3/0/3

DEREE COLLEGE SYLLABUS FOR:

ITC 4480 ARTIFICIAL INTELLIGENCE PRINCIPLES – LEVEL 6

UK CREDITS:15

(Updated **January 2011)**

PREREQUISITES:

CS 1070 Introduction to Information Systems
CS 2188 Introduction to Programming
MA1001 Finite Mathematics
MA 1105 Applied Calculus

CATALOG DESCRIPTION:

Theoretical foundations of artificial intelligence. Unstructured problem solving: problem analysis, research tools. Knowledge representation. Inference rules. Search strategies. Heuristics. Expert systems. Uncertainty. Natural language understanding. Symbol-based machine learning. Neural networks. Genetic algorithms. Agents. AI application languages (Prolog, LISP).

RATIONALE:

The course is designed to introduce students to the structures and strategies used for unstructured problem solving. Emphasis is given on knowledge representation, reasoning under uncertainty, machine learning, natural language processing, and implementation using a natural languages.
The course is suitable for students who seek a career in artificial intelligence, computer science, information technology, or software engineering.

LEARNING OUTCOMES:

As a result of taking this course, the student should be able to:

1. Identify and apply knowledge representation formalisms with emphasis on propositional and predicate calculus but also with conceptual graphs, including representation of uncertainty.
2. Analyse problems as state space graphs, and apply heuristic state space searches including planning using Prolog or Lisp.
3. Evaluate a state space search algorithm in terms of admissibility, monotonicity, and informedness.
4. Analyze and evaluate expert systems.
5. Identify learning techniques: symbol based (supervised and unsupervised), reinforcement, neural networks, and genetic algorithms
6. Analyse the main approaches to natural language processing

METHOD OF TEACHING AND LEARNING:

In congruence with the learning and teaching strategy of the College, the following tools/activities are used:

- Classroom lectures, discussions, and review of theoretical concepts. Laboratory practical sessions.
- Office hours: Students are encouraged to make full use of the office hours of their instructor, where they can ask questions and go over lecture material.
- Use of the Blackboard Learning platform, where instructors post lecture notes, assignment instructions, timely announcements, as well as additional resources.

ASSESSMENT:

Summative:

Research Project (1,500-2,000 words; case study, data collection, synthesis, critical evaluation, program development)	**50**
Final Examination (2-hour comprehensive): Short answers to essay questions, problem solving	**50**

Formative:

In class, problem solving	**0**

The formative assessments aim to shape teaching along the semester and prepare students for the summative assessments.

The research paper assesses Learning Outcomes 2,3,4,5

The final examination assesses Learning Outcomes 1-6.

(Guidelines and assessment rubrics are distributed on the first day of classes along with the course outline.)

INDICATIVE READING:

REQUIRED READING:

Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, Latest Edition

RECOMMENDED READING:

George F. Luger. Artificial Intelligence: Structures and Strategies for Complex Problem Solving: Addison-Wesley, latest edition. ISBN 0-201-64866-0.

Christopher M. Bishop, "Pattern Recognition and Machine Learning", Springer, 2006

Nikos Vlassis, "A concise introduction to multiagent systems and distributed artificial intelligence", Morgan & Claypool Publisher, 2007

James Lu and Jerud J. Mead, "Prolog, A Tutorial Introduction", Computer Science Department, Bucknell University (http://www.soe.ucsc.edu/classes/cmps112/Spring03/languages/prolog/PrologIntro.pdf)

COMMUNICATION REQUIREMENTS:

Daily access to the course's site on the College's Blackboard CMS. Use of word processing, spreadsheet and/or presentation graphics software for documentation of assignments

SOFTWARE REQUIREMENTS:

Prolog, Lisp, Java

WWW RESOURCES:

American Association for Artificial Intelligence (AAAI) http://www.aaai.org

European Coordinating Committee for Artificial Intelligence (ECCAI) http://www.eccai.org

MIT Artificial Intelligence Lab http://www.ai.mit.edu

German Institute of Artificial Intelligence (DFKI) http://www.dfki.de/web/welcome?set_language=en&cl=en

Artificial Intelligence Applications Institute (AIAI) http://www.aiai.ed.ac.uk

INDICATIVE CONTENT:

1. AI: HISTORY AND APPLICATIONS
   - 1.1. Attitudes toward Intelligence, Knowledge, and Human Artifice
   - 1.2. Overview of AI Application Areas
   - 1.3. Artificial Intelligence: An Attempted Definition
2. THE PREDICATE CALCULUS
   - 2.1. The Propositional Calculus
   - 2.2. The Predicate Calculus
   - 2.3. Using Inference Rules to Produce Predicate Calculus Expressions
3. STRUCTURES AND STRATEGIES FOR STATE SPACE SEARCHES
   - 3.1. Graph Theory
     - 3.1.1. Structures for State Space Searches
     - 3.1.2. State Space Representations of Problems
   - 3.2. Strategies for State Space Searches
     - 3.2.1. Data-Driven and Goal-Driven Searches

- 3.2.2. Depth-First and Breadth-First Searches
- 3.3. Using the State Space to Represent Reasoning
  - 3.3.1. State Space Descriptions of a Logical System
  - 3.3.2. And/Or Graphs

4. HEURISTIC SEARCHES
   - 4.1. "Best-First" Searches
   - 4.2. Heuristic Searches and Expert Systems
   - 4.3. Admissibility, Monotonicity, Informedness
   - 4.4. Heuristics in Games
     - 4.4.1. The Minimax Procedure
     - 4.4.2. The Alpha-Beta Procedure
   - 4.5. Complexity Issues
5. CONTROL AND IMPLEMENTATION OF STATE SPACE SEARCHES
   - 5.1. Recursion-Based Searches
   - 5.2. Pattern-Directed Searches
   - 5.3. Production Systems
   - 5.4. The Blackboard Architecture for Problem Solving
6. KNOWLEDGE REPRESENTATION
   - 6.1. AI Representational Schemes
     - 6.1.1. Semantic Networks
     - 6.1.2. Scripts, Frames
   - 6.2. Conceptual Graphs
     - 6.2.1. Types, Individuals, and Names
     - 6.2.2. The Type Hierarchy
     - 6.2.3. Generalization and Specialization
     - 6.2.4. Propositional Nodes
     - 6.2.5. Logic
   - 6.3. Alternatives to Explicit Representation
   - 6.4. Agent-Based and Distributed Problem Solving
7. STRONG METHOD PROBLEM SOLVING
   - 7.1. Expert Systems Technology
   - 7.2. Rule-Based Expert Systems
     - 7.2.1. Goal-Driven and Data-Driven Reasoning
     - 7.2.2. Heuristics and Control
   - 7.3. Model-Based, Case-Based, and Hybrid Systems
   - 7.4. Planning
8. REASONING UNDER UNCERTAINTY
   - 8.1. Logic-Based Abductive Inferences
   - 8.2. Abduction: Alternatives to Logic
     - 8.2.1. The Stanford Certainty Factor
     - 8.2.2. Fuzzy Sets
     - 8.2.3. The Dempster-Shafer Theory of Evidence
   - 8.3. The Stochastic Approach to Uncertainty
9. LANGUAGES AND PROGRAMMING TECHNIQUES FOR ARTIFICIAL INTELLIGENCE

- 9.1. Prolog Implementation
  - 9.1.1. Syntax for Predicate Calculus Programming
  - 9.1.2. Lists and Recursions
  - 9.1.3. Search Controls
  - 9.1.4. Abstract Data Types
- 9.2. LISP Implementation (Overview)

10. UNDERSTANDING NATURAL LANGUAGE
    - 10.1. Deconstructing Language
    - 10.2. Syntax
      - 10.2.1. Specification and Parsing Using Context-Free Grammars
      - 10.2.2. Transition Network Parsers
      - 10.2.3. The Chomsky Hierarchy and Context-Sensitive Grammars
      - 10.2.4. ATN Parsers
    - 10.3. Stochastic Tools for Language Analysis (Overview)
    - 10.4. Natural Language Applications
11. MACHINE LEARNING: SYMBOL-BASED
    - 11.1. A Framework for Symbol-Based Learning
    - 11.2. Version Space Searches
    - 11.3. The ID3 Decision Tree Induction Algorithm (Overview)
    - 11.4. Inductive Bias and Learnability
    - 11.5. Knowledge and Learning
      - 11.5.1. Meta-DENDRAL
      - 11.5.2. Explanation-Based Learning
      - 11.5.3. EBL and Knowledge-Level Learning
      - 11.5.4. Analogical Learning
    - 11.6. Unsupervised Learning
      - 11.6.1. Discovery
      - 11.6.2. Conceptual Clustering
      - 11.6.3. COBWEB (Overview)
    - 11.7. Reinforcement Learning
12. MACHINE LEARNING: CONNECTIONIST
    - 12.1. Foundations for Connectionist Networks
    - 12.2. Perceptron Learning
    - 12.3. Backpropagation Learning
    - 12.4. Competitive Learning
      - 12.4.1. A Kohonen Network
      - 12.4.2. Outstar Networks and Counterprobagation
    - 12.5. Hebbian Coincidence of Learning (Overview)
    - 12.6. Attractor Networks or "Memories" (Overview)
13. MACHINE LEARNING: SOCIAL AND EMERGENT
    - 13.1. The Genetic Algorithm
    - 13.2. Classifier Systems and Genetic Programming
    - 13.3. Artificial Life and Society-Based Learning
      - 13.3.1. The Game of Life

13.3.2. Evolutionary Programming