial-time algorithm has a polynomial-space algorithm.

B. Regular Languages (35 marks)

B.1 Draw a deterministic finite automaton (DFA) for a password verification module that rejects the password string '$aaba$' and accepts all other strings over the alphabet $\{a, b\}^*$.

B.2 Give the state diagram of a nondeterministic finite automaton (NFA) recognizing the concatenation of the languages $L_1 = \{w | w \in \{a, b\}^* \text{ and the length of } w \text{ is at most } 5\}$ and $L_2 = \{w | w \in \{a, b\}^* \text{ and every odd position of } w \text{ is an } a\}$.

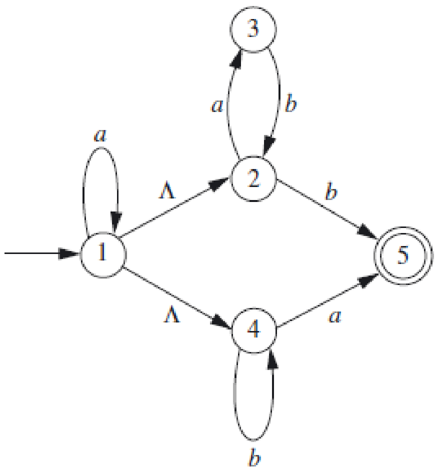
B.3 Convert the following nondeterministic finite automaton (NFA) to an equivalent deterministic finite automaton (DFA) (*Hint:* Build the NFA transition table to define the minimum number of DFA states). The symbol Λ denotes the empty string.



B.4 Give a nondeterministic finite automaton (NFA) recognizing the language generated by the regular expression $(((00)^*(11))\cup 01)^*$.

B.5 Give a regular expression generating the language of strings in $\{a, b\}^*$ ending with b and every a is followed immediately by bb.

B.6 Convert the following finite automata to a regular expression:



B.7 Show that the language $L = \{w : w \epsilon \{a, b\}^*, n_a(w) = n_b(w)\}$ is not a regular language (Hint: Provide a proof by contradiction based on the string $s = a^p b^p$ where p denotes the pumping length).

C. Context-Free Languages (20 marks)

C.1 For this set of context-free grammar (CFG) rules, specify the following:

$S \rightarrow S_1 \mid S_2$
$S_1 \rightarrow \text{if } (\, E\,)\, S_1 \text{ else } S_1 \mid OS$
$S_2 \rightarrow \text{if } (\, E\,)\, S \mid \text{if } (\, E\,)\, S_1 \text{ else } S_2$

- the variables,
- the start variable,
- the terminals,
- one nonacceptable string, and
- one acceptable string (with the corresponding parse tree).

In our notation, *E* is short for <expression>, *S* for <statement>, and *OS* for <otherstatement>.

C.2 Give a context-free grammar (CFG) generating the set of even-length strings over the alphabet $\{a, b\}$ whose first and last symbols are all the same.

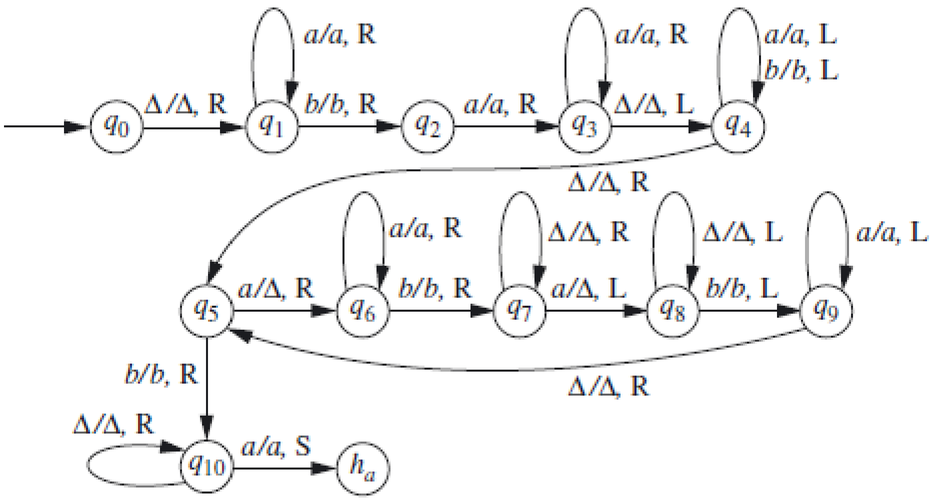
C.3 For the following set of CFG rules, construct an equivalent push-down automata (PDA).

$$S \rightarrow \varepsilon \mid aB \mid bA$$
$$A \rightarrow aS \mid bAA$$
$$B \rightarrow bS \mid aBB$$

C.4 For a string w, let w^R denote the string written backwards. Construct a push-down automata (PDA) for the language $\{w\#s\#w^R : w, s \,\epsilon\, \{a, b\}^*\}$.

D. Turing Machines (15 marks)

D.1 Here is the state diagram for a Turing machine M accepting $L = \{a^i b a^j \mid 0 \leq i < j\}$, where Δ denotes the empty string.



Give the sequence of configurations that M enters when started on the input string $abaa$. Describe what this machine does in plain English.

D.2 Show that a 3-tape deterministic Turing machine is equivalent to a 2-tape deterministic Turing machine.

E. Decidability and Undecidability (10 marks)

E.1 Let EQ_{3DFA} = $\{< A, B, C >| \ A, B$, and C are DFAs and $L(A) = L(B) = L(C)\}$. Show that EQ_{3DFA} is a decidable language (Hint: Construct a new DFA D from A, B, and C where D accepts only those strings that are accepted by either A, B, or C but not all of them).

E.2 Show that E_{TM} = $\{<M> \mid M$ is a TM and $L(M) = \emptyset\}$, the emptiness problem of a Turing machine, is undecidable (Hint: Show that A_{TM}, the acceptance problem of a Turing machine, is reducible to E_{TM}).

F. Complexity (10 marks)

F.1 Let G represent an undirected graph. Also, let $SPATH$ = $\{< G, a, b, k >| \ G$ contains a simple path of length at most k from a to $b\}$, where a simple path is a path that doesn't repeat any nodes. Show that $SPATH \in P$.

F.2 A triangle in an undirected graph is a 3-clique. Show that $TRIANGLE \in PSPACE$ (It has a polynomial-space algorithm), where $TRIANGLE$ = $\{< G >| \ G$ contains a triangle$\}$.