EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

MATH. - NATURWISS. FAKULTÄT
FACHBEREICH INFORMATIK
KOGNITIVE SYSTEME · PROF. A. ZELL

Artificial Intelligence

Assignment 1

Assignment due by: 02.11.2016, Discussions on: 08.11.2016

Important: If you have not already done so, please register via ILIAS until 01.11.2016 at the latest.

Notes: Please always justify your answers, even for yes/no questions, unless the description explicitly states otherwise. Full grades can only be awarded when answers are justified. Programs which fail to compile or run automatically yield 0 points. Please hand in all programs and functions as text files (with the appropriate file extension).

Question 1 Environments for agents 1 (5 points)

Explain with your own words the following task environment properties and provide at least one example for each of them.

- Fully observable – partially observable
- Deterministic – stochastic
- Episodic – sequential
- Static – dynamic
- Discrete – continuous

Question 2 Environments for agents 2 (4 points)

Give one example for each of the following environments.

- Multi agent – discrete – fully observable – static
- Single agent – deterministic – continuous – fully observable
- Multi agent – partially observable – sequential – continuous
- Single agent – partially observable – static – discrete

Question 3 Programming in LISP (2+2+2+2+3=11 points)

In preparation for this assignment, you need to install an interpreter for Common-LISP, e.g. GNU CLIPS (`http://www.clisp.org`) and get familiar with it. When doing these exercises you are **not** allowed to use built in functions like `sort` that solve the exercise by themselves. In addition, you are **not** allowed to use looping constructs as these exercises are meant to give familiarity with a functional programming approach.

(a) Implement a function $(\text{factorial } n)$, with $n \in \mathbb{N}$, returning $n!$.

(b) Implement a function $(\text{listSum } n\ s)$. n and s are both integers. The function listSum should return a list with two elements: (i) a list composed of consecutive numbers from 1 to n where n is the size of the list and (ii) the added values of the odd numbers in the list if s is set 0 or the added values of the even numbers in the list if s is set to 1.

(c) Given two lists $v_1 \in \mathbb{R}^n$ and $v_2 \in \mathbb{R}^m$, write a function $(\text{scalarProduct } v_1\ v_2)$ which returns the usual scalar product when it is defined or NIL when it is not defined.

(d) Implement a function $(\text{insert } a\ z\ p)$, where a is a list composed of n real valued elements, z is a real valued element, $p \leq n$ is an integer, such that the element z is inserted at the position p in the list. The function should return the resulting list now containing $n + 1$ elements.

(e) Implement a function $(\text{mySort } a\ s)$, where a is a list composed of n real valued elements and s is an integer, returning the sorted list a in ascending order if s is set to 0 or descending order if s is set to 1.