Clayton School of Information Technology
Faculty of Information Technology
Monash University

Introduction to Theory of Computation

SAMPLE EXAM

2nd Semester, 2013

Instructions:

10 minutes reading time.

3 hours writing time.

No books, calculators or devices.

Total marks on the exam = 120.

Question 1 **(4 marks)**

Annie, Henrietta, Radhanath and Williamina have been shortlisted for two jobs as computers. Let A, H, R, W be propositions with the following meanings.

A: Annie gets one of the jobs.
H: Henrietta gets one of the jobs.
R: Radhanath gets one of the jobs.
W: Williamina gets one of the jobs.

Use A, H, R and W to write a proposition, in Conjunctive Normal Form, that is True precisely when exactly two people get jobs.

Question 2 **(3 marks)**

Suppose you have the predicates `prolog` and `elvish`, with the following meanings:

`prolog`(X): X knows the Prolog language.
`elvish`(X): X knows the Elvish language.

(a) Write a universal statement in predicate logic with the meaning:

"Nobody knows both Prolog and Elvish."

(b) Suppose that the statement in (a) is *False*. Starting with its negation, derive an existential statement meaning that someone knows both these languages.

Question 3 **(2 marks)**

Give a regular expression for the set of all real numbers, represented in binary, that are greater than 0, less than 1, and have a finite binary representation.

(Assume that such binary numbers always have a bit before the binary point (i.e., what we would normally call the "decimal point"), and at least one bit after it.)

Question 4 **(4 marks)**

(a) Write down all strings of at most 8 letters, over alphabet {0,1}, that match the regular expression $((101)^* \cup (00))^*$.

(b) Give an NFA that recognises the language described by this regular expression.

Question 5 **(3 marks)**

Prove that the class of regular languages is closed under complement.

Question 6 **(3 marks)**

What kinds of regular languages can be described by regular expressions that do not use the Kleene star? Explain.

Question 7 **(3 marks)**

Let L be the language of nonempty strings over $\{\mathtt{x},\mathtt{y}\}$ that must start and finish with the same letter, and in the middle have at least one of the other letter. Draw a FA to recognise L.

Question 8 **(4 marks)**

Given the Finite Automaton described by the following table, find an FA with fewest states that recognises the same language.

state	a	b
1	2	6
2	3	6
3	6	3
4	5	4
5	4	6
6	3	6

Write your new FA in the blank table below.

state	a	b
Start		

Question 9 **(5 marks)**

(a) Prove that the language of strings of ever length is regular.

(b) Given the closure properties of regular languages, and the fact that the language of *strings of* ***even*** *length that are* ***not*** *palindromes* is not regular, prove that the language of palindromes is not regular.

Question 10 **(6 marks)**

Consider the following Regular grammar:

$$\begin{aligned} \mathbf{S} &\rightarrow \varepsilon \mid \mathbf{aT} \mid \mathbf{bU} && (1)\\ &\rightarrow \mathbf{aT} \mid \mathbf{bS} && (2)\\ &\rightarrow \mathbf{aS} && (3) \end{aligned}$$

(a) Give (i) a derivation, and (ii) a parse tree, for the string **baab** .

(b) Find the Finite Automaton for the language defined by the above grammar.

(c) Give a regular expression for the language defined by the above grammar.

Question 11 **(3 marks)**

A string over the alphabet $\{0,+,-\}$ is said to be *balanced* if it satisfies both the following:

- (i) for any i, the first i characters in the string contain at least as many $+$ as $-$;
- (ii) the whole string contains the same number of $+$ as $-$.

Give a Context-Free Grammar for the language.

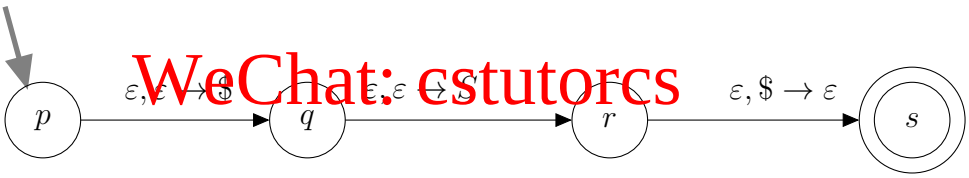
Question 12 **(9 marks)**

The language **Luke** has the following Context-Free Grammar:

$$S \rightarrow Z \quad (1)$$
$$S \rightarrow Sooo \quad (2)$$
$$Z \rightarrow nZ \quad (3)$$
$$Z \rightarrow \varepsilon \quad (4)$$

(a) Give a derivation of the string *nnooooooooo*, indicating which production rule is used at each step.

(b) Complete the following diagram to give a Pushdown Automaton for Luke.



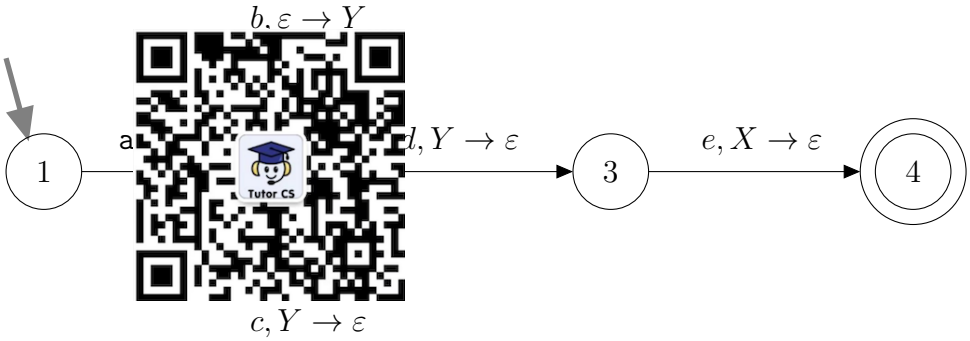
(c) Is the above CFG a regular grammar? (Explain.)

(d) Is Luke a regular language? (Explain.)

(e) Convert the grammar into Chomsky Normal Form.

Question 13 **(5 marks)**

Find a Context-Free Grammar for the language accepted by the following PDA.



Your CFG must use only the nonterminal symbols $S, A_{11}, A_{22}, A_{33}, A_{44}, A_{14}, A_{23}$.

Write the CFG in the space below. Five production rules have already been written in, to get you started.

$S \to A_{14}$

$A_{11} \to \varepsilon$

$A_{22} \to \varepsilon$

$A_{33} \to \varepsilon$

$A_{44} \to \varepsilon$

Question 14 **(8 marks)**

(a) Prove that the language of strings representing powers of 4, in **binary** form, is regular.

(b) Prove that the  representing powers of 2, in **unary** form, is **not** context-free.

Question 15 **(3 marks)**

State two important results that can be proved using the Chomsky Normal Form for Context-Free Grammars.

Question 16 **(6 marks)**

Write a Turing machine that **flips the middle bit** (i.e., changes 0 to 1, and 1 to 0) of a binary string of odd length, and leaves a string of even length unchanged.

For example, if the input string is 011110, then the output must be 011010. If the input string is 011110, then the output is also 011110.

Question 17 **(1 marks)**

The *characteristic function* f_L of a language L over some alphabet is defined by:

$$f_L(w) = \begin{cases} 1, & x \in L, \\ 0, & x \notin L, \end{cases}$$

for any string w over

State the prope ve, for the language L to be decidable.

Question 18 **(4 marks)**

For each of the following decision problems, indicate whether or not it is decidable.

You may assume that, when Turing machines are encoded as strings, this is done using the Code-Word Language (CWL).

Decision Problem	your answer (tick **one** box in each row)	
Input: Turing machines M and N. Question: Are the encoded forms of M and N identical?	☐ Decidable	☐ Undecidable
Input: Turing machines M and N. Question: Do M and N have the same time complexity?	☐ Decidable	☐ Undecidable
Input: a Turing machine M. Question: Does M correctly determine whether or not its input string is a palindrome?	☐ Decidable	☐ Undecidable
Input: a Turing machine M, and a string w. Question: Does M ever change any letter of w on the tape?	☐ Decidable	☐ Undecidable

Question 19 **(9 marks)**

The Venn diagram at the left shows several classes of languages. For each language (a)–(l) in the list below, indicate which classes it belongs to, and which it doesn't belong to, by placing its corresponding letter in the correct region of the diagram.

If a language do[illegible] of these classes, then place its letter above the top of the diagram.

(a) The Dyck lan[illegible]

(b) The set of all [illegible]esented in binary.

(c) The set of all [illegible]thmetic expressions, using integers and the symbols $+$, $-$, $\times$, $/$, and parentheses.

(d) The Code-Word Language (CWL).

(e) The set of all encodings of Turing machines (encoded using strings from CWL).

(f) DOUBLEWORD, the set of all strings consisting of a string concatenated with itself.

(g) The set of all palindromes (i.e., strings that are the same forwards or backwards).

(h) The set of all Turing machines that accept every binary string.

(i) The set of all regular expressions.

(j) The set of all polynomials (with any number of variables) with an integer root.

(k) The set of all satisfiable Boolean expressions in Conjunctive Normal Form with at most two literals in each clause.

(l) The set of all satisfiable Boolean expressions in Conjunctive Normal Form with at most three literals in each clause.

Which, if any, of these languages are NP-complete?

Question 20 **(7 marks)**

Prove that the following problem is undecidable.

Input: a Turing machine M, and a positive integer t.
Question: Is there an input string x of length at least t such that, if M is run on x, it eventually halts?

You may use the fact that the halting problem is undecidable.

Question 21 **(5 marks)**

For this question, recall that the composition of two polynomial-time reductions is again a polynomial-time reduction, and that the notation $\leq_p$ indicates the existence of a polynomial-time reduction.

Prove by induct... $L_1, \ldots, L_n$ are languages, and $L_i \leq_p L_{i+1}$ for all i in the range $1 \leq i \leq n-1$

Question 22 **(6 marks)**

(a) Define the class of *NP-complete* languages.

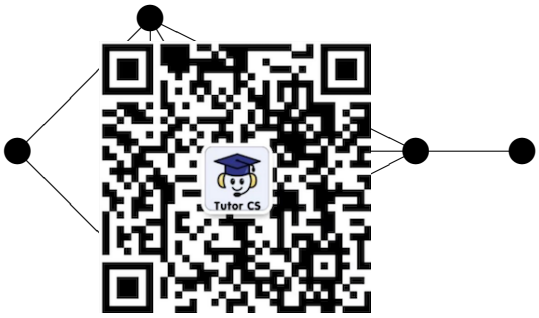
(b) Prove that, if $K \leq_p L$ and $L \in$ NP, then L is NP-complete.

Question 23 **(17 marks)**

Consider the language CUBIC SUBGRAPH, which consists of all graphs G which have a subgraph, with at least one edge, whose vertices all have degree 0 or 3.

(a) Prove that t SUBGRAPH is in NP.

Now, let W be the following graph:



(b) Construct a Boolean expression E_W in Conjunctive Normal Form such that the satisfying truth assignments for E_W correspond to solutions to the CUBIC SUBGRAPH problem on the above graph W. (I.e., they correspond to subgraphs of W for which every vertex has degree 0 or 3 in the subgraph.)

(c) Give a polynomial-time reduction from CUBIC SUBGRAPH to SATISFIABILITY.

(d) Give the usual name for the set of all languages that are polynomial-time Turing reducible to SATISFIABILITY.

(e) If it were shown that all algorithms for CUBIC SUBGRAPH take exponential time, what would you conclude about the time complexity of SATISFIABILITY?

Blank Page for Working

END OF EXAMINATION