Project

Weight: 25% of your final grade

Due: after Unit 8

This project has two versions: a research-oriented version and an application-oriented version. You can choose to complete either version, but are required to inform your tutor about your choice.

The Two Versions of the Project

1. **Research-oriented**: Students who want to explore some advanced concepts of AI get the opportunity to do so. In this version of the project, you select a topic from the ones listed in the project outline or come up with an AI topic that interests you and write an essay about it.

   The essay consists of a literature review, a critical analysis, and conclusions or recommendations elaborated by you. In the literature review, you should include pertinent recent papers from libraries such as the IEEE database (http://0-ieeexplore.ieee.org.aupac.lib.athabascau.ca/Xplore/dynhome.jsp), ACM database (http://0-www.acm.org.aupac.lib.athabascau.ca/dl/), CiteSeerx (http://citeseerx.ist.psu.edu/index;jsessionid=DD705B2DF220521BCC2DC2403874DE2A), and other resources listed in the course study guide.

2. **Application-oriented**: You will implement an expert system in either Prolog or Lisp for solving the farmer problem from Assignment 3. The implementation consists of developing a system shell and a knowledge base for the farmer problem. To implement the shell, you can use either EXSHELL for Prolog or Lisp-shell for Lisp.

The Research-oriented Project

In this version of the project, you are asked to explore and/or investigate one of the AI topics we didn't cover in detail in the course. You may either explore one of the topics listed here or another topic that interests you.

☞ If you select to investigate a topic that is not listed here, please contact your tutor to get his/her approval.

Your exploration or research should focus on an in-depth literature review about theories, techniques, algorithms, approaches, mechanisms, and implementation of the selected topic. The list of proposed topics is as follows:

- Uncertainty models for knowledge representation
- Reasoning in uncertainty

- Case-based reasoning
- AI planning
- Agent-based systems
- Agent-oriented programming
- Unsupervised learning
- Competitive learning
- Genetic and emergent models of learning
- Natural language processing
- AI for game development

Your topic could come from a sub-part of the listed topics or any other advanced AI research topic. You can also report on a successful (or a planned) implementation in one of the above topics (if you want to explore an existing AI software or tool package).

Your investigation will be based on recent papers, technical documents, and software packages (open source preferred). You are encouraged to read some papers from the following sources for new research developments in AI:

- IEEE Xplore digital library: http://0-ieeexplore.ieee.org.aupac.lib.athabascau.ca/Xplore/dynhome.jsp
- ACM digital library: http://0-www.acm.org.aupac.lib.athabascau.ca/dl/
- AAAI (The Association for the Advancement of Artificial Intelligence): http://www.aaai.org/
- JAIR (*Journal of Artificial Intelligence Research*): http://www.jair.org/
- AI Depot – News, knowledge, and discussion for the AI enthusiast: http://ai-depot.com/Main.html
- Artificial Intelligence Foundation: http://www.alicebot.org/

To ensure that your topic has reasonable depth and coverage, you are encouraged to write a one-page outline about it, and email it to your tutor in advance. Your tutor may either agree with it or provide some comments and suggestions for you to make further selection or modification. If you are having any difficulties in selecting a topic, discuss some possibilities with your tutor. Note that topic selection is part of your work in this assignment, and you should not simply rely on your tutor for help.

Once your topic is decided, you need to set up feasible yet meaningful outcomes for your exploration or research on the topic. Overall, the outcomes should be interesting and beneficial to the professional community of AI research and development. Some possible outcomes are

- your analysis, findings, and discovery of problems.
- your proposed new or improved methods, algorithms, etc.
- your insight on future directions.
- results of your test, survey, and comparison.
- your useful implementation.

You are asked to write a report of no more than 10 pages (2500–3000 words), which should include the following:

- Title
- Abstract: no more than 300 words.
- Introduction/background: motivation and overall introduction, including necessary literature review and citations.
- Methods: describe the focus of your research topic and the methods you used to explore it.
- Results and findings: give research results, or explore your findings. Include experimental or theoretical analysis accomplished based on your methods.
- Related work: summarize the related work; compare your methods and results with others' work. **Citations are required**.
- Conclusion and future work: conclude your exploration and research, and point out possible future work.
- References: **include all cited references**.

Important!

- **Audience**: Assume you're writing this report to be published in a technical magazine; i.e., it's more like an article you might find in a publication such as *PC Magazine* and not like one you'd find in an email business "rag" or its web-based counterpart. Your report needs to be sufficiently technical that it could be used by IT professionals to decide whether the subject is worth further research.

- **Plagiarism and intellectual honestly:** A very important issue in reusing the ideas and information of others is to give credit to the original author in an accepted way; i.e., to learn how to cite information and how to quote, paraphrase, and summarize information. Not following these established rules is commonly known as *plagiarism* and is a serious academic offense. If you haven't written many formal papers before and do not know how to identify plagiarism and how to avoid it, you should read AU's statement on intellectual honesty and, especially, plagiarism (http://calendar.athabascau.ca/undergrad/current/page11.php#acad_misconduct). There is additional helpful information at the AU Write Site.

- **APA style.** The style of the documents, including all citations and references, must follow APA style (http://www.apastyle.org/). You should not need to read all about APA style to use it: as with much other pattern recognition and use in this course, you should be able to pick up a good enough idea of the style by scanning a description and seeing some examples (or vice versa). You can check the resources on the AU Write Site for more information.

- Make sure you always include proper citation for viewpoints, methods, algorithms, data, results, figures, tables, etc., that you have taken from other papers or contributions whenever you use them in your report. Most of your cited passages should be paraphrases of the original. Keep direct quotations to a minimum. All the references should be published, or at least publicly and stably accessible from the Internet. Provide live links to sources (preferably permalinks) whenever possible.

Marking Scheme

Technical content: 50 marks

Synthesis and organization: 30 marks

Writing style and language: 20 marks

– No spelling errors?

– Follows the instructions and style requirements accurately?

– Plagiarism free? Please follow standard citation and quotation rules to avoid any copyright issues.

The Application-oriented Project

This version of the project is about developing an expert system for solving the farmer problem. For this project, you should write a report that explains how your expert system is implemented and how it works, and give some execution examples.

The expert system has two parts: the knowledge base and the shell that can be fed with different knowledge bases and give an answer (i.e., recommendations) based on the knowledge base fed. For implementing the shell you can either use an existing expert system shell or write your own. For Prolog you can use the EXSHELL presented in Chapter 6 of the textbook companion resource *AI Algorithms, Data Structures, and Idioms in Prolog, Lisp, and Java*. For the Lisp you can use the Lisp-shell presented in Chapter 17 of the textbook companion resource *AI Algorithms, Data Structures, and Idioms in Prolog, Lisp, and Java*. Source code for these shells can also be downloaded from internet but it's your responsibility to make them work on your compiler.

You should also create a knowledge base representing the famous problem of a farmer having a goat, a wolf, and a cabbage. The farmer wants to take them all across the river from the east shore to the west shore, but his boat is small. The boat has space for only the farmer who can row and one of the items: cabbage, wolf, or goat. The farmer cannot leave the wolf alone with the goat or the goat alone with the cabbage. Your program should be able to display for the user the list the possible steps the farmer may take to move his items across river safely and provide an explanation of how the solution was derived. If the user asks for more solutions, your system should also be able to provide them if they exist.

There are many ways to design a knowledge base, and you are free to design your knowledge base as you like, but it should be expressed in terms of facts and rules. For example, you can design the problem in terms of a state space search with an initial value for the tuple (Farmer, Goat, Cabbage, Wolf) and a goal state to reach. The initial state represented in terms of fact would be state (e, e, e, e), which means that all items are on the east shore of the river. The goal state will then be state (w, w, w, w). Each rule will represent a possible move, and when the rule is fired it will change the value of the state until it reaches the goal state (w, w, w, w).

Note: all data about the state space and the initial state of the system should be defined in the knowledge base as facts.

Your system will consist mainly of two parts: the first is the expert system shell, and the second is the knowledge base—production rules—for the farmer problem. The expert system shell when executed will ask for the name of the file containing the knowledge base. When the user enters the name of this file, the shell will seek it from the disk, parse the rules to make sure they are written properly, then start dealing with them.

Your program should list for each solution the possible steps the farmer can take to move his items across the river safely. You should explain how your predicate works and give some execution examples.

You should submit a working system as well as a description of it. Explain the principles and theories you applied, and show where you have programmed them in the coding.

Marking Scheme

Code: 20 marks

Documentation: 30 marks

Functionality: 50 marks

☞ **Note**: For the sake of fairness to all students, your tutor is not supposed to provide help with your program coding (e.g., pick up compiler errors, send a working code for part of the program, correct your wrong code and convert it to a working one). S/he may, however, give advice about your steps and/or your techniques. If you need help with your coding in order to complete your project or to keep you going with it, then you may lose marks.