Assignment 3

Markov Decision Processes

Afnan Mousa Mousa

Khadija Assem Saad

Shimaa Kamal El-Deen

Algorithms

Algorithm

- First of all, we need to get the possible actions, states, and transitions. So the possible actions are: right, left, up and down as specified.

  For the states, there are 9 states including the “**r**” state and the final state.

  For the transitions, these are all the possible moves any state can make by taking an action from the 4 actions.
- It’s required to calculate the maximum reward that each state can collect by value iteration method.
- So, first we get the transitions by the create_transition**()** function that determines for each state its possible action and stores them in a dictionary called transitions.
- For the **valueIteration** method:

  There is a V vector that contains the expected utility state that needs to be modified after each iteration. It’s calculated as shown from this equation:

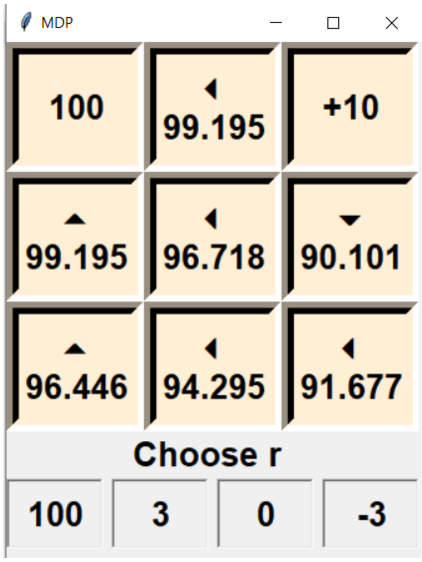
  $$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s')\right]$$

  So, for each state there are 4 values as there are 4 directions, (AKA **Q**), so we get all these 4 values and maximize them to get the new **V** value.

- We repeat this process until we reach a terminal state (r or +10), or after performing all the iterations, or if the change between the old **V** and the new **V** is less than the specified max. Error (**epsilon** in our implementation)

Policy

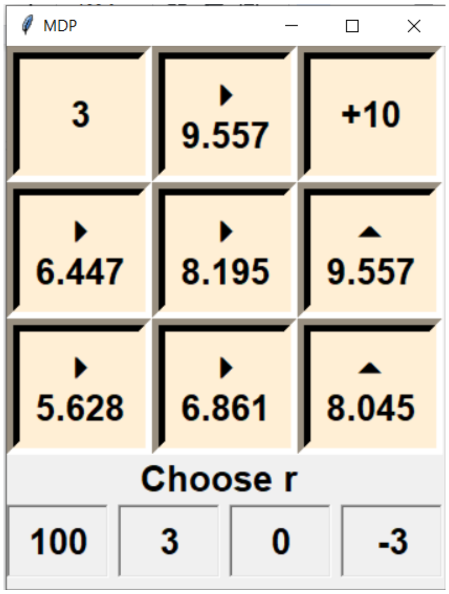
For r = 100



100	◀ 99.195	+10
▲ 99.195	◀ 96.718	▼ 90.101
▲ 96.446	◀ 94.295	◀ 91.677

⟶ Here, the policy choses the directions to be towards the cell of +100, as in all cases its reward will be greater than any other case.

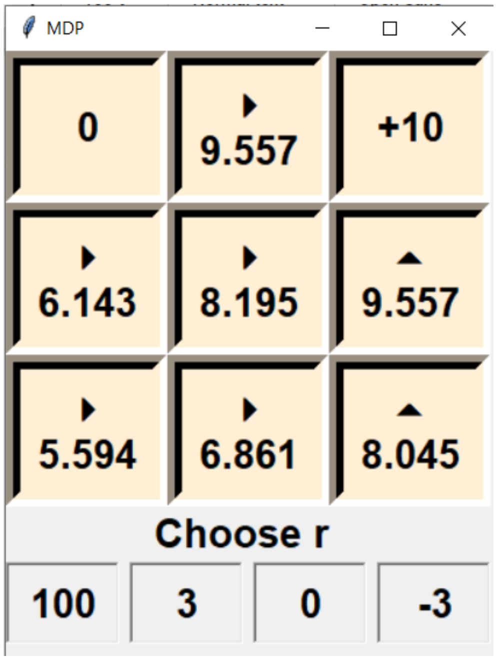
For r = 3:



3	▶ 9.557	+10
▶ 6.447	▶ 8.195	▲ 9.557
▶ 5.628	▶ 6.861	▲ 8.045

⟶ Here, the policy choses the directions to be towards the cell of +10 to collect more rewards than going through the 3 reward cell.

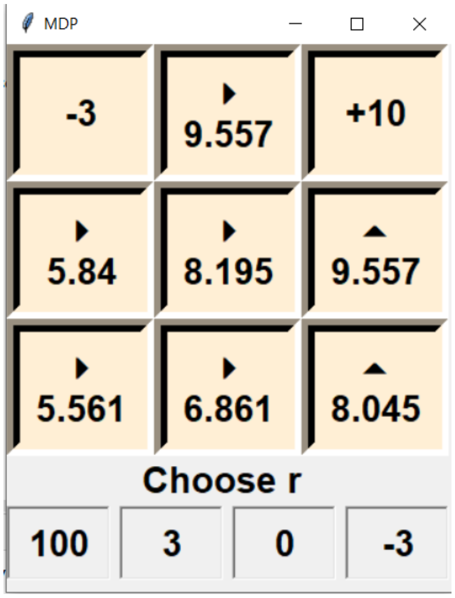
For r = 0



0	▶ 9.557	+10
▶ 6.143	▶ 8.195	▲ 9.557
▶ 5.594	▶ 6.861	▲ 8.045

⟶ Here, the policy also chooses the directions to be towards the cell of +10 to collect more rewards than going through the 0 cell as it has zero effect on the reward.

For r = -3



-3	▶ 9.557	+10
▶ 5.84	▶ 8.195	▲ 9.557
▶ 5.561	▶ 6.861	▲ 8.045

⟶ Here, the policy chooses to always avoid the cell of -3 to not decrease its reward. If it goes through any other cell it will be better.