Last update: March 9, 2010

# Review for the Midterm Exam

## CMSC 421: Midterm Review

# Midterm Exam

◇ Scope: Chapters 1–6, and Common Lisp
   The test won't include Chapters 7 and 8

◇ Open book, open notes

◇ No electronic devices

My exams can sometimes be very hard
But don't let it bother you, because I grade on a curve
e.g., when I taught CMSC 421 in Fall 2007:

| Midterm | Final | GPA |
|---------|-------|------|
| 62%     | 64%   | 2.88 |

To help you prepare, the "private materials" web page has the midterm and final exams for the last four times I taught the course

It also has the answers to the first two homework assignments
Later today, I'll post the answers to the third one

# Chapter 1: Intelligent Agents

◇ What AI is:

◇ thinking versus acting

◇ humanly versus rationally

I won't ask any questions about Chapter 1

# Chapter 2: Intelligent Agents

◇ Agents and environments

◇ Rationality

◇ PEAS (Performance measure, Environment, Actuators, Sensors)

◇ Environment types

◇ Agent types

I probably won't ask much about Chapter 2

# Chapter 3: Search

◇ Problem types:
  deterministic, nondeterministic,
  fully observable, partially observable, non-observable
 example: vacuum world

◇ Tree-search algorithms
  Breadth-first search
  Uniform-cost search
  Depth-first search
  Depth-limited search
  Iterative deepening

◇ tree search versus graph search

# Chapter 4: Informed Search and Exploration

◇ Heuristic search algorithms
    Greedy search
    A* on trees or on graphs with consistent heuristics
    A* on graphs with inconsistent heuristics

◇ Heuristic functions
    admissibility
    consistency
    dominance
    problem relaxation

◇ Iterative improvement algorithms
    Hill climbing, simulated annealing,
    local beam search, genetic algorithms

◇ Not on the exam:
    IDA*
    sections 4.4 (continuous spaces) and 4.5 (online search)

# Common Lisp

◇ lists, atoms, list notation

◇ defining your own Lisp functions

◇ built-in Lisp operators (functions, predicates, special forms, macros)

◇ recursion, loops, and mapping functions

◇ passing functions as arguments

◇ operators for sequences (lists, vectors, strings)

◇ good programming style
(no direct questions on this, but don't write sloppy code!)

◇ Not on the exam:
- destructive operations (e.g., `setf`, `nconc`)
  versus nondestructive operations (e.g., `setq`, `append`)
- the xkcd comics 😵
- interacting with the debugger

# Chapter 5: Constraint Satisfaction

$\diamond$ Definition: variables, constraints

$\diamond$ Representation: constraint graphs

$\diamond$ Backtracking search

$\diamond$ Variable selection heuristics:

    MRV (minimum remaining values)

    degree (most constraints on remaining variables)

$\diamond$ Value selection heuristic: least constraining value

$\diamond$ Pruning techniques

    forward checking

    arc consistency (constraint propagation)

$\diamond$ Problem structure:

    independent subproblems

    tree-structured CSPs

    cutset conditioning

# Chapter 6: Adversarial Search

◇ What type of game:
two-player, perfect information, zero sum

◇ Game trees, minimax values

◇ Alpha-beta pruning

◇ Depth-bounded search, static evaluation functions

◇ Node ordering

◇ Nondeterministic game trees (e.g., backgammon)
    expectiminimax