# 1 CSP as IP

Alice, Bob, and Charles want to study in Gates, which has 9 floors. To maximize productivity, we need to assign each of them to a separate floor without violating the following constraints:

- Alice only has access to floors 4 through 8
- Bob only has access to floors 3 through 7
- Charles must be at least 2 floors higher than Bob
- Alice must be at least 1 floor higher than Bob
- Alice must be at least 1 floor lower than Charles

Our goal is to assign Alice, Bob, and Charles to the highest possible floors.

1. Formulate the problem as a CSP.

2. Formulate the problem as an IP problem.

## 2 Baymax's Factory

Baymax and the 281 TAs have opened a factory to produce special medicine and bandages. These are really difficult to produce and require the collaboration of robots and humans.

To produce an ounce of medicine, it takes 0.2 hours of human labor and 4 hours of robot labor. To produce an inch of bandage, it takes 0.5 hours of human labor and 2 hours of robot labor. An ounce of medicine sells for \$30 and an inch of bandages sells for \$30. Medicine and bandages can be sold in fractions of an ounce or inch.

We want to maximize our profit so we can buy gifts for all the students. However, the TAs are really busy so they can only devote 90 human hours. In addition, Baymax can only devote 800 robot hours because he has other obligations to tend to. How can we maximize our profit?

1. Is this a linear, mixed or integer programming problem? Formulate and solve it.

2. Now suppose the items can only be sold in whole units (by ounce/inch). Is this a linear, mixed, or integer programming problem? Perform branch and bound for one branch level. You do not have to evaluate; writing out the constraints will suffice.

3. Now assume medicine can be sold in fractions but bandages can only be sold in whole units. What kind of a programming problem would this be, and how would our evaluation process differ from the problem type in part b?

4. How many optimal solutions can a LP have? How about IP?

# 3 4-Queens

Recall the 4-Queens problem. The goal is to place 4 chess queens on a 4x4 chess board such at no two queens are in the same row, column and diagonal.

Formulate the 4-Queens problem as an integer programming problem.

# 4 Conceptual Review

1. Vocabulary check: Are you familiar with the following terms?
   - Symbols:
   - Operators:
   - Sentences:
   - Equivalence:
   - Literals:
   - Knowledge Base:
   - Entailment:
   - Clauses (Definite vs. Horn Clauses):
   - Model Checking:
   - Theorem Proving:
   - Modus Ponens:

2. Recall the definitions of satisfiability and entailment.

- **Satisfiability**:

- **Entailment**:

3. What is the difference between satisfiability and entailment?

4. Suppose $A \models B$. Consider all models assigning values to variables in sentences $A$ and $B$. Which of the following sentences must be true in all possible models (even if either or both $A/B$ are false)?

   (a) $A \wedge B$
   (b) $A \Rightarrow B$
   (c) $B \Rightarrow A$
   (d) $A \vee B$
   (e) $B$

5. Determine which of the following are correct, and explain your reasoning.

   - $(A \vee B) \models (A \Rightarrow B)$

   - $A \iff B \models A \vee \neg B$

   - $(A \vee B) \wedge \neg(A \Rightarrow B)$ is satisfiable

6. How would we formulate the SAT problem as a CSP? What are the variables? Domains? Constraints?

7. Suppose we have an algorithm which determines whether a sentence is satisfiable or not. Given two sentences $A$ and $B$, how could we determine whether $A \models B$?

# 5 SATurdays are for everyone

1. Determine whether the sentences below are satisfiable or unsatisfiable (using any method you like).

   (a) $(\neg(Y \vee \neg Y) \vee X) \wedge (X \vee (Z \iff \neg Z))$

   (b) $\neg(X \vee \neg(X \wedge (Z \vee \top))) \implies \neg(Y \wedge (\neg Y \vee (\top \implies \bot)))$

   (c) $(((\top \iff \neg(X \vee \neg X)) \vee Z) \vee Z) \wedge \neg(Z \wedge ((Z \wedge \neg Z) \implies X))$

2. Given the following knowledge base (propositions we know to be true):

   - $P$
   - $\neg P \vee Q$
   - $P \vee \neg Q \vee R$
   - $\neg Q \vee R$

   Prove $R$ to be true using resolution.

# 6 Wandering in Wumpus World

We bring together what we have learned in lecture as well as the ideas of search so far in order to construct wumpus world agents that use propositional logic. The first step is to enable the agent to deduce, to the extent possible, the state of the world given its percept history. This requires writing down a complete logical model of the effects of actions. We also show how the agent can keep track of the world efficiently without going into the percept history for each inference. Finally, we show how the agent can use logical inference to construct plans that are guaranteed to achieve its goals.

Try it out: http://thiagodnf.github.io/wumpus-world-simulator/ Note that there are some slight differences between this online version and the version we describe below.

Throughout this question, we will present several screenshots from the Wumpus World simulator linked previously. Note that the location of the explorer can be ignored. We just tried to place him somewhere off screen!

Recall that an agent in the Wumpus World has access to the following percepts:

- In the square containing the wumpus and in the directly (not diagonally) adjacent squares, the agent will perceive a Stench.
- In the squares directly adjacent to a pit, the agent will perceive a Breeze.
- In the square where the gold is, the agent will perceive a Glitter.
- When an agent walks into a wall, it will perceive a Bump.
- When the wumpus is killed, it emits a woeful Scream that can be perceived anywhere in the cave.

1. Consider the following Wumpus World state:

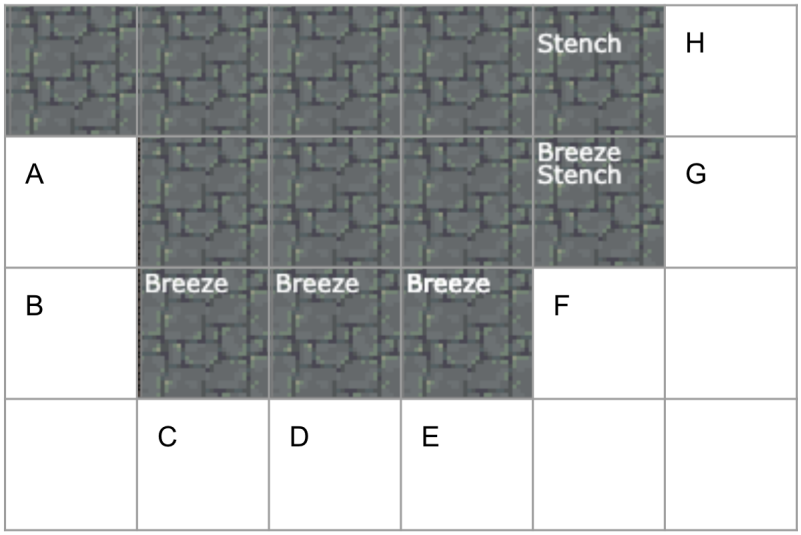


Figure 1: Safe, not safe, or unsure?

For the squares A-H, mark each of them with '+' if the square is definitely safe, '–' if it is definitely not safe, and '?' otherwise.

2. Consider the PL symbols $S_A, S_B \dots S_H$ that represent whether or not each square A-H is safe or not safe. We also have symbols $W_i$ and $P_i$ for whether square $i$ has a wumpus or a pit, respectively.

   Let our Knowledge Base (KB) be a propositional logic sentence in CNF which contains all of the logic for the rules of Wumpus World as well as the percepts for the squares already visited (i.e., info about stenches, breezes, or not in the 12 dark squares in the figure). Assume the KB also has logic related to the safety of a square, i.e.,$\neg W_i \wedge \neg P_i \iff S_i$.

   You are given access to a black box SAT solver function `PL_SATISFIES(pl_sentence}` which returns a PL model satisfying the given sentence *if* it is satisfiable and `None` otherwise.

   Write pseudocode using the SAT solver and given propositional logic symbols to determine is square $i$ is:

   (a) Definitely safe

   (b) Definitely not safe

   (c) Unsure