Announcement

Project 1 preliminary submission requirement:

- Submit whatever you have done up till 3/8/02 (Friday).
- The program does not need to run.
- The preliminary submission will not be counted toward your project grade, however, if you don’t submit it, 3% will be taken off from your final project 1 grade.

Office hour today: 4:30-5:30pm (due to committee and faculty meetings)

Midterm Review: Overview

- Clarifications
- AI basics and LISP: about 10%
- Search as a problem solving strategy
- Game playing
- Propositional logic

Clarification: Ridges in Hill-Climbing

When the only operations allowed are $N/S/W/E$:

- At 20 ft. all directions $N/S/W/E$ are downhill.
- The only way to go up is either $N$ then $E$, or $E$ then $N$.
- Greedy algorithms only look one step ahead, so they get stuck in a local minima on a ridge.
- A 2-step look ahead strategy solves the problem in this case.

Clarification: Order of Quantifiers

$\exists x \forall y \text{ Loves}(x, y)$

$\forall y \exists x \text{ Loves}(x, y)$

$\forall x, \exists y \neq \exists y, \forall x$

- $\exists x, \forall y, \text{ Loves}(x, y)$
  There is someone who loves the entire human population.

- $\forall y, \exists x, \text{ Loves}(x, y)$
  For any person, there is someone who loves that person.
Lectures 1–5

- Disciplines with ties to AI: think about how they did and would contribute
- What are the hard problems in AI? Why are they hard?
- LISP basics: know how to write a simple recursive function (defun, if, cond, arithmetic functions)
  - no need to know setq, setf, cons, car, cdr, etc.

Lecture 6: Key Points

- Description of a search problem: initial state, goals, operators, etc.
- Considerations in designing a representation for a state
- Evaluation criteria
- BFS, UCS, DFS: time and space complexity, completeness
- Differences and similarities between BFS and UCS
- When to use one vs. another
- Node visit orders for each strategy
- Tracking the stack or queue at any moment

Lecture 7: Key Points

- DLS, IDS, BDS search order, expansions, and queueing
- DLS, IDS, BDS evaluation
- DLS, IDS, BDS: suitable domains
- Repeated states: why removing them is important
- Constraint Satisfaction Search: what kind of domains? why important?
- Why greedy search can fail?

Lecture 8: Key Points

- best-first-search: definition
- heuristic function $h(n)$: what it is
- greedy search: relation to $h(n)$ and evaluation. How it is different from DFS (time complexity, space complexity)
- $A^*$: definition, evaluation, conditions of optimality
- designing good heuristics: several rule-of-thumbs
- Basic idea of $IDA^*$
- Basic idea of iterative improvement algorithms
Lecture 9: Key Points

- Complexity of A*: relation to error in heuristics
- IDA* details: evaluation, time and space complexity (worst case)
- Hill-climbing problems and strategies
- Beam search concept
- Simulated annealing details: core algorithm, effect of $T$ and $\Delta E$, source of randomness.

Lecture 10: Key Points

- Game playing: what are the types of games?
- Minimax: definition, and how to get minmax values
- Minimax: evaluation
- $\alpha$-$\beta$ pruning: why it saves time

Lecture 11: Key Points

- formal $\alpha - \beta$ pruning algorithm: know how to apply pruning
- $\alpha - \beta$ pruning properties: evaluation
- games with an element of chance: what are the added elements? how does the minmax tree get augmented?

Lecture 12: Key Points

- Knowledge representation: logic and frames, pros and cons
- Knowledge bases: the basic components
- Propositional Logic: basic laws
- Inference rules: what is inference, basic inference rules
- Normal forms: definitions
Lecture 13: Key Points

- Propositional Logic: basic laws
- Inference rules: what is inference, basic inference rules, applying inference rules
- Normal forms: definitions, know how to convert, applying basic laws and inference rules
- Theorem proving: basic approaches. forward and backward chaining concept

Lecture 14: Key Points

- understand why resolution works: proof by contradiction
- standard formulas used in resolution (CNF, and negation of conclusion).
- know how to follow resolution steps to prove (or disprove) a theorem.
- given a natural language description, know how to convert to propositional logic formulas.

Lecture 15: Key Points

- know how to do resolution in propositional logic

General Rule-of-Thumb

- Get used to basic concepts (representation of search problems, game playing basics, propositional logic constructs and rules)
- Try not to blindly memorize what's there, try to understand why/how something works or does not work:
  - example; time and space complexity of search strategies
- Try out the exercises done in the class.
- Try out exercises in the textbook related to the material presented in the lectures.