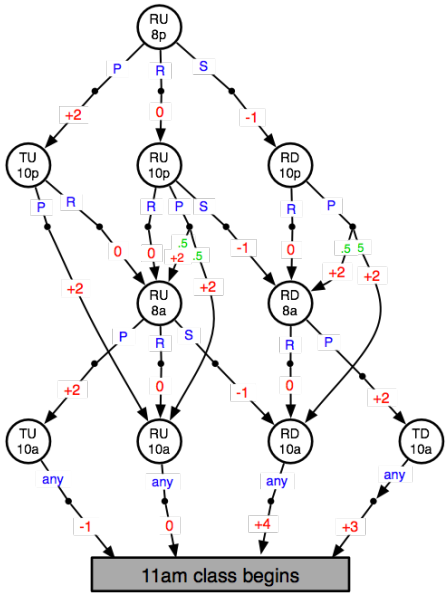
CMPSCI 383, Fall 2011
Homework 5
Due in class or in the main office of the Computer Science building by 4:00 PM,
December 8, 2011

Programming Assignment: (100 points)



States:

- R = Rested
- T = Tired
- D = homework Done
- U = homework Undone
- 8p = eight o'clock pm

Actions:

- P = Party
- R = Rest
- S = Study
- *any* means any action has the same effect.

Notice that not all actions are possible in all states. Red numbers are rewards. Green number are transition probabilities (all those not labeled are probability 1.0). The gray rectangle denotes a *terminal* state.

Part 1 (50 points)

Implement a program that models the Party Problem described above. Use any programming language of your choice. Assume that the agent follows a random equiprobable policy (i.e. the probability of picking a particular action while in a given state is equal to 1 / number of actions that can be performed from that state). Run your program for 50 episodes. For each episode, have your program print out the agent's sequence of experience (i.e. the ordered sequence of states/actions/rewards that occur in the episode) as well as the sum of the rewards received in that episode (i.e. the Return with respect to the start state) in a readable form.

What to hand in (on paper):

- The sequence of experience from each episode, including the Return observed in that episode.
- The values of each state (computed by hand using the Bellman equations).
- The average Return from the fifty episodes.
- The source code of your program.

Part 2 (50 points)

Implement ϵ-greedy Sarsa to find an optimal policy for the Party Problem described above. Tune the parameters (learning rate, γ, and ϵ) to get good performance.

What to hand in (on paper):

- An optimal policy.
- An analysis of how you searched for good parameters, including discussion of why it doesn't work when each is set to be too big or too small. Describe what different values of γ represent in the real world.
- The source code of your program.