

Exam System Validation, 2IW26

Tuesday, June 24, 2014, 9:00-12:00

It is neither allowed to use the study material nor a computer. The axioms formulated in the book are given as an appendix to this exam. The answers to the questions can be formulated in English or Dutch. This exam consists of **6** questions. Good luck!

1. Prove exactly using the process axioms and the rules and equations for the data type \mathbb{B} that

$$\sum_{e:\mathbb{B}} X(e) = X(\text{true}) + X(\text{false}).$$

2. Express the following properties using the modal μ -calculus.
 - (a) There is a path to a state where an a action can be done.
 - (b) On every path from the initial state, an a action must be done.
 - (c) It is possible to do an a action infinitely often, directly from the initial state.
 - (d) From the initial state there is a trace consisting of at least 200 a -actions.
 - (e) No livelock can be reached from the initial state. A livelock is a state where an infinite number of τ -actions are possible.
3. Consider the following pairs of modal formulas. Give a labelled transition system only containing actions a and b for which the first formula is true and the second is not. Reversely, provide a labelled transition system for which the first formula is false and the second is true. If no such transition systems exist, clearly indicate why, using the laws for modal logical formulas included in this exam.
 - (a) $[a^*]\langle a \wedge b \rangle \text{true}$ and $\mu X.([b]X \wedge \langle a \vee b \rangle \text{true})$.
 - (b) $\langle a \rangle \text{true}$ and $[a] \text{true}$.
4. Consider a process $X = a \cdot a \cdot X$ and $Y = a \cdot a \cdot a \cdot Y$. Prove using RSP that $X = Y$.
5. Consider the process equations $X = a \cdot b \cdot X + Y$ and $Y = a \cdot (X + Y)$.
 - (a) Give a linear process of which the behaviour is strongly bisimilar to that of X .
 - (b) Is $\tau_{\{a\}}(X)$ τ -confluent? Is $\tau_{\{a\}}(X)$ τ -convergent? Draw a state space of $\tau_{\{a\}}(X)$ after applying τ -priorisation to its maximal extent. Does this reduction preserve branching bisimulation? Explain all your answers; a simple yes or no does not suffice.
6. (a) Assume that a sort V of function symbols that represent variables is given. Typical elements of V are x and y . Define the sort of *BoolExpression* that can be built up using ‘variables’ from V , the operator *and*, and the operator *not*. A typical term of sort *BoolExpression* is $\text{and}(\text{var}(x), \text{not}(\text{var}(y)))$. Take care that an equality on expressions is defined also, such that only expressions that are syntactically the same are equal.
 - (b) Prove (precisely using the axioms defining the datatype *BoolExpression*) that an expression $\text{var}(x)$ is not equal to $\text{not}(\text{var}(x))$.

- (c) Define an interpretation function $eval : BoolExpression \times (V \rightarrow \mathbb{B})$ that given a boolean expression and a variable evaluation function that maps variables to booleans, yields the truth value belonging to an expression. So, if for some variable evaluation function σ it holds that $\sigma(x) = true$ and $\sigma(y) = false$, where x, y are variables in V , then $eval(and(var(x), not(var(y))), \sigma) = true$.
- (d) Describe an ‘*evaluation-machine*’-process that reads a term of sort $BoolExpression$ via an action *in* and delivers the evaluation of this term via an action *out*. It uses an internal variable evaluation function. Via an action *set* this variable evaluation function can be changed. As an example, $set(x, true)$ indicates that the variable evaluation function should assign *true* to the variable *x*. Initially, this evaluation function maps all variables to *false*.

END

Score: $(10 + n)/10$ where n is the cumulative judgement given by the following table:

| question | (a) | (b) | (c) | (d) | (e) |
|----------|-----|-----|-----|-----|-----|
| 1 | 10 | | | | |
| 2 | 4 | 4 | 4 | 4 | 4 |
| 3 | 6 | 6 | | | |
| 4 | 5 | 5 | | | |
| 5 | 5 | 5 | | | |
| 6 | 6 | 4 | 10 | 8 | |

| | |
|------|---|
| MA1 | $\alpha \beta = \beta \alpha$ |
| MA2 | $(\alpha \beta) \gamma = \alpha (\beta \gamma)$ |
| MA3 | $\alpha \tau = \alpha$ |
| MD1 | $\tau \setminus \alpha = \tau$ |
| MD2 | $\alpha \setminus \tau = \alpha$ |
| MD3 | $\alpha \setminus (\beta \gamma) = (\alpha \setminus \beta) \setminus \gamma$ |
| MD4 | $(a(d) \alpha) \setminus a(d) = \alpha$ |
| MD5 | $(a(d) \alpha) \setminus b(e) = a(d) (\alpha \setminus b(e)) \quad \text{if } a \not\equiv b \text{ or } d \not\approx e$ |
| MS1 | $\tau \sqsubseteq \alpha = \text{true}$ |
| MS2 | $a(d) \sqsubseteq \tau = \text{false}$ |
| MS3 | $a(d) \alpha \sqsubseteq a(d) \beta = \alpha \sqsubseteq \beta$ |
| MS4 | $a(d) \alpha \sqsubseteq b(e) \beta = a(d) (\alpha \setminus b(e)) \sqsubseteq \beta \quad \text{if } a \not\equiv b \text{ or } d \not\approx e$ |
| MAN1 | $\underline{\tau} = \tau$ |
| MAN2 | $\underline{a(d)} = a$ |
| MAN3 | $\underline{\alpha \beta} = \underline{\alpha} \underline{\beta}$ |

Table 1: Axioms for multi-actions

| | |
|------------------|--|
| A1 | $x + y = y + x$ |
| A2 | $x + (y + z) = (x + y) + z$ |
| A3 | $x + x = x$ |
| A4 | $(x + y) \cdot z = x \cdot z + y \cdot z$ |
| A5 | $(x \cdot y) \cdot z = x \cdot (y \cdot z)$ |
| A6 \boxtimes | $x + \delta = x$ |
| A7 | $\delta \cdot x = \delta$ |
| Cond1 | $\text{true} \rightarrow x \diamond y = x$ |
| Cond2 | $\text{false} \rightarrow x \diamond y = y$ |
| THEN \boxtimes | $c \rightarrow x = c \rightarrow x \diamond \delta$ |
| SUM1 | $\sum_{d:D} x = x$ |
| SUM3 | $\sum_{d:D} X(d) = X(e) + \sum_{d:D} X(d)$ |
| SUM4 | $\sum_{d:D} (X(d) + Y(d)) = \sum_{d:D} X(d) + \sum_{d:D} Y(d)$ |
| SUM5 | $(\sum_{d:D} X(d)) \cdot y = \sum_{d:D} X(d) \cdot y$ |

Table 2: Axioms for the basic operators

| | |
|-----------------|---|
| M | $x \parallel y = x \parallel y + y \parallel x + x y$ |
| LM1 \boxtimes | $\alpha \parallel x = \alpha \cdot x$ |
| LM2 \boxtimes | $\delta \parallel x = \delta$ |
| LM3 \boxtimes | $\alpha \cdot x \parallel y = \alpha \cdot (x \parallel y)$ |
| LM4 | $(x+y) \parallel z = x \parallel z + y \parallel z$ |
| LM5 | $(\sum_{d:D} X(d)) \parallel y = \sum_{d:D} X(d) \parallel y$ |
| S1 | $x y = y x$ |
| S2 | $(x y) z = x (y z)$ |
| S3 | $x \tau = x$ |
| S4 | $\alpha \delta = \delta$ |
| S5 | $(\alpha \cdot x) \beta = \alpha \beta \cdot x$ |
| S6 | $(\alpha \cdot x) (\beta \cdot y) = \alpha \beta \cdot (x \parallel y)$ |
| S7 | $(x+y) z = x z + y z$ |
| S8 | $(\sum_{d:D} X(d)) y = \sum_{d:D} X(d) y$ |
| TC1 | $(x \parallel y) \parallel z = x \parallel (y \parallel z)$ |
| TC2 | $x \parallel \delta = x \cdot \delta$ |
| TC3 | $(x y) \parallel z = x (y \parallel z)$ |

Table 3: Axioms for the parallel composition operators

| | | | |
|----|---|----|---|
| C1 | $\Gamma_C(\alpha) = \gamma_C(\alpha)$ | C4 | $\Gamma_C(x \cdot y) = \Gamma_C(x) \cdot \Gamma_C(y)$ |
| C2 | $\Gamma_C(\delta) = \delta$ | C5 | $\Gamma_C(\sum_{d:D} X(d)) = \sum_{d:D} \Gamma_C(X(d))$ |
| C3 | $\Gamma_C(x+y) = \Gamma_C(x) + \Gamma_C(y)$ | | |

Table 4: Axioms for the communication operator

| | | | |
|-----|--|--|---|
| V1 | $\nabla_V(\alpha) = \alpha$ if $\underline{\alpha} \in V \cup \{\tau\}$ | V4 | $\nabla_V(x+y) = \nabla_V(x) + \nabla_V(y)$ |
| V2 | $\nabla_V(\alpha) = \delta$ if $\underline{\alpha} \notin V \cup \{\tau\}$ | V5 | $\nabla_V(x \cdot y) = \nabla_V(x) \cdot \nabla_V(y)$ |
| V3 | $\nabla_V(\delta) = \delta$ | V6 | $\nabla_V(\sum_{d:D} X(d)) = \sum_{d:D} \nabla_V(X(d))$ |
| TV1 | | $\nabla_V(\nabla_W(x)) = \nabla_{V \cap W}(x)$ | |

Table 5: Axioms for the allow operator

| | | | |
|-----|---|----|---|
| E1 | $\partial_B(\tau) = \tau$ | E6 | $\partial_B(x + y) = \partial_B(x) + \partial_B(y)$ |
| E2 | $\partial_B(a(d)) = a(d)$ | E7 | $\partial_B(x \cdot y) = \partial_B(x) \cdot \partial_B(y)$ |
| E3 | $\partial_B(a(d)) = \delta$ | E8 | $\partial_B(\sum_{d:D} X(d)) = \sum_{d:D} \partial_B(X(d))$ |
| E4 | $\partial_B(\alpha \beta) = \partial_B(\alpha) \partial_B(\beta)$ | E5 | $\partial_B(\delta) = \delta$ |
| E10 | $\partial_H(\partial_{H'}(x)) = \partial_{H \cup H'}(x)$ | | |

Table 6: Axioms for the blocking operator

| | | | |
|----|---|--|---|
| R1 | $\rho_R(\tau) = \tau$ | | |
| R2 | $\rho_R(a(d)) = b(d)$ | | if $a \rightarrow b \in R$ for some b |
| R3 | $\rho_R(a(d)) = a(d)$ | | if $a \rightarrow b \notin R$ for all b |
| R4 | $\rho_R(\alpha \beta) = \rho_R(\alpha) \rho_R(\beta)$ | | |
| R5 | $\rho_R(\delta) = \delta$ | | |
| R6 | $\rho_R(x + y) = \rho_R(x) + \rho_R(y)$ | | |
| R7 | $\rho_R(x \cdot y) = \rho_R(x) \cdot \rho_R(y)$ | | |
| R8 | $\rho_R(\sum_{d:D} X(d)) = \sum_{d:D} \rho_R(X(d))$ | | |

Table 7: Axioms for the renaming operator

| | | | |
|-----|---|----|---|
| H1 | $\tau_I(\tau) = \tau$ | H6 | $\tau_I(x+y) = \tau_I(x) + \tau_I(y)$ |
| H2 | $\tau_I(a(d)) = \tau$ | H7 | $\tau_I(x \cdot y) = \tau_I(x) \cdot \tau_I(y)$ |
| H3 | $\tau_I(a(d)) = a(d)$ | H8 | $\tau_I(\sum_{d:D} X(d)) = \sum_{d:D} \tau_I(X(d))$ |
| H4 | $\tau_I(\alpha \beta) = \tau_I(\alpha) \tau_I(\beta)$ | H5 | $\tau_I(\delta) = \delta$ |
| H10 | $\tau_I(\tau_{I'}(x)) = \tau_{I \cup I'}(x)$ | | |

Table 8: Axioms for the hiding operator

| | |
|-------------------------------------|--|
| W\boxtimes | $x \cdot \tau = x$ |
| BRANCH\boxtimes | $x \cdot (\tau \cdot (y + z) + y) = x \cdot (y + z)$ |

Table 9: Axioms for τ , valid in rooted branching bisimulation for untimed processes

| | |
|---|--|
| Failures equivalence Trace equivalence Language equivalence Weak trace equivalence | F1 \boxtimes $a \cdot (b \cdot x + u) + a \cdot (b \cdot y + v) = a \cdot (b \cdot x + b \cdot y + u) + a \cdot (b \cdot x + b \cdot y + v)$ F2 \boxtimes $a \cdot x + a \cdot (y + z) = a \cdot x + a \cdot (x + y) + a \cdot (y + z)$ RDIS $x \cdot (y + z) = x \cdot y + x \cdot z$ Lang1 $x \cdot \delta = \delta$ RDIS $x \cdot (y + z) = x \cdot y + x \cdot z$ RDIS $x \cdot (y + z) = x \cdot y + x \cdot z$ WT $\tau \cdot x = x$ W $x \cdot \tau = x$ |
|---|--|

Table 10: Axioms for some other equivalences for untimed processes

| | |
|--|---|
| <p>Proposition logic</p> $\phi \wedge \psi = \psi \wedge \phi$ $(\phi \wedge \psi) \wedge \chi = \phi \wedge (\psi \wedge \chi)$ $\phi \wedge \phi = \phi$ $\neg \text{true} = \text{false}$ $\phi \wedge \text{true} = \phi$ $\phi \wedge \text{false} = \text{false}$ $\phi \wedge (\psi \vee \chi) = (\phi \wedge \psi) \vee (\phi \wedge \chi)$ $\neg(\phi \wedge \psi) = \neg\phi \vee \neg\psi$ $\neg\neg\phi = \phi$ $\phi \Rightarrow \psi = \neg\phi \vee \psi$ | $\phi \vee \psi = \psi \vee \phi$ $(\phi \vee \psi) \vee \chi = \phi \vee (\psi \vee \chi)$ $\phi \vee \phi = \phi$ $\neg \text{false} = \text{true}$ $\phi \vee \text{true} = \text{true}$ $\phi \vee \text{false} = \phi$ $\phi \vee (\psi \wedge \chi) = (\phi \vee \psi) \wedge (\phi \vee \chi)$ $\neg(\phi \vee \psi) = \neg\phi \wedge \neg\psi$ $\phi \rightarrow \psi = \neg\phi \vee \psi$ $\phi \Leftrightarrow \psi = \phi \Rightarrow \psi \wedge \psi \Rightarrow \phi$ |
| <p>Predicate logic</p> $\forall d:D.\phi = \phi$ $\neg\forall d:D.\Phi(d) = \exists d:D.\neg\Phi(d)$ $\forall d:D.(\Phi(d) \wedge \Psi(d)) = \forall d:D.\Phi(d) \wedge \forall d:D.\Psi(d)$ $\forall d:D.(\Phi(d) \vee \psi) = \forall d:D.\Phi(d) \vee \psi$ $\forall d:D.\Phi(d) \Rightarrow \Phi(e)$ | $\exists d:D.\phi = \phi$ $\neg\exists d:D.\Phi(d) = \forall d:D.\neg\Phi(d)$ $\exists d:D.(\Phi(d) \vee \Psi(d)) = \exists d:D.\Phi(d) \vee \exists d:D.\Psi(d)$ $\exists d:D.(\Phi(d) \wedge \psi) = \exists d:D.\Phi(d) \wedge \psi$ $\Phi(e) \Rightarrow \exists d:D.\Phi(d)$ |
| <p>Action formulas</p> $\overline{\text{true}} = \text{false}$ $\overline{\alpha_1 \cup \alpha_2} = \overline{\alpha_1} \cap \overline{\alpha_2}$ $\overline{\exists d:D.A(d)} = \forall d:D.\overline{A(d)}$ | $\overline{\text{false}} = \text{true}$ $\overline{\alpha_1 \cap \alpha_2} = \overline{\alpha_1} \cup \overline{\alpha_2}$ $\overline{\forall d:D.A(d)} = \exists d:D.\overline{A(d)}$ |
| <p>Hennessy-Milner logic</p> $\neg\langle a \rangle \phi = [a]\neg\phi$ $\langle a \rangle \text{false} = \text{false}$ $\langle a \rangle (\phi \vee \psi) = \langle a \rangle \phi \vee \langle a \rangle \psi$ $\langle a \rangle \phi \wedge [a]\psi \Rightarrow \langle a \rangle (\phi \wedge \psi)$ | $\neg[a]\phi = \langle a \rangle \neg\phi$ $[a]\text{true} = \text{true}$ $[a](\phi \wedge \psi) = [a]\phi \wedge [a]\psi$ $[a](\phi \vee \psi) \Rightarrow \langle a \rangle \phi \vee [a]\psi$ |

Table 11: Equivalences between modal formulas (part I)

| | |
|--|--|
| <p>Fixed point equations</p> $\mu X.\phi(X) \Rightarrow \nu X.\phi(X)$ $\mu X.\phi = \phi$ $\mu X.X = \text{false}$ $\mu X.\langle R \rangle X = \text{false}$ $\neg \mu X.\phi(X) = \nu X.\neg\phi(\neg X)$ $\mu X.\phi(X) = \phi(\mu X.\phi(X))$ $\text{if } \phi(\psi) \Rightarrow \psi \text{ then } \mu X.\phi(X) \Rightarrow \psi$ <p>Regular formulas</p> $\langle \varepsilon \rangle \phi = \phi$ $\langle \text{false} \rangle \phi = \text{false}$ $\langle af_1 \cup af_2 \rangle \phi = \langle af_1 \rangle \phi \vee \langle af_2 \rangle \phi$ $\langle af_1 \cap af_2 \rangle \phi \Rightarrow \langle af_1 \rangle \phi \wedge \langle af_2 \rangle \phi$ $\langle \exists d:D.AF(d) \rangle \phi = \exists d:D.\langle AF(d) \rangle \phi$ $\langle \forall d:D.AF(d) \rangle \phi \Rightarrow \forall d:D.\langle AF(d) \rangle \phi$ $\langle R_1 + R_2 \rangle \phi = \langle R_1 \rangle \phi \vee \langle R_2 \rangle \phi$ $\langle R_1 \cdot R_2 \rangle \phi = \langle R_1 \rangle \langle R_2 \rangle \phi$ $\langle R^* \rangle \phi = \mu X.(\langle R \rangle X \vee \phi)$ $\langle R^+ \rangle \phi = \langle R \rangle \langle R^* \rangle \phi$ $\neg \langle R \rangle \phi = [R]\neg\phi$ $[R]\text{true} = \text{true}$ $\langle R \rangle (\phi \vee \psi) = \langle R \rangle \phi \vee \langle R \rangle \psi$ $\langle R \rangle \phi \wedge [R]\psi \Rightarrow \langle R \rangle (\phi \wedge \psi)$ | $\nu X.\phi = \phi$ $\nu X.X = \text{true}$ $\nu X.[R]X = \text{true}$ $\neg \nu X.\phi(X) = \mu X.\neg\phi(\neg X)$ $\nu X.\phi(X) = \phi(\nu X.\phi(X))$ $\text{if } \psi \Rightarrow \phi(\psi) \text{ then } \psi \Rightarrow \nu X.\phi(X)$ <p></p> $[\varepsilon]\phi = \phi$ $[\text{false}]\phi = \text{true}$ $[af_1 \cup af_2]\phi = [af_1]\phi \wedge [af_2]\phi$ $[af_1 \cap af_2]\phi \Leftarrow [af_1]\phi \vee [af_2]\phi$ $[\exists d:D.AF(d)]\phi = \forall d:D.[AF(d)]\phi$ $[\forall d:D.AF(d)]\phi \Leftarrow \exists d:D.[AF(d)]\phi$ $[R_1 + R_2]\phi = [R_1]\phi \wedge [R_2]\phi$ $[R_1 \cdot R_2]\phi = [R_1][R_2]\phi$ $[R^*]\phi = \nu X.([R]X \wedge \phi)$ $[R^+]\phi = [R][R^*]\phi$ $\neg[R]\phi = \langle R \rangle \neg\phi$ $\langle R \rangle \text{false} = \text{false}$ $[R](\phi \wedge \psi) = [R]\phi \wedge [R]\psi$ $[R](\phi \vee \psi) \Rightarrow \langle R \rangle \phi \vee [R]\psi$ |
|--|--|

Table 12: Equivalences between modal formulas (part II)