Chair of Foundations of Software Reliability and Theoretical Computer Science
School of Computation, Information and Technology
Technical University of Munich

Eexam
Place student sticker here

Note:

- During the attendance check a sticker containing a unique code will be put on this exam.
- This code contains a unique number that associates this exam with your registration number.
- This number is printed both next to the code and to the signature field in the attendance check list.

Petrinetze

Exam:	IN2052 / Retake	**Date:**	Thursday 1st October, 2026
Examiner:	Prof. Dr. Javier Esparza	**Time:**	12:00 – 13:15

Working instructions

- This exam consists of **8 pages** with a total of **4 problems**.
  Please make sure now that you received a complete copy of the exam.
- The total amount of achievable credits in this exam is **44 credits**, including **4 bonus credits**.
- To pass the exam, 17 credits are *sufficient*.
- Detaching pages from the exam is prohibited.
- Allowed resources:
  - one **analog dictionary** English ↔ native language
- **Answers are only accepted if the solution approach is documented.** Give a reason for each answer unless explicitly stated otherwise in the respective subproblem.
- Do not write with red or green colors nor use pencils.
- Physically turn off all electronic devices, put them into your bag and close the bag.

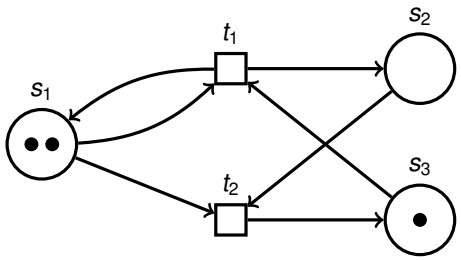
Left room from _____ to _____ / Early submission at _____

In all problems, Petri nets do not have arc weights.

Problem 1 Coverability and well-quasi-orders (8 credits)

0 1 2 3

a) Perform a step of the backwards coverability algorithm starting with $m_0 = \{(2,0,1)\}$. Recall that this requires computing $pre(R[t_1] \wedge m_0, t_1)$ and $pre(R[t_2] \wedge m_0, t_2)$, and then computing $m_1 = \min(m_0 \cup \bigcup_{t \in T} pre(R[t] \wedge m_0, t))$. You only need to give the results.



$pre(R[t_1] \wedge m_0, t_1) =$ $pre(R[t_2] \wedge m_0, t_2) =$ $m_1 =$

0 1 2 3 4 5

b) Let $A_1, A_2, ...$ be an infinite sequence of **upward-closed** sets of markings of a net N such that $A_1 \subseteq A_2 \subseteq ....$ Using Dickson's lemma, show that this sequence stabilizes, i.e. that there exists $n_0 \in \mathbb{N}$ such that for all $n \geq n_0$, we have $A_n = A_{n_0}$.
Dickson's lemma: Let $M_1, M_2, ...$ be an infinite sequence of markings of a net N. Then there exist two indices $i < j$ such that $M_i \leq M_j$.

Problem 2 **Construction (15 credits)**

For this exercise, it is sufficient to draw the Petri net. You are welcome to add explanations, but a correct Petri net already gives full points.

a) Construct an unbounded Petri net (N, M_0) with an S-invariant I such that $I(s) \neq 0$ for every place s of N.

0 1 2 3

b) **Bonus points:** Construct the Petri net corresponding to the net program

1: $x := x + 1$**;**
2: goto 1 or goto 3
3: gosub 6;
4: $x := x - 1$**;**
5: halt
6: $y := y - 1$**;**
7: return

0 1 2 3 4

0 1 2 3 4

c) Construct a live and bounded Petri net such that there is a place which does not belong to any S-component.

0 1 2 3 4

d) Construct a non-live Petri net in which every proper siphon contains an initially marked trap.

Problem 3 **Reductions (10 credits)**

a) Given a net N and two markings M_1, M_2, show how to construct a Petri net (N', M_0') and a marking M' such that

(N, M_1) and (N, M_2) have the same set of reachable markings **iff** M' is reachable in (N', M_0').

You do not need to give a formal definition of N' and M_0', but describe them sufficiently. Sketch a proof of correctness of the construction. Your construction should be implementable as a program that runs in polynomial time in the size of N, M_1 and M_2.

0 1 2 3 4 5

b) Given a Petri net (N, M_0) with $N = (S, T, F)$, show how to construct a Petri net (N', M_0') and a transition t' such that

(N, M_0) has an infinite firing sequence **iff** (N', M_0') has a firing sequence which contains t' infinitely often.

Indicate clearly which are the sets of places, transitions and arcs of N', and which is the marking M_0'. Sketch a proof of correctness of the construction. Your construction should be implementable as a program that runs in polynomial time in the size of (N, M_0).

0 1 2 3 4 5

Problem 4 True or false (11 credits)

For each of the following statements, decide whether the statement is true. If it is true, give a proof. If it is false, give a counterexample.

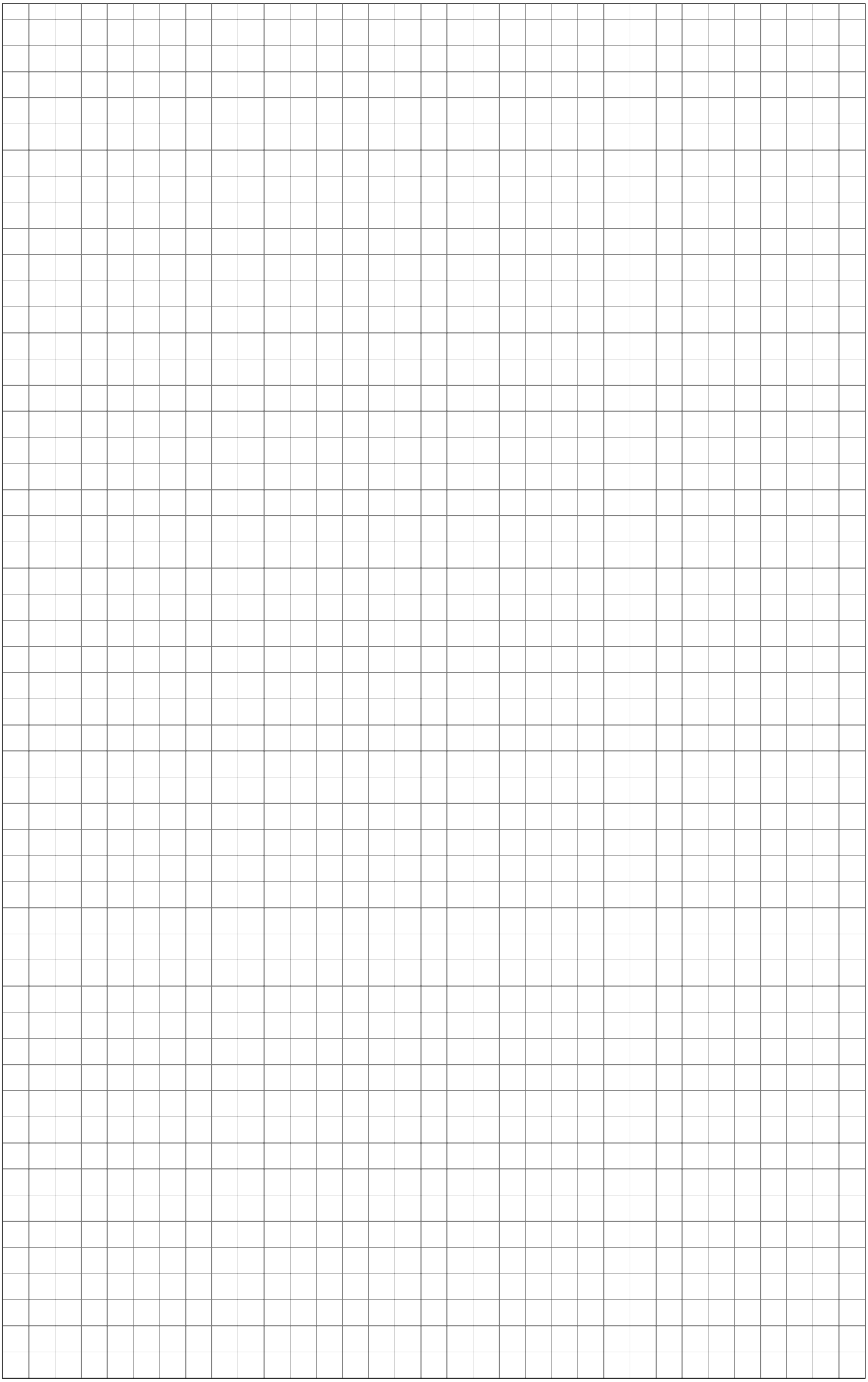
0 1 2 3

a) Let N be a net and let M_0, M_0' be markings such that (N, M_0) and (N, M_0') are bounded and $M := \frac{M_0 + M_0'}{2}$ is a marking. Then (N, M) is bounded.

0 1 2 3 4

b) Let (N, M_0) be a Petri net such that the zero marking is reachable, and M_0 is not the zero marking. Then there exist infinitely many markings M such that the zero marking is reachable in (N, M).

c) Let (N, M_0) be a bounded Petri net. If an infinite firing sequence is enabled at M_0, then N has a semi-positive T-invariant.

0 1 2 3 4

Additional space for solutions–clearly mark the (sub)problem your answers are related to and strike out invalid solutions.