

# Prüfung Diskrete Ereignis Systeme

Samstag, 30. Januar 2010  
9:00 – 12:00**Nicht öffnen oder umdrehen bevor die Prüfung beginnt!**

Die Prüfung dauert 180 Minuten und es gibt insgesamt 180 Punkte. Die Anzahl Punkte pro Teilaufgabe steht jeweils in Klammern bei der Aufgabe. Sie dürfen die Prüfung auf Englisch oder Deutsch beantworten. Begründen Sie alle Ihre Antworten und beschriften Sie Skizzen und Zeichnungen verständlich. Schreiben Sie zu Beginn Ihren Namen und Ihre Legi-Nummer in das folgende dafür vorgesehene Feld.

|      |          |
|------|----------|
| Name | Legi-Nr. |
| <br> | <br>     |

## Punkte

| Frage Nr. | Erreichte Punkte | Maximale Punkte |
|-----------|------------------|-----------------|
| 1         |                  | 58              |
| 2         |                  | 45              |
| 3         |                  | 33              |
| 4         |                  | 44              |
| Total     |                  | 180             |



# 1 Sprachen (58 Punkte)

A) Konstruieren Sie jeweils einen *deterministischen* endlichen Automaten (DFA), der über das Alphabet  $\Sigma = \{d, e, s\}$  folgende Sprachen akzeptiert:

*Hinweis:* Sie dürfen Fail-States weglassen.

- 1) [5]  $\mathcal{L}_1$  ist die Menge aller Wörter, welche das Teilwort ‘des’ mindestens zwei Mal enthalten.
- 2) [5]  $\mathcal{L}_2$  ist die Menge aller Wörter, welche durch den folgenden regulären Ausdruck beschrieben werden:  $(d(d \cup e)^*d) \cup (e(d \cup e)^*e)$ .
- 3) [5]  $\mathcal{L}_3$  ist die Menge aller Wörter, welche mit ‘des’ beginnen und auf ‘sed’ enden.

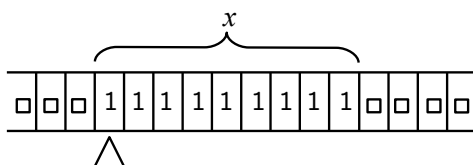
B) Entscheiden Sie für folgende Sprachen, ob sie regulär, kontextfrei, beides oder keines von beidem sind. Beweisen Sie Ihre Antworten.

- 1) [5]  $\mathcal{L}_4 = \{1^a(1 \cup \epsilon) \mid a \bmod 4 = 1\}$   
*Hinweis:* Der Modulo-Operator *mod* berechnet den Rest aus der Ganzzahldivision. Zum Beispiel ist  $14 \bmod 3 = 2$ , da  $14 = 4 \cdot 3 + 2$ .
- 2) [8]  $\mathcal{L}_5 = \{1^a 0^b 1^c \mid a \geq 1 \text{ und } b = c\}$
- 3) [5]  $\mathcal{L}_7 = \{1010^k \mid k = \lceil \sqrt{n} \rceil \text{ und } n \in \mathbb{N}_0\}$

C) Konstruieren Sie eine Turing-Maschine  $M$ , die eine Zahl  $x$  in unärer Codierung (bspw.  $6 \mapsto 111111$ )  $\bmod 5$  rechnet. Zu Beginn enthält das Band die unär codierte Zahl  $x$ , und der Lesekopf zeigt auf die erste Eins. Nach der Berechnung soll das Band nur noch das ebenfalls unär codierte Resultat  $y = x \bmod 5$  enthalten und der Lesekopf soll auf die Eins ganz links zeigen. Ist  $y = 0$ , soll das Band leer sein.

- 1) [10] Beschreiben Sie die grobe Funktionsweise Ihrer Maschine in 1-2 Sätzen und geben Sie einen DFA mit möglichst wenigen Zuständen an, der  $M$  steuert. Nehmen Sie dabei an, dass  $M$  nur Symbole lesen und löschen, aber keine neuen Symbole aufs Band schreiben kann. Benutzen Sie folgende Notation für Ihre Transitionen:

- ‘ $\alpha|\gamma$ ’      Lies  $\alpha$  an der aktuellen Position und bewege danach den Lesekopf nach links wenn  $\gamma = L$  oder nach rechts wenn  $\gamma = R$ .
- ‘ $\alpha \rightarrow \square|\gamma$ ’      Lies  $\alpha$  an der aktuellen Position, lösche es und bewege den Lesekopf in Richtung  $\gamma$ .



Turing-Band mit dem Lesekopf  $\Delta$ , sowie leeren Zellen (Blank-Symbol  $\square$ ) links und rechts vom Wort, das untersucht werden soll.

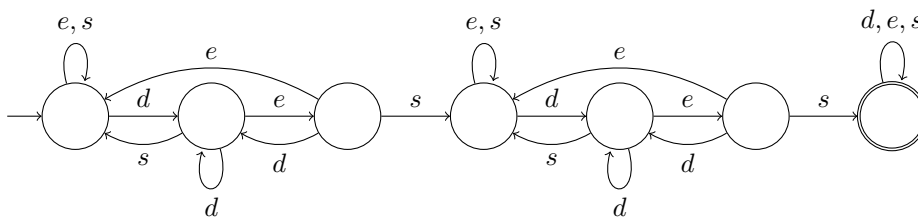
- 2) [3] Erläutern Sie den letzten Satz des folgenden Zitats:  
*“Man kann [...] eine universelle Turingmaschine definieren, welche die Kodierung einer Turingmaschine als Teil ihrer Eingabe nimmt und das Verhalten der kodierten Turingmaschine auf der ebenfalls gegebenen Eingabe simuliert. Aus der Existenz einer solchen universellen Turingmaschine folgt zum Beispiel die Unentscheidbarkeit des Halteproblems.”*<sup>1</sup>

D) Überprüfen Sie folgende Aussagen auf ihre Richtigkeit. Geben Sie für jede Aussage entweder einen Beweis oder ein Gegenbeispiel an.

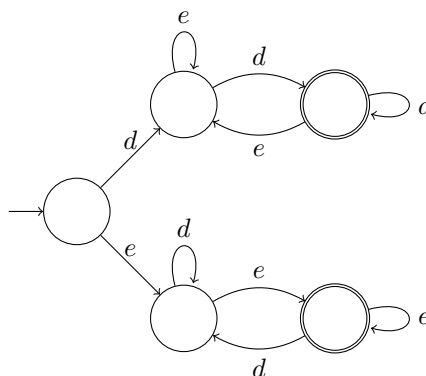
- 1) [6] “ $L_1, L_2$  sind kontextfrei  $\Rightarrow L_3 = (L_1 \cup L_2)$  ist kontextfrei”
- 2) [6] “ $L_1, L_2$  sind kontextfrei  $\Rightarrow L_3 = (L_1 \cap L_2)$  ist kontextfrei”

<sup>1</sup><http://de.wikipedia.org/wiki/Turingmaschine>

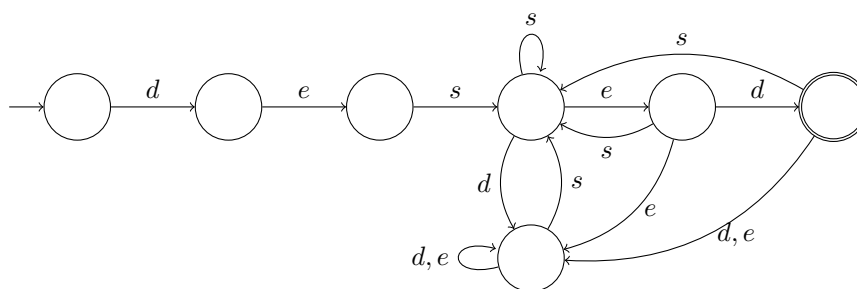
A) 1)



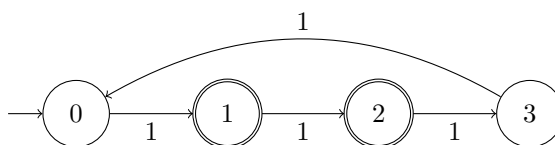
2)



3)



B) 1) Folgender FA erkennt  $\mathcal{L}_4$ :

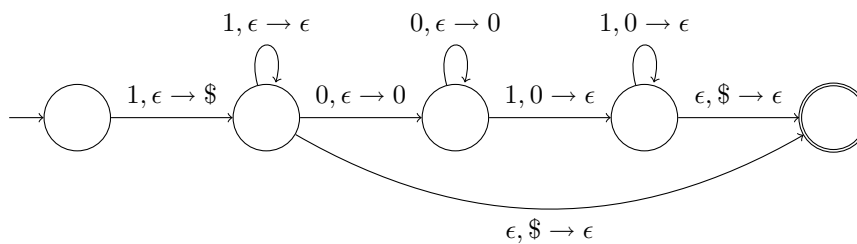


Folglich ist  $\mathcal{L}_4$  regulär. Da alle regulären Sprachen auch kontextfrei sind, ist  $\mathcal{L}_4$  auch kontextfrei.

2)  $\mathcal{L}_5$  ist *kontextfrei*, denn wir können dafür eine CFG oder einen PDA angeben:

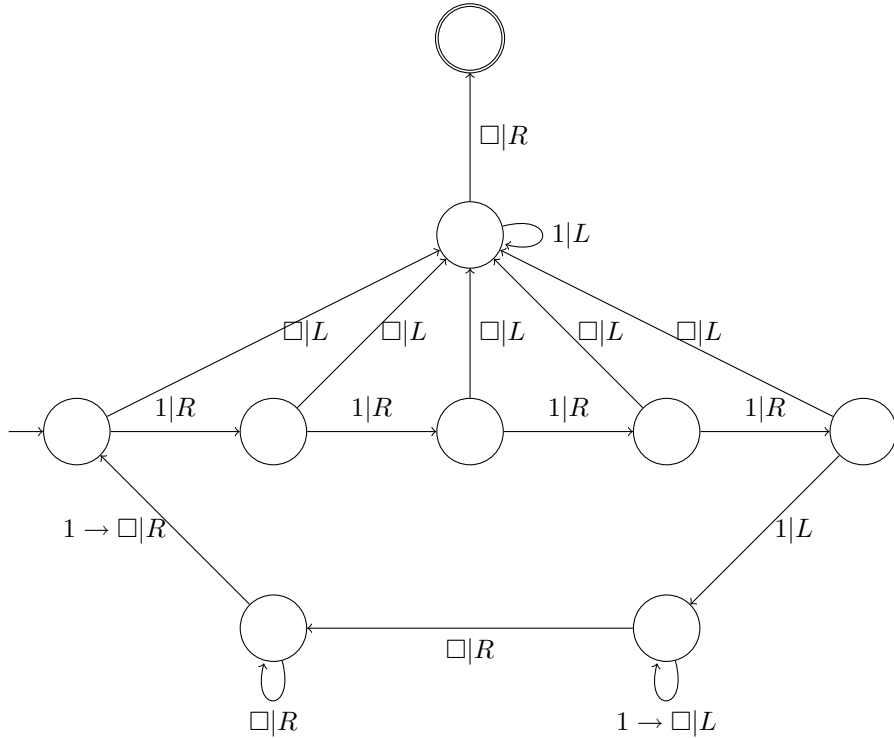
$$S \rightarrow 1A \mid 1S$$

$$A \rightarrow 0A1 \mid \epsilon$$



$\mathcal{L}_5$  ist aber *nicht regulär*. Beweis mittels Pumping Lemma. Angenommen  $\mathcal{L}_5$  wäre regulär, so wäre jedes genug grosse Wort  $w \in \mathcal{L}_5$  pumpbar. Wähle parametrisiertes Wort  $w = 10^p 1^p$ . Gemäss Pumping Lemma müsste  $w$  in  $xyz$  aufteilbar sein, so dass  $|xy| \leq p$ ,  $y \geq 1$  und  $xy^i z \in \mathcal{L}_5$ . Falls  $xy = 1$ , so ist  $xy^0 z = 0^p 1^p \notin \mathcal{L}_5$ . Falls  $|xy| > 1$ , so enthält  $y$  mindestens eine 0. In einem resultierenden Wort  $w' = xy^i z$  mit  $i \geq 2$  wird die Anzahl der 0en also erhöht, wobei die Anzahl der 1en im  $z$ -Teil des Wortes unverändert bleibt. In  $w'$  ist also  $b > c$ . Folglich gilt  $w' \notin \mathcal{L}_5$ . Das Wort  $w$  ist nicht pumpbar. Dies ist ein Widerspruch zur Annahme  $\mathcal{L}_5$  sei regulär und folglich ist  $\mathcal{L}_5$  nicht regulär.

- 3) Die Quadratwurzelfunktion  $\sqrt{n}$  ist sublinear.  $k$  kann alle Werte in  $\mathbb{N}_0$  annehmen.  $\mathcal{L}_6$  ist deshalb äquivalent zur regulären Expression  $1010^*$  und somit regulär sowie kontextfrei.
- C) 1)  $M$  überprüft, ob noch fünf 1en hier sind. Wenn ja, löscht sie diese und wiederholt diesen Vorgang bis nur noch das Resultat dasteht.



- 2) Erst durch die Existenz der universellen Turingmaschine lässt sich der Widerspruchsbeweis aus Übung 6.1 führen. Angenommen es gäbe eine TM  $H$  die für eine kodierte TM  $P$  entscheidet, ob  $P$  hält oder nicht, so können wir eine TM  $X$  bauen, welche  $H$  mit sich selbst als Input simuliert. Falls  $H$  positiv entscheidet, soll  $X$  ewig loopen und sonst terminieren. Dies führt zum erwünschten Widerspruch.
- D) 1) Richtig. Sei  $S_1$  das Startsymbol der CFG zu  $L_1$  in Normalform, bzw.  $S_2$  das Startsymbol der CFG zu  $L_2$ . Weiter nehmen wir o.B.d.A. an, die Schnittmenge der verwendeten Nicht-Terminal Symbole beider CFGs sei leer. Eine CFG zu  $L_3$  ist dann gegeben durch die Vereinigung aller Regeln der beiden CFGs zu  $L_1$  und  $L_2$  und der Startregel  $S \rightarrow S_1 \mid S_2$ .
- 2) Falsch. Die Sprachen  $L_1 = \{1^m 2^n 3^n \mid m, n > 0\}$  und  $L_2 = \{1^m 2^m 3^n \mid m, n > 0\}$  sind beide kontextfrei. Ihr Schnitt  $L_3 = (L_1 \cap L_2) = \{1^k 2^k 3^k \mid k > 0\}$  ist aber nicht kontextfrei (Slide 2/46).

## 2 Am Flughafen (45 Punkte)

A) Bei der Sicherheitskontrolle am Flughafen kommt im Durchschnitt alle 4 Minuten ein Passagier an. Dort steht er in der Schlange bis er an der Reihe ist und mit dem neuen Nacktscanner durchleuchtet wird. Es dauert durchschnittlich 2 Minuten, einen Passagier zu scannen. Nehmen Sie an, die Ankunfts- und Bedienzeiten seien poissonverteilt.

- 1) [2] Modellieren Sie die Situation als Markov-Kette.
- 2) [2] Wie lange dauert es im Durchschnitt insgesamt, anzustehen und gescannt zu werden?
- 3) [2] Wie lange muss ein Passagier im Erwartungswert in der Schlange anstehen?

Wegen des grossen Erfolgs des ersten Nacktscanners wird nun ein zweiter angeschafft. Dieser wird immer dann in Betrieb genommen, sobald in der Schlange 3 Personen stehen. Die Passagiere stehen weiterhin in *einer* Schlange für beide Scanner an. Wenn nach den Scannen eines Passagiers nur noch höchstens zwei Personen in der Schlange stehen, wird einer der beiden Scanner wieder abgeschaltet.

- 4) [6] Modellieren Sie die Situation als Markov-Kette.
- 5) [6] Angenommen, die Wahrscheinlichkeit, dass sich niemand im System befindet, sei im stationären Zustand  $p$ . Berechnen Sie die Wahrscheinlichkeit im stationären Zustand, dass genau 2 Personen in der Schlange stehen, in Abhängigkeit von  $p$ .
- 6) [6] Berechnen Sie  $p$ .

*Hinweis:*

$$\sum_{i=0}^{\infty} c^i = \frac{1}{1-c}.$$

Da mit der oben beschriebenen Methode ein Nacktscanner immer wieder an- und abgeschaltet werden musste, geht das Sicherheitspersonal zu folgender Variante über: Es wird weiterhin der zweite Scanner eingeschaltet, sobald 3 Personen in der Schlange anstehen. Aber erst wenn keine Passagiere mehr in der Schlange stehen und ein Scanner gerade nicht benutzt wird, wird dieser wieder abgeschaltet.

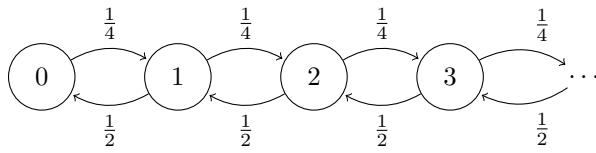
- 7) [6] Modellieren Sie die Situation als Markov-Kette.

B) 100 Passagiere gehen nacheinander an Bord eines vollständig ausgebuchten Flugzeugs. Der erste Passagier kann sich nicht mehr daran erinnern, welcher sein Platz ist, und wählt zufällig gleichverteilt irgendeinen der 100 Plätze. Die restlichen Passagiere nehmen nun, wenn sie an Bord gehen, ihren Platz ein, falls er noch frei ist. Ansonsten wählen sie aus den übrigen freien Plätzen zufällig gleichverteilt einen aus.

- 1) [5] Zeigen Sie, dass von den Plätzen der draussen wartenden Passagiere zu jeder Zeit höchstens einer falsch besetzt ist.
- 2) [10] Wie hoch ist die Wahrscheinlichkeit, dass der letzte Passagier seinen eigenen Platz bekommt?

*Hinweis:* Modellieren Sie das Szenario mit einer *zeitinhomogenen* Markov-Kette mit höchstens 5 Zuständen!

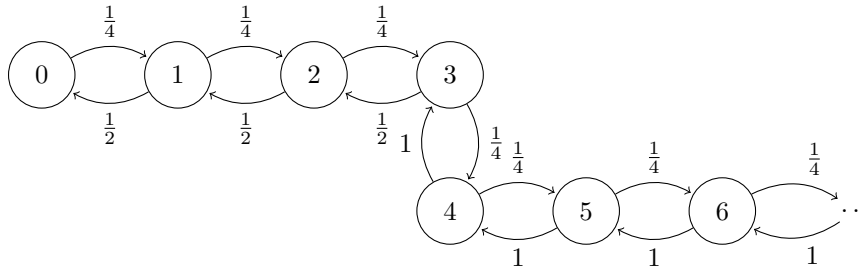
A) 1)



2) Antwortzeit  $T = \frac{1}{\mu - \lambda} = \frac{1}{\frac{1}{2} - \frac{1}{4}} = 4$

3) Wartezeit  $W = \frac{\rho}{\mu - \lambda} = \frac{1}{2} \cdot 4 = 2$

4)



5) Es gibt zwei Möglichkeiten, wann genau zwei Personen in der Schlange stehen, nämlich mit einem oder mit zwei eingeschalteten Scannern. Diese beiden Möglichkeiten werden durch die Zustände 3 und 4 der Markov-Kette aus Teilaufgabe 1d repräsentiert. Also ist die Wahrscheinlichkeit, dass genau 2 Personen in der Schlange stehen,  $q = \pi_3 + \pi_4$ . Natürlich gilt  $p = \pi_0$ , wir berechnen also die Wahrscheinlichkeit  $q$  in Abhängigkeit von  $\pi_0$ :

$$\begin{aligned} \pi_1 &= \frac{1}{2}\pi_0 \\ \pi_2 &= \frac{1}{2}\pi_1 = \frac{1}{4}\pi_0 \\ \pi_3 &= \frac{1}{2}\pi_2 = \frac{1}{8}\pi_0 \\ \left(\frac{1}{4} + \frac{1}{2}\right)\pi_3 &= \frac{1}{4}\pi_2 + 1\pi_4 \\ \Rightarrow \pi_4 &= \frac{3}{4}\pi_3 - \frac{1}{4}\pi_2 = \frac{3}{4}\pi_3 - \frac{2}{4}\pi_3 = \frac{1}{4}\pi_3 = \frac{1}{32}\pi_0 \end{aligned}$$

Also ergibt sich:

$$q = \pi_3 + \pi_4 = \frac{1}{8}\pi_0 + \frac{1}{32}\pi_0 = \frac{5}{32}\pi_0.$$

6) Es ist

$$\sum_{i=0}^{\infty} \pi_i = 1 \Leftrightarrow \pi_0 = \frac{1}{\sum_{i=0}^{\infty} \frac{\pi_i}{\pi_0}} = \frac{1}{1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \sum_{i=4}^{\infty} \frac{\pi_i}{\pi_0}}.$$

Wir haben  $\pi_1, \pi_2, \pi_3$  und  $\pi_4$  bereits in Abhängigkeit von  $\pi_0$  berechnet. Für  $\pi_5$  ergibt sich:

$$\begin{aligned} \left(\frac{1}{4} + 1\right)\pi_4 &= \frac{1}{4}\pi_3 + 1\pi_5 \\ \Rightarrow \pi_5 &= \frac{5}{4}\pi_4 - \frac{1}{4}\pi_3 = \frac{5}{4}\pi_4 - \frac{1}{4} \cdot 4\pi_4 = \frac{1}{4}\pi_4. \end{aligned}$$

Man erkennt nun leicht, dass sich für alle  $k \geq 4$  ergibt:

$$\pi_k = \frac{1}{4} \pi_{k-1} = \frac{1}{4^{k-3}} \pi_3 = 4^3 \cdot \frac{1}{4^k} \cdot \frac{1}{8} \pi_0.$$

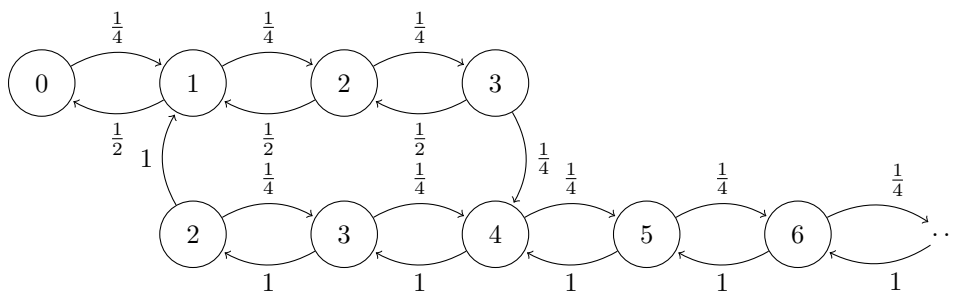
Damit erhalten wir

$$\begin{aligned} \sum_{i=4}^{\infty} \frac{\pi_i}{\pi_0} &= \frac{1}{8} \cdot 4^3 \cdot \sum_{k=4}^{\infty} \frac{1}{4^k} = 8 \cdot \left( \frac{1}{1 - \frac{1}{4}} - \frac{1}{4^0} - \frac{1}{4^1} - \frac{1}{4^2} - \frac{1}{4^3} \right) \\ &= 8 \cdot \frac{4}{3} - 8 \cdot \frac{64 + 16 + 4 + 1}{64} = \frac{32}{3} - \frac{85}{8} = \frac{256}{24} - \frac{255}{24} = \frac{1}{24}. \end{aligned}$$

Dann ist also die Wahrscheinlichkeit, dass sich niemand im System befindet (Schlange + Scanner), gleich

$$\pi_0 = \frac{1}{\frac{15}{8} + \frac{1}{24}} = \frac{1}{\frac{45}{24} + \frac{1}{24}} = \frac{24}{46} = \frac{12}{23}.$$

7)



- B) 1) Wir beweisen die Aussage per Induktion. Wir nehmen an, die Passagiere kommen in der Reihenfolge 1, 2, 3, ..., 99, 100 an Bord.

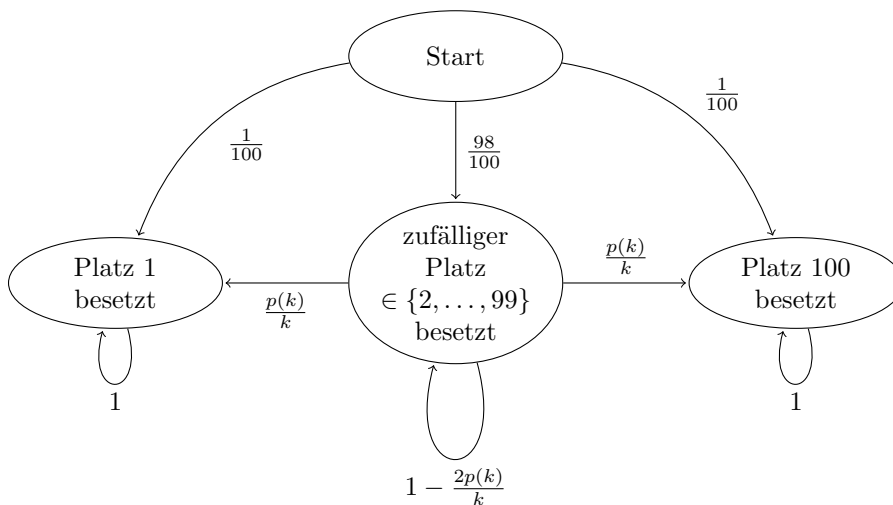
*Induktionsanfang:*

Wenn alle Passagiere draussen stehen, ist noch gar kein Platz besetzt und die Annahme ist trivial erfüllt.

*Induktionsschritt:*

Wir überlegen uns, was passiert, wenn der  $i$ -te Passagier an Bord geht. Es gibt zwei Möglichkeiten: Entweder sein Platz ist frei, dann setzt er sich auf den richtigen Platz und die Anzahl der falsch besetzten Plätze bleibt gleich. Oder sein Platz ist bereits besetzt. Dann muss sich Passagier  $i$  auf den Platz eines Passagiers  $j$ ,  $j > i$  setzen, und es wird ein zusätzlicher Platz falsch besetzt. Da aber vorher der Platz von Passagier  $i$  laut Induktionsvoraussetzung der einzige falsch besetzte Platz der noch draussen wartenden Passagiere gewesen sein muss und Passagier  $i$  nun nicht mehr draussen wartet, ist die Induktionsvoraussetzung wieder erfüllt. Nun ist der Platz von Passagier  $j$  der einzige falsch besetzte Platz der Passagiere, die draussen warten.

2)





Sobald sich ein Passagier auf Platz 100 setzt, muss sich der 100. Passagier auf einen fremden Platz setzen. Sobald sich jemand auf Platz 1 setzt, ist kein Platz der noch draussen wartenden Passagiere mehr besetzt und alle, die jetzt an Bord gehen, bekommen ihren eigenen Platz, insbesondere auch Passagier 100.

Der erste Passagier setzt sich jeweils mit Wahrscheinlichkeit  $\frac{1}{100}$  auf Platz 1 bzw. 100, ansonsten besetzt er zufällig gleichverteilt einen der Plätze  $2, \dots, 99$ . Angenommen, es ist ein Platz aus  $2, \dots, 99$  falsch besetzt und es sind noch  $k$  Plätze frei, das heisst, als nächstes kommt Passagier  $100 - k + 1$  an Bord. Jetzt gibt es zwei Möglichkeiten: Erstens, sein Platz ist der fälschlicherweise besetzte. Das ist mit Wahrscheinlichkeit  $p(k)$  der Fall. Dann setzt sich Passagier  $100 - k + 1$  jeweils mit Wahrscheinlichkeit  $1/k$  auf Platz 1 bzw. Platz 100, mit Wahrscheinlichkeit  $(k - 2)/k$  besetzt er einen der anderen freien Plätze. Zweitens, sein Platz ist noch frei, was mit Wahrscheinlichkeit  $1 - p(k)$  passiert. Dann setzt er sich auf jeden Fall auf seinen eigenen Platz. Insgesamt bleibt die Kette also im Zustand "zufälliger Platz" mit Wahrscheinlichkeit  $p(k) \cdot \frac{k-2}{k} + 1 - p(k) = 1 - \frac{2p(k)}{k}$  und geht über in Zustand "1" bzw. "100" jeweils mit Wahrscheinlichkeit  $p(k)/k$ .

Die Markov-Kette besitzt zwei absorbierende Zustände. Spätestens nach 99 Schritten befindet man sich in einem dieser beiden Zustände. (Logischerweise muss das so sein, weil nur noch ein Platz frei ist, sobald sich 99 Passagiere gesetzt haben, und somit Platz 1 oder 100 besetzt sein muss. An der Markov-Kette erkennt man dies daran, dass nach 98 Schritten gilt:  $k = 2$ , also sind wir entweder schon in einem absorbierenden Zustände oder es sind nur noch die Plätze 1 und 100 frei. Im letzteren Fall gilt dann aber  $p(k) = 1$ , weil Platz  $100 - k + 1 = 99$  besetzt ist, und somit  $(k - 2p(k))/k = 0$  gilt. Damit ist die Wahrscheinlichkeit, in dem Zustand "zufälliger Platz" zu bleiben (sofern man noch nicht einen absorbierenden Zustand erreicht hat) gleich 0. Das heisst, spätestens einen Schritt später ist entweder Platz 1 oder 100 besetzt.)

Da die Markov-Kette völlig symmetrisch ist und die Wahrscheinlichkeiten, mit denen man in den beiden absorbierenden Zuständen landet, identisch sind, muss am Ende gelten:  $P(\text{Platz 1 besetzt}) = P(\text{Platz 100 besetzt}) = 1/2$ .

### 3 Petrinetze & CTL (33 Punkte)

A) In den folgenden Teilaufgaben soll durch ein Petrinetz jeweils eine Funktion  $f_i(x, y)$  simuliert werden. Das heisst, das Netz soll zwei Stellen  $P_x$  und  $P_y$  besitzen, auf der zu Anfang  $x$  bzw.  $y$  viele Tokens liegen, und wenn das Netz tot ist, sollen auf einer Stelle  $P_z$   $f_i(x, y)$  viele Tokens liegen. Das Netz soll für eine beliebige Anzahl von Tokens in  $P_x$  und  $P_y$  funktionieren.

1) [3]  $f_1(x, y) = 5x + y \quad \forall x, y \geq 0$

2) [3]  $f_2(x, y) = x - 2y \quad \forall y \geq 0, x \geq 2y$

3) [10]  $f_3(x, y) = x \cdot y \quad \forall x, y \geq 0$ . Erklären Sie in einem Satz, wie Ihr Netz funktioniert!

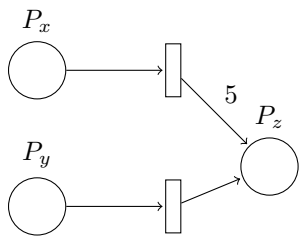
B) Seien die Kripke-Struktur  $\mathcal{K}$  und die Pfadformeln  $\Omega_1$  und  $\Omega_2$  gegeben:

$$\Omega_1 = \exists \square (\neg \exists \bigcirc (b \wedge true)), \quad \Omega_2 = \neg \exists (-d \bigcup c),$$

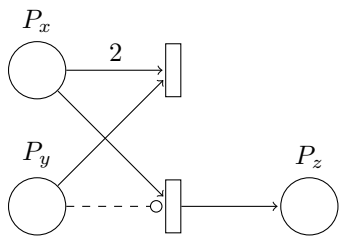
$$\mathcal{K} := \left\{ \begin{array}{l} \mathbb{S} := \{1, 2, 3, 4, 5, 6, 7\}; \\ \mathbb{S}_0 := \{1\}; \\ \mathbb{E} := \{(1, 2), (2, 3), (3, 1), (1, 5), (5, 4), (5, 6), (6, 7), (7, 6)\}; \\ \mathcal{AP} := \{a, b, c, d\}; \\ \mathcal{L} := \{1 \mapsto a, 2 \mapsto a, 3 \mapsto b, 4 \mapsto b, 5 \mapsto d, 6 \mapsto c, 7 \mapsto d\}; \end{array} \right\}.$$

- 1) [3] Zeichnen Sie den Graphen der Kripke-Struktur  $\mathcal{K}$ .
- 2) [3] Zeichnen Sie den Computation-Tree bis zur Tiefe 7 für  $\mathcal{K}$ .
- 3) [9] Konstruieren Sie die Syntaxbäume für  $\Omega_1$  und  $\Omega_2$  und beschriften Sie die Knoten mit den zugehörigen Erfüllbarkeitsmengen bezüglich der Kripke-Struktur  $\mathcal{K}$ .
- 4) [2] Werden  $\Omega_1$  und  $\Omega_2$  von  $\mathcal{K}$  erfüllt? Begründen Sie Ihre Entscheidung!

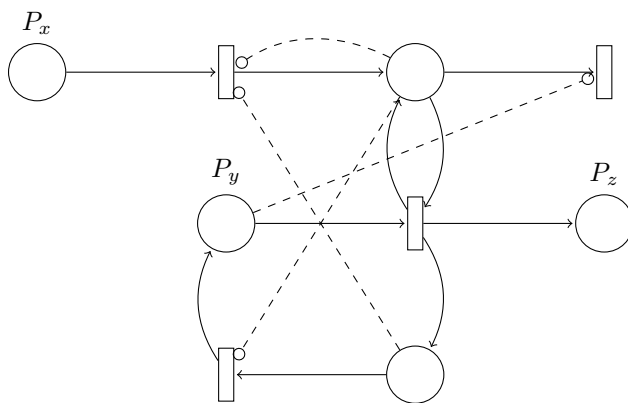
A) 1)



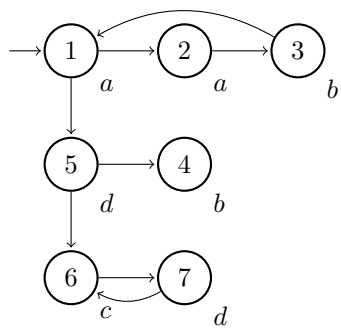
2)



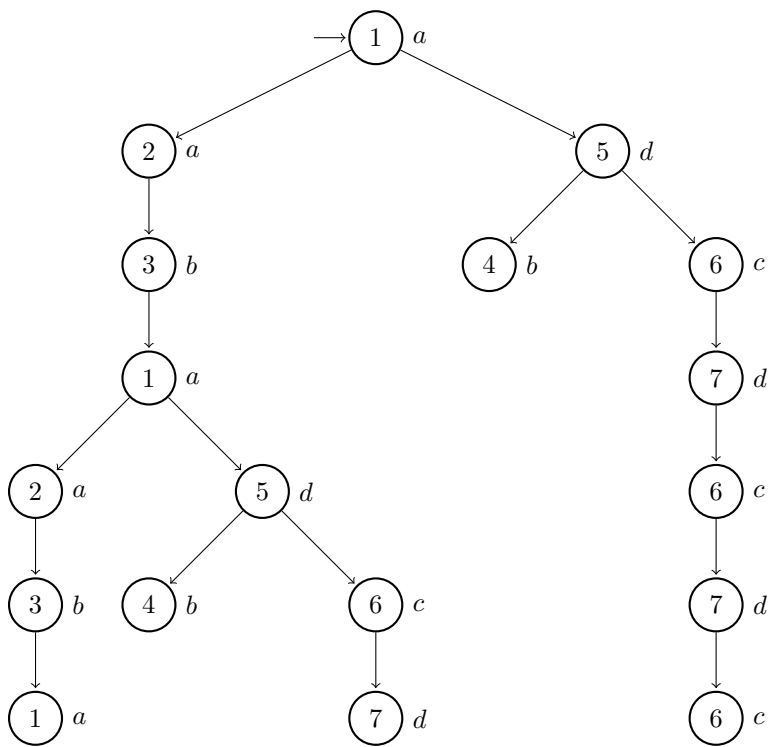
3)



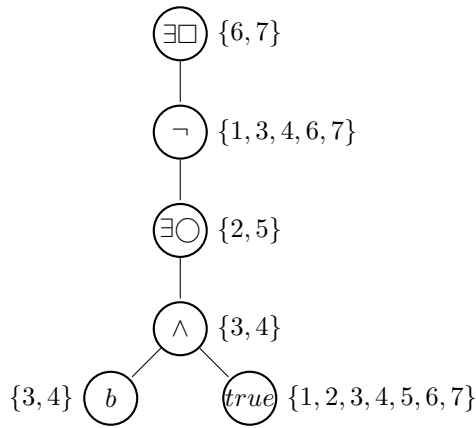
B) 1)



2)

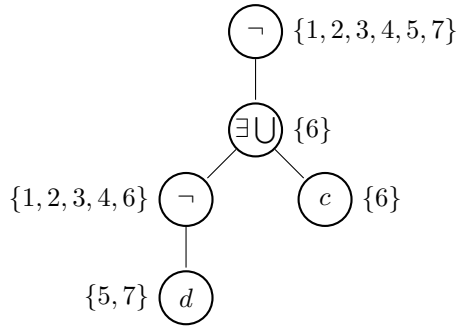


3) Syntaxbaum für  $\Omega_1$ :



- $Sat_0 = \{1, 3, 4, 6, 7\}$
- $Sat_1 = \{3, 6, 7\}$
- $Sat_2 = \{6, 7\}$
- $Sat_3 = \{6, 7\}$

Syntaxbaum für  $\Omega_2$ :



- $Sat_0 = \{6\}$
- $Sat_1 = \{6\}$

- 4) Eine Klausel  $\Omega$  wird genau dann von  $\mathcal{K}$  erfüllt, wenn der Startzustand 1 an der Wurzel des Syntaxbaums zu  $\Omega$  steht. Daher wird  $\Omega_1$  nicht erfüllt,  $\Omega_2$  aber schon.

## 4 Let's Go Surfing (44 Punkte)

- A) Vier Freunde, John, Paul, George und Ringo, gehen an den Strand surfen. Sie wissen, dass die Wellen heute Nachmittag zwischen 1 und 9 Meter hoch sind. Allerdings kann das Wetter unvorhergesehen umschlagen, so dass es von einem Augenblick auf den anderen keine Wellen mehr gibt. Jeder möchte natürlich eine möglichst hohe Welle erwischen. Um für einen bevorstehenden Contest zu üben, vereinbaren sie, dass jeder nur einen Versuch hat, eine möglichst hohe Welle zu erwischen. Sie paddeln also alle raus und entscheiden bei jeder Welle, ob sie sie nehmen oder nicht. Hat sich einer dazu entschlossen eine Welle zu nehmen, lässt er sich von ihr bis zum Strand tragen und wartet dort auf die anderen.

Die vier Surfer wählen ihre Strategien folgendermassen:

- John nimmt die erste Welle mit Höhe  $h \geq 2$  Meter.
- Paul nimmt die erste Welle mit Höhe  $h \geq 4$  Meter.
- George nimmt die erste Welle.
- Ringo wartet  $k \geq 1$  Wellen ab, merkt sich die bisher grösste Höhe und nimmt jetzt die nächste Welle, die mindestens so hoch ist.

Gehen Sie in den folgenden beiden Teilaufgaben davon aus, dass die Surfer nicht erkennen können, ob nach der aktuellen Welle noch eine weitere kommen wird oder nicht.

- 1) [5] Georges Strategie ist unter der Annahme des Worst-Case die beste dieser vier Strategien. Erläutern Sie wieso und berechnen Sie ihre strikte Competitive Ratio!
- 2) [12] Könnte ein Surfer mittels Randomisierung eine Strategie entwickeln, die im Erwartungswert besser ist als Georges Strategie? Geben Sie entweder eine solche randomisierte Strategie an und berechnen Sie deren Competitive Ratio oder beweisen Sie, dass Randomisierung nicht hilft. Nehmen Sie an, dass der Adversary seine Strategie, die Anzahl Wellen und deren Höhen, vor der Ausführung Ihres Algorithmus festlegen muss. Er kann sich also nicht an die randomisierten Entscheidungen Ihrer Strategie anpassen.

Nehmen Sie in den nächsten zwei Teilaufgaben an, dass es die Surfer einer Welle ansehen, ob sie bereits die letzte ist oder nicht.

- 3) [5] Welcher der vier Surfer wählt jetzt die im Worst-Case beste Strategie, wenn jeder spätestens die letzte Welle nimmt?
- 4) [4] Helfen Sie den Surfern! Geben Sie eine im Worst-Case möglichst gute deterministische Strategie an, um eine möglichst hohe Welle zu erwischen. Was ist die Competitive Ratio Ihrer Strategie?

Nehmen Sie nun an, dass unendlich viele Wellen kommen. Die Wellen seien auch nicht mehr nur zwischen 1 und 9 Metern hoch, sondern ihre Höhen seien exponentialverteilt mit Parameter  $\lambda = 1/3$ .

- 5) [8] Welcher der vier Freunde wird nun die erwartungsgemäss höchste Welle erwischen, wenn bei Ringos Strategie  $k = 1$  gilt?

- B) [10] Am nächsten Tag befinden sich die Wellenhöhen wieder in Worst-Case-Manier zwischen 1 und 9 Metern. Nun wollen die Surfer nicht mehr einfach die höchstmögliche Welle erwischen, sondern innert zwei Stunden auf möglichst vielen *Big Waves* surfen. Big Waves sind Wellen mit Höhe  $h \geq 4$  Meter. Die Surfer paddeln jeweils 40 Meter hinaus ins Wasser, warten auf eine geeignete Welle, lassen sich von dieser an den Strand tragen und paddeln wieder raus. Nehmen Sie an, dass 40m vom Strand alle 15s eine Welle kommt.

Wenn ein Surfer eine Welle nimmt, beträgt die Zeit, die er braucht, um ans Ufer zu surfen und wieder 40m aufs Wasser rauszupaddeln,  $t(h)$  Sekunden.  $t(h)$  hängt von der Wellenhöhe  $h$  (in Metern) ab<sup>2</sup> und ist gegeben durch:  $t(h) = 110 + \frac{120}{h}$ .

Geben Sie eine im Worst-Case möglichst gute, deterministische oder randomisierte Strategie an. Wie gross ist ihre Competitive Ratio? Beweisen Sie Ihre Antwort.

---

<sup>2</sup>Je höher eine Welle, desto schneller ist sie.

- A) 1) Georges Strategie ist die einzige kompetitive Strategie. Ihre kompetitive Ratio ist 9, denn im Worst-Case ist die erste Welle  $1m$  und die beste Welle  $9m$  hoch. Die anderen warten auf eine Welle, die eine gewisse Höhe  $h$  hat. Allerdings kann es sein, dass nur kleinere Wellen kommen und sie somit gar keine Welle erwischen.
- 2) Randomisierung hilft. Eine bessere randomisierte Strategie wäre z.B.:
- Nimm erste Welle mit Wahrscheinlichkeit  $p$ .
  - Falls erste Welle nicht genommen, nimm Welle sobald  $h \geq c$ .

Eine erste Strategie des Adversarys wäre, genau eine Welle mit Höhe  $h_1$  zu bringen, die vom Algorithmus mit Wahrscheinlichkeit  $(1 - p)$  nicht genommen wird.

$$\rho_0 = \frac{OPT}{E[ALG]} = \frac{h_1}{p \cdot h_1} = \frac{1}{p}$$

Um besser als der deterministische Algorithmus zu sein, muss also  $p > 1/9$  gewählt werden. Bei mehr als zwei Wellen bleiben dem Adversary zwei Strategien, nachdem der Algorithmus die erste Welle nicht genommen hat. Er kann eine Welle mit Höhe  $c$  bringen und danach eine  $9m$  Welle. So wird  $E[ALG] = p \cdot h_1 + (1 - p) \cdot c$  und  $OPT = 9$ . Andererseits kann er eine Welle mit Höhe  $c - \epsilon$  und keine höhere bringen. Dies ergibt  $E[ALG] = p \cdot h_1$  und  $OPT = c - \epsilon$ . Für beide erreichbaren kompetitiven Ratios ergibt das

$$\rho_1 = \frac{OPT}{E[ALG]} = \frac{9}{p \cdot h_1 + (1 - p) \cdot c}$$

$$\rho_2 = \frac{OPT}{E[ALG]} \leq \frac{c}{p \cdot h_1}$$

Der Adversary will beide Ratios maximieren und setzt  $h_1$  so klein wie möglich, also  $h_1 = 1$ . Für eine geeignete Wahl von  $c$  und  $p$  werden aber trotzdem beide Ratios kleiner als 9. Mit  $p = 1/2$  und  $c = 2$  wird  $\rho_0 = 2$ ,  $\rho_1 = 6$  und  $\rho_2 \leq 4$  und wir erhalten einen 6-kompetitiven Algorithmus.

3)

$$\rho_J = \max \left\{ \frac{2 - \epsilon}{1}, \frac{9}{2} \right\} = 4.5$$

$$\rho_P = \max \left\{ \frac{4 - \epsilon}{1}, \frac{9}{4} \right\} = 4 - \epsilon$$

$$\rho_G = \frac{9}{1} = 9$$

$$\rho_R = \frac{9}{1} = 9$$

Pauls Strategie hat die beste kompetitive Ratio.

4) Wir parametrisieren John und Pauls Strategie:

- Nimm erste Welle mit  $h \geq c$  oder die letzte.

Adversary bringt entweder eine Welle mit  $h = c - \epsilon$  und eine  $1m$ -Welle zuletzt oder er bringt eine Welle mit  $h = c$  und eine  $9m$ -Welle danach. Das ergibt folgende worst-case Ratios:

$$\rho_1 \leq \frac{c}{1}$$

$$\rho_2 = \frac{9}{c}$$

Gleichsetzen und nach  $c$  auflösen ergibt eine maximale Ratio von 3 für  $c = 3$ .

- 5) Wir können ausnützen, dass die Exponentialverteilung gedächtnislos ist. Es gilt:

$$\begin{aligned}F_{X|X>y}(x+y) &= Pr[X \leq x+y | X > y] \\&= 1 - Pr[X > x+y | X > y] \\&= 1 - Pr[X > x] \\&= Pr[X \leq x] \\&= F_X(x)\end{aligned}$$

Für den Erwartungswert erhalten wir folglich  $E[X | X > y] = y + E[X]$ . George erwartet eine Wellenhöhe von  $E[X] = 1/\lambda = 3$ . John kann eine Wellenhöhe von  $E[X | X > 2] = 2 + E[X] = 5$  und Paul eine Höhe von  $4 + 3 = 7$  erwarten. Ringo erreicht im Erwartungswert eine Höhe von  $E[E[X | X > Y]] = E[Y] + E[X] = 3 + 3 = 6$ , wobei  $Y$  die Zufallsvariable der ersten Wellenhöhe ist.

- B) 1) Der Worst-Case aus Sicht des Surfers ist, dass er eine 4-Meter-Welle nimmt und danach gleich eine 9-Meter-Welle kommt, denn mit der 9-Meter Welle wär er schneller wieder im Wasser draussen:  $t(4) = 140s$  und  $t(9) = 110 + 120/9 = 110 + 13.33 \approx 123s$ . Würde er auf die 9-Meter-Welle warten, wäre er also in  $15s + 123s = 138s$  zurück anstatt in  $140s$ . Allerdings würde die 9-Meter-Welle trotzdem keinen Vorteil bringen, denn er würde damit keine frühere Welle erwischen können. Zwischen der 135. und der 149. Sekunde kommt nämlich keine Welle. Die nächste Welle ist in beiden Fällen die Welle nach  $150s$ . Die simple *GREEDY*-Strategie ist also optimal ( $\rho = 1$ ):

- Nimm jeweils die nächste Big Wave.

Gäbe es eine bessere Strategie  $\sigma$ , so müsste sie die *GREEDY*-Strategie irgendwann überholen, d.h. schon gleich viele Big Waves erwischt haben und mindestens eine Welle vor *GREEDY* wieder 40m im Wasser sein. Nimmt  $\sigma$  dieselbe Welle wie *GREEDY*, so ist sie auch zur selben Zeit wieder zurück. Nimmt  $\sigma$  eine Welle nach *GREEDY*, ist sie ebenfalls frühestens in der selben Periode wieder zurück. Da *GREEDY* bereits die allererste Big Wave nimmt, kann  $\sigma$  *GREEDY* nie überholen.