Albert-Ludwigs-Universität Freiburg
Institut für Informatik

Prof. Dr. L. De Raedt
Prof. Dr. W. Burgard

Foundations of Artificial Intelligence
Summer Semester 2006

Time: September 27th 2006, 10:00-12:00

Full name: ______________________

Matriculation no.: ______________________

Degree course: ______________________

- This is a closed-book exam. Using books, lecture notes, calculators etc. is **forbidden.**
- Cheating and attempted cheating will lead to disqualification from the exam.
- Write your answers directly underneath the exercises. Continue on the following blank page if you need additional space.
- Write your name and matriculation number on each sheet.

exercise	score	max. score
1		9
2		11
3.1		2
3.2		3
3.3		3
3.4		4
4		11
5.1		6
5.2		5
6.1		6
6.2		5
7.1		4
7.2		3
7.3		3
7.4		3
8.1		6
8.2		5
9.1		4
9.2		3
9.3		4
bonus		+14
total		100

Name: Matr. no.:

Exercise 1 (9 marks)

Prove the *contradiction theorem*:

$$\Theta \cup \{\varphi\} \text{ is unsatisfiable if and only if } \Theta \models \neg\varphi$$

Alternative formulation:

$$\Theta \cup \{\varphi\} \models \bot \text{ if and only if } \Theta \models \neg\varphi$$

You may assume that the *deduction theorem* and *contraposition theorem* are true.

Name: Matr. no.:

Exercise 2 (11 marks) REASONING IN PREDICATE LOGIC

Consider the knowledge base

$$\Theta = \{\forall x(\neg(\exists z(A(z) \wedge B(x))) \vee C(x)),$$
$$\forall y(\neg B(y) \Rightarrow C(y))\}.$$

Prove that $\Theta \models (\forall x A(x)) \Rightarrow (\exists y C(y))$.

Name: Matr. no.:

Exercise 3 (2+3+3+4 marks) BAYESIAN NETWORKS

Consider the Bayesian network which is given by the following conditional probability tables:

$\mathbf{P}(G)$
0.90

$\mathbf{P}(S)$
0.20

$\mathbf{P}(M \mid G, S)$	G	S
0.90	true	true
0.10	true	false
0.30	false	true
0.60	false	false

$\mathbf{P}(T \mid M)$	M
0.37	true
0.61	false

1. Draw the graph representation of the Bayesian network.
2. Indicate the *Markov blanket* for nodes M and S, respectively.
3. The information represented by the given Bayesian network could also be represented by a joint probability table. **Briefly (at most 3 lines!)** explain the main advantages and disadvantages of each approach.
4. Compute $P(M)$.

Name: Matr. no.:

Exercise 4 (11 marks) PARSING

Consider the following grammar:

1	S	$\rightarrow$	$NP\ VP$
2	NP	$\rightarrow$	$D\ N$
3	VP	$\rightarrow$	V
4	VP	$\rightarrow$	$V\ NP$
5	D	$\rightarrow$	`the`
6	N	$\rightarrow$	`professor`
7	N	$\rightarrow$	`student`
8	V	$\rightarrow$	`examines`

Parse the following sentence using a *bottom up parser*:

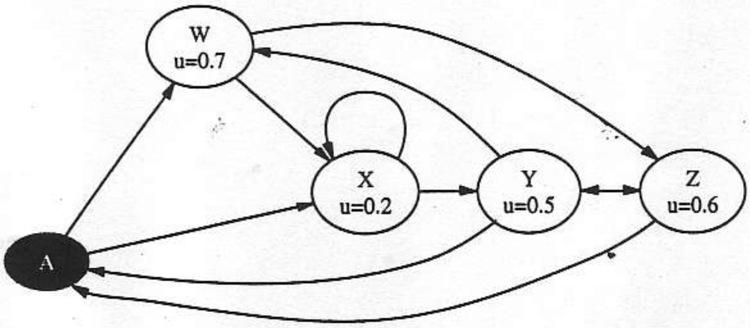
`the professor examines the student`

Give the trace of the algorithm using the notation introduced in the lecture.

Name: Matr. no.:

Exercise 5 (6+5 marks) DECISION THEORY

Consider the graph depicted below. An agent moves along the graph by following the arcs. Every movement action succeeds with a (small) probability of 0.2; the remaining probability is distributed among the other successors of the current vertex (the agent *must* move by following an arc). The values depicted for vertices W, X, Y and Z denote the utility of the respective state.



1. Calculate the utility of state A. The immediate reward for this state is 0.2, and the discount factor is $\gamma = 0.25$. Write down all steps of your calculation.

2. Which is the best action for the agent in state Y? Write down all steps of your calculation and justify your answer.

Name: Matr. no.:

Exercise 6 (6+5 marks) BOARD GAMES

The *Nim* game is played as follows: Two players take turns removing one, two, or three coins from a stack of coins. The player who removes the last coin loses the game.

1. Draw the game tree for an initial state with 4 coins, and analyze it with the *Minimax* algorithm. Show an optimal strategy for both players by marking an optimal move in each of the inner nodes of the game tree.

2. Which of the players can force a win in the case of 5 or 6 coins in the initial state, respectively? Use the result from the first part of the exercise. Do not draw a complete game tree.

Name: Matr. no.:

Exercise 7 (4+3+3+3 marks) CONSTRAINT SATISFACTION PROBLEMS

Consider the following instance of the 2×2 *Sudoku puzzle*:

1			
	2		
			1
4			3

The objective is to fill those cells which are initially empty with the numbers 1, 2, 3 and 4 in such a way that each of these numbers appears exactly once in each row, in each column, and in each of the four 2×2 subgrids. *Well-formed* Sudoku puzzles do not require "guesses".

1. Specify the general 2×2 Sudoku puzzle as a CSP.
2. Describe (briefly!) a backtracking algorithm for solving the problem. Name two heuristics that can potentially improve the performance of the algorithm.
3. Describe (briefly!) a hill climbing algorithm for solving the problem.
4. Which of the two algorithms from parts 2. and 3. would you prefer to use in an efficient solver for well-formed Sudoku puzzles? Justify your answer briefly.

Exercise 8 (6+5 marks) PLANNING

The (simplified) *Towers of Hanoi* problem consists of three pegs and three discs of different size. Initially, all discs are stacked on peg 1, ordered by size (with the largest one at the bottom). The objective is to move all discs to peg 3, ordered in the same way as they are initially. Only one disc may be moved at a time.

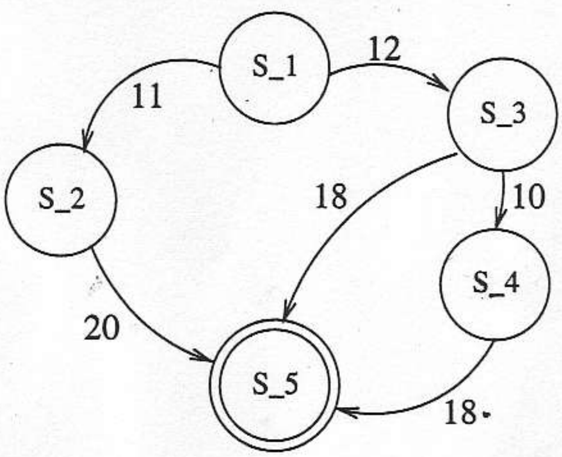
The following rules apply:

- Only the topmost disc of each peg may be moved.
- Discs may only be moved on top of larger discs or onto empty pegs.

1. Express the problem as a planning problem in the STRIPS formalism (initial state, actions, goals). Use the three predicates *clear*, *smaller* and *on*, three objects for the three discs and three objects for the three pegs. A single action *move* is sufficient.
2. State a solution for the problem.

Exercise 9 (4+3+4 marks) INFORMED SEARCH

Consider the following state space with initial state s_1 and goal state s_5:



Moreover, consider the following heuristic function:

$$h(s_1) = 17, \quad h(s_2) = 17, \quad h(s_3) = 24, \quad h(s_4) = 17, \quad h(s_5) = 0$$

1. Why might the A^* algorithm, using this heuristic function, compute a suboptimal solution?
2. Modify the heuristic function in such a way that A^* is guaranteed to compute an optimal solution.
3. Apply the A^* algorithm to the problem with the modified heuristic function. Record all search nodes which are generated during the search along with their respective path costs and heuristic estimates. Also state in which order the nodes are expanded.