Modelle der Informatik - Lösung der Übungsblätter

Übungsblatt 1 & 2

Aufgabe 1: Begriffe und Definitionen

1. Definieren Sie die Begriffe *Alphabet*, *Wort* und *Sprache*! *(1-2)*

Alphabet: eine Menge von Zeichen (Symbolen, Buchstaben)
Wort: endliche Zeichenkette von Symbolen eines Alphabets
$w = a_1, a_2, a_3, \ldots, a_k$ (ein Wort der Länge k)
leeres Wort = ε (Länge 0)
A^* = Menge aller Worte inklusive ε
A^+ = Menge aller nicht-leeren Worte
Sprache: Teilmenge von A^*

2. Nennen und erläutern Sie die Bestandteile einer *Grammatik*! *(1-8)*

Eine Sprache L(G) wird definiert durch eine Grammatik G = (N, T, P, S), wobei gilt:
N sind die nichtterminalen Symbole (Variable),
T sind die terminalen Symbole (Terminale), wobei $T \cap N = \emptyset$,
P sind die Produktionsregeln der Form $\alpha \rightarrow \beta$, wobei $\alpha \in (N \cup T)^+$, $\beta \in (N \cup T)^*$,
$S \in N$ ist das Startsymbol, ein spezielles nichtterminales Symbol.

3. Wie stehen die Begriffe *Grammatik* und *Sprache* miteinander in Beziehung? *(1-7)*

Eine Grammatik ist ein System zur Ableitung von Worten einer Sprache.

4. Was beschreibt die *Chomsky-Hierarchie*? *(1-15ff)*

Die Chomsky-Hierarchie beschreibt eine Hierarchie von vier Grammatiktypen zur Erzeugung von formalen Sprachen:

Typ 0-Sprache (rekursiv aufzählbar): keine Einschränkungen bezüglich der Produktion

Typ 1-Sprache (kontextsensitiv): auf der linken Seite vom Produktionspfeil sind Terminal- und Nichtterminalsymbole erlaubt
$\alpha_1 A \alpha_2 \rightarrow \alpha_1 \beta \alpha_2$, wobei $A \in N$, $\alpha_1, \alpha_2 \in (N \cup T)^*$, $\beta \in (N \cup T)^+$

Typ 2-Sprache (kontextfrei): auf der linken Seite vom Produktionspfeil ist nur ein Nichtterminalsymbol erlaubt
$A \rightarrow \beta$, wobei $A \in N$, $\beta \in (N \cup T)^+$

Typ 3-Sprache (regulär): analog zu Typ2, jedoch dürfen Worte entweder nur von links nach rechts (rechtsregulär) oder nur von rechts nach links (linksregulär) erzeugt werden
$A \rightarrow aB$ oder $A \rightarrow a$, wobei $A, B \in N$ und $a \in T$

5. Worin unterscheiden sich *kontextfreie* und *kontextsensitive* Grammatiken?

Der Unterschied zwischen diesen beiden Grammatiken besteht darin, dass bei einer kontextfreien Grammatik auf der linken Seite vom Produktionspfeil nur ein nichtterminales Symbol erlaubt ist, während bei einer kontextsensitiven Grammatik ein oder mehrere nichtterminale und terminale Symbole erlaubt sind.

Aufgabe 2: Grammatiken

Gegeben sei folgende Grammatik mit dem Startsymbol S:

N = {*S*, *A*, *B*}
T = {a, b, c}
P = {*S* → abc, *S* → a*A*bc, *A*b → b*A*, *A*c → *B*bcc, b*B* → *B*b, a*B* → aa*A*, a*B* → aa}

1. Welchen Typ hat die Grammatik bzgl. der Chomsky-Hierarchie?

Die Grammatik ist kontextsensitiv, da auf der linken Seite der Produktionsregeln sowohl Zeichen aus N und aus T vorkommen.

2. Zeigen Sie, dass die von dieser Grammatik erzeugte Sprache gegeben ist durch $L(G) = \{a^n b^n c^n \mid n>0\}$ = {abc, aabbcc, aaabbbccc, …}!

S → abc ✓
S → aAbc → abAc → abBbcc → aBbbcc → aabbcc ✓
→ aaAbbcc

analoger Aufbau, Rekursion

Aufgabe 3: Grammatiken

Gegeben sei der folgende Auszug aus einer Grammatik:

<Expression> → <number>
<Expression> → (<Expression>)
<Expression> → <Expression> + <Expression>
<Expression> → <Expression> - <Expression>
<Expression> → <Expression> * <Expression>
<Expression> → <Expression> / <Expression>
<number> → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

1. Wie lautet die Definition der entsprechenden Grammatik, welchen Typ hat sie?

G = (N, T, P, S)
N = {<Expression>, <number>}
T = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +. -, *, /, (,)}
S = <Expression>
P = siehe oben

Es handelt sich um eine Typ 2-Grammatik (kontextfrei), da auf der linken Seite der Produktions-reageln nur ein Nicht-Terminalsymbol vorhanden ist.

2. Prüfen Sie, ob folgende Worte Elemente der durch die Grammatik beschriebenen Sprache sind: (1+2)*3, (1-2)+(4-13), -1, (-1)*(+1), 3-1++2

(1+2)*3: S → <Expression>
→ <Expression> * <Expression>
→ (<Expression>) * <Expression>
→ (<Expression> + <Expression>) * <Expression>
→ (<number> + <number>) * <number>
→ (1+2)*3 ✓

(1-2)+(4-13) keine zweistelligen Zahlen ableitbar ↯
-1 keine negativen Zahlen ableitbar ↯
(-1)*(+1) keine negativen Zahlen ableitbar,
+-Operator ist in dieser Grammatik ein binärer Operator ↯
3-1++2 kein doppeltes-Pluszeichen ableitbar ↯

3. Definieren Sie Varianten dieser Grammatik:

3.1 Erweiterung auf reele Zahlen.

Expression = [Vorzeichen] number [',' { number} numberohne0]
Expression = [Vorzeichen] numberohne0 {number} [',' {number} numberohne0]
Vorzeichen = '+' | '-'
numberohne0 = ('1' | '2' | '3' | '4' | '5' | '6' | '7' | '8' | '9')

3.2 Erweiterung um Vorzeichen.

Expression = [Vorzeichen] Expression
Vorzeichen = '+' | '-'

Aufgabe 4: EBNF

1. **Was ist die Erweiterte Backus-Naur-Form?**

   Die EBNF ist eine kompakte, formale Metasprache zur Darstellung von Typ 2-Grammatiken.

2. **Notieren Sie die oben angegebene Grammatik in EBNF!**

   Expression = (Expression ('+' | '-' | '*' | '/') Expression) | '(' Expression ')' | ('0' | '1' | … | '8' | '9');

Aufgabe 5: Begriffe und Definitionen

1. **Was sind Mealy-, was sind Moore-Automaten? *(1-29, 1-32)***

   Ein Mealy-Automat ist ein endlicher Automat mit Ausgabe, während ein Moore-Automat ein endlicher Automat ohne Ausgabe ist.

2. **Geben Sie die Definition von Deterministischen Endlichen Automaten (DEA) an! *(1-29, 1-32)***

   Ein DEA ist gegeben durch ein 5-Tupel $\Omega = (E, Z, z_0, \delta, F)$, wobei gilt
   1. Das Eingabealphabet E ist eine endliche Menge von Eingabezeichen $e_0, e_1, \ldots, e_n$.
   2. Die Zustandsmenge Z ist eine endliche Menge von Zuständen $z_0, z_1, \ldots, z_p$.
   3. Ein Initialzustand $z_0 \in Z$ heißt Anfangszustand oder Startzustand.
   4. Die Zustandsübergangsfunktion ist gegeben durch $\delta = E \times Z \rightarrow Z$.
   5. Die Menge der Endzustände ist gegeben durch F, $F \subseteq Z$.

   Für einen DEA mit Ausgabe gibt es zusätzlich ein Ausgabealphabet A und die Endzustände werden durch eine Ausgabefunktion λ: $E \times Z \rightarrow A$ ersetzt.

3. **Was bedeutet es, wenn die Zustandsübergangsfunktion partiell ist? *(1-36f)***

   Es werden nicht für alle Kombinationen von Eingabezeichen und Zustände Folgezustände definiert. Der Automat bleibt beim Einlesen eines Zeichens ohne definierten Folgezustand stecken bzw. stirbt.

4. **Geben Sie die Definition von Nichtdeterministischen Endlichen Automaten (NEA) an! *(1-45)***

   Ein NEA ist gegeben durch ein 5-Tupel $\Omega = (E, Z, z_0, \delta, F)$, wobei gilt
   1. Das Eingabealphabet E ist eine endliche Menge von Eingabezeichen $e_0, e_1, \ldots, e_n$.
   2. Die Zustandsmenge Z ist eine endliche Menge von Zuständen $z_0, z_1, \ldots, z_p$.
   3. Ein Initialzustand $z_0 \in Z$ heißt Anfangszustand oder Startzustand.
   4. Die Zustandsübergangsrelation ist gegeben durch $\delta = E \times Z \times Z$.
   5. Die Menge der Endzustände ist gegeben durch F, $F \subseteq Z$.

5. **Wo befinden sich NEAs in der Chomsky-Hierarchie?**

   NEAs gehören zur Gruppe der Typ 3-Sprachen.

6. **Wann akzeptiert ein NEA ein Wort? *(1-46)***

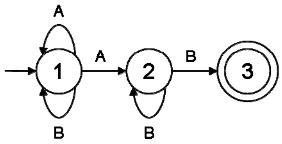
   Es existiert ein Weg mit einem Label, das dem Wort entspricht und in einem akzeptierten Endzustand endet.

7. **Worin unterscheiden sich NEAs und DEAs?**

   Sie unterscheiden sich nur durch Zustandsübergangsfunktion bzw. -relation.

Aufgabe 6: Spracherkennung

Gegeben sei der folgende NEA:



a) Geben Sie Beispiele für Worte, die von dem Automaten akzeptiert werden.

Alle Kombinationen aus A und B, die ein A beinhalten und auf B enden.

b) Geben Sie einen Pfad an, durch den der NEA folgende Worte akzeptiert: ABABABB, AAABBB, AB, ABABABBA

1 $\xrightarrow{A}$ 1 $\xrightarrow{B}$ 1 $\xrightarrow{A}$ 1 $\xrightarrow{B}$ 1 $\xrightarrow{A}$ 2 $\xrightarrow{B}$ 2 $\xrightarrow{B}$ 3 ✓

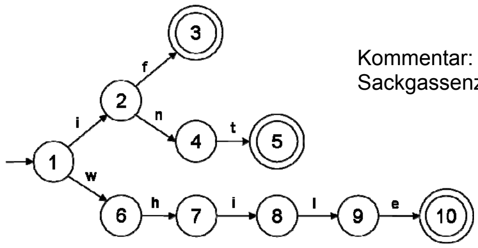
1 $\xrightarrow{A}$ 1 $\xrightarrow{A}$ 1 $\xrightarrow{A}$ 2 $\xrightarrow{B}$ 2 $\xrightarrow{B}$ 2 $\xrightarrow{B}$ 3 ✓

1 $\xrightarrow{A}$ 2 $\xrightarrow{B}$ 3 ✓

1 $\xrightarrow{A}$ 1 $\xrightarrow{B}$ 1 $\xrightarrow{A}$ 1 $\xrightarrow{B}$ 1 $\xrightarrow{A}$ 1 $\xrightarrow{B}$ 1 $\xrightarrow{B}$ 1 $\xrightarrow{A}$ 2

Aufgabe 7: Konstruktion eines DEA

Entwerfen Sie einen DEA Ω über dem Alphabet Λ = {a, b, c, …, z}, der die Sprache L(Ω) = {if, while, int} erkennt, d.h. genau die drei Schlüsselworte `if`, `while` und `int` akzeptiert.

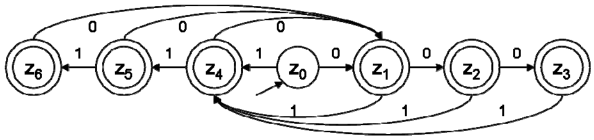


Kommentar: Es fehlen die Sackgassenzustände.

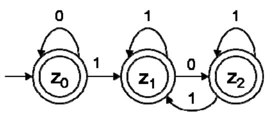
Aufgabe 8: Konstruktion von NEAs

Entwerfen Sie NEAs für die folgenden Sprachen:

a) Alle Folgen von Nullen und Einsen, wobei jede Ziffer höchstens dreimal hintereinander auftreten darf.



b) Alle Zeichenketten von Nullen und Einsen, die nicht die Folge 100 enthalten.



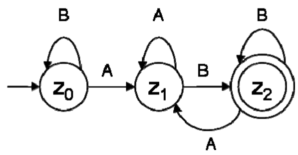
c) Geben Sie die Automatentafel des NEAs aus Aufgabe a) an.

Zustand	1	0
z_0	z_4	z_1
z_1	z_4	z_2
z_2	z_4	z_3
z_3	z_4	$\emptyset$
z_4	z_5	z_1
z_5	z_6	z_1
z_6	$\emptyset$	z_1

Aufgabe 9: Transformation von NEAs in DEAs

Überführen Sie den in Aufgabe 6 gegebenen NEA in einen DEA. Verwenden Sie die Technik der Teilmengenkonstruktion.

Neue Namen	Potenzmenge	A	B
z_0	S → {1}	{1,2}	{1}
z_1	{1,2}	{1,2}	{1,2,3}
z_2	F → {1,2,3}	{1,2}	{1,2,3}



Übungsblatt 3

Aufgabe 1: Definitionen

1. Was ist der Ertrag eines Ableitungsbaumes? ***(1-87)***

Der Wert der Blätter von links nach rechts gelesen.

2. Geben Sie die Definition von Regulären Ausdrücken an! ***(1-62)***

A sei ein Alphabet; R bezeichnet einen regulären Ausdruck, L(R) bezeichnet die zum dem regulären Ausdruck gehörige Sprache.

1. $\emptyset \notin A$ ist ein regulärer Ausdruck für die Sprache $L(\emptyset) = \emptyset$

2. $\varepsilon \notin A$ ist ein regulärer Ausdruck für die Sprache $L(\varepsilon) = \{\varepsilon\}$

3. Für jedes $a \in A$ ist a ein regulärer Ausdruck für die Sprache $L(a) = \{a\}$

4. Seien R und S reguläre Ausdrücke für die Sprachen L(R) und L(S). Dann sind auch die folgenden Ausdrücke reguläre Ausdrücke:

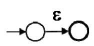
   - Vereinigung R|S
     mit der Sprache $L(R|S) = L(R) \cap L(S) = \{w \mid w \in L(R) \text{ oder } w \in L(S)\}$
   - Konkatenation RS mit der Sprache
     mit der Sprache $L(RS) = L(R)L(S) = \{wv \mid w \in L(R) \text{ und } v \in L(S)\}$
   - Abschluss R*
     mit der Sprache $L(R^*) = \{w_1 \dots w_n \mid w_i \in L(R), i = 1, 2, \dots, n\} \cup \varepsilon$

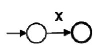
3. Wie lautet die Rangfolge der Operatoren für reguläre Ausdrücke? ***(1-67)***

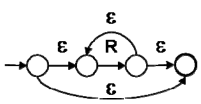
1. Abschluss R*
2. Konkatenation RS
3. Vereinigung R|S

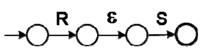
4. Geben Sie die Konstruktionsregeln der ε-Transformation an! ***(1-74)***

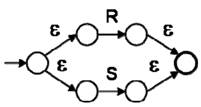
∅: 

ε: 

a: 

R*: 

RS: 

R|S: 

Aufgabe 2: Reguläre Ausdrücke

Beschreiben Sie die Sprachen, die durch folgende Ausdrücke gegeben sind:

a) (0 | 1)* a | 1b

Beliebige Kombinationen von 0 und 1 mit a am Ende oder das Wort 1b.

b) 0 ((ε | 0) 1*)* 0

Jedes Wort beginnt und endet mit einer 0. Wahlmöglichkeiten:
- beliebige Wiederholung von 0

- beliebige Wiederholung von 1
- beliebige Wiederholung von 0 gefolgt von beliebigen Einsen

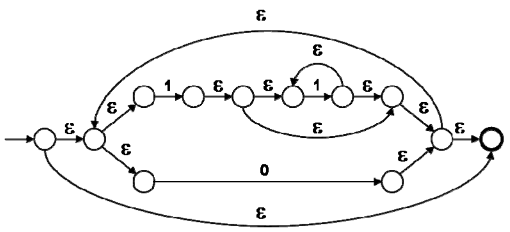
c) 0 1 0* 1 0* 1 0* (1 0)*

Jedes Wort beginnt mit einer Null, gefolgt von drei Einsen, hinter denen jeweils beliebig viele Nullen stehen können. Am Ende stehen beliebige Wiederholungen der Zeichenkette 10.

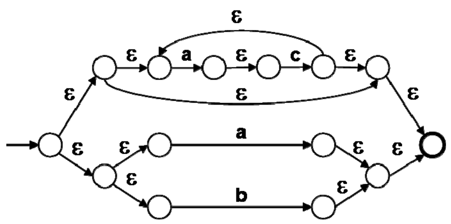
Aufgabe 3: Konvertieren regulärer Ausdrücke in Automaten

Konstruieren Sie Automaten mit ε-Transformationen für die folgenden regulären Ausdrücke:

a) (0 | 1 1*)*



b) (a c)* | a | b



Aufgabe 4: Konvertieren regulärer Ausdrücke

Gegeben sei der folgende reguläre Ausdruck: (a b)* a (a | b)*

1. Welche Sprache beschreibt dieser reg. Ausdruck?

Beliebige Kombinationen der Wortkette "ab", gefolgt von einem "a", und anschließend beliebige Kombinationen der Zeichen "a" und "b". Der reguläre Ausdruck a (a | b)* beschreibt die gleiche Sprache.

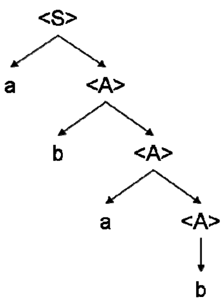
2. Geben Sie eine reguläre Grammatik für diese Sprache an!

T = {a, b}
N = {S, A}
P = {S → a,
S → aA,
A → a,
A → b,
A → aA,
A → bA}

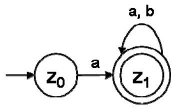
3. Geben Sie den Ableitungsbaum für die Worte a, abab an!

a:

abab:



4. Überführen Sie den Ausdruck in einen NEA!



5. Überführen Sie den Ausdruck in einen DEA!

Da der angegebene Ausdruck bereits ein DEA ist, muss keine Umformung erfolgen. Ansonsten würde das Prinzip der Teilmengenkonstruktion angewandt werden.

Aufgabe 5:

Prüfen Sie, ob die gegebenen Regulären Audrücke die gleichen Sprachen erzeugen

a) ((ε | 0) 1*)* und (0 1)*

Nein, links ist z.B. das Wort 1 möglich.

b) (0* | 1 | 1*) und (0* 1*)

Nein, rechts ist z.B. das Wort 01 möglich.

c) (0 | 1 | 1*)* und (0* 1*)*

Ja, es sind beliebige Kombinationen von 0 und 1 möglich.

d) (0* | 1*) und (1* | 0*)

Ja, Vereinigung ist kommutativ.

e) ((ε | 0)* | 0 | 1*)* und (ε | 0 | 1*)*

Ja, es sind beliebige Kombinationen von 0 und 1 möglich.

Übungsblatt 4

Aufgabe 1: Definitionen, etc.

1. **Beschreiben Sie alle logischen Funktionen mit 2 Variablen durch entsprechende Wahrheitstabellen. Benennen Sie die Ihnen bekannten Funktionen!**

x_1	0	0	1	1		
x_2	0	1	0	1		
f_0	0	0	0	0	0	Kontradiktion, Nullfunktion
f_1	0	0	0	1	$x_1 \wedge x_2$	Konjunktion
f_2	0	0	1	0	$x_1 \wedge \neg x_2$	Inhibition x_1
f_3	0	0	1	1	x_1	Identität (von x_1)
f_4	0	1	0	0	$\neg x_1 \wedge x_2$	Inhibition x_2
f_5	0	1	0	1	x_2	Identität (von x_2)
f_6	0	1	1	0	$(x_1 \equiv x_2)$ $(x_1 \wedge \neg x_2) \vee (\neg x_1 \wedge x_2)$	Antivalenz, XOR(x_1,x_2), Alternative
f_7	0	1	1	1	$x_1 \vee x_2$	Disjunktion
f_8	1	0	0	0	$\neg(x_1 \vee x_2) = \neg x_1 \wedge \neg x_2$	NOR(x_1,x_2), Pierce-Funktion
f_9	1	0	0	1	$x_1 \equiv x_2$ $y=(x_1 \wedge x_2) \vee (\neg x_1 \wedge \neg x_2)$	Äquivalenz
f_{10}	1	0	1	0	$\neg x_2$	Negation (von x_2)
f_{11}	1	0	1	1	$x_1 \vee (\neg x_2)$	Replikation
f_{12}	1	1	0	0	$\neg x_1$	Negation (von x_1)
f_{13}	1	1	0	1	$x_1 \rightarrow x_2$ $\neg x_1 \vee x_2$	Implikation
f_{14}	1	1	1	0	$\neg(x_1 \wedge x_2)$ $\neg x_1 \vee \neg x_2$	NAND(x_1,x_2), Sheffer-Funktion
f_{15}	1	1	1	1	1	Tautologie, Einsfunktion

2. **Wie ist die Rangfolge der logischen Operatoren (NICHT, ODER, NODER, UND, NUND, IDENTITÄT, IMPLIKATION) bezüglich ihrer Bindungsstärke? *(2-17)***

   NICHT, NUND, NODER, UND, ODER, IMPLIKATION, IDENTITÄT

3. **Was ist eine Tautologie? *(2-20)***

   Tautologien sind logische Ausdrücke, die unabhängig von der Belegung der Variablen immer den Wert TRUE ergeben.

4. **Was bedeutet Widerspruchsfreiheit? *(2-29)***

   Eine Menge von Aussagen heißt widerspruchsfrei (oder konsistent), wenn es eine Belegung der Variablen gibt, so dass alle Aussagen wahr sind.

5. **Wann sind zwei boolsche Ausdrücke werteverlaufsgleich?**

   Seien A und B zwei aussagenlogische Ausdrücke, so sind diese werteverlaufsgleich, wenn gilt:
   $A \leftrightarrow B$ ist eine Tautologie.

6. **Nennen Sie die Definition der CNF und DNF! *(2-49f)***

   CNF (Konjunktive Normalform)
   Ein aussagenlogischer Ausdruck ist in CNF, wenn er folgende Form hat:
   $A_1 \wedge A_2 \wedge \ldots \wedge A_n$,
   wobei jedes A_i eine Disjunktion von Literalen ist. D.h. ein Ausdruck in CNF besteht aus einer Konjunktion von Disjunktionen von Literalen.

DNF (Disjunktiive Normalform):
Ein aussagenlogischer Ausdruck ist in DNF, wenn er folgende Form hat:
$A_1 \vee A_2 \vee ... \vee A_n$,
wobei jedes A_i eine Konjunktion von Literalen ist. D.h. ein Ausdruck in DNF besteht aus einer Disjunktion von Konjunktionen von Literalen.

Aufgabe 2: Grundlagen

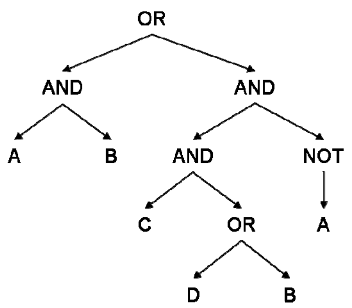
1. Erstellen Sie die Wahrheitstabelle für P ≡ Q AND P → Q OR Q → P.

P ≡ (((Q AND P) → (Q OR Q)) → P)

P	Q	① Q ∧ P	② Q ∨ Q	③ ① → ②	④ ③ → P	④ ≡ P
0	0	0	0	1	0	1
0	1	0	1	1	0	1
1	0	0	0	1	1	1
1	1	1	1	1	1	1

2. Geben Sie den Syntaxbaum an für: (A AND B OR C AND (D OR B) AND NOT A)

((A AND B) OR ((C AND (D OR B)) AND (NOT A)))



Aufgabe 3: Tautologien

Prüfen Sie, ob die folgenden booleschen Ausdrücke tautologisch sind.

1. (P → Q) ≡ (NOT P → Q)

P	Q	¬P	① P → Q	② ¬P → Q	① ≡ ②
0	0	1	1	0	0
0	1	1	1	1	1
1	0	0	0	1	0
1	1	0	1	1	1

Der Ausdruck ist nicht tautologisch.

2. NOT (P AND Q) ≡ (NOT P OR NOT Q)

P	Q	¬P	¬Q	(P ∧ Q)	① ¬ (P ∧ Q)	② ¬P ∨ ¬Q	① ≡ ②
0	0	1	1	0	1	1	1
0	1	1	0	0	1	1	1
1	0	0	1	0	1	1	1
1	1	0	0	1	0	0	1

Der Ausdruck ist tautologisch (de Morgansches Gesetz).

Aufgabe 4: Widerspruchsfreiheit

Gegeben seien folgende Fakten:

S: Die Nichtbestehensquote bei Prüfungen sinkt
R: Die Studenten lernen viel
H: Die Professoren sind unglücklich

Prüfen Sie, ob folgende Aussagen a), b), c), d) untereinander widerspruchsfrei sind.

a) R → S, b) H → NOT S, c) R AND H, d) H

R	S	H	¬S	① R → S	② H → ¬S	③ R ∧ H	④ ① ∧ ②	⑤ ④ ∧ ③	⑤ ∧ H
0	0	0	1	1	1	0	1	0	0
0	0	1	1	1	1	0	1	0	0
0	1	0	0	1	1	0	1	0	0
0	1	1	0	1	0	0	0	0	0
1	0	0	1	0	1	0	0	0	0
1	0	1	1	0	1	1	0	0	0
1	1	0	0	1	1	0	1	0	0
1	1	1	0	1	0	1	0	0	0

Die Aussagen sind untereinander nicht widerspruchsfrei.

Aufgabe 5: Basisoperator

Weisen Sie nach, dass der OR Operator unter ausschließlicher Verwendung des NAND Operators nachgebildet werden kann. Nutzen Sie dazu die Äquivalenzregeln des Skriptes.

$(A \vee B) \equiv \neg\neg(A \vee B)$ *nach Regel 9: Doppelte Negation*
$\equiv \neg(\neg A \wedge \neg B)$ *nach Regel 19: de Morgansches Gesetz*
$\equiv \neg(\neg(A \wedge A) \wedge \neg(B \wedge B))$ *nach Regel 5: Idempotenz der Konjunktion*

(A NAND A) NAND (B NAND B)

Aufgabe 6: Normalformen

a) (a OR b) AND NOT a OR (a → NOT b) AND a
b) (a OR b) AND (NOT a OR NOT b)
c) NOT (NOT (NOT a AND b) AND (NOT (a AND (NOT a OR b) OR a) OR b))
d) (NOT a AND b OR a AND NOT b) OR (a ↔ b) AND a AND NOT b

1. Zeigen Sie mit Hilfe der Äquivalenzregeln im Skript die Werteverlaufsgleichheit der Ausdrücke a) bis d)

a) $((a \vee b) \wedge \neg a) \vee ((a \rightarrow \neg b) \wedge a)$
$(\neg a \wedge (a \vee b)) \vee (a \wedge (a \rightarrow \neg b))$ *nach Regel 10, Kommutativität der Konjunktion*
$(\neg a \wedge (a \vee b)) \vee (a \wedge (\neg a \vee \neg b))$ *nach Regel 22, Umwandlung von → in ∨ und ¬*
$(\neg a \wedge b) \vee (a \wedge \neg b)$ *nach Regel 17, Absorbtionsgesetz*

b) $(a \vee b) \wedge (\neg a \vee \neg b)$
$(a \wedge (\neg a \vee \neg b)) \vee (b \wedge (\neg a \vee \neg b))$ *nach Regel 12, Distributivität der Konjunktion*
$(a \wedge \neg a) \vee (a \wedge \neg b) \vee (\neg a \wedge b) \vee (b \wedge \neg b)$ *nach Regel 12, Distributivität der Konjunktion*
$(a \wedge \neg b) \vee (\neg a \wedge b)$ *nach Regel 7, Kontradiktion*
$(\neg a \wedge b) \vee (a \wedge \neg b)$ *nach Regel 10, Kommutativität der Konjunktion*

c) $\neg(\neg(\neg a \wedge b) \wedge (\neg((a \wedge (\neg a \vee b)) \vee a) \vee b))$
$\neg(\neg(\neg a \wedge b) \wedge (\neg((a \wedge b) \vee a) \vee b))$ *nach Regel 17, Absorbtionsgesetz*
$\neg(\neg(\neg a \wedge b) \wedge (\neg(a \vee (a \wedge b)) \vee b))$ *nach Regel 11, Kommutativität der Disjunktion*
$\neg(\neg(\neg a \wedge b) \wedge (\neg a \vee b))$ *nach Regel 15, Absorptionsgesetz*
$\neg((\neg\neg a \vee \neg b) \wedge (\neg a \vee b))$ *nach Regel 18, de Morgansches Gesetz*
$\neg((a \vee \neg b) \wedge (\neg a \vee b))$ *nach Regel 9, Doppelte Negation*
$\neg(a \vee \neg b) \vee \neg(\neg a \vee b)$ *nach Regel 18, de Morgansches Gesetz*
$(\neg a \wedge \neg\neg b) \vee (\neg\neg a \wedge \neg b)$ *nach Regel 19, de Morgansches Gesetz*
$(\neg a \wedge b) \vee (a \wedge \neg b)$ *nach Regel 9, Doppelte Negation*

d) $((\neg a \wedge b) \vee (a \wedge \neg b)) \vee ((a \leftrightarrow b) \wedge a \wedge \neg b)$

es wird erstmal nur der rechte Teil untersucht:

$(a \leftrightarrow b) \wedge a \wedge \neg b$	
$(a \rightarrow b) \wedge (b \rightarrow a) \wedge a \wedge \neg b$	*nach Regel 24*
$((\neg a \vee b) \wedge (\neg b \vee a)) \wedge a \wedge \neg b$	*nach Regel 22*
$((a \wedge (\neg a \vee b)) \vee (\neg b \wedge (\neg a \vee b))) \wedge a \wedge \neg b$	*nach Regel 12*
$((a \wedge \neg a) \vee (a \wedge b) \vee (\neg a \wedge \neg b) \vee (b \wedge \neg b)) \wedge a \wedge \neg b$	*nach Regel 12*
$((a \wedge b) \vee (\neg a \wedge \neg b)) \wedge a \wedge \neg b$	*nach Regel 7*
$((a \wedge \neg b) \wedge (a \wedge b)) \vee ((a \wedge \neg b) \wedge (\neg a \wedge \neg b))$	*nach Regel 15*
$(a \wedge a \wedge \neg b \wedge b) \vee (a \wedge \neg a \wedge \neg b \wedge \neg b)$	*nach Regel 10*
$(a \wedge a \wedge 0) \vee (0 \wedge \neg b \wedge \neg b)$	*nach Regel 7*
$0 \vee 0$	*nach Regel 1*
0	*nach Regel 6*

somit gilt nach Regel 3 lediglich der linke Term des Ausgangsaudrucks:
$(\neg a \wedge b) \vee (a \wedge \neg b)$

2. Erzeugen Sie die DNF und CNF der vereinfachten Ausdrücke!

DNF:

$(\neg a \wedge b) \vee (a \wedge \neg b)$	*Ergebnis der Umformungen*

CNF *(siehe auch Aufgabenstellung b))*:

$(\neg a \wedge b) \vee (a \wedge \neg b)$	
$(a \vee (\neg a \wedge b)) \wedge (\neg b \vee (\neg a \wedge b))$	*nach Regel 17*
$(a \vee \neg a) \wedge (a \vee b) \wedge (\neg b \vee \neg a) \wedge (\neg b \vee b)$	*nach Regel 13*
$1 \wedge (a \vee b) \wedge (\neg b \vee \neg a) \wedge 1$	*nach Regel 8*
$(a \vee b) \wedge (\neg b \vee \neg a)$	*nach Regel 2*

Übungsblatt 5

Aufgabe 1: Begriffe und Definitionen

1. Was ist das Prinzip der Resolution, der Refutation und Resolutionsrefutation? ***(2-55ff)***

Resolution: Systematische automatisierbare Methode zur Ableitung der Widerspruchsfreiheit von aussagenlogischen Ausdrücken
Refutation: der Widerspruchsbeweis
Resolutionsrefutation: Ableitung von Falsum durch Resolution aus einer Klauselmenge

2. Welche Regeln gelten für die Resolvierung von Klauseln? ***(2-58)***

res (C1, C2) = (C1 \ {L}) ∪ (C2 \ {¬L})
res ({L}, {¬L}) = ⊥

3. **Erläutern Sie die Begriffe Prädikat und Quantor!** ***(2-68f)***

Prädikat: zur Beschreibung von Objekt-Eigenschaften und von Beziehungen zwischen Objekten
Quantor: ∀ All-Quantor, "für alle"
∃ Existenz-Quantor, "es existiert"

4. Wann ist ein Prädikat, eine Formel erfüllbar bzw. gültig? ***(2-72)***

Erfüllbarkeit von Prädikaten:
Ein n-stelliges Prädikat P heißt erfüllbar, wenn es mindestens $x_1, \ldots, x_n \in M$ gibt, so dass $P(x_1, \ldots, x_n) = TRUE$

Gültigkeit von Prädikaten:
Ein n-stelliges Prädikat P heißt gültig, wenn für alle $x_1, \ldots, x_n \in M$ gilt, so dass $P(x_1, \ldots, x_n) = TRUE$

Erfüllbarkeit von Formeln:
Eine Formel heißt erfüllbar, wenn sie mindenstens eine Interpretation besitzt, welche die Formel zu einem erfüllbaren Prädikat macht.

Gültigkeit von Formeln:
Eine Formel heißt gültig, wenn jede Interpretaion die Formel zu einem erfüllbaren Prädikat macht.

Aufgabe 2: Resolventen

Bestimmen Sie die Resolventen folgender Klauselpaare:

a) {p1}, {¬p1}

res({p1}, {¬p1}) = ⊥

b) {p1, ¬p2, p3}, {p1, ¬p3}

res({p1, ¬p2, p3}, {p1, ¬p3}) = {p1, ¬p2}

c) {p1, ¬p2, ¬p3, p4}, {p2, ¬p3}

res({p1, ¬p2, ¬p3, p4}, {p2, ¬p3}) = {p1, ¬p3, p4}

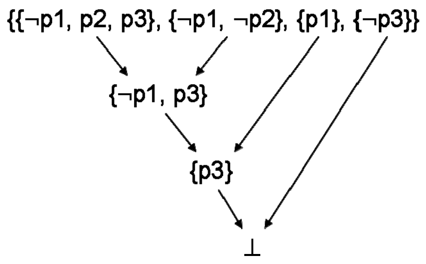
d) {p1, ¬p2, p3, p4}, {p1, p2, ¬p3}

res({p1, ¬p2, p3, p4}, {p1, p2, ¬p3}) = {p1, p3, ¬p3, p4} oder {p1, p2, ¬p2, p4}

Aufgabe 3: Resolution

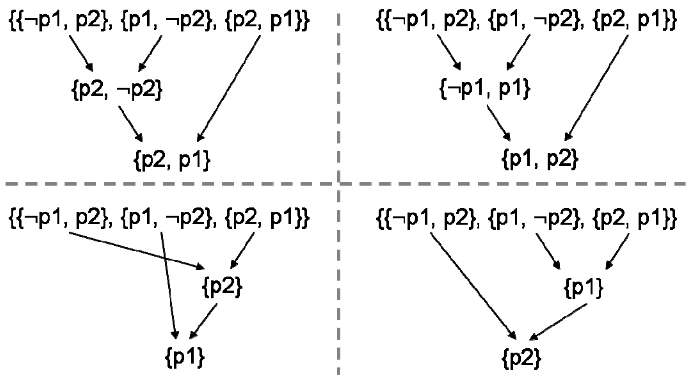
Welche der folgenden Klauselmengen sind nicht widerspruchsfrei:

a) {{¬p1, p2, p3}, {¬p1, ¬p2}, {p1}, {¬p3}}



Die Klauselmengen sind nicht widerspruchsfrei.

b) {{¬p1, p2}, {p1, ¬p2}, {p2, p1}}

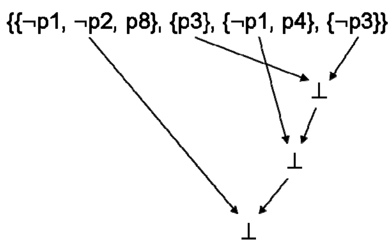


Die Klauselmengen sind widerspruchsfrei.

c) {{p1, ¬p2, ¬p3}, {¬p1, ¬p3}, {¬p1, p2, ¬p3}}

Das Literal ¬p3 steht in jeder Klausel und lässt sich somit nicht eliminieren → Die Klauselmengen sind widerspruchsfrei.

d) {{¬p1, ¬p2, p8}, {p3}, {¬p1, p4}, {¬p3}}



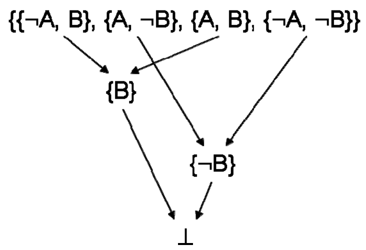
Die Klauselmengen sind nicht widerspruchsfrei.

Aufgabe 4: Resolutionsrefutation

Sind folgende Aussagen wahr? Prüfen Sie mit Hilfe des Resolutionsprinzips!

1. **Wenn $A \rightarrow B$ gilt und $\neg(\neg A \wedge B)$ gilt, dann gilt auch $A \leftrightarrow B$**

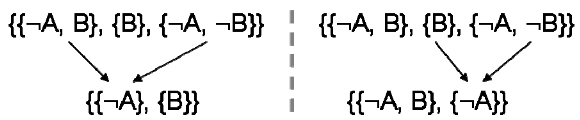
   $(A \rightarrow B) \wedge \neg(\neg A \wedge B) \wedge \neg(A \leftrightarrow B)$
   $(\neg A \vee B) \wedge (A \vee \neg B) \wedge \neg((\neg A \vee B) \wedge (A \vee \neg B))$
   $(\neg A \vee B) \wedge (A \vee \neg B) \wedge (\neg(\neg A \vee B) \vee \neg(A \vee \neg B))$
   $(\neg A \vee B) \wedge (A \vee \neg B) \wedge ((A \wedge \neg B) \vee (\neg A \wedge B))$
   $(\neg A \vee B) \wedge (A \vee \neg B) \wedge (A \vee B) \wedge (\neg A \vee \neg B)$



   Nicht widerspruchsfrei --> zu beweisende Aussage ist wahr!

2. **Wenn $A \rightarrow B$ wahr ist und B wahr ist, dann ist auch $A \wedge B$ wahr**

   $(A \rightarrow B) \wedge (B) \wedge \neg(A \wedge B)$
   $(\neg A \vee B) \wedge (B) \wedge (\neg A \vee \neg B)$



   Nach Prüfen aller Wege widerspruchsfrei --> zu beweisende Aussage ist falsch!

Aufgabe 5: Prädikate und Quantoren

Übersetzen Sie in prädikatenlogische Ausdrücke

a) **Jede gerade Zahl ist durch 2 teilbar**

   $(\forall x)\ (\text{Gerade}(x) \rightarrow \text{Teilbar}(x,2))\ x \in \mathbb{N}$

b) **Es gibt keine Primzahl zwischen 1 und 100**

   $\neg(\exists x)\ (\text{Primzahl}(x) \wedge \text{Zwischen}(x,1,100))\ x \in \mathbb{N}$

c) **Ist die natürliche Zahl x größer als die natürliche Zahl y, dann ist auch der Nachfolger von x größer als der Nachfolger von y**

   $(\forall x,y,z,a)\ ((\text{Größer}(x,y) \wedge \text{Nachfolger}(x,z) \wedge \text{Nachfolger}(y,a)) \rightarrow \text{Größer}(z,a))$

d) **Jeder liebt Paris (die Stadt)**

   $(\forall x)$ liebt (x,P) $x \in$ Person, P = Stadt Paris

e) **Nicht jeder ist reich**

   $\neg(\forall x)\ (\text{Person}(x) \rightarrow \text{Reich}(x))$

Übungsblatt 6

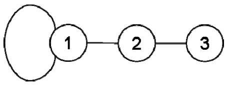
Aufgabe 1: Definitionen

1. Geben Sie für folgende Konstrukte die entsprechenden Definitionen an:

a) ungerichteter Graph, gerichteter Graph, kantenmarkierter und ungerichteter Graph, knotenmarkierter und gerichteter Graph *(3-9ff)*

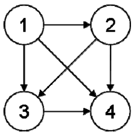
- ungerichteter Graph:

G = (V,E) heißt ungerichteter Graph, falls gilt:
(i) $V \neq \emptyset$ ist endliche Menge von Knoten (V = Vertices)
(ii) E ist eine Menge ein- und zweielementiger Teilmengen von V.
E heißt Kantenmenge, ein Paar $\{u,v\} \in E$ heißt Kante. Eine Kante {u} heißt Schlinge (E = Edges).



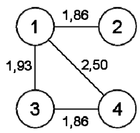
- gerichteter Graph:

G = (V,E) heißt gerichteter Graph, falls gilt:
(i) $V \neq \emptyset$ ist endliche Menge von Knoten (V = Vertices)
(ii) $E \subseteq V \times V$ heißt Kantenmenge. Elemente von E heißen Kanten (E: Edges). Schreibweise: (u,v) oder $u \rightarrow v$.



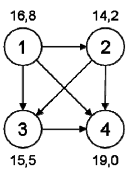
- kantenmarkierter und ungerichteter Graph:

G = (V,E,g)
g: $E \rightarrow M_E$ (Menge natürlicher/reeler Zahlen)



- knotenmarkierter und gerichteter Graph:

G = (V,E,f)
f: $V \rightarrow M_V$ (Menge natürlicher/reeler Zahlen)



b) Kantenfolge, Kantenzug, Weg, Zyklus *(3-11f)*

Kantenfolge: Eine Verbindung zwischen zwei Knoten, wobei alle Kanten $\in$ E sein müssen.

Kantenzug: Eine Kantenfolge, in der keine Kante doppelt vorkommt.

Weg: Ein Kantenzug, in dem kein Knoten doppelt vorkommt.

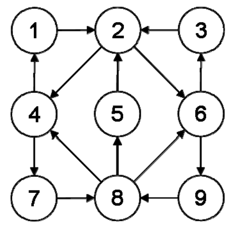
Zyklus: Eine geschlossene Kantenfolge, bei dem ein Weg entsteht, wenn der Endknoten entfernt wird.

2. Nennen Sie die Eigenschaften des Euler-Zyklus und des Hamilton-Zyklus. ***(3-29, 3-34)***

Euler-Zyklus: Ein Eulerscher Zyklus in einem Graphen G ist ein geschlossener Kantenzug, in dem jede Kante von G genau einmal durchlaufen wird.

Hamilton-Zyklus: Ein Hamiltonscher Zyklus in einem Graphen G ist ein Zyklus, der jeden Knoten von G enthält. D.h. es gibt einen Zyklus der Länge |V|, der alle Knoten genau einmal besucht. Graphen mit dieser Eigenschaft heißen Hamiltonsche Graphen.

Aufgabe 2: Kantenfolge, Kantenzug, Weg, Zyklus



Überprüfen Sie für jede der angegebenen Sequenzen von Knoten k_i (für den gegebenen Graphen) die folgenden Aussagen (1 bis 4):

1. **Die Sequenz ist ein Kantenzug (evtl. geschlossen?)**
2. **Die Sequenz ist eine Kantenfolge**
3. **Die Sequenz ist ein Weg**
4. **Die Sequenz ist ein Zyklus**

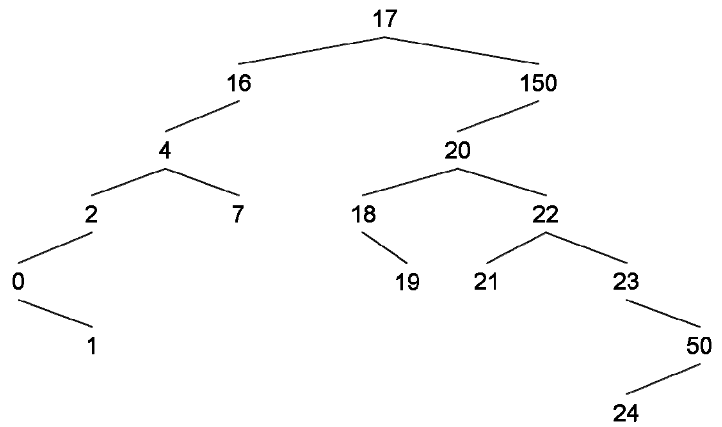
k_1 = (1,2,4,7), k_2 = (1,2,6,8,9), k_3 = (3,2,4,1,2,6,3,2), k_4 = (6,9,8,4,1,2,6,9), k_5 = (2,6,9,8,6,3,2), k_6 = (4,1,2,6,3,2,4,1,2,4), k_7 = (3,2,4,1,2), k_8 = (4,1,2,6,9,8)

	Kantenzug	Kantenfolge	Weg	Zyklus
k_1	ja	ja	ja	nein
k_2	nein	nein	nein	nein
k_3	nein	ja	nein	nein
k_4	ja (geschl.)	ja	nein	ja
k_5	ja (geschl.)	ja	nein	nein
k_6	nein	ja	nein	nein
k_7	ja	ja	nein	nein
k_8	ja	ja	ja	nein

Aufgabe 3: Binäre Suchbäume

Bauen Sie für die folgenden Schlüssel einen binären Suchbaum auf und stellen Sie diesen graphisch dar. Die Schlüssel sind genau in der vorgegebenen Reihenfolge einzufügen:

Schlüsselfolge: 17, 16, 150, 20, 22, 21, 4, 23, 18, 2, 0, 19, 1, 50, 7, 24



a) Durchlaufen Sie den Baum nach den drei Strategien Preorder, Inorder und Postorder. ***(3-26)***

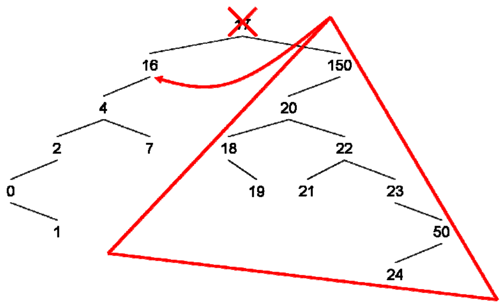
Preorder (Wurzel, linker Teilbaum, rechter Teilbaum):
17, 16, 4, 2, 0, 1, 7, 150, 20, 18, 19, 22, 21, 23, 50, 24

Inorder (linker Teilbaum, Wurzel, rechter Teilbaum):
0, 1, 2, 4, 7, 16, 17, 18, 19, 20, 21, 22, 23, 24, 50, 150

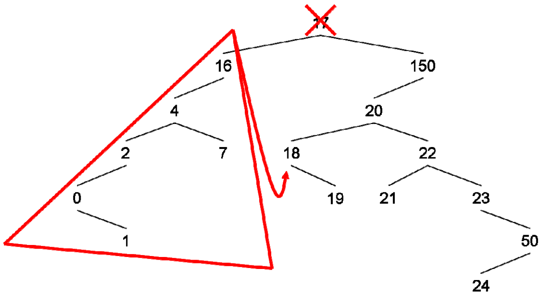
Postorder (linker Teilbaum, rechter Teilbaum, Wurzel):
1, 0, 2, 7, 4, 16, 19, 18, 21, 24, 50, 23, 22, 20, 150, 17

b) Es soll die Wurzel (Knoten 17) gelöscht werden, ohne dass der Baum völlig neu aufgebaut wird. Verwenden Sie die beiden hierfür aus der Vorlesung bekannten Verfahren, und erläutern Sie diese jeweils kurz.

Methode 1:
Hänge rechten Teilbaum von *n* ein als rechten Teilbaum des größten Knotens des linken Sohns von *n*.

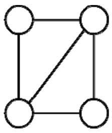


Methode 2:
Hänge linken Teilbaum von *n* ein als linken Teilbaum des kleinsten Knotens des rechten Sohns von *n*.

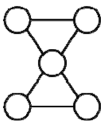


Aufgabe 4: Zyklus

1. **Geben Sie einen Graphen an, der einen Hamilton-Zyklus, aber keinen Euler-Zyklus enthält.**



2. **Geben Sie einen Graphen an, der einen Euler-Zyklus, aber keinen Hamilton-Zyklus enthält.**

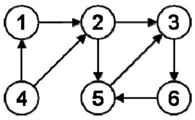


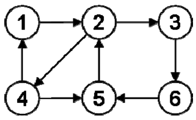
3. **Existiert ein Euler-Zyklus für den äquivalenten ungerichteten Graph zum Graph aus Aufgabe 2? Begründen Sie die Antwort.**

   Nein, dies ist nicht möglich, da nicht alle Knoten einen graden Grad besitzen.

Übungsblatt 7

Aufgabe 1: Zusammenhang

G1


G2


a) Grenzen Sie Zusammenhangskomponente und Äquivalenzklasse voneinander ab! (3-54f, 3-58)

Zusammenhangskomponente
Ein zusammenhängender Teilgraph Z heißt Zusammenhangskomponente von G, wenn er maximal ist, d.h. es existiert kein Knoten $v \in G \wedge v \notin Z$, so dass der um v erweiterte Graph Z zusammenhängend wäre.

Äquivalenzklasse
Sei $G = (V_G, E_G)$ ein gerichteter Graph, $u,v \in V_G$. $u \sim v$ (wegeäquivalent): $\leftrightarrow$ es gibt eine Kantenfolge von u nach v und eine von v nach u. K[u] bezeichnet die Äquivalenklasse von u, d.h. $K[u] = \{v \in V \mid u \sim v\}$.

Während das Ergebnis einer Zusammenhangskomponente also ein Teilgraph darstellt, bestehen Äquivalenzklassen aus Knotenmengen.

b) Was ist die Kanten- bzw. Knotenzusammenhangszahl? (3-60f)

Kantenzusammenhangszahl: Wieviele Kanten muss man minimal entfernen, damit der Graph in mindestens 2 Teilgraphen zerfällt.
Knotenzusammenhangszahl: Wieviele Knoten muss man minimal entfernen, damit der Graph in mindestens 2 Teilgraphen zerfällt.

c) Erstellen Sie die Adjazenzmatrix für den Graphen G1!

	1	**2**	**3**	**4**	**5**	**6**
1	0	1	0	0	0	0
2	0	0	1	0	1	0
3	0	0	0	0	0	1
4	1	1	0	0	0	0
5	0	0	1	0	0	0
6	0	0	0	0	1	0

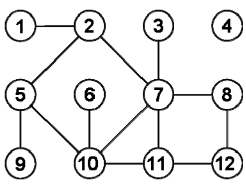
d) Bestimmen Sie alle Äquivalenzklassen von G1!

$K_1 = \{3,6,5\}$, $K_2 = \{2\}$, $K_3 = \{1\}$, $K_4 = \{4\}$

e) Bestimmen Sie Kanten- bzw. Knotenzusammenhangszahl von Graph G2!

Kantenzusammenhangszahl: 2 (z.B. (2,3) und (3,6) entfernen, Knoten 3 isoliert)
Knotenzusammenhangszahl: 2 (z.B. Knoten 2 und 5 entfernen, es entstehen zwei Teilgraphen)

Aufgabe 2: Traversierung ungerichteter Graphen



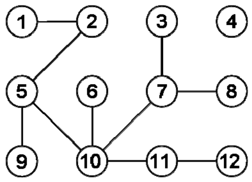
a) Geben Sie die Definitionen des graphisch repräsentierten Graphen an.

G = (V,E)
V = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}
E = {(1,2), (2,5), (2,7), (3,7), (5,9), (5,10), (6,10), (7,8), (7,10), (7,11), (8,12), (10,11), (11,12)}

b) Führen Sie bei dem Graphen Tiefen- und Breitendurchläufe durch, jeweils beginnend mit Knoten 1 und 11. Falls an einem Knoten alternative Fortsetzungen möglich sind, wählen Sie die Kante, die jeweils zum Knoten mit der größerene Knotenmarkierung führt. (3-45, 3-51)

DFS (Start 1): 1, 2, 7, 11, 12, 8, 10, 6, 5, 9, 3
DFS (Start 11): 11, 12, 8, 7, 10, 6, 5, 9, 2, 1, 3
BFS (Start 1): 1, 2, 7, 5, 11, 10, 8, 3, 9, 12, 6
BFS (Start 11): 11, 12, 10, 7, 8, 6, 5, 3, 2, 9, 1

c) Geben Sie einen Spannbaum des Graphen an.



Aufgabe 3: Kürzeste Wege

a) Was ist die Länge eines Wegs (formale Definition)? (3-70)

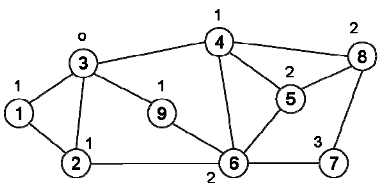
Summe der Kantenmarkierungen entlang eines Weges.

b) Welche 3 typischen Fragestellungen sind Ihnen im Zusammenhang mit kürzesten Wegen innerhalb von Graphen bekannt? (3-71)

1) Gesucht wird der kürzeste Weg zwischen zwei gegebenen Knoten.
2) Gesucht werden die kürzesten Wege von einem gegebenen Knoten zu allen anderen Knoten.
3) Alle Kanten haben die Bewertung 1, gesucht wird der Weg mit den wenigsten zu besuchenden Knoten.

Aufgabe 4: Minimale Wege in unbewerteten Graphen

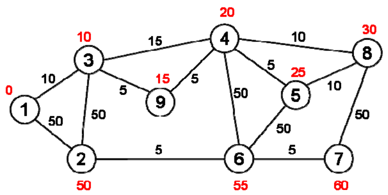
Der gegebene unbewertete und ungerichtete Graph G = (V,E) stellt ein Rechnernetz dar. Die Übertragungsraten und Entfernungen zwischen den einzelnen Knoten werden als identisch angenommen, wodurch die Gewichtung der Kanten entfällt.
Ein Paket soll von Rechner 3 zu Rechner 7 übertragen werden. Welche Routing-Strecke sollte gewählt werden, damit das Paket schnellstmöglich ans Ziel gelangt?



Kürzeste Wege: 3 - 4 - 8 - 7
3 - 4 - 6 - 7
3 - 2 - 6 - 7
3 - 9 - 6 - 7

Aufgabe 5: Minimale Wege in kantenbewerteten Graphen

Der gegebene kantenbewertete ungerichtete Graph G = (V,E,γ) stellt ein Rechnernetz dar, die Kantengewichtungsfunktion γ: E → R+ gibt die Zeit in ms (= Millisekunden) an, in der ein Datenpaket konstanter Größe über die Strecke E übertragen wird.
Ein Paket soll von Rechner 1 zu Rechner 7 übertragen werden. Welche Routing-Strecke sollte gewählt werden, damit das Datenpaket schnellstmöglich ans Ziel gelangt?

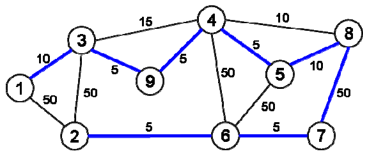


Kürzester Weg: 1 - 2 - 6 - 7

Aufgabe 6: Minimaler Spannbaum

Ermitteln Sie den minimalen Spannbaum für den Graphen aus Aufgabe 5. Verwenden Sie die Algorithmen nach

a) Kruskal

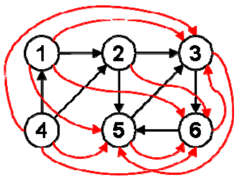


b) Prim.

Es ergibt sich der gleiche minimale Spannbaum wie in Aufgabe a).

Aufgabe 7: Transitive Hülle

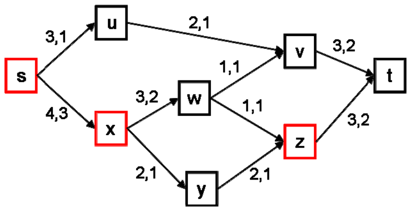
Geben Sie die transitive Hülle für den Graph G1 aus Aufgabe 1 an! (3-57)



Übungsblatt 8

Aufgabe 1: Flüsse in Netzwerken

In dem unten angegebenen Netzwerk N = (G,c,s,t) ist ein Fluss bereits eingetragen.



a) Was sind die Bestandteile eines Netzwerks? *(3-83)*

G = Graph mit den Eigenschaften
- schwach zusammenhängend
- schlicht (d.h. keine Schlingen, keine Mehrfachkanten)
- gerichtet

c = Kapazitätsfunktion E → N^+, die jeder Kante e aus E eine Kapazität c(e) zuweist

s = Quelle, Knoten mit Eingangsgrad 0

t = Senke, Knoten mit Ausgangsgrad 0

b) Ist der angegebene Fluss gültig? *(3-84)*

Ja, er ist gültig, da er folgende Eigenschaften besitzt:

1) $0 \leq f(e) \leq c(e)$

2) Für s und t gilt
$$\sum_{e \in O(s)} f(e) = \sum_{e \in I(t)} f(e)$$

3) Für alle $v \neq s,t$ gilt die Flusserhaltung
$$\sum_{e \in O(v)} f(e) = \sum_{e \in I(v)} f(e)$$

c) Was ist der Wert des Flusses? Lässt sich der Wert des Flusses erhöhen? *(3-85)*

$$d = \sum_{e \in O(s)} f(e) = \sum_{e \in I(t)} f(e) = 4$$

Der Wert lässt sich auf 6 erhöhen, da sowohl auf dem Weg (s, u, v, t) als auch auf dem Weg (s, x, y, z, t) jeweils noch eine freie Kapazität von 1 vorhanden ist.

d) Gegeben sei der Schnitt A(X,X) mit X = {s,x,z}. Welche Kapazität besitzt A, wie groß ist der Wert des Flusses von X nach X (f(X,X))?

Strecke	Kapazität	Wert d. Flusses
s,u	3	1
x,w	3	2
x,y	2	1
z,t	3	2
Gesamt	**11**	**6**

Aufgabe 2: Flüsse u. Netzwerke

a) Was besagt der Satz der Flusserhaltung (natürlichsprachlich u. formal)? *(3-89)*

Der Wert des Flusses der eingehenden Kanten eines Knotens muss dem Wert des Flusses der ausgehenden Kanten entsprechen.

Flusserhaltung in einem Netzwerk:
Der Fluss f im Netzwerk N habe den Wert d, $A(X,\underline{X})$ sei ein Schnitt in N, dann gilt:

1) $d = f(X,\underline{X}) - f(\underline{X},X)$ (Flussbilanz bzgl. Schnitt ist identisch mit dem Fluss im Netzwerk)
2) $d \leq c(X,\underline{X})$ (Fluss wird durch Kapazität beschränkt)

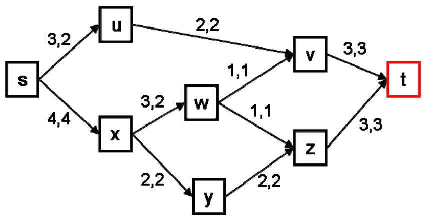
b) Was besagt das Max-Flow-Min-Cut-Theorem? *(3-92)*

Sei N ein Netzwerk mit Kapazitätsfunktion c. N besitzt einen maximalen Fluss f in N mit dem Wert

$\min \{c(X,\underline{X}) : A(X,\underline{X}) \text{ ist ein Schnitt}\}$

Der maximale Fluss entspricht also dem minimalen Schnitt.

c) Erzeugen Sie zu dem unten angegebenen Fluss inkrementell den maximalen Fluss. Zeigen Sie, dass es sich um den maximalen Fluss handelt.



1) (s,u,v,t) → d = 1
2) (s,u,v,t) → d = 2
3) (s,x,w,v,t) → d = 3
4) (s,x,w,z,t) → d = 4
5) (s,x,y,z,t) → d = 5
6) (s,x,y,z,t) → d = 6

Zeige, dass der Fluss maximal ist:

Betrachtet man die Senke t, so haben die eingehenden Kanten eine maximale Kapazität c von 6. Der angegebene Fluss hat ebenfalls einen Wert von 6. Bei einer weiteren Erhöhung wäre die notwendige Eigenschaft, dass der Wert des Flusses die Kapazität nicht übersteigen darf, nicht mehr erfüllt.

Aufgabe 3: Hashing

Erläutern Sie folgende Begriffe:

a) perfektes Hashing *(3-115)*

Ein Hashing ohne Kollisionen nennt man perfektes Hashing.

b) lineares und quadratisches Sondieren *(3-121ff)*

Lineares Sondieren:
Tritt beim Berechnen der Hash-Adresse eine Kollision auf, so wird vom Kollisionsort sequentiell der nächst freie Speicherbereich gesucht, bis ein freier Speicherplatz gefunden oder der Ausgangspunkt erreicht wird.

Quadratisches Sondieren:
Anstelle der sequentiellen Suche wird die Schrittweite beim Quadratischen Sondieren von Schritt zu Schritt modifiziert ($h(s) = (h(s) + i^2)$, i = Anzahl der Kollisionen).

c) doppeltes Hashing *(3-124)*

Beim doppelten Hashing werden zwei voneinander unabhängige Hashfunktionen zum Ermitteln der Adresse für einen Schlüssel verwendet.

Aufgabe 4: Hashing

Es sind Schlüssel aus dem Wertebereich [0 - (106-1)] (6-stellige natürliche Zahlen) in eine Hash-Tabelle mit dem Adressbereich [0 - 999] (3 stellige natürliche Zahlen) abzubilden. Gegeben ist folgende Hash-Funktion (zunächst ohne Kollisionsauflösung):

h(s) = h(s1 s2 s3 s4 s5 s6) = 100*s4 + 10*s5 + 1*s6, d.h. h(s) = s mod 1000

a) **Geben Sie für folgende Schlüssel s die Adresse h(s) an:**

000455, 123456, 223344, 100344, 012300, 123345, 099343, 998345

s	h(s)
000455	455
123456	456
223344	344
100344	344
012300	300
123345	345
099343	343
998345	345

b) **Zur Kollisionsauflösung verwenden Sie eine erweiterte Form des quadratischen Sondierens in Form des so genannten Doppel-Hashings:**

hi(s) = (h(s) + g(s)*i*i) mod 1000.

Verwenden Sie für g die Funktion

g(s) = 100*s1 + 10*s2 + 1*s3

Zunächst verwenden Sie i=1, bei erneuter Kollision setzen Sie i=2, usw. Berechnen Sie die Adressen für die in a) genannten Schlüssel, diesmal mit Kollisionsauflösung.

s	h(s)	$h_1(s)$	$h_2(s)$
000455	455		
123456	456		
223344	344		
100344	344	444	
012300	300		
123345	345		
099343	343		
998345	345	343	337

Übungsblatt 9

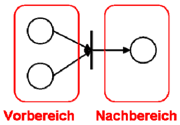
Aufgabe 1: Petri-Netze

a) Petri-Netze sind bipartite Graphen. Was bedeutet das? *(4-3)*

Petri-Netze bestehen aus zwei unterschiedlichen Knotentypen. So bestehen zum Beispiel S/T-Netze aus Stellen und Transitionen.

b) Was versteht man unter dem Vorbereich einer Transition? Was ist der Nachbereich einer Transition? *(4-7)*

Stellen, die zur Transition hinführen, nennt man Vorbereich der Transition.
Stellen, die von der Transition wegführen, nennt man Nachbereich.

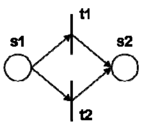


c) Kann der Vorbereich einer Transition aus der leeren Menge bestehen? Wie ist es mit dem Nachbereich? *(4-21)*

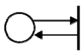
Ja, sowohl der Vor- als auch der Nachbereich einer Transition kann aus der leeren Menge bestehen.

d) Müssen Vor- und Nachbereiche von jeweils zwei oder auch mehreren Transitionen disjunkt sein? Kann der Vorbereich einer Transition identisch mit ihrem Nachbereich sein? Geben Sie jeweils kleine Beispiele an, welche die Situation illustrieren. *(4-31)*

Nein, mehrere Transitionen dürfen jeweils identische Vor- und Nachbereiche haben.

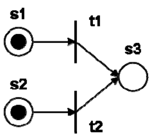


Ja, der Vorbereich einer Transition kann identisch mit dem Nachbereich sein.

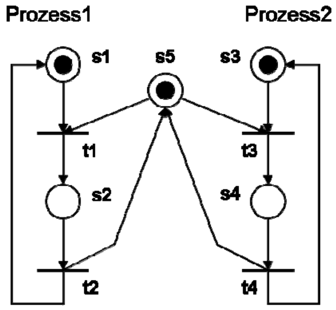


e) Betrachten Sie zwei aktivierte Transitionen, welche disjunkte Vorbereiche besitzen und daher auch nicht um Marken konkurrieren. Können diese Transitionen dann gleichzeitig schalten? *(4-13)*

Nein, gleichzeitiges Schalten ist in einem Petri-Netz nicht möglich. Es kann immer nur eine Transition schalten.



Aufgabe 2: Petri-Netze und wechselseitiger Ausschluss

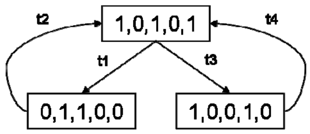


a) Geben Sie zwei zulässige Schaltfolgen der Länge > 4 an! ***(4-40f)***

b) Geben Sie für jede dieser Schaltfolgen den Häufigkeitsvektor an! ***(4-40f)***

a) f1 = t1, t2, t1, t2, t1 b) w(f1) = (3,2,0,0)
f2 = t3, t4, t1, t2, t3 w(f2) = (1,1,2,1)

c) Erstellen Sie den Erreichbarkeitsgraphen! ***(4-38)***

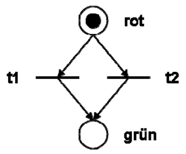


d) Ist der wechselseitige Aussschluss auch noch unter den Startmarkierungen M = (2,0,3,0,1) und M' = (1,0,1,0,2) gewährleistet?

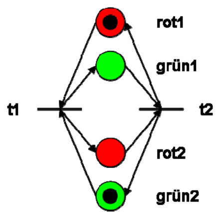
Für den wechselseitigen Ausschluss ist es für dieses spezielle Netz nötig, dass auf den Stellen S_2, S_4, und S_5 zusammen nur eine Marke liegt. Deswegen funktioniert der wechselseitige Ausschluss für die Markierung M, für die Markierung M' jedoch nicht.

Aufgabe 3: Verkehrsampel als S/T-Netz

Das folgende S/T-Netz zeigt den permanenten rot-grün-Wechsel an einer Verkehrsampel.



a) Konstruieren Sie ein S/T-Netz, welches zwei gekoppelte Ampeln darstellt, die jeweils gegensätzliche Farben zeigen.



b) Welche Möglichkeiten existieren, die Korrektheit der Schaltung nachzuweisen?

- Markenspiel
- Inzidenzmatrix *(4-10)*
- Erreichbarkeitsbaum *(4-38)*
- Überdeckungsbaum *(4-38)*

Übungsblatt 10

Aufgabe 1: S/T-Netze

a) Was ist ein S/T-Netz? *(4-9)*

Ein S/T-Netz ist ein spezielles Petri-Netz mit Stellen und Transitionen als Knoten.

Formale Definition:
Ein Stellen/Transitions-Netz ist ein 4-Tupel ST = (S, T, C^-, C^+), wobei gilt:
1) S ist eine endliche, nichtleere Menge von n Stellen, als $|S| = n$.
2) T ist eine endliche, nichtleere Menge von m Transitionen, also $|T| = m$.
3) Die Rückwärts-Inzidenzdmatrix C^- ist eine n x m - Matrix mit C^-_{ij} aus N_0.
4) Die Vorwärts-Inzidenzmatrix C^+ ist eine n x m - Matrix mit C^+_{ij} aus N_0.

b) Welche Lebendigkeitsgrade unterscheidet man innerhalb von S/T-Netzen? *(4-29f)*

Lebendigkeitsgrade von Transitionen:

- Eine Transition t eines markierten S/T-Netzes (S, T, C^-, C^+, M_0) heißt *tot*, wenn sie unter keiner erreichbaren Markierung aktiviert ist.
- Eine Transition t heißt *aktivierbar*, wenn sie unter mindestens einer Folgemarkierung aktiviert ist.
- Eine Transition t heißt *lebendig*, wenn sie unter allen Folgemarkierungen aktivierbar ist.

Lebendigkeitsgrade von S/T-Netzen:

- Ein markiertes S/T-Netz heißt *tot*, wenn alle Transitionen tot sind.
- Ein markiertes S/T-Netz heißt *schwach lebendig* oder *deadlockfrei*, wenn es unter keiner Folgemarkierung tot ist, d.h. mindestens eine Transition muss lebendig sein.
- Ein markiertes S/T-Netz heißt *stark lebendig* oder *lebendig*, wenn all seine Transitionen lebendig sind.

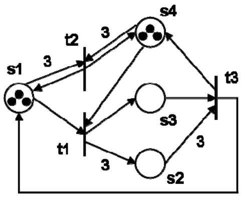
c) Was bedeutet Beschränktheit eines S/T-Netzes? *(4-36)*

Ein S/T-Netz heißt beschränkt, wenn die Markenanzahl auf jeder Stelle eine gegebene obere Schranke nicht überschreiten kann. So bedeutet z.B. k-Beschränktheit, dass bei jeder erreichbaren Markierung auf der Stellen maximal k Marken liegen.

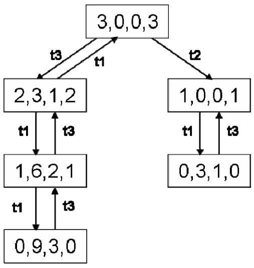
d) Wie lässt sich für ein gegebenes markiertes S/T-Netz eine Folgemarkierung rechnerisch ermitteln? *(4-15, 4-42)*

Für eine erreichbare Folgemarkierung M und einen Häufigkeitsvektor w (mit einer gültigen Schaltfolge) gilt: $M = M_0 + C * w$.
Die direkte Folgemarkierung M' ist gegeben durch $M' = M + C * e_j$.

Aufgabe 2: Erreichbarkeitsgraphen



a) Geben Sie für das angegebene Netz den Erreichbarkeitsgraphen an.



b) Welchen Lebendigkeitsgrad haben die Transitionen t1, t2, t3 und das Gesamtsystem?

Transitionen t1 und t3 sind lebendig, Transitionen t2 ist aktivierbar. Somit ist das Gesamtsystem schwach lebendig.

c) Welche Eigenschaften hat ein Erreichbarkeitsgraph, wenn ein System tot, schwach lebendig, stark lebendig ist? *(4-32)*

Wenn es eine Markierung gibt, aus der keine Kante herausführt, so ist das System in dieser Markierung tot. Wenn diese Markierung die Startmarkierung M_0 ist, so ist das Gesamtsystem tot.

Das System ist schwach lebendig, wenn jeder Knoten mindestens eine ausgehende Kante besitzt.

Das System ist stark lebendig, wenn für jeden Knoten eine gültige Kantenfolge existiert, die alle Transitionen des Netzes als Etikett besitzt.

d) Wie lässt sich der Lebendigkeitsgrad einer Transition aus einem Erreichbarkeitsgraphen ermitteln?

tot: von einem bestimmten Knoten aus ist sie nicht mehr Bestandteil für alle gültigen Kantenfolgen

lebendig: für jeden Knoten gibt es eine Kantenfolge, die diese Transition als Etikett enthält

aktivierbar: für mind. einen Knoten gibt es eine Kantenfolge, die diese Transition als Etikett enthält

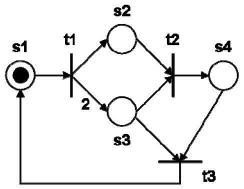
aktiviert: für den Startknoten gibt es eine Kantenfolge der Länge 1, die diese Transition als Etikett enthält

e) Grenzen Sie Erreichbarkeitsbaum, Erreichbarkeitsgraph und Überdeckungsbaum voneinander ab!

Der Erreichbarkeitsbaum bildet innere Knoten des Erreichbarkeitsgraphen als Blatt ab.

Der Überdeckungsbaum beschreibt einen unendlichen Erreichbarkeitsbaum mit endlichen Mitteln.

Aufgabe 3: Eigenschaften von Petri-Netzen



a) Ermitteln Sie, ob die Markierung (0,0,1,1) erreichbar ist. Wie ist das rechnerisch zu belegen?

$M = M_0 + C * w$

$M_0 = (1,0,0,0)$

$$C = \begin{pmatrix} -1 & 0 & 1 \\ 1 & -1 & 0 \\ 2 & -1 & -1 \\ 0 & 1 & -1 \end{pmatrix}$$

Gleichungssystem:

$0 = 1 - w_1 + w_3$
$0 = w_1 - w_2$
$1 = 2w_1 - w_2 - w_3$
$1 = w_2 - w_3$

Lösungsvektor:

$w = (\lambda+1, \lambda+1, \lambda), \lambda \in \mathbb{N}_0$

Die Markierung (0,0,1,1) ist also zum Beispiel mit dem gültigen Häufigkeitsvektor (1,1,0) erreichbar.

b) Ermitteln Sie rechnerisch die Markierung, die sich aus der Schaltfolge f = (t1,t2,t3,t1) ergibt.

$M = M_0 + C * w$

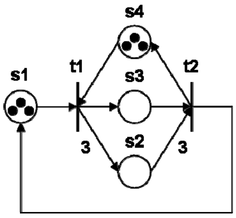
$w = (2,1,1)$

$$M = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} -1 & 0 & 1 \\ 1 & -1 & 0 \\ 2 & -1 & -1 \\ 0 & 1 & -1 \end{pmatrix} * \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 0 \end{pmatrix}$$

c) Überprüfen Sie das angegebene Petri-Netz auf folgende Eigenschaften:
- Lebendigkeitsgrad der Transition t3
- Lebendigkeitsgrad des Systems

Sowohl die Transition t3 als auch das Gesamtsystem sind stark lebendig.

Aufgabe 4: S/T-Invarianten



a) Was sind S- und T-Invarianten (formale Definition, Interpretation)? *(4-54, 4-59)*

S-Invariante:
v * M ist konstant für alle von M erreichbaren Markierungen.
Ein Vektor $v \in Z^n$, $v \neq 0$, heißt S-Invariante genau dann, wenn v * C = 0.
Zusätzlich muss gelten, dass $v * M = v * M_0$ für alle von M_0 erreichbaren Markierungen.

T-Invariante:
Nach einer gültigen Schaltfolge wird die Ausgangsmarkierung wieder erreicht.
Ein Häufigkeitsvektor w aus Z^m, $w \neq 0$, heißt T-Invariante genau dann, wenn gilt C * w = 0.

b) Argumentieren Sie, welche S- und T-Invarianten zu erwarten sind!

S-Invarianten: $v_1 = (1,0,1,0)$, $v_2 = (0,0,1,1)$, $v_3 = (3,1,0,0)$, $v_4 = (0,1,0,3)$
T-Invarianten: w = (1,1)

c) Berechnen Sie die S- und T-Invarianten!

$$C = \begin{pmatrix} -1 & 1 \\ 3 & -3 \\ 1 & -1 \\ -1 & 1 \end{pmatrix}$$

S-Invarianten: v * C = 0
$-v_1 + 3v_2 + v_3 - v_4 = 0$
$v_1 - 3v_2 - v_3 + v_4 = 0$

T-Invarianten: C * w = 0
$w_1 = w_2$

d) Geben Sie für die gegebene Markierung die sog. Summenschreibweise der S-Invarianten an!

v_1: #s1 + #s3 = 3
v_2: #s3 + #s4 = 3
v_3: 3 * #s1 + #s2 = 9
v_4: #s2 + 3 * #s4 = 9

e) Argumentieren Sie mittels der S-Invarianten, ob die Markierung (1,4,2,1) erreichbar ist.

Die Markierung erfüllt die S-Invarianten v_3 und v_4 nicht, somit ist sie nicht erreichbar.

f) Welche Schaltfolgen führen zu der Ausgangsmarkierung?

Alle T-Invarianten, also Schaltfolgen mit dem Häufigkeitsvektor $w = (\lambda,\lambda)$, $\lambda \in \mathbb{N}_0$.

Übungsblatt 11

Aufgabe 1: S/T-Netze

a) Was bedeutet Reproduzierbarkeit bei S/T-Netzen? *(4-58)*

Reproduzierbarkeit einer Markierung bedeutet, dass diese Markenbelegung nach einer Folge von Transitionsschaltungen wieder auftritt (→ T-Invariante).

b) Nennen Sie eine notwendige Bedingung für die Erreichbarkeit einer Markierung M eines S/T-Netzes. *(4-42)*

Für eine erreichbare Folgemarkierung M und einen gültigen Häufigkeitsvektor w gilt:
$M = M_0 + C * w$

c) Was bedeutet Widerspruchsfreiheit in einem B/E-Netz? *(4-76)*

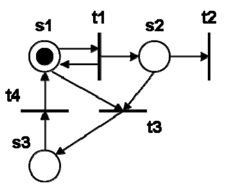
Ein B/E-Netz heißt widerspruchsfrei, wenn für alle Stellen S_i gilt:
Entweder Stelle S_i oder ihr Gegenteil S_j trägt eine Marke.

d) Können Sie mit Hilfe einer Erreichbarkeitsanalyse die gleichen Aussagen gewinnen wie durch eine Invariantenanalyse?

Invariantenanalyse: $C * w = 0$, die Markierung spielt keine Rolle
Erreichbarkeitsanalyse: $M' = M_0 + C * W$, mehr Analysemöglichkeiten

	Erreichbarkeitsanalyse	Invariantenanalyse
Beschränktheit	✓	✓
Lebendigkeit	✓	x
Reproduzierbarkeit	✓	✓

Aufgabe 2: Überdeckungsbaum



a) Geben Sie für das oben angegebene Petri-Netz die formale Definition an.

$ST = (S, T, C^+, C^-, M_0)$
$S = \{s_1, s_2, s_3\}$
$T = \{t_1, t_2, t_3, t_4\}$
$M_0 = (0,1,1)$

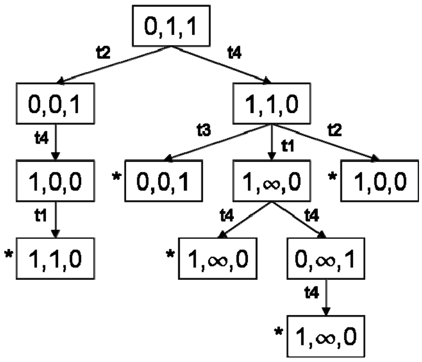
C^+ (t_j über den Spalten, s_i bei den Zeilen)

	1	2	3	4
1	1	0	0	1
2	1	0	0	0
3	0	0	1	0

C^-

	1	2	3	4
1	1	0	1	0
2	0	1	1	0
3	0	0	0	1

b) Erstellen Sie für das unten angegebene Netz unter der Startmarkierung M0 = (0,1,1) den Überdeckungsbaum.



c) Überprüfen Sie die Transitionen und das gesamte Netz bzgl. ihrer Lebendigkeit unter der M' = (0,0,1).

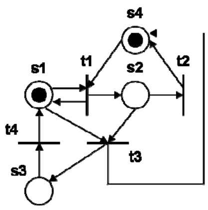
Da alle Blätter als innere Knoten wieder vorkommen, gibt es keine Markierung in der das Netz tot ist. Es sind alle Transitionen und somit auch das gesamte Netz (stark) lebendig.

d) Zeigen Sie, dass das Netz unbeschränkt ist.

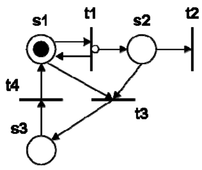
Die Stelle s2 ist unbeschränkt → im Überdeckungsbaum an dem Zeichen ∞ erkennbar. Wenn eine Stelle unbeschränkt ist, so ist das gesamte Netz unbeschränkt.

e) Wie lässt sich das angegebene Netz 1-beschränken?

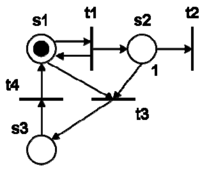
Möglichkeit 1 (zusätzliche Kontrollstelle s4):



Möglichkeit 2 (Verbotskante):



Möglichkeit 3 (Stellenkapazität beschränken):



f) Ist das 1-beschränkte S/T-Netz vollständig?

Nein, da z.B. t2 einen Vorbereich, aber keinen Nachbereich besitzt.

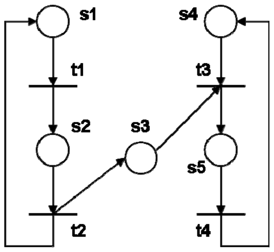
g) Ermitteln Sie die S- und T-Invarianten ohne Rechnung.

S-Invarianten:	T-Invarianten:
$v_1 = (1,0,1)$	$w(f_1) = (1,0,1,1)$
	$w(f_2) = (1,1,0,0)$

h) Zeigen Sie über S-Invarianten, dass die Markierung M'' = (2,0,1) von M_0 nicht erreichbar ist.

Die Markierung M'' erfüllt die S-Invariante #s1 + #s3 = 1 nicht, also ist sie nicht erreichbar.

Aufgabe 3: Produzent / Konsument



a) Ermitteln Sie für das oben angegebene Produzenten/Konsumenten-Netz die S- und T-Invarianten ohne Rechnung.

S-Invarianten:
v1 = (1,1,0,0,0)
v2 = (0,0,0,1,1)

T-Invarianten:
$w(f_1) = (1,1,1,1)$

b) Berechnen Sie die S- und T-Invarianten!

C $\qquad t_j$

		1	2	3	4
	1	-1	1	0	0
	2	1	-1	0	0
s_i	**3**	0	1	-1	0
	4	0	0	-1	1
	5	0	0	1	-1

S-Invariante: v * C = 0

-v1 + v2 = 0
v1 - v2 + v3 = 0
-v3 - v4 + v5 = 0
v4 - v5 = 0

v1 = v2, v3 = 0, v4 = v5 → S-Invarianten (1,1,0,0,0) und (0,0,0,1,1)

T-Invariante: C * w = 0

-w1 + w2 = 0
w1 - w2 = 0
w2 - w3 = 0
-w3 + w4 = 0
w3 - w4 = 0

w1 = w2 = w3 = w4 → T-Invariante (1,1,1,1)

c) Wie lassen sich die S- und T-Invarianten interpretieren?

S-Invarianten: v * M ist konstant für alle von M erreichbaren Markierungen.
T-Invarianten: Nach einer gültigen Schaltfolge wird die Ausgangsmarkierung wieder erreicht.

d) Zeigen Sie, dass die Folgemarkierung M'' = (1,0,5,1,0) erreichbar ist (sowohl rechnerisch als auch durch Markenspiel).

$M = M_0 + C * w$
Annahme: $M_0 = (1,0,0,1,0)$
gültiger Häufigkeitsvektor w = (5,5,0,0)

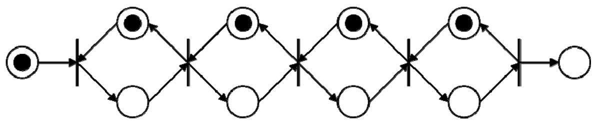
e) Zeigen Sie, dass die Markierung M''' = (2,0,1,1,0) nicht erreichbar ist.

S-Invariante #s1 + #s2 = 1 nicht erfüllt.

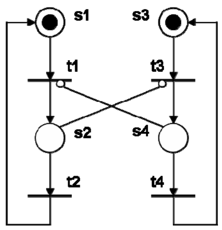
Aufgabe 4: Modellierung

Entwerfen Sie (möglichst kleine) Petri-Netze, welche folgende Systeme modellieren. Prüfen Sie durch Markenspiel, Erreichbarkeitsanalyse und durch Suchen nach Invarianten die Eigenschaften ("Korrektheit") Ihres Entwurfs.

a) Eine FIFO-Warteschlange (First-In-First-Out) der Aufnahmekapazität 4, welche die aus einer Quelle eintreffenden Marken in der gegebenen Reihenfolge in eine Senke weiterreicht.



b) Modellieren Sie den wechselseitigen Ausschluss unter Verwendung von Verbotskanten.



Übungsblatt 12

Aufgabe 1:

a) Was versteht man unter einem stochastischen Petri-Netz? *(5-4)*

Ein stochastisches Petri-Netz (SPN) basiert auf einem Stellen-/Transitions-Netz, bei dem jeder Transition t_i, i = 1,…,m eine exponentielle Schaltrate λ_i zugeordnet ist. Damit kann ein stochastisches Petri-Netz dargestellt werden als

$SPN = (S, T, C^-, C^+, M_0, \Lambda)$

wobei $ST = (S, T, C^-, C^+)$ ein S/T-Netz, M_0 eine Anfangsmarkierung und $\Lambda = (\lambda_1,\ldots, \lambda_m)$ ein Vektor von Schaltraten ist.

b) Erläutern Sie die Beziehung zwischen mittlerer Schaltrate und mittlerer Schaltdauer. *(5-4)*

mittl. Schaltdauer und mittl. Schaltrate verhalten sich reziprok zueinander:
mittl. Schaltdauer = $1 / \lambda_i$

c) Wann besitzt ein Prozess die Markov-Eigenschaft? *(5-10)*

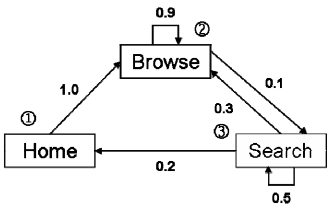
Wenn die Historie des Prozesses, d.h. die in der Vergangenheit angenommenen Zustände, irrelevant für seine zukünftige Entwicklung ist ("Gedächtnislosigkeit").

d) Welche Anforderungen werden an "gute" Zufallszahlen gestellt? *(5-22)*

- unabhängig	n-te Zufallszahl ist unabhängig von allen Vorgängern
- gleichverteilt	in einem Intervall (a,b), meist (0,1)
- große Periode	bevor eine Wiederholung der Zahlen eintritt
- reproduzierbar	damit Simulationsprogramm deterministisch ist
- einfach und schnell berechenbar	

Aufgabe 2: Diskrete Markovkette

Die folgende zeitdiskrete Markovkette zeigt das Verhalten eines Benutzers im WWW. Die Startverteilung sei p(t=0) = (1.0, 0.0, 0.0).



a) Geben Sie die stochastische Transitionsmatrix P an. *(5-12)*

$$P = \begin{pmatrix} 0.0 & 1.0 & 0.0 \\ 0.0 & 0.9 & 0.1 \\ 0.2 & 0.3 & 0.5 \end{pmatrix}$$

b) Berechnen Sie p(t=2).

$p(t+1) = p(t) * P$

$$p(t=1) = \begin{pmatrix} 1.0 \\ 0.0 \\ 0.0 \end{pmatrix} * \begin{pmatrix} 0.0 & 1.0 & 0.0 \\ 0.0 & 0.9 & 0.1 \\ 0.2 & 0.3 & 0.5 \end{pmatrix} = \begin{pmatrix} 0.0 \\ 1.0 \\ 0.0 \end{pmatrix}$$

$$p(t=2) = \begin{pmatrix} 0.0 \\ 1.0 \\ 0.0 \end{pmatrix} * \begin{pmatrix} 0.0 & 1.0 & 0.0 \\ 0.0 & 0.9 & 0.1 \\ 0.2 & 0.3 & 0.5 \end{pmatrix} = \begin{pmatrix} 0.0 \\ 0.9 \\ 0.1 \end{pmatrix}$$

Aufgabe 3: Pseudo-Zufallszahlen

Gegeben ist ein gemischter Kongruenzgenerator laut Definition im Skript mit folgenden Parametern: a = 4, m = 10, c = 3, x_0 = 2

a) **Berechnen Sie die erzeugte Zufallszahl für i = 4.**

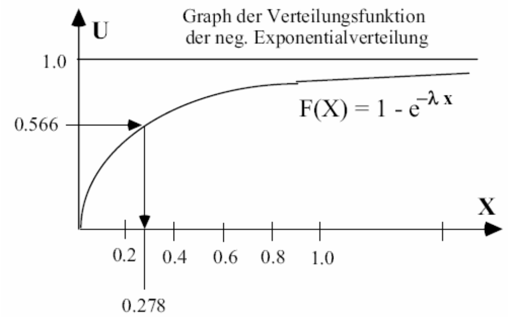
$x_{i+1} = (a * x_i + c) \bmod m$
$x_1 = (4 * 2 + 3) \bmod 10 = 1$
$x_2 = (4 * 1 + 3) \bmod 10 = 7$
$x_3 = (4 * 7 + 3) \bmod 10 = 1$
$x_4 = (4 * 1 + 3) \bmod 10 = 7$

b) **Erzeugt dieser Generator gute Zufallszahlen? Wie könnte man den Generator verbessern?**

Nein, der Generator hat eine zu kleine Periode (2), ist nicht unabhängig und nicht gleichverteilt.

Mögliche Verbesserungsmöglichkeiten:
- m deutlich größer wählen, evlt. als Primzahl
- a deutlich größer wählen

c) **Wie wandeln SIe die erzeugten Zufallszahlen in exponentialverteilte Zufallszahlen um? Was muss bei der Transformation beachtet werden?**



Gegeben: Wahrscheinlichkeit $u_j \in (0,1)$
Gesucht: Wert von der Zufallsvariablen mit der Verteilungsfunktion F(x)
Antwort: Umkehrfunktion $F^{-1}(u_j) = x_j$

Voraussetzung ist, dass die Umkehrfunktion existiert.