9.5/10

Nico Bachmann
Ruben Hutter

3/3 Exercise 3.1

(a) A possible derivation of the word "abaababaaba" is:

$$1\ (aSa) \rightarrow 2\ (abSba) \rightarrow 1\ (abaSaba) \rightarrow 1\ (abaaSaaba) \rightarrow 4\ (abaabaaba)$$

(b) $\mathcal{L}(G)$ is the definition of a Palindrome. The word can always be read either forward or backward and you get the same result.

(c) The Grammar that represents a representation of binary trees is:

$$G = \langle \{\, S \,\},\ \{\, [, \circ, \square,] \,\},\ R,\ S \rangle$$

with R:

1) $S \rightarrow \square$
2) $S \rightarrow [S \circ S]$

1/1 Exercise 3.2

(a) The grammar $G_1 = \langle \{\, S, X, Y \,\}, \{\, a, b \,\}, R_1, S \rangle$ is of type-0 because there is rule $S \rightarrow \epsilon$ and rule $S \rightarrow aS$ which violates the special case, where S never occurs on the right-hand side if S also maps to the empty word.

(b) The grammar $G_2 = \langle \{\, S, X, Y \,\}, \{\, a, b \,\}, R_2, S \rangle$ is of type-1, and so also of type-0, because $baX \rightarrow baaX$ is context-sensitive. The other rules are of type-2 or type-3.

1.5/2 Exercise 3.3

The regular grammar for the given DFA is:

$$G = \langle \{\, S, U, V \,\},\ \{\, a, b, c \,\},\ R,\ S \rangle$$

with rules R as follows:

$$\begin{array}{lll} S \rightarrow aU & S \rightarrow bV & S \rightarrow c \\ U \rightarrow aU & U \rightarrow cS & U \rightarrow c \\ V \rightarrow bV & V \rightarrow cS & V \rightarrow c \end{array}$$

empty word should be accepted but is not -0.5

Nico Bachmann
Ruben Hutter

1/1

Exercise 3.4

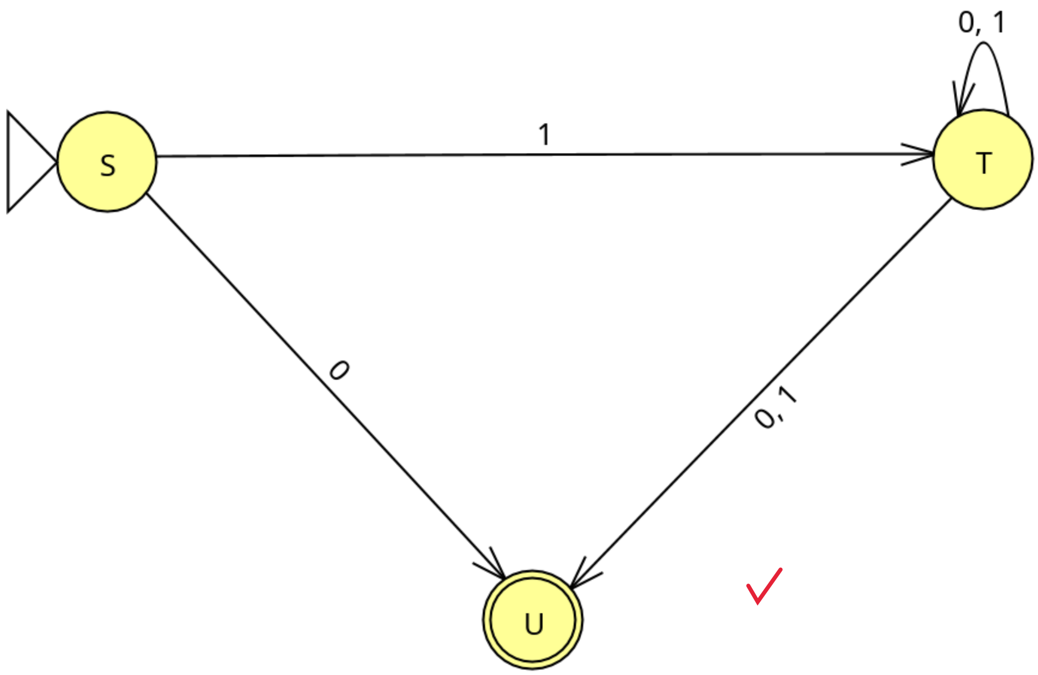

Figure 1: NFA for G

3/3

Exercise 3.5

Fist step

We find the complement of G_2 by inverting the corresponding DFA.

Second step

Given the equivalence problem: $A = B \quad \text{iff} \quad (A \cap B^c) \cup (A^c \cap B) = \emptyset$ which applies in both directions, we can extrapolate only the $(A \cap B^c)$ part in order to obtain $A \subseteq B \quad \text{iff} \quad (A \cap B^c) = \emptyset$. Because regular languages are closed under intersection and complement, and we know algorithms for these operations.

Third step

Decide if $(A \cap B^c)$ is $\emptyset$ by using the algorithm for the emptiness problem $P_\emptyset$.