THE UNIVERSITY OF SYDNEY

CONFIDENTIAL EXAM PAPER

This paper is not to be removed from the exam venue.

Computer Science

EXAMINATION

Semester 2 - Practice, 2021

COMP2022 Models of Computation

EXAM WRITING TIME: 3 hours
READING TIME: 10 minutes

EXAM CONDITIONS:
This is an OPEN book examination. You are allowed to use passive information sources (i.e., existing written materials such as books and websites); however, you must not ask other people for answers or post questions on forums; always answer in your own words. You must not reveal the questions to anyone else.
All work must be **done individually** in accordance with the University's "Academic Dishonesty and Plagiarism" policies.

INSTRUCTIONS TO STUDENTS:

1. Type your answers in your text editor and submit a **single PDF** via Canvas; all prose must be typed. Figures/diagrams can be rendered any way you like (hand drawn, latex, etc), as long as they are perfectly readable and part of the submitted PDF.
2. Start by typing your student ID at the top of the first page of your submission. Do **not** type your name.
3. Submit only your answers to the questions. Do **not** copy the questions. Start each of the five problems on a new page.
4. If the required level of justification is not stated, you should briefly justify your answer.

For examiner use only:

Problem	1	2	3	4	5	Total
Marks						
Out of	13	11	14	8	4	50

Problem 1. These questions are about Propositional Logic. If the required level of justification is not stated, you should briefly justify your answer.

a) Is it the case that the formulas $p \to (p \to q)$ and $q \to (q \to p)$ are logically equivalent? [2 marks]

b) Using the equivalences from the tables provided with the exam, show that [3 marks]

$$((p \vee (q \vee r)) \wedge (r \vee \neg p)) \equiv ((q \wedge \neg p) \vee r)$$

You may type your answer in a table or as a sequence of lines, or you may draw the table and insert it into your pdf. No marks will be awarded for proofs that do not use the rules taught in this course and summarised in the tables provided with the exam.

c) Prove the following consequent in natural deduction: [4 marks]

$$b \to d, d \to a, c \vdash a \vee (b \to \neg c)$$

You may type your answer in a table or as a sequence of lines, or you may draw the table and insert it into your pdf. No marks will be awarded for proofs that do not use the rules taught in this course and summarised in the tables provided with the exam.

d) An undirected graph $G = (V, E)$ is called *bipartite* if there is a subset $X \subseteq V$ such that every edge in G has one endpoint in X and the other not in X. Say $V = \{1, 2, \cdots, n\}$. To encode the problem of deciding if a graph is bipartite or not, you introduce variables x_i for $1 \leq i \leq n$. The meaning of x_i being true is that $i \in X$. Write a formula Φ_G in CNF over the variables $x_1, x_2, \cdots, x_n$ that expresses that the graph is bipartite, i.e., Φ_G is satisfiable if and only if G is bipartite. [3 marks]

e) What is wrong with the following recursive algorithm deciding whether a given propositional formula F in NNF is satisfiable or not: [2 marks]

1. If F is an atom or the negation of an atom, return "Satisfiable".
2. If $F = (F_1 \wedge F_2)$ then check if both F_1 and F_2 are satisfiable. If they are, then return "Satisfiable", else return "Unsatisfiable".
3. If $F = (F_1 \vee F_2)$ then check if either F_1 or F_2 are satisfiable. If at least one is, then return "Satisfiable", else return "Unsatisfiable".

Problem 2. These questions are about Context-free Languages. If the required level of justification is not stated, you should briefly justify your answer.

a) Consider the following CFG in CNF: [4 marks]

$$\begin{aligned} S &\to AX \mid AB \mid \epsilon \\ T &\to AX \mid AB \\ X &\to TB \\ A &\to a \\ B &\to b \end{aligned}$$

1. State the variables and terminals of the grammar.
2. Which variables derive the string $aabbb$?
3. The CYK algorithm computes a 2D-array $Table$ where $Table(i, j)$ are the variables that derive the substring starting at position i and ending at position j. In order to compute $Table(2, 5)$, which entries of the table are directly needed to be checked/computed?

b) Explain why if L is a regular language then L is generated by an unambiguous context-free grammar. [2 marks]

c) Write a context-free grammar for the following language: the set of strings of the form $u\#v$ where $u, v \in \{a, b\}^*$ and $|u| = |v|$ and the reverse of u is not equal to v. [4 marks]

Problem 3. These questions are about Regular Languages. If the required level of justification is not stated, you should briefly justify your answer.

a) Give a regular expression that recognises the language of all strings over $\{a, b\}^*$ which has neither aa nor bb as a substring. [2 marks]

b) Consider the NFA $N = (Q, \Sigma, \delta, q_0, F)$ where $Q = \{0, 1, 2, 3, 4\}$, $\Sigma = \{a, b\}$, $q_0 = 0$, $F = \{4\}$, and δ is as follows: [4 marks]

	a	b	ϵ
0	0	0, 1	
1	3	2	
2			4
3		4	
4	4	4	

If you were to apply the technique you learned in class to convert N into an equivalent DFA D,

1. What is the initial state of D?
2. What states of D are reachable from the initial state?
3. What states of D are reachable from the intial state and are also accepting states of D?

No additional explanation is needed.

c) Your friend tells you there is a regular expression that matches those documents that contain mis-matched parentheses. Explain why your friend is mistaken by defining a language that models "documents with mis-matched parentheses" and proving, using methods taught in the course, that there is no regular expression for it. You may use the fact that the language of balanced parentheses is not regular. [2 marks]

d) Let $G = (V, E)$ be a finite directed-graph, i.e., V is a finite set of vertices and $E \subseteq V \times V$ is a set of edges. A path in G is a finite non-empty sequence of vertices $v_0 v_1 \cdots v_k$ such that $(v_i, v_{i+1}) \in E$ for every i with $0 \leq i < k$. Show that the set of paths of G is a regular language over the alphabet V. [3 marks]

e) Consider the operation neg that maps a binary string $u \in \{0, 1\}^*$ to the string in which every 0 is replaced by 1 and vice versa. So, e.g., $neg(0111) = 1000$. Show that if $L \subseteq \{0, 1\}^*$ is regular, then so is $\{neg(u) : u \in L\}$. [3 marks]

Problem 4. These questions are about Turing Machines. If the required level of justification is not stated, you should briefly justify your answer.

a) Is it the case that every finite language is decidable? [2 marks]

b) One of your assignments showed one way of encoding a context-free grammar as a single string (if you consider the new-line/carriage-return as a separator). Is the set of encodings of context-free grammars Turing-decidable? [2 marks]

c) Show that if L is in **P** and L' is in **NP** then $L \cap L'$ is in **NP**. [2 marks]

d) Show that if L is in **P** then L^* is in **P**. [2 marks]

Problem 5. These questions are about Predicate Logic. If the required level of justification is not stated, you should briefly justify your answer.

a) Consider the domain of people, that includes Mick and Sue, and the following predicates: [4 marks]

- `child`(x, y) is a binary predicate saying that person x is a child of person y.
- `eq`(x, y) is a binary predicate saying that x and y are the same person.

Use this to write the following statements in predicate logic:

1. Mick is either the child of Sue, or Sue is the child of Mick.
2. Every person is a child.
3. x and y have exactly one child together.
4. Mick has at least one child with Sue, and no children with anyone else.

No additional explanation is needed.