# Exam 1 Review (Chapters 1 – 4, 6 – 9)

## Types of questions

- Definitions
- Short questions
- Comparisons
- Problem solving (simple problems)
- Proofs

- **State Space**
  - States, state transition rules/operators/actions, and costs associate with operations
  - State space, node generation and **node expansion**, open/closed nodes, open/closed lists.
  - Solution, solution path and its cost.
  - Be able to represent simple problem-solving as state space search

- **Uninformed (blind) Search Methods**
  - Search methods (BF, DF, **IDDF**, **Uniform-cost**), their algorithms, time and space complexities, optimality and completeness, their advantages and limitations.

- **Informed Search methods**
  - Evaluation function f(n),
  - Heuristic estimate function h(n)
    - what does h(n) estimate
    - admissible h(n), null h(n), perfect h*(n), more informed h(n)
  - Best first search:
    - node selection from open list according to f(n)
    - delayed goal testing
  - Algorithm A and A*
    - f(n) = g(n) + h(n): what does each of the terms stand for?
    - algorithm (maintaining open/closed lists, delayed goal test; node expansion, handling duplicate nodes, back pointers);
    - difference between algorithms A and A*
    - time and space complexity, completeness and optimality of A*
    - be able to apply A* to simple problems.
    - Be able to prove simple properties related to A* search
  - Ways to improve A* search
    - IDA* (basic idea; how to set f_limit at each iteration; advantages over A*)
    - Pruning open list by f+, where f+ is an upper bound of the cost for the optimal solution (e.g., the cost of any known solution)
  - Greedy search and hill-climbing (algorithms, time and space complexity, completeness and optimality)
  - Basic ideas of simulated annealing for seeking optimal solutions

- **Game-Tree Search**
  - Game tree (Max and Min nodes; look ahead, terminal and leave nodes)

  - What to search for (one move for Max with maximum guaranteed payoff)
  - Heuristic evaluation function f(n) (merit of a board configuration)
  - Minimax rule for game tree search
  - Alpha-beta pruning, its time and space complexities.
  - Difference between general state space search and game tree search
  - Be able to apply Minimax rule and alpha-beta pruning to simple problems.

- **Propositional Logic (PL)**
  - Syntax
  - Semantics
    - Interpretation (an assignment of truth values to all prepositional symbols); models
    - Truth tables for logical connectives
    - Valid (tautology), satisfiable and inconsistent (contradiction) sentences
    - Logical consequence or entailment (S |= X)
  - Equivalence laws
    - P ≡ Q iff they have the same truth tables
    - P => Q ≡ ~P v Q; distribution /associative/communicative laws, De Morgan's laws
  - Deductive inference
    - Using truth table (S |= X iff S => X is valid)
    - Using deductive rules
      - Modus Ponens, Modus Tollens, Chaining, And Introduction, And Elimination, etc.
      - Soundness of deductive rules
      - **Resolution rule** (and CNF)
    - Proof procedure, Soundness and completeness of proof procedures

- **First Order Logic (FOL)**
  - Syntax
    - Terms, predicates, atoms, literals, quantifiers, wff
  - Semantics
    - Interpretations and models, valid, satisfiable, and inconsistent sentence(s), Logical consequences
    - Be able to translate between English sentences and FOL sentences
    - Soundness and completeness of proof theory in FOL

- **Deductive Inference in FOL**
  - Convert first order sentences to clause form
    - Definition of clauses, converting FOL sentences to clause form (**Skolemization**)
  - Unification (obtain mgu θ)
  - Resolution .
  - Resolution Refutation
    - Write the axioms as FOL sentences and convert them into clause form
    - Write the goal (theorem) as a FOL sentence
    - Negate the goal and convert it to clause form
    - Select a pair of clauses for resolution which are i) resolvable, and ii) promising toward deriving a null clause,

    - Inference stops when a null clause is derived
- Be able to do resolution refutation on simple problems.
- Other issues
    - Semi-decidability
    - Forward and backward chaining