# Department of Computing

# CS370: Artificial Intelligence

# Class: BSCS-5AB

# Lab 9: Markov Decision Process

# Date: 15-12-2017

# Time: 10am-1pm & 2pm-5pm

# Instructor: Dr. Omar Arif

# Lab 9: MDP

**Introduction**

A Markov Decision Process is a [discrete time](https://en.wikipedia.org/wiki/Discrete_time) [stochastic](https://en.wikipedia.org/wiki/Stochastic) [control](https://en.wikipedia.org/wiki/Optimal_control_theory) process. At each time step, the process is in some state , and the decision maker may choose any action  that is available in state . The process responds at the next time step by randomly moving into a new state , and giving the decision maker a corresponding reward .

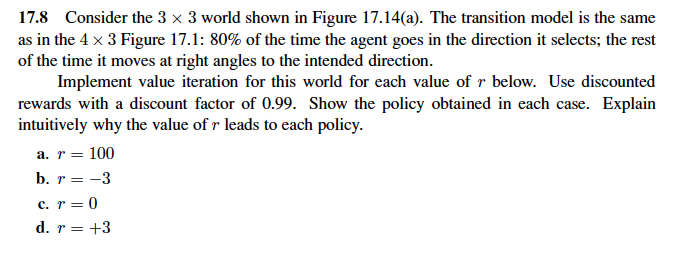
**Objectives**

In this lab you will use MDP to solve grid world problems similar to volcano world. See the supplied implementation of MDP in mdp.py. Understand how policy iteration and value iteration are implemented. See mdp.txt to understand how to use the mdp.py for solving grid worlds. mdp.txt solves the grid world problem of fig 17.1 of your text book.

**Pre-requisites:**

**Tools/Software Requirement**

Python

**Lab Tasks**

**Solve the following problem using MDP implementation.**

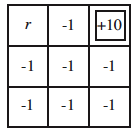
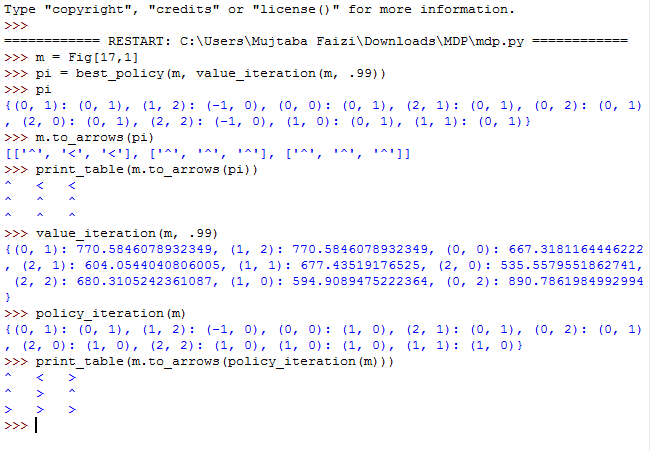


Figure 17.14(a): Upper right state is terminal state.

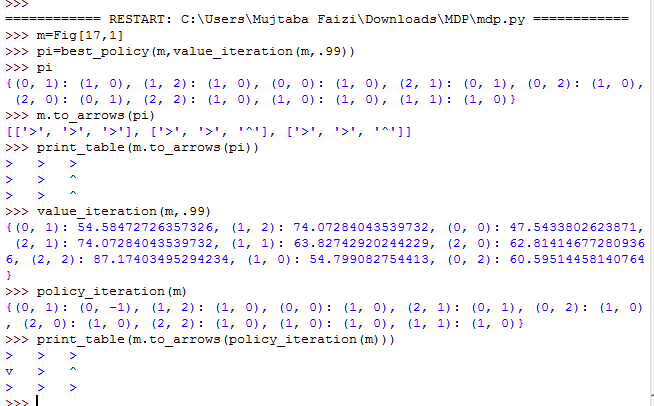
**R=100**

If the reward in the red square is 100, the agent will likely want to stay in this square forever and hence avoid to go to the final state (in gray). As we are dealing with a stochastic environment (we go in the direction we want with probability 0.8 and in the perpendicular directions with probability 0.1), the arrow around the final gray state need to point in the opposite direction to avoid going into the final state.



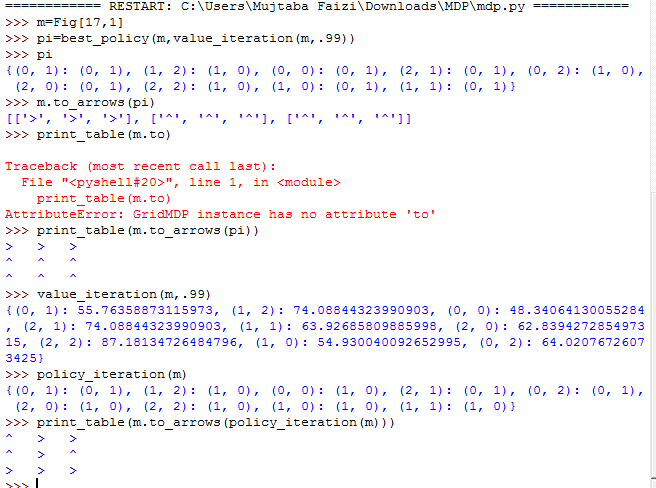
**R= -3**

If the reward in the red square is -3, then, as the reward of the white squares are -1 and the reward in the final square is +10, the agent we likely want to avoid the red square and go as fast as possible to the gray square. However, we don’t have a down arrow in (1,2) because if we were to put a down arrow in (1,2) the agent will likely make a detour that can can cost more than -3 points.



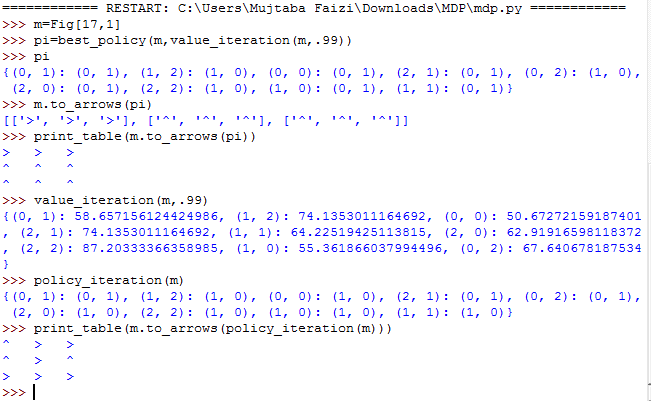
**R=0**

Here the reward for the red square is 0, so, as the rewards in the white squares are -1, the agent will want to go through the red square before reaching the final gray square. It won’t want to stay in the red square as the final square offer a +10 reward. That explains the sense of the arrows.



**R= 3**

Here r = 3, so the agent will want to stay in the red square indefinitely (same explanations as in **a**).



**Deadline:**

End of lab.