Calculators may be used in this examination provided they are not capable of being used to store alphabetical information other than hexadecimal numbers

UNIVERSITY OF BIRMINGHAM

School of Computer Science

Theories of Computation

Main Summer Examinations 2023

Time allowed: 2 hours

[Answer all questions]

Turn Over

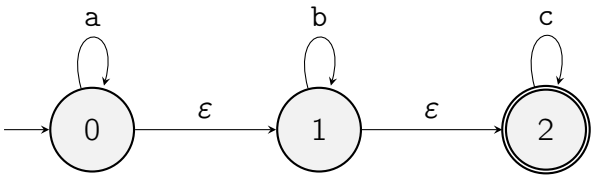
Note

Answer ALL questions. Each question will be marked out of 20. The paper will be marked out of 60, which will be rescaled to a mark out of 100.

Question 1

Take $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ to be the alphabet for this question.

(a) Explain how to remove ε-transitions from an εNFA and illustrate the procedure on the following automaton.



[6 marks]

(b) Consider the following context-free grammar $\mathcal{G}$

$$\begin{array}{rrcl} \Rightarrow & S & ::= & B\mathtt{c} \mid S\mathtt{c} \\ & B & ::= & \mathtt{a}B\mathtt{b} \mid \varepsilon \end{array}$$

where the $\Rightarrow$ symbol indicates the Start variable.

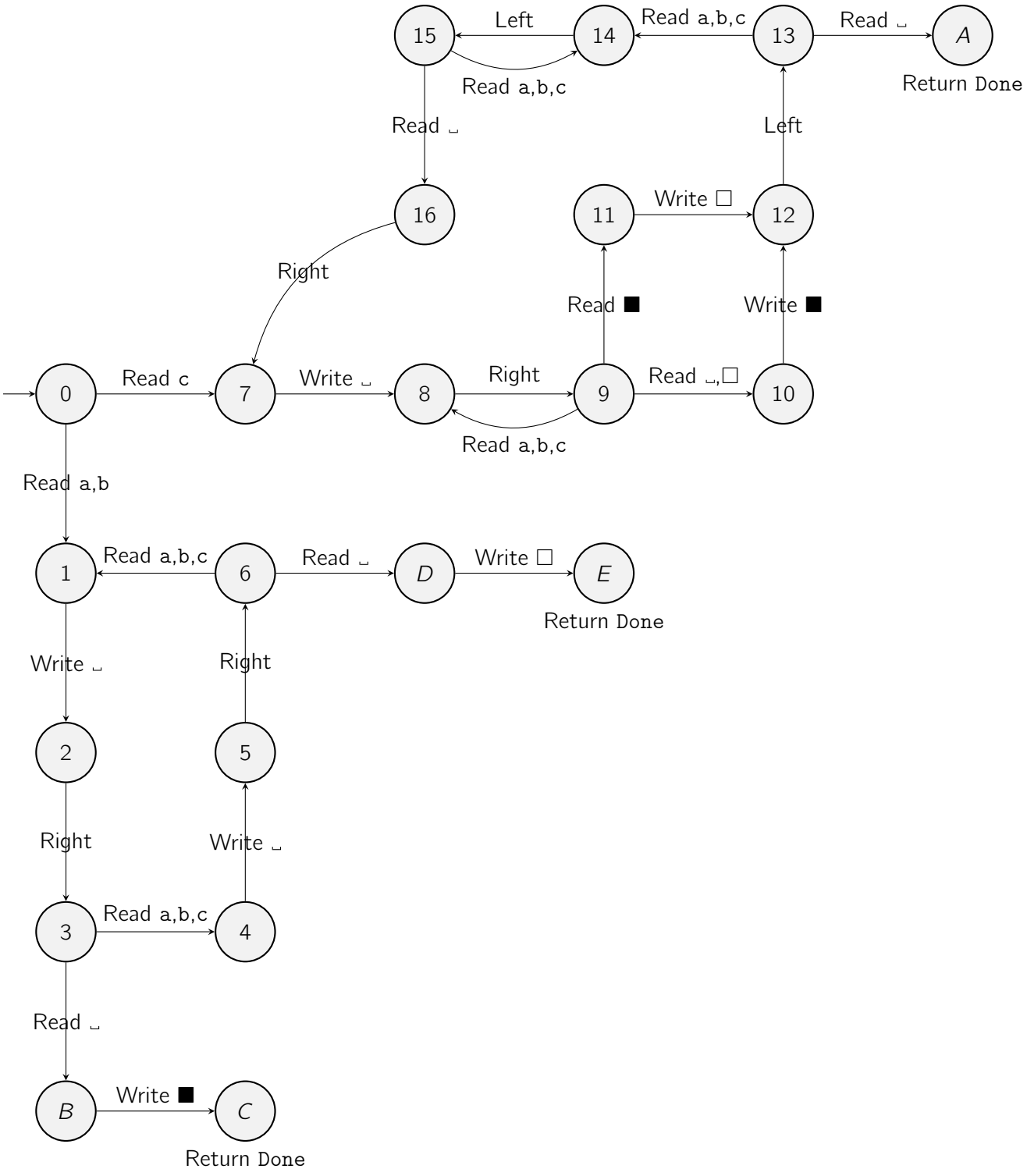
(i) Is the word `abcc` generated by this grammar? If yes, give a derivation tree for it; if no, explain why not. **[4 marks]**

(ii) What is the language $L(\mathcal{G})$ generated by the grammar $\mathcal{G}$? **[4 marks]**

(c) Is the language $\{\mathtt{a}^n\mathtt{b}^n\mathtt{c}^n \mid n \geq 0\}$ regular? Explain your answer. **[6 marks]**

Question 2

Consider the following Turing machine $\mathcal{M}$ on alphabet $\Omega = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \square, \blacksquare, \text{␣}\}$ with return value set $V = \{\mathtt{Done}\}$.



The tape initially contains a non-empty block of a's, b's and c's on an otherwise blank tape, with the head on the leftmost non-blank character.

(a) Give the complete run of machine $\mathcal{M}$ above on the word abc, with the head positioned on a.

At each step, indicate the tape contents, the position of the head, the current state and the instruction (including the result if it is a Read). **[6 marks]**

Hint: No more than 14 steps are required.

(b) The processing time on an input of length $n \geq 1$ is given by the function $f(n)$ when the first character of the input is a or b, and by the function $g(n) = 2n^2 + 9n - 1$ when the first character of the input is a c.

(i) Three students have tried to compute $f(n)$

- Akash found $f(n) = 3n + 3$;
- Belinda found $f(n) = 2n + 5$;
- Chao found $f(n) = 3n^2 + 1$.

Which student performed the correct computation? Explain your answer. **[3 marks]**

(ii) Assume that a, b and c are equally probable and all characters in the input are independent. What is the average processing time of machine $\mathcal{M}$? Prove that this average time is in $O(n^2)$. **[6 marks]**

(c) We would like to convert this Fancy Turing machine on alphabet $\Omega = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \square, \blacksquare, \text{␣}\}$ into a Simple one that only uses alphabet $\Sigma = \{a, b, c, \text{␣}\}$ via the following encoding

Character on Fancy tape	Represented on Simple tape
$\square$	aa
$\blacksquare$	ab
a	ba
b	bb
c	bc
␣	␣␣

The head of the Simple machine is situated on the leftmost of the two characters corresponding to the Fancy machine's head position. For example, the configuration ␣ a $\overset{\bullet}{\text{b}}$ $\blacksquare$ ␣ would be represented as ␣ ␣ b a $\overset{\bullet}{\text{b}}$ b a b ␣ ␣

Give a Turing machine that could be used as a macro on the Simple machine to simulate the Write $\square$ instruction from the Fancy machine. **[5 marks]**

Question 3

(a) State Rice's theorem. **[4 marks]**

(b) A team of TAs for the Object Oriented Programming module have to mark the sorting programs of a class of 500 students. They want to write automated marking software that gives full marks to any correct answer and less than full marks to any incorrect one.

(i) Can the TAs do this in Java? Explain your answer. **[5 marks]**

(ii) Can they do this in a different language? Explain your answer. **[5 marks]**

(c) Recall that Primitive Java is a language with the type `nat` and the following basic facilities.

- `nat i = 0`
- `i++`
- `i--`, which does nothing if `i==0`
- `if (i == 0)` $\{M\}$ `else` $\{N\}$
- `repeat i times` $\{M\}$

The following facilities are derivable from the above basic facilities and may be used in answering this question.

- `nat j = i`
- `j = 0`
- `j = i`
- `if (i <= j)` $\{M\}$ `else` $\{N\}$
- `if (i < j)` $\{M\}$ `else` $\{N\}$
- `if (i == j)` $\{M\}$ `else` $\{N\}$
- `i = j + k`
- `i = j * k`
- `i = max(j-k,0)`

Note: In all of the instructions shown above, the variables used are arbitrary and can be replaced with any others.

To show that the exponentiation function is primitive recursive, give a Primitive Java encoding of:

$$\texttt{i = j}^{\texttt{k}}$$

[6 marks]

End of Paper

Do not complete the attendance slip, fill in the front of the answer book or turn over the question paper until you are told to do so

Important Reminders

- Coats/outwear should be placed in the designated area.
- Unauthorised materials (e.g. notes or Tippex) must be placed in the designated area.
- Check that you do not have any unauthorised materials with you (e.g. in your pockets, pencil case).
- Mobile phones and smart watches **must** be switched off and placed in the designated area or under your desk. They must not be left on your person or in your pockets.
- You are not permitted to use a mobile phone as a clock. If you have difficulty seeing a clock, please alert an Invigilator.
- You are not permitted to have writing on your hand, arm or other body part.
- Check that you do not have writing on your hand, arm or other body part – if you do, you must inform an Invigilator immediately
- Alert an Invigilator immediately if you find any unauthorised item upon you during the examination.

Any students found with non-permitted items upon their person during the examination, or who fail to comply with Examination rules may be subject to Student Conduct procedures.