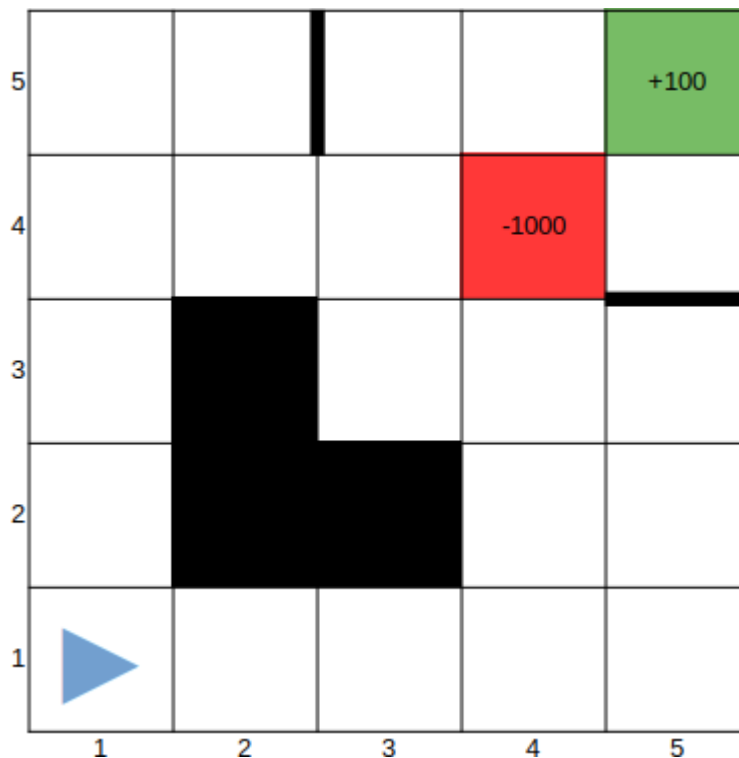


CPSC 4420/6420: ARTIFICIAL INTELLIGENCE

ASSIGNMENT 2

NAME: Charanjit Singh

Consider the following puzzle. The green and red states are both terminal states, with the rewards as shown (so we can consider the green state the “goal”, and the red a “game over” state with a large negative reward). Thick borders between cells represent walls that the robot cannot cross, and the black squares contain obstacles and cannot be entered. The robot player is represented by the blue triangle, and the direction the triangle points is the way the robot is facing.



Let's represent a state with (x,y,d) , where x and y represent the horizontal and vertical positions (i.e. location), and d represents the direction the robot is facing (1: up, 2: down, 3: left, and 4: right).

The robot can take the following actions:

- A_1 : Move one cell forward in the direction it is facing. Cost: 1.5
- A_2 : Move two cells forward in the direction it is facing. Cost: 2
- A_3 : Turn to its left and stay in the same cell. Cost: 0.5
- A_4 : Turn to its right and stay in the same cell. Cost: 0.5

Note that each action has a different cost value. This can also be considered an immediate negative reward. For example, we have $R(s,A_1,s') = -1.5$. The cost is evaluated on the current state, (the state the robot is in when it begins the action, not the one it lands on after

performing the action). In the same way, the value of state $V(s)$ represents the value of the current state and you should initialize the algorithm with $V_1(5,5,x)=+100$, $V_1(4,4,x)=-1000$ (for $x=1,2,3,4$ representing the robot orientation/direction), and zero for all other states.

So, for example, if the robot is in state $(4,1,4)$, it means that it is in location $(4,1)$ and facing right. The result of possible actions for this state are as follows:

A_1 (move 1 cell forward) $\rightarrow (5,1,4)$

A_2 (move 2 cells forward) \rightarrow impossible remains in the current state $(4,1,4)$

A_3 (turn left) $\rightarrow (4,1,1)$: the robot stays in $(4,1)$ but now faces up

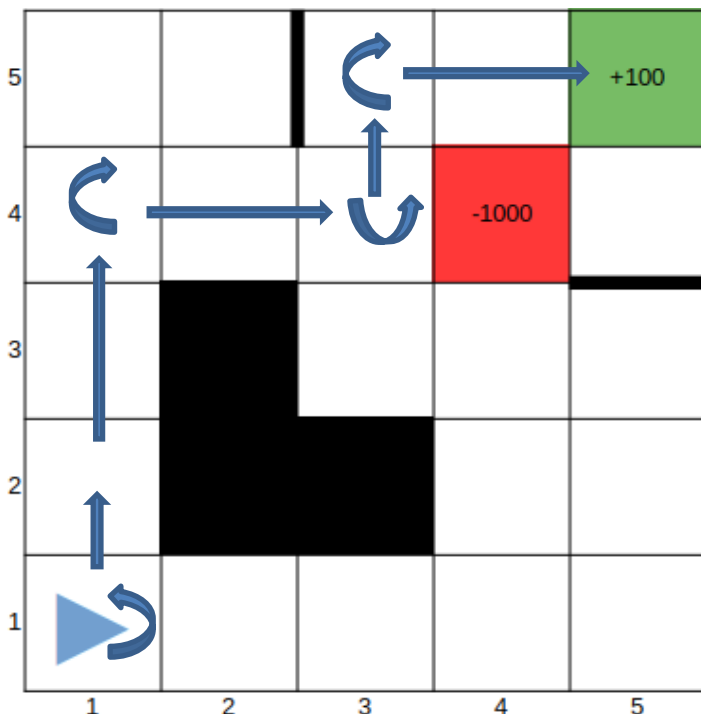
A_4 (turn right) $\rightarrow (4,1,2)$: the robot stays in $(4,1)$ but now faces down

A move is impossible if it would result in landing on a blocked cell, like $(2,2)$, $(2,3)$, or $(3,2)$, or if it would result in crossing a barrier, like moving from state $(2,5)$ to $(3,5)$, or $(5,3)$ to $(5,4)$. A move that would take the robot outside of our 5x5 grid is also impossible.

Note that we have more states than the number of cells, because the robot facing a different direction produces a new state, even if it does not change location. In the example above, if we move to $(4,1,1)$, where the robot is facing up, this is a different state from the one we were in, $(4,1,4)$, even though the robot has not moved cells.

A) If there is no living reward/penalty, no noise, and no discount ($\gamma = 1$), use your common sense to find the best possible route from $(1,1)$ to $(5,5)$.

Ans) Considering no living reward/penalty, no noise and discount factor = 1, the best possible route will be the one which costs the least. So, the following path should be followed for most optimal results:



Current State	Action	Resulting State	Cost
$(1,1,4)$	A_3	$(1,1,1)$	-0.5
$(1,1,1)$	A_1	$(1,2,1)$	-1.5
$(1,2,1)$	A_2	$(1,4,1)$	-2.0
$(1,4,1)$	A_4	$(1,4,4)$	-0.5
$(1,4,4)$	A_2	$(3,4,4)$	-2.0
$(3,4,4)$	A_3	$(3,4,1)$	-0.5
$(3,4,1)$	A_1	$(3,5,1)$	-1.5
$(3,5,1)$	A_4	$(3,5,4)$	-0.5
$(3,5,4)$	A_2	$(5,5,4)$	-2.0
Total Cost			-11

- B) With no discount ($\gamma = 1$), no living reward, and no noise, use the Value Iteration Algorithm with 100 iterations to update the optimal values for each state and print the result [only for the first 10 iterations] in the following format:

```
iter 1:
state (1,1,1) V = (some value)    Best Action: Ai (where i is some number 1-4)
state (1,1,2) V = (some value)    Best Action: Aj
...
state (5,5,4) V = (some value)

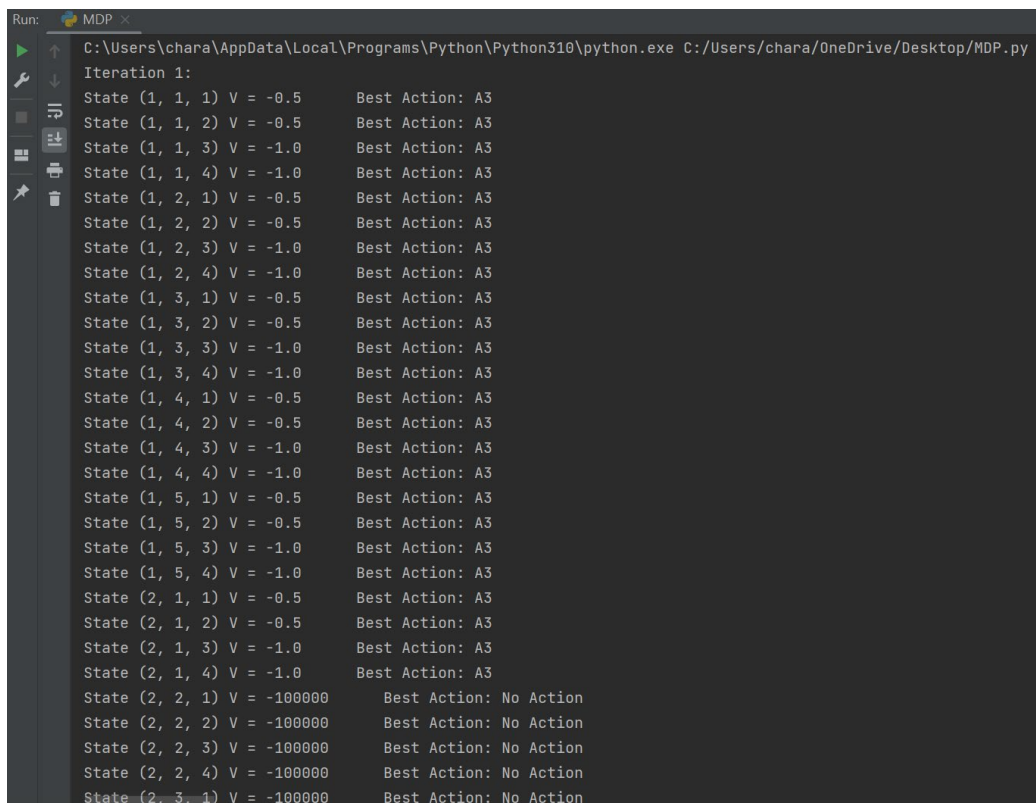
iter 2:
state (1,1,1) V = (some value)    Best Action: Ai (where i is some number 1-4)
state (1,1,2) V = (some value)    Best Action: Aj
...
state (5,5,4) V = (some value)
```

If two actions are tied for best, you can select one at random or always choose the one with the smallest index.

Ans) The result of first 10 value iterations is attached at the end.

[Click on the following hyperlink for result: Iteration Value Results](#)

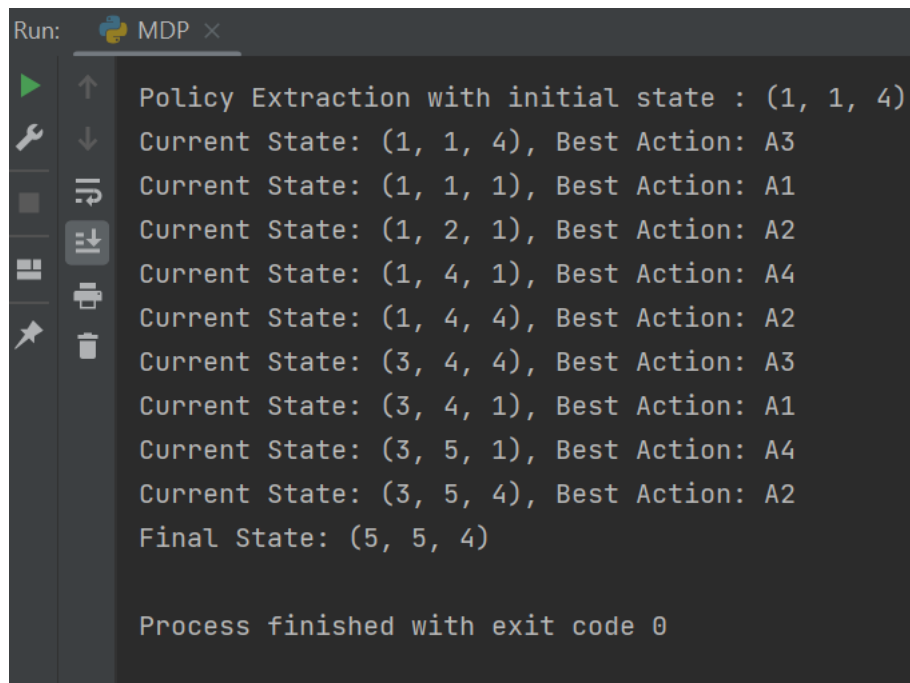
[Refer to HW2_B_Charanjit_Singh.py file for python code.](#)



```
Run: MDP x
C:\Users\chara\AppData\Local\Programs\Python\Python310\python.exe C:/Users/chara/OneDrive/Desktop/MDP.py
Iteration 1:
State (1, 1, 1) V = -0.5    Best Action: A3
State (1, 1, 2) V = -0.5    Best Action: A3
State (1, 1, 3) V = -1.0    Best Action: A3
State (1, 1, 4) V = -1.0    Best Action: A3
State (1, 2, 1) V = -0.5    Best Action: A3
State (1, 2, 2) V = -0.5    Best Action: A3
State (1, 2, 3) V = -1.0    Best Action: A3
State (1, 2, 4) V = -1.0    Best Action: A3
State (1, 3, 1) V = -0.5    Best Action: A3
State (1, 3, 2) V = -0.5    Best Action: A3
State (1, 3, 3) V = -1.0    Best Action: A3
State (1, 3, 4) V = -1.0    Best Action: A3
State (1, 4, 1) V = -0.5    Best Action: A3
State (1, 4, 2) V = -0.5    Best Action: A3
State (1, 4, 3) V = -1.0    Best Action: A3
State (1, 4, 4) V = -1.0    Best Action: A3
State (1, 5, 1) V = -0.5    Best Action: A3
State (1, 5, 2) V = -0.5    Best Action: A3
State (1, 5, 3) V = -1.0    Best Action: A3
State (1, 5, 4) V = -1.0    Best Action: A3
State (2, 1, 1) V = -0.5    Best Action: A3
State (2, 1, 2) V = -0.5    Best Action: A3
State (2, 1, 3) V = -1.0    Best Action: A3
State (2, 1, 4) V = -1.0    Best Action: A3
State (2, 2, 1) V = -100000    Best Action: No Action
State (2, 2, 2) V = -100000    Best Action: No Action
State (2, 2, 3) V = -100000    Best Action: No Action
State (2, 2, 4) V = -100000    Best Action: No Action
State (2, 3, 1) V = -100000    Best Action: No Action
```

Figure 1: Screenshot of Value Iteration results for reference

The optimal policy recommended is shown in figure 2:



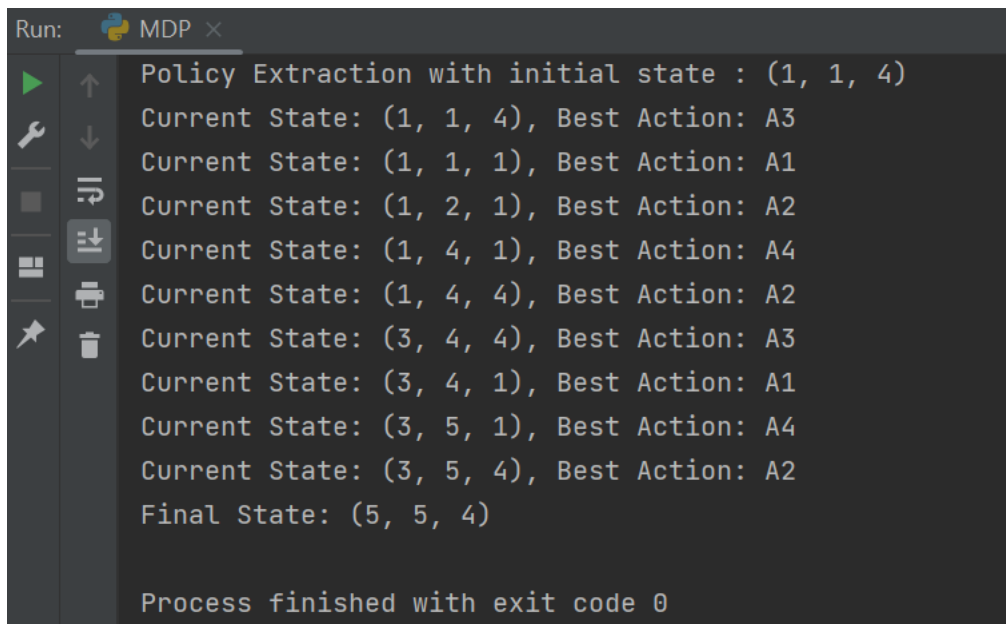
```
Run: MDP x
Policy Extraction with initial state : (1, 1, 4)
Current State: (1, 1, 4), Best Action: A3
Current State: (1, 1, 1), Best Action: A1
Current State: (1, 2, 1), Best Action: A2
Current State: (1, 4, 1), Best Action: A4
Current State: (1, 4, 4), Best Action: A2
Current State: (3, 4, 4), Best Action: A3
Current State: (3, 4, 1), Best Action: A1
Current State: (3, 5, 1), Best Action: A4
Current State: (3, 5, 4), Best Action: A2
Final State: (5, 5, 4)

Process finished with exit code 0
```

Figure 2: Optimal path recommended by the policy from state (1,1,4) to terminal state

- C) If you start from state (1,1,4) and follow the optimal policy you found in part B, does it follow the same path you proposed in part A?

Ans) **Yes**, it follows the same path as proposed in part A.



```
Run: MDP x
Policy Extraction with initial state : (1, 1, 4)
Current State: (1, 1, 4), Best Action: A3
Current State: (1, 1, 1), Best Action: A1
Current State: (1, 2, 1), Best Action: A2
Current State: (1, 4, 1), Best Action: A4
Current State: (1, 4, 4), Best Action: A2
Current State: (3, 4, 4), Best Action: A3
Current State: (3, 4, 1), Best Action: A1
Current State: (3, 5, 1), Best Action: A4
Current State: (3, 5, 4), Best Action: A2
Final State: (5, 5, 4)

Process finished with exit code 0
```

Figure 3: Screenshot of path followed by optimal policy found in part B

- D) Repeat part B with the same assumptions, except for $\gamma = 0.8$ (discount factor). Compare the results with that from part B. Do they match?

Ans) The result of first 10 value iterations is attached at the end.
Click on the following hyperlink for result: [Iteration Value Results](#)
Refer to HW2_D_Charanjit_Singh.py file for python code.

Comparing the results with part B, although the values of V^* are different for cases when discount factor = 1 (for Part B) or discount factor = 0.8 (for Part D).
But, the best action obtained by following the path with maximum expected utility, i.e.

$$\pi^*(s) = \arg \max_a Q^*(s, a)$$

is **same for both parts B and D**.

So, the action A to be followed in a particular state S, will be the same if we choose to follow policy obtained in part D or in part B.

```
Run: MDP
C:\Users\chara\AppData\Local\Programs\Python\Python310\python.exe C:/Users/chara/OneDrive/Desktop/MDP.py
Iteration 1:
State (1, 1, 1) V = -0.5 Best Action: A3
State (1, 1, 2) V = -0.5 Best Action: A3
State (1, 1, 3) V = -0.9 Best Action: A3
State (1, 1, 4) V = -0.9 Best Action: A3
State (1, 2, 1) V = -0.5 Best Action: A3
State (1, 2, 2) V = -0.5 Best Action: A3
State (1, 2, 3) V = -0.9 Best Action: A3
State (1, 2, 4) V = -0.9 Best Action: A3
State (1, 3, 1) V = -0.5 Best Action: A3
State (1, 3, 2) V = -0.5 Best Action: A3
State (1, 3, 3) V = -0.9 Best Action: A3
State (1, 3, 4) V = -0.9 Best Action: A3
State (1, 4, 1) V = -0.5 Best Action: A3
State (1, 4, 2) V = -0.5 Best Action: A3
State (1, 4, 3) V = -0.9 Best Action: A3
State (1, 4, 4) V = -0.9 Best Action: A3
State (1, 5, 1) V = -0.5 Best Action: A3
State (1, 5, 2) V = -0.5 Best Action: A3
State (1, 5, 3) V = -0.9 Best Action: A3
State (1, 5, 4) V = -0.9 Best Action: A3
State (2, 1, 1) V = -0.5 Best Action: A3
State (2, 1, 2) V = -0.5 Best Action: A3
State (2, 1, 3) V = -0.9 Best Action: A3
State (2, 1, 4) V = -0.9 Best Action: A3
State (2, 2, 1) V = -100000 Best Action: No Action
State (2, 2, 2) V = -100000 Best Action: No Action
```

Figure 4: Screenshot of Value Iteration results for reference

```
Run: MDP
Policy Extraction with initial state : (1, 1, 4)
Current State: (1, 1, 4), Best Action: A3
Current State: (1, 1, 1), Best Action: A1
Current State: (1, 2, 1), Best Action: A2
Current State: (1, 4, 1), Best Action: A4
Current State: (1, 4, 4), Best Action: A2
Current State: (3, 4, 4), Best Action: A3
Current State: (3, 4, 1), Best Action: A1
Current State: (3, 5, 1), Best Action: A4
Current State: (3, 5, 4), Best Action: A2
Final State: (5, 5, 4)

Process finished with exit code 0
```

Path is same as seen in figure 2

Figure 5: Optimal Path for discount factor = 0.8

- E) Repeat part B with the same assumptions, except for $\gamma = 0.2$. Compare the results with that from parts B and D. Do they match?

Ans) The result of first 10 value iterations is attached at the end.

Click on the following hyperlink for result: [Iteration Value Results](#)

Refer to HW2_E_Charanjit_Singh.py file for python code.

Comparing with results from Parts B and D, the results (optimal paths recommended) do **NOT** match.

As the value of discount factor (γ) has decreased considerably, the utility of future rewards has reduced drastically owing to discounting of rewards.

$$U_{\pi} = \sum_{t=0}^H \gamma^t R(s_t)$$

This tends the agent more towards myopic behavior, i.e. the agent will consider the immediate rewards more strongly as compared to future rewards (as the reward component of equation is weighed more).

$$V^*(s) = \max_a (R(s, a, s') + \gamma V^*(s'))$$

So, for the starting states (1,1,1), (1,1,2), (1,1,3) and (1,1,4), the best action recommended at V^* values convergence is A3 (which costs the least [-0.5]).

This makes the agent fall in an infinite loop of rotating left while in the cell (1,1), as can be seen in figure 7.

```

Run: MDP
C:\Users\chara\AppData\Local\Programs\Python\Python310\python.exe C:/Users/chara/OneDrive/Desktop/MDP.py
Iteration 1:
State (1, 1, 1) V = -0.5 Best Action: A3
State (1, 1, 2) V = -0.5 Best Action: A3
State (1, 1, 3) V = -0.6 Best Action: A3
State (1, 1, 4) V = -0.6 Best Action: A3
State (1, 2, 1) V = -0.5 Best Action: A3
State (1, 2, 2) V = -0.5 Best Action: A3
State (1, 2, 3) V = -0.6 Best Action: A3
State (1, 2, 4) V = -0.6 Best Action: A3
State (1, 3, 1) V = -0.5 Best Action: A3
State (1, 3, 2) V = -0.5 Best Action: A3
State (1, 3, 3) V = -0.6 Best Action: A3
State (1, 3, 4) V = -0.6 Best Action: A3
State (1, 4, 1) V = -0.5 Best Action: A3
State (1, 4, 2) V = -0.5 Best Action: A3
State (1, 4, 3) V = -0.6 Best Action: A3
State (1, 4, 4) V = -0.6 Best Action: A3
State (1, 5, 1) V = -0.5 Best Action: A3
State (1, 5, 2) V = -0.5 Best Action: A3
State (1, 5, 3) V = -0.6 Best Action: A3
State (1, 5, 4) V = -0.6 Best Action: A3
State (2, 1, 1) V = -0.5 Best Action: A3
State (2, 1, 2) V = -0.5 Best Action: A3
State (2, 1, 3) V = -0.6 Best Action: A3
State (2, 1, 4) V = -0.6 Best Action: A3
State (2, 2, 1) V = -100000 Best Action: No Action
State (2, 2, 2) V = -100000 Best Action: No Action

```

Figure 6: Screenshot of Value Iteration results for reference

```

Run: MDP
Current State: (1, 1, 4), Best Action: A3
Current State: (1, 1, 1), Best Action: A3
Current State: (1, 1, 3), Best Action: A3
Current State: (1, 1, 2), Best Action: A3
Current State: (1, 1, 4), Best Action: A3
Current State: (1, 1, 1), Best Action: A3
Current State: (1, 1, 3), Best Action: A3
Current State: (1, 1, 2), Best Action: A3
Current State: (1, 1, 4), Best Action: A3
Current State: (1, 1, 1), Best Action: A3
Current State: (1, 1, 3), Best Action: A3
Current State: (1, 1, 2), Best Action: A3
Current State: (1, 1, 4), Best Action: A3
Current State: (1, 1, 1), Best Action: A3
Current State: (1, 1, 3), Best Action: A3
Current State: (1, 1, 2), Best Action: A3
Current State: (1, 1, 4), Best Action: A3
Current State: (1, 1, 1), Best Action: A3
Current State: (1, 1, 3), Best Action: A3
Current State: (1, 1, 2), Best Action: A3
Current State: (1, 1, 4), Best Action: A3

```

Figure 7: Path followed by agent for discount factor = 0.2 (infinite loop)

- F) **(Optional for 4420)** Repeat part B, but this time with noise = 0.2, and gamma = 0.9 and no living reward. With a noise of 0.2, every time you take an action, the result will be the expected action with Probability 0.8 (80%), but 20% of the time, the robot will instead take a different action (taken randomly out of unexpected actions, with equal probability). If the action is impossible, it remains in the same cell.

For example, if we are in state (4,1,4), location (4,1) and facing right, and we take action A_1 (moving one cell forward), the resulting state will be:

$s' = (5,1,2)$ with probability 0.8 [because A_1 is rendered]
 $s' = (4,1,4)$ with probability 0.2/3 [renders A_2 which is impossible]
 $s' = (4,1,1)$ with probability 0.2/3 [because A_3 is rendered]
 $s' = (4,1,2)$ with probability 0.2/3 [because A_4 is rendered]

Compare the results with that of the previous parts and explain observations.

Ans) The result of first 10 value iterations is attached at the end.

Click on the following hyperlink for result: [Iteration Value Results](#)

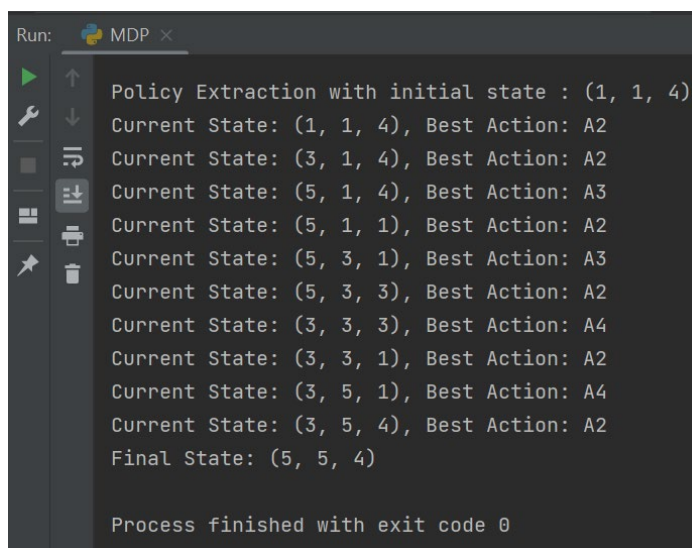
Refer to HW2_F_Charanjit_Singh.py file for python code.

In the previous parts, there was no noise which meant the actions were executed with a probability of 1. As in this probabilistic case, there is a noise of 20%, there is a risk of falling in negative terminal state (4,4) which makes the agent skip the states (3,4,4) and (4,3,1) (which now have a negative V^*).

$$V^*(3,4,4) = -20.56, V^*(4,3,1) = -35.27$$

Considering initial state as (1,1,4), the agent adopts a different path, via (3,1,4) to (5,1,1) It avoids going upwards as it will eventually come across (3,4,4). It avoids going through (4,1,1) as it will eventually come across (4,3,1).

So, it follows the safest option available to it which is guaranteed to give maximum expected utility. The path is shown in figure 8.



```
Run: MDP x
Policy Extraction with initial state : (1, 1, 4)
Current State: (1, 1, 4), Best Action: A2
Current State: (3, 1, 4), Best Action: A2
Current State: (5, 1, 4), Best Action: A3
Current State: (5, 1, 1), Best Action: A2
Current State: (5, 3, 1), Best Action: A3
Current State: (5, 3, 3), Best Action: A2
Current State: (3, 3, 3), Best Action: A4
Current State: (3, 3, 1), Best Action: A2
Current State: (3, 5, 1), Best Action: A4
Current State: (3, 5, 4), Best Action: A2
Final State: (5, 5, 4)

Process finished with exit code 0
```

Figure 8: Optimal path followed for discount factor = 0.9 and noise = 0.2

Markov Decision Process - Value Iteration Algorithm


```

1  '''
2  Code Introduction:
3  The defined class MDP takes 2 positional arguments:
4  Discount factor, Noise
5  This class has the following inbuilt functions built
6  in them:
7  1. Transition Model: Takes current state and action
8  as inputs and returns resulting state.
9  2. q_value function: Takes state and action as inputs
10 and returns q value
11 3. value_iterations function: prints Q*(s,a) value
12 for all states along with the best recommended action
13 4. policy function: Takes current state as input and
14 prints out the optimal path from current state to
15 terminal state
16 '''
17
18 # Defining a class MDP which takes discount factor
19 and noise as inputs
20 # i.e. inputs: 0 < Discount Factor, Noise < 1
21 class Mdp:
22     def __init__(self, discount_factor, noise):
23         self.discount_factor = discount_factor
24         self.noise = noise
25         self.actions = ["A1", "A2", "A3", "A4"]
26         self.action_cost = {
27             "A1": -1.5,
28             "A2": -2,
29             "A3": -0.5,
30             "A4": -0.5
31         }
32         self.state_value = {}
33         self.new_state_value_actions = {}
34         self.states_best_actions = {}
35         self.new_state_value = {}
36
37     # Transition model which takes current state and
38     action as inputs and returns resulting state
39     # Input state in format (column, row, direction)
40     # Input action as "Ai" where i = 1,2,3,4
41     def transition_model(self, state, action):

```

```

33         column_initial, row_initial,
        direction_initial = state
34
35         # Defining inaccessible states for agent
36         blocked_states = [(2, 2), (2, 3), (3, 2)]
37         blocked_moves = []
38         for z, y in blocked_states:
39             for i in [1, 2, 3, 4]:
40                 blocked_moves.append((z,y,i))
41
42         # returning initial state if agent tries to
run in barriers
43         if (column_initial, row_initial,
        direction_initial) in [(2, 5, 4), (3, 5, 3), (5, 3, 1
        ), (5, 4, 2)]:
44             if action in ["A1", "A2"]:
45                 return column_initial, row_initial,
        direction_initial
46
47         # Assigning new direction to the agent
depending on action A3 or A4
48         rotating_actions = [1, 3, 2, 4, 1, 3, 2, 4]
49         if action == "A3":
50             direction = rotating_actions[
        rotating_actions.index(direction_initial)+1]
51         elif action == "A4":
52             direction = rotating_actions[
        rotating_actions.index(direction_initial)-1]
53         else:
54             direction = direction_initial
55
56         # Defining the effect of actions A1 and A2 in
initial direction 1/2/3/4
57         steps = 1
58         action_definition = {
59             1: (column_initial, (row_initial+steps),
        direction),
60             2: (column_initial, row_initial-steps,
        direction),
61             3: ((column_initial-steps), row_initial,
        direction),

```

```

62         4: (column_initial+steps, row_initial,
        direction)
63     }
64
65     '''
66     Defining the number of steps agent should
        take in a particular action i.e.
67     for A1, no. of steps = 1 in facing direction
68     for A2, no. of steps = 2 in facing direction
69     for A3 and A4, no steps
70     '''
71     if action == 'A1':
72         steps = 1
73     elif action == 'A2':
74         (c, r, d) = action_definition[
        direction_initial]
75         if (c, r) in blocked_states or (c, r) in
        [(4, 4), (5, 5), (5, 3), (2, 5), (3, 5)]:
76             return column_initial, row_initial,
        direction_initial
77         else:
78             steps = 2
79     else:
80         steps = 0
81
82     # Updating the dictionary to according to
        the number of steps the agent should take
83     action_definition = {
84         1: (column_initial, (row_initial+steps
        ), direction),
85         2: (column_initial, row_initial-steps,
        direction),
86         3: ((column_initial-steps), row_initial
        , direction),
87         4: (column_initial+steps, row_initial,
        direction)
88     }
89     # Assigning the appropriate resulting stage
        after factoring in the effect of action on initial
        state
90     resulting_state = action_definition[

```

```

90 direction_initial]
91         (column, row, direction) = resulting_state
92
93         # Filtering out the result in case action is
making the agent fall out of the grid
94         if (resulting_state in blocked_moves) or row
> 5 or column > 5 or row < 1 or column < 1:
95             return column_initial, row_initial,
direction_initial
96         else:
97             return resulting_state
98
99     '''
100     Function to calculate Q value, which returns the
expected value of utility for a particular action a
in
101     a state s.
102     '''
103     def q_value(self, state, action):
104         actions = ["A1", "A2", "A3", "A4"]
105         actions.remove(action)
106         qval = (1-self.noise)*(self.action_cost[
action] +
107                     self.discount_factor*
self.state_value[self.transition_model(state, action
)])+\
108         (self.noise/3)*(self.action_cost[actions[0
]] +
109                     self.discount_factor*self.
state_value[self.transition_model(state, actions[0
]])+\
110         (self.noise/3)*(self.action_cost[actions[1
]] +
111                     self.discount_factor*self.
state_value[self.transition_model(state,actions[1
]])+\
112         (self.noise/3)*(self.action_cost[actions[2
]] +
113                     self.discount_factor*self.
state_value[self.transition_model(state,actions[2
]]))

```

```

114         return qval
115
116         # Defining a Value Iteration function which
prints first 10 iterations and final values after
100 iterations
117         def value_iterations(self):
118             states = []
119             # Initializing the states list containing
all states on the grid
120             for i in [1, 2, 3, 4, 5]:
121                 for t in [1, 2, 3, 4, 5]:
122                     for robot_direction in [1, 2, 3, 4]:
123                         states.append((i, t,
robot_direction))
124
125                 # Assign an initial value of 0 to all states
126                 for x in states:
127                     self.state_value[x] = 0
128
129                 # For terminal and blocked states, assign
suitable values
130                 for col, ro, cost in [(4, 4, -1000), (5, 5,
100), (2, 3, -100000), (2, 2, -100000), (3, 2, -
100000)]:
131                     for d in [1, 2, 3, 4]:
132                         self.state_value[col, ro, d] = cost
133
134                 # Value Iteration containing 100 iterations
135                 for i in range(100):
136                     if i < 10:
137                         print(f'Iteration {i+1}:')
138                     for a, b, c in states:
139                         # If state is blocked/terminal,
value is fixed
140                         if (a, b) in [(4, 4), (5, 5), (3, 2
), (2, 2), (2, 3)]:
141                             self.new_state_value_actions[a,
b, c] = (self.state_value[a, b, c], "No Action")
142                             # for accessible states, value
should be updated in subsequent iterations
143                             else:

```

```

144         self.new_state_value[a, b, c
    ] = [round(self.q_value((a, b, c), act), 2) for act
    in
145         ["A1", "A2", "A3", "A4"]]
146         self.new_state_value_actions[a,
    b, c] = max(self.new_state_value[a, b, c]),\
147             self.actions[(self.
    new_state_value[a, b, c]).index(max(self.
    new_state_value[a, b, c]))]
148         # Forming a dictionary
states_best_actions to keep track of best action in
a particular state
149         val, act = self.
    new_state_value_actions[a, b, c]
150         self.states_best_actions[(a,b,c)] =
    f'State {a,b,c} V = {val}      Best Action: {act}'
151         # Updating the state value
dictionary with new values
152         self.state_value[a, b, c] = val
153         # Printing the first 10 value iterations
154         if i < 10:
155             for key, value in self.
    states_best_actions.items():
156                 print(value)
157             i += 1
158             print(f'\n(Values, Best Action) after 100
    iterations: {self.new_state_value_actions}')
159
160     '''
161     Defining a policy function with input: current
state
162     It returns the optimal path to reach the
terminal state from current state
163     '''
164     def policy(self, state):
165         print(f"\nPolicy Extraction with initial
state : {state}")
166         (c, r, d) = state
167         while (c, r) not in [(4, 4), (5, 5)]:
168             (v, ac) = self.new_state_value_actions[(

```

```
168 c, r, d)]
169         print(f'Current State: {state}, Best
      Action: '
170             f'{self.actions[(self.
      new_state_value[c,r,d]).index(max(self.
      new_state_value[c,r,d]))]}')
171         state = self.transition_model((c, r, d
      ), ac)
172         (c, r, d) = state
173         print(f'Final State: {state}')
174
175 # initialize a class instance: puzzle = Mdp(1, 0)
176 # print value iterations using puzzle.
      value_iterations()
177 # Print policy: puzzle.policy((1, 1, 4))
```

Value Iteration Answers

State (4, 1, 4) V = 90.5	Best Action: A3
State (4, 2, 1) V = 91.5	Best Action: A1
State (4, 2, 2) V = 90.5	Best Action: A3
State (4, 2, 3) V = 91.0	Best Action: A4
State (4, 2, 4) V = 91.0	Best Action: A3
State (4, 3, 1) V = 93.0	Best Action: A3
State (4, 3, 2) V = 93.0	Best Action: A4
State (4, 3, 3) V = 93.5	Best Action: A1
State (4, 3, 4) V = 93.5	Best Action: A3
State (4, 4, 1) V = -1000	Best Action: No Action
Action	
State (4, 4, 2) V = -1000	Best Action: No Action
State (4, 4, 3) V = -1000	Best Action: No Action
Action	
State (4, 4, 4) V = -1000	Best Action: No Action
Action	
State (4, 5, 1) V = 98.0	Best Action: A4
State (4, 5, 2) V = 98.0	Best Action: A3
State (4, 5, 3) V = 98.5	Best Action: A3
State (4, 5, 4) V = 97.5	Best Action: A1
State (5, 1, 1) V = 90.5	Best Action: A2
State (5, 1, 2) V = 89.5	Best Action: A3
State (5, 1, 3) V = 90.0	Best Action: A4
State (5, 1, 4) V = 90.0	Best Action: A3
State (5, 2, 1) V = 91.0	Best Action: A1
State (5, 2, 2) V = 91.0	Best Action: A3
State (5, 2, 3) V = 90.5	Best Action: A4
State (5, 2, 4) V = 90.5	Best Action: A3
State (5, 3, 1) V = 92.5	Best Action: A3
State (5, 3, 2) V = 92.5	Best Action: A4
State (5, 3, 3) V = 93.0	Best Action: A2
State (5, 3, 4) V = 92.0	Best Action: A3
State (5, 4, 1) V = 98.5	Best Action: A1
State (5, 4, 2) V = 97.5	Best Action: A3
State (5, 4, 3) V = 98.0	Best Action: A4
State (5, 4, 4) V = 98.0	Best Action: A3
State (5, 5, 1) V = 100	Best Action: No Action
State (5, 5, 2) V = 100	Best Action: No Action
State (5, 5, 3) V = 100	Best Action: No Action
State (5, 5, 4) V = 100	Best Action: No Action
Iteration 10:	
State (1, 1, 1) V = 89.5	Best Action: A1
State (1, 1, 2) V = 89.5	Best Action: A3
State (1, 1, 3) V = 88.0	Best Action: A4
State (1, 1, 4) V = 89.0	Best Action: A3
State (1, 2, 1) V = 91.0	Best Action: A2
State (1, 2, 2) V = 90.0	Best Action: A3
State (1, 2, 3) V = 90.5	Best Action: A4
State (1, 2, 4) V = 90.5	Best Action: A3
State (1, 3, 1) V = 91.5	Best Action: A1
State (1, 3, 2) V = 91.5	Best Action: A3
State (1, 3, 3) V = 91.0	Best Action: A4
State (1, 3, 4) V = 91.0	Best Action: A3
State (1, 4, 1) V = 93.0	Best Action: A4
State (1, 4, 2) V = 93.0	Best Action: A3
State (1, 4, 3) V = 92.5	Best Action: A3
State (1, 4, 4) V = 92.5	Best Action: A2
State (1, 5, 1) V = 90.5	Best Action: A3
State (1, 5, 2) V = 91.5	Best Action: A1
State (1, 5, 3) V = 91.0	Best Action: A3
State (1, 5, 4) V = 91.0	Best Action: A4
State (2, 1, 1) V = 88.0	Best Action: A3
State (2, 1, 2) V = 88.0	Best Action: A4
State (2, 1, 3) V = 87.5	Best Action: A1
State (2, 1, 4) V = 88.5	Best Action: A2
State (2, 2, 1) V = -100000	Best Action: No Action
Action	
State (2, 2, 2) V = -100000	Best Action: No Action
Action	
State (2, 2, 3) V = -100000	Best Action: No Action
Action	
State (2, 2, 4) V = -100000	Best Action: No Action
Action	
State (2, 3, 1) V = -100000	Best Action: No Action
Action	
State (2, 3, 2) V = -100000	Best Action: No Action
Action	
State (2, 3, 3) V = -100000	Best Action: No Action
Action	
State (2, 3, 4) V = -100000	Best Action: No Action
Action	
State (2, 4, 1) V = 93.5	Best Action: A4
State (2, 4, 2) V = 93.5	Best Action: A3
State (2, 4, 3) V = 93.0	Best Action: A3
State (2, 4, 4) V = 94.0	Best Action: A1
State (2, 5, 1) V = 91.0	Best Action: A3
State (2, 5, 2) V = 92.0	Best Action: A1
State (2, 5, 3) V = 91.5	Best Action: A3
State (2, 5, 4) V = 91.5	Best Action: A4
State (3, 1, 1) V = 88.5	Best Action: A4
State (3, 1, 2) V = 88.5	Best Action: A3
State (3, 1, 3) V = 88.0	Best Action: A3
State (3, 1, 4) V = 89.0	Best Action: A1
State (3, 2, 1) V = -10000	Best Action: No Action
Action	
State (3, 2, 2) V = -10000	Best Action: No Action
Action	
State (3, 2, 3) V = -10000	Best Action: No Action
Action	
State (3, 2, 4) V = -10000	Best Action: No Action
Action	
State (3, 3, 1) V = 95.5	Best Action: A2
State (3, 3, 2) V = 94.5	Best Action: A3
State (3, 3, 3) V = 95.0	Best Action: A4
State (3, 3, 4) V = 96.0	Best Action: A3
State (3, 4, 1) V = 95.0	Best Action: A1
State (3, 4, 2) V = 95.0	Best Action: A3
State (3, 4, 3) V = 95.5	Best Action: A4
State (3, 4, 4) V = 95.5	Best Action: A3
State (3, 5, 1) V = 97.5	Best Action: A4
State (3, 5, 2) V = 97.5	Best Action: A3
State (3, 5, 3) V = 97.0	Best Action: A3
State (3, 5, 4) V = 98.0	Best Action: A2
State (4, 1, 1) V = 91.0	Best Action: A2
State (4, 1, 2) V = 90.0	Best Action: A3
State (4, 1, 3) V = 90.5	Best Action: A4
State (4, 1, 4) V = 90.5	Best Action: A3
State (4, 2, 1) V = 91.5	Best Action: A1
State (4, 2, 2) V = 91.0	Best Action: A3
State (4, 2, 3) V = 91.0	Best Action: A4
State (4, 2, 4) V = 91.0	Best Action: A3
State (4, 3, 1) V = 93.0	Best Action: A3
State (4, 3, 2) V = 93.0	Best Action: A4
State (4, 3, 3)	

State (4, 1, 4) V = 12.19	Best Action: A3
State (4, 2, 1) V = 16.36	Best Action: A1
State (4, 2, 2) V = 9.57	Best Action: A3
State (4, 2, 3) V = 12.59	Best Action: A4
State (4, 2, 4) V = 12.59	Best Action: A3
State (4, 3, 1) V = 22.32	Best Action: A3
State (4, 3, 2) V = 22.32	Best Action: A4
State (4, 3, 3) V = 28.52	Best Action: A1
State (4, 3, 4) V = 17.36	Best Action: A3
State (4, 4, 1) V = -1000	Best Action: No Action
State (4, 4, 2) V = -1000	Best Action: No Action
State (4, 4, 3) V = -1000	Best Action: No Action
State (4, 4, 4) V = -1000	Best Action: No Action
State (4, 5, 1) V = 62.3	Best Action: A4
State (4, 5, 2) V = 62.3	Best Action: A3
State (4, 5, 3) V = 49.34	Best Action: A3
State (4, 5, 4) V = 78.5	Best Action: A1
State (5, 1, 1) V = 15.54	Best Action: A2
State (5, 1, 2) V = 9.04	Best Action: A3
State (5, 1, 3) V = 11.93	Best Action: A4
State (5, 1, 4) V = 11.93	Best Action: A3
State (5, 2, 1) V = 16.04	Best Action: A1
State (5, 2, 2) V = 9.36	Best Action: A3
State (5, 2, 3) V = 12.33	Best Action: A4
State (5, 2, 4) V = 12.33	Best Action: A3
State (5, 3, 1) V = 21.92	Best Action: A3
State (5, 3, 2) V = 21.92	Best Action: A4
State (5, 3, 3) V = 28.02	Best Action: A2
State (5, 3, 4) V = 17.04	Best Action: A3
State (5, 4, 1) V = 78.5	Best Action: A1
State (5, 4, 2) V = 49.34	Best Action: A3
State (5, 4, 3) V = 62.3	Best Action: A4
State (5, 4, 4) V = 62.3	Best Action: A3
State (5, 5, 1) V = 100	Best Action: No Action
State (5, 5, 2) V = 100	Best Action: No Action
State (5, 5, 3) V = 100	Best Action: No Action
State (5, 5, 4) V = 100	Best Action: No Action
Iteration 10:	
State (1, 1, 1) V = 11.09	Best Action: A1
State (1, 1, 2) V = 6.2	Best Action: A3
State (1, 1, 3) V = 8.37	Best Action: A4
State (1, 1, 4) V = 8.37	Best Action: A3
State (1, 2, 1) V = 15.74	Best Action: A2
State (1, 2, 2) V = 9.17	Best Action: A3
State (1, 2, 3) V = 12.09	Best Action: A4
State (1, 2, 4) V = 12.09	Best Action: A3
State (1, 3, 1) V = 16.24	Best Action: A1
State (1, 3, 2) V = 9.49	Best Action: A3
State (1, 3, 3) V = 12.49	Best Action: A4
State (1, 3, 4) V = 12.49	Best Action: A3
State (1, 4, 1) V = 22.17	Best Action: A4
State (1, 4, 2) V = 22.17	Best Action: A3
State (1, 4, 3) V = 17.24	Best Action: A3
State (1, 4, 4) V = 28.34	Best Action: A2
State (1, 5, 1) V = 9.49	Best Action: A3
State (1, 5, 2) V = 16.24	Best Action: A1
State (1, 5, 3) V = 12.49	Best Action: A3
State (1, 5, 4) V = 12.49	Best Action: A4
State (2, 1, 1) V = 5.7	Best Action: A4
State (2, 1, 2) V = 5.7	Best Action: A3
State (2, 1, 3) V = 5.2	Best Action: A1
State (2, 1, 4) V = 7.75	Best Action: A2
State (2, 2, 1) V = -100000	Best Action: No Action
State (2, 2, 2) V = -100000	Best Action: No Action
State (2, 2, 3) V = -100000	Best Action: No Action
State (2, 2, 4) V = -100000	Best Action: No Action
State (2, 3, 1) V = -100000	Best Action: No Action
State (2, 3, 2) V = -100000	Best Action: No Action
State (2, 3, 3) V = -100000	Best Action: No Action
State (2, 3, 4) V = -100000	Best Action: No Action
State (2, 4, 1) V = 22.57	Best Action: A4
State (2, 4, 2) V = 22.57	Best Action: A3
State (2, 4, 3) V = 17.56	Best Action: A3
State (2, 4, 4) V = 28.84	Best Action: A1
State (2, 5, 1) V = 9.7	Best Action: A3
State (2, 5, 2) V = 16.56	Best Action: A1
State (2, 5, 3) V = 12.75	Best Action: A3
State (2, 5, 4) V = 12.75	Best Action: A4
State (3, 1, 1) V = 6.1	Best Action: A4
State (3, 1, 2) V = 6.1	Best Action: A3
State (3, 1, 3) V = 4.7	Best Action: A2
State (3, 1, 4) V = 8.25	Best Action: A1
State (3, 2, 1) V = -100000	Best Action: No Action
State (3, 2, 2) V = -100000	Best Action: No Action
State (3, 2, 3) V = -100000	Best Action: No Action
State (3, 2, 4) V = -100000	Best Action: No Action
State (3, 3, 1) V = 47.52	Best Action: A2
State (3, 3, 2) V = 29.52	Best Action: A3
State (3, 3, 3) V = 37.52	Best Action: A4
State (3, 3, 4) V = 37.52	Best Action: A3
State (3, 4, 1) V = 48.02	Best Action: A1
State (3, 4, 2) V = 29.84	Best Action: A3
State (3, 4, 3) V = 37.92	Best Action: A4
State (3, 4, 4) V = 37.92	Best Action: A3
State (3, 5, 1) V = 61.9	Best Action: A4
State (3, 5, 2) V = 61.9	Best Action: A3
State (3, 5, 3) V = 49.02	Best Action: A3
State (3, 5, 4) V = 78.0	Best Action: A2
State (4, 1, 1) V = 15.86	Best Action: A2
State (4, 1, 2) V = 9.25	Best Action: A3
State (4, 1, 3) V = 12.19	Best Action: A4
State (4, 1, 4) V = 12.19	Best Action: A3
State (4, 2, 1) V = 16.36	Best Action: A1
State (4, 2, 2) V = 9.57	Best Action: A3
State (4, 2, 3) V = 12.59	Best Action: A4
State (4, 2, 4) V = 12.59	Best Action: A3
State (4, 3, 1) V = 22.32	Best Action: A3
State (4, 3, 2) V = 22.32	Best Action: A4
State (4, 3, 3) V = 28.52	Best Action: A1
State (4, 3, 4) V = 17.36	Best Action: A3
State (4, 4, 1) V = -1000	Best Action: No Action
State (4, 4, 2) V = -1000	Best Action: No Action
State (4, 4, 3) V = -1000	Best Action: No Action
State (4, 4, 4) V = -1000	Best Action: No Action
State (4, 5, 1) V = 62.3	Best Action: A4
State (4, 5, 2) V = 62.3	Best Action: A3
State (4, 5, 3) V = 49.34	Best Action: A3
State (4, 5, 4) V = 78.5	Best Action: A1
State (5, 1, 1) V = 15.54	Best Action: A2
State (5, 1, 2) V = 9.04	Best Action: A3
State (5, 1, 3) V = 11.93	Best Action: A4
State (5, 1, 4) V = 11.93	Best Action: A3
State (5, 2, 1) V = 16.04	Best Action: A1
State (5, 2, 2) V = 9.36	Best Action: A3
State (5, 2, 3) V = 12.33	Best Action: A4
State (5, 2, 4) V = 12.33	Best Action: A3
State (5, 3, 1) V = 21.92	Best Action: A3
State (5, 3, 2) V = 21.92	Best Action: A4
State (5, 3, 3) V = 28.02	Best Action: A2
State (5, 3, 4) V = 17.04	Best Action: A3
State (5, 4, 1) V = 78.5	Best Action: A1
State (5, 4, 2) V = 49.34	Best Action: A3
State (5, 4, 3) V = 62.3	Best Action: A4
State (5, 4, 4) V = 62.3	Best Action: A3
State (5, 5, 1) V = 100	Best Action: No Action
State (5, 5, 2) V = 100	Best Action: No Action
State (5, 5, 3) V = 100	Best Action: No Action
State (5, 5, 4) V = 100	Best Action: No Action

[illegible]

Ans F)

Iteration 1:		State (3, 1, 4) V = -2.12		Best Action: A3		State (5, 2, 4) V = -2.92		Best Action: A3		State (2, 3, 4) V = -100000		Best Action: No Action		State (4, 4, 4) V = -1000		Best Action: No Action		State (2, 2, 2) V = -100000		Best Action: No Action			
State (1, 1, 1) V = -0.67		Best Action: A3		State (3, 1, 2) V = -2.57		Best Action: A3		State (5, 3, 1) V = -2.57		Best Action: A3		State (2, 4, 1) V = -8.77		Best Action: A1		State (4, 5, 1) V = 67.88		Best Action: A4		State (2, 2, 2) V = -100000		Best Action: No Action	
State (1, 1, 2) V = -0.67		Best Action: A3		State (3, 2, 1) V = -100000		Best Action: No Action		State (5, 3, 2) V = -2.67		Best Action: A3		State (2, 4, 2) V = -10.11		Best Action: A4		State (4, 5, 2) V = -0.25		Best Action: A3		State (2, 2, 3) V = -100000		Best Action: No Action	
State (1, 1, 3) V = -1.19		Best Action: A3		State (3, 2, 2) V = -100000		Best Action: No Action		State (5, 3, 3) V = 15.08		Best Action: A2		State (2, 4, 3) V = -6.39		Best Action: A1		State (4, 5, 3) V = 55.21		Best Action: A1		State (2, 2, 4) V = -100000		Best Action: No Action	
State (1, 1, 4) V = -1.19		Best Action: A3		State (3, 2, 3) V = -100000		Best Action: No Action		State (5, 3, 4) V = -2.93		Best Action: A3		State (2, 4, 4) V = -69.36		Best Action: A3		State (4, 5, 4) V = 79.42		Best Action: A1		State (2, 2, 5) V = -100000		Best Action: No Action	
State (1, 2, 1) V = -0.71		Best Action: A3		State (3, 2, 4) V = -100000		Best Action: No Action		State (5, 4, 1) V = 78.8		Best Action: A1		State (2, 5, 1) V = -3.86		Best Action: A4		State (4, 5, 5) V = 14.38		Best Action: A2		State (2, 2, 6) V = -100000		Best Action: No Action	
State (1, 2, 2) V = -0.71		Best Action: A3		State (3, 3, 1) V = -1.68		Best Action: A3		State (5, 4, 2) V = 48.69		Best Action: A3		State (2, 5, 2) V = -0.42		Best Action: A3		State (4, 5, 6) V = 0.42		Best Action: A3		State (2, 2, 7) V = -100000		Best Action: No Action	
State (1, 2, 3) V = -1.19		Best Action: A4		State (3, 3, 2) V = -1.68		Best Action: A3		State (5, 4, 3) V = -1.22		Best Action: A4		State (2, 5, 3) V = -4.17		Best Action: A4		State (4, 5, 7) V = 9.09		Best Action: A4		State (2, 2, 8) V = -100000		Best Action: No Action	
State (1, 2, 4) V = -1.19		Best Action: A3		State (3, 3, 3) V = -1.19		Best Action: A3		State (5, 4, 4) V = 66.56		Best Action: A3		State (2, 5, 4) V = -4.13		Best Action: A4		State (4, 5, 8) V = 9.95		Best Action: A4		State (2, 2, 9) V = -100000		Best Action: No Action	
State (1, 3, 1) V = -0.67		Best Action: A3		State (3, 3, 4) V = -1.19		Best Action: A3		State (5, 5, 1) V = 100		Best Action: No Action		State (2, 5, 5) V = 4.13		Best Action: A4		State (4, 5, 9) V = 9.95		Best Action: A4		State (2, 2, 10) V = -100000		Best Action: No Action	
State (1, 3, 2) V = -0.75		Best Action: A3		State (3, 3, 5) V = -2.12		Best Action: A3		State (5, 5, 2) V = 100		Best Action: No Action		State (3, 1, 1) V = -3.86		Best Action: A4		State (5, 2, 1) V = 14.75		Best Action: A1		State (2, 2, 11) V = -100000		Best Action: No Action	
State (1, 3, 3) V = -1.19		Best Action: A4		State (3, 3, 6) V = -1.19		Best Action: A3		State (5, 5, 3) V = 100		Best Action: No Action		State (3, 1, 2) V = -3.86		Best Action: A4		State (5, 2, 2) V = 0.84		Best Action: A3		State (2, 2, 12) V = -100000		Best Action: No Action	
State (1, 3, 4) V = -1.19		Best Action: A3		State (3, 3, 7) V = -2.34		Best Action: A3		State (5, 5, 4) V = 100		Best Action: No Action		State (3, 1, 3) V = -4.17		Best Action: A3		State (5, 2, 3) V = 9.45		Best Action: A4		State (2, 2, 13) V = -100000		Best Action: No Action	
State (1, 4, 1) V = -0.67		Best Action: A3		State (3, 4, 1) V = -0.67		Best Action: A3		Iteration 4:				State (3, 1, 4) V = -4.16		Best Action: A3		State (5, 2, 4) V = 10.27		Best Action: A3		State (2, 2, 14) V = -100000		Best Action: No Action	
State (1, 4, 2) V = -0.75		Best Action: A3		State (3, 4, 2) V = -0.67		Best Action: A3		State (1, 1, 1) V = -3.22		Best Action: A3		State (3, 1, 1) V = -3.22		Best Action: A3		State (5, 3, 1) V = 28.66		Best Action: A3		State (2, 2, 15) V = -100000		Best Action: No Action	
State (1, 4, 3) V = -1.19		Best Action: A4		State (3, 4, 3) V = -1.19		Best Action: A2		State (1, 1, 2) V = -3.22		Best Action: A3		State (3, 1, 2) V = -3.22		Best Action: A3		State (5, 3, 2) V = 26.13		Best Action: A3		State (2, 2, 16) V = -100000		Best Action: No Action	
State (1, 4, 4) V = -1.19		Best Action: A3		State (3, 4, 4) V = -1.19		Best Action: A3		State (1, 1, 3) V = -3.53		Best Action: A3		State (3, 1, 3) V = -3.53		Best Action: A3		State (5, 3, 3) V = 38.66		Best Action: A2		State (2, 2, 17) V = -100000		Best Action: No Action	
State (1, 5, 1) V = -0.67		Best Action: A3		State (3, 5, 1) V = 49.69		Best Action: A3		State (1, 1, 4) V = -3.53		Best Action: A3		State (3, 1, 4) V = -3.53		Best Action: A3		State (5, 3, 4) V = 23.57		Best Action: A2		State (2, 2, 18) V = -100000		Best Action: No Action	
State (1, 5, 2) V = -0.76		Best Action: A3		State (3, 5, 2) V = 49.34		Best Action: A3		State (1, 2, 1) V = -3.25		Best Action: A3		State (3, 2, 1) V = -3.25		Best Action: A3		State (5, 4, 1) V = 79.42		Best Action: A1		State (2, 2, 19) V = -100000		Best Action: No Action	
State (1, 5, 3) V = -1.19		Best Action: A4		State (3, 5, 3) V = 37.93		Best Action: A4		State (1, 2, 2) V = -3.27		Best Action: A3		State (3, 2, 2) V = -3.27		Best Action: A3		State (5, 4, 2) V = 54.69		Best Action: A3		State (2, 2, 20) V = -100000		Best Action: No Action	
State (1, 5, 4) V = -1.19		Best Action: A3		State (3, 5, 4) V = 80.19		Best Action: A2		State (1, 2, 3) V = -3.55		Best Action: A4		State (3, 2, 3) V = -3.55		Best Action: A4		State (5, 4, 3) V = -0.22		Best Action: A4		State (2, 2, 21) V = -100000		Best Action: No Action	
State (2, 1, 1) V = -0.67		Best Action: A3		State (4, 1, 1) V = -8.88		Best Action: A4		State (1, 2, 4) V = -3.55		Best Action: A4		State (3, 2, 4) V = -3.55		Best Action: A4		State (5, 4, 4) V = 67.94		Best Action: A3		State (2, 2, 22) V = -100000		Best Action: No Action	
State (2, 1, 2) V = -0.67		Best Action: A3		State (4, 1, 2) V = -1.68		Best Action: A3		State (1, 3, 1) V = -3.26		Best Action: A3		State (3, 3, 1) V = -3.26		Best Action: A3		State (5, 5, 1) V = 100		Best Action: No Action		State (2, 2, 23) V = -100000		Best Action: No Action	
State (2, 1, 3) V = -0.67		Best Action: A3		State (4, 1, 3) V = -0.67		Best Action: A3		State (1, 3, 2) V = -3.32		Best Action: A3		State (3, 3, 2) V = -3.32		Best Action: A3		State (5, 5, 2) V = 100		Best Action: No Action		State (2, 2, 24) V = -100000		Best Action: No Action	
State (2, 1, 4) V = -1.26		Best Action: A3		State (4, 1, 4) V = -2.55		Best Action: A3		State (1, 3, 3) V = -3.56		Best Action: A4		State (3, 3, 3) V = -3.56		Best Action: A4		State (5, 5, 3) V = 100		Best Action: No Action		State (2, 2, 25) V = -100000		Best Action: No Action	
State (2, 2, 1) V = -1.19		Best Action: A3		State (4, 2, 1) V = -68.07		Best Action: A3		State (1, 3, 4) V = -3.56		Best Action: A4		State (3, 3, 4) V = -3.56		Best Action: A4		State (5, 5, 4) V = 100		Best Action: No Action		State (2, 2, 26) V = -100000		Best Action: No Action	
State (2, 2, 2) V = -100000		Best Action: No Action		State (4, 2, 2) V = -2.96		Best Action: A2		State (1, 4, 1) V = -4.07		Best Action: A3		State (3, 4, 1) V = -4.07		Best Action: A3		State (5, 5, 5) V = 41.62		Best Action: A3		State (2, 2, 27) V = -100000		Best Action: No Action	
State (2, 2, 3) V = -100000		Best Action: No Action		State (4, 2, 3) V = -7.46		Best Action: A3		State (1, 4, 2) V = -4.07		Best Action: A3		State (3, 4, 2) V = -4.07		Best Action: A3		State (5, 5, 6) V = 23.56		Best Action: A1		State (2, 2, 28) V = -100000		Best Action: No Action	
State (2, 2, 4) V = -100000		Best Action: No Action		State (4, 2, 4) V = -6.81		Best Action: A1		State (1, 4, 3) V = -4.18		Best Action: A4		State (3, 4, 3) V = -4.18		Best Action: A4		State (5, 5, 7) V = 37.29		Best Action: A4		State (2, 2, 29) V = -100000		Best Action: No Action	
State (2, 2, 5) V = -100000		Best Action: No Action		State (4, 3, 1) V = -68.09		Best Action: A4		State (1, 4, 4) V = -10.65		Best Action: A3		State (3, 4, 4) V = -10.65		Best Action: A3		State (5, 5, 8) V = -23.81		Best Action: A3		State (2, 2, 30) V = -100000		Best Action: No Action	
State (2, 2, 6) V = -100000		Best Action: No Action		State (4, 3, 2) V = -3.74		Best Action: A2		State (1, 5, 1) V = -3.23		Best Action: A3		State (3, 5, 1) V = 71.0		Best Action: A4		State (5, 5, 9) V = 71.0		Best Action: A4		State (2, 2, 31) V = -100000		Best Action: No Action	
State (2, 2, 7) V = -100000		Best Action: No Action		State (4, 3, 3) V = -7.53		Best Action: A1		State (1, 5, 2) V = -3.37		Best Action: A3		State (3, 5, 2) V = 65.95		Best Action: A3		State (5, 5, 10) V = 65.95		Best Action: A3		State (2, 2, 32) V = -100000		Best Action: No Action	
State (2, 2, 8) V = -100000		Best Action: No Action		State (4, 3, 4) V = -6.86		Best Action: A1		State (1, 5, 3) V = -3.54		Best Action: A3		State (3, 5, 3) V = 61.59		Best Action: A4		State (5, 5, 11) V = 61.59		Best Action: A4		State (2, 2, 33) V = -100000		Best Action: No Action	
State (2, 3, 1) V = -100000		Best Action: No Action		State (4, 4, 1) V = -1000		Best Action: No Action		State (1, 5, 4) V = -3.54		Best Action: A3		State (3, 5, 4) V = 83.2		Best Action: A2		State (5, 5, 12) V = 83.2		Best Action: A2		State (2, 3, 1) V = -100000		Best Action: No Action	
State (2, 3, 2) V = -100000		Best Action: No Action		State (4, 4, 2) V = -1000		Best Action: No Action		State (2, 1, 1) V = -3.25		Best Action: A4		State (3, 5, 5) V = -11.97		Best Action: A4		State (5, 5, 13) V = -11.97		Best Action: A4		State (2, 3, 2) V = -100000		Best Action: No Action	
State (2, 3, 3) V = -100000		Best Action: No Action		State (4, 4, 3) V = -1000		Best Action: No Action		State (2, 1, 2) V = -3.25		Best Action: A3		State (3, 5, 6) V = -4.69		Best Action: A3		State (5, 5, 14) V = -4.69		Best Action: A3		State (2, 3, 3) V = -100000		Best Action: No Action	
State (2, 3, 4) V = -100000		Best Action: No Action		State (4, 4, 4) V = -1000		Best Action: No Action		State (2, 1, 3) V = -3.59		Best Action: A3		State (3, 5, 7) V = -5.26		Best Action: A3		State (5, 5, 15) V = -5.26		Best Action: A3		State (2, 3, 4) V = -100000		Best Action: No Action	
State (2, 3, 5) V = -100000		Best Action: No Action		State (4, 4, 5) V = -1000		Best Action: No Action		State (2, 1, 4) V = -3.6		Best Action: A3		State (3, 5, 8) V = -5.21		Best Action: A1		State (5, 5, 16) V = -5.21		Best Action: A1		State (2, 3, 5) V = -100000		Best Action: No Action	
State (2, 3, 6) V = -100000		Best Action: No Action		State (4, 5, 1) V = -47.14		Best Action: A2		State (2, 2, 1) V = -100000		Best Action: No Action		State (3, 5, 9) V = -70.91		Best Action: A1		State (5, 5, 17) V = -70.91		Best Action: A1		State (2, 3, 6) V = -100000		Best Action: No Action	
State (2, 4, 1) V = -0.67		Best Action: A3		State (4, 5, 2) V = -0.67		Best Action: A3		State (2, 2, 2) V = -100000		Best Action: No Action		State (3, 5, 10) V = -9.12		Best Action: A1		State (5, 5, 18) V = -9.12		Best Action: A1		State (2, 3, 7) V = -100000		Best Action: No Action	
State (2, 4, 2) V = -1.26		Best Action: A3		State (4, 5, 3) V = 34.27		Best Action: A3		State (2, 2, 3) V = -100000		Best Action: No Action		State (3, 5, 11) V = -16.42		Best Action: A3		State (5, 5, 19) V = -16.42		Best Action: A3		State (2, 3, 8) V = -100000		Best Action: No Action	
State (2, 4, 3) V = -1.19		Best Action: A3		State (4, 5, 4) V = -61.19		Best Action: A1		State (2, 2, 4) V = -100000		Best Action: No Action		State (3, 5, 12) V = 76.46		Best Action: A3		State (5, 5, 20) V = 76.46		Best Action: A3		State (2, 3, 9) V = -100000		Best Action: No Action	
State (2, 5, 1) V = -0.67		Best Action: A3		State (5, 1, 1) V = -1.68		Best Action: A4		State (2, 2, 5) V = -100000		Best Action: No Action		State (3, 5, 13) V = -1.68		Best Action: A4		State (5, 5, 21) V = -1.68		Best Action: A4		State (2, 3, 10) V = -100000		Best Action: No Action	
State (2, 5, 2) V = -0.71		Best Action: A3		State (5, 1, 2) V = -1.68		Best Action: A4		State (2, 2, 6) V = -100000		Best Action: No Action		State (3, 5, 14) V = -1.68		Best Action: A4		State (5, 5, 22) V = -1.68		Best Action: A4		State (2, 3, 11) V = -100000		Best Action: No Action	
State (2, 5, 3) V = -1.26		Best Action: A4		State (5, 1, 3) V = -2.27		Best Action: A3		State (2, 2, 7) V = -100000		Best Action: No Action		State (3, 5, 15) V = -2.27		Best Action: A3		State (5, 5, 23) V = -2.27		Best Action: A3		State (2, 3, 12) V = -100000		Best Action: No Action	
State (2, 5, 4) V = -1.19		Best Action: A3		State (5, 1, 4) V = -1.19		Best Action: A3		State (2, 2, 8) V = -100000		Best Action: No Action		State (3, 5, 16) V = -2.12		Best Action: A3		State (5, 5, 24) V = -2.12		Best Action: A3		State (2, 3, 13) V = -100000		Best Action: No Action	
State (3, 1, 1) V = -0.67		Best Action: A3		State (5, 2, 1) V = -1.69		Best Action: A4		State (2, 2, 9) V = -100000		Best Action: No Action		State (3, 5, 17) V = -1.69		Best Action: A4		State (5, 5, 25) V = -1.69		Best Action: A4		State (2, 3, 14) V = -100000		Best Action: No Action	
State (3, 1, 2) V = -0.67		Best Action: A3		State (5, 2, 2) V = -1.76		Best Action: A4		State (2, 2, 10) V = -100000		Best Action: No Action		State (3, 5, 18) V = -1.76		Best Action: A4		State (5, 5, 26) V = -1.76		Best Action: A4		State (2, 3, 15) V = -100000		Best Action: No Action	
State (3, 1, 3) V = -1.34		Best Action: A3		State (5, 2, 3) V = -2.53		Best Action: A4		State (2, 2, 11) V = -100000		Best Action: No Action		State (3, 5, 19) V = -2.53		Best Action: A4		State (5, 5, 27) V = -2.53		Best Action: A4		State (2, 3, 16) V = -100000		Best Action: No Action	
State (3, 1, 4) V = -1.19		Best Action: A3		State (5, 2, 4) V = -2.13		Best Action: A3		State (2, 2, 12) V = -100000		Best Action: No Action		State (3, 5, 20) V = -2.13		Best Action: A3		State (5, 5, 28) V = -2.13		Best Action: A3		State (2, 3, 17) V = -100000		Best Action: No Action	
State (3, 2, 1) V = -100000		Best Action: No Action		State (5, 3, 1) V = -1.7		Best Action: A4		State (2, 2, 13) V = -100000		Best Action: No Action		State (3, 5, 21) V = -1.7		Best Action: A4		State (5, 5, 29) V = -1.7		Best Action: A4		State (2, 3, 18) V = -100000		Best Action: No Action	
State (3, 2, 2) V = -100000		Best Action: No Action		State (5, 3, 2) V = -1.82		Best Action: A4		State (2, 2, 14) V = -100000		Best Action: No Action		State (3, 5, 22) V = -1.82		Best Action: A4		State (5, 5, 30) V = -1.82		Best Action: A4		State (2, 3, 19) V = -100000		Best Action: No Action	
State (3, 2, 3) V = -100000		Best Action: No Action		State (5, 3, 3) V = -2.58		Best Action: A4		State (

State (4, 1, 4) V = 13.79 Best Action: A1
State (4, 2, 1) V = -58.36 Best Action: A4
State (4, 2, 2) V = 4.21 Best Action: A3
State (4, 2, 3) V = -1.83 Best Action: A3
State (4, 2, 4) V = 10.12 Best Action: A1
State (4, 3, 1) V = -35.57 Best Action: A3
State (4, 3, 2) V = 27.28 Best Action: A4
State (4, 3, 3) V = 36.58 Best Action: A1
State (4, 3, 4) V = 19.81 Best Action: A1
State (4, 4, 1) V = -1000 Best Action: No
Action
State (4, 4, 2) V = -1000 Best Action: No
Action
State (4, 4, 3) V = -1000 Best Action: No
Action
State (4, 4, 4) V = -1000 Best Action: No
Action
State (4, 5, 1) V = 68.01 Best Action: A4
State (4, 5, 2) V = -0.16 Best Action: A3
State (4, 5, 3) V = 55.36 Best Action: A4
State (4, 5, 4) V = 79.44 Best Action: A1
State (5, 1, 1) V = 26.11 Best Action: A2
State (5, 1, 2) V = 16.0 Best Action: A3
State (5, 1, 3) V = 19.66 Best Action: A4
State (5, 1, 4) V = 21.44 Best Action: A3
State (5, 2, 1) V = 26.55 Best Action: A1
State (5, 2, 2) V = 16.57 Best Action: A3
State (5, 2, 3) V = 20.45 Best Action: A4
State (5, 2, 4) V = 21.84 Best Action: A3
State (5, 3, 1) V = 34.2 Best Action: A3
State (5, 3, 2) V = 32.13 Best Action: A4
State (5, 3, 3) V = 40.7 Best Action: A2
State (5, 3, 4) V = 29.32 Best Action: A3
State (5, 4, 1) V = 79.43 Best Action: A1
State (5, 4, 2) V = 54.84 Best Action: A3
State (5, 4, 3) V = -0.2 Best Action: A4
State (5, 4, 4) V = 67.97 Best Action: A3
State (5, 5, 1) V = 100 Best Action: No Action
State (5, 5, 2) V = 100 Best Action: No Action
State (5, 5, 3) V = 100 Best Action: No Action
State (5, 5, 4) V = 100 Best Action: No Action
Iteration 10:
State (1, 1, 1) V = 2.72 Best Action: A4
State (1, 1, 2) V = 3.3 Best Action: A3
State (1, 1, 3) V = 1.68 Best Action: A3
State (1, 1, 4) V = 8.93 Best Action: A2
State (1, 2, 1) V = -4.16 Best Action: A3
State (1, 2, 2) V = 0.38 Best Action: A1
State (1, 2, 3) V = -1.07 Best Action: A3
State (1, 2, 4) V = -1.07 Best Action: A4
State (1, 3, 1) V = -4.29 Best Action: A3
State (1, 3, 2) V = 0.18 Best Action: A2
State (1, 3, 3) V = -1.24 Best Action: A3
State (1, 3, 4) V = -1.24 Best Action: A4
State (1, 4, 1) V = -5.69 Best Action: A3
State (1, 4, 2) V = -2.15 Best Action: A1
State (1, 4, 3) V = -3.18 Best Action: A3
State (1, 4, 4) V = -7.99 Best Action: A4
State (1, 5, 1) V = -5.26 Best Action: A3
State (1, 5, 2) V = -2.37 Best Action: A2
State (1, 5, 3) V = -3.3 Best Action: A3
State (1, 5, 4) V = -3.32 Best Action: A4
State (2, 1, 1) V = 4.12 Best Action: A4
State (2, 1, 2) V = 4.12 Best Action: A3
State (2, 1, 3) V = 2.6 Best Action: A3
State (2, 1, 4) V = 9.86 Best Action: A1
State (2, 2, 1) V = -100000 Best Action: No
Action
State (2, 2, 2) V = -100000 Best Action: No
Action
State (2, 2, 3) V = -100000 Best Action: No
Action
State (2, 2, 4) V = -100000 Best Action: No
Action
State (2, 3, 1) V = -100000 Best Action: No
Action
State (2, 3, 2) V = -100000 Best Action: No
Action
State (2, 3, 3) V = -100000 Best Action: No
Action
State (2, 3, 4) V = -100000 Best Action: No
Action
State (2, 4, 1) V = -10.51 Best Action: A1
State (2, 4, 2) V = -11.22 Best Action: A4
State (2, 4, 3) V = -5.41 Best Action: A1
State (2, 4, 4) V = -70.14 Best Action: A3
State (2, 5, 1) V = -5.61 Best Action: A3
State (2, 5, 2) V = -5.99 Best Action: A3
State (2, 5, 3) V = -4.8 Best Action: A1
State (2, 5, 4) V = -5.73 Best Action: A3
State (3, 1, 1) V = 10.41 Best Action: A4
State (3, 1, 2) V = 10.41 Best Action: A3
State (3, 1, 3) V = 7.71 Best Action: A3
State (3, 1, 4) V = 15.75 Best Action: A2
State (3, 2, 1) V = -100000 Best Action: No
Action
State (3, 2, 2) V = -100000 Best Action: No
Action
State (3, 2, 3) V = -100000 Best Action: No
Action
State (3, 2, 4) V = -100000 Best Action: No
Action
State (3, 3, 1) V = 58.99 Best Action: A2
State (3, 3, 2) V = 43.68 Best Action: A4
State (3, 3, 3) V = 50.48 Best Action: A4
State (3, 3, 4) V = 47.37 Best Action: A3
State (3, 4, 1) V = 54.66 Best Action: A1
State (3, 4, 2) V = 33.19 Best Action: A1
State (3, 4, 3) V = 40.16 Best Action: A4
State (3, 4, 4) V = -20.56 Best Action: A3
State (3, 5, 1) V = 71.71 Best Action: A4
State (3, 5, 2) V = 67.72 Best Action: A3
State (3, 5, 3) V = 62.53 Best Action: A4
State (3, 5, 4) V = 83.37 Best Action: A2
State (4, 1, 1) V = 3.91 Best Action: A4
State (4, 1, 2) V = 10.4 Best Action: A3
State (4, 1, 3) V = 7.67 Best Action: A3
State (4, 1, 4) V = 15.72 Best Action: A1
State (4, 2, 1) V = -55.62 Best Action: A4
State (4, 2, 2) V = 7.39 Best Action: A3
State (4, 2, 3) V = 1.1 Best Action: A3
State (4, 2, 4) V = 12.04 Best Action: A1
State (4, 3, 1) V = -35.27 Best Action: A3
State (4, 3, 2) V = 27.93 Best Action: A4
State (4, 3, 3) V = 36.7 Best Action: A1
State (4, 3, 4) V = 20.46 Best Action: A1
State (4, 4, 1) V = -1000 Best Action: No
Action
State (4, 4, 2) V = -1000 Best Action: No
Action
State (4, 4, 3) V = -1000 Best Action: No
Action
State (4, 4, 4) V = -1000 Best Action: No
Action
State (4, 5, 1) V = 68.01 Best Action: A4
State (4, 5, 2) V = -0.16 Best Action: A3
State (4, 5, 3) V = 55.36 Best Action: A4
State (4, 5, 4) V = 79.44 Best Action: A1
State (5, 1, 1) V = 26.92 Best Action: A2
State (5, 1, 2) V = 17.87 Best Action: A3
State (5, 1, 3) V = 20.71 Best Action: A4
State (5, 1, 4) V = 22.36 Best Action: A3
State (5, 2, 1) V = 27.35 Best Action: A1
State (5, 2, 2) V = 18.35 Best Action: A3
State (5, 2, 3) V = 21.42 Best Action: A4
State (5, 2, 4) V = 22.75 Best Action: A3
State (5, 3, 1) V = 34.5 Best Action: A3
State (5, 3, 2) V = 32.57 Best Action: A4
State (5, 3, 3) V = 40.81 Best Action: A2
State (5, 3, 4) V = 29.65 Best Action: A3
State (5, 4, 1) V = 79.43 Best Action: A1
State (5, 4, 2) V = 54.84 Best Action: A3
State (5, 4, 3) V = -0.2 Best Action: A4
State (5, 4, 4) V = 67.97 Best Action: A3
State (5, 5, 1) V = 100 Best Action: No Action
State (5, 5, 2) V = 100 Best Action: No Action
State (5, 5, 3) V = 100 Best Action: No Action
State (5, 5, 4) V = 100 Best Action: No Action