Übungsaufgaben- Lösungen

1. Aussagenlogik

- Bitte kreuzen Sie an (☒), ob die folgende logische Formel erfüllbar, falsifizierbar, allgemeingültig (Tautologie) und/oder unerfüllbar (Kontradiktion) ist.
  $((C \wedge \neg A \wedge B) \vee (A \wedge B)) \Rightarrow C \vee A$

☒ erfüllbar ❑ falsifizierbar ☒ Tautologie ❑ Kontradiktion

A	B	C	**1** $C \wedge \neg A \wedge B$	**2** $A \wedge B$	**3** $1 \vee 2$	**4** $C \vee A$	$3 \Rightarrow 4$
W	W	W	F	W	W	W	W
W	W	F	F	W	W	W	W
W	F	W	F	F	F	W	W
W	F	F	F	F	F	W	W
F	W	W	W	F	W	W	W
F	W	F	F	F	F	F	W
F	F	W	F	F	F	W	W
F	F	F	F	F	F	F	W

- Überprüfen Sie, ob für die logischen Formeln φ und ξ gilt:
  $\varphi \models \xi$ $\xi \models \varphi$ und/ oder $\varphi \equiv \xi$

  φ: $A \Rightarrow B \vee C$ ξ: $\neg C \Rightarrow (\neg B \Rightarrow \neg A)$

  Führen Sie den Beweis
  a) durch Aufstellen der Wahrheitstabelle und

A	B	C	**1** $B \vee C$	φ $A \Rightarrow 1$	**2** $\neg B \Rightarrow \neg A$	ξ $\neg C \Rightarrow 2$
W	W	W	W	W	W	W
W	W	F	W	W	W	W
W	F	W	W	W	F	W
W	F	F	F	F	F	F
F	W	W	W	W	W	W
F	W	F	W	W	W	W
F	F	W	W	W	W	W
F	F	F	F	W	W	W

b) durch Umformen der Formeln.

$A \Rightarrow B \vee C$ ➢ $\neg A \vee B \vee C$

$\neg C \Rightarrow (\neg B \Rightarrow \neg A)$ ➢ $C \vee (\neg B \Rightarrow \neg A)$ ➢ $C \vee B \vee \neg A$

c) Kennzeichnen Sie mittels ☒, zu welchem Ergebnis Sie gekommen sind.

☒ $\varphi \models \xi$ ☒ $\xi \models \varphi$ ☒ $\varphi \equiv \xi$

2. Prädikatenlogik

Übersetzen Sie folgende umgangssprachlichen Sätze in eine prädikatenlogische Formel:

- Jeder Mensch weiß etwas.
  $\forall X\ mensch(X) \Rightarrow \exists Y\ (wissen(Y) \land weiss(X,Y))$
- Kein Mensch weiß alles.
  $\neg\exists X\ (mensch(X) \land \forall Y\ (wissen(Y) \Rightarrow weiss(X,Y)))$
- Manche Menschen wissen gar nichts.
  $\exists X\ (mensch(X) \land \forall Y\ (wissen(Y) \Rightarrow \neg weiss(X,Y)))$
  $\exists X\ (mensch(X) \land \neg\exists Y\ (wissen(Y) \land weiss(X,Y)))$
- Kein Mensch kennt jemanden (einen Menschen), der alles weiß.
  $\neg\exists X\ \exists Y\ (mensch(X) \land mensch(Y) \land kennt(X,Y) \land \forall Z\ (wissen(Z) \Rightarrow weiss(Y,Z)\ )\ )$
- Jeder Mensch duzt sich mit irgendjemandem (einem Menschen).
  $\forall X\ mensch(X) \Rightarrow \exists Y\ (mensch(Y) \land duzt(X,Y))$
- Es gibt eine Mannschaft, gegen die der FC St. Pauli noch nie verloren hat.
  $\exists X\ mannschaft(X) \land \neg verlieren(stpauli, X)$
- In Deggendorf scheint im Sommer immer die Sonne.
  $\exists X\ sommer(X) \Rightarrow sonne\_scheinen(deggendorf,X)$
- Nachts sind alle Katzen grau.
  $\exists X\ nachts(X) \land \forall Y\ katze(Y) \Rightarrow farbe(Y,grau,X)$
- Die Bayern lieben ihre Nachbarn.
  $\forall X,Y\ bayer(X) \land nachbar(X,Y) \Rightarrow lieben(X,Y)$
- Die Deggendorfer essen nur Fleisch.
  $\forall X,Y\ deggendorfer(X) \land essen(X,Y) \Rightarrow fleisch(Y)$
- Ich kenne jemanden, der jemanden kennt, der schon mal in Australien war.
  $\exists X \exists Y\ mensch(X) \land kennen(ich,X) \land mensch(Y) \land kennen\ (X,Y) \land war(Y,australien)$
- Eine Aktie ist attraktiv, wenn ihr Kurs steigt.
  $\forall X\ aktie(X) \land kurs(X,steigen) \Rightarrow attraktiv(X)$
- Eine Aktie habe ich.
  $\exists X\ aktie(X) \land haben(X,ich)$

3. Resolutionsbeweis

- Aussagenlogik:
  - Beweisen Sie mittels Resolution, ob die folgende Schlussfolgerung aus den gegebenen Sätzen abgeleitet werden kann oder nicht.
    Gegeben:
    Ich trage Gummistiefel (A) oder die Sonne scheint (B). $A \lor B$
    Die Sonne scheint nicht ($\neg B$) oder ich trage einen Sonnenhut (C). $\neg B \lor C$
    Schlussfolgerung:
    Ich trage Gummistiefel (A) oder ich trage einen Sonnenhut (C). $A \lor C$

KNF:
(1) $A \lor B$
(2) $\neg B \lor C$
Zu beweisende Formel negiert hinzufügen:
(3) $\neg(A \lor C)$ wird zu (3) $\neg A$ (4) $\neg C$

Beweis *beispielsweise*:
(5) (1)+(3) Resolvente B
(6) (1)+(4) Resolvente $\neg$B
(7) (5)+(6) Resolvente {} w.z.b.w.

- Prädikatenlogik
  - **Gegeben** sind die folgenden Formeln
    Pferde sind schneller als Hunde.
    $\forall P\ \forall H\ (pferd(P) \wedge hund(H)) \Rightarrow schneller(P,H)$
    Es gibt einen Windhund, der schneller als jeder Hase ist.
    $\exists W\ \forall R\ windhund(W) \wedge (hase(R) \Rightarrow schneller(W,R))$
    Ein Windhund ist ein Hund.
    $\forall W\ windhund(W) \Rightarrow hund(W)$
    Fury ist ein Pferd.
    pferd(fury)
    Bunny ist ein Hase.
    hase(bunny)
    Beweisen Sie mittels Resolution, dass Fury schneller als Bunny ist.

$\forall P\ \forall H\ (pferd(P) \wedge hund(H)) \Rightarrow schneller(P,H)$
$\exists W\ \forall R\ windhund(W) \wedge (hase(R) \Rightarrow schneller(W,R))$
$\forall W\ windhund(W) \Rightarrow hund(W)$
pferd(fury)
hase(bunny)
Zu beweisen: **schneller(fury, bunny)**

KNF:
(1) $\neg$pferd(P) $\vee$ $\neg$hund(H) $\vee$ schneller(P,H)
(2) windhund(W)
(3) $\neg$hase(R) $\vee$ schneller(W1,R)
(4) $\neg$schneller(A,B), $\neg$schneller(B,C), schneller(A,C)
(5) $\neg$windhund(W2) $\vee$ hund(W2)
(6) pferd(fury)
(7) hase(bunny)
Zu beweisende Formel negiert hinzufügen:
(8) $\neg$schneller(fury, bunny)

Beweis *beispielsweise*:
(9) (4)+(8) Resolvente $\neg$schneller(fury,B) $\vee$ $\neg$schneller(B,bunny)
(10) (1)+(9) Resolvente $\neg$pferd(fury) $\vee$ $\neg$hund(B) $\vee$ $\neg$schneller(B,bunny)
(11) (6)+(10) Resolvente $\neg$hund(B) $\vee$ $\neg$schneller(B,bunny)
(12) (5)+(11) Resolvente $\neg$windhund(B) $\vee$ $\neg$schneller(B,bunny)
(13) (2)+(12) Resolvente $\neg$schneller(B,bunny)
(14) (3)+(13) Resolvente $\neg$hase(bunny)
(15) (7)+(14) Resolvente {} w.z.b.w.

- Gegeben sind die folgenden Formeln

  Politiker mögen niemanden, der knausrig ist.

  $\forall P, X\ politiker(P) \wedge knausrig(X) \Rightarrow \neg mag(P,X)$

  $\forall P\ politiker(P) \Rightarrow \neg\exists X\ mag(P, X) \wedge knausrig(X)$

  Jeder Politiker mag eine Firma.

  $\forall P\ politiker(P) \Rightarrow \exists X\ firma(X) \wedge mag(P,X)$

  Es gibt Politiker.

  $\exists P\ politiker(P)$

  Beweisen Sie, dass es eine Firma gibt, die nicht knausrig ist.

  $\exists F\ firma(F) \wedge \neg knausrig(F)$

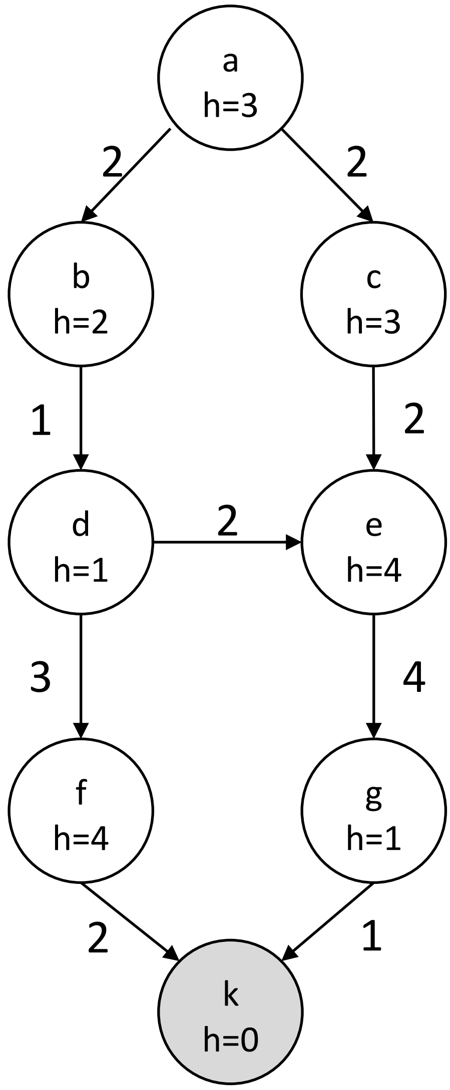
(1) $\forall P\ politiker(P) \Rightarrow \neg\exists X\ mag(P, X) \wedge knausrig(X)$
(2) $\forall P\ politiker(P) \Rightarrow \exists X\ firma(X) \wedge mag(P,X)$
(4) $\exists P\ politiker(P)$
Negiertes Ziel
(5) $\neg(\exists F\ firma(F) \wedge \neg knausrig(F))$

KNF:
(1) $\neg politiker(P) \vee \neg mag(P, X) \vee \neg knausrig(X)$
(2) $\neg politiker(P1) \vee firma(f(P1))$
(3) $\neg politiker(P2) \vee mag(P2,f(P2))$
(4) $politiker(a)$
(5) $\neg firma(F) \vee knausrig(F)$

Beweis beispielsweise:
(6) (1)+(5) Resolvente $\neg politiker(P) \vee \neg mag(P, X) \vee \neg firma(X)$
(7) (6)+(3) Resolvente $\neg politiker(P2) \vee \neg firma(f(P2))$
(8) (7)+(2) Resolvente $\neg politiker(P1)$
(9) (8)+(4) Resolvente {} w.z.b.w.

4. Suche

Aufgabe1:
Gegeben ist der nachfolgende Graph
Startknoten ist a, Zielknoten ist k.



Die Kosten stehen an den Kanten, im Knoten ist die Heuristik notiert. Wie finden Suchstrategien die Lösung?

Hinweis:
Wenn 2 Nachfolgeknoten die gleiche Bewertung haben, soll der bevorzugt werden, der im Alphabet weiter vorn steht.

Gesucht ist pro Suchalgorithmus

- Die Reihenfolge der besuchten Knoten.
- Der Pfad zum Ziel.

Suchalgorithmen:

- **Tiefensuche**:
  - Besuchte Knoten: a,b,d,e,g,k
  - Pfad: a,b,d,e,g,k
- **Breitensuche**:
  - Besuchte Knoten: a,b,c,d,e,f,g,k
  - Pfad: a,b,d,f,k
- **Bestensuche**:
  - Besuchte Knoten: a,b,d,c,e,g,k
  - Pfad: a,b,d,e,g,k
- **Hill Climbing**:
  - Besuchte Knoten: a,b,d lokales Maiximum
  - Pfad: -
- **Hill Climbing with Backtracking**:
  - Besuchte Knoten: a,b,d,c lokales Maximum
  - Pfad: -
- **Nearest Neighbour Heuristic**:
  - Besuchte Knoten: a,b,d,e,g,k
  - Pfad: a,b,d,e,g,k
- **A /A***:
  - Besuchte Knoten: a,b,d,c,e,g,k
  - Pfad: a,c,e,g,k

Findet der A* Algorithmus den optimalen Weg?

☐ ja ☒ nein

Falls Nein, woran liegt das?

Heuristik für f wird überschätzt. Beste Lösung mit Kosten=8: a,b,d,f,k

Aufgabe2:
Gegeben ist ein Brett mit 5 Feldern, links 2 weiße, rechts 2 schwarze Steine, der mittlere Platz ist leer.

Start

○	○		●	●

Ziel

●	●		○	○

Man darf

- Einen Stein schieben
- Über einen anderen Stein springen oder
- Über 2 Steine springen

Kosten: Schieben/Springen über 1 Steine=1, über 2 Steine=2
Wie löst man das Problem mittels Suche?

Das geht mit A* Algorithmus.
Darstellung bspw.:
Zustände: 5 Felder vom Brett [w, w, e, b, b]
Startzustand: [w, w, e, b, b]
Zielzustand: [b, b, e, w, w]
Zustandsübergänge:
Ein Stein, der direkt neben dem e liegt, tauscht mit e den Platz (schieben)
Ein Stein, der 2 Positionen vom e weg ist, tauscht mit e den Platz (springen über 1 Stein)
Ein Stein, der 3 Positionen vom e weg ist, tauscht mit e den Platz (springen über 2 Steine)
Kosten g: Kosten 1 bzw. 2
Heuristik h: Summe der Entfernungen der beiden weißen Steine zu ihren Zielfeldern.

5. Bayessche Formel

- Im Mittel sagt der Wetterbericht für den kommenden Tag zu 60% schönes und zu 40% schlechtes Wetter voraus; die Trefferquote liegt für die Voraussage "schön" bei 80% und für die Voraussage "schlecht" bei 90 %.
  (a) Wieviel % schöne Tage gibt es?
  (b) Trotz schönen Wetters ist Kumpel Bob nicht zum verabredeten Fallschirmsprung erschienen mit dem Hinweis, der gestrige Wetterbericht wäre schlecht gewesen, so dass er anders disponierte. Mit welcher Wahrscheinlichkeit war dies bei Unkenntnis des gestrigen Wetterberichts nur eine Ausrede?

Aus der Aufgabe ergeben sich die folgenden Wahrscheinlichkeiten:
P (SV+ = Schön vorausgesagt) = 0,6
P (SV- = schlecht vorausgesagt) = 0,4
P (S+ = schönes Wetter | SV+) = 0,8
P (S- = schlechtes Wetter | SV-) = 0,9

a) gesucht: P(S+)

	SV+	**SV-**	
S+	0,48	0,04	0,52
S-	0,12	0,36	0,48
	0,6	0,4	

Oder
P(S+) = P(S+|SV+) *P(SV+) + P(S+|SV-)*P(SV-) = 0,8 * 0,6 + 0,1 * 0,4 = 0,52 = 52%

b) Gesucht ist P(SV+ | S+)

$P\ (SV+ \mid S+) = \frac{P\ (S+|SV+) * P\ (SV+)}{P\ (S+)} = \frac{0{,}8 * 0{,}6}{0{,}52} = 0{,}923 \approx 92\%$

- In Deggendorf wird im Mittel zu 10% Schwarzgefahren. 70% der Schwarzfahrer haben keine Fahrkarte, während die anderen 30% gefälschte oder illegal besorgte Karten besitzen. Von den ehrlichen Fahrgästen haben im Mittel 5% ihre Fahrkarte vergessen. Mit welcher Wahrscheinlichkeit ist ein kontrollierter Fahrgast, der keine Karte vorzeigen kann, ein Schwarzfahrer?

Aus der Aufgabe ergeben sich die folgenden Wahrscheinlichkeiten:
P (S = Schwarzfahrer) = 1/10= 0,1
P (K = keine Karte | S) = 7/10= 0,7
P (F = falsche Karte | S) = 3/10= 0,3
P (K|E = ehrliche Fahrgäste) = 0,05

Gesucht: Ereignis P (Schwarzfahrer | keine Karte)

$P\ (S|K) = \frac{P\ (K|S) * P\ (S)}{P\ (K)}$

Wir brauchen dafür P(K):

	S+	E=(S-)	
K+	0,03	0,855	0,885
K-	0,07	0,045	0,115
	0,1	0,9	

Oder
P(K) = P(K|E) *P(E) + P(K|S)*P(S) = 0,05 * 0,9 + 0,7 * 0,1 = 0,115

so dass sich als Ergebnis ergibt

$P(S \mid K) = \frac{0{,}7 * 0{,}1}{0{,}115} \approx 60{,}8\% \approx 61\%$

6. Fuzzy

Tante Erna möchte in den Urlaub fahren. Ihre Kriterien sind:
- Der Urlaub soll möglichst in die Ferienzeit 20.6. – 10.8. fallen.
- Sie will an die See.
- Die Reise soll preiswert oder der Urlaubsplatz “sonnensicher” sein.

Es liegen ihr drei Angebote vor:

Reiseziel	Zeitraum	Preis
Hamburg	19.6.-2.7.	1000
Bordeaux	1.7.-14.7.	2000
Mallorca	30.7.-12.8.	1800

Aus diesen Angeboten soll mit Hilfe der Fuzzy-Logik das Angebot auswählt werden, welches am besten auf Tante Ernas Anforderungen passt. Die Fuzzy-Werte für die Bewertung sind wie folgt gegeben.

Achtung: nachfolgend wurde nach Fuzzy-Wert sortiert!!!

Preis	**Fuzzy-Wert**
2000	0,2
1800	0,33
1000	0,87

Sonne	**Fuzzy-Wert**
Hamburg	0,8
Bordeaux	0,9
Mallorca	0,95

See	**Fuzzy-Wert**
Bordeaux	0,8
Hamburg	0,95
Mallorca	0,99

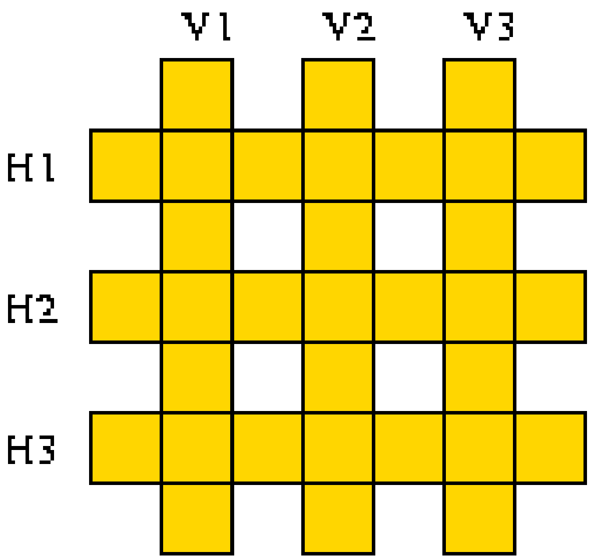
Zeit	**Fuzzy-Wert**
30.7.-12.8.	0,86
19.6.-2.7.	0,93
1.7.-14.7.	1

Ermitteln Sie die Bewertung für alle 3 Reisen mittels der 3 Ihnen bekannten Berechnungsvorschriften.

Reiseziel	**V1**	**V2**	**V3**
Hamburg	0,87	0,88	0,86
Bordeaux	0,8	0,8	0,74
Mallorca	0,86	0,85	0,82

7. Constraints

Gegeben sind das folgende Kreuzgitter und die 6 Lösungsworte.



abalone
abandon
enhance
anagram
connect
elegant

Wie würden Sie dieses Problem als Constraint-Solving-Problem beschreiben?

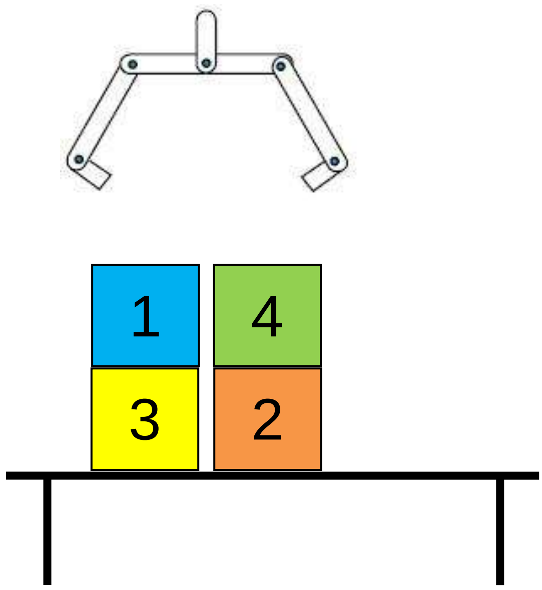
6 Constraintvariablen: H1, H2, H3, V1, V2, V3
Wertebereich jeder dieser Variablen {abalone, abandon, enhance, anagram, connect, elegant}
Notation:
H11 -> erster Buchstabe des Wortes H1
H12 -> zweiter Buchstabe des Wortes H1
usw.usf.
Constraint: (Name, Variablenmenge, Relation)

1. (C1,{H1,V1},H12=V12)
2. (C2,{H1,V2},H14=V22)
3. (C3,{H1,V3},H16=V32)
4. (C4,{H2,V1},H22=V14)
5. (C5,{H2,V2},H24=V24)
6. (C6,{H2,V3},H26=V34)
7. (C7,{H3,V1},H32=V16)
8. (C8,{H3,V2},H34=V26)
9. (C9,{H3,V3},H36=V36)
10. (C10,{H1,H2,H3,V1,V2,V3},H1≠H2≠H3≠V1≠V2≠V3)

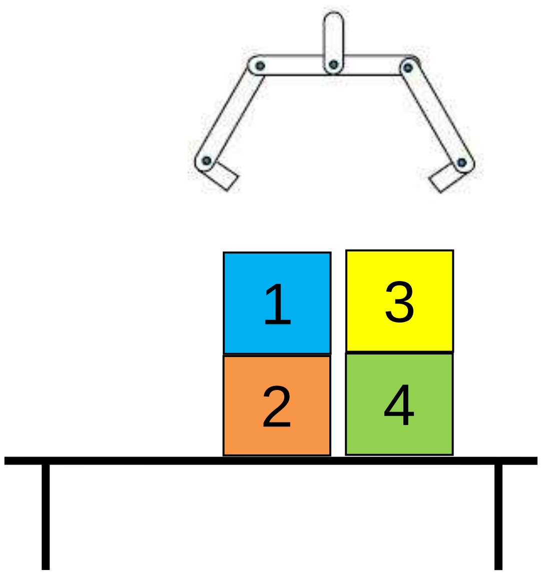
8. Planen

Blocksworld: Mittels STRIPS soll das folgende Problem gelöst werden: Ein Greifarm kann ein freies Klötzchen greifen (und anheben) sowie bewegen (und absetzen). Auf dem Tisch ist genügend Platz, um weitere Türmchen zu bauen.
Bitte notieren Sie die Zustandsbeschreibungen und Operatoren entsprechend.

Ausgangszustand



Zielzustand



Zustandsbeschreibung:
liegt_auf(wer, auf_wem)
ist_frei(wer)
hält(wen)

Anfangszustand:
liegt_auf(block1, block3), liegt_auf(block3, tisch), liegt_auf(block4, block2), liegt_auf(block2, tisch), ist_frei(block1), ist_frei(block4), ist_frei(greifer)

Endzustand:
liegt_auf(block1, block2), liegt_auf(block2, tisch), liegt_auf(block3, block4), liegt_auf(block4, tisch), ist_frei(block1), ist_frei(block3), ist_frei(greifer)

Operatoren:

- **greifen(wen)**
  Vorbedingung:
  ist_frei(greifer), ist_frei(wen), liegt_auf(wen,wo)
  add-Liste
  ist_frei(wo), hält(wen)
  delete-Liste
  ist_frei(greifer), ist_frei(wen), liegt_auf(wen,wo)

- **bewegen(wen,wohin)**:
  Vorbedingung:
  ist_frei(wohin), hält(wen)
  add-Liste
  ist_frei(greifer), liegt_auf(wen,wohin), ist_frei(wen)
  delete-Liste
  ist_frei(wohin), hält(wen)

Könnte man dieses Problem auch mit metrisch monotoner Suche lösen?

☐ ja ☒ nein

Falls Nein, woran liegt das?

Das Teilziel *ist_frei(greifer)* muss temporär verletzt werden.