CS 550 – Artificial Intelligene
Mo 16:30-17:45, Wed: 16:30-17:45
Instructor: Dr. Shota Tsiskaridze
Office Hours: Upon in advance agreement
Contact: shota.tsiskaridze@gmail.com
Course Web Page: https://github.com/stsiskar/IU/tree/master/CS550 (access is granted individually, over email)

Materials

Required:

- Russel, S.J. and Norvig. P (2010), Artificial Intelligence: A Modern Approach (third edition, Prentice Hall, Upper Saddle River).
- Deep Learning with Python FRANÇOIS CHOLLET (©2018 by Manning Publications Co.)
- Python for Programmers, Paul Deitel Harvey Deitel, Pearson, 2019

Optional:

- Programming in this course is in Python: Optional text: Martelli, A. (2006). Python in a nutshell (O'Reilly, Sebastopol, CA). or Lutz, M (2013). Learning Python (O'Reilly Media, Sebastopol, CA).
- Paul R. Halmosh, Finite-Dimensional Vector Spaces, Springer, 1958.

Course Information:

Catalog description: Heuristic approaches to problem solving, System methods of search of the problem state space. Theorem proving by Machine.

Prerequisites: CS 108, Mathematics: Discrete Math (245) or Math Logic (523), Calculus and Linear Algebra courses would be beneficial

Goals:

After successful completion of the course, students will be able to:

- Construct intelligent agents capable of interacting with their environments.
- Understand and implement heuristic searches for problem solving and game playing (adversarial search)
- Analyze constraint satisfaction problems and resolve them through search.
- Understand the math and be able to apply machine learning algorithms

Relationship to CS Program Course Outcomes

CS 550 addresses the following CS Program course outcomes:

a) An ability to apply knowledge of computing and mathematics
b) An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
c) An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs
d) An ability to function effectively on teams to accomplish a common goal
h) Recognition of the need for and an ability to engage in continuing professional development

i) An ability to use current techniques, skills, and tools necessary for computing practice

j) An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

k) An ability to apply design and development principles in the construction of software systems of varying complexity.

Topics Covered:

Elements of Linear Algebra, Calculus, Probability and Statistics, Intelligent Agents, Theorem Proving, Space Search, Machine Learning;

Course Schedule:

A week by week schedule may be found on the blackboard. There will be five problem sets, will be assigned with deadlines varying from few days to one week depending upon the complexity. Problem sets consist of a combination of short answer prose, quantitative problems, and programming. Work must be shown for quantitative problems, and programs must use appropriate levels of abstraction and be well commented. See the course guidelines provided during the first lecture on appropriate program structure. Solution of each homework will be discussed in detail after the deadline. Besides the homework there will be assigned optional programming exercises, solution of which will be also discussed during the lecture.

Midterm dates are specified in the detailed course schedule available on blackboard.

Grading policies:

Maximum points for each homework assignment is 6 points, there are 5 homework assignments during the course. Total contribution of homework assignments grades to the final mark is 30%.

Midterm Exam I - 20% of total grade

Midterm Exam II - 20% of total grade

Final Exam - 30% of total grade

This course uses a coarse grading system. It is very difficult to justly and systematically determine that one answer is worth N points and another is worth N±ε. Consequently, points are assigned based upon broad categories that indicate your mastery of the concept:

A+ - Excellent	B – Mostly right	VE – Valiant effort
A - Good	C – Right track	F – Unacceptable

Accommodation of disabilities

If you are a student with a disability and believe you will need accommodations for this class, it is your responsibility to contact Student Disability Services at (619) 594-6473. To avoid any delay in the receipt of your accommodations, you should contact Student Disability Services as soon as possible. Please note that accommodations are not retroactive, and that accommodations based upon disability cannot be provided until you have presented your instructor with an accommodation letter from Student Disability Services. Your cooperation is appreciated.

Academic Honesty:

You are free to discuss ideas and strategies for approaching problems with others, but students must complete work on their own. Using other people's work in any form (i.e. the web, other students) will result in disciplinary action and failing the course. Note that pair programming teams are an exception to this rule. Plagiarism is unacceptable and will not be tolerated. You are responsible for understanding plagiarism; the library has a tutorial at http://library.sdsu.edu/guides/tutorial.php?id=28. If you have any questions about plagiarism after taking the tutorial, I will be happy to assist you.