

UNIVERSITY OF BIRMINGHAM

School of Computer Science

Theories of Computation

May 2012

Theories of Computation

Exam length: 1.5 hours. Answer THREE questions out of the four. The marks available total 102, but will be capped at 100.

Question 1 : Regular Languages and Automata

- (a) (i) Write out a regular expression that is matched by exactly those strings over an alphabet $\Sigma = \{a, b, c\}$ in which after any b there can be no more occurrences of a. **[4 marks]**
- (ii) Consider the two strings ccaccbcc, which has the given property, and ccbccacc, which does not. Explain why ccaccbcc matches your pattern, but ccbccacc does not. **[8 marks]**
- (b) Design a deterministic finite automaton that accepts exactly those strings that match the regular expression $(ab^* \mid (ab)^*)$. **[12 marks]**
- (c) Consider the language \mathcal{L} consisting of all those strings over the alphabet $\Sigma = \{a, b\}$ that contain more a's than b's. Show that \mathcal{L} is not regular. **[10 marks]**

Question 2 : Context-free languages

In this question we consider a grammar given, over an alphabet $\Sigma = \{l, r, x\}$, by

$$\begin{aligned} \Rightarrow S &::= \varepsilon \mid TS \\ T &::= x \mid lSr \end{aligned}$$

- (a) Of the two strings $xlrxxrlx$ and $xlxlxxrx$, one is derivable (from the initial S) in the grammar and one is not. Which is which? Justify your answer by giving in each case either a derivation tree or a careful argument to show that every attempt to find a derivation fails. **[14 marks]**

- (b) *Removed. Not on the syllabus this year.* **[6 marks]**

In the rest of the question we shall use the notion of **bracket count**. (Think of l and r as left and right brackets.) Imagine counting along from the left of a string, starting at 0 and adding $+1$ for each l , -1 for each r and 0 for each x . At each point of the string, the total so far is called the bracket count there. For example, in the string $xxlxlrrrx$ the bracket counts are shown as subscripts as

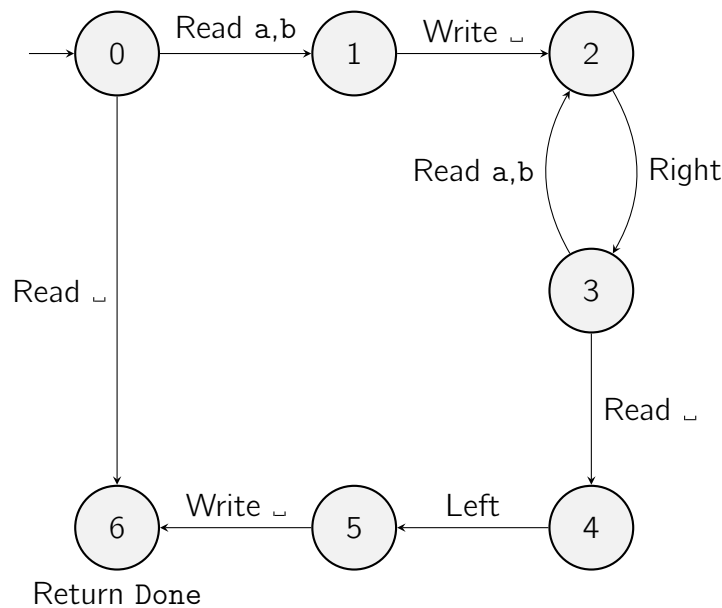
$${}_0x{}_0x{}_0l{}_1x{}_1l{}_2r{}_1r{}_0r{}_0x{}_0$$

We say that a string is **well bracketed** if its bracket count finishes at 0 and never goes negative at any intermediate point.

- (c) Show by induction that every string derived from S is well bracketed. **[7 marks]**
- (d) *Removed. Not on the syllabus this year.* **[7 marks]**

Question 3 : Turing Machines and Complexity

- (a) Consider the following deterministic Turing machine \mathcal{M} on alphabet $\Omega = \{a, b, \sqcup\}$ with return value set $V = \{\text{Done}\}$. The tape initially contains a non-empty block of a's and b's on an otherwise blank tape, with the head on the leftmost non-blank character. The transition function is given by the following diagram:



- (i) *Removed. Not on the syllabus this year.* **[6 marks]**
- (ii) Write out the configurations for the execution of this machine if it starts with input abb , i.e., the initial configuration is

•
$\sqcup \ a \ b \ b \ \sqcup$

[6 marks]
- (b) What does it mean to say that a Turing machine executes in polynomial time?
Note. You may use the \mathcal{O} notation without defining it. **[4 marks]**
- (c) (i) Outline how a Turing machine \mathcal{M} with two tapes, each with its own head, can be simulated by a Turing machine \mathcal{N} with one tape. **[5 marks]**
 (ii) Explain why if \mathcal{M} executes in polynomial time then so does \mathcal{N} . **[5 marks]**
Note. Your outline in part (i) can be very brief, but it should include enough detail to support your explanation in part (ii).
- (d) Which of the following operations are known to be executable in polynomial time on a Turing machine?
 (i) Determining for two integers n and m whether m is a factor of n . **[2 marks]**

(ii) Determining for an integer n whether it has a factor other than 1 and n . **[2 marks]**

(iii) Finding the smallest factor (other than 1) of n . **[2 marks]**

How would your answers change if the “ $P = NP?$ ” problem were solved, either way?
Give reasons. **[2 marks]**

Question 4 : Computability and Models of Computation

(a) We consider Java programs that take their input from a file and print their output to `System.out` when the program terminates. What does it mean to say that a property of such programs is semantic? What does Rice's Theorem tell us about semantic properties? **[5 marks]**

(b) The following two properties of programs are both undecidable.

(i) The output of the program can differ, depending on the first character of the input.

(ii) The output of the program can differ, depending on the name of the input file.

Which one is a semantic property? Outline how you would prove undecidability of each property, using Rice's Theorem where appropriate. **[12 marks]**

(c) *Removed. Not on the syllabus this year.* **[5 marks]**

(d) *Removed. Not on the syllabus this year.* **[6 marks]**

(e) *Removed. Not on the syllabus this year.* **[6 marks]**