Miguel Angel Montoya
montoya5@purdue.edu
ID: 0031152048

Homework 1

1. **Problem 3.6**
   Give a complete problem formulation for each of the following. Choose a formulation that is precise enough to be implemented.

   a) *Using only four colors, you have to color a planar map in such a way that no two adjacent regions have the same color.*
      - Initial state: Regions with no color.
      - Actions: Color a selected region.
      - Transition model: Map before current coloring, plus the selected region with color
      - Goal test: Map with all regions colored, without adjacent regions with the same color.
      - Path cost: Each coloring of a region increases the total cost.

   b) *A 3-foot-tall monkey is in a room where some bananas are suspended from the 8-foot ceiling. He would like to get the bananas. The room contains two stackable, movable, climbable 3-foot-high crates.*
      - Initial state: Crates not stacked somewhere in the room. Monkey somewhere in the room. Banana in the 8 foot-high ceiling.
      - Actions: Monkey move towards crate, move crate to point, stack crate on other crate, climb crate, take banana.
      - Transition model: Room that reflects the change that the monkey caused. (Cratesmoved, monkey moved, crates stacked...)
      - Goal test: Monkey has the banana.
      - Path cost: Each action increments the cost.

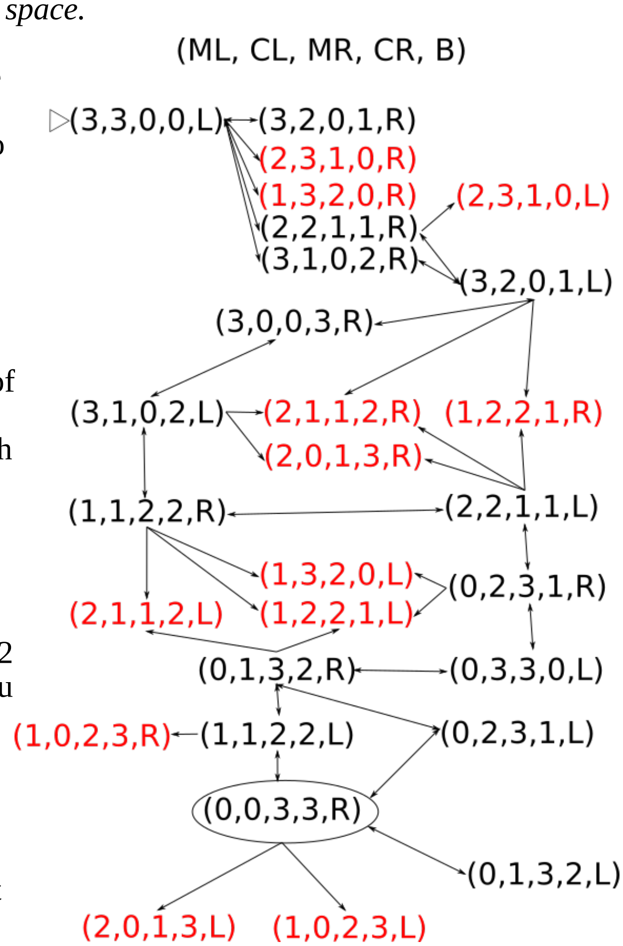
   c) *You have a program that outputs the message "illegal input record" when fed a certain file of input records. You know that processing of each record is independent of the other records. You want to discover what record is illegal.*
      - Initial state: No record analyzed.
      - Actions: Divide file in parts, test part.
      - Transition model: Tested parts without problems are discarded a candidates to contain the faulty record.
      - Goal test: Find faulty record
      - Path cost: Each analysis increases path cost.

   d) *You have three jugs, measuring 12 gallons, 8 gallons, and 3 gallons, and a water faucet. You can fill the jugs up or empty them out from one to another or onto the ground. You need to measure out exactly one gallon.*
      - Initial state: 3 empty jugs
      - Actions: Deposit water to a jug from the faucet, deposit water to a jug from another jug, drop water from jog to the floor.
      - Transition model: Change in how much liquid does a jug (Or pair of jugs) contain. Increase in one if filled with the faucet; decrease in one and increase in another if transporting water from one to another; or decrease in one jug when water is dropped.
      - Goal test: A jug contains one gallon.
      - Path cost: Each action increases path cost.

2. **Problem 3.9**
The missionaries and cannibals problem is usually stated as follows. Three missionaries and three cannibals are on one side of a river, along with a boat that can hold one or two people. Find a way to get everyone to the other side without ever leaving a group of missionaries in one place outnumbered by the cannibals in that place. This problem is famous in AI because it was the subject of the first paper that approached problem formulation from an analytical viewpoint (Amarel, 1968).

a) *Formulate the problem precisely, making only those distinctions necessary to ensure a valid solution. Draw a diagram of the complete state space.*

- Initial state: 3 missionaries and 3 cannibals on one side of the river. Boat is on the same side.
- Actions: Take one missionary or cannibal to the other side, or take two cannibals/two missionaries/one missionary and one cannibal to the other side. The action transport the boat to the other side too.
- Transition model: One side is reduced in population and the other side is incremented, depending on the population of the boat that is moving.
- Path cost: The total cost is increased on each travel.



b) *Implement and solve the problem optimally using an appropriate search algorithm. Is it a good idea to check for repeated states?*
Heuristic used: h(x) = (people left on left side)/2
Its an admissible heuristic as it assumes that you can move two people from the left to the right each time. This is in fact false, so the heuristic will always be less than the actual cost.
- A* already implements the checking to some extent. It only reopens an already explored node, if the total cot to reach it from the current node, is less than the already stored.
- Implementing the checking for IDFS would be unnecessary, as it defeats the main purpose of the algorithm that is to impede the formation of infinite loops. Storing which states have been visited might make the algorithm a bit faster as it impedes a branch from expanding without necessity.

c) *Why do you think people have a hard time solving this puzzle, given that the state space is so simple?*
Because there are states that to achieve progress require to take even two people back to the left side. That might sound counter intuitive, or even impossible to do for some algorithms (local search).

Note: n part b, use iterative deepening depth-first search and also use A with a heuristic of your choice. The initial state is defined in the book: 3 missionaries and 3 cannibals are on one side of the river, and they need to go to to the other side. By the end of each algorithm you need to print in the console the states and actions from the beginning to the goal (state graph path).*

3. **Explain the difference between local search and greedy best-first search on a graph.**
   The greedy algorithm will choose the next node to explore with the best heuristic from all the possible successors from the already explored nodes. Meanwhile, local search will only select the best successor from the current explored node.

4. **Give an example where local search is better or explain why this is impossible.**
   The case where even though both algorithms reach the global maxima, the greedy method might not return the optimal path. Meanwhile, because local search only sticks to the current possible successors, it might miss the option that has better heuristic but worse path cost, providing the optimal path.

5. **Give an example where local search is worse or explain why this is impossible.**
   Local search can be worse than greedy if it reaches a local minima. Even if using simulated annealing (That would let a number of bad moves), local search might never reach the global maxima.

6. **Give an example where local search is better than A* search. Explain why this is possible even though A* is optimal.**
   The disadvantage of A*, even if its complete and optimal, it's its space complexity. It has to store every previous explored node, meanwhile, local search only cares about the current node and next possible current node among the successors candidates.