https://www.inf.ed.ac.uk/teaching/courses/inf1/cl/tools/turing-doodle-master/

UNIVERSITY OF EDINBURGH

COLLEGE OF SCIENCE AND ENGINEERING

SCHOOL OF INFORMATICS

INFORMATICS 1 — COMPUTATION & LOGIC

Saturday 1st April 2017

00:00 to 00:00

INSTRUCTIONS TO CANDIDATES

This is a take-home exercise. It will not be marked anonymously. The examination, which will be marked anonymously, will have a similar format.

Bring your completed script with you to class on Thursday 24th November. You will mark your paper in class and then submit it for feedback on any outstanding queries.

In addition to answering these questions you may find it useful to create your own variations on these examples.

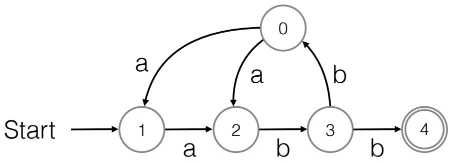
Please give your student number and tutorial group below.

Student ID:

Tutorial group:

THIS EXAMINATION WILL BE MARKED ANONYMOUSLY

6. (a) Which of the following strings are accepted by the NFA in the diagram? (The start state is indicated by an arrow and the accepting state by a double border.)



i. abb

ii. abbabbabbaaabb

iii. abbabbaabbabbabb

iv. abbabaabbabbabb [*3 marks*]

(b) Write a regular expression for the language accepted by this NFA. [*3 marks*]

(c) Draw a DFA that accepts the same language. Label the states of your DFA to make clear their relationship to the states of the original NFA. [*10 marks*]

(d) For each of the following regular expressions, draw a non-deterministic finite state machine that accepts the language described by the regular expression.

i. x^*y

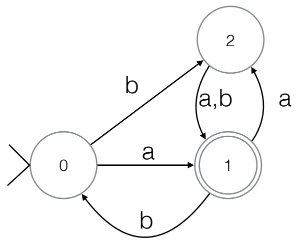
ii. $(x^*|y)$

iii. $(x^*y)^*$ [*9 marks*]

5. Each diagram shows an FSM. In each case give a regular expression for the language accepted by the FSM, make a mark in the check box against each string that it accepts (and no mark against those strings it does not accept), make a mark in the DFA check box if it is deterministic, and draw an equivalent DFA if it is not.

[*4 marks*]

(a)

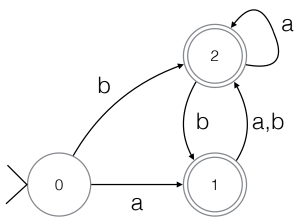


aab □
aba □
bab □
aaa □
bbb □
DFA □

regex:

[*4 marks*]

(b)

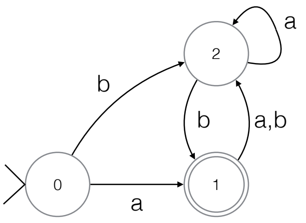


aab □
aba □
bab □
aaa □
bbb □
DFA □

regex:

[*4 marks*]

(c)

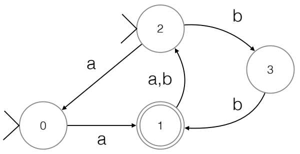


aab □
aba □
bab □
aaa □
bbb □
DFA □

regex:

[*4 marks*]

(d)

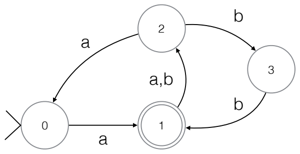


aab □
aba □
bab □
aaa □
bbb □
DFA □

regex:

[*4 marks*]

(e)



aab □
aba □
bab □
aaa □
bbb □
DFA □

regex: