

Programming for Artificial Intelligence and Cognitive Science

[CSCI 4969, COGS 4960, PSYC 4962]

Where/When:

- ▶ Sage 2715; Monday and Thursday 12:00–1:50pm

Class Instructor:

- ▶ Dr. V. Daniel Veksler [vekslv@rpi.edu; x6067]

Office Hours:

- ▶ by appointment only (walk-ins may try Carnegie 213 on Wednesdays, 2-4pm)

Class Website: <http://www.cogsci.rpi.edu/~vekslv/classes/paics09/>



Course Description

This course is an introduction to research programming for Artificial Intelligence and Cognitive Science. The course will emphasize the use of various programming languages (e.g. Python, LISP, etc.) for AI and CogSci projects (e.g. adaptive agents, NLP, reinforcement learning, neural networks, experimental design, games, statistical analysis, etc.). This will be a project-oriented course with an emphasis on research.

Prerequisites

Familiarity with computer programming, CSCI-I 100 [CSCI-I 200 preferable].



Course Objectives

Upon completion of this course, students should:

- ▶ Understand the phrase “the right tool for the right job”
- ▶ Be able to assign various programming ‘tools’ for their appropriate CogSci and AI ‘jobs’
- ▶ Be able to develop RL and NN agents
- ▶ Be able to develop models of semantic relatedness
- ▶ Have experience writing research code and analyzing the results



Required Text

No required text.

Helpful readings:

- ▶ *Reinforcement Learning: An Introduction* by Sutton & Barto
- ▶ *On Intelligence* by Jeff Hawkins
- ▶ *Computational Explorations in Cognitive Neuroscience* by O’Rielly & Munakata
- ▶ *ANSI Common Lisp* by Paul Graham



Grading and Attendance

- ▶ **30% homework/quizzes**
 - ▶ There is currently one homework assignment scheduled, due before the Sep 10th lecture
 - ▶ There will also be short quizzes at the end of each lecture, on the material presented during that lecture
- ▶ **30% attendance, meetings, and experiment participation**
 - ▶ There are only 23 scheduled lectures, plus 3 short meetings. Be there.
 - ▶ You will be required to participate in a 1hr experiment.
- ▶ **10% project presentation**
 - ▶ During the final lectures of the course, each student will present their work to the class. These will not be graded on your presentation skills, but rather on the content of the presentation.
- ▶ **30% final project**
 - ▶ Each student is responsible for scheduling a final meeting (before grades are due, exact date TBA) with the instructor to hand in their project and explain their work. All code that is not written by you must include a citation.



Class structure

- ▶ In general, the course will follow a lecture format with ample opportunity for class discussion.
- ▶ There will be at least three mandatory personal meetings with each student (check the schedule for dates). Students are encouraged to schedule additional meeting times, as well.

Mid-term Assessment Method

- ▶ Personal meetings will serve as the forum for periodic progress assessments. It is each student's responsibility to schedule meetings with the instructor, and to ask for clarification if their status is unclear
-



Date	Lecture	
31-Aug	Introduction to PAICS	
3-Sep	Programming Languages	
7-Sep	No Class	
10-Sep	Reinforcement Learning Agents	Assignment due (lang comparison)
14-Sep	Semantic Models	
17-Sep	Meetings	
21-Sep	Programming Tools	
24-Sep	Programming Tools	
28-Sep	No Class	
1-Oct	Brain/Cognition	
5-Oct	Neural Nets	
8-Oct	Neural Nets	
12-Oct 13-Oct	Probably No Class	
15-Oct	Meetings	
19-Oct	Neural Nets	
22-Oct	Categorization	
26-Oct	HTM	
29-Oct	HRL	
2-Nov	Param Search, Genetic Algorithms	
5-Nov	Meetings	
9-Nov	Experiments	
12-Nov	Experiments	
16-Nov	Stats	
19-Nov	Cognitive Architectures	
23-Nov	Cognitive Architectures	
26-Nov	No Class	
30-Nov	Presentations	
3-Dec	Presentations	
7-Dec	Presentations	
10-Dec	Presentations	
13-Dec (date tentative)		Final Project submissions due

IMPORTANT: This schedule is subject to change. You are responsible to be aware of these changes. Any changes will be announced during class times. We will make every effort to post any changes to the class website as soon as any changes are made.

Research Projects in AI & CogSci

- ▶ Pattern Recognition; Voice Recognition
- ▶ Vision; NLP
- ▶ Machine Learning; Smart Agents/Tools; Games
- ▶ Cognitive Psychology Experiments
- ▶ Cognitive and Social Simulations
- ▶ Human-Computer Interaction
- ▶ Statistical Analysis
- ▶ Parameter Search; Genetic Algorithms



Tools for AI & CogSci

- ▶ **Compiled code**
 - ▶ e.g. C, C++
- ▶ **Interpreter**
 - ▶ e.g. Python, LISP, MATLAB
- ▶ **Web Apps**
 - ▶ e.g. CGI, SQL, DHTML, AJAX
- ▶ **Cognitive Frameworks**
 - ▶ e.g. ACT-R, SOAR, CLARION, Polyscheme, HTM, LEABRA
- ▶ **Statistical Packages**
 - ▶ e.g. Excel, SPSS, R



AI vs CogSci

- ▶ **Artificial Intelligence**
 - ▶ Code developed with more regard for solving a particular problem than for simulating human/animal cognition
- ▶ **Cognitive Systems/Cognitive Models**
 - ▶ Code developed with more regard for human/animal cognition than for addressing a specific problem
- ▶ **Not necessarily at odds with each other...**



AI vs CogSci

- ▶ **Best AI comes from Cognitive theory**
 - ▶ e.g. Neural Networks, Reinforcement Learning, NLP, HCI
- ▶ **Best Cognitive Models come from solving real problems**
 - ▶ e.g. Cognitive tutors, driving simulations, NLP, HCI
- ▶ **Not necessarily at odds with each other...**
 - ▶ "There is nothing so useful as a good theory." - K. Lewin (1951)
 - ▶ "Nothing drives basic science better than a good applied problem." Newell, A., & Card, S. K. (1985)



Why did you take this class?



The right tool for the right job

- ▶ **Processing Speed**
 - ▶ compiled code may be faster interpreter
 - ▶ vectors may be faster than lists
- ▶ **Coding Speed**
 - ▶ some tools may be easier to use than others
 - ▶ as a developer you may be more familiar with some tools
- ▶ **Networking and Cross-Platform Constraints**
 - ▶ web apps may be more appropriate for many projects
 - ▶ some programming languages are platform-bound
- ▶ **Multiprocessing**
 - ▶ grid vs cluster?
 - ▶ different mp environments have different requirements
 - ▶ some languages are better for mp than others
- ▶ **Simplicity**
 - ▶ the right tool for the right job...



How many of you know...

- ▶ C
- ▶ Python
- ▶ LISP
- ▶ JAVA
- ▶ MatLab
- ▶ SQL (mySQL)
- ▶ R
- ▶ Excel
- ▶ SPSS
- ▶ ACT-R, SOAR, CLARION, PolyScheme



Class Projects

- ▶ **Research, Not homework**
 - ▶ Nobody has done what you will be doing
- ▶ **Research Projects**
 - ▶ Adaptive Autonomous Agents
 - ▶ Applied Models of Natural Language Processing



Best case scenario...

▶ Publications

- ▶ Your work may be publishable [and I would help you]
- ▶ A publication sets your resume apart from others
- ▶ A publication gets you into graduate school
- ▶ Your work will have an impact on the scientific world
- ▶ ...but, publications are hard work, that will surely extend beyond the semester

▶ Recommendations

- ▶ This is a small senior-level course, I will likely get to know each of you

