Prozeßdatenverarbeitung und Robotik 3 Sommersemester 1992

Hauptklausur

Name: _____

Matr.-Nr.: _____ Fachbereich: _____

Bitte sauber und *nicht* mit rotem Stift schreiben! Die Fragen sind genau und in gebotener Kürze (wo möglich: *Stichworte*) zu beantworten. Die Antworten sind auf die Teilfragen zu beziehen. Sowohl Vorder- als auch Rückseite der Aufgabenblätter können zur Beantwortung der Fragen verwendet werden. Weitere Blätter werden bei Bedarf von den Betreuern ausgeteilt. Schreiben Sie auf jedes Blatt Ihren Namen und Ihre Matrikelnummer! Bitte vergewissern Sie sich, daß alle Blätter vorhanden sind. Es werden sechs Aufgaben gestellt, die alle zu bearbeiten sind.
Kreuzen Sie hier bitte an, ob Sie eine Übungs- oder eine Prüfungsklausur schreiben möchten, und bestätigen Sie Ihre Entscheidung durch Ihre Unterschrift.

____ Übungsklausur ____ Prüfungsklausur _____

Unterschrift

Mit dieser Unterschrift versichern Sie außerdem an Eides Statt, daß Sie im Fach PDVROB 3 noch keine Prüfungs- bzw. Übungsleistung erbracht haben. Klausuren, bei denen Unterschrift oder Kreuz fehlt, werden *nicht gewertet*.

Aufgabe	1	2	3	4	5	6	Summe	Note
Punkte								
Tendenz								

Aufgabe 1: Grundlagen: Prolog (15 Punkte)

a) Bei der Ausführung eines Operators im folgenden Strips-Programm (siehe Seite 3) wird die Hinzufügeliste durch `append` zum aktuellen Zustand hinzugefügt. Dabei wird davon ausgegangen, daß die Elemente der Hinzufügeliste nicht schon im Zustand enthalten sind.

Schreiben Sie ein Prologprädikat

```
vereinigung(Add,Zustand1,NeuerZustand),
```

das diese Voraussetzung nicht macht.

b) Im vorgegebenen Stripsprogramm gibt es einen *cut* (!). Wird dadurch in diesem Anwendungsgebiet der Ablauf des Programms grundlegend geändert, oder dient der *cut* nur zu Beschleunigung des Programms? Begründen Sie Ihre Antwort.

c) Ergänzen Sie das Stripsprogramm auf der folgenden Seite so, daß Zyklen durch eine Tiefenbeschränkung vermieden werden. Begründen Sie Ihre (sinnvolle!) Wahl der maximalen Tiefe für eine Klötzchenwelt mit drei Klötzchen.

d) Ändern Sie die Klausel

```
strips( [BL, op(Operator, Add)|Ziele], Zustand,
        [Operator|Operationen]) :-
    konjunktion_erfuellbar(BL, Zustand),
    entferne(BL, Zustand, Zustand1),
    append(Add, Zustand1, NeuerZustand),
    strips(Ziele, NeuerZustand, Operationen).
```

so um, daß Sie mit dem Klauselkopf

```
strips( [op(Operator)|Ziele], Zustand,
        [Operator|Operationen])
```

funktioniert.

e) Betrachten Sie die dritte Regel des folgenden Strips-Programms. In dieser Klausel erscheint dreimal der Ausdruck `[K_Ziel | K_Ziele]`, ohne daß eine der beiden Variablen später einzeln benutzt wird. Beschreiben Sie die Auswirkungen wenn der Ausdruck durch `[K_Ziele]` ersetzt würde.

STRIPS-Programm zu Aufgabe 1

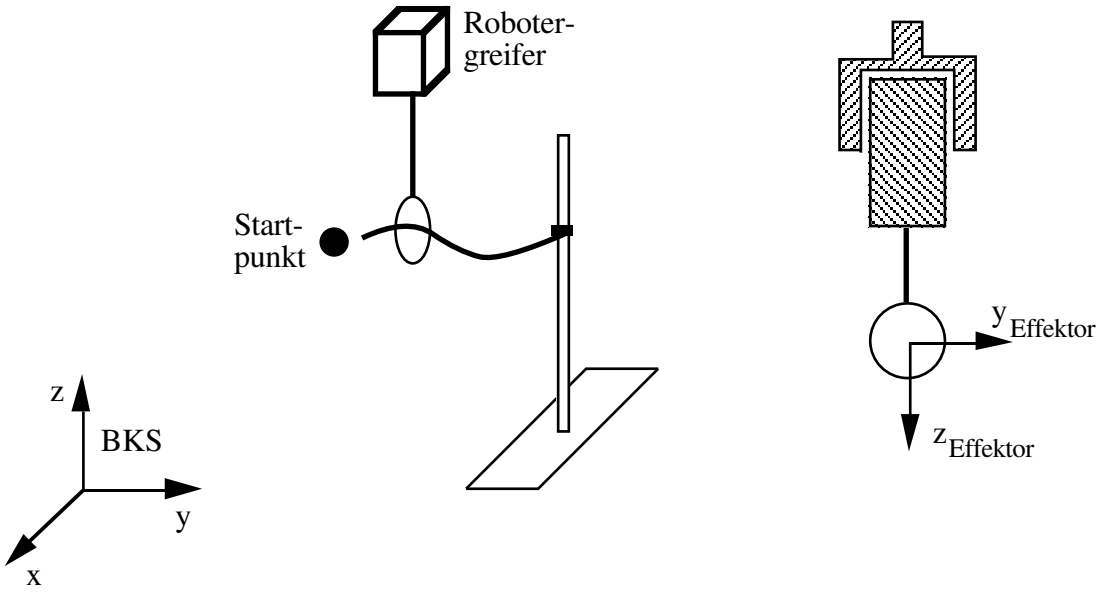
```
/*  operator( Operator, Bedingungen&Loeschen, Add )            */

operator( pickup( X ),
          [ ontable( X ), clear( X ), handempty ],
          [ holding( X ) ]).
operator( putdown( X ),
          [ holding( X )],
          [ ontable( X ), clear( X ), handempty ]).
operator( unstack( X, Y ),
          [ on( X, Y ), clear( X ) , handempty ],
          [ clear( Y ), holding( X ) ]).
operator( stack( X, Y ),
          [ clear( Y ), holding( X ) ],
          [on( X, Y ), clear( X ), handempty ]).

/*  Wenn nur noch die Konjunktion der Ziele vorhanden ist, dann
    ueberpruefe, ob diese Ziele erfuellt sind.                 */
strips( [ Ziele ] , ZielZustand,  [ ] ):-
    konjunktion_erfuellt( Ziele, ZielZustand).

/*  Wenn ein ausfuehrbarer Operator vorliegt und alle
    Bedingungen erfuellt sind, dann fuehre den Operator aus.   */
strips( [ BL, op( Operator, Add) | Ziele ], Zustand,
        [ Operator | Operationen ]) :-
    konjunktion_erfuellt( BL, Zustand ),
    entferne( BL, Zustand, Zustand1 ),
    append( Add, Zustand1, NeuerZustand ),
    strips( Ziele, NeuerZustand, Operationen ).

/*  Wenn die Konjunktion der Ziele nicht erfuellt ist, dann
    fuege die Einzelziele vor der Konjunktion ein. Versuche, die
    Ziele zu erfuellen.                                        */
strips([[ K_Ziel | K_Ziele ] | Ziele], Zustand, Operationen):-
    append([ K_Ziel | K_Ziele ],
           [[ K_Ziel | K_Ziele ] | Ziele], NeuesZiel ),
    strips( NeuesZiel, Zustand, Operationen ).

/*  Wenn ein Ziel erreicht ist, dann bearbeite die weiteren
    Ziele.                                                     */
strips( [ Ziel | Ziele ], Zustand, Operationen ) :-
    member( Ziel, Zustand ), !,
    strips( Ziele, Zustand, Operationen ).

/*  Wenn ein Ziel nicht erreicht ist, dann suche einen Operator,
    der dieses Ziel erreichen kann. Fuege die Bedingungen des
    Operators und deren Konjunktion vor den Operator und die
    uebrigen Ziele ein. Versuche, die Bedingungen des Operators
    zu erfuellen.                                              */
strips( [ Ziel | Ziele ], Zustand, Operationen ) :-
    waehle_op( Ziel, Zustand, Operator ),
               operator( Operator, BL, Add ),
    strips( [ BL, op( Operator,  Add ) | Ziele ], Zustand,
            Operationen ).
```

Aufgabe 2: Programmierung von Industrierobotern (AdaIR) (15 Punkte)

Versuchsaufbau: Innerhalb des Roboterarbeitsraumes ist ein gekrümmter Draht montiert, dessen Kurvenverlauf mit Hilfe des Roboters ermittelt werden soll. Dazu ist der Greifer des Roboters mit einem ringförmigen Sensor (Durchmesser 20 mm) ausgestattet, der eine Unterbrechung auslöst, falls der Draht berührt wird.

Schreiben Sie ein AdaIR-Programm, mit dessen Hilfe der im Raum gekrümmte Draht auf einer Länge von 100 mm abgetastet werden kann.

Abtaststrategie: Der Startpunkt ist konstant und vorgegeben undkollisionsfrei (`C_STARTPUNKT`). Die Abtastung der Kurve beginnt, indem sich der Roboter in 2 mm Schritten solange entlang der positiven y-Achse des BKS bewegt bis eine Unterbrechung ausgelöst wird. An dieser Stelle ist der Sensor durch geeignete Bewegungen so zu positionieren, daß der Draht sich in der Mitte des Ringes befindet. Dabei muß der Sensor **nicht** gedreht werden. Auf diese Weise wird weiter verfahren, bis die Abtastlänge von 100 mm erreicht ist. Die ermittelten Zentrumspunkte sind in Form einer Bahn abzuspeichern.



Aufgabe 3: Fragen zur temporalen Logik (15 Punkte)

1. Zwischen den Intervallen A und B gelten die folgenden Beziehungen:

a)	A	(m = mi)	B		B		A
b)	A	(fi f)	B		B		A

Tragen Sie in der rechten Spalte, in der gleichen Schreibweise, jeweils die Beziehung zwischen B und A ein.

2. Zwischen den Intervallen A, B und C gelten die folgenden Beziehungen:

a)	A	(>)	B	(=)	C	A		C
b)	A	(fi)	B	(di)	C	A		C
c)	A	(m mi)	B	(f fi)	C	A		C

Tragen Sie in der rechten Spalte die jeweilige Beziehung zwischen A und C ein.

3.)

a) Stellen Sie in der Notation der temporalen Logik dar, daß die beiden Intervalle A und B ohne Überschneidung direkt aufeinander folgen (dabei liegt die Reihenfolge nicht fest)

A B

b) Stellen Sie in der Notation der temporalen Logik dar, daß das Intervall A beendet ist bevor das Intervall B beendet ist

A B

Aufgabe 4: Fragen zur Planung (15 Punkte)

Hinweis: In allen Fragen zu Strips und zur Klötzchenwelt sind die Operatoren wie in den Vorlesungsunterlagen definiert.

a) Begründen Sie, warum die Konstruktion von Roboterplänen meist besser mit B-Regeln erreichbar ist.

b) In den praktischen Aufgaben zur Vorlesung sollten Sie eine Abstripswelt für einen mobilen Roboter modellieren. Eins der dabei auftretenden Probleme ist die Suche von Wegen von einem Raum in einen anderen. Begründen Sie, warum der Einsatz von Strips zur Wegsuche nicht sinnvoll ist.

c) Nennen Sie ein Problem, das Strips nicht optimal lösen kann.

d) Gibt es in der in den Vorlesungsunterlagen definierten Klötzchenwelt Probleme die Strips nicht lösen kann?

e) Erläutern Sie kurz den Begriff "Gradientenmethode".

f) Erläutern Sie kurz den Begriff "Heuristik".

Aufgabe 5: Fragen zu Strips und Abstrips (20 Punkte)

a) Planen Sie mit Hilfe des Strips-Algorithmus ausgehend von der Startstellung

```
ontable(B), ontable(A), on(C,A), clear(C), clear(B)
```

die Operatoren, die zur Zielstellung

```
on(A,B), on(B,C)
```

führen. Geben Sie dazu jeweils den Zielkeller und den aktuellen Zustand bis zur ersten Ausführung eines stack-Operators an. Sich wiederholende Teile des Zielkellers sind nicht jedesmal anzugeben.

b) Wie lautet Ihr gefundener Plan ?

c) Beschreiben Sie, an welchen Stellen im Strips-Algorithmus eine nichtdeterministische Auswahl möglich ist.

d) Ist der Strips-Algorithmus in der Lage einen optimalen Plan für den Aufgabenteil a zu finden? Begründen Sie Ihre Antwort.

e) In einer Welt, in der mit Abstrips geplant werden soll, ist der Operator O(X,Y) durch die folgenden Bedingungs-, Hinzufüge- und Löschliste definiert:

Bedingungsliste:	P1, P2(X,Y), P3(Z)
Hinzufügeliste:	P4, P5(X,Z), P3(Y)
Löschliste:	P1, P2(X,Y), P3(Z)

Die Schwierigkeitsgrade der Prädikate P1 bis P5 sind in drei Stufen klassifiziert
P1 und P4 sind leicht zu erreichen
P2 und P5 sind schwierig zu erreichen
P3 liegt im Schwierigkeitsgrad dazwischen

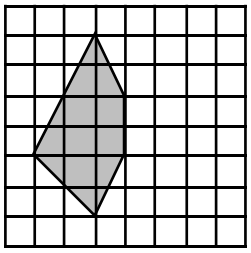
Die Planung auf der obersten, abstraktesten Planungsebene hat für das Ziel P2(A,B),P3(C),P2(A,C),P3(F), P4
den folgenden Plan ergeben:

O(A,B), O(A,C)

Geben Sie den Zielkeller für die nächste Planungsebene an.

Aufgabe 6: Fragen zur Wegplanung (20 Punkte)

6.1) Stellen Sie das folgende Objekt durch einen Würfelbaum (Quadtree) dar. Die kleinste Verfeinerungsstufe ist durch die Gitterlinien bestimmt.



6.2)

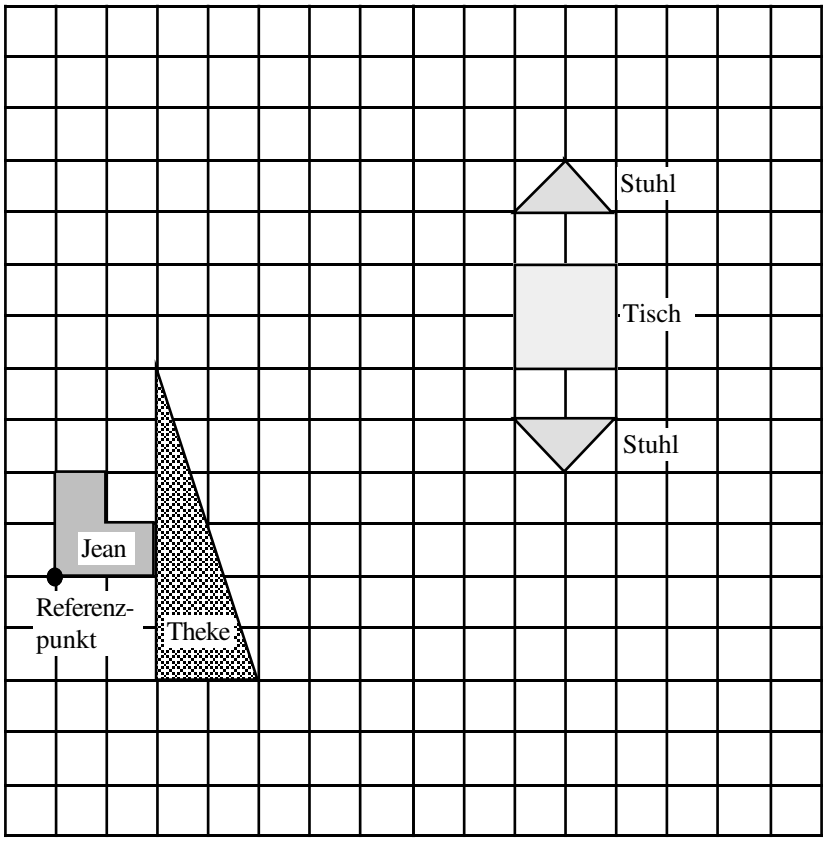
a) Wovon hängt die Anzahl der Dimensionen eines Konfigurationsraumes ab?

b) Wieviele Dimensionen besitzt der Konfigurationsraum für eine Fracht, die fliegen kann? Begründen Sie Ihre Antwort.

c) Wird mit den in den Vorlesungsunterlagen vorgestellten Wegplanungsverfahren immer ein Weg gefunden, wenn einer existiert? Begründen sie Ihre Antwort und unterscheiden Sie zwischen den Fällen mit und ohne Rotationen.

6.3)

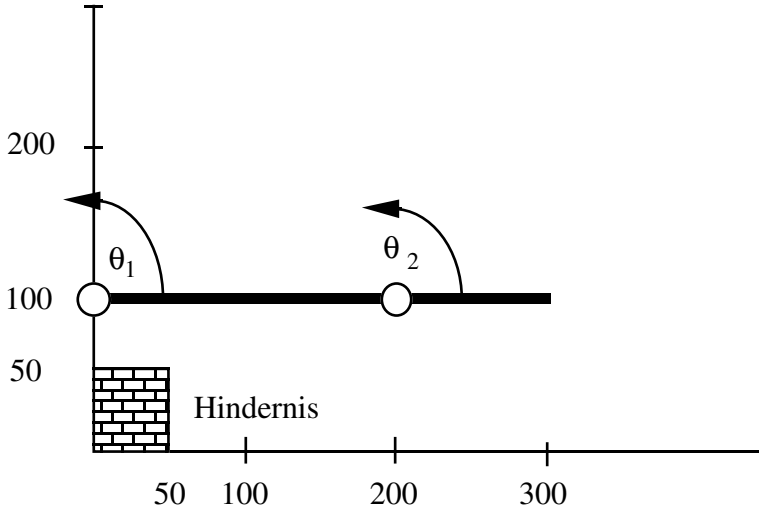
a) Der Roboter Jean bedient in einer Bar mit einer Theke, einem Tisch und zwei Stühlen seine Gäste. Helfen Sie Jean bei der kollisionsfreien (!) Wegplanung, indem Sie die vier Konfigurationshindernisse zeichnen. Beachten Sie dabei, daß Jean sich nicht drehen kann.



b) Zeichnen Sie den Sichtbarkeitsgraphen ein

6.4)

Geben ist der folgende zweiachsige Roboter in seiner Arbeitsumgebung:



Roboter im Arbeitsraum in Nullstellung

Dabei besitzen die Rotationsgelenke die folgenden Verfahrbereiche:

$$-270° \leq \theta_1 \leq 0°$$

$$-90° \leq \theta_2 \leq 90°$$

Die Pfeile geben die positive Bewegungsrichtung an.

Zeichnen Sie den Konfigurationsraum mit den Konfigurationshindernissen.