# Project Summary

*In this scenario, we are a resident of the Pacific Northwest that wishes to grow a vegetable garden. However, the EPA has declared the trees on our property to be environmentally protected. Undeterred, we set out to grow a garden around these annoying pine trees.*

This project aims to track the status of a virtual grid-based garden over 10 time points. For each place in the grid, there will be a plant. This plant can be a pine tree or it can be one of four types. Each of the four types of plant has a positive effect on one other type of plant and a negative effect on one other type of plant. Pine trees negatively affect all other plants. If a plant is negatively affected by another plant and it is not positively affected by any other plant, it will die. Dead plants no longer effect

# Propositions/Constraints

A nearly-complete wiki of our project scenario, including propositions and constraints to be implemented, is attached alongside this document in the same folder location. It also includes potential model explorations to be done in the future.

# Model Exploration

The longest phase of the project was in attempting to design a model wherein the solution was not trivial or impossible and that was within the scope of this project. During our first brainstorming session, we hit upon the idea of virtual gardening on a grid and the idea that plants could help or hinder each other. However, the constraints of the project were vastly different. Plants could be struggling or thriving if they were alive, and the objective was instead a fully thriving garden. Plants would not be thriving unless they were completely helped by all other plants. Pine trees were plants like any other with plants that they helped and plants that they hindered. Every plant had to be used once, so even undesirable plants would be used.

However, we realized that this model had several problems. Firstly, the many states of a plant made it difficult to decide on what the goal state of the model should be. Should it be if all the plants should be thriving or if they had to be merely alive? In both of those cases, it meant that any plant that was not in the goal state was functionally in a “loss state,” so it wouldn’t matter if it was struggling or dead. The same problem but in reverse applied if the goal state was that all plants were alive. There was also debate over what conditions would cause a plant to be dead rather than merely struggling. Progress stalled as our group tried to establish the basic propositions of our proposal.

However, the biggest problem with this model was that there was very little variation in what a functional model would look like. Since the grid was always identical, if a configuration worked once it would always work. We considered the idea of a randomly placed “water” tile that would cause any plan next to it to always thrive. However, this did not change that there was always a trivial solution of building a grid made of mutually-benificial plants that worked every time.

We therefore needed a new model. After some iteration, we decided on pine trees scattered throughout the grid as the independent variable for our new model. This necessitated the removal of any plant that benefited from pine trees as those plants would be dominant, and this change necessitated that plants would now be as alive as possible, lest any plant next to a tree always die. We also changed our catalogue of plants so that each plant would positively effect one plant and harm one other plant. With this model, we submitted out project proposal.

However, this model was too simple.

# Jape Proof Ideas

An idea of a jape proof would be to represent a simple sample problem within a 2 by 2 grid that our model would hope to solve, originally, we developed a proof that used a variable E, to represent an empty slot, this was quickly scraped because it did not properly follow the rules we had set, so we changed our approach, and while we have not fully solidified how this would be represented, we have a couple of possibilities, here are two possible representations of a problem:

(C(0,0)PT)∧(C(0,1)CO)∧(C(1,0)B)∧(C(1,1)T) ⊢ A(0,0)∧A(0,1)∧A(1,0)∧A(1,1)

PT(0,0)∧C(0,1)∧B(1,0)∧T(1,1 )⊢ A(0,0)∧A(0,1)∧A(1,0)∧A(1,1)

Of these, the second given option seems more appropriate to answer the problem, however no consensus has been reached.

# Requested Feedback

* The feedback from our project proposal suggested we increase the complexity of our project in order to allow for additional complexity and model exploration. With the current additions of watering/fencing and time intervals (see wiki document), do we now have adequate complexity to move forward, or should we add more features?
* At this point in time, we are unsure as to what exactly we should be attempting to prove or demonstrate using the Jape proofs. Could one use be, for example, proving that a garden configuration that is optimal in *practice* is truly optimal?
* Is the use of a “garden plot” class to encapsulate all the logic data of a cell (x,y) and time *t* a good idea? Or should we stick more with basic arrays of all garden data? (Like how Prof. Muise uses arrays to store all possible edge/distance propositions)