

2IW05 Model Examination

Software Specification

Faculteit Wiskunde en Informatica
Technische Universiteit Eindhoven (TU/e)

January 9, 2012

Important Notes. It is not allowed to use study material, computers, and calculators during the examination. The examination comprises 6 question in 3 pages. Please check beforehand whether your copy is properly printed. Justify each step and do not confine yourself to giving the final answer. The answers may be given in Dutch or English. **Good luck!**

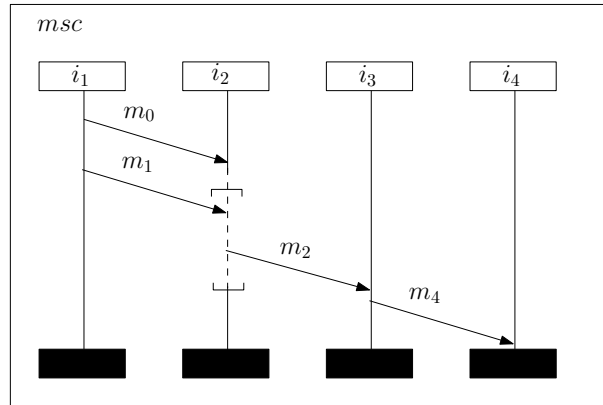
Exercise 1 (20 points) Consider the following state and operation schemas which (partially) specify the state of the Peach system and the operation of joining a group in Peach. Calculate the pre-condition of `joinGroup`.

[*Student, Course, Group*]

<i>Peach</i>
$courses : Course \mapsto \mathbb{P} Student$ $groups : Course \mapsto \mathbb{P}(Group \times Student)$
$dom\ groups \subseteq dom\ courses$ $\forall c : Course \bullet$ $c \in dom\ groups \Rightarrow ran\ groups(c) \subseteq courses(c)$

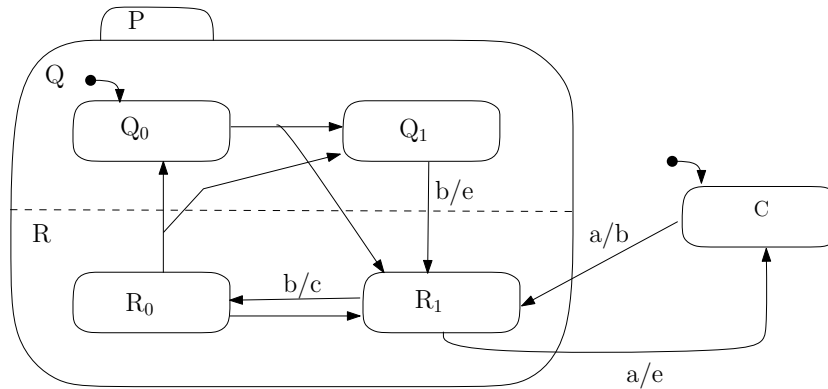
<i>joinGroup</i>
$\Delta Peach$ $c? : Course$ $g? : Group$ $s? : Student$
$courses' = courses$ $groups' = groups \oplus \{c? \mapsto (groups(c?) \cup (g? \mapsto s?))\}$

Exercise 2 (20 points) Consider the following MSC.



1. Draw an LTS which is consistent with the above-given MSC. Each state in the LTS cannot have more than three outgoing transitions. Assume that the LTS has the same set of events as the MSC. (10 points)
2. Combine entities i_2 and i_3 and draw the resulting MSC. (5 points)
3. In the original MSC, abstract from message m_2 and draw the resulting MSC. (5 points)

Exercise 3 (20 points) Consider the following statechart.

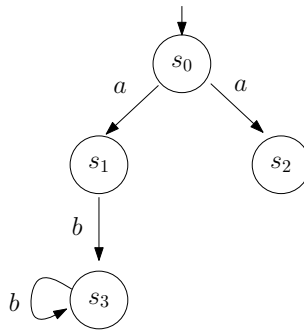


- a. Is it well-structured? If not, identify the anomalies and remove them by making minimal and local changes. (5 points)
- b. Identify the scope of all transitions in the outcome of part a. (Take the original statechart, if it has been well-structured.) (5 points)
- c. Draw its LTS with two transitions of the environment labeled respectively a and c interleaved with two transitions of the statechart. (10 points)

Exercise 4 (20 points) Specify the following informal properties in the modal μ -calculus. (You may use recursion or regular expressions, if so desired. Assume that the set of actions comprises a , b and c only.)

- At the initial state, after each and every action a , an action b can be done. (5 points)
- Nowhere an action a can be done twice in a row without a b in between. (5 points)
- After each two (immediate) occurrences of b in a row a c will be eventually performed. (5 points)
- There is a reachable infinite path comprising only c actions. (5 points)

Exercise 5 (10 points) Determine in which states of the following LTS the formula $[a]X$ holds, where $X \stackrel{\text{min}}{=} [b]X$. Give the steps towards the final answer.



Exercise 6 (10 points) Draw a class diagram for the following informal specification of a customer service unit.

For each caller name, telephone and address are stored. Moreover, some callers are clients, i.e., they already have a contract with the company, in which case their customer-id, already present in the set of customers saved in the client-manager system, is checked and noted. A caller is connected to a unique handling callee, which is either an automatic answering machine or an operator. An operator can sign a new contract with a caller and add her/him to the set of customers. Each operator has an identifier. The operations possible through an answering machine are left out for the sake of simplicity.

Answer 1

pre *joinGroup* =

$\exists \text{ courses}' : \text{Course} \leftrightarrow \mathbb{P} \text{ Student}; \text{ groups}' : \text{Course} \leftrightarrow \mathbb{P}(\text{Group} \times \text{Student}) \bullet$

joinGroup
$\text{courses}, \text{courses}' : \text{Course} \leftrightarrow \mathbb{P} \text{ Student}$ $\text{groups}, \text{groups}' : \text{Course} \leftrightarrow \mathbb{P}(\text{Group} \times \text{Student})$ $c? : \text{Course}$ $g? : \text{Group}$ $s? : \text{Student}$
$\text{courses}' = \text{courses}$ $\text{groups}' = \text{groups} \oplus \{c? \mapsto \text{groups}(c?) \cup (g? \mapsto s?)\}$ $\text{dom groups} \subseteq \text{dom courses}$ $\forall c : \text{Course} \bullet$ $c \in \text{dom groups} \Rightarrow \text{ran groups}(c) \subseteq \text{courses}(c)$ $\text{dom groups}' \subseteq \text{dom courses}'$ $\forall c : \text{Course} \bullet$ $c \in \text{dom groups}' \Rightarrow \text{ran groups}'(c) \subseteq \text{courses}'(c)$

=

pre joinGroup
$\text{courses} : \text{Course} \leftrightarrow \mathbb{P} \text{ Student}$ $\text{groups} : \text{Course} \leftrightarrow \mathbb{P}(\text{Group} \leftrightarrow \text{Student})$ $c? : \text{Course}$ $g? : \text{Group}$ $s? : \text{Student}$
$\exists \text{ courses}' : \text{Course} \leftrightarrow \mathbb{P} \text{ Student}; \text{ groups}' : \text{Course} \leftrightarrow \mathbb{P}(\text{Group} \leftrightarrow \text{Student}) \bullet \text{ courses}' = \text{courses}$ $\text{groups}' = \text{groups} \oplus \{c? \mapsto \text{groups}(c?) \cup (g? \mapsto s?)\}$ $\text{dom groups} \subseteq \text{dom courses}$ $\forall c : \text{Course} \bullet$ $c \in \text{dom groups} \Rightarrow \text{ran groups}(c) \subseteq \text{courses}(c)$ $\text{dom groups}' \subseteq \text{dom courses}'$ $\forall c : \text{Course} \bullet$ $c \in \text{dom groups}' \Rightarrow \text{ran groups}'(c) \subseteq \text{courses}'(c)$

=

pre *joinGroup*

courses : *Course* \leftrightarrow \mathbb{P} *Student*
groups : *Course* \leftrightarrow \mathbb{P} (*Group* \leftrightarrow *Student*)
c? : *Course*
g? : *Group*
s? : *Student*

\exists *courses'* : *Course* \leftrightarrow \mathbb{P} *Student*; *groups'* : *Course* \leftrightarrow \mathbb{P} (*Group* \leftrightarrow *Student*) • *courses'* = *courses*
groups' = *groups* \oplus {*c?* \mapsto *groups*(*c?*) \cup (*g?* \mapsto *s?*)}
 $\text{dom } \textit{groups} \subseteq \text{dom } \textit{courses}$
 $\forall c : \textit{Course} \bullet$
 $c \in \text{dom } \textit{groups} \Rightarrow \text{ran } \textit{groups}(c) \subseteq \textit{courses}(c)$
 $\text{dom } \textit{groups}' \subseteq \text{dom } \textit{courses}'$
 $\forall c : \textit{Course} \bullet$
 $c \in \text{dom } \textit{groups}' \Rightarrow \text{ran } \textit{groups}'(c) \subseteq \textit{courses}'(c)$

=

pre *joinGroup*

courses : *Course* \leftrightarrow \mathbb{P} *Student*
groups : *Course* \leftrightarrow \mathbb{P} (*Group* \times *Student*)
c? : *Course*
g? : *Group*
s? : *Student*

\exists *groups'* : *Course* \leftrightarrow \mathbb{P} (*Group* \times *Student*) • *courses* \in *Course* \leftrightarrow \mathbb{P} *Student*
groups' = *groups* \oplus {*c?* \mapsto *groups*(*c?*) \cup (*g?* \mapsto *s?*)}
 $\text{dom } \textit{groups} \subseteq \text{dom } \textit{courses}$
 $\forall c : \textit{Course} \bullet$
 $c \in \text{dom } \textit{groups} \Rightarrow \text{ran } \textit{groups}(c) \subseteq \textit{courses}(c)$
 $\text{dom } \textit{groups}' \subseteq \text{dom } \textit{courses}$
 $\forall c : \textit{Course} \bullet$
 $c \in \text{dom } \textit{groups}' \Rightarrow \text{ran } \textit{groups}'(c) \subseteq \textit{courses}(c)$

=

pre *joinGroup*

courses : *Course* \leftrightarrow \mathbb{P} *Student*
groups : *Course* \leftrightarrow \mathbb{P} (*Group* \times *Student*)
c? : *Course*
g? : *Group*
s? : *Student*

\exists *groups'* : *Course* \leftrightarrow \mathbb{P} (*Group* \times *Student*) • *groups'* = *groups* \oplus {*c?* \mapsto *groups*(*c?*) \cup (*g?* \mapsto *s?*)}
 $\text{dom } \textit{groups} \subseteq \text{dom } \textit{courses}$
 $\forall c : \textit{Course} \bullet$
 $c \in \text{dom } \textit{groups} \Rightarrow \text{ran } \textit{groups}(c) \subseteq \textit{courses}(c)$
 $\text{dom } \textit{groups}' \subseteq \text{dom } \textit{courses}$
 $\forall c : \textit{Course} \bullet$
 $c \in \text{dom } \textit{groups}' \Rightarrow \text{ran } \textit{groups}'(c) \subseteq \textit{courses}(c)$

=

pre *joinGroup*

courses : *Course* \mapsto \mathbb{P} *Student*
groups : *Course* \mapsto \mathbb{P} (*Group* \times *Student*)
c? : *Course*
g? : *Group*
s? : *Student*

groups \oplus $\{c? \mapsto \text{groups}(c?) \cup (g? \mapsto s?)\} \in \text{Course} \mapsto \mathbb{P}(\text{Group} \times \text{Student})$
 $\text{dom groups} \subseteq \text{dom courses}$
 $\forall c : \text{Course} \bullet$
 $c \in \text{dom groups} \Rightarrow \text{ran groups}(c) \subseteq \text{courses}(c)$
 $\text{dom groups} \cup \{c? \mapsto \text{groups}(c?) \cup (g? \mapsto s?)\} \subseteq \text{dom courses}$
 $\forall c : \text{Course} \bullet$
 $c \in \text{dom groups} \cup \{c? \mapsto \text{groups}(c?) \cup (g? \mapsto s?)\} \Rightarrow$
 $\text{ran groups} \cup \{c? \mapsto \text{groups}(c?) \cup (g? \mapsto s?)\}(c) \subseteq \text{courses}(c)$

=

pre *joinGroup*

courses : *Course* \mapsto \mathbb{P} *Student*
groups : *Course* \mapsto \mathbb{P} (*Group* \times *Student*)
c? : *Course*
g? : *Group*
s? : *Student*

$\text{dom groups} \subseteq \text{dom courses}$
 $\forall c : \text{Course} \bullet$
 $c \in \text{dom groups} \Rightarrow \text{ran groups}(c) \subseteq \text{courses}(c)$
 $\text{dom groups} \cup \{c?\} \subseteq \text{dom courses}$
 $\forall c : \text{Course} \bullet$
 $c \in \text{dom groups} \cup \{c?\} \Rightarrow \text{ran}((\text{groups} \oplus \{c? \mapsto \text{groups}(c?) \cup (g? \mapsto s?)\})(c)) \subseteq \text{courses}(c)$

=

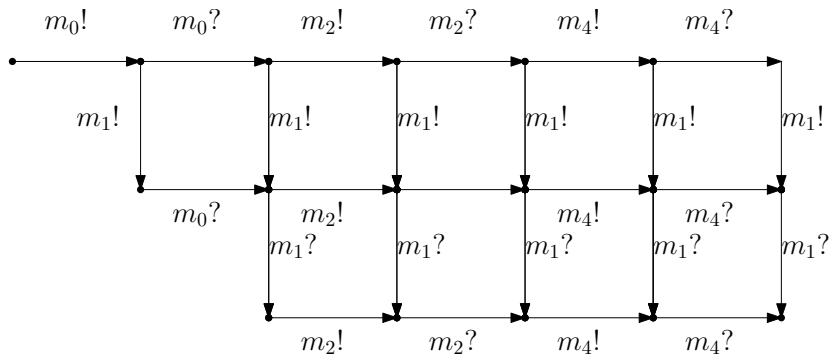
pre *joinGroup*

Peach
c? : *Course*
g? : *Group*
s? : *Student*

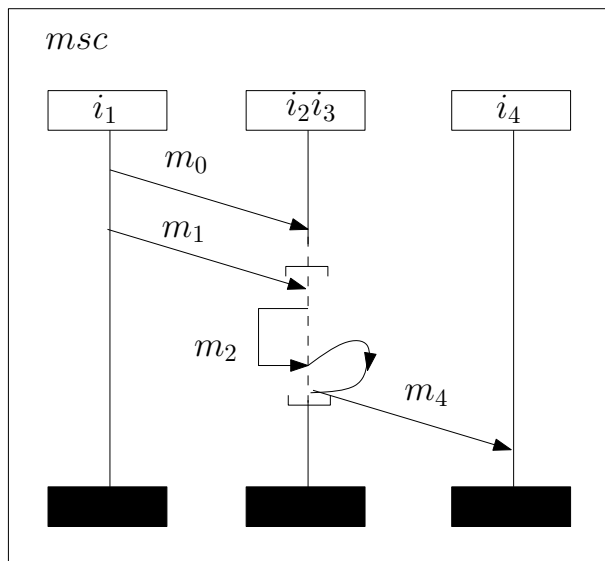
$c? \in \text{dom courses}$
 $s? \in \text{courses}(c?)$

Answer 2

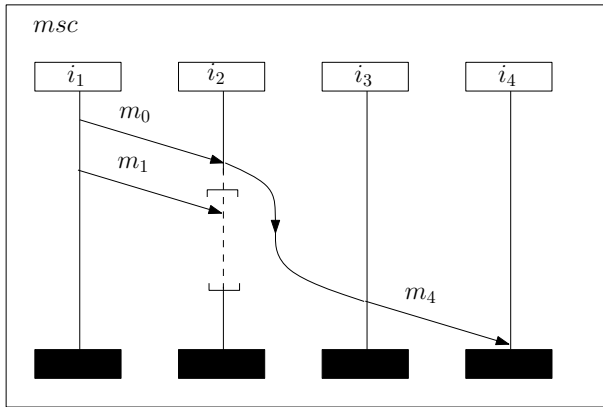
1.



2.

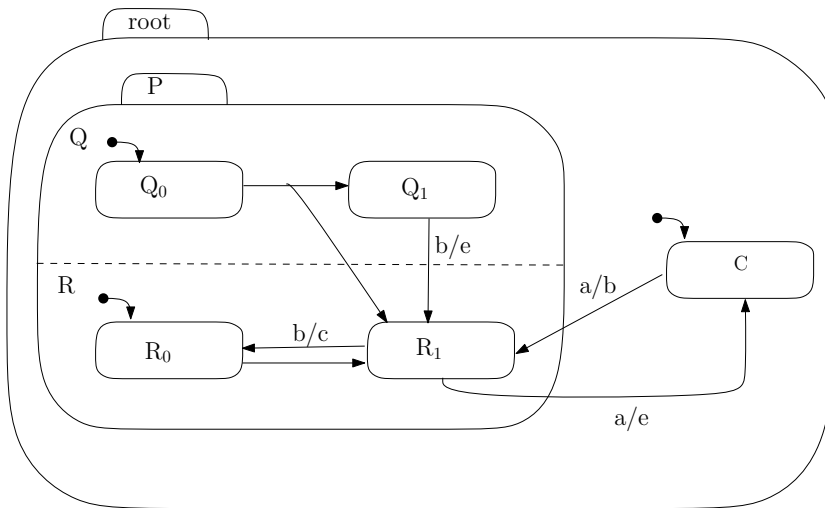


3.



Answer 3

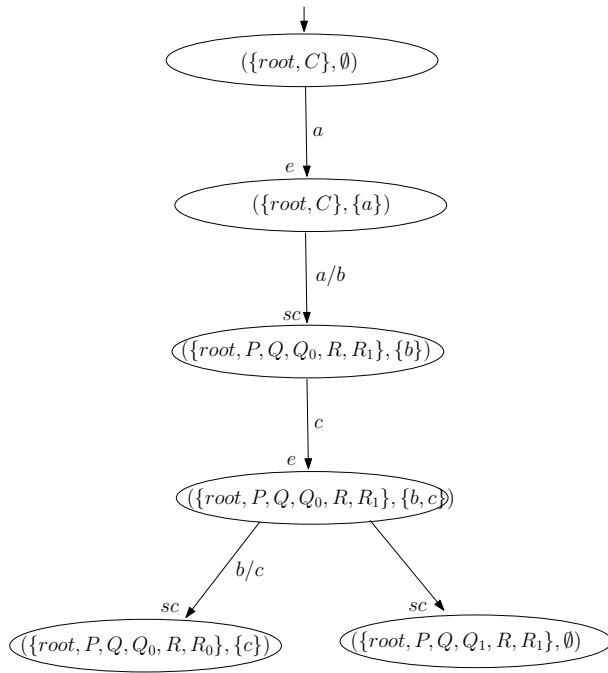
1. There is no root. The transition originating from R_0 is not well-defined. R does not have a default child.



2.

- $scope(\emptyset[\emptyset]/\emptyset) = root,$
- $scope(b/c) = R,$
- $scope(b/e) = root,$
- $scope(a/e) = root,$
- $scope(a/b) = root$

3.



Answer 4

- a. $[a]\langle b \rangle true$
- b. $[true^*.a.\bar{b}^*.a]false$
- c. $[true^*.b.b]X$ where $X \stackrel{\min}{=} [\bar{c}]X \wedge \langle true \rangle true$
- d. $\langle true^* \rangle Y$ where $Y \stackrel{\max}{=} \langle c \rangle Y$

Answer 5

First we calculate the solution to the equation $X \stackrel{\min}{=} [b]X$. We start with substituting the empty set of states for X in the right-hand side and we obtain $\{s_0, s_2\}$ as the left-hand side (first approximate). Substituting $\{s_0, s_2\}$ for X in the right-hand side, we obtain $\{s_0, s_2\}$ in the left-hand side, which is a fixed-point and thus the minimal solution to X .

Next, we put $X = \{s_0, s_2\}$ in the semantics of the formula $[a]X$ and we obtain $\{s_1, s_2, s_3\}$ as the outcome, i.e., the formula $[a]X$ holds in $\{s_1, s_2, s_3\}$.

Answer 6

