Midterm Review
OPEN BOOK, OPEN NOTES, TAKE-HOME EXAM
Will be out Wednesday 2/9 and turned in Saturday 2/12 11:59pm

CSE 473
1. Search
   • Be able to give a formal state-space model for a problem expressed in English. Formal means to specify $S$, $s$, $A$, $f$, $g$ and $c$ as sets or functions as appropriate.

   • Be able to specify what would be the dead states for a given problem.

   • Be able to generate part of a search tree for a given model, either depth-first or breadth-first.
• Be able to answer questions about the completeness and complexity of the various search variants given in Chapter 3-4.

• Be able to explain the use of a heuristic function in a search or to give an example of one for a stated problem.

• Be able to motivate the use of heuristic-search vs. blind search.
• Be able to apply any of the following search methods to a well-stated problem and show a portion of the search.
  – greedy best-first search
  – A* algorithm
  – steepest-ascent hill climbing
• Be able to answer questions about admissibility and consistency with respect to heuristic functions for A*.

• Be able to describe or analyze the simulated annealing approach and its advantages/disadvantages and variants.

• Be able to answer questions about complexity, completeness, and optimality for the above algorithms.
2. Game Playing
- Be able to develop a utility function for a given game or show how a given one works.

- Be able to show how a basic minimax search works for some given example.

- Be able to show how the alpha-beta procedure works for some given example.

- Be able to show how shallow search might be used to improve the alpha-beta procedure.

- Be able to answer questions about how Samuel’s checker player works.

- Be able to answer questions on how minimax generalizes to games of chance.
3. Constraint Satisfaction Problems

• Be able to formalize a constraint satisfaction problem by specifying the sets of variables, possible values, and constraints.

• Be able to explain or illustrate how a backtracking tree search for a constraint satisfaction problem would work: alone, with forward checking, or with arc consistency checks.

• Be able to compare how a general heuristic search would compare with a constraint satisfaction search when both are applicable to a given problem.
4. Machine Learning

• For this exam, be able to try out the classifiers on WEKA.

https://sourceforge.net/projects/weka/