Student name: _____
Student number: _____

Faculteit der Exacte Wetenschappen

Tentamen Ontwerp van Multi-agentsystemen / Design of Multi-Agent Systems

Vrije Universiteit Amsterdam **11 februari 2004**

Opgave/Exercise	**1**	**2**	**3**	**4**	**bonus**
Punten/points	25	25	20	20	10

Normering:	*Norm:*
Het **tentamencijfer T** is gelijk aan het totaal behaalde punten voor de tentamenopgaven plus de bonus punten gedeeld door 10.	The tentamination mark **T** equals sum of the points scored for the exercises plus 10 bonus points divided by 10.
Het **Eindcijfer** voor het hoorcollege Ontwerp van Multi-agentsystemen wordt als volgt berekend.	The endmark **Eindcijfer** for the course Design of Multi-Agent Systems is calculated as follows:
Eindcijfer = (T + H + P) / 3	
Waarbij	Where
T = tentamencijfer H = cijfer huiswerkopgave P = cijfer voor het klein practicum	T = tentamination mark H = mark for the home work exercises P = mark for the small practicum

U treft aan:

4 opgaven
5 appendices

You find:

4 exercises
5 appendices

Student name: _____
Student number: _____

Opgave 1 (25 punten)

Lees Appendix 1 en beantwoord de volgende drie vragen.

Opgave 1a (5 points)

Geef een grafische representatie van het top-level van een proces abstractie voor het MOBIE systeem. Laat de menselijke klanten als agenten voorkomen. Motiveer elke link die je tussen processen aanbrengt en leg uit welke soorten informatie worden uitgewisseld.

Opgave 1b (10 points)

In hoofdstuk 6 van de syllabus bestaat de meest complexe agent uit 6 verschillende componenten: agent_interaction_management, world_interaction_management, maintenance_of_-agent_information, maintenance_of_world_information, own_process_control, en agent_specific_task. Welk van deze componenten heb je nodig en welke niet om een persoonlijk assistent agent van het MOBIE systeem te modelleren? Motiveer je antwoord en refereer expliciet aan de beschrijving van het MOBIE systeem.

Opgave 1c (10 points)

De interacties die de persoonlijk agent kan hebben zijn nogal complex. Het soort communicatie dat de agent heeft met zijn menselijke klanten verschilt van de communicatie met andere software agenten. Bovendien moet de agent voor de communicatie met mensen gebruik kunnen maken van verschillende communicatiekanalen. Stel dat de component comp_c van de agent verantwoordelijk is voor al deze vormen van interactie. Dan is het handig om de component op te bouwen uit andere componenten. Geef een proces compositie van comp_c en de informatie verbindingen die binnen comp_c nodig zijn om deze interactie processen te modelleren. Motiveer je antwoord in een rationale.

English:

Question 1 (25 points)

Read Appendix 1 and answer the following three questions.

Student name: _____
Student number: _____

Question 1a (5 points)

Provide a graphical representation of the top-level of process abstraction for the MOBIE system. Include the human customers as agents in that picture. Motivate each link between processes, and explain the types of information exchanged.

Question 1b (10 points)

In Chapter 6 the most complex agent is composed of 6 different components agent_interaction_management, world_interaction_management, maintenance_of_agent_information, maintenance_of_world_information, own_process_control, and agent_specific_task. Which of these components do you need and which do you not need to model a personal assistant agent of the MOBIE system? Motivate your answer and make explicit references to the description of the MOBIE system

Question 1c (10 points)

The process of interaction that the personal agent needs is rather complicated. The type of communication that it needs with the human customers is rather different from that with the other software agents. Furthermore, for communication with the human user it has different channels. Suppose that component comp_c of the agent is responsible for all this. Then component comp_c should be composed. Provide a process composition of comp_c and the links needed within comp_c to model these processes. Motivate your answers in a rationale.

English:

Question 2 (25 points). Een Nederlandse vertaling van Question 2 is niet beschikbaar.

This question builds on your understanding of the generic model for Reasoning with and about Assumptions (Chapter 11). For your convenience a rather detailed partial specification of that model is given in Appendix 3. Be careful to focus directly on the parts of the specification that you need, so that you don't waste time. This generic model will be used in this exercise to diagnose the problems of growing sunflowers. Read Appendix 2 "Horse Shape Problem".

Student name: _____
Student number: _____

Question 2a (10 points)

Give a knowledge base for component assumption_determination that reflects the knowledge in Appendix 2. Motivate your answer in a rationale.

Question 2b (10 points)

Give a knowledge base of component observation_result_prediction that reflects the knowledge in Appendix 2. Motivate your answer in a rationale.

Question 2c (5 points)

Design the information types causes and symptoms for this domain. You can do this in one information type, but you are also allowed to make more levels of abstraction. Motivate your answers, refer back to your answers to questions a) and b) as well.

Opgave 3 (20 punten).

Het doel van deze opgave is het bestuderen van informatie toestanden, informatie links, en het revisie proces. Beschouw weer de partiele specificatie van Appendix 3. Neem aan dat de informatie types causes and symptoms hebben de volgende specificatie:

information type **causes**
relations a;
end information type

information type **symptoms**
relations b;
end information type

Opgave 3a (10 points)

Beschouw informatie link predictions van de specificatie. Beschouw de volgende informatie toestanden:

output informatie toestand van component observation_result_prediction is

[predicted_for(b, pos, a, pos)].

input informatie toestand van component assumption_evaluation is

[assumed(a, pos), predicted_for(b, neg, a, pos)]

Geef de input informatie toestand van component assumption_evaluation na uitvoering van de link predictions op basis van deze informatie toestanden. Motiveer Uw antwoord.

Student name: _____
Student number: _____

Opgave 3b(10 points)

Geef een trace van het gedrag van component assumption_evaluation gegeven dat de achtereenvolgende **input informatie toestanden** van die component als in Appendix 4 gepresenteerd zijn. Gebruik de speciale antwoordformulieren uit Appendix 4. Motiveer Uw antwoord.

English:

Question 3 (20 points).

The purpose of this question is to study information states, information links and the revision process. The partial specification of Appendix 3 is used again. Suppose that the information types causes and symptoms are specified as follows:

information type **causes**
relations a;
end information type

information type **symptoms**
relations b;
end information type

Question 3a(10 points)

Consider information link predictions of the specification. Consider the following information states:

output information state of component observation_result_prediction is

[predicted_for(b, pos, a, pos)].

input information state of component assumption_evaluation is

[assumed(a, pos), predicted_for(b, neg, a, pos)]

Give the input information state of component assumption_evaluation after execution of link predictions on the basis of the above information states. Motivate your answer.

Question 3b(10 points)

Give a trace of the behaviour of component assumption_evaluation given that the subsequent **input information states** of that component are as is presented in Appendix 4. Use the answer sheet and fill in your answer in Appendix 4. Motivate your answer.

Student name: _____
Student number: _____

Opgave 4 (20 punten)

Het onderwerp van deze opgave betreft informatietoestanden en redeneren. Bestudeer de partiële specificatie van Appendix 5. Gegeven is de object level public information state S van component mouse_a.

S = [observation_result(at_position(self, p0), pos),
observation_result(at_position(food, p1), pos),
observation_result(at_position(screen, p0), neg)]

Opgave 4a (8 punten)

Geef een informatietoestand S' die S verfijnt en bovendien gesloten en consistent is ten opzichte van de kennisbank van component mouse_a.

Opgave 4b (12 punten)

- Motiveer dat S' een verfijning (refinement) is van S (4 punten).
- Motiveer dat S' gesloten (closed) is ten opzichte van de kennisbank van component mouse_a (4 punten).

Motiveer dat S' consistent is ten opzichte van de kennisbank van component mouse_a (4 punten).

English:

Question 4 (20 points)

This question is about information states and reasoning. Study the partial specification of Appendix 5. This is the object level public information state S of component mouse_a.

S = [observation_result(at_position(self, p0), pos),
observation_result(at_position(food, p1), pos),
observation_result(at_position(screen, p0), neg)]

Question 4a (8 points)

Provide an information state S' that refines S and is also closed and consistent with respect to the knowledge base of component mouse_a.

Student name: _____
Student number: _____

Question 4b (12 points)

- Motivate that S' is a refinement of S (4 points).
- Motivate that S' is closed with respect to the knowledge base of component mouse_a (4 points).
- Motivate that S' is consistent with respect to the knowledge base of component mouse_a (4 points).

Student name: _____
Student number: _____

APPENDIX 1 Het MOBIE systeem (also in English)

Het gebruik van prepay mobiele telefoons is de laatste jaren sterk toegenomen. Het aanvullen van het bel-tegoed moet echter nog steeds vrijwel geheel door de gebruiker zelf worden uitgevoerd. Er is een systeem nodig dat het bel-tegoed van een gebruiker automatisch kan ophogen en zich daarbij houdt aan de persoonlijke wensen van die gebruiker. Dit systeem gaat MOBIE heten. Het MOBIE multi-agent systeem bestaat uit persoonlijke assistent agenten voor de klanten en zakelijke agenten voor de aanbieders van mobiele diensten. Het MOBIE systeem moet zorgen voor de personalisatie van de agenten, voor veiligheid, en moet de mens verschillende modaliteiten voor interactie bieden.

Om automatisch het bel-tegoed op te kunnen hogen moeten de aanbieders van telefonische diensten op een stabiele en betrouwbare manier met de persoonlijk assistent agenten kunnen interacteren. Gegeven het grote aantal van dat soort interacties, moet ook dit proces geautomatiseerd worden. In dit project is de keuze is om speciale zakelijke agenten te introduceren die deze interacties aankunnen.

De persoonlijk agenten die de gebruikers moeten vertegenwoordigen kunnen de volgende hoofdtaken uitvoeren:

1. De persoonlijke agent createert en onderhoudt een profiel van de gebruiker. Het profiel bevat tenminste:
    a. De criteria waaronder de agent het bel-tegoed moet ophogen.
    b. De informatie die nodig is om de ophoging uit te kunnen voeren, zoals de bedragen waarmee de agent mag ophogen, en informatie waarmee de betaling kan worden uitgevoerd.
2. De persoonlijke agent vergelijkt de criteria tegen het actuele bel-tegoed.
3. De persoonlijke agent vraagt de zakelijke agent om noodzakelijke informatie zoals:
    a. Het huidige bel-tegoed.
    b. Het gebruiks-patroon van het mobieltje voor een specifieke periode.
4. De persoonlijke agent om het bel-tegoed op te hogen.
5. De persoonlijke agent kan de aanbieder van de telefonische diensten (via de zakelijke agent) vragen om het bel-tegoed van de gebruiker met bedrag x op te hogen.
6. In overeenstemming met het klantprofiel houdt de persoonlijke agent de klant op de hoogte.
7. De persoonlijke agent via de volgende kanalen met de klant interacteren:
    a. Het Internet,
    b. WAP (als de klant een WAP telefoon heeft)
    c. Spraak.

De persoonlijke assistent agenten functioneren in een omgeving die bestaat uit zakelijke agenten die de aanbieders van telefonische diensten vertegenwoordigen, financiele instellingen (zoals banken waarmee de betaling uiteindelijk mee geregeld moet worden) en de klanten. De persoonlijk assistent agenten nemen niet zelf contact op met de financiele instellingen. Ze vragen de aanbieder van de telefonische diensten om het bel-tegoed op te hogen, de aanbieder neemt dan contact op met de juiste financiele instelling.

Student name: _____
Student number: _____

APPENDIX 1 The MOBIE system (ook in het NL)

Prepay usage as a percentage of overall mobile (also called cell) phone access has increased sharply over the past several years. However, the recharging process is still largely manual with personalization provided by the user. A system is needed capable of automatically recharging the prepaid account of a mobile phone in a personalized manner. This visionary system is called MOBIE. The MOBIE multi-agent system consists of personal assistant agents for the consumers and business agents for the mobile telecommunication service providers. The MOBIE system has to take care of the personalization of the agents, security, and human agent interaction modalities.
To accommodate the automated recharging process for the user the mobile phone service providers need to be able to interact with the personal assistant agents in a reliable and secure manner. Because of the expected high frequency of such interactions the service providers need to automate these customer interactions. The option chosen in this project is to introduce business agents that are capable of the required interactions with the personal agents of the users. The personal assistant agent that represents the customer is capable of the following main tasks.

1. The personal agent creates and maintains a profile of the customer. The profile contains at least:
   a) The criteria that tell the agent when to recharge the account.
   b) The information needed to execute recharging, like the amounts it can use, and payment information.
2. The personal agent matches the criteria against the actual balance of the prepaid account.
3. The personal agent requests the necessary information from the business such as:

a) The balance of the prepaid account.
   b) The actual usage pattern of the phone for a specified period of time.

4. The personal agent is capable of recharging the prepaid account.
5. The personal agent can ask the telecom companies (through the business agents that represent them) to recharge the prepaid account with amount x.
6. The personal agent is responsible for keeping the customer informed in accordance
   to the customer profile.
7. The personal agent is able to interact with the customer through different channels:
   a) web-based,

b) WAP (for those customers that have a WAP enabled mobile phone)
   c) voice.

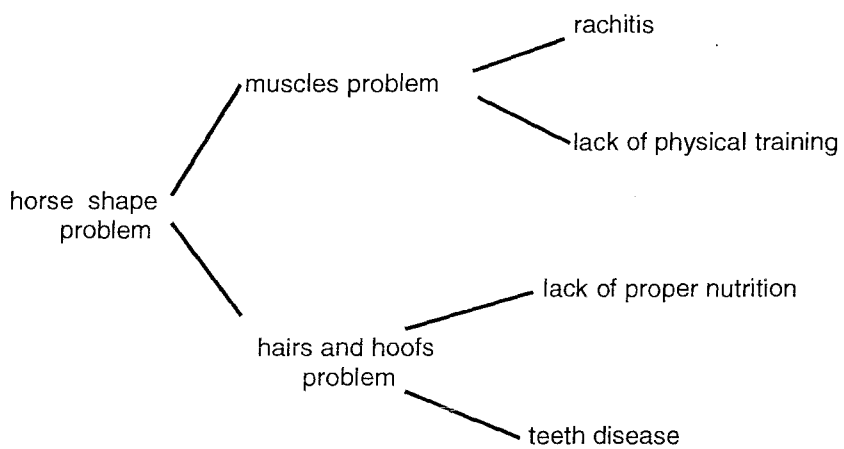
The personal assistant agents function within MOBIE in an environment consisting of business agents that represent the different telecom companies, financial institutions (like banks, with whom the actual payment is to be arrange d), and human customers. The personal assistant agents do not contact the financial institutions themselves. They can ask telecom company to recharge the prepaid account, the telecom company will then contact the appropriate financial institution.

Student name: _____
Student number: _____

Appendix 2: Horse Shape Problem

(Een Nederlandse vertaling van Horse Shape Problem is niet beschikbaar.)

Peter is an owner of a horse farm. He is always concerned about a shape of his horses, their height, shape of muscles, condition of their hoofs and hairs. If he observes some deviations from the standard, he tries to determine the nature of the problem using a line of reasoning modelled by the generic model for reasoning with and about assumptions (see Appendix 3). That model proceeds along the following lines: making assumptions (in some kind of order), predicting observation results for that assumption, and then evaluating the assumption by making the appropriate observations and comparing them to the assumption. If necessary, the old assumptions are rejected, and new ones are made. To efficiently order the assumptions he can make, Peter uses the following hierarchy (taxonomy) of the subproblems of horse shape problems:



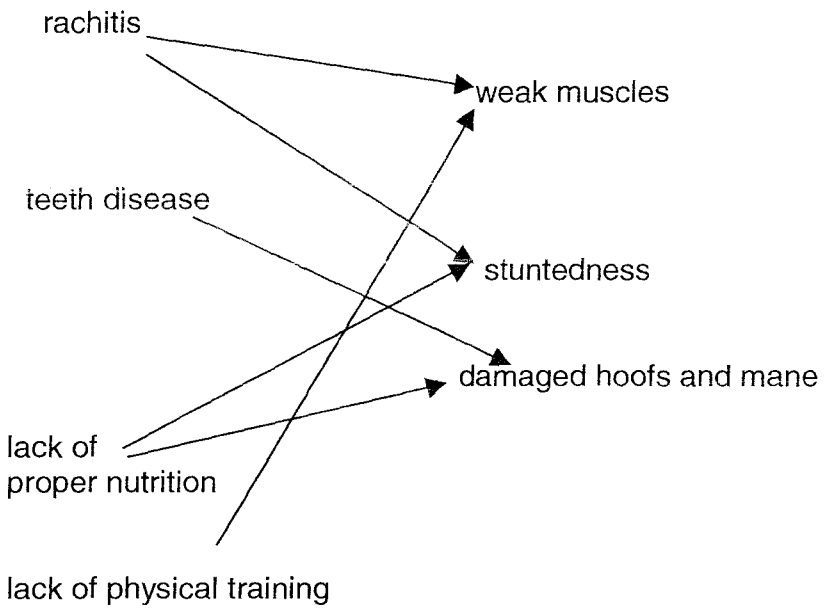
Peter can make the following observations:

- Is a horse stunted (niet tot volle ontwikkeling gekomen) or normal?
- Are its muscles weak or strong?
- Are its hoofs and mane (hoeven en manen) damaged or not?

The relations between causes and symptoms are depicted below:

Student name: _____

Student number: _____



If a horse has rachitis then it becomes small and stunted and its muscles are weak. A teeth disease can cause damage of its hoofs and mane. A stunted shape can be also a result of improper nutrition, lack of vitamins and minerals. If a horse has not enough physical training, its muscles become weak.

Student name: _____
Student number: _____

Appendix 3: Reasoning with and about assumptions

information types

information type **truth_indication**
  sorts SIGN
  objects pos, neg: SIGN
end information type

information type **obs_to_be_performed**
  sorts INFO_ELEMENT
  relations to_be_observed: INFO_ELEMENT ;
end information type

information type **observation_results**
  sorts INFO_ELEMENT,
  SIGN
  relations observation_result: INFO_ELEMENT * SIGN ;
end information type

information type **assumptions_hypotheses_and_such**
  sorts INFO_ELEMENT, SIGN
  relations assumed: INFO_ELEMENT * SIGN ;
  rejected: INFO_ELEMENT * SIGN ;
  has_been_considered: INFO_ELEMENT * SIGN ;
  possible_assumption: INFO_ELEMENT * SIGN ;
  predicted: INFO_ELEMENT * SIGN ;
  predicted_for: INFO_ELEMENT * SIGN * INFO_ELEMENT * SIGN ;
  known_to_hold: INFO_ELEMENT * SIGN ;
end information type

information type **causes**
... ...
end information type

information type **symptoms**
... ...

Student name: _____
Student number: _____

end information type

information type **world_info**
  information types symptoms, causes;
end information type

information type **world_meta_info**
 sorts WORLD_INFO_ELEMENT,
  INFO_ELEMENT
meta-descriptions
  world_info : WORLD_INFO_ELEMENT;
sub sorts WORLD_INFO_ELEMENT: INFO_ELEMENT;
end information type

information type **domain_meta_info**
  information types world_meta_info;
end information type

information type **observation_info**
  information types obs_to_be_performed, domain_meta_info;
end information type

information type **observation_result_info**
  information types observation_results, domain_meta_info, truth_indication;
end information type

information type **assumption_info**
  information types domain_meta_info, truth_indication, assumptions_hypotheses_and_such;
end information type

component assumption_determination

input information types assumption_info, observation_result_info;
output information type assumption_info;

initial kernel information level_2
  assumption(has_been_considered(HYP: INFO_ELEMENT, S: SIGN), neg);

knowledge base assumption_determination_local_kbs

Student name: _____
Student number: _____

information types assumption_info, observation_result_info;

contents

/* use as many rules as you like, you may also create additional information types if you like. */

... ...

end knowledge base

component assumption_evaluation

input information types observation_result_info, assumption_info;

output information type observation_info, assumption_info;

knowledge base assumption_evaluation_local_kbs

information types observation_result_info, assumption_info, observation_info;

contents

if predicted_for(OBS: INFO_ELEMENT, S1: SIGN, HYP: INFO_ELEMENT, S2: SIGN)

then to_be_observed(OBS: INFO_ELEMENT);

if assumed(HYP: INFO_ELEMENT, S: SIGN)

and predicted_for(OBS: INFO_ELEMENT, pos, HYP: INFO_ELEMENT, S: SIGN)

and observation_result(OBS: INFO_ELEMENT, neg)

then rejected(HYP: INFO_ELEMENT, S: SIGN)

and has_been_considered(HYP: INFO_ELEMENT, S: SIGN);

if assumed(HYP: INFO_ELEMENT, S: SIGN)

and predicted_for(OBS: INFO_ELEMENT, neg, HYP: INFO_ELEMENT, S: SIGN)

and observation_result(OBS: INFO_ELEMENT, pos)

then rejected(HYP: INFO_ELEMENT, S: SIGN)

and has_been_considered(HYP: INFO_ELEMENT, S: SIGN);

end knowledge base

component observation_result_prediction

input information types assumption_info;

output information type assumption_info;

Student name: _____
Student number: _____

knowledge base observation_result_prediction_local_kbs
information types assumption_info;

contents
/* use as many rules as you like */
*** *** ***
end knowledge base

information links

private link **hypotheses** : object - object
domain assumption_determination
information type assumption_info;
co-domain assumption_evaluation
information type assumption_info;

sort links identity
object links identity
term links identity
atom links
(possible_assumption(HYP: INFO_ELEMENT, S: SIGN),
assumed(HYP: INFO_ELEMENT, S: SIGN)): <<true,true>, <false,false>,<unknown, unknown>>;
end link

private link **assessments** : object - object
domain assumption_evaluation
information type assumption_info;
co-domain assumption_determination
information type assumption_info;

sort links identity
object links identity
term links identity
atom links
(rejected(HYP: INFO_ELEMENT, S: SIGN),
rejected(HYP: INFO_ELEMENT, S: SIGN)): <<true, true>, <false, false>>;

(has_been_considered(HYP: INFO_ELEMENT, S: SIGN),
has_been_considered(HYP: INFO_ELEMENT, S: SIGN)): <<true, true>, <false, false>>;

```
end link
private link required_observations : object - target
domain assumption_evaluation
        information type observation_info;
co-domain external_world
        information type target_observation_result_info; /* standard type */

sort links (WORLD_INFO_ELEMENT, OA)
object links identity
term links identity
atom links
        (to_be_observed(OBS: WORLD_INFO_ELEMENT),
                target(observations, OBS: OA, determine)) :
                                <<true, true>, <unknown, false>, <false,false>>;
end link


private link observation_results : epistemic - object
domain external_world
        information type epistemic_world_info;        /* standard meta-level */
co-domain assumption_evaluation
        information type observation_result_info; /* object level */

sort links (OA, WORLD_INFO_ELEMENT)
object links identity
term links identity
atom links
        (true(OBS: OA), observation_result(OBS: WORLD_INFO_ELEMENT, pos)) :
                        <<true, true>,<false,false>>;
        (false(OBS: OA), observation_result(OBS: WORLD_INFO_ELEMENT, neg)) :
                        <<true, true>,<false,false>>;
end link


private link assumptions : object - object
domain assumption_determination
        information type assumption_info;
co-domain observation_result_prediction
         information type assumption_info;
```

Student name: _____
Student number: _____

```
sort links identity
object links identity
term links identity
atom links
        (possible_assumption(HYP: INFO_ELEMENT, S: SIGN),
                assumed(HYP: INFO_ELEMENT, S: SIGN)) :
                        <<true, true>, <unknown, false>, <false,false>>;
end link


private link predictions : object - object
domain observation_result_prediction
        information type assumption_info;
co-domain assumption_evaluation
        information type assumption_info;

sort links identity
object links identity
term links identity
atom links
        (predicted_for(OBS: INFO_ELEMENT, S1: SIGN, HYP: INFO_ELEMENT, S2: SIGN),
                predicted_for(OBS: INFO_ELEMENT, S1: SIGN, HYP: INFO_ELEMENT, S2: SIGN)) :
                        <<true, true>, <unknown, unknown>, <false,false>>;
end link
```

Student name: _____

Student number: _____

Appendix 4 Answersheet (1 out of 1)

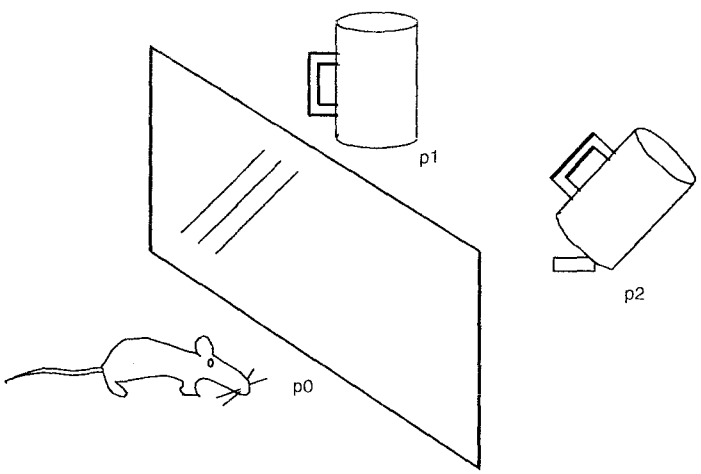
Input (1)	[assumed(a, pos), predicted_for(b, pos, a, pos)]
Output after revision but before reasoning	
Output after reasoning	
Input (2)	[assumed(a, pos), predicted_for(b, pos, a, pos), observation_result(b, neg)]
Output after revision but before reasoning	
Output after reasoning	
Input (3)	[assumed(a, neg), predicted_for(b, neg, a, neg), observation_result(b, neg)]
Output after revision but before reasoning	
Output after reasoning	

Student name: _____
Student number: _____

Appendix 5 Mouse A

5.1 Problem Description

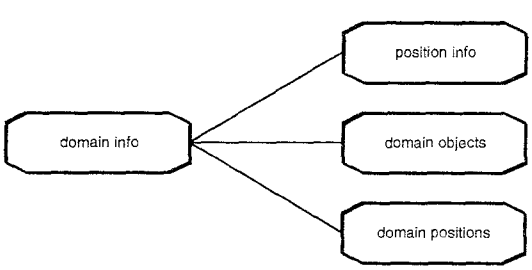
Separated by a transparent screen (a window, at position p0), at each of two positions p1 and p2 a cup (upside down) and/or a piece of food can be placed. At some moment (with variable delay) the screen is raised, and the mouse is free to go to any position. A genuine mouse is known to go to food and eat it.



In sections 3.2 and 3.3 of this Appendix a partial specification can be found of the artificial mouse.

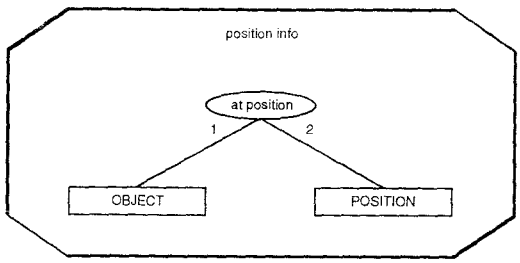
5.2 Informatie typen

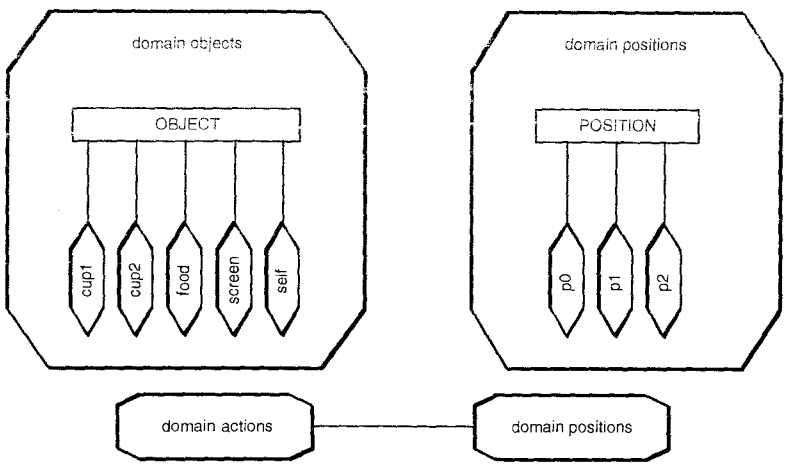
De informatie typen die in het hele systeem gebruikt worden zijn:

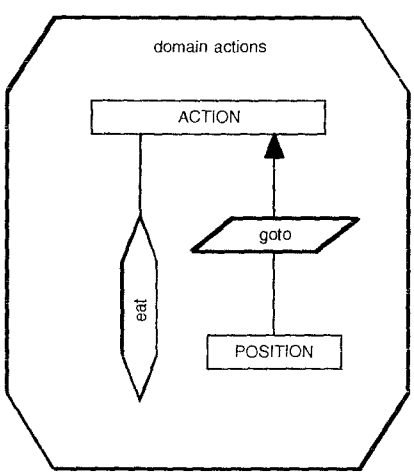


Student name: _____

Student number: _____







information type truth_indication

sorts SIGN ;

objects pos, neg : SIGN;

end information type

```
information type observation_results
        sorts                                   INFO_ELEMENT, SIGN ;
        relations       observation_result:     INFO_ELEMENT * SIGN;
end information type

information type domain_meta_info
        sorts                                   INFO_ELEMENT ;
        meta-descriptions domain_info :  INFO_ELEMENT;
end information type

information type observation_result_info
        information types       truth_indication,
                                observation_results,
                                domain_meta_info;
end information type

information type actions_to_be_performed
        sorts                                   ACTION ;
        relations       to_be_performed:        ACTION;
end information type

information type action_info
        information types       actions_to_be_performed,
                                domain_actions;
end information type
```

5.3 Fragmenten van specificatie van de component

De component is primitief en wordt hier kort beschreven.

De component mouse_a

De interfaces worden gedefiniëerd door:
input interface: de informatietypen observation_result_info;
output interface: het informatietype action_info;

De targets bij task control focus determine_action zijn:
target(determine_action, to_be_performed(X: ACTION), confirm);

De initial extent is: all-p

De kennisbank is:

```
if      observation_result(at_position(food, P:POSITION), pos)
  and   observation_result(at_position(screen, p0), neg)
```

Student name: _____
Student number: _____

and observation_result(at_position(self, P:POSITION), neg)
then to_be_performed(goto(P:POSITION))

if observation_result(at_position(self, P:POSITION), pos)
and observation_result(at_position(food, P:POSITION), pos)
then to_be_performed(eat)