Prof. Javier Esparza
Philipp Czerner

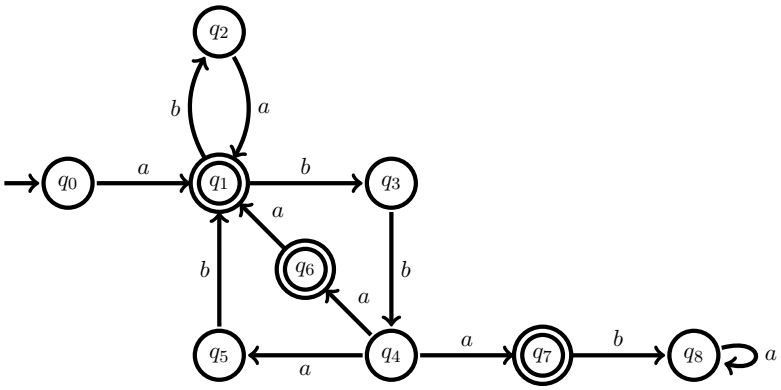
Technical University of Munich
Chair for Foundations of Software Reliability

Automata and Formal Languages

Winter Term 2023/24 – Exercise Sheet 12

Exercise 12.1.

Let B be the following Büchi automaton:



(a) Execute the emptiness algorithm *NestedDFS* on B.

(b) Recall that *NestedDFS* is a non-deterministic algorithm and different choices of runs may return different lassos. Which lassos of B can be found by *NestedDFS*?

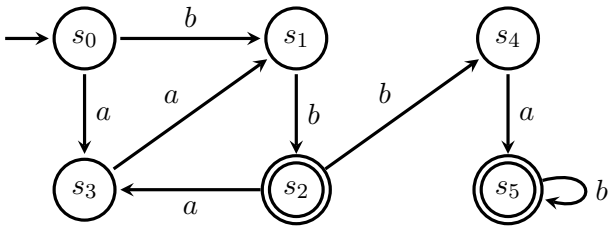
(c) Show that *NestedDFS* is non optimal by exhibiting some search sequence on B.

(d) Execute the emptiness algorithm *SCCsearch* on B.

(e) Which lassos of B can be found by *SCCsearch*?

Exercise 12.2.

Let B be the following Büchi automaton.



(a) For every state of B, give the discovery time and finishing time assigned by a DFS on B starting in s_0 (i.e. the moment they first become grey and the moment they become black). Visit successors s_i of a given state in the ascending order of their indices i. For example, when visiting the successors of s_2, first visit s_3 and later s_4.

(b) The language of B is not empty. Give the witness lasso found by applying $NestedDFS$ to B following the same convention for the order of successors as above.

(c) Given a non-empty NBA, we use the following definition of optimal execution of NestedDFS: the algorithm reports NONEMPTY at the earliest time such that all the states of a witness lasso have been explored. Is the execution in (b) optimal? Does there exists an optimal execution of $NestedDFS$ on B with a different order for visiting successors?

Exercise 12.3.

A Büchi automaton is weak if none of its strongly connected components contains both accepting and non-accepting states. Give an emptiness algorithm for weak Büchi automata. What is the complexity of the algorithm?