

**Exercise Sheet 4**Jun.-Prof. Roland Meyer, Reiner Hüchting, Georgel Călin Due: Tue, Nov 13 (**noon**)**Exercise 4.1**

Show that the following languages are star-free:

- (a)  $\{a\}^*\{b\}^*$ ,
- (b) the set of words  $w \in \{a, b\}^*$  that contain the subword  $ab$  as often as the subword  $ba$ .

**Exercise 4.2**

Given an automaton  $A$  and a WMSO[ $<$ , suc]-formula  $\varphi$ , the *model checking* problem asks whether every word accepted by  $A$  satisfies  $\varphi$ . If yes, we write  $A \models \varphi$ .

Show that the model checking problem is Turing reducible to the problem of whether in a finite automaton, one given state can be reached.

**Exercise 4.3**

Let LTL (linear time logic) formulas be defined by the grammar

$$\varphi ::= \text{true} \mid a \mid \varphi_1 \wedge \varphi_2 \mid \neg\varphi \mid \bigcirc\varphi \mid \varphi_1 \mathbf{U} \varphi_2 \mid \diamond\varphi \mid \square\varphi,$$

where  $a \in \Sigma$  are possible *actions* of a system, e.g. *request*, *response*, etc.. Intuitively, an LTL formula describes properties of action sequences, when they are consumed from left to right. The formulas  $\bigcirc\varphi$ ,  $\varphi_1 \mathbf{U} \varphi_2$ ,  $\diamond\varphi$ , and  $\square\varphi$  mean "next  $\varphi$ ", " $\varphi_1$  until  $\varphi_2$ ", "eventually  $\varphi$ ", and "globally  $\varphi$ ", respectively. LTL formulas are commonly used to describe temporal properties of systems, for example

- $\square(\neg\text{ack} \mathbf{U} \text{req})$  : There is no acknowledge before a request.
- $\square(\text{req} \rightarrow \diamond\text{ack})$  : Every request is followed by an acknowledge.
- $\square(\text{req} \rightarrow \bigcirc\text{ack})$  : Every request is immediately followed by an acknowledge.

Give a recursive procedure which transforms every LTL formula into a corresponding FO[ $<$ , suc] formula. *Hint*: if you can express  $\diamond$  and  $\square$  by the other LTL constructs, you do not have to translate them.

**Exercise 4.4**

Construct a finite automaton for the Presburger formula  $\exists y. x = 3y$ .