Last Name: First Name:

Matriculation Number:

Seat:

Exam
Künstliche Intelligenz 1

Feb 12., 2018

	To be used for grading, do not write here				
prob.	1	2	Sum	grade	
total	12	10	80		
reached					

Organizational Information

Please read the following directions carefully and acknowledge them with your signature.

1. Please place your student ID card and a photo ID on the table for checking
2. The point distributions for the problems are provisional.
3. You can reach 80 points if you fully solve all problems. You will only need 55 points for a perfect score, i.e. 25 points are bonus points.
4. No resources or tools are allowed except for a pen.
5. You have 90 min (sharp) for the test
6. Write the solutions directly on the sheets, no other paper will be graded.
7. If you have to abort the exam for health reasons, your inability to sit the exam must be certified by an examination at the University Hospital. Please notify the exam proctors and have them give you the respective form.
8. Please make sure that your copy of the exam is complete (16 pages excluding cover sheet and organizational information pages) and has a clear print. **Do not forget to add your personal information on the cover sheet and to sign this declaration.**

Declaration: With my signature I certify having received the full exam document and having read the organizational information above.

Erlangen, Feb 12., 2018 ....

(signature)

Organisatorisches

Bitte lesen die folgenden Anweisungen genau und bestätigen Sie diese mit Ihrer Unterschrift.

1. Bitte legen Sie Ihren Studierendenausweis und einen Lichtbildausweis zur Personenkontrolle bereit!
2. Die angegebene Punkteverteilung gilt unter Vorbehalt.
3. Sie können 80 Punkte erreichen, wenn Sie alle Aufgaben vollständig lösen. Allerdings zählen 55 Punkte bereits als volle Punktzahl, d.h. 25 Punkte sind Bonuspunkte.
4. Es sind keine Hilfsmittel erlaubt außer eines Stifts.
5. Die Bearbeitungszeit beträgt genau 90 min.
6. Schreiben Sie die Lösungen direkt auf die ausgeteilten Aufgabenblätter. Andere Blätter werden nicht bewertet.
7. Wenn Sie die Prüfung aus gesundheitlichen Gründen abbrechen müssen, so muss Ihre Prüfungsunfähigkeit durch eine Untersuchung in der Universitätsklinik nachgewiesen werden. Melden Sie sich in jedem Fall bei der Aufsicht und lassen Sie sich das entsprechende Formular aushändigen.
8. Überprüfen Sie Ihr Exemplar der Klausur auf Vollständigkeit (16 Seiten exklusive Deckblatt und Hinweise) und einwandfreies Druckbild! **Vergessen Sie nicht, auf dem Deckblatt die Angaben zur Person einzutragen und diese Erklärung zu unterschreiben!**

Erklärung: Durch meine Unterschrift bestätige ich den Empfang der vollständigen Klausurunterlagen und die Kenntnisnahme der obigen Informationen.

Erlangen, Feb 12., 2018 ....

(Unterschrift)

Please consider the following guidelines to avoid losing points:

- If you continue an answer on another page, clearly give the problem number on the new page and a page reference on the old page.
- If you do not want something to be graded, clearly cross it out. Adding a wrong statement to a correct solution may lead to deductions.
- The instructions "Give X", "List X" or similar mean that only X is needed. If you additionally justify your answer, we will try to give you partial credit for a wrong answer (but there is no guarantee that we will).
- The instruction "Assume X" means that X is information that you may use in your answer.
- The instruction "Model X as a Y" means that you have to describe X formally and exactly as an instance of Y using the definition of Y from the lecture.
- If you are uncertain how long or complex an answer should be, use the number of points as an indication: 1 point roughly corresponds to 1 minute.

1 Prolog

Problem 1 (The Zip Function) 4 pt

The zip function takes two lists with lengths that differ at most by 1, and outputs a list of lists containing one element from the first list and the element with the same index from the other list, possibly followed by a one-element list with the left-over argument.

Create a `ProLog` predicate with 3 arguments: the first two would be the two lists you want to zip, and the third one would be the result. For instance:

```
?- zip([1,2,3],[4,5,6],L).
L = [[1, 4], [2, 5], [3, 6
```

```
?- zip([1,2],[3,4,5],L).
L = [[1, 3], [2, 4], [5]].
```

Feel free to implement any helper functions.

Hint: Remember that you can pattern match a list L as [HEAD|TAIL].

Problem 2 (DFS in Prolog) 12 pt

We want to implement DFS in `ProLog` using the following data structures for search trees:

```
subtrees([]).
subtrees([(Cost,T)|Rest]) :- number(Cost),istree(T), subtrees(Rest).
istree(tree(_,Children)) :- subtrees(Children).
```

Write a Prolog predicate dfs such that `dfs(G,T,X,Y)` on a tree `T` returns the path to the goal `G` in `X` and the cost of the path in `Y`

2 Search

Problem 1 (Astar vs. Greedy) 5 pt

Shortly explain the principle of operation of the A* search. Explain (in few sentences) how it differs from the greedy search?

What is the condition on the heuristic function that makes A^* optimal? Does a heuristic with this condition always exist?

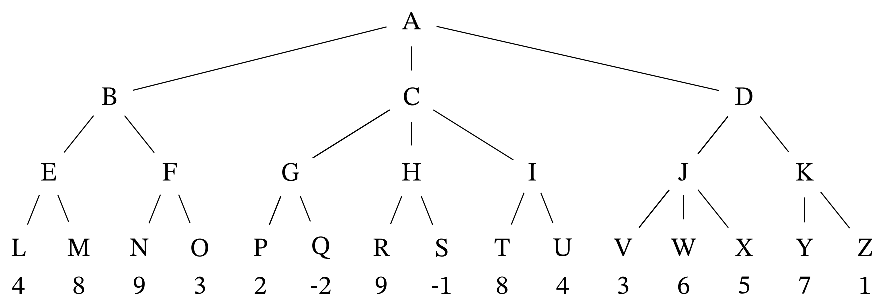
3 Adversarial Search

Problem 1 (Minimax Restrictions) 3 pt

Name at least five criteria that a game has to satisfy in order for the *minimax algorithm* to be applicable.

Problem 2 (Minimax Algorithm and Alphabeta Pruning) 10 pt

Consider the following (complete!) search tree:



1. What is the minimax value at node B?
2. Which subtrees in the tree can be pruned during alpha-beta search? What is the criterion for pruning a subtree?

4 Constraint Satisfaction Problems & Inference

Problem 1 (Constraint Networks) 4 pt

A *constraint network* is a triple $\langle V, D, C \rangle$. Explain the roles of V, D, and C. If you use the word "constraint" you have to define and briefly explain it.

Problem 2 (Constraint Networks) 10 pt

Describe the following problems as constraint networks

1. Sudoku
2. The 8 queens problem

This page was intentionally left blank for extra space

5 Logic

Problem 1 (Propositional Resolution) 5 pt

Prove the following formula using the resolution calculus:

$$\neg(\neg P \Rightarrow Q) \vee (Q \vee R) \vee \neg(P \Rightarrow R)$$

To convert the formula to CNF, you can use the extended rules for the transformation calculus from the lectures, for connectives $\vee$, $\neg$, $\Rightarrow$, and $\wedge$. It is ok to do a few steps at a time, but you should annotate each row with the rules you used. Name the rules after the connective and label at the assumption. For example, the rule

$$\frac{\mathbf{C} \vee (\mathbf{A} \vee \mathbf{B})^{\mathsf{T}}}{\mathbf{C} \vee \mathbf{A}^{\mathsf{T}} \vee \mathbf{B}^{\mathsf{T}}}$$

would be named $\vee^{\mathsf{T}}$.

Problem 2 (First-Order Tableau) 10 pt

Prove the following formula using the first-order free variable tableaux calculus. We have $P \in \Sigma_2^p$, $f, g \in \Sigma_1^f$ and $a, b \in \Sigma_0^f$.

$$\exists X\Big(\big(\exists Y \neg P(Y, f(b))\big) \vee \big(\neg P(b, f(X)) \Rightarrow \neg P(g(a), f(X))\big)\Big)$$

6 Planning

Problem 1 12 pt

Cheeta is an intelligent and lazy monkey. Your task is to help him grab a banana.

Initially, the monkey and a boat are “Low” on the ground at position “A” and the banana hangs “High” at position “B” on a tree. Cheeta can climb up the tree where the banana hangs, meaning that it can climb from height “Low” to “High” at B. Cheeta can grab the banana when they are both at the same position and at the same height.

There is a river between “A” and “B”, and Cheeta can only travel between these positions by boat. Cheeta can only enter the boat if they are at the same position and height (the boat is a height “Low”).

The following STRIPS model is used. The facts are:

- $At(x, y)$: $x \in \{Boat, Banana, Cheeta\}$ is at position $y \in \{A, B, Boat\}$.
- $Height(x, y)$: $x \in \{Boat, Banana, Cheeta\}$ is at height $y \in \{Low, High\}$.
- $Climbable(x)$: Cheeta can climb at position $x \in \{A, B\}$.
- $Grabbed()$: Cheeta has grabbed the banana.

The goal for Cheeta is to grab the banana using the following available actions:

- $Drive(x, y)$ to get from x to y
- $GetIn(x)$ to get in the boat (at location x)
- $GetOut(x)$ to get out of the boat (at location x)
- $Climb(x, y, z)$ to climb at position x from height y to height z
- $Grab(x, y)$ to grab the banana (at position x and height y)

(a) Properly define the actions $Drive(x, y)$ and $GetIn(x)$

(b) Give the initial state and a solution to this planning problem

This page was intentionally left blank for extra space

This page was intentionally left blank for extra space

This page was intentionally left blank for extra space

This page was intentionally left blank for extra space