Lab 10

FORWARD PROPAGATION

- Abhishek Singh
- Anisha Katiyar
- Kalash Shah
- Madhur Gupta

Q.1

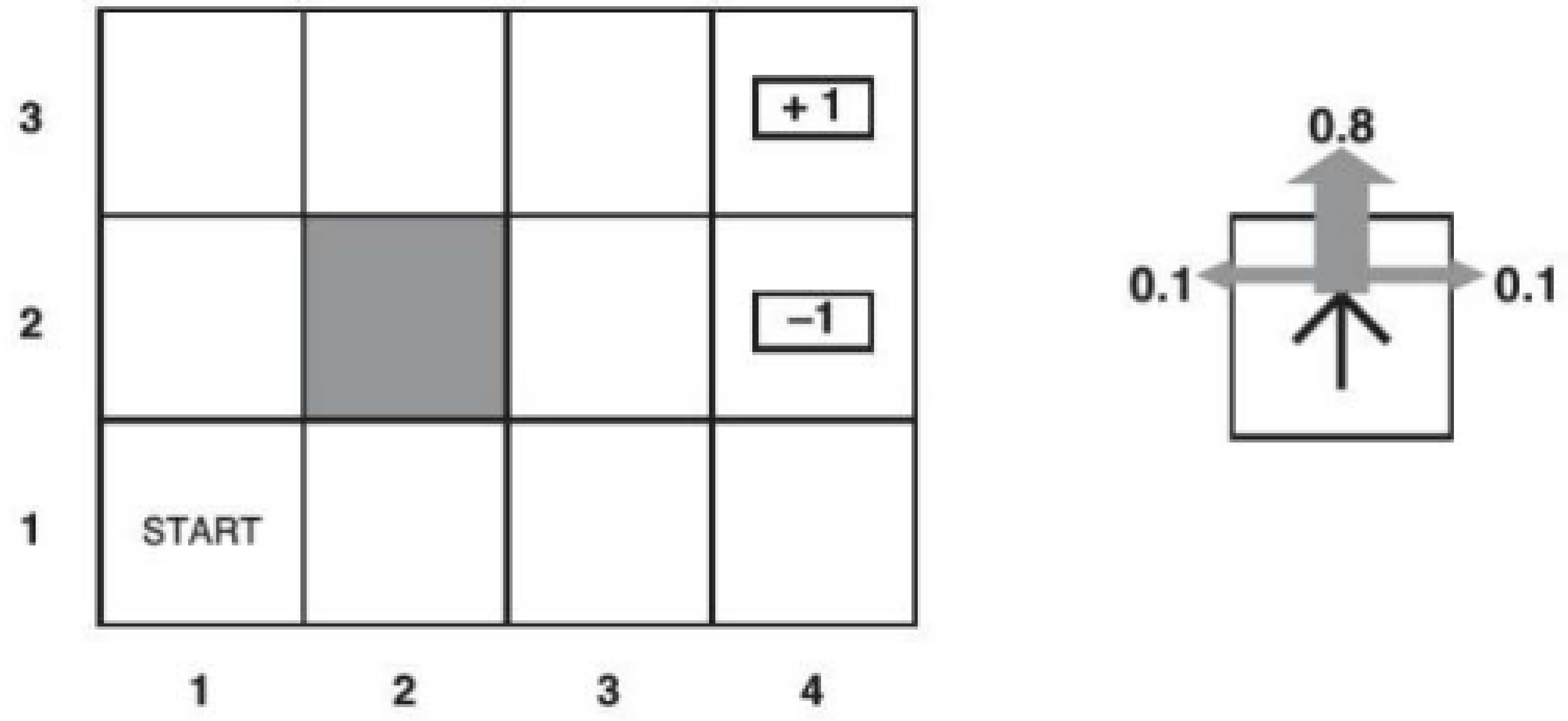


Figure 1: 4 × 3 grid world with uncertain state change.

Problem Statement

Suppose that an agent is situated in the 4x3 environment as shown in Figure 1. Beginning in the start state, it must choose an action at each time step. The interaction with the environment terminates when the agent reaches one of the goal states, marked +1 or –1. We assume that the environment is fully observable, so that the agent always knows where it is. You may decide to take the following four actions in every state: Up, Down, Left and Right.

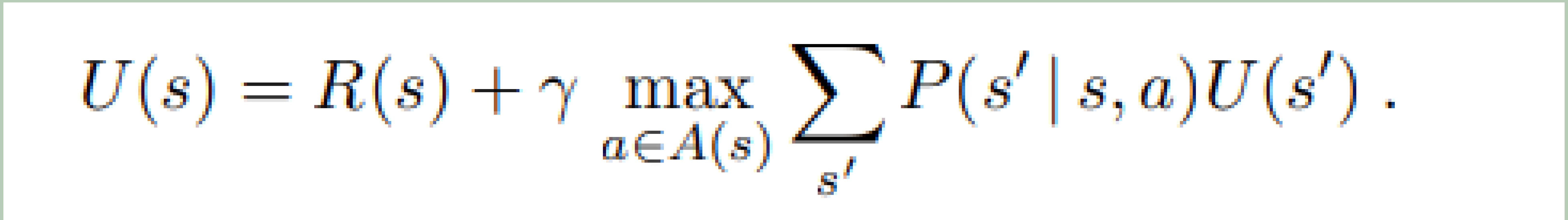
However, the environment is stochastic, that means the action that you take may not lead you to the desired state. Each action achieves the intended effect with probability 0.8, but the rest of the time, the action moves the agent at right angles to the intended direction with equal probabilities. Furthermore, if the agent bumps into a wall, it stays in the same square. The immediate reward for moving to any state (s) except for the terminal states S+ is r(s)= –0.04. And the reward for moving to terminal states is +1 and –1 respectively.

Find the value function corresponding to the optimal policy using value iteration.

Utility of a state

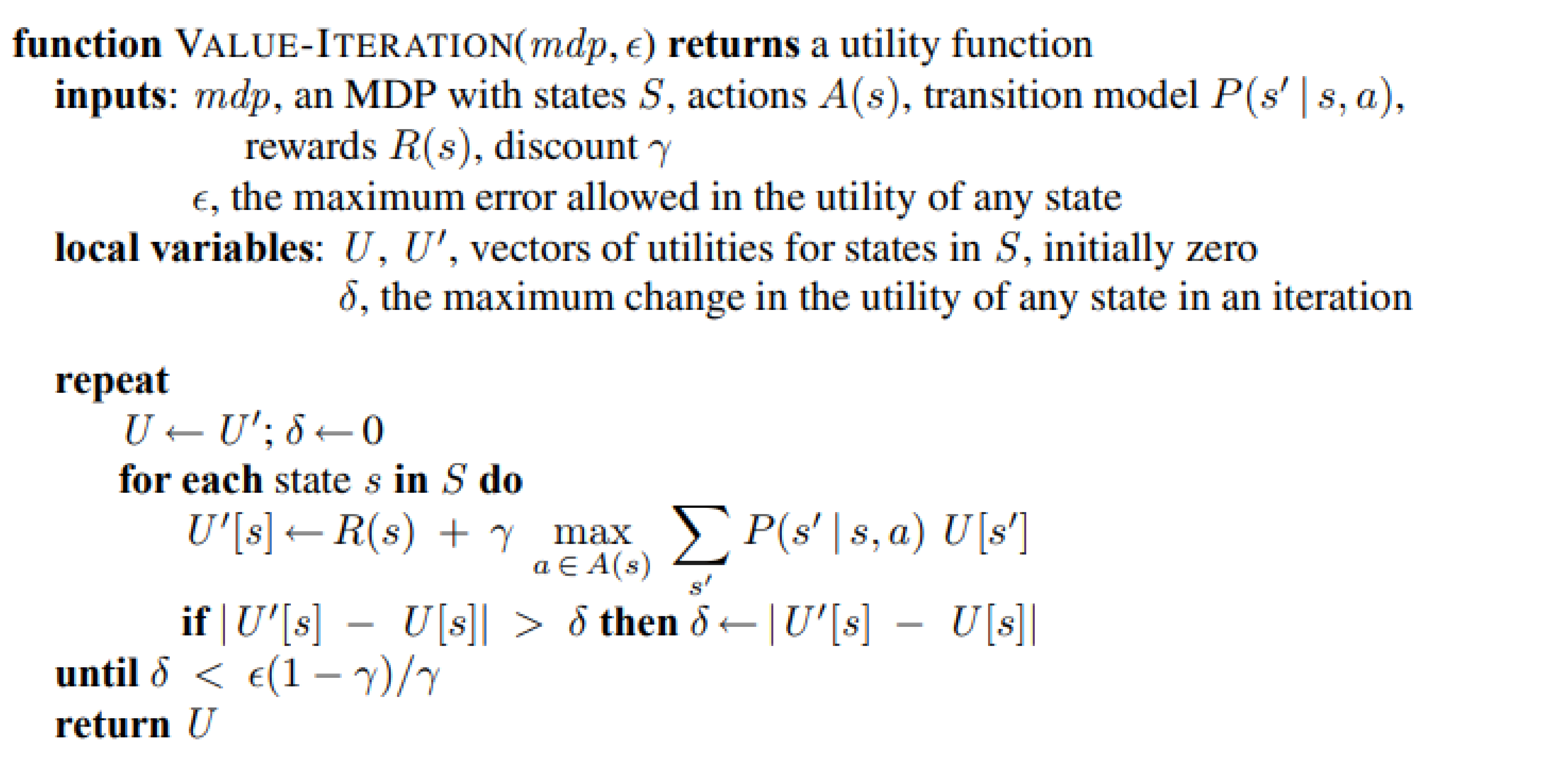
the utility of a state is the immediate reward for that state plus the expected discounted utility of the next state, assuming that the agent chooses the optimal action

Bellman Equation



$$U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' \mid s, a) U(s') \,.$$

Value Iteration Algorithm



```
function VALUE-ITERATION($mdp, \epsilon$) returns a utility function
    inputs: $mdp$, an MDP with states $S$, actions $A(s)$, transition model $P(s' \mid s, a)$,
                rewards $R(s)$, discount $\gamma$
            $\epsilon$, the maximum error allowed in the utility of any state
    local variables: $U$, $U'$, vectors of utilities for states in $S$, initially zero
                     $\delta$, the maximum change in the utility of any state in an iteration

    repeat
        $U \leftarrow U'$; $\delta \leftarrow 0$
        for each state $s$ in $S$ do
            $U'[s] \leftarrow R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s' \mid s, a)\, U[s']$
            if $|U'[s] - U[s]| > \delta$ then $\delta \leftarrow |U'[s] - U[s]|$
    until $\delta < \epsilon(1-\gamma)/\gamma$
    return $U$
```

Q.2

Gbike : Bicycle Rental

You are managing two locations for Gbike. Each day, some number of customers arrive at each location to rent bicycles. If you have a bike available, you rent it out and earn INR 10 from Gbike. If you are out of bikes at that location, then the business is lost. Bikes become available for renting the day after they are returned. To help ensure that bicycles are available where they are needed, you can move them between the two locations overnight, at a cost of INR 2 per bike moved.

Assumptions: Assume that the number of bikes requested and returned at each location are Poisson random variables. Expected numbers of rental requests are 3 and 4 and returns are 3 and 2 at the first and second locations respectively. No more than 20 bikes can be parked at either of the locations. You may move a maximum of 5 bikes from one location to the other in one night. Consider the discount rate to be 0.9.

Formulate the continuing finite MDP, where time steps are days, the state is the number of bikes at each location at the end of the day, and the actions are the net number of bikes moved between the two locations overnight.

Problem Statement

Tasks to Perform:

1. Find out how many Bikes should be moved overnight between each location to maximize the Total Expected Reward that is what should be the Strategy(Policy) given a Situation(State).

2. If the Strategy is known, how can it be compared which situations are better than the others(Value).

Terms to Know!

1. Policy
2. State
3. Value
4. Reward

Policy Iteration Algorithm!

1. Initialization
2. Evaluation
3. Improvement

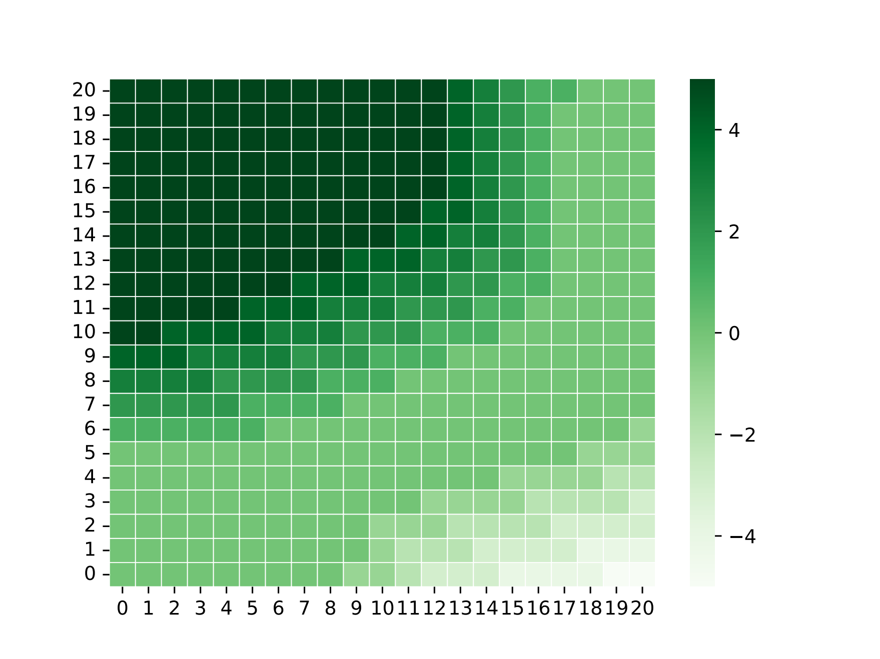
--- >>>

Running Policy Evaluation and Policy Improvement in a loop until the Policy gets Stable

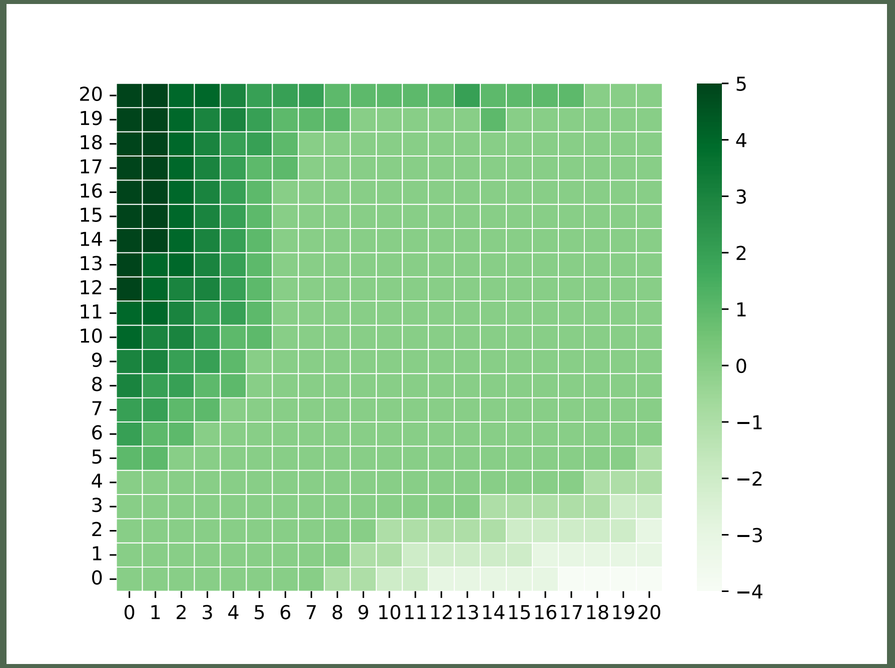
Results

Policy after each Iteration

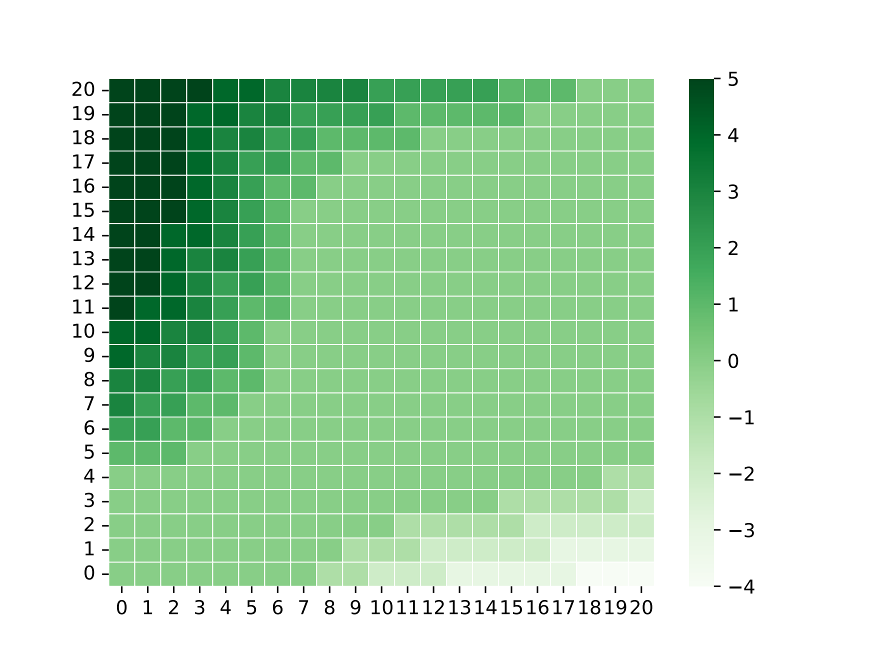
Policy - 1

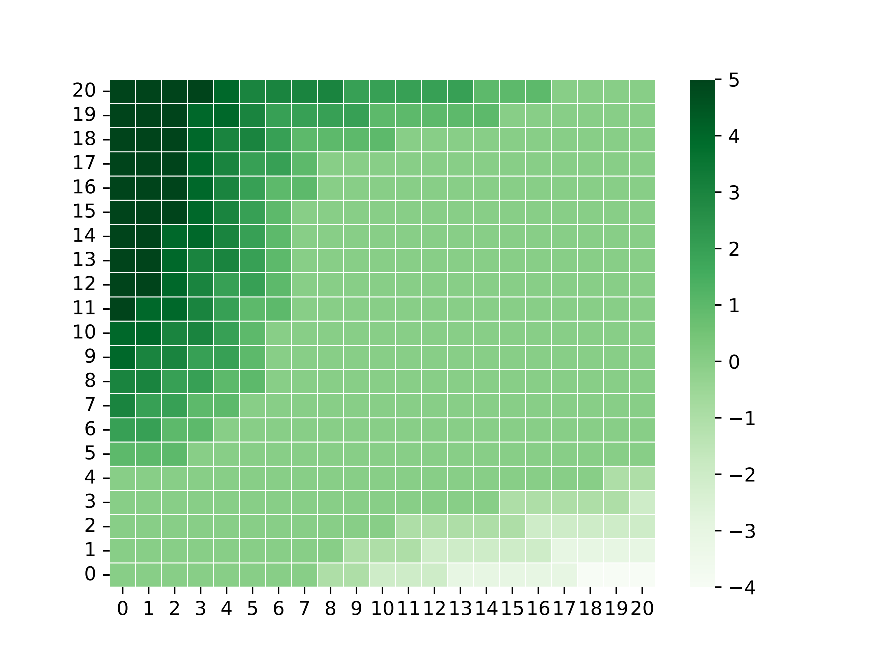


Policy - 2

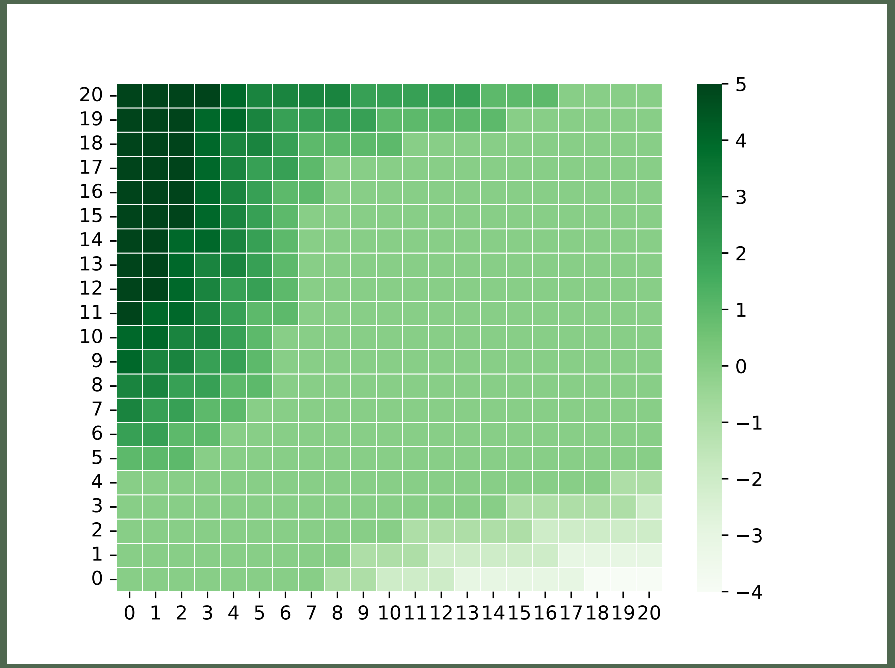


Policy - 3





Policy - 4

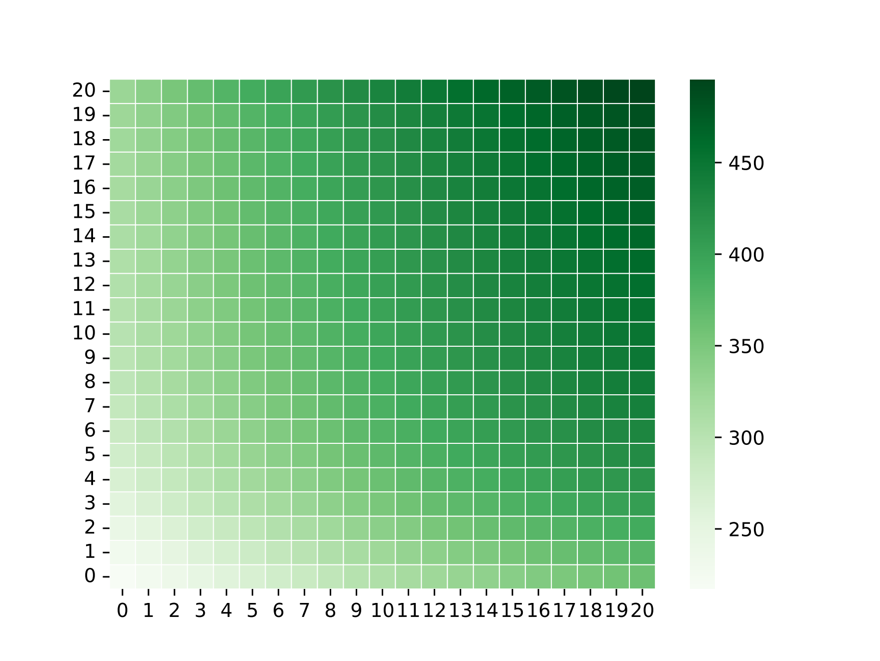


Policy - 5

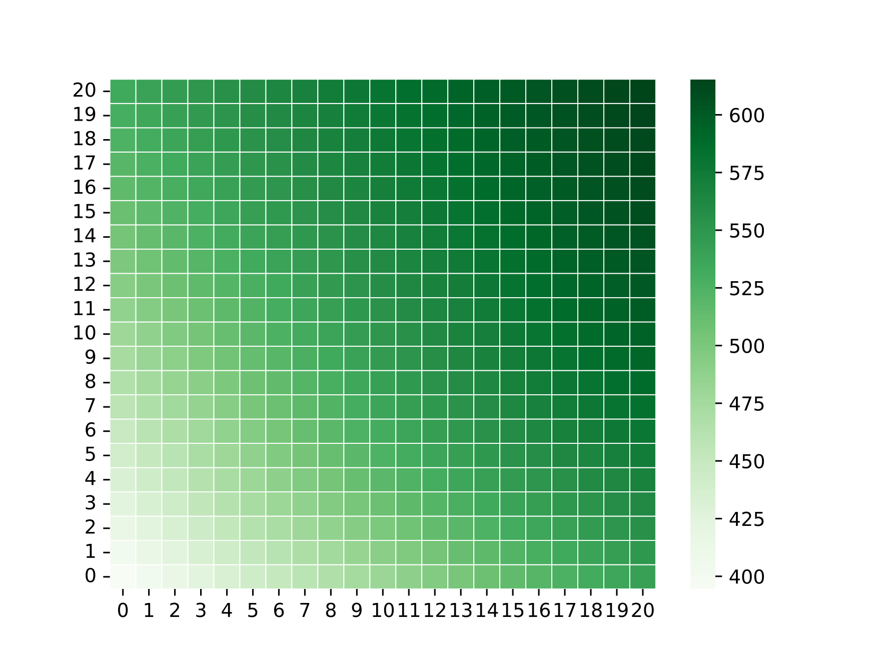
Results

Value after each Iteration

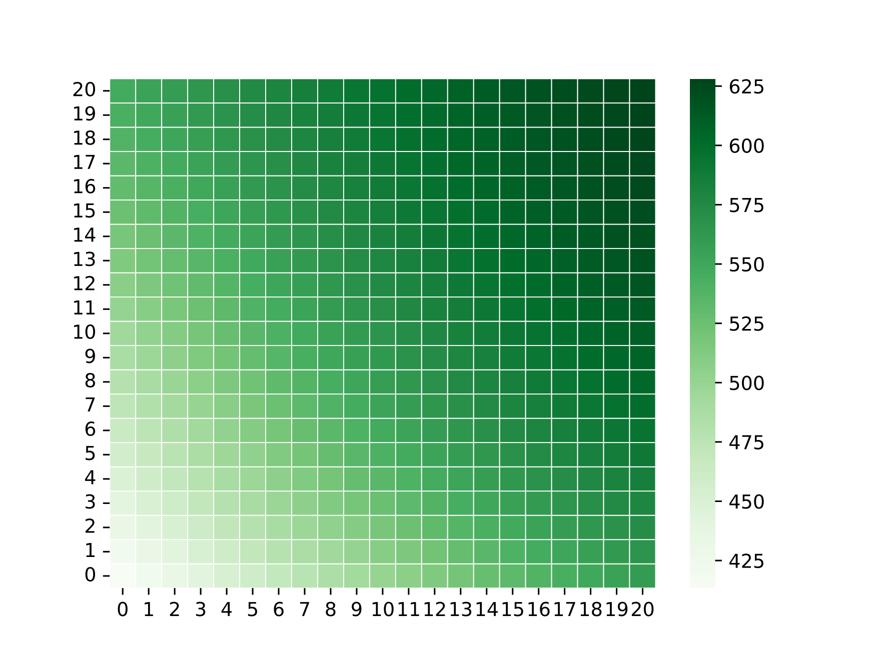
Value - 1



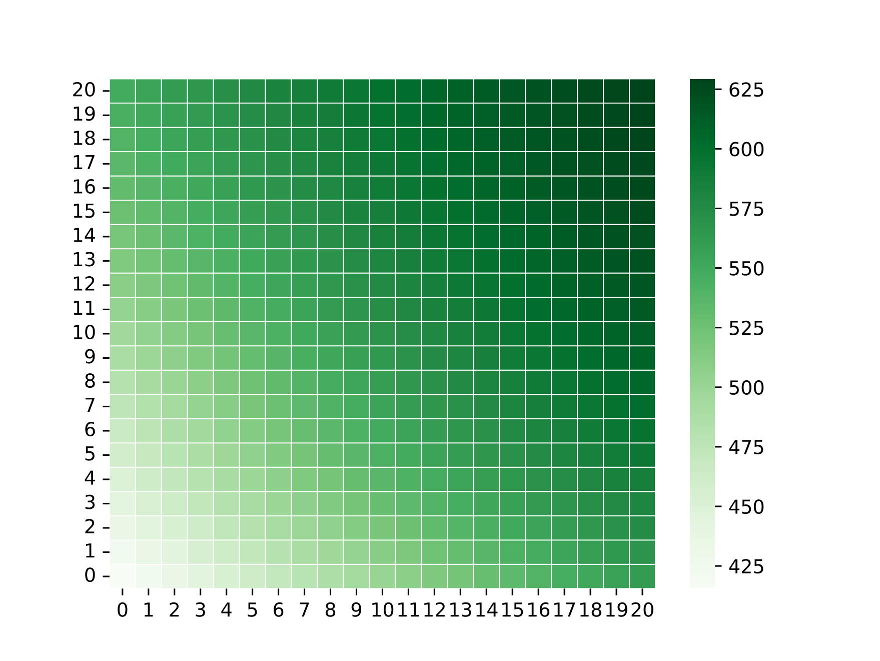
Value - 2



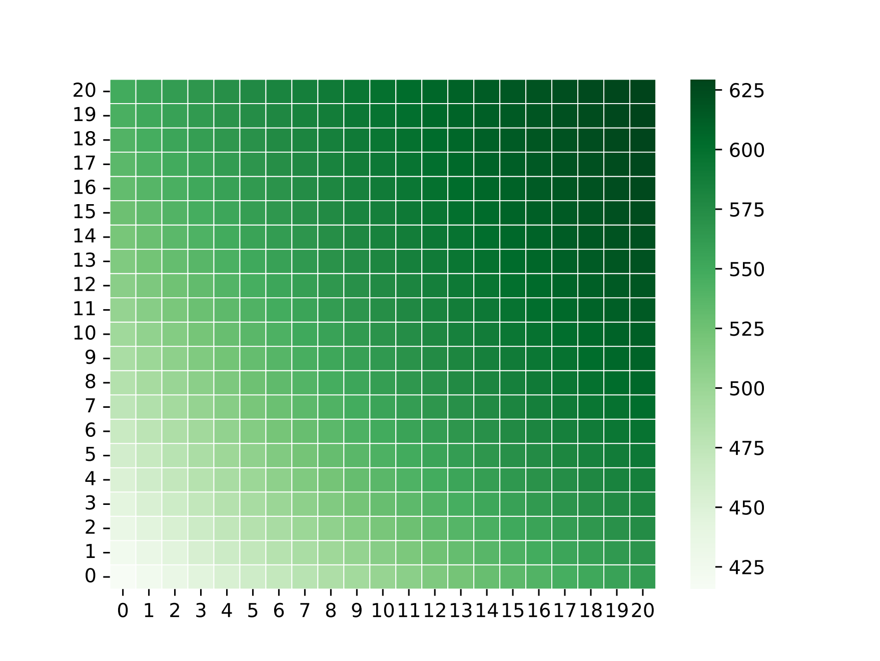
Value - 3



Value - 4



Value - 5



Q.3

Poisson Distribution

A Poisson distribution is a discrete probability distribution. It gives the probability of an event happening a certain number of times within a given interval of time or space.

We assume that the number of cars requested and returned at each location is a Poisson random variable.
Suppose λ is 3 and 4 for rental requests at the first and second locations and 3 and 2 for returns.

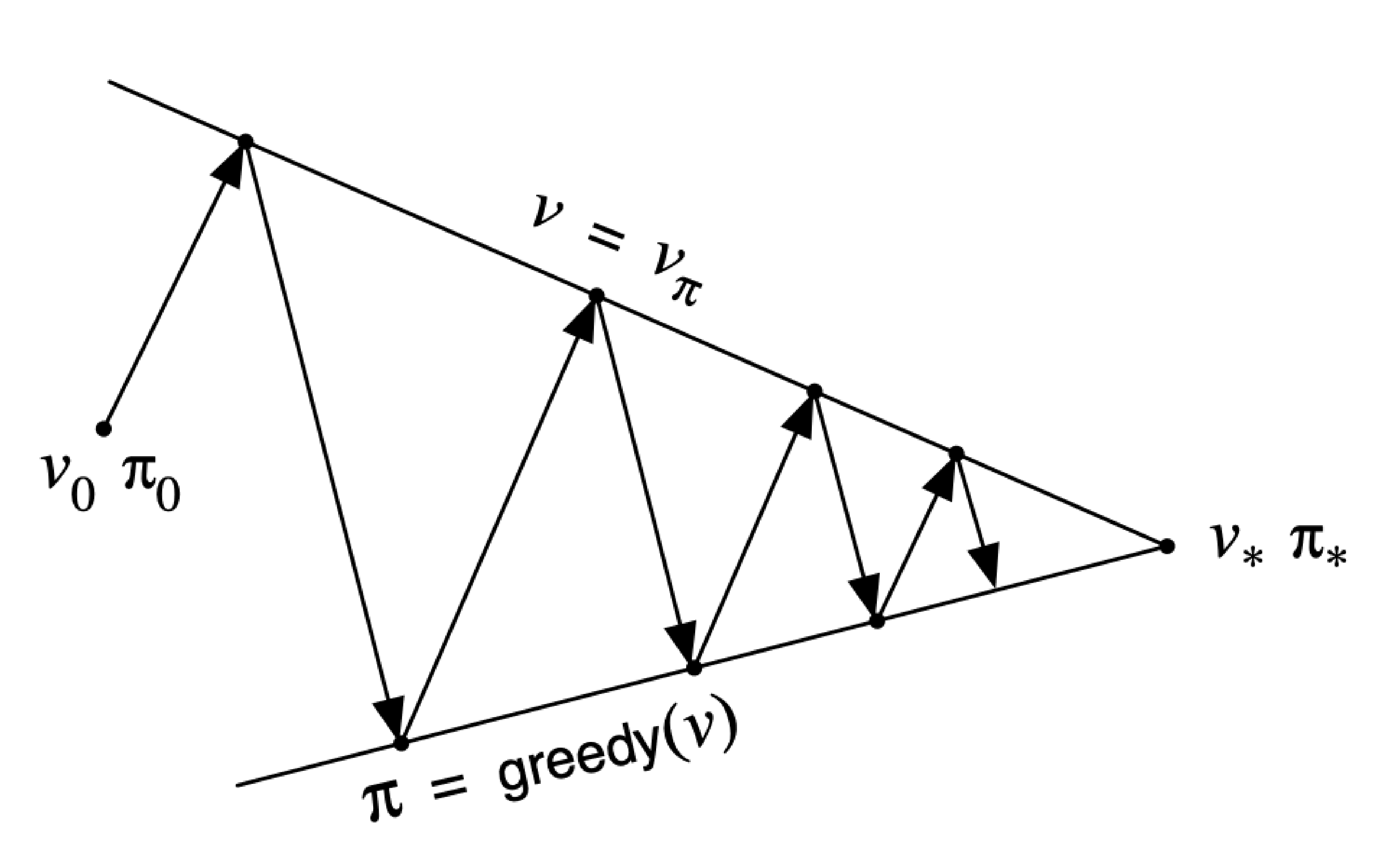
If x is a Poisson Random Variable, then

$$P(x) = \frac{\lambda^x e^{-\lambda}}{x!}$$

Question

One of your employees at the first location rides a bus home each night and lives near the second location. She is happy to shuttle one bike to the second location for free. Each additional bike still costs INR 2, as do all bikes moved in the other direction. In addition, you have limited parking space at each location. If more than 10 bikes are kept overnight at a location (after any moving of cars), then an additional cost of INR 4 must be incurred to use a second parking lot (independent of how many cars are kept there).

- The state is the number of cars at each location at the end of the day.
- The actions are the net numbers of cars moved between the two locations overnight.
- Rewards are the ₹10 reward when he rents a car, and the second being the -₹2 reward for each car that he moves from one location to the other

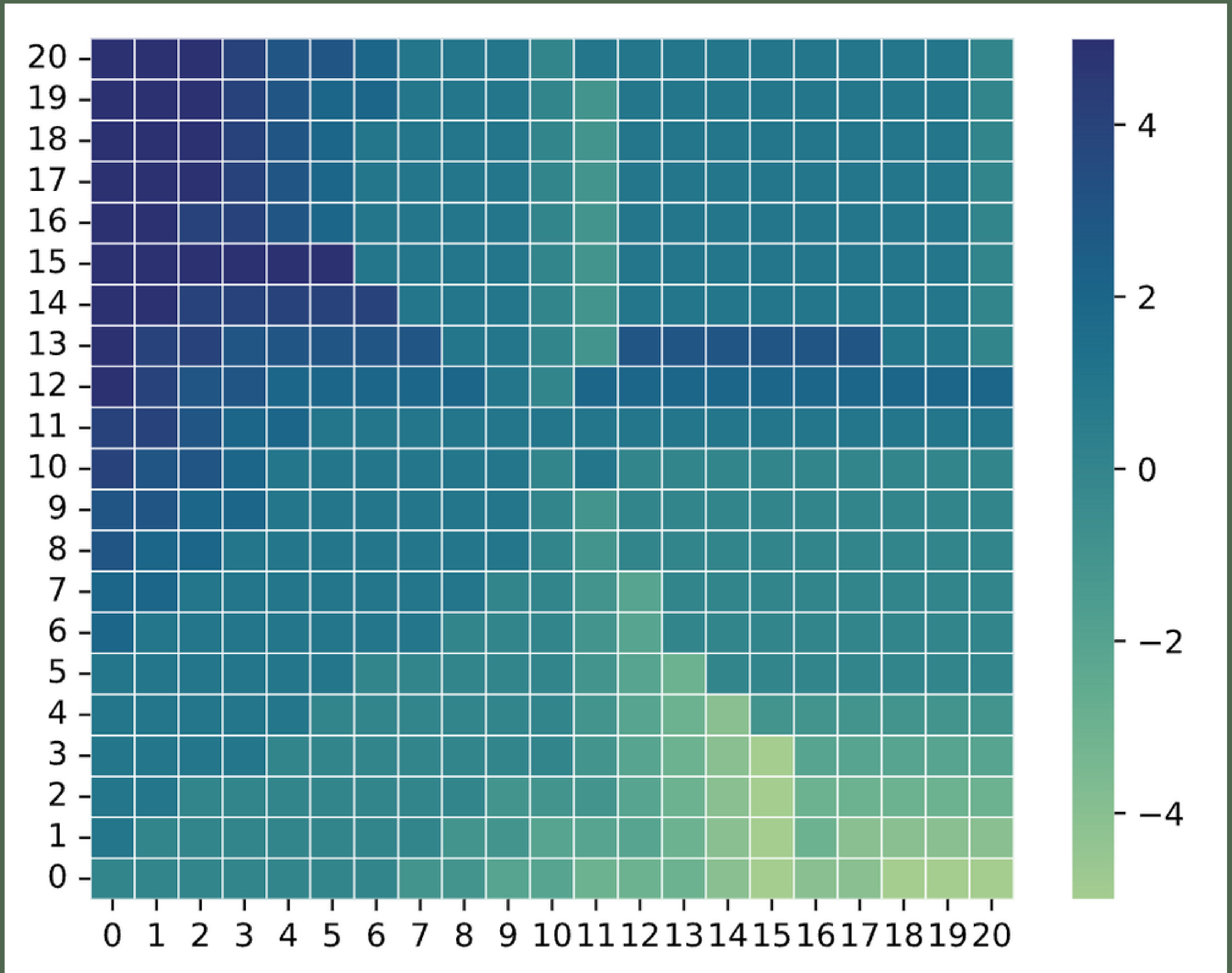


Policy Iteration

Results

This is final policy that our code converged to.

No. of cars in location 1 vs No. of cars in location 2



References

1) Reinforcement Learning: An Introduction, Richard S. Sutton and Andrew G. Barto

2) Artificial Intelligence A Modern Approach, Stuart Russell and Peter Norvig

3) https://towardsdatascience.com/elucidating-policy-iteration-in-reinforcement-learning-jacks-car-rental-problem-d41b34c8aec7