Fall 2002: MCS 380: Introduction to Artificial Intelligence

Dr. William C. Regli
Department of Computer Science
College of Engineering (Yea!)
Office: Korman 283
Tel: +1 (215) 895.6827
Fax: +1 (215) 895.1582
regli@drexel.edu
http://www.mcs.drexel.edu/~wregli

Course Time/Place:
M-W-F, 14:00-15:00; Room: Curtis 341.

Office hours:
M-W-F, 15:00-16:00; or by appointment.

Teaching Assistants:
Max Peysakhov (umpeysak@mcs.drexel.edu),
Joe Kopena (joe@plan.mcs.drexel.edu),
Andy Mroczkowski (arm22@drexel.edu).

TA Office Hours: by appointment.

TA Office: Learning Center, Korman 249

Required Textbooks:


Other Resources:

• Course Homepage: http://edge.mcs.drexel.edu/regli/AI
• Textbook and CODE REPOSITORY: http://www.cs.berkeley.edu/~russell/aima.html
• Course Email List: mcs380@lists.mcs.drexel.edu. To subscribe to class email list: send email message to majordomo@lists.mcs.drexel.edu with the words “subscribe mcs380” in the body of the email message. To unsubscribe from this list send email message to majordomo@lists.mcs.drexel.edu with the words “unsubscribe mcs380” in the body of the email message.

I will post (on the email list and on the web site) homework assignments and other announcements, and will keep a running list of the topics we have covered. If I post something to the email list, you are assumed to have read and understood the message.

Course Objectives. This course will provide a foundational introduction to topics in Artificial Intelligence, emphasizing search, knowledge representation and planning.

Pre-Requisites. Students are required to have taken MCS260 (Introductory Algorithms) and MCS360 (Programming Languages). Students are assumed to either have an introductory knowledge of the LISP and/or the Scheme programming language or a willingness to learn it. While some LISP overview is given in this class, it should be noted that this course is an “Intro to AI,” not a full-blown “Intro to LISP” and that students are expected to learn the LISP they need to know to complete the assignments on their own.

Students should also have a thorough understanding of algorithms and algorithm design techniques, the use and analysis of pseudo-code. Many of the homework assignments in the class will be highly mathematical—students should be well versed in proof techniques (proof by induction, contradiction, etc.) and the mathematics of recursion and summations. You should be intimate with all material in R& N Appendices A and B.

Lectures. This list is tentative and will be modified at the instructor’s discretion. The course will focus on AI search, Representation and AI planning. We will cover Chapters 1-8, and 11-13.
Week 1: Introduction to AI, Uninformed and Heuristic Search (Chapters 1-4), LISP
Week 2: LISP, Heuristic Search (Chapter 4)
Week 3: Heuristic Search and Game Playing (Chapters 4-5)
Week 4: Heuristic Search and Game Playing (Chapters 4-5)
Week 5: Game Search, AND/OR Search (Chapter 5); Midterm Review
Week 6: Logic and Knowledge-Bases (Chapters 6-8)
Week 7: Logic and Knowledge-Bases (Chapters 6-8)
Week 8: Introductory Planning (STRIPS) (Chapters 11-12)
Week 9: Real-World Planning (HTN Planning, SHOP)
Week 10: Other topics in AI (robotics, vision, CBR, etc.)

We may hold 1-2 BRIEFINGS after class or at other various times during the semester. These briefings will expand on the topics covered in class in greater detail and be open forums for free-form discussion. Attendance is optional. Specific briefing topics will be determined as the class progresses (e.g., LISP, search, planning, etc.).

Grading: Subject to the constraints in the section Ways to Fail this Class, the grading will be computed as follows:

40% Midterm and Final Exams (20% each)
30% Homework Assignments (7-10)
30% Programming Assignments (6-10)

Class participation is important and will be considered in borderline cases. The final exam will be cumulative. An “A” grade requires excellence in all of the grading categories, with all homeworks completed in their entirety and handed in on the due date in class. A passing grade requires the demonstration of a minimum proficiency in each aspect of the class (exams, homework and programming).

Ways to Fail this Class: To assist students interested in spending the term working toward earning an “F” in this class, here are some ways to earn your “F” more quickly: (1) ignore all of the programming assignments by not handing them in or by handing in projects that do not run, in an effort to get partial credit; (2) fail to hand in more than 50% of the homeworks; (3) miss any one exam or obtain less than 25% on all of the exams; (4) falsify results; (5) mis-represent another’s work as your own (i.e., plagiarism) or violate the course “Discussion and Collaboration Policy.”

Workload and Assignments. There will homework and/or programming assignments due every week. Expect to spend 10 hours per week on this class. Programming assignments are to be submitted in LISP. Homeworks and programming assignments should always be submitted in class. Homeworks and programming assignments will not be accepted after the class starts. It is also assumed that students read the textbooks, documentation and web pointers provided over the course of the class.

Makeup Exams. There will be no makeup exams. To get credit for a missed exam, you will need a valid medical excuse. This means an official letter (including the dates of incapacitation for your illness), either from your private physician or from the University’s Health Center. If you have a valid medical excuse, I’ll give you credit for the missed exam based on your performance on the other assignments and exams.

Discussion and Collaboration Policies. All students should be familiar with the University’s policies on academic dishonesty, as well as those governing all aspects of our collegial life, as described in the University Handbook. Incidents of academic dishonesty will be brought to the honor board and face possible academic suspension or expulsion. While I encourage students to collaborate, all homework, proofs, and code submitted as part of assignments must be the student’s own.