KBAI Exam 1 Unofficial Study Guide

Daniel Torres

## References:

* Lecture Slides
* AI - Patrick Henry Winston (3rd edition)
* <https://quizlet.com/309763851/kbai-ch-1-9-flash-cards/>
* <https://quizlet.com/322380942/flashcards>
* <https://quizlet.com/357056229/kbai-01-intro-kbai-flash-cards/>

## Content

[01: Introduction to Knowledge-Based AI (45)](#_4wkdblh0hoz)

[02: Introduction to CS7637 (60)](#_i0b2vgkucf1a)

[03: Semantic Networks (60)](#_d9vl5keto4n5)

[04: Generate & Test (30)](#_l2ernq37e2a8)

[05: Means-Ends Analysis (60)](#_apiozw10igr4)

[06: Production Systems (60)](#_dsmeiqu1oj7f)

[07: Frames (45)](#_dl010yj35o8z)

[08: Learning by Recording Cases (30)](#_r7nep2b4xua3)

[09: Case-Based Reasoning (60)](#_m6gthbrxf4y)

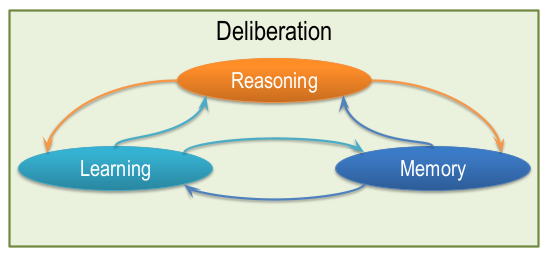
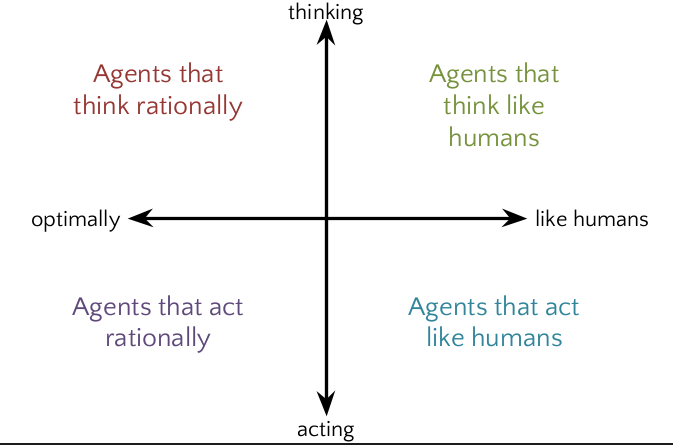
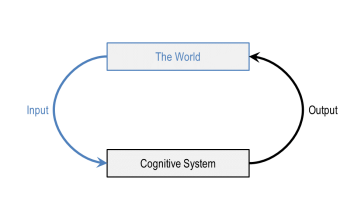
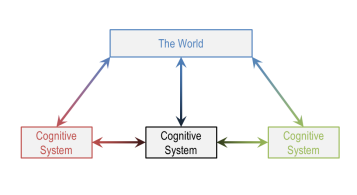
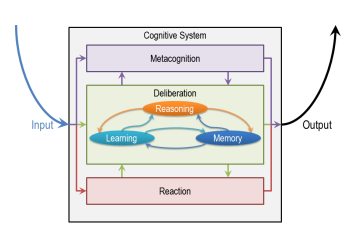
[10: Incremental Concept Learning (60)](#_4gqv6n9l312g)

[11: Classification (45)](#_6iwph8i2o12p)

[12: Logic (90)](#_qj5rzp4crfx0)

## 

## 01: Introduction to Knowledge-Based AI (45)

* Lesson Preview
  + What is knowledge-based artificial intelligence?
  + How does it fit into the rest of artificial intelligence?
  + What can I expect to learn from this course?
  + What is the structure of this course?
* Fundamental Conundrums of Artificial Intelligence
  + Intelligent agents have limited resources.
  + Computation is local, but problems have global constraints.
  + Logic is deductive, but many problems are not.
  + The world is dynamic, but knowledge is limited.
  + Problem solving, reasoning, and learning are complex, but explanation and justification are even more complex.
* Characteristics of AI Problems
  + Knowledge often arrives incrementally.
  + Problems exhibit recurring patterns.
  + Problems have multiple levels of granularity.
  + Many problems are computationally intractable.
  + The world is dynamic, but knowledge of the world is static.
  + The world is open-ended, but knowledge is limited.
* Characteristics of AI Agents
  + Agents have limited computing power.
  + Agents have limited sensors.
  + Agents have limited attention.
  + Computational logic is fundamentally deductive.
  + AI agents’ knowledge is incomplete relative to the world.
* Deliberation - Reasoning/Learning/Memory - Fundamental processes
  + 
* Four Schools of AI
  + Thinking/acting - optimally/like humans (human not optimal :O )
  + 
  + Thinking/optimal - rational agent Ex.: Machine Learning
  + Thinking like humans Ex.:Semantic Web
  + Acting optimally Ex.: Airplane autopilot
  + Acting like humans Ex.: Improvisational robots
* Cognitive Systems
  + What are cognitive systems?
    - Cognitive: dealing with human-like intelligence.
    - Systems: multiple interacting components such as learning,
    - reasoning, and memory.
    - Cognitive Systems: Systems that exhibit human-like intelligence
    - through processes like learning, reasoning, and memory.
  + Cognitive System Architecture
    - 
    - 
    - 
* Topics in AI
  + Fundamentals
    - Semantic Networks
    - Generate and Test
      * Means End Analysis
      * Problem Reduction
    - Production Systems
  + Planning
    - Logic
    - Planning
  + Common Sense Reasoning
    - Frames
      * Understanding
        + Common Sense Reasoning

Scripts

* + Learning
    - Learning by Recording Cases
      * Incremental Concept Learning
        + Classification
        + Version Spaces
  + Analogical Reasoning
    - Learning by recording cases
  + Visuospatial Reasoning
    - Constraint Propagation
      * Visuospatial Reasoning
  + Design & Creativity
    - Configuration
      * Diagnosis
        + Design

Creativity

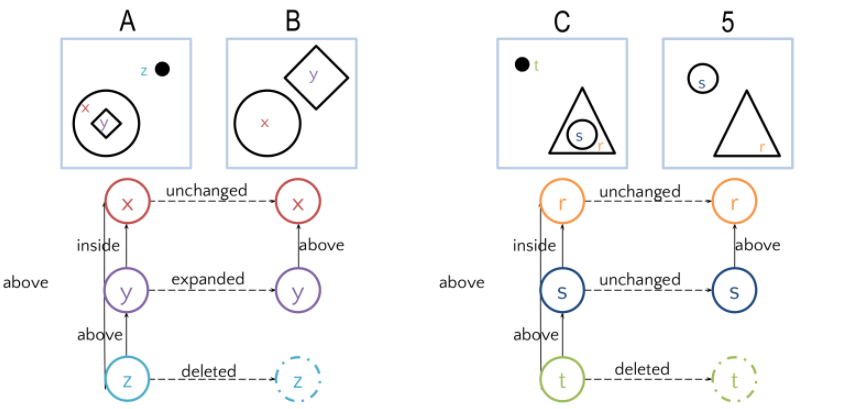
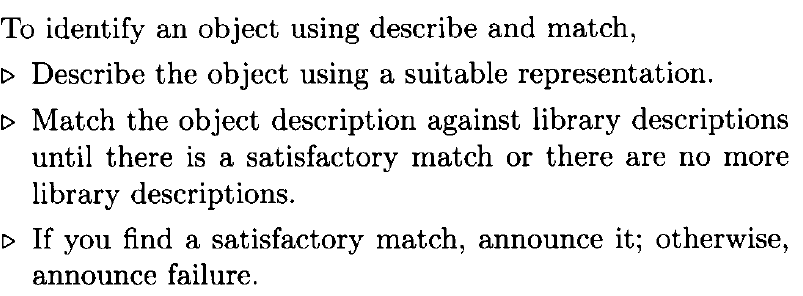
* + Metacognition
    - Learning by Correcting Mistakes
      * Meta-Reasoning
        + Ethics in AI

## 02: Introduction to CS7637 (60)

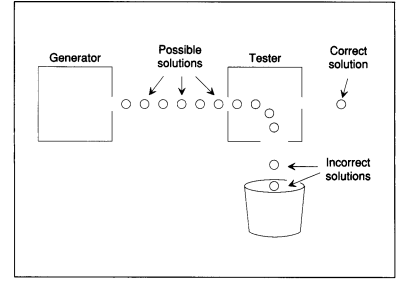
* Raven’s Progressive Matrices
  + Test written in the 1930s to examine general intelligence.
  + Consists of 60 multiple-choice visual analogy problems.
  + Unique in that problems are strictly visual.
  + Widespread usage as a valid test for intelligence.
* Principles of CS7637
  + 1. KBAI agents represent and organize knowledge into knowledge structures to guide and support reasoning.
  + 2. Learning in KBAI agents is often incremental.
  + 3. Reasoning in KBAI agents is top-down as well as bottom-up.
  + 4. KBAI agents match methods to tasks.
  + 5. KBAI agents use heuristics to find solutions that are good enough, though not necessarily optimal.
  + 6. KBAI agents make use of recurring patterns in the problems they solve.
  + 7. The architecture of KBAI agents enables reasoning, learning, and memory to support and constrain each other.
* Frequently-used readings
  + Artificial Intelligence by Patrick Winston
  + Knowledge Systems by Mark Stefik
  + Artificial Intelligence by Elaine Rich and Kevin Knight
  + Artificial Intelligence: A Modern Approach by Stuart Russell and Peter Norvig

## 

## 03: Semantic Networks (60)

* Structure of Semantic Networks
  + Lexically: nodes
  + Structurally: directional links
  + Semantically: application-specific labels
  + 
* Characteristics of Good Representations
  + Make relationships explicit
  + Expose natural constraints
  + Bring objects and relations together
  + Exclude extraneous details
  + Transparent, concise, complete, fast, computable
* Guards & Prisoners Problem
  + Also known by other names (cannibals and missionaries, jealous husbands, brothers and sisters).
  + Originally appeared in the 1200-year-old text Propositiones ad Acuendos Juvenes.
  + Used throughout AI for problem representation.
* Guards & Prisoners Problem - Statement
  + Three guards and three prisoners must cross the river.
  + Boat may take only one or two people at a time.
  + Prisoners may never outnumber guards on either coast, though prisoners may be alone on either coast).
* Similarity Weights
  + 5 points Unchanged
  + 4 points Reflected
  + 3 points Rotated
  + 2 points Scaled
  + 1 points Deleted
  + 0 points Shape Changed
* Describe-and-match method (p23 in Winston)
  + Identify an object by describing it and then searching for a matching description in a description library
  + 
  + Application: Feature-based object identification (eg: select electrical-box covers)
    - Feature extractor: measure simple characteristics - obtains a feature point in feature space
    - Feature evaluator

## 04: Generate & Test (30)

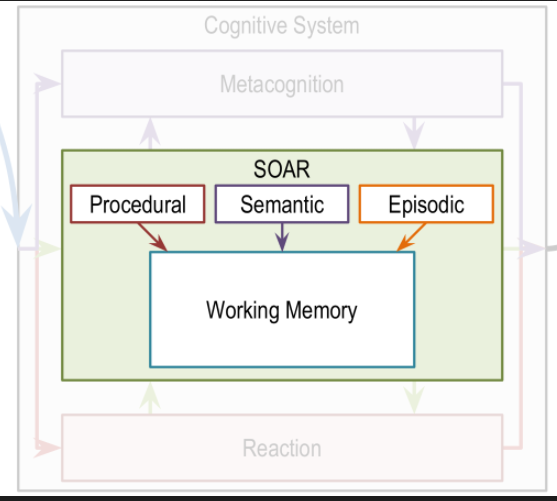
* Good Generators
  + Complete
    - Eventually produce all possible solutions
  + Efficient - Non redundant
    - Never compromise efficiency by proposing the same solution twice (or more than once)
  + Smart - informed
    - They use possibility limiting information, restricting the solutions that they propose accordingly
* To perform generate and test:
  + **While** a satisfactory solution is **not** found, or no more candidate solutions can be generated:
    - Generate a candidate solution
    - Test the candidate solution
  + IF an acceptable solution is found, announce it, otherwise announce failure
  + Pg 48 winston
  + 

## 05: Means-Ends Analysis (60)

* State spaces
  + Initial state
  + Goal State
  + Different in State Spaces
    - Like measuring lines in distance. Trying to pick a path by continuing the reduction of path - a notion of distance.
* Means-ends analysis
  + a process of searching for the means or steps to reduce differences between the current situation and the desired goal
  + For each operator that can be applied:
    - Apply the operator to the current state
    - Calculate difference between new state and goal state
  + Prefer state that minimizes distance between new state and goal state
* Performing Means-end Analysis (winston pg53)
  + Until the goal is reached or no more procedures are available,
    - Describe the current state, the goal state, and the difference between the two
    - Use difference between the current and goal state (possibly with the description of the current or goal state to select a promising procedure)
    - Use the promising procedure and update the current state
  + If the goal is reached announce success otherwise announce failure
* Problem reduction
  + An approach to problem solving that converts the problem into a number of sub-problems, each of which can be solved separately.
  + Hard problem → 3 easier problems
  + Goal trees (Winston p57)
    - Difficult goals → lower-level subgoals (goal reduction)
    - And goals: solved if all immediate subgoals are satisfied
    - Or goals: solved if any of their immediate subgoals are satisfied
    - Leaf goals: goals that are satisfied directly without reference to any subgoals
* Reference: <https://quizlet.com/362203592/05-means-ends-analysis-flash-cards/>

## 

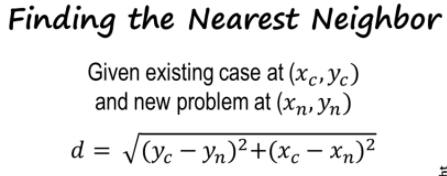
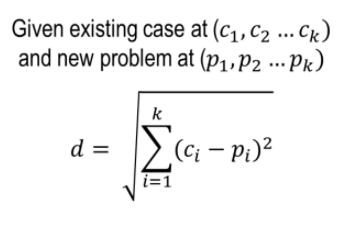
## 06: Production Systems (60)

* Definition - sets of computerized if-then statements that state the specific actions that will be taken under certain conditions
* Cognitive architectures
  + a framework for understanding human cognition as computer programs
  + Metacognition, Deliberation, Reaction
  + Deliberation
    - Reasoning
    - Learning
    - Memory
  + Levels of Deliberation
    - Knowledge Level
      * Task - doing something
    - Symbol Level
      * Algorithm - means end analysis/semantic network
    - Implementation Level
      * Hardware - brain/transistor
  + Assumptions of a Cognitive Architecture
    - Goal-oriented
    - Rich, complex environment
    - Significant knowledge
    - Symbols and abstractions
    - Flexible and function of the environment
    - Learning
  + Architecture + Content = Behavior
  + Function for cognitive architectures:
    - f: P\* → A
    - Percepts → Action
* Production systems
  + Deliberation Example: SOAR
    - 
    - Procedural
      * the gradual acquisition of skills as a result of practice, or "knowing how" to do things
    - Semantic
      * Semantic knowledge/memory
      * -finding words in general conversation
      * -navigating to familiar places
    - Episodic
      * Events based knowledge
      * Useful in breaking impasse (chunking)
    - Working Memory
  + Chunking
    - Changing production system to accommodate new situation
    - Learning technique - using passed history
* Reference: <https://quizlet.com/362268738/06-production-systems-flash-cards/>

## 07: Frames (45)

* Function of frames
  + Ashok ate a frog.
  + Ate
    - subject : Ashok
    - object : a frog
    - location :
    - time :
    - utensils :
    - object-alive : false
    - object-is : in-subject
    - subject-mood : happy
* Properties of frames
  + Represent stereotypes
  + Provide default values
  + Exhibit inheritance
* Relationship between frames and previous topics
  + Can be used to represent semantic network
* Frames for advanced sense-making
  + Today, an extremely serious earthquake of magnitude 8.5 hit Lower Slabovia, killing 25 people and causing $500 million in damage. The President of Lower Slabovia said that the hard-hit area near the Sadie Hawkins fault has been a danger zone for years.
  + Earthquake
    - day : Today
    - location : lower Slabovia
    - damage : $500 million
    - fatalities : 25
    - faultline : Sadie Hawkins
    - magnitude : 8.5
    - time :
    - type :
    - duration :

## 08: Learning by Recording Cases (30)

* Learning by recording cases
  + Block world
    - Given new problem a
    - Retrieve most similar prior problem, b, from memory
    - Apply b’s solution to problem
* Nearest neighbor method
  + 
* Cases in the real world
  + Streets by origin or destination
* k-Nearest Neighbor
* 
* Works for by checking closest for both “euclidean distance”

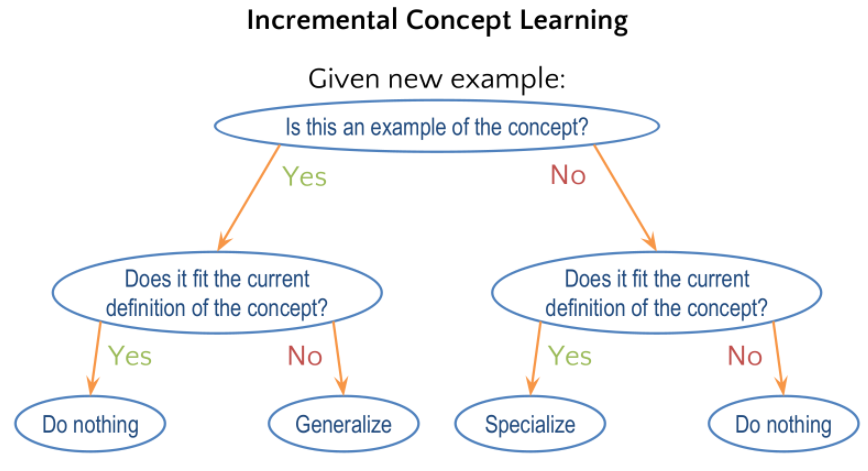
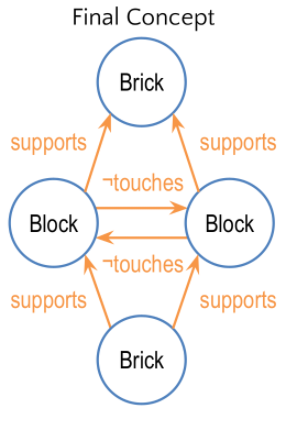
## 

## 09: Case-Based Reasoning (60)

* Need for case-based reasoning
  + New block - (pink block)
  + historical if-then-else cases are used to recognize patterns
* Case adaptation, evaluation, and storage
  + Retrieval
    - Retrieving a case from memory similar to the current problem
  + Adaptation
    - Adapting the solution to that case to fit the current problem
  + Evaluation
    - Evaluating how well the adapted solution addresses the current problem
  + Storage
    - Storing the new problem and solution as a case
* Assumptions of Case-Based Reasoning
  + Patterns exist in the world
  + Similar problems have similar solutions
* Case retrieval revisited
  + Binary tree vs list storage
  + Discrimination Tree
    - Recursive Path following for a case
* Advanced case-based reasoning
  + Evaluation found the solution failed; try adapting again
  + Evaluation found the solution failed; try retrieving a different solution
  + The retrieved solution could not be adapted; retrieve a different solution
  + Retrieved case perfectly matches current problem; no adaptation needed

## 

## 10: Incremental Concept Learning (60)

* Purpose of incremental concept learning
  + Learning is incremental - Learn from one example at a time. Humans (and cognitive agents) are not always given tons of examples right at the beginning - we get one at a time
  + Often examples we get are labeled - Teachers tell "positive example" or "negative" example (supervised learning)
  + Different from case-based reasoning - CBR had old examples that were stored in raw form in memory - In this case we abstract concepts from it
  + This method of incremental learning differs quite a bit from standard algorithms in machine learning Often in ML, the agent is given large number of examples to begin with (thousands, millions) - its easier to apply statistical methods - can find patterns with large set of examples
  + But if number of examples are small, and examples come one at a time (as in incremental learning) - it becomes harder to apply statistical methods to detect patterns in data - algorithm must make use of background knowledge to decide what and how to learn
  + 
  + Foo
    - Final concept
    - 
* Variabilization
  + A categorical variable is a value that assumes a limited and fixed number of possible values, allowing a data unit to be assigned to a broad category for classification
  + The important difference between a categorical variable and and nominal variable is that the categorical values have no relative relationship among each other, such a fork, knife and so on, whereas nominal variables may indicate the degree of something such as very sad, sad, neutral, happy, very happy.
  + <https://deepai.org/machine-learning-glossary-and-terms/categorical-variable>
* Specialization
  + Tighten the concept with more requirements or deletion
* Generalization
  + Widen the concept by adding links or enlarging range of values
  + Generalization to ignore features
* Heuristics for specialization and generalization
  + Drop-link heuristic (extra link not required)
    - link is not necessary to be a positive example of the concept
  + Require-link heuristic (must support)
    - link must be present to be a positive example of the concept
  + “enlarge-set” heuristic (brick/wedge → brick OR wedge)
    - multiple objects or links may fit one role in the concept
  + Forbid-link heuristic (forbidden) - not touches
    - link must be absent to be a positive example of the concept
  + “Climb-tree” heuristic - (brick or wedge → block) AKA generalize
    - generalize over multiple objects in the same role based on knowledge
  + Close-internal heuristic (range of values)
    - expand range of values to be a positive example of the concept
    - Suppose a child has only come across dogs that were small. Now child comes across large dogs - child expands range of size values that dog can be

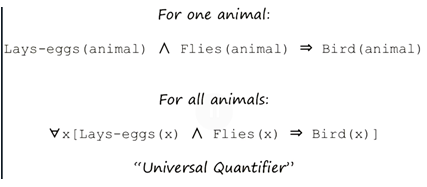
## Classification

## 11: Classification (45)

* Why classification
  + Dealing with cognitive system that has certain percepts coming in
  + Mapping sets of percepts in the world into equivalence classes so that we can act in the world in an efficient manner
  + Classification is done all the time
  + Ai agents need to do classification all the time too
  + N is the number of percepts
  + 2^N combinations of percepts available
  + One way of thinking intelligent agent is how can we map percepts into actions - Intelligence is about action selection
  + Reason classification is so ubiquitous b/c it allows us to select actions
* Equivalence classes
  + Works for concepts that have a formal nature
  + “necessary and sufficient conditions”
* Concept hierarchies/ Types of concepts
  + More formal - can define logical conditions that are necessary and sufficient for that concept
    - Axiomatic (math type concepts)
      * Circle - all points in a plane that are equidistant from a single point
    - Prototype (chair, stool)
      * Base concept defined by a typical example with overridable properties
      * “Classes” in programming are prototypical
      * Relationship of concepts and frames is quite close
  + Less Formal - hard to define necessary and sufficient conditions
    - Exemplar (Beauty, Freedom)
    - Don't even have typical conditions let alone necessary and sufficient conditions
* Top-down vs Bottom-up search
  + Tags: top down bottom up
  + Top/Down (Establish/Refine) method - is very well suited for situations where we know something about the top level concept and trying to establish next node
  + Bottom/Up (Identify/Abstract) We know something about the value of all the leaf nodes. Must make a prediction of root node

## 

## 12: Logic (90)

* Why do we need formal logic?
  + Soundness: Only valid conclusions can be proven.
  + Completeness: All valid conclusions can be proven.
  + Very formal language that allows us to make assertions about the world in a very precise way
  + It is important in its own right, but also forms basis of other topics like planning
* Two parts of AI agent:
  + 1) knowledge base Agent's knowledge about the world. Knowledge represented in form of sentence in the language of logic
  + 2) inferences engine - Apply rules of inference to knowledge the agent has
  + Certain situations where we may want AI agent to only generate provably correct situations
    - ○ Need 2 things to guarantee provably correct:
    - ■ Complete and correct knowledge base
    - ■ Rules of inference that will give guarantees of correctness of answer
* **Logic uses predicate** - A function that maps object arguments to true or false
  + Feathers(animal) → True/False based on animal
  + If Feathers(animal): Then Bird(animal)
    - Two predicates and one object
    - Capture relationship between two predicates (via if/then)
    - Implicated relationship
    - Read as "feathers animal implies bird animal" - or "if animal has feathers implies the animal is a bird"
* Formal notation
* Conjunctions, disjunctions, negations, implications
  + Symbols
  + Lays-eggs(animals) ^ flies(animals)
  + ^ is "AND" conjunction
  + : is "implication" or → is implication
  + > is "OR" disjunction
  + ! is "negation"
* Truth tables
  + Help find truth of complex sentences
  + These properties will allow us to rewrite the sentences in such a way that we can apply the rules of inferences
  + If we know truth value of various predicates in sentence, then we can compute truth value of sentence as a whole
* Rules of inference
  + Modus Ponens
    - S1 P implies Q (p → q )
    - S2 P
    - then we can therefore infer Q
  + Modus Tollens
    - S1 P implies Q (p → q )
    - S2 not Q (!Q)
    - then we can therefore infer not P (!P)
* Resolution theorem proving
  + **proof by refutation** Used when other methods (rules of inference, truth tables) are computationally infeasible
    - -must first convert to conjunctive normal form
      * S1 => knowledge
      * S2 => observation/percept
      * S3 => negation of what we're trying to prove
    - -if we can find a positive and negative literal, we can cancel them out.
    - -If we can cancel out all literals - we have a null condition, and S3 must be false, so what we're trying to prove must be true
    - **Always begin with negation of what you're trying to prove (and find a negation and resolve on that). Take remaining literals**
* 1st order Logic
  + - has variables/quantifiers
  + Universal Quantifier
    - Applies for all X (all animals) - allow for all values for the variable\*
    - 
  + Existential Quantifier: For at least some, at least one animal
  + -predicate logic
  + 0th order Logic - first order logic without variables or quantifiers
* Conjunctive Normal Form
  + 1 of 3 conditions for conjunctive normal form:
    - ■ 1) Literal with positive or negative (A , B)
    - ■ 2) Disjunctive literals (A or B)
    - ■ 3) Conjunction of disjunction of literals ( [A or B] -and- [C or D] )
* Horn Clause
  + a disjunction that contains at most one positive literal
  + Computational efficient of Resolution theorem proving arises because of Horn Clauses - This is where focus of attention comes from
  + Focus of attention. When problem space is very complex (# of sentences if very large and complex) - hard for agent to decide what to focus on. But because we have converted everything into conjunctive normal form - and because resolution theorem proving is making use of resolution - at any particular time the agent knows what to focus on.
  + **Always begin with negation of what you're trying to prove (and find a negation and resolve on that). Take remaining literals**
* Connection to Human Cognition
  + ● Logic provides a formal and precise way of reasoning (formal notation for expressing how intelligent agents reason - whether or not they use logical reasoning)
  + ● Even if behavior appears to be logical does not mean we used logic to solve (might encounter a new problem and use a previous case - behavior only appears to be logical
  + ● Logic we considered so far is deductive , however a lot of human reasoning is inductive or abductive
  + ○ Deductive (top-down - facts to facts) - reasoning logically from factual statements to reach a conclusion. (I'm A,B,C, therefore I can reach my goal D)
  + ○ Inductive (bottom-up - generalizing) - starts with question and works its way to theory
* Deduction is reasoning from causes to effects
* Abduction is effects to causes
  + ○ Going to doctor come with certain effects and doctor comes up with cause
    - ● Induction
  + ○ Given relationship of cause and effect for a sample, how do we generalize cause and effect for larger population