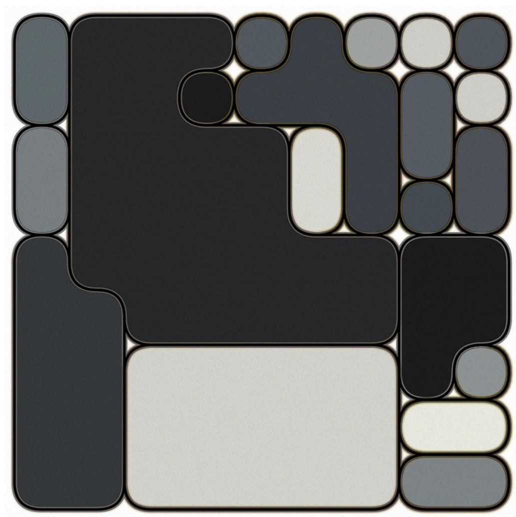
CCPS 721 Labs, Ilkka Kokkarinen

This document contains the lab problems for the course CCPS 721 Artificial Intelligence, taught by Ilkka Kokkarinen. These questions come from all over the place; some are taken directly from the course textbook, some are created by the instructor based on his lived experiences, and the rest are adapted from various classic AI textbooks and online exam banks of AI courses offered by the finest universities that we should aim to emulate.

There are ten labs in the course, starting the second week. Each lab takes one hour. The students should read through and think about the lab questions before coming to the lab session. The lab time is then spent with the TA leading the discussion about these questions with hopefully vigorous student participation.

Lab 1: Prolog I



Build it and they will come. To practice handling Prolog lists and recursion, in this exercise we will implement a whole bunch of list predicates that are already SWI-Prolog built-ins. **All these predicates must be written from scratch, so that you are not allowed to use any existing list predicates of SWI Prolog.** (Of course, after you implement a predicate on your own, you are allowed to use that predicate in implementing the latter predicates.) Name each predicate with the prefix `my_`, so that, for example, your version of the `sublist` predicate is named `my_sublist`. Make sure that your predicates are as generative and reversible as you are able to make them. The SWI-Prolog list predicates to implement in this exercise are:

1. `same_length/2`
2. `length/2`
3. `prefix/2`
4. `memberchk/2` (otherwise like `member/2`, but cuts after first success)

5. [nth1/2]
6. [delete/3]

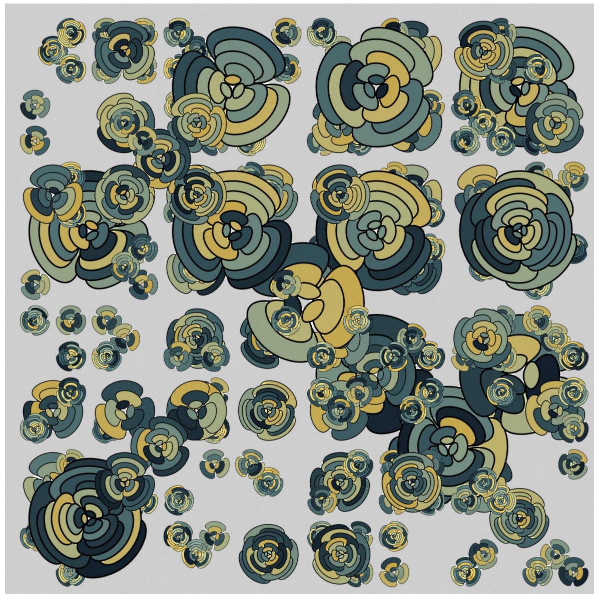
Build it and they will come II: Electric Boogaloo. Same question as the previous one, but now we implement some predicates that are not SWI-Prolog built-ins but instead defined in the *hprolog.pl* compatibility extension. The list predicates to implement in this exercise are:

1. `sublist/2`
2. `take/3`
3. `drop/3`
4. `bounded_sublist/3`
5. `membercheck_eq/2`
6. `between/3`

Input/output loop. Let us strengthen our confidence in the fact that Prolog is computationally universal by simulating a simple `while`-loop of imperative languages with it. Write a predicate `squares` that repeatedly asks the user to enter a term. If the term equals `stop`, the predicate terminates. If the term is an integer, write its square. Otherwise, write the message "Please enter an integer." You need to use system predicates `read` and `write` for the input/output.

Thick as thieves. Using predicates `steals(X, Y)`, `thief(X)` and `rich(X)` to mean "*x* steals from *y*", "*x* is a thief" and "*x* is rich", write the Prolog formulas to encode the following set of claims: (a) `robinhood` steals from the rich. (The natural language is a bit imprecise here: the intended meaning is "If you are rich, `robinhood` steals from you" as opposed to "If `robinhood` steals from you, you are rich", which is a completely different claim.) (b) The `sheriff` steals from the poor. (Ditto.) (c) Anyone who steals is a thief. Make a query `thief(robinhood), thief(sheriff).` trying to establish that both Robin Hood and the Sheriff are both thieves. What additional premises do you need to add to the knowledge base to make your query succeed?

Lab 2: Prolog II



Hailstones. Write a predicate `collatz(N, S)` that computes how many Collatz sequence steps (if current `N` is even, move to `N / 2`, otherwise move to `3 * N + 1`) are needed to reach the goal state 1 starting from `N`, and unifies `S` with the result.

Example query	**Expected solution**
`collatz(123, S).`	`S = 46`
`collatz(-3, S).`	`false`

Arithmetic. Write a predicate `digits(N, D)` that succeeds if `D` is the list of digits of the given positive integer `N`, in the same order that they occur in the number. (Hint: to extract the last digit of an integer `N`, use `N mod 10`, and to extract the rest of its digits, use `N div 10`.)

Example query	**Expected solution**
`digits(1234567, D).`	`D = [1, 2, 3, 4, 5, 6, 7]`
`digits(1234567,[_,_,_,X,_,_,_]).`	`X = 4`

Extra challenge: Use suitable metapredicates to make your predicate reversible so that `digits(N, [1,2,3])` succeeds with the solution `N = 123`.

Hard searching made easy. Write a predicate `subset_sum(L, S, G)` that succeeds if there exists some sublist `S` of the given list of integers `L` so that the elements of your chosen sublist `S` add up to exactly to the goal value `G`. (Hint: you either take the first element `X` into the subset, leaving you the new goal value `G-X`, or you don't take that element in your sublist, leaving you with the same goal `G`.)

Example query	**Expected solution**
`subset_sum([3, 6, -7, 10, 11, 16, -2], S, 30).`	Exactly four solutions
`subset_sum([2, 6, 9, 12, 14, 19, 20, 23, 27, 30], S, 100).`	Exactly 13 solutions

Generating randomness. Write a predicate `randomlist(N, Min, Max, L)` that unifies `L` with a list of `N` random integers so that each integer is between `Min` and `Max`, inclusive. (Find out in the SWI-Prolog system predicate documentation how to generate individual random numbers.)

List alternation. Write a predicate `zip(L1, L2, R)` that succeeds if `R` is a list constructed by combining the elements of lists `L1` and `L2`, alternating between the two lists. If one of the lists `L1` and `L2` is longer than the other, the extra elements should end up as they are to the end of the result list.

Example query	**Expected solution**
`zip([1, 2, 3], [4, 5, 6], R).`	`R = [1, 4, 2, 5, 3, 6]`
`zip([7, 1, 7], [5, 5, 5, 5, 5], R).`	`R = [7, 5, 1, 5, 7, 5, 5, 5]`
`zip(L1, L2, [1, 2, 3, 4, 5]).`	(there should be exactly six solutions)

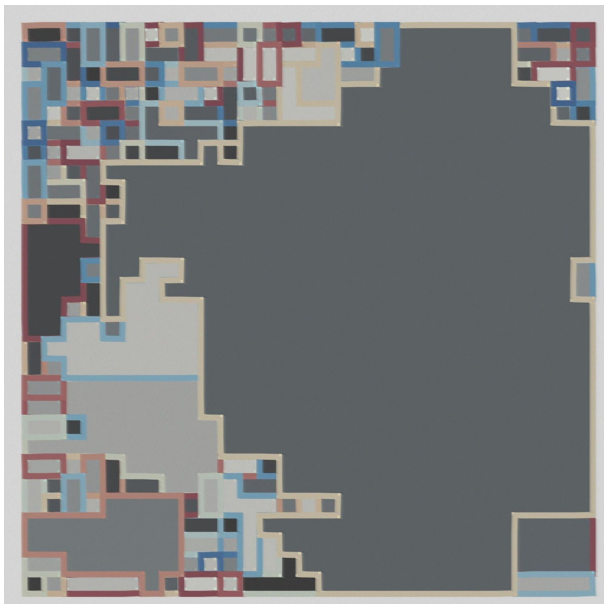
Accumulating values. Write a predicate `accum(L1, L2)` that succeeds if `L2` is the **accumulation** of the list `L1`, that is, has the same length as `L1`, and each element equals the sum of all elements of `L1` up to that position. (Hint: define an auxiliary version of this predicate `accum(L1, L2, A)` where the accumulator `A` is the sum of previously processed elements, and then define a startup rule `accum(L1, L2) :- accum(L1, L2, 0).`)

Example query	**Expected solution**
`accum([1,2,3,4], L).`	`L = [1, 3, 6, 10]`
`accum([3,-1,4,-1,5,-9,2,-6], L).`	`L = [3, 2, 6, 5, 10, 1, 3, -3]`

Bonus. Using metapredicates and multiple rules, make your predicate `accum` more flexible so that it can correctly handle the following query:

Example query	Expected solution
`accum([1, 2, X, 4], [1, Y, 6, 10]).`	`X = 3, Y = 3`

Lab 3: State space search



Classic puzzle. Farmer (*F*) has a wolf (*W*), goose (*G*) and bread (*B*) with him. He needs to take all three to the other side of the river in a small boat that can carry at most one of those things with him. The wolf cannot be left alone with the goose, and the goose cannot be left alone with the bread.

1. Draw the state space of this problem. Identify the start state and the goal state.
2. Which states would the breadth-first search and depth-first search algorithms examine looking for the solution path?

(Traditionally this puzzle is stated in terms of missionaries and cannibals, but… well, you know.)

Popping those pills like they were candy. Pacman moves around in a two-dimensional $n*n$ grid of tiles. Each tile is either a floor or an impenetrable wall. Some red and blue pellets, exactly k of each kind, have been sprinkled around on the floor tiles. In each step, Pacman can move one tile up, down, left or right into a floor tile, automatically eating the pellet on that tile if there is one. The aim is to eat **at least one red pellet and at least one blue pellet**, and make the smallest number of moves to achieve this.

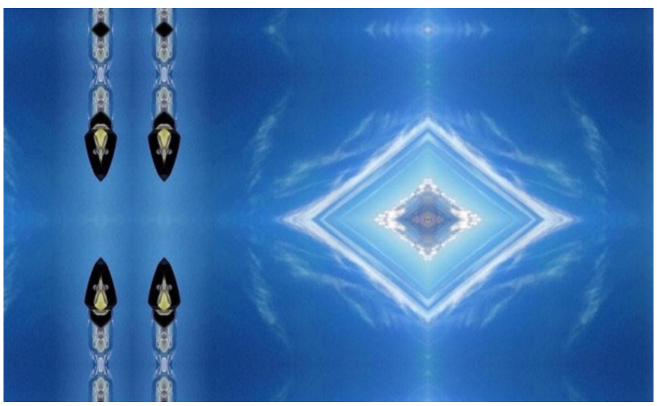
1. Give a state space formulation for this problem. How many different states are there? Assuming that Pacman starts at tile (x, y), what is the initial state? What are the goal states of the search?

2. For the purposes of the A* search, which of the following heuristics are admissible and consistent? (a) The total number of pellets remaining (b) The Manhattan distance to the pellet closest to Pacman (c) The Manhattan distance of the pellets that are furthest apart from each other (d) The Manhattan distance between the pair of red and blue pellets that are closest to each other.

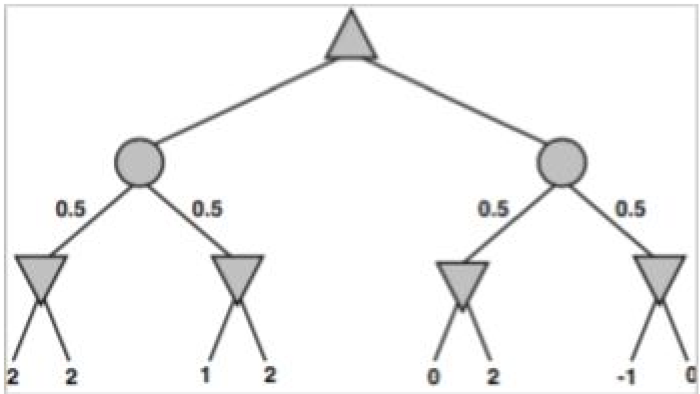
Algorithmic equivalences. True or false? Explain.

1. Breadth-first search is a special case of uniform cost search.
2. Depth-first search always expands at least as many nodes as A* search with an admissible heuristic function.
3. Depth-first search is a special case of uniform cost search.
4. Suppose that we had a heuristic function h that was exact, so that for each state s, $h(s)$ would equal exactly the length of the shortest path to the goal. Aided with such an *oracle*, the A* algorithm would expand only the nodes that lie on the shortest path from start state to goal, and thus run in time linear with respect to the path length.
5. If h_1 and h_2 are two admissible and consistent heuristics, their maximum and average are also admissible and consistent for that problem.
6. Bidirectional breadth-first search is always at least as fast (in the sense of never expanding more nodes) than ordinary breadth-first search, assuming that $B \geq 2$ and $L \geq 4$.
7. Adding the same positive constant value to the cost of every transition cannot change the path returned by uniform cost search.

Lab 4: Constrained and adversarial search

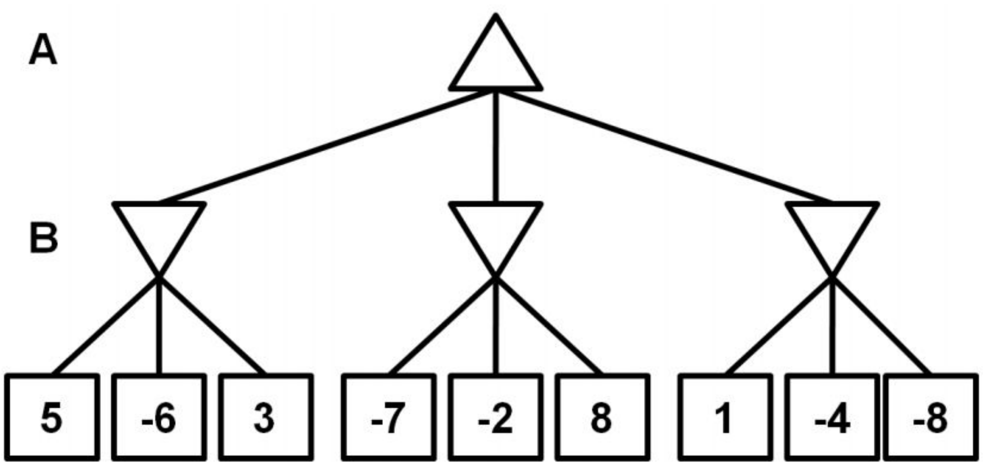


Minimax tree evaluation. [AIMA 5.16] Consider the following complete game tree for a toy game where *Max* moves first, then a fair coin is tossed to move either left or right, and then *Min* makes the last move.



1. Compute the values of the internal nodes bottom up.
2. Given the values of the first six leaves, do you actually need to know the values of the last two leaves on the right?
3. Suppose that all leaf node values are known to be in the range [-2, 2], inclusive. After the first two leaf nodes are evaluated, what is the known value range for the left chance node?
4. Circle all the nodes that need not be evaluated under the range assumption [-2, 2] for leaf nodes.

Dominating alphas. Consider this game tree for a two-player zero-sum game, with A maximizing:



Assuming the minimax search expands the children of each node left to right, which terminal nodes will be pruned using alpha-beta pruning?

Glass bead head game. After [Haugeland] (a) If you tripled the size of the soccer ball, how would this affect the game? If you tripled the size of your chess pieces and the board, how would this affect the game? (b) If you made your golf balls of granite, how would this affect the game? If you made your chess pieces of granite, how would this affect the game? (c) Why is the game of poker typically played with discrete chips instead of carefully measuring sugar or water (or some other amply available powder or liquid) to denote the bet sizes? (d) Is golf or soccer played on computers still real golf or soccer? Is chess or poker played on a computer still real chess or poker? (e) What is the fundamental reason why computers fare so much better at playing chess or poker than they do in playing soccer or golf? (f) In general, what kind of things can computers do better than humans?

Omniscience and its limits. True or false? Explain why, or why not.

1. When using alpha-beta pruning, the computational time savings are independent of the order in which the successor states are expanded.
2. In a fully observable, turn-taking, zero-sum game between two rational players, it does not help the first player to know how the second player would respond to each of the possible first moves.
3. A perfectly rational backgammon player never loses.

Order in the ballcourt. MAX and MIN are playing a turn-taking complete information zero sum game with a finite number of possible moves. MAX calculates the minimax value of the root to be *M*. You may assume that each player has at least two possible moves at every turn. You may also assume that in the game tree, each sequence of moves will always lead to a different final score (that is, no two sequences of moves can produce the same final score). Identify the claim that is **false**.

(a) Assume that MIN is playing suboptimally, following a policy that chooses a non-optimal move in some states, but MAX does not know that MIN is doing this, so MAX will always make the optimal move for himself. The final outcome of the game can be better for MAX than *M*.
(b) Assume that MIN is playing totally randomly, and MAX knows this. There exists a policy for MAX such that MAX can guarantee a better outcome than *M*.
(c) Assume that MAX knows that MIN is playing suboptimally at all states (that is, never makes his best move in any state), and furthermore, MAX knows which move MIN will make in each state. There exists a policy for MAX such that MAX can guarantee a better outcome than *M*.
(d) Assume that MIN always chooses his worst move, but MAX does not know this. If MAX always makes the worst possible move (that is, lowest minimax value), the result can be better for MAX than it would be if MAX always made his best possible move (that is, highest value).

ADD THE ODD DOG GOD. You are given a dictionary of 3-letter words. Consider the problem of crossword generation, filling a 3*3 grid with letters so that each row and each column is a word from this dictionary.

1. Express this problem as a constraint satisfaction problem. Do you prefer having nine variables (one variable per grid tile) or six variables (one variable per each row and column)? What are the sets of possible values of each variable in your chosen formulation?
2. For your chosen formulation, what does your constraint graph look like? How would you prune the sets of possible values to achieve arc consistency?

Lab 5: Propositional Logic

As above, so below. According to legend, one mathematics multiple choice exam contained the following question:

> Which answer in this list is the correct answer to *this* question?
>
> 1. All of the below.
> 2. None of the below.
> 3. All of the above.
> 4. One of the above.
> 5. None of the above.
> 6. None of the above.

To clear this thicket, let the proposition X_i stand for the claim "The answer i is true." Convert each of the six answers to a propositional logic formula that encodes the claim that the answer makes about the other answers. If you have the energy, feed these formulas into some propositional logic solver (or solve them by hand either by inference rules, or truth table enumeration if you feel like exercising through $2^6 = 64$ truth value combinations) to solve for the singular truth value assignment that makes all six formulas simultaneously true.

Truths about truths. Identify the claim about logical entailment and inference that is **true**.

(a) If a sound and complete inference mechanism for a bivalent logic produces formula φ from the knowledge base KB, then $\neg\varphi$ is false in every world.
(b) It is possible for some knowledge base KB to entail both formulas φ and $\neg\varphi$ for some φ.
(c) If a sound inference mechanism cannot produce formula φ from the knowledge base KB, then $\neg\varphi$ is true in every world where KB is true.

(d) If formulas φ_1 and φ_2 are logically equivalent, a sound inference mechanism can produce φ_1 from the knowledge base KB if and only if it can produce φ_2 from the same KB.

Let George do it. [AIMA 7.4] Which of the following claims about propositional logic are correct?

1. $False \vDash True$
2. $True \vDash False$
3. $(A \land B) \vDash (A \Leftrightarrow B)$
4. $A \Leftrightarrow B \vDash A \lor B$
5. $A \Leftrightarrow B \vDash \neg A \lor B$
6. $(A \land B) \Rightarrow C \vDash (A \Rightarrow C) \lor (B \Rightarrow C)$
7. $(A \lor B) \land (\neg A \Rightarrow B)$ is satisfiable.
8. $(A \Leftrightarrow B) \land (\neg A \lor B)$ is satisfiable.

An equal valence. Which formula is the conjunctive normal form of $(P \Rightarrow (Q \Leftrightarrow R))$?

(a) $(\neg P \lor \neg Q \lor \neg R) \land (P \lor \neg Q \lor \neg R)$
(b) $(\neg P \lor Q \lor R) \land (\neg P \lor \neg Q \lor \neg R)$
(c) $(\neg P \lor \neg Q \lor R) \land (\neg P \lor Q \lor \neg R)$
(d) $(P \lor Q \lor R) \land (\neg P \lor \neg Q \lor \neg R)$

Son, I can already see that you are a born winner. "Heads I win, tails you lose." Using four propositions `Heads`, `Tails`, `IWin` and `YouLose`, encode this famous carnival barker ejaculation as propositional logic formulas in conjunctive normal form. Use resolution refutation to try to prove that `YouLose`. What other non-logical axioms (that might seem too self-evident to even mention despite the fact that they are necessary) do you need to add to the knowledge base for the proof to go through?

Either you do it, or you cry and do it. (a) Surely all you millennials, unfortunate to have been born right into the era of "Average is Over" but too early before the "Age of Em", must be familiar with the Catch-22 principle "You need experience to get hired for a job, but to gain experience, you need to have a job." Express this idea in propositional logic, and then prove that you won't get a job. (b) Morton's fork, named after the 15th century tax collector John Morton, is the observation that if a man is living modestly, then surely he must be accumulating savings and can therefore afford to be taxed heavily; and if a man is not living modestly, then surely he must have excess wealth to burn and can therefore afford to be taxed heavily. Express this concept in propositional logic, and prove that the man will be taxed heavily no matter what.

They both reached for the gun. In an olde-timey murder trial from the swinging jazz age, a lovable rascal defense lawyer (in vein of Billy Flynn in *Chicago*) argues before the jury that his client "Tony the Torpedo" was not even in the same town as the victim "Sammy the Squealer" at the time of shooting, so his client must be found innocent. However, should the jury decide that his client was indeed there and did shoot the victim, fair enough, he will then present more than enough evidence to prove that the shooting was a clear case of justified self-defense! Were you sitting in the jury box, and assuming that you accepted all the rest of the reasoning to be undeniably true, would you accept this line of argumentation in principle? If you do not (perhaps because you unknowingly subscribe to the intuitionistic logic school of thought), explain what exactly do you think is wrong with this type of argumentation, especially when compared to the answers of the previous Catch-22 and Morton's fork questions.

Guided by semantics. [AIMA 7.10] Decide whether each of the following statements is valid, unsatisfiable, or neither.

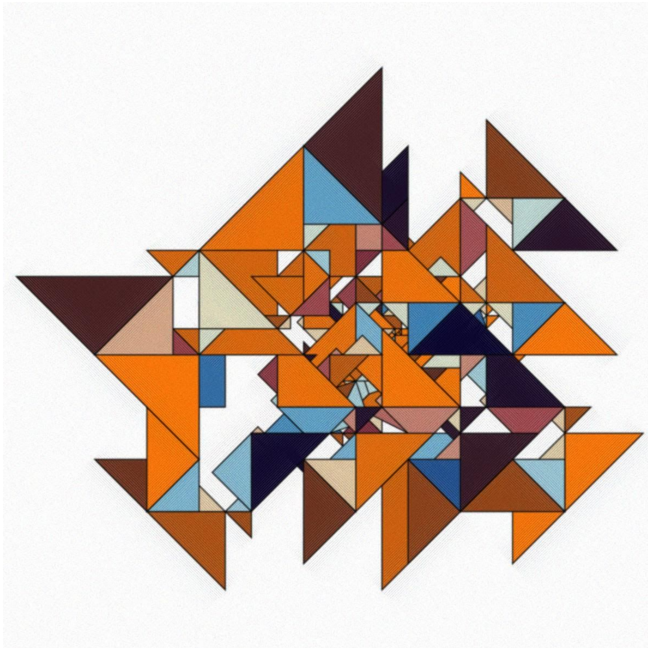
1. $Smoke \Rightarrow Smoke$
2. $Smoke \Rightarrow Fire$
3. $(Smoke \Rightarrow Fire) \Rightarrow (\neg Smoke \Rightarrow \neg Fire)$
4. $Smoke \lor Fire \lor \neg Fire$
5. $((Smoke \land Heat) \Rightarrow Fire) \Rightarrow ((Smoke \Rightarrow Fire) \lor (Heat \Rightarrow Fire))$
6. $(Smoke \Rightarrow Fire) \Rightarrow ((Smoke \land Heat) \Rightarrow Fire)$
7. $Big \lor Dumb \lor (Big \Rightarrow Dumb)$

The Kaboom Kids strike back. [AIMA 7.22] Assuming a fixed-size $N * M$ gamefield, the game of Minesweeper can be modelled with $N * M$ distinct propositions $X_{i,j}$ to denote that there is a mine in the cell in the row i and column j. The rows are numbered from 1 to N, and the columns are numbered from 1 to M.

1. Write a propositional logic formula to assert that there are exactly two mines in the neighbourhood of the top left corner cell (1, 1).
2. Generalize the assertion of the previous question to say that there are exactly k mines in the neighbourhood of the given cell.
3. Explain precisely how an agent could use DPLL to prove that some cell does (or does not) contain a mine.
4. Give an example of a minesweeper configuration where finding out whether some cell contains a mine or not would instantly tell you whether some other cell on the far opposite edge of the board whose value is currently unknown contains a mine. (Hint: use a $1*N$ board.)

The truest sentence you know. (a) Analyze the work "One True Sentence" by conceptual artist Henry Flynt from a cold logical point of view. Is the set of sentences depicted in the work internally logically consistent? Valid? Neither? (b) A friend tells you that it is raining outside, but that he also does not believe that it is raining. Is your "fair-weather friend" necessarily being logically inconsistent? (You don't need to look outside the window to answer this question.)

Lab 6: Predicate Logic



One born every minute. [Papadimitriou] (a) Let the predicate `canfool(x, t)` denote that person `x` can be fooled at time `t`. Express in predicate logic Abraham Lincoln's famous dictum "You can fool some people all the time, and you can fool everybody some time, but you can't fool everybody all the time." (b) Convert your formula to the conjunctive normal form, using `goober` as the Skolemization constant for that person who can be fooled all the time.

I'm okay, you're okay. Using the first order logic predicates $knows(x, y)$ (to mean "person x knows person y") and $happy(x)$ (to mean "person x is happy"), which one of the following formulas encodes the claim "Every happy person knows somebody who is not happy" ? How would you express the other three formulas in natural language?

(a) $\forall x : (happy(x) \Rightarrow \exists\ y : (knows(x, y) \Rightarrow happy(y)))$
(b) $\forall x : \exists\ y : (happy(x) \land knows(x, y)) \Rightarrow \neg happy(y)$
(c) $\exists\ x : \forall\ y : happy(x) \Rightarrow knows(x, y)$
(d) $\forall x : \exists\ y : \neg happy(x) \lor (knows(x, y) \land \neg happy(y))$

Let him who is my brother cast the first qualifier. Let the predicate $brother(x, y)$ mean that y is brother of x. Which one of the following formulas encodes the claim "John has exactly one brother" ?

(a) $\exists x\!: \exists y\!: brother(John, x) \land brother(John, y) \land x = y$
(b) $\exists x\!: brother(John, x) \Rightarrow \forall y\!: (brother(John, y) \land x = y)$
(c) $\exists x\!: brother(John, x) \Rightarrow \forall y\!: (brother(John, y) \Rightarrow x = y)$

(d) $\exists x$: $brother(John, x) \land \forall y$: $(brother(John, y) \Rightarrow x = y)$

Existenz. The existential operator $\exists$ does not say anything about how many objects exist that satisfy the given property, it just says that there is at least one. However, clever techniques allow us to be more specific. Express the following claims using predicate logic and the previous predicate *happy*(x).

(a) There exists exactly one happy person.
(b) There exist exactly two happy people. (Generalize this technique in your mind for any n.)
(c) There exist infinitely many happy people. (To be able to say this we have to assume that all people are standing in a line, and the additional predicate *before*(x, y) is true if person x stands in line before person y, satisfying the axiom $\forall x$: $\forall y$: $before(x, y) \Leftrightarrow x \neq y \land \neg before(y, x)$.

Your pachyderms are showing. [Nilsson] Sam, Clyde and Oscar are elephants. (1) Sam is pink. (2) Clyde is gray and likes Oscar. (3) Oscar is either pink or gray (but not both), and likes Sam. Express these claims as clauses of predicate logic using suitable predicates and a knowledge base of formula. Then use resolution refutation to express and prove the claim that some gray elephant likes some pink elephant.

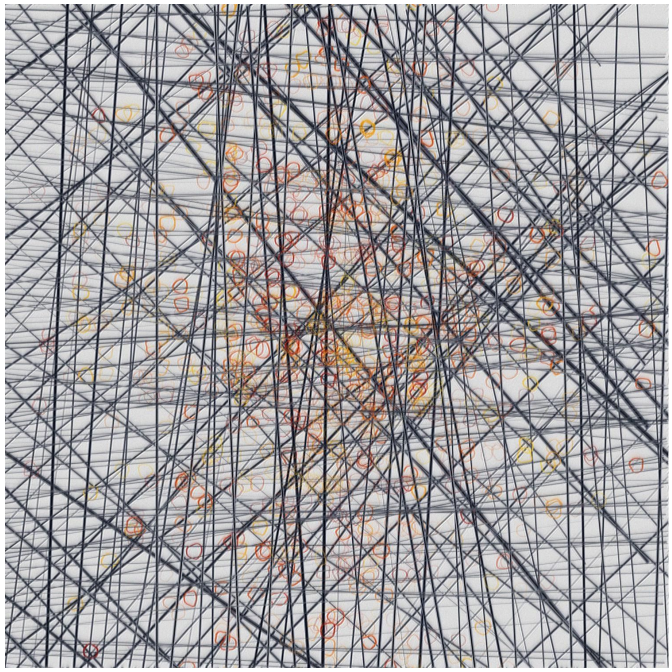
Not our Russell's razor. Let the predicate `shaves(x, y)` mean that `x` shaves `y`. Express in first order logic that the `barber` shaves precisely those people who don't shave themselves. Convert your formula to the standard clause form. Prove that the `barber` shaves himself. Next, prove that the barber does not shave himself. What does this contradiction prove?

Our Aristotle. [AIMA 9.23] In a fair and a just world, the premise "Horses are animals" seemingly ought to entail "The head of a horse is the head of an animal". Demonstrate that this inference is valid by carrying out the following steps:

1. Translate the premise and the conclusion into first-order logic formulas. Use three predicates *HeadOf*(h, x) meaning "h is the head of x", *Horse*(x) meaning "x is a horse" and *Animal*(x) meaning "x is an animal".
2. Negate the conclusion and convert the premise and the negated conclusion into conjunctive normal form.
3. Use resolution to show that the conclusion follows from the premise.

Contratemporal complexities. [Goldberg] During his life, Mozart visited Vienna a total of three times. Mozart died in Vienna. During which of his three visits to Vienna did Mozart die? Prove this rigorously without the slightest waving of hands by defining suitable predicates and sufficient logical formulas to reason about these predicates to allow the resolution refutation proof go through. (Note that the impossibility of time travel to the past, something that we tend to intuitively take for granted, is *not* a logical tautology, and must therefore be somehow expressed in your axioms! The whole point of this exercise is to show that time and space, as simple as they seem for us humans, are surprisingly difficult to model rigorously using logic.)

Lab 7: Bayesian Probability I



Opa papa. [from AIMA 13.21, but this old chestnut can be found by some divine decree in every AI textbook ever written] Coming home late at night from a raucous party, you become a witness to a hit-and run accident involving a taxi in Athens. 90% of taxis in Athens are green and 10% are blue. Under the dim lighting conditions and your state of inebriation, discrimination between blue and green taxis is 75% reliable (that is, if you see a taxi, you can identify its true color correctly with 75% probability). What is the probability that the fleeing taxi really is blue, assuming that the taxi appeared blue to your eyes that night? (Distinguish carefully between the propositions “The taxi is blue” and “The taxi appears blue”.)

Divide by two and add on seven more. You millennials seem to like to “chillax” with the sounds of your favourite bands *One Direction* and *Nickelback*, so to reach you better across the generational chasm, here is one problem involving your beloved crooners. In one third-year programming course, 2/3 of students are science majors, and the other 1/3 are engineering majors. 75% of science majors enjoy the melodic stylings of One Direction, whereas 25% of them torment their fellow man with the gutter noises of Nickelback. Engineering majors are a bit rougher bunch, so 25% of them listen to One Direction and 75% prefer Nickelback. If the professor sees an unknown student in this course listening to Nickelback, what is the probability that this student is an engineering major?

Rather this than the one about the old maid and her two beds. In a lame joke that the instructor still remembers reading back in his childhood (back when the world was in many ways very different from how things are today) during his few breaks from walking ten miles to school uphill both ways, some smart alec who was worried about air travel because of the possibility of a terrorist bomb on the plane achieved the peace of mind for travel by sneaking a real (but made safely inactive) bomb on the plane on his carry-on, optimistically reasoning that the probability of there being two bombs simultaneously on the same plane is too vanishingly small for even a neurotic freak like him to worry about. Ignoring all the far more serious levels of wrong in this man's thinking, explain his simple probabilistic reasoning error.

Hand waving a longer life span. (a) A statistical study once found that conductors of symphony orchestras live, on average, ten years longer than the rest of the population. Could the mild exercise of waving the baton somehow extend life that much, or is there some more mundane explanation for this massive statistical discrepancy? (b) In another bad joke from the nineties, one decrepit old man, scared of the cold clammy hand of death heavily weighing on his shoulder, intentionally got himself infected with HIV because he had read in the newspaper that the advances of modern medicine now allow HIV-positive people stay alive even for several decades after infection. Again, ignoring the more serious psychological and socio-political problems in this tale, what is the elementary probabilistic reasoning error committed by this desperate man? (Answer this properly involving probability formulas without any "Everyone can see that..." hand waving.)

Live while we're young. Let O and N be the propositions that you, an exhausted CS major looking to relax after all the hard work in this AI course, try to get tickets to the upcoming concerts of your two favourite bands, **O**ne Direction and **N**ickelback. You estimate that your probability of successfully getting a ticket to each concert (all your fellow millennials are also trying to snatch tickets) is $P(O) = 0.5$ and $P(N) = 0.3$, and that $P(O \wedge N) = 0.2$. (a) Assuming that you manage to get a One Direction ticket, what is the probability that you will also get a Nickelback ticket? (b) Is the event "You get a One Direction ticket" independent of the event "You get a Nickelback ticket" ?

To attract one means to repel its opposite. Identify the one false claim about conditional probabilities. (It is given that $P(E) > 0$ and $P(\neg E) > 0$.)

(a) If $P(A \mid E) > P(A \mid \neg E)$, then $P(E \mid A) > P(\neg E \mid A)$.
(b) If $P(A \mid E) > P(B \mid E)$ and $P(A \mid \neg E) > P(B \mid \neg E)$, then $P(A) > P(B)$.
(c) If $P(A \mid E) < P(A)$, then $P(A \mid \neg E) > P(A)$.
(d) If $P(A \mid E) = 0$, then $P(E \mid A) = 0$.

Lab 8: Bayesian Probability II

Putting your money where your mouth is. (a) Suppose P(A) = 0.1 and P(B) = 0.3. What is the range of the possible values for P($A \wedge B$) and P($A \vee B$)? (b) How about if P(A) = 0.7 and P(B) = 0.8 ? (c) Joe Palooka, a happy-go-lucky sort of hearty fella who considers himself a free man and singular individualist whose life and mind are not constrained by anything so pedestrian as the trifle axioms of probability, chooses to believe that P(A) = 0.1, P(B) = 0.4 and P($A \wedge B$) = 0.3. Construct a Dutch book for a series of bets so that each individual bet has either zero or positive expected value for Joe, but for all four possible truth value combinations of A and B, these bets taken together give him a guaranteed negative net result.

Two sides of the same coin. In the following sort of problems, it is assumed that boys and girls are born at all times with 50-50 probability, independently of any previously conceived children. Answer the following questions properly using the Bayes theorem instead of hand waving.

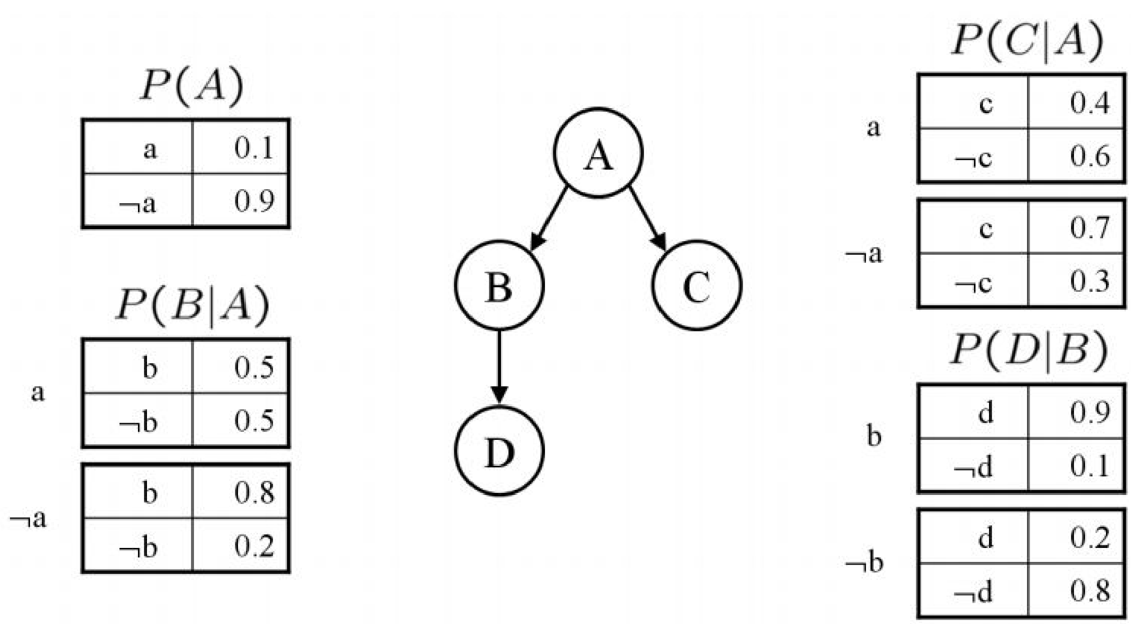
(a) I have two children. What is the probability that the older one is a boy?
(b) I have two children. Given that at least one of them is a boy, what is the probability that I have two boys?

(c) I have two children. Given that the older one is a boy, what is the probability that I have two boys?
(d) I have two children. Given that the older one is a boy born on Tuesday, what is the probability that I have two boys?

Even if the heavens opened. (a) Prove from the axioms of probability, that if P(X) = 0, then no new evidence can emerge that can change this probability. That is, P(X | E) = 0 for any E for which P(E) > 0. Prove similarly that if P(X) = 1, then also P(X | E) = 1 for any new evidence E that may come. (b) If P(X) is strictly between 0 and 1, then new evidence could in principle always swing it arbitrarily far to either direction. Suppose that some X we have P(X) = 0.01, but the appearance of evidence E would make X massively more likely so that P(X | E) = 0.99. Give an approximate upper bound for P(E) to show that the probability of appearance of such striking evidence E that would drastically swing other probabilities is rather small. (Hint: for any formulas X and E, it must necessarily be that P(E) ≥ P($X \land E$).)

Sure, that's like the facts or something, but it still does not *always* correlate. Think up a real-world example of three propositions A, B and C so that P(A) < P(A | B) and P(B) < P(B | C), but P(A) > P(A | C). That is, B increases the probability of A, and C increases probability of B, but C decreases the probability of A. (This example of **non-transitivity of attraction** shows that uncertain knowledge can't be **localized** the same way as in propositional or predicate logic reasoning, but all knowledge must always be considered clumped together in a Bayesian network fashion, unless very strong independence assumptions can be made to simplify the Bayesian network.)

Explain that away, boyo! Consider the following belief network for propositions A, B, C and D.



(a) How much is P($B \land C$ | A) ?
(b) How much is P(D | A) ?

(c) How much is P(A | D) ?

Mens rea in corpore sano. (This question dates from before exact DNA testing. Again, no energetic hand waving or turn-off-the-brain moralistic buzzwords allowed.) (a) During the commission of a murder, the killer left some of his own blood at the murder scene. The defense attorney argues that since the lab test showed that the defendant has a different blood type than the blood found on the crime scene, his client is clearly innocent and should be set free. Do you agree with this argument? Why? (b) Suppose the lab trial finds that the blood types match, and that particular rare blood type is found only in 1% of the relevant population. The prosecutor argues that therefore there is only a 1% chance that the defendant is innocent. Do you agree with this argument? Why?

Lab 9: Decision Theory I

Living for tomorrow. In the lectures, an environment is defined to be *Markovian* if the probabilities of future states are conditionally independent of the previous states, given the current state. Is the environment still Markovian if the future probabilities are conditionally independent of the states $k + 1$ or more steps in the past, given the most recent k states?

Draw, pardner! Suppose you are holding active cards `[J, 7, 2]` and some useless fourth card, and it is now one of the drawing rounds. (a) If you draw one card, discarding your useless fourth card, what is your probability of making some badugi? (b) If you draw two cards, discarding also your active jack, what is your probability of making a three-card badugi better than your jack-high? What is your probability of making some four-card badugi in this one draw? (c) Holding this hand on the last drawing round, if your opponent also draws one card, what should you do? (d) Holding this hand on the last drawing round, if your opponent stands pat, having drawn cards in the previous rounds and bet in the previous betting round, to represent a made badugi, what should you do?

Utilities trading. [AIMA 16.8] Tickets to the lottery cost \$1. There are two possible prizes: \$10 with probability 1/50, and \$1,000,000 with probability 1/2,000,000. What is the expected monetary value of such a ticket? Ignoring the value of the thrill of gambling, what do we need to assume about the utilities of money for it to be rational to purchase a ticket? Be precise: show an actual equation with utilities. To make the calculations easier (remember that you can always shift and scale any consistent utility function to anchor any particular state to 0 and any other state to any fixed utility value of your choosing), you can assume that $U(k) = 0$ and $10\ U(k + 1) = U(k + 10)$, where k is your current amount of money, meaning that the loss of \$$k$ is significant to make a difference to your current utility.

Protecting the gift. [After Gavin de Becker] Let us assume that one in ten thousand people is a child molester who will, if given a reasonable chance to get away with it, smooth talk, snatch and molest a child. Most adults will not approach a small child, but a child molester will do so if he is the only adult around. Suppose that a small child gets lost. (a) If the child goes to the nearest adult to ask for help, what is the probability that this adult is a child molester? (b) If the child waits for the first adult to come to him, what is the probability that this adult is a child molester? (c) Given the above probabilities, what simple policy should you teach a small child to follow should he get lost? (d) Could you refine this policy in choosing which adult to approach to greatly improve the kid's chances even further, while keeping the policy so simple that a small child in distress is still able to understand and follow it?

Casinos hate this Mississauga man! In a cash game of Texas Hold'Em poker played with no-limit betting (which, by the way, would be much more accurately called "stack limit betting", unlike in all those old movies where the rich villain makes a huge raise pulling a wad of money out of his pocket big enough that the underdog hero can't afford to match) between n players at the table, one of the players immediately goes all in every hand without even looking at his cards. He claims that such kamikaze juggernaut strategy maximizes the number of pots that he wins. He not only wins big in every pot where he legitimately ends up with the best hand (which happens with probability $1/n$), but he also wins many other pots where the opponent hands that would have ended up winners in the final showdown before that. As a result, he will win more pots than the average player at the table. (Being all in from the beginning, he has no decisions left to make, and thus can't be bluffed or otherwise pushed off the pot, but will see every pot to the end.) Surely the game of Texas Hold'Em cannot possibly be this trivial to beat, so what is the fatal flaw in this argument?

Lab 10: Decision Theory II

Nothing beats the trusty old rock. Let us modify the game of rock-paper-scissors so that the loser of each throw pays the winner \$1, except that whenever you lose to paper by playing rock, you have to pay \$2. Write out the normal form and the indifference equations, and compute the Nash equilibrium mixed strategy for this variation. Since this variation is still symmetric, it is enough to do the calculations for one player. (Follow-up question: What would happen to this strategy if you had to pay a thousand dollars every time that you lose by playing rock?)

Snap off those bluffs! In the game of heads up badugi with ante of 1 and 2-2-4-8 limit betting, you are dealt the best three-card badugi *A23*. Even if unimproved over the three draws, your hand will beat anybody who doesn't catch a four-card badugi. Your hand has three good draws to a four-card badugi, most of which are near cinches in the heads-up game if they hit. The rest of the play of the hand now goes the following way:

- With three draws remaining, you bet, and the opponent calls. Pot is now 6 chips. Both players draw one. You miss.
- With two draws remaining, you bet, and the opponent calls. Pot is now 10 chips. Both players draw one. You miss again.
- With one draw remaining, you bet, and the opponent calls. Pot is now 18 chips. Both players draw one. You miss again. The three-card badugi A23 is your final hand.
- With zero draws remaining, you check. Your opponent gives you a smirk, mentions some duke that he knows, and bets. Pot is now 26 chips, 8 to call. With what probability should you call this bet?

Your answer can't possibly be either of the pure strategies 0 or 1, otherwise you would be trivial to exploit at the poker table. Instead, you must call with probability q and fold with probability $1 - q$. How does your answer depend on the tendencies of your opponent, and how would you model these tendencies as probabilities in these calculations? Remember the opponent's probability for catching a four-card badugi on his last draw is approximately (why?) 10/45. For simplicity, let's assume that the opponent will bet with every made badugi, and if he misses, he tries a bluff with probability p, and checks the hand otherwise. (Start by computing this p so that your calling and folding have equal expected value.)

Back up your words. [AIMA 17.10] Consider an undiscounted Markov Decision Problem having three states 1, 2 and 3, with rewards -1, -2, and 0, respectively. State 3 is the terminal state. In states 1 and 2, there are two possible actions *a* and *b*.

- In states 1 and 2, action *a* moves the agent to the other state with probability 0.8, and stays put with probability 0.2.
- In states 1 and 2, action *b* moves the agent to state 3 with probability 0.1, and stays put with probability 0.9.

(a) Apply policy iteration, showing each step in full, to determine the optimal policy for the values of the states 1 and 2. Assume that the initial policy has action *b* for both states. (b) Write out the Bellman equations for the values of the states 1, 2 and 3. If you feel like it (or happen to have some symbolic algebra solver available), solve the state values.

The value of nothing. Identify the claim about multistage decisions that is **false**.

(a) Given the same training set of state sequences and their observed rewards, **Monte Carlo** and **Temporal Difference** learning methods will always converge to the same state values.
(b) If the state space with n states and m possible actions for each state has no loops, then given its goal state values, the values of the other states can be computed exactly in time $O(nm)$.

(c) If the transition probabilities are known, knowing the Q-values of each (*state*, *action*) pair allows you to exactly calculate the values of each state.

(d) If the transition probabilities are known, knowing the state values allows you to exactly calculate the Q-values of each (*state*, *action*) pair.