CPSC 433 F22 Midterm Study Questions (Topics 1-9)
Most questions are short answer questions.

1Organization
• Name the subareas of AI (from organization slides).

2Introduction
• What is a good definition for AI?
• What are challenges in creating a definition for AI?
• What theory of technology development can be used to describe the challenge of AI's historical progress?
• What do we mean by AI being able to be defined as Computational Rationality?
• What is a rational agent?
• What type of AI existed before AI-Winter I/what about that type of AI didn't succeed as expected?
• First AI philosopher?
• Well known thought experimental test for judging if we've reach full human level AI is called?
• What type of AI succeed after AI Winter I before AI Winter II?
• What types of AI have succeeded post AI Winter II? What are some pivotal events post AI-Winter II in AI?
• Compare connectionist AI to symbolic AI.
• What are some future/currently developing AI areas from class slides?

3Knowledge Representation
• What is a reflex agent?
• What is a planning agent?
• What is the general structure of an AI system, what knowledge is used and why do we have abstraction levels?
• What is the difference between computation and search?
• What are the advantages of computation, what are the disadvantages?

4Search Definitions
• What does Learning mean in AI, what does Planning/Deduction?
• What is the 'No Free Lunch Theorem'?
• What is the definition of a search model? (be able to describe the parts not just A = (S,T))
• What is the definition of a search process? (be able to describe the parts)
• What is the definition of a search instance? (be able to describe the parts)
• What is the definition of a search derivation? (be able to describe the parts)
• For what do we need the components of a search model, process, instance, derivation?
• What is a state/search space? What is a value space/fitness space?
• What type of information would be put in the environment versus the states of a search?

5/6/7 General
• What are the problem specific components of

1. set-based search
2. and-tree-based search
3. or-tree-based search?

• What are the differences between and-tree-based search and or-tree-based search?
• Where to we need to make sure we handle tie-break in and/or/set based search? Why do we need to do so?
• What is the purpose of a problem specific component? (Ex. What does the declaration of F create for us in a Set-Based model?)

5Set Based

• Given two terms, decide if they are unifiable and if yes, generate their mgu. [medium length question]

Examples:

1. $f(x, y, g(x, c, d)) \approx f(g(y, a, d), a, g(x, c, d))$
2. $f(x, y, x)) \approx f(a, g(x, b, c), d)$
3. $f(x, c, d) \approx f(f(x, c, d), c, d)$

• Given two terms, apply the set-based search process for generating their mgu and write down a possible search derivation. Examples: see above
• Given a solvable unification problem and our set-based search model for it. If we have two different search processes for the model, will the mgus produced by the processes be different?

• Given two clauses and the Resolution inference rule, produce all clauses that are the result of applying the rule to the clauses. [medium length question]

Examples:

1. $Q(a) \vee Q(b), \neg Q(x)$
2. $P(a, b) \vee P(x, d), \neg P(a, y)$
3. $P(a, b) \vee P(x, d), \neg P(b, y)$

• Given a clause and the inference rule Factorization, produce all clauses resulting from the application of the rule. [medium length question]

Examples:

1. $Q(a) \vee Q(x)$
2. $P(a, b) \vee P(x, d) \vee P(a, y)$
3. $\neg P(b, y) \vee \neg P(x, y)$
4. $P(a, x) \vee P(b, y)$

6And Tree

• What do we call the unexplored are of a search tree? What storage structure do we use to store it? Do we usually store the complete search tree why/why not?
• If I use the provided and-tree definition what does ERW define (ERW*)? Do I need to re-define them for my specific problem?
• What is a single search state in an and-tree? What is an Atree?
• Why is a reason we would make an and-tree with backtracking? What is a reason we might try to avoid it in an and-tree solution?

• If our and-tree algorithm is a loop that starts with the starting root in our minheap, what does each loop of our algorithm do (using f_leaf and f_trans in your explanation)?
• If a recursive brand and bound algorithm is being considered an and-tree, would be consider it to be a back-tracking design or not? Why?
• Given some clauses and a state in a model-elimination search, produce all possible successor states. [medium length question]
• Given a state in model-elimination search that has only leafs with sol-entry yes, check if it really fulfills the end condition.

7Or Tree
• Given an or-tree-based search state for a constraint satisfaction problem instance (and the instance) and a leaf in this state, list all possible successor states that extend the leaf.
• What is the difference between and-tree-based search and and-or-tree-based search? (in terms of purpose?, in terms of function?)
• Are or-trees good for optimization problems? What about and-trees?
• In terms of tree definition do we use both ERW and ERW*? Why/ why not?
• What are a couple of or-tree parts that are just rename versions of and-tree parts?
• What do we like or-trees for CSP problems? What are some examples of CSP type problems?
• Given some variables X, the domains for those variables D, and some constraints for those variables C, perform an or-tree based search like given in lecture. [medium length question]
• What were general considerations made when designed f_leaf/Altern for the CSP problem in lecture? What is another name for some of these indeterminate rules that were used?
• In CSP what is meant by the critical ratio?
• Why are some CSP pre-processed for structure changes and then multiple CSP run on the result and combined? What could be another name for an algorithm that runs and combines (conjuncts) multiple or-trees?

8Other search models
• What is the difference between tree-based search and graph-based search?
• What is a benefit of graph-based search? What is a challenge?
• Why did blocks world gain from a graph-based consideration?
• What type of search was declared as fitting And-Or-Tree search well?
• What is meant by a zero-sum game?
• What game did Deep Blue play? What game did UofA research give solution for, what was that system named? What recent game did an AI winning a competition change what AI was considered capable of (ahead of expected timeline)?
• What is a simple summary of how the min and max parts of minimax algorithms inter-operate? What are some assumptions that minimax (and some similar algorithms require to give the correct answer?)
• What is a complete algorithm? What is an optimal algorithm?
• How do nodes in tree scale if b is branches per node, and m is the depth tree is at? (Big-O)
• Why would we prune a tree? Why does depth matter? What algorithm was described which changed minimax to add pruning? What is simple summary of how alpha and beta are used?

• What is the simple motivation that resulted in Expectimax search? What is danger of optimism with stochastic games? Pessimism?

9Search Controls
• What are general things that can be measured by a search control?
• Which parts of states can be measured by search controls?
• Compare DFS to BFS. What is brief description of iterative deepening? How is uniform cost different from BFS? (I will not ask for asymptotic comparisons (big-O) or calculations!)
• What is informed search versus uninformed search and what search controls fit in each category?
• What is a heuristic? What is an example of a distance heuristic?
• What does greedy search do?
• What is an admissible heuristic?
• What two algorithm designs is A* a combinations of? What property does A* require to be optimal?
• What is mean by local search? Which of and-tree, or-tree, set-based is most often a local search?
• What is hill-climbing defined by? What does simulated annealing add to the idea of hill-climbing?
• Compare particle swarm optimization and genetic algorithms?

Long answer question types
• Given a concrete search model, a concrete search process to the model and a search instance, write down the search derivation produced by the process for the search instance. [harder question on exam – at most 1]

• Given a problem and a search paradigm, produce a search model and search process that solves instances of the problem and follows the paradigm. [hardest question type on exam – at most 1]