

HS 2018

Prof. L. Thiele, Prof. L. Vanbever, Prof. R. Wattenhofer

Discrete Event Systems

Exam

Tuesday, 5th February 2019, 14:00–16:00.

Do not open until told to by the supervisors!

The exam lasts 120 minutes, and there is a total of 120 points. The maximal number of points for each question is indicated in parentheses. Your answers must be in English. Be sure to always justify your answers.

Give your solutions on the page corresponding to the exam question and/or the empty one(s) following it. In case you run out of space, we also added extra pages – in case you use them, please indicate which question you are solving! Should even this not be enough, please contact a supervisor.

Please write down your name and Legi number (student ID) in the boxes below. Once the exam starts, also write your name on every page in the top right corner.

Name	Legi-Nr.

Points

Question	Topic	Achieved Points	Maximal Points
1	Regular Languages		25
2	Context-free Languages		15
3	Zoo Queueing		20
4	Futuristic Ski Rental		20
5	True or False		6
6	Binary Decision Diagram		10
7	CLT logic		6
8	Petri Nets		18
Total			120

1 Regular Languages

(25 points)

1.1 True or False [10 points]

For each of the following statements, indicate whether they are **TRUE** or **FALSE** (circle *one* answer) and briefly justify your answer. Answers without justification do not give any point.

- a) **TRUE** **FALSE** | If L_1 is regular, then $L_2 = \{ab \mid a \in L_1 \wedge b \notin L_1\}$ is regular.
- b) **TRUE** **FALSE** | If $L = L_1 \cup L_2$ is a regular language *and* L_1 is a regular language, then L_2 is a regular language.
- c) **TRUE** **FALSE** | If $L_1 \subseteq L_2$ *and* L_1 is a regular language, then L_2 is a regular language.
- d) **TRUE** **FALSE** | If $L_1 \subseteq L_2$ *and* L_2 is a regular language, then L_1 is a regular language.
- e) **TRUE** **FALSE** | If L satisfies the pumping lemma, then L is regular.

1.2 Draw me a NFA [5 points]

Assume that the alphabet Σ is $\{0, 1\}$ and consider the language $L = \{w \mid \text{there exists two zeros in } w \text{ that are separated by a string whose length is } 4i \text{ for some } i \geq 0\}$. For example the strings 1001 and 10110101 belong to L , whereas the strings 101 and 010101 do not. Design a NFA that recognizes L with 6 states or less.

1.3 The art of being regular [5 points]

Assume that the alphabet Σ is $\{0, 1\}$ and consider the language $L = \{x\#y \mid x+y = 3y\}$ in which x and y are binary numbers. For instance, the string $1000\#100$ belongs to L . Is L regular? If so, exhibit a finite automaton (deterministic or not) or a regular expression recognizing it. If not, prove it formally using the pumping lemma or the closure properties of regular languages.

1.4 Zero-sum languages [5 points]

Assume that the alphabet Σ is $\{0, 1\}$ and consider any language L defined over it. We define a new operation *zero* which takes as input a language L and for each string $w \in L$ replaces it with a string of 0s with the same length as w . That is, $zero(L) = \{0^{|w|} \mid w \in L\}$.

Are regular languages *closed* under the *zero* operation? That is, if L_1 is a regular language then is $zero(L_1)$ also regular? Justify your answer. There is no need to give a formal proof, a description of your reasoning is enough.

2 Context-free languages

(15 points)

2.1 Draw me a PDA [5 points]

Assume that the alphabet $\Sigma = \{0, 1\}$ and consider the following language:

$$L = \{w \in \Sigma^* \mid w \text{ is a palindrome such that } |w| \text{ is divisible by 4 and } w \text{ ends with } 1\}$$

Design a PDA that accepts L . The designed PDA must have 9 or *less* states. Hint: example strings recognized by L are: 1111, 1001, and 11000011.

2.2 Write me a grammar [10 points]

Most people rely on infix notations when writing logical expressions. When using infix notation, the operator appears in between the operands (*expr operator expr*). For instance, “ $a \vee b$ ” is an expression written in infix notation. In contrast, when using postfix notation, the operator appears before the operands (*operator expr expr*). For instance, “ $\vee a b$ ” is an expression written in postfix notation.

Consider the language L representing the set of all logical expressions written using postfix notations. The alphabet of L is composed of *true*, *false* and three operators: \wedge (and), \vee (or), and \neg (not). Logical expressions evaluate to *true* or *false*. For example, the following expressions evaluate to *true*: “*true*”, “ \neg *false*”, “ \vee *false true*”, and “ \vee \neg *false* \wedge *true false*” (which is equivalent to “ $(\neg$ *false*) \vee (*true* \wedge *false*)” in infix). In contrast, the following expressions evaluate to *false*: “*false*” and “ \vee \neg *true* \wedge *true false*”.

Give a context-free grammar G for the subset of L that contains *all* logical expressions which evaluate to *true*.

3 Zoo Queueing

(20 points)

A zoo has 2 ticket offices (left and right). People arrive at the zoo according to a Poisson process with parameter λ . Each ticket office has a service rate of μ (exponentially distributed).

- a) [5] Assume that visitors are ordered and numbered according to their arrival time. A manager of the zoo proposes that odd numbered visitors go to the left ticket office and even numbered visitors go right. The manager claims that in this way people still arrive at the left ticket office according to a Poisson process. Is he correct? If yes, find the parameter of the new Poisson process. If no, explain why.

Assume now there is only a single queue in front of the two ticket offices.

b) [3] What is the condition so that the waiting queue in front of the ticket office does not grow indefinitely?

c) [5] What is the expected time a person needs until getting a ticket, including waiting time? (Assume stationary distribution and a stable system.)

The zoo has n animal houses. The time that a person spends on every house is exponentially distributed with parameter ν , and every house allows infinite visitors. Every person uniformly randomly chooses an order to visit all the houses after buying a ticket, and we ignore the walking time between any two houses.

d) [4] Let the random variable $X_i, 1 \leq i \leq n$ be the i -th visited house. Is X_1, X_2, \dots, X_n a Markov chain?

e) [3] What is the expected time a person spends inside the zoo (without the ticket office)?

(Additional space for question 3)

4 Futuristic Ski Rental

(20 points)

We consider the online ski rental problem. The ski equipment can be bought for a cost 1 and can be rented at the cost of t for duration t , where $t > 0$. At a given time t , the decision whether the ski should be bought or continued to be rented is taken by the online algorithm. The adversary chooses the time u at which the person gives up skiing.

Now, there is a twist: The person renting or buying the ski has some power to predict the future. At time t , the online algorithm knows whether or not the skiing activity will have stopped by the time $t + f$, where f is the duration that can be foreseen by the online algorithm.

- a) [6] Design an online algorithm for $f = 1$. What is the competitive ratio achieved?

b) [6] Design an online algorithm for $f = 1/2$. What is the competitive ratio achieved?

c) [8] Give a lower bound on the competitive ratio for $f = 1/2$.

5 True or False

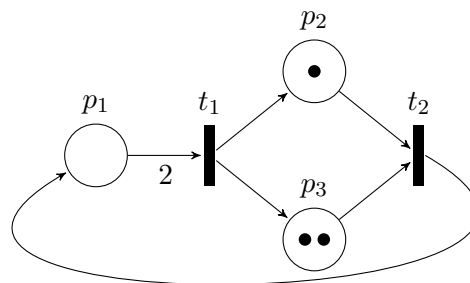
(6 points)

For each of the following statements, assess if it is true or false and tick the corresponding box. No justification is needed.

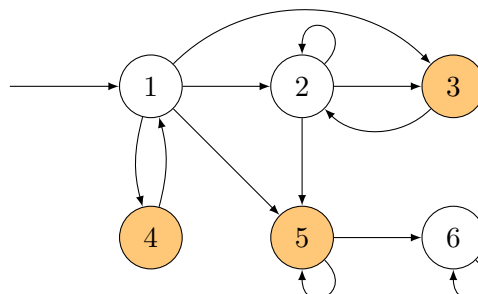
Every correct answer grants one point. Leaving a statement blank gives 0 point. Every incorrect answer loses one point on the total, with a minimum of 0 point for the whole question.

- Notes:**
- Petri net \mathbf{P} is depicted in Figure 1(a).
 - Automaton \mathbf{A} is depicted in Figure 1(b).
 - $\llbracket p \rrbracket$ denotes the set of states which satisfies property p .
For example for automaton \mathbf{A} , $\llbracket p \rrbracket = \{3, 4, 5\}$.

	Statement	True	False
1	Given an ordering of the variables, the Reduced Binary Decision Diagram of a boolean function is unique.		
2	Regular (i.e., non-timed) Petri nets, e.g., \mathbf{P} , are more expressive than deterministic finite automata.		
3	The state of Petri net \mathbf{P} is fully described by its marking.		
4	For automaton \mathbf{A} , $\llbracket EX p \rrbracket = \{1, 2, 5\}$.		
5	For automaton \mathbf{A} , $\llbracket EX ((EG \bar{p}) \text{ AND } (EF p)) \rrbracket = \{1, 2, 3, 4\}$.		
6	Automaton \mathbf{A} satisfies $AX p$.		



(a) Petri net \mathbf{P}



(b) Automaton \mathbf{A} – 1 is the initial state. 2, 5 and 6 are accepting (or final) states. Property p is true only in states 3, 4, and 5.

Figure 1: Petri net \mathbf{P} (1(a)) and Automaton \mathbf{A} (1(b))

6 Binary Decision Diagram

(10 points)

- a) [6] Given the boolean expression of function f and the ordering of variables $x_1 < x_2 < x_3 < x_4$ (i.e., x_1 is the first variable), construct the BDD (Binary Decision Diagram) of f . Merge all equivalent nodes, including the leaves.

Note Use solid lines for *True* arcs and dashed lines for *False* arcs.

$$f : x_1 \cdot x_2 \cdot (\overline{x_3} + x_3 \cdot \overline{x_4}) + \overline{x_2 \cdot x_3 + \overline{x_1} \cdot \overline{x_2} \cdot x_3 \cdot x_4}$$

b) [2] Consider the BDD of the function g in Figure 2. Express g as a boolean function.

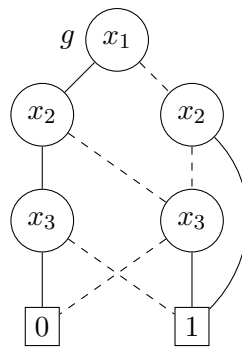


Figure 2: BDD of the boolean function g

c) [2] Simplify the BDD of g (Figure 2) when $x_3 = 0$.

7 CTL logic

(6 points)

We look at the execution of a computer program and define two atomic properties:

- e : the program is in an error state.
- r : the program resets to its initial state.

For each of the following sub-questions, write the given property as a CTL formula.

Reminder CTL formula can contain boolean operators (e.g., AND or NOT) and logical connectors (e.g., the inclusion, denoted by \Rightarrow).

- a) [2] From its initial state, the program never enters an error state.
- b) [2] From the program's initial state, there is a sequence of transitions such that the following holds: From every visited state in the sequence, all possible sequences of transitions will lead to a program reset.
- c) [2] If the program enters an error state, then at some point in the future the program will reset.

8 Petri nets

(18 points)

This question contains 3 independent sub-questions related to Petri nets. Throughout this question, we use the following notations.

- X^t denotes the transpose of vector X .
- $M^t = [p_1, p_2, p_3, p_4]$ and $T^t = [t_1, t_2, t_3, t_4, t_5, t_6]$ are marking and firing vectors of \mathbf{P} respectively.
 p_i denotes the number of tokens in place i .
 t_i denotes the number of firings of transition i .

Let us first consider the Petri net $\mathbf{P1}$ in Figure 3.

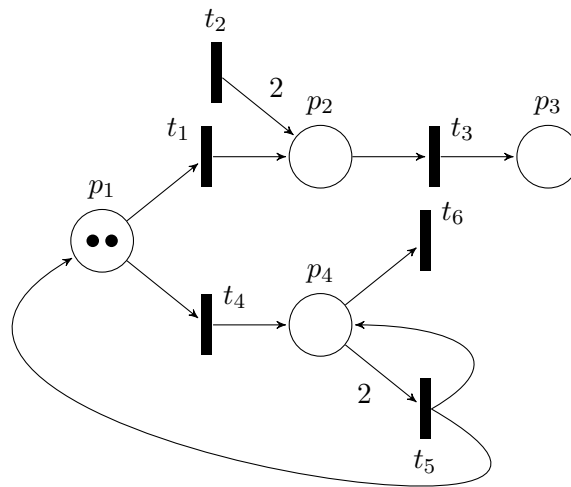


Figure 3: Petri net $\mathbf{P1}$ – Circles, dots and bars represent places, tokens and transitions, respectively. Arc's weights are marked close by the arc when they are different from 1.

8.1 Reachability [7 points]

- a) [2] Derive the incidence matrix A of the Petri net $\mathbf{P1}$ from Figure 3.

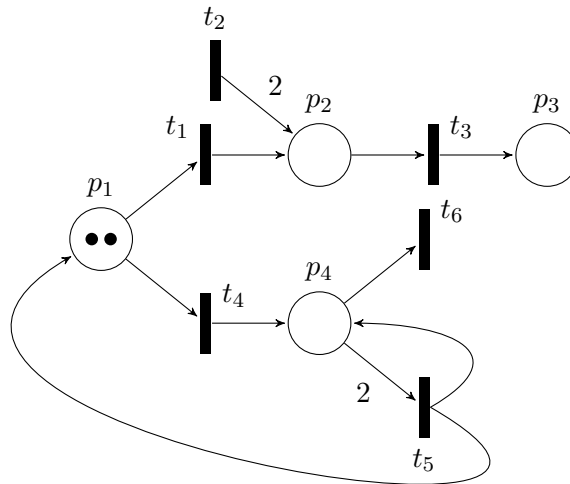
- b) [2] Consider the firing vector $T_S^t = [0, 2, 3, 1, 1, 0]$, where S denotes a firing sequence containing t_2 twice, t_3 three times, t_4 once and t_5 once. Use the incidence matrix and the state equation of the Petri net **P1** to compute the marking M_1 obtained from the initial marking $M_0^t = [2, 0, 0, 0]$ after firing S .

- c) [1] Assume M_1 is a valid marking. Is it sufficient to conclude that S is a valid firing sequence with respect to M_0 ? Why?

- d) [2] Does there exist a valid firing sequence S having $T_S^t = [0, 2, 3, 1, 1, 0]$ as firing vector? Give one or explain why there is none.

8.2 Capacity [1 point]

The Petri net **P1** from Figure 3 is redrawn below. Modify it to create a capacity constraint of 3 tokens in place p_2 . The rest of the net must otherwise remain unchanged.



b) [1] From the coverability graph, can you conclude whether or not the marking

$$M^t = [7, 10, 2]$$

is reachable from the initial marking $M_0^t = [1, 0, 0]$? Justify.

c) [2] Is $M^t = [7, 10, 2]$ reachable from the initial marking $M_0^t = [1, 0, 0]$? Justify.

