Reinforcement Learning **Fall Semester, 1999/2000**

Homework 2: Nov. 25

Lecturer: Yishay Mansour

Homework number 2.

Question 1

Show an example of an MDP with n states and two actions, where the Policy Iteration algorithm requires n iterations to converge to the optimal policy.
(**Open problem:** Show an example that requires more than n iterations.)

Question 2

Let π_1 and π_2 be two deterministic stationary policies for a given MDP M. Construct a new policy π in the following way,

$$\pi(s) = \begin{cases} \pi_1(s) & \text{if } v_\lambda^{\pi_1}(s) \geq v_\lambda^{\pi_2}(s) \\ \pi_2(s) & \text{if } v_\lambda^{\pi_1}(s) < v_\lambda^{\pi_2}(s) \end{cases}$$

Show that $v_\lambda^\pi \geq \max\{v_\lambda^{\pi_1}, v_\lambda^{\pi_2}\}$.

Question 3

We can use the approximate value function, computed in the Value Iteration algorithm, to define a policy.

Prove that if the number of states and actions are finite, then after a finite number of iterations this policy equals the optimal policy and never changes.

Give an example of an MDP where the number of iterations until the policy defined by value iteration converges to the optimal policy is at least $1/(1-\lambda)$. (Hint: build an MDP such that from the start state you can branch two two different states, once you reach a state you remain in it.)

The homework is due in two weeks