Homework 1 - Robert Choe

September 11, 2018

1 CMSC421: Homework #1

1.1 Problem 2.10: pg 63

2.10 Consider a modified version of the vacuum environment in Exercise 2.8, in which the agent is penalized one point for each movement.

a. Can a simple reflex agent be perfectly rational for this environment? Explain.

b. What about a reflex agent with state? Design such an agent.

c. How do your answers to a and b change if the agent's percepts give it the clean/dirty status of every square in the environment?

A) A simple reflex agent cannot be perfectly rational in this environment. Since the agent is penalized one point for each movement and the reflex agent has no history of what has already done, then the agent will eventually take unnecessary movements. Take for example the agent has cleaned both room A and room B and ends up in room B. At this point, it does not have any historical knowledge that it did this and will turn Left (according to the function map) and check if Room A again and repeat.

b) A reflex agent with state can be perfectly rational in this environment. Since the reflex agent is able to maintain an internal model/state representing the vacuum world, it can map those states to certain reflexes.

```
Function state-reflex-vacuum-agent([location,status]) returns an action
    if state[location] == Clean :
       if location = A and status[B] == Dirty:
           return Right
       elif location = B and status[A] == Dirty:
           return Left
       else:
           return Done
    else:
        state[location] == Clean
        return suck
```

c) If the agent's percepts give it the clean/dirty of every square in the environment then it can perform perfectly rational. The simple reflex agent can have a reflex for every scenario based on the precepts it was given. The simple reflex agent with state can also update it's internal state based on the perepts given and execute it's reflexes rationally.

1.2 Problem 3.15 (a,b): pg 116

3.15 Consider a state space where the start state is number 1 and each state k has two successors: numbers 2k and 2k + 1.

a. Draw the portion of the state space for states 1 to 15.

b. Suppose the goal state is 11. List the order in which nodes will be visited for breadth-first search, depth-limited search with limit 3, and iterative deepening search.

```
In [6]: from binarytree import Node, tree

        root = Node(1)
        root.left = Node(2)
        root.right = Node(3)
        root.left.left = Node(4)
        root.left.right = Node(5)
        root.right.left = Node(6)
        root.right.right = Node(7)
        root.left.left.left = Node(8)
        root.left.left.right = Node(9)
        root.left.right.left = Node(10)
        root.left.right.right = Node(11)
        root.right.left.left = Node(12)
        root.right.left.right = Node(13)
        root.right.right.left = Node(14)
        root.right.right.right = Node(15)
```

a)

```
In [7]: print(root)
```

```
        ________1________
       /                 \
    __2___             ___3___
   /      \           /       \
  4       _5        _6        _7
 / \     /  \      /  \      /  \
8   9   10   11   12   13   14   15
```

b)

Breadth-first-search: 1,2,3,4,5,6,7,8,9,10,11
Depth-Limited-search: 1,2,4,8,9,5,10,11
Iterative-Deepening:
1st Iteration: 1
2nd Iteration: 1,2,3
3rd Iteration: 1,2,4,5,3,6,7
4th Iteration: 1,2,4,8,9,5,10,11

1.3 Problem 3.18: pg 117

3.18 Describe a state space in which iterative deepening search performs much worse than depth-first search (for example, O(n2) vs. O(n)).

m = maximum depth

A state space where the depth is large and the goal is at the bottom is a space where iterative deepining search will perform much worse than depth-first search. Depth first search will perform better because it will reach the bottom of the state space very quickly. Iterative deepening search will reach the bottom on it's m'th iteration.