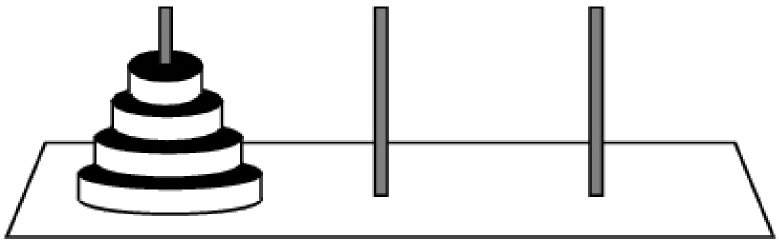
CS594: Artificial Intelligence, Fall 2004
Due: Noon, Friday, November 19 (either bring to my office or email to parker@cs.utk.edu)

Homework 6

1. (Rewards in Reinforcement Learning) Consider the "Tower of Hanoi" puzzle, pictured below.



In this puzzle, we are given a stack of *n* disks arranged from the largest on the bottom to the smallest on top, placed on a rod, together with two empty rods. The objective is to move the stack of disks from one rod to another with the minimum number of valid moves. A valid move places a smaller disk on top of a larger disk (i.e., no larger disks are allowed to be on smaller disks), or places a disk on an empty rod.

Let us consider three ways to set up the rewards for reinforcement learning; in all 3 cases, the discount factor is 1 and the goal state is a terminal state:

- Every legal move has an associated reward of –1.
- Every legal move has an associated reward of 0, except for transitions *into* the goal state, which have a reward of +10.
- Every legal move has an associated reward of -1, except for the transitions into the goal state, which have a reward of +10.

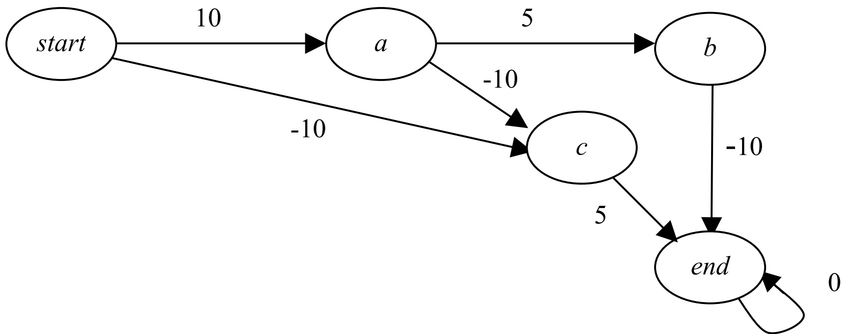
What is the essential difference between these formulations? What is the optimal policy under each? (You can answer this "in words" by analyzing the problem at a high level – you don't have to formulate it mathematically or show the results of the outcome of a specific algorithm.)

2. (Types of Reinforcement Learning) Compare and contrast the following types of Reinforcement Learning:

- Passive ADP Learning
- Passive TD Learning
- Active ADP Learning
- Active TD Learning (not TD Q-learning)
- Active TD Q-Learning
- TD Learning with function approximation
- Q-Learning with function approximation

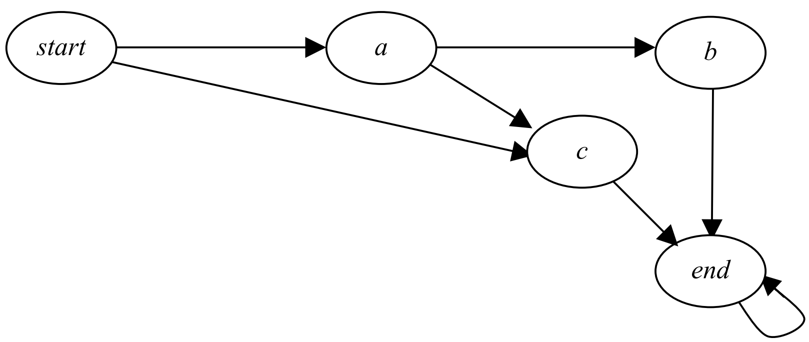
Your analysis should consider issues such as differences in what is learned, what information is needed for learning, convergence rates, scalability, etc.

3. (Q-learning) Consider the deterministic reinforcement environment drawn below. The numbers on the arcs indicate the *immediate* rewards of taking an action leading to a next state. Let the discount rate, γ, equal 0.9.



a. What is the best route for going from *start* to *end?* Why?

b. Represent the *Q table* by placing Q values on the arcs on the state-action graph; initialize all of the Q values to 2 except initialize all of the arcs directly involving node '*a*' to have a Q value of -1. For Step 1, do *exploitation*. Show on the graph below the full Q table *after* Step 1. Specify the actions chosen.



c. Assume that after Step 1, the RL agent is magically transported back to the state *start*. Show the resulting Q table after the learner takes its second step from the starting state. Step 2 should be *exploration*. Specify the actions chosen.