RHEINISCH-
WESTFÄLISCHE
TECHNISCHE
HOCHSCHULE
AACHEN

LEHRSTUHL FÜR INFORMATIK 2
RWTH Aachen · D-52056 Aachen
http://www-i2.informatik.rwth-aachen.de/

Prof. Dr. Ir. J.-P. Katoen
T. Han & A. Mereacre

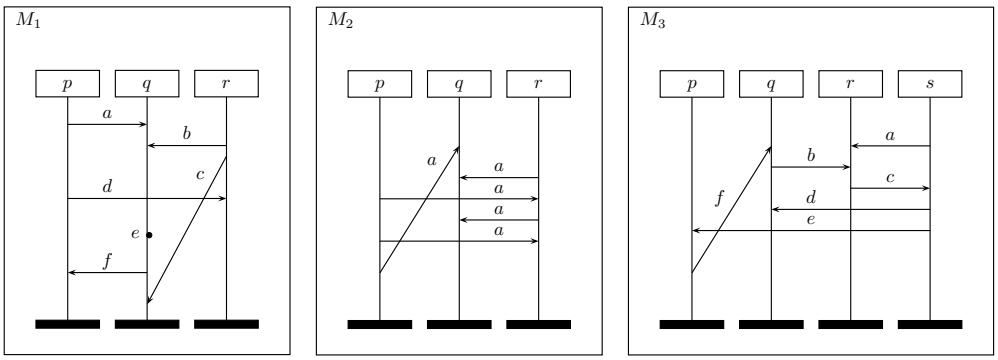
Foundations of UML
Winter term 2009

– Assignment 1 –

October 28th

Exercise 1 (5 points)

Let the following pictures M_1, M_2, M_3 be given:

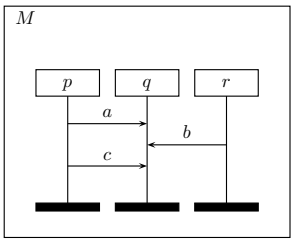


a) Write down the formal description of MSC M_1 as it was presented in the lecture.

b) Prove or disprove that M_2 and M_3 are MSCs.

Exercise 2 (5 points)

Determine all linearizations of the following MSC M:



Exercise 3 **(10 points)**

In this exercise we consider words over sending and receiving actions, only (i.e., there are no local actions). Write down a pseudo-code function that, given a word $\mathtt{w} \in Act^*$, determines whether `w` is a linearization of an MSC. If `w` is not a linearization of an MSC the algorithm has to terminate at the first location where a contradiction to an MSC linearization occurs. The header of the function to implement looks as follows:

```
public static boolean isMSCLinearization(Act[] w)
```

Use the following methods to ease your work:

Class ChannelSystem:

A ChannelSystem is a collection of channels.

```
ChannelSystem()
     //constructor for an empty channel system
boolean addChannel(Process from, Process to)
     //creates a new channel (from,to) (if it does not exist, yet) and
     //returns true iff new channel was created
void putToChannelEnd(Process from, Process to, Message m)
     //appends m to channel (from,to) if channel exists
Message lookAtChannelHead(Process from, Process to)
     //peeks at head of channel without removing the element and returns message
     //content of head element
void removeFromChannelHead(Process from, Process to)
     //removes the element at the head of buffer (from,to)
boolean allChannelsEmpty()
     //returns true iff all channels within the channel system are empty
boolean channelExists(Process from, Process to)
     //returns true iff channel (from,to) exists
```

Class *Act*:

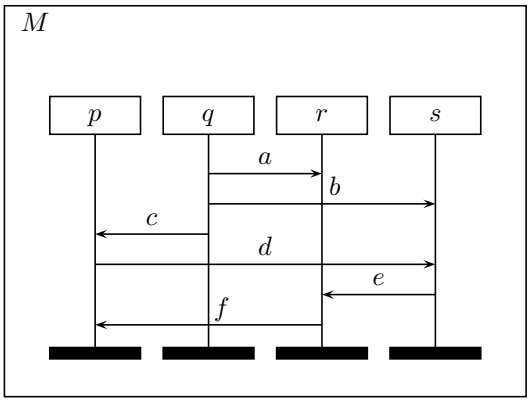
```
boolean isSending()
     //returns true iff this action is of type sending
boolean isReceiving()
     //returns true iff this action is of type receiving
Process getSendingProcess()
     //returns the sending process of this action
Process getReceivingProcess()
     //returns the receiving process of this action
Message getMessage()
     //returns the message content of this action
```

Class Message:

```
boolean equals(Message m)
     //returns whether this message is equal to m
```

Exercise 4 **(5 points)**

Given the following MSC M:



a) Determine the causal order relation $\ll$.

b) Check whether M has races.