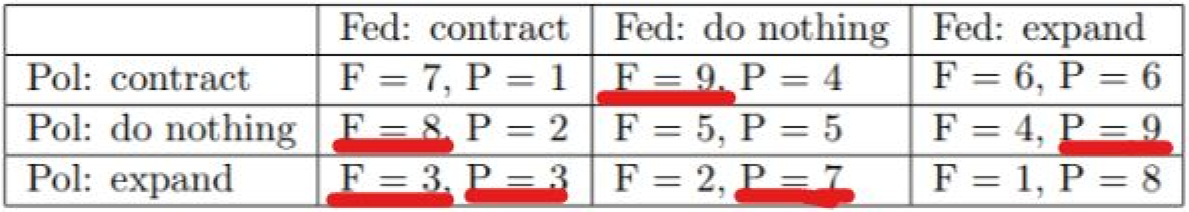
COMP 557 Homework 3 - Linus Shih (yls2), Ege Ersu (ee31)

Question 1

1.



	Fed: contract	Fed: do nothing	Fed: expand
Pol: contract	F = 7, P = 1	F = 9, P = 4	F = 6, P = 6
Pol: do nothing	F = 8, P = 2	F = 5, P = 5	F = 4, P = 9
Pol: expand	F = 3, P = 3	F = 2, P = 7	F = 1, P = 8

Using the "underlining method", we found the pure strategy Nash Equilibrium to be **(F=3,P=3)**. This is **not** the Pareto optimal solution. The pairs (5,5) and (6,6) give more optimal solutions for both the Politicians and Federal Reserve since both parties get better payoff from these squares (5 > 3 and 6 > 3). However, it should be noted that neither (5,5) nor (6,6) are Nash Equilibria.

2.

Question 1.2

	r	p	s	f	w
R	0,0	-1,1	1,-1	-1,1	1,-1
P	1,-1	0,0	-1,1	-1,1	1,-1
S	-1,1	1,-1	0,0	-1,1	1,-1
F	1,-1	1,-1	1,-1	0,0	-1,1
W	-1,1	-1,1	-1,1	1,-1	0,0

Actions = R, P, S, F, W
r, p, s, f, w

$E(R) = (0)p_r + (-1)p_p + (1)p_s + (-1)p_f + (1)p_w = p_s + p_w - p_p - p_f$

$E(P) = (1)p_r + 0(p_p) + (-1)p_s + (-1)p_f + (1)p_w = p_r + p_w - p_s - p_f$

$E(S) = (-1)p_r + (1)p_p + (0)p_s + (-1)p_f + (1)p_w = p_p + p_w - p_r - p_f$

$E(F) = (1)p_r + (1)p_p + (1)p_s + (0)p_f + (-1)p_w = p_r + p_p + p_s - p_w$

$E(W) = (-1)p_r + (-1)p_p + (-1)p_s + (1)p_f + (0)p_s = p_f - p_r - p_p - p_s$

Solving the following system of equations:

$p_s + p_w - p_p - p_f = p_r + p_w - p_s - p_f = p_p + p_w - p_r - p_f = p_r + p_p + p_s - p_w$
$= p_f - p_r - p_p - p_s$

and $1 = p_r + p_p + p_s + p_f + p_w$

we get the following values for p_r, p_p, p_s, p_f, and p_w:

$p_r = \frac{1}{9}$, $p_p = \frac{1}{9}$, $p_s = \frac{1}{9}$, $p_f = \frac{1}{3}$, and $p_w = \frac{1}{3}$

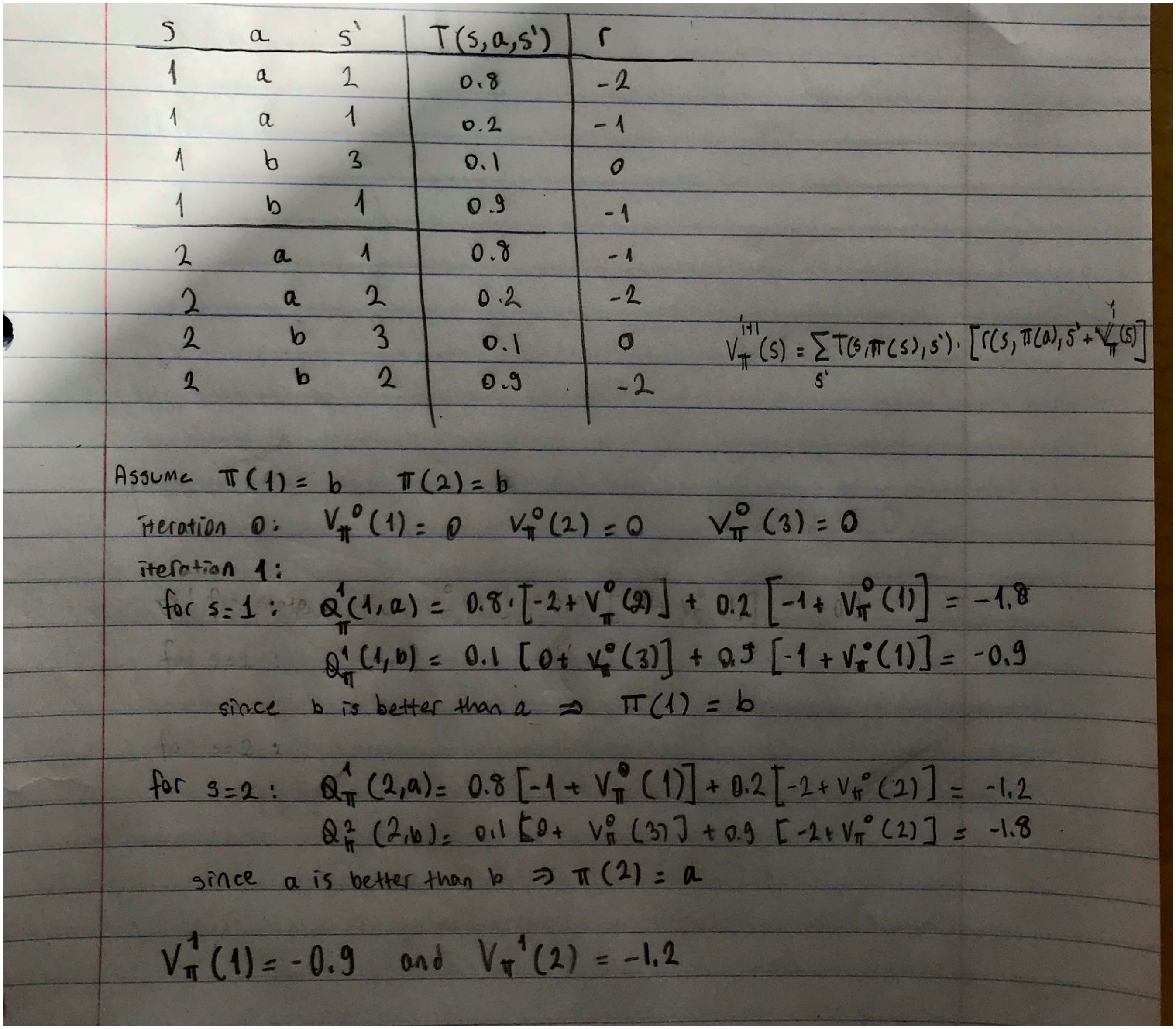
$\Rightarrow$ The mixed strategy equilibrium = $\boxed{\left(\frac{1}{9}, \frac{1}{9}, \frac{1}{9}, \frac{1}{3}, \frac{1}{3}\right)}$

Question 2

a) At state 1 optimal policy would be to take action b. The expected reward is -0.9, where as expected reward for taking action a is: 0.8 * -2 + 0.2 * -1 = -1.4

At state 2 taking action a results in an expected reward of 0.8 * -1 + 0.2 * -2 = -1.2 Taking action b has an expected reward of 0.1 * 0 + 0.9 * -2 = -1.8. So taking action a would be the optimal policy.

b) Image1 for iteration 1, Image2 for iteration 2:



s	a	s'	T(s,a,s')	r
1	a	2	0.8	-2
1	a	1	0.2	-1
1	b	3	0.1	0
1	b	1	0.9	-1
2	a	1	0.8	-1
2	a	2	0.2	-2
2	b	3	0.1	0
2	b	2	0.9	-2

$$V_\pi^{i+1}(s) = \sum_{s'} T(s,\pi(s),s') \cdot \left[r(s,\pi(a),s' + V_\pi^{i}(s) \right]$$

Assume $\pi(1) = b$ $\pi(2) = b$

iteration 0: $V_\pi^0(1) = 0$ $V_\pi^0(2) = 0$ $V_\pi^0(3) = 0$

iteration 1:

for $s = 1$: $Q_\pi^1(1,a) = 0.8 \cdot [-2 + V_\pi^0(2)] + 0.2\,[-1 + V_\pi^0(1)] = -1.8$

$Q_\pi^1(1,b) = 0.1\,[0 + V_\pi^0(3)] + 0.9\,[-1 + V_\pi^0(1)] = -0.9$

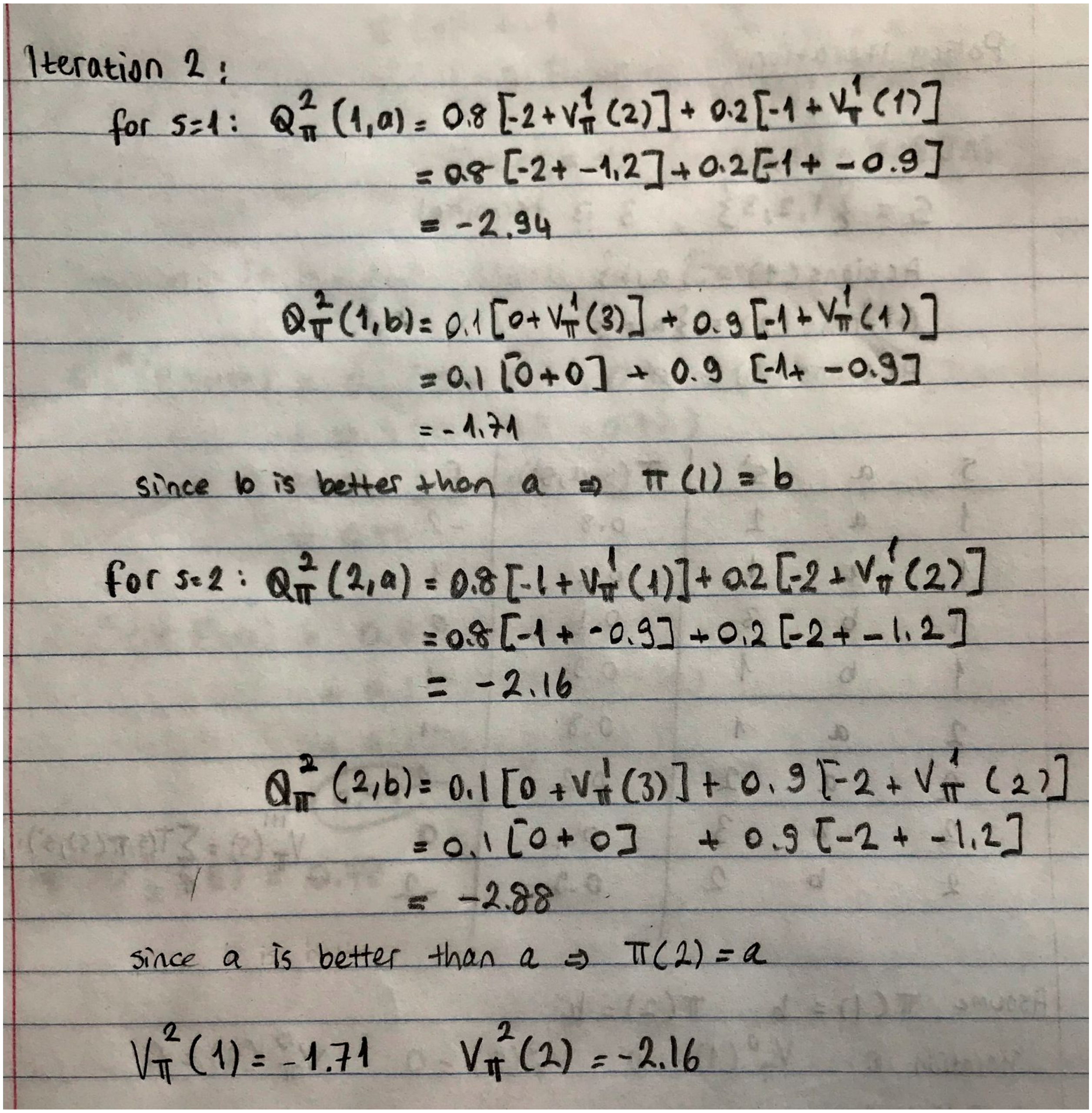
since b is better than a ⇒ $\pi(1) = b$

for $s = 2$: $Q_\pi^1(2,a) = 0.8\,[-1 + V_\pi^0(1)] + 0.2\,[-2 + V_\pi^0(2)] = -1.2$

$Q_\pi^2(2,b) = 0.1\,[0 + V_\pi^0(3)] + 0.9\,[-2 + V_\pi^0(2)] = -1.8$

since a is better than b ⇒ $\pi(2) = a$

$V_\pi^1(1) = -0.9$ and $V_\pi^1(2) = -1.2$



Iteration 2:

for $s=1$: $Q^2_\pi(1,a) = 0.8[-2+V^1_\pi(2)] + 0.2[-1+V^1_\pi(1)]$

$= 0.8[-2+-1.2]+0.2[-1+-0.9]$

$= -2.94$

$Q^2_\pi(1,b) = 0.1[0+V^1_\pi(3)] + 0.9[-1+V^1_\pi(1)]$

$= 0.1[0+0] + 0.9[-1+-0.9]$

$= -1.71$

since b is better than a ⇒ $\pi(1) = b$

for $s=2$: $Q^2_\pi(2,a) = 0.8[-1+V^1_\pi(1)] + 0.2[-2+V^1_\pi(2)]$

$= 0.8[-1+-0.9] + 0.2[-2+-1.2]$

$= -2.16$

$Q^2_\pi(2,b) = 0.1[0+V^1_\pi(3)] + 0.9[-2+V^1_\pi(2)]$

$= 0.1[0+0] + 0.9[-2+-1.2]$

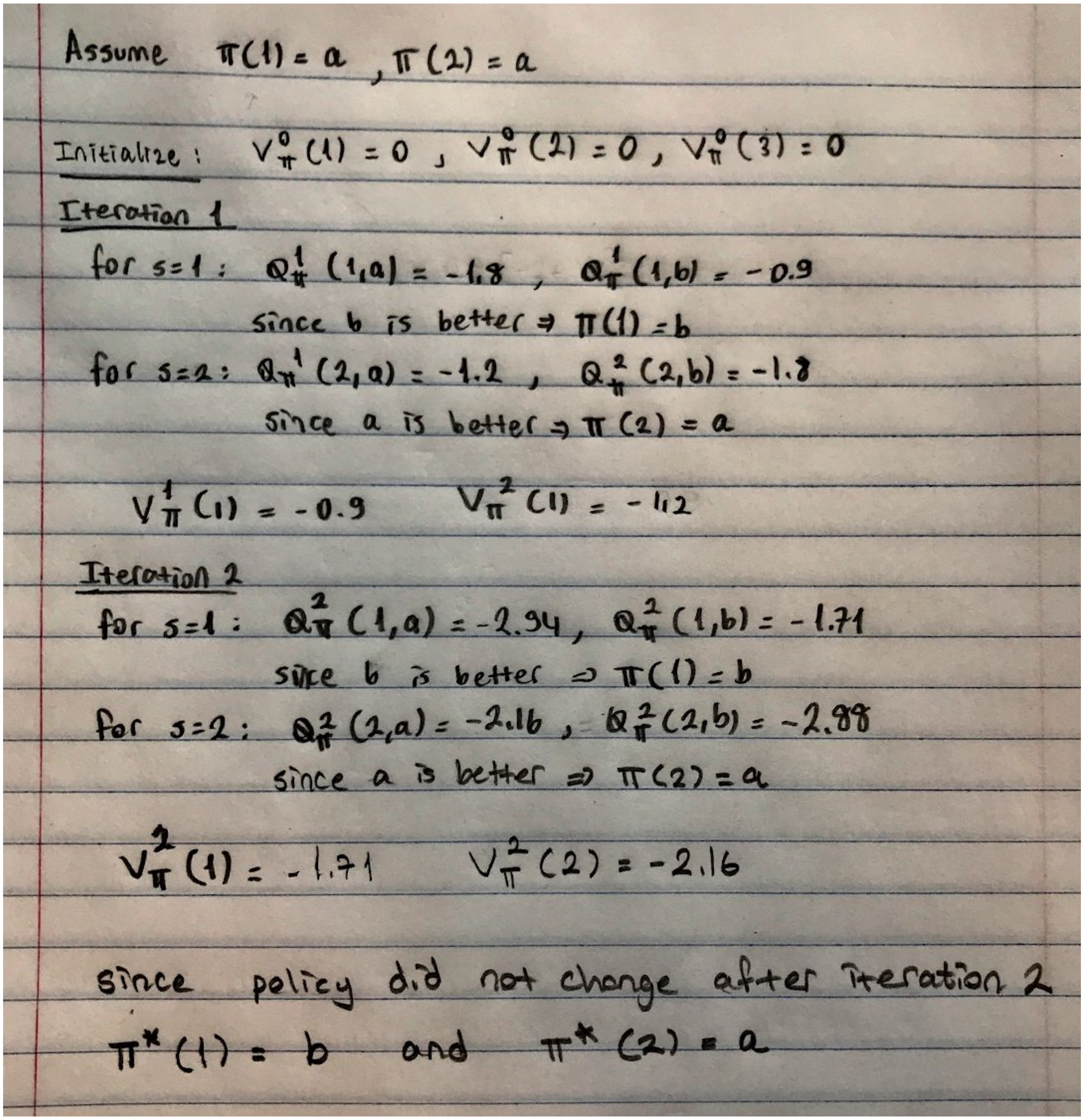
$= -2.88$

since a is better than a ⇒ $\pi(2) = a$

$V^2_\pi(1) = -1.71$ $V^2_\pi(2) = -2.16$

After iteration 2, the policy did not change, so we can stop the algorithm.

The optimal policy is: **π(1) = b , π(2) = a** c)



Assume $\pi(1) = a$, $\pi(2) = a$

Initialize : $V_{\pi}^{0}(1) = 0$, $V_{\pi}^{0}(2) = 0$, $V_{\pi}^{0}(3) = 0$

Iteration 1

for $s=1$: $Q_{\pi}^{1}(1,a) = -1.8$, $Q_{\pi}^{1}(1,b) = -0.9$

since b is better ⇒ $\pi(1) = b$

for $s=2$: $Q_{\pi}^{1}(2,a) = -1.2$, $Q_{\pi}^{2}(2,b) = -1.8$

since a is better ⇒ $\pi(2) = a$

$V_{\pi}^{1}(1) = -0.9$ $V_{\pi}^{2}(1) = -1.2$

Iteration 2

for $s=1$: $Q_{\pi}^{2}(1,a) = -2.94$, $Q_{\pi}^{2}(1,b) = -1.71$

since b is better ⇒ $\pi(1) = b$

for $s=2$: $Q_{\pi}^{2}(2,a) = -2.16$, $Q_{\pi}^{2}(2,b) = -2.88$

since a is better ⇒ $\pi(2) = a$

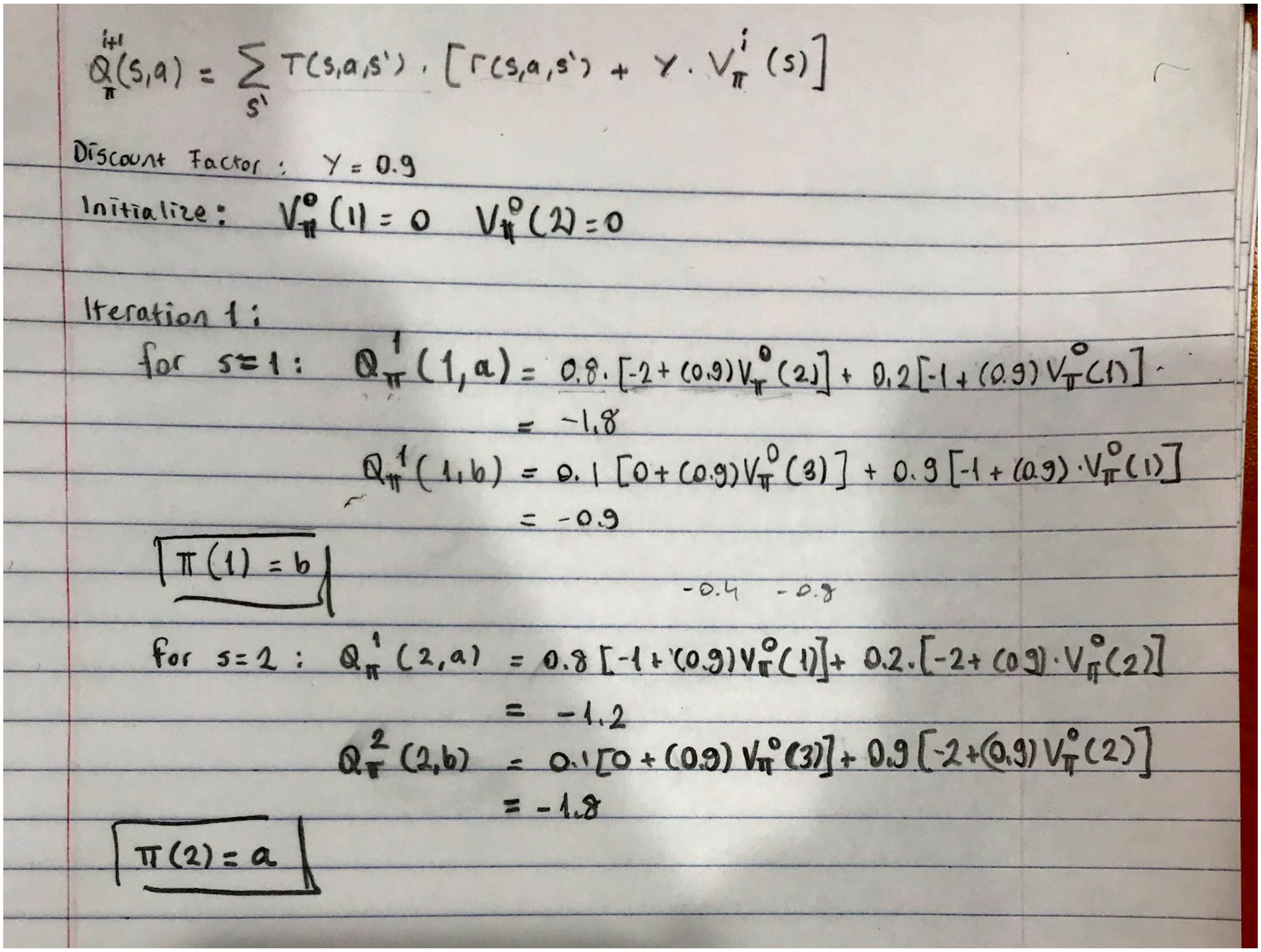
$V_{\pi}^{2}(1) = -1.71$ $V_{\pi}^{2}(2) = -2.16$

since policy did not change after iteration 2

$\pi^{*}(1) = b$ and $\pi^{*}(2) = a$

It doesn’t matter what policy we start with, since policy iteration picks the action that maximizes Q(s,a) on each iteration, the initial policy does not matter. Algorithm will converge to the correct actions eventually.

d)



$$Q_{\pi}^{i+1}(s,a) = \sum_{s'} T(s,a,s') \cdot [r(s,a,s') + \gamma \cdot V_{\pi}^{i}(s)]$$

Discount Factor : $\gamma = 0.9$

Initialize: $V_{\pi}^{0}(1) = 0$ $V_{\pi}^{0}(2) = 0$

Iteration 1:

for $s=1$: $Q_{\pi}^{1}(1,a) = 0.8 \cdot [-2 + (0.9) V_{\pi}^{0}(2)] + 0.2 [-1 + (0.9) V_{\pi}^{0}(1)]$

$= -1.8$

$Q_{\pi}^{1}(1,b) = 0.1 [0 + (0.9) V_{\pi}^{0}(3)] + 0.9 [-1 + (0.9) \cdot V_{\pi}^{0}(1)]$

$= -0.9$

$\boxed{\pi(1) = b}$

-0.4 -0.8

for $s=2$: $Q_{\pi}^{1}(2,a) = 0.8 [-1 + (0.9) V_{\pi}^{0}(1)] + 0.2 \cdot [-2 + (0.9) \cdot V_{\pi}^{0}(2)]$

$= -1.2$

$Q_{\pi}^{2}(2,b) = 0.1 [0 + (0.9) V_{\pi}^{0}(3)] + 0.9 [-2 + (0.9) V_{\pi}^{0}(2)]$

$= -1.8$

$\boxed{\pi(2) = a}$

Since we are calculating the first iteration only, the discount factor will be multiplied by the Vpi(s'), which is initialized to 0 for all states. So the result of the policy iteration does not depend on the discount factor if we are only doing one iteration. We have shown the result for discount factor = 0.9. If the discount factor is 0.1, the resulting policy will also be **π(1) = b , π(2) = a,** since the resulting factor will also be multiplied by 0 in all cases.

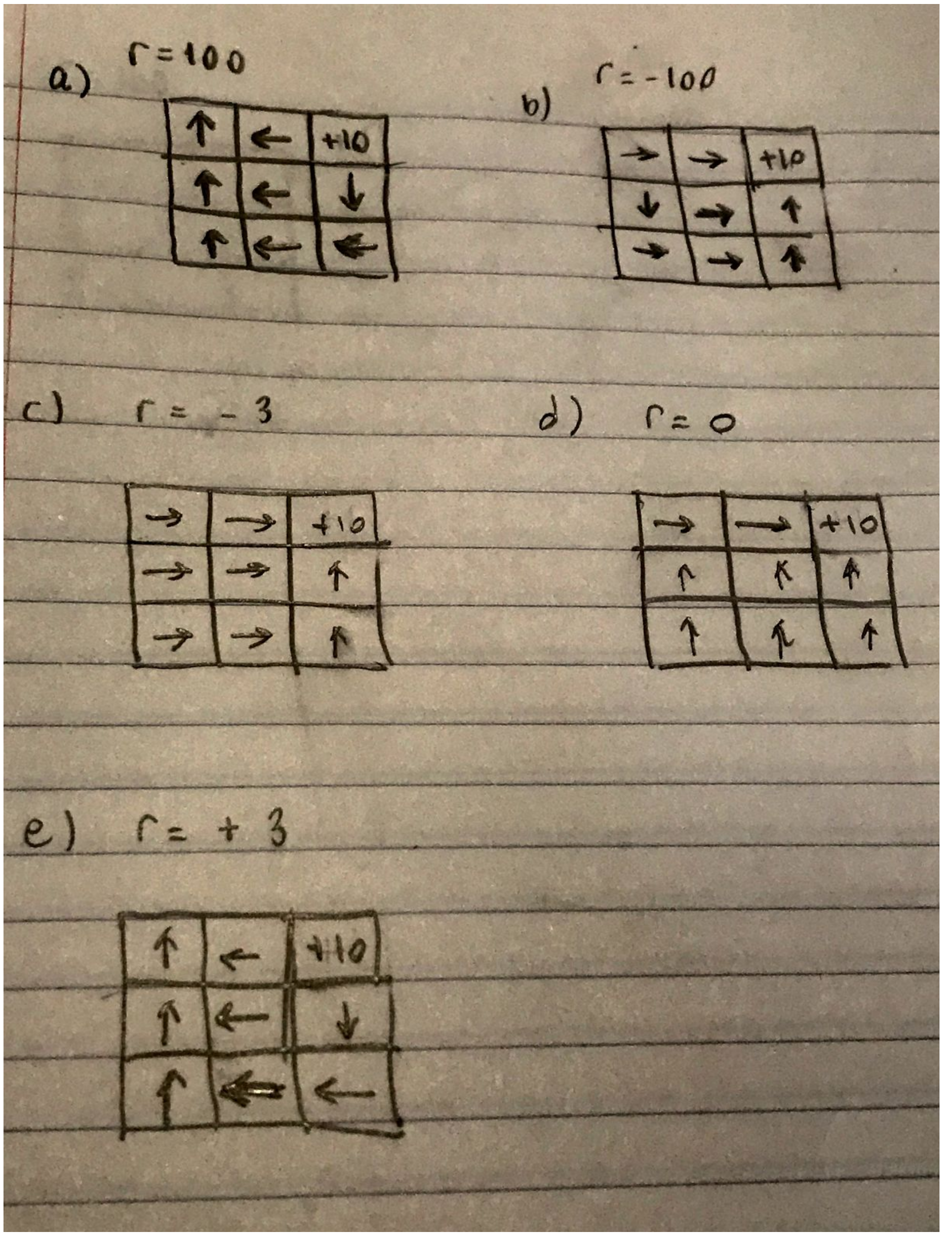
Policy iteration would still work with that initial policy. But the discount factor has to be >= 0, and <1, for it to converge to an optimal policy.

Question 3

3.1.6

It is **not** always the case that $V_1(s_0) > V_2(s_0)$. See submission.py for a counterexample.

Question 4



a) r=100

↑	←	+10
↑	←	↓
↑	←	←

b) r=-100

→	→	+10
↓	→	↑
→	→	↑

c) r= -3

→	→	+10
→	→	↑
→	→	↑

d) r=0

→	→	+10
↑	↑	↑
↑	↑	↑

e) r= + 3

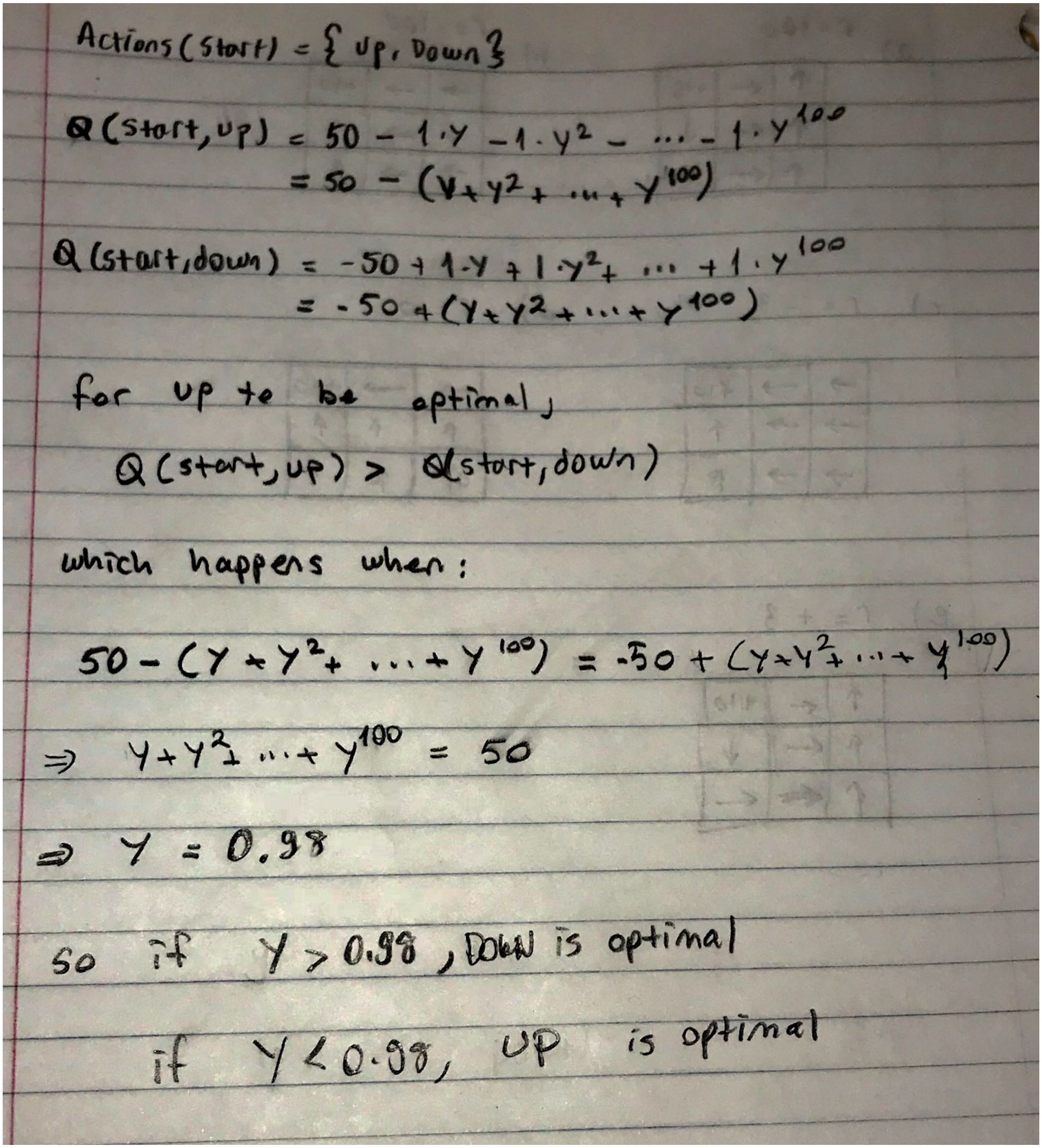
↑	←	+10
↑	←	↓
↑	←	←

a) If r=100, the agent should both move away from the terminal state +10, and move towards +100. If it reaches +10, it exists the game, that's why it's trying to move away at south of +10. Also it first prefers to first go left, then move up, so that it doesn't get near +10.
b) If r = -100, the agent is going to try to get to +10 as quickly as possible and end the game since it's losing points every turn. It will also try to go down first, and then go right, to minimize the probability of falling into -100 by staying at the upper rows.

c) If r = -3, the agent will try to get to +10 and end the game, since there is no other way to earn any rewards. That’s why it will go right and go up if it’s at 10’s column.

d) If r=0, the agent will try to get to +10 and end the game, since there are no other ways to earn any rewards. But it will rather go through r = 0, instead of the other grids where r=-1. That’s why it will first move north, then try to go right. So that it passes through 0, instead of -1.

e) If r=+3, the agent will try to go to r=+3 and stay around there as long as possible, gaining +3 after each iteration. It looks like the simple game we looked at in class, where we want to keep getting +3 instead of exiting the game. The agent will make use of the wall when it is on r=+3, and go north, so that 90% of the time it stays on top r=3. Even if it goes right, it has %80 chance to reach back to r = +3. It would basically want to stay there forever, avoiding +10, the terminal state, at all times.



Actions(Start) = { Up, Down }

$Q(\text{Start}, \text{Up}) = 50 - 1\cdot\gamma - 1\cdot\gamma^2 - \dots - 1\cdot\gamma^{100}$

$= 50 - (\gamma + \gamma^2 + \dots + \gamma^{100})$

$Q(\text{Start}, \text{down}) = -50 + 1\cdot\gamma + 1\cdot\gamma^2 + \dots + 1\cdot\gamma^{100}$

$= -50 + (\gamma + \gamma^2 + \dots + \gamma^{100})$

for up to be optimal,

$Q(\text{start}, \text{up}) > Q(\text{start}, \text{down})$

which happens when:

$50 - (\gamma + \gamma^2 + \dots + \gamma^{100}) = -50 + (\gamma + \gamma^2 + \dots + \gamma^{100})$

$\Rightarrow \gamma + \gamma^2 + \dots + \gamma^{100} = 50$

$\Rightarrow \gamma = 0.98$

so if $\gamma > 0.98$, DOWN is optimal

if $\gamma < 0.98$, UP is optimal

Intuition: if discount factor is high, the agent cares more about future rewards. So it will take -50 first, then start gathering +1s.

If discount factor is low, the agent cares less about future rewards, but more about immediate rewards. So it will take the +50 first, and then gather -1s. It will end up with a negative utility in the long run, so don’t dump pollutants into a lake, it will kill you eventually.