Name: Blayten Jones
ID: 010979697

1. Constraints Satisfaction Problem

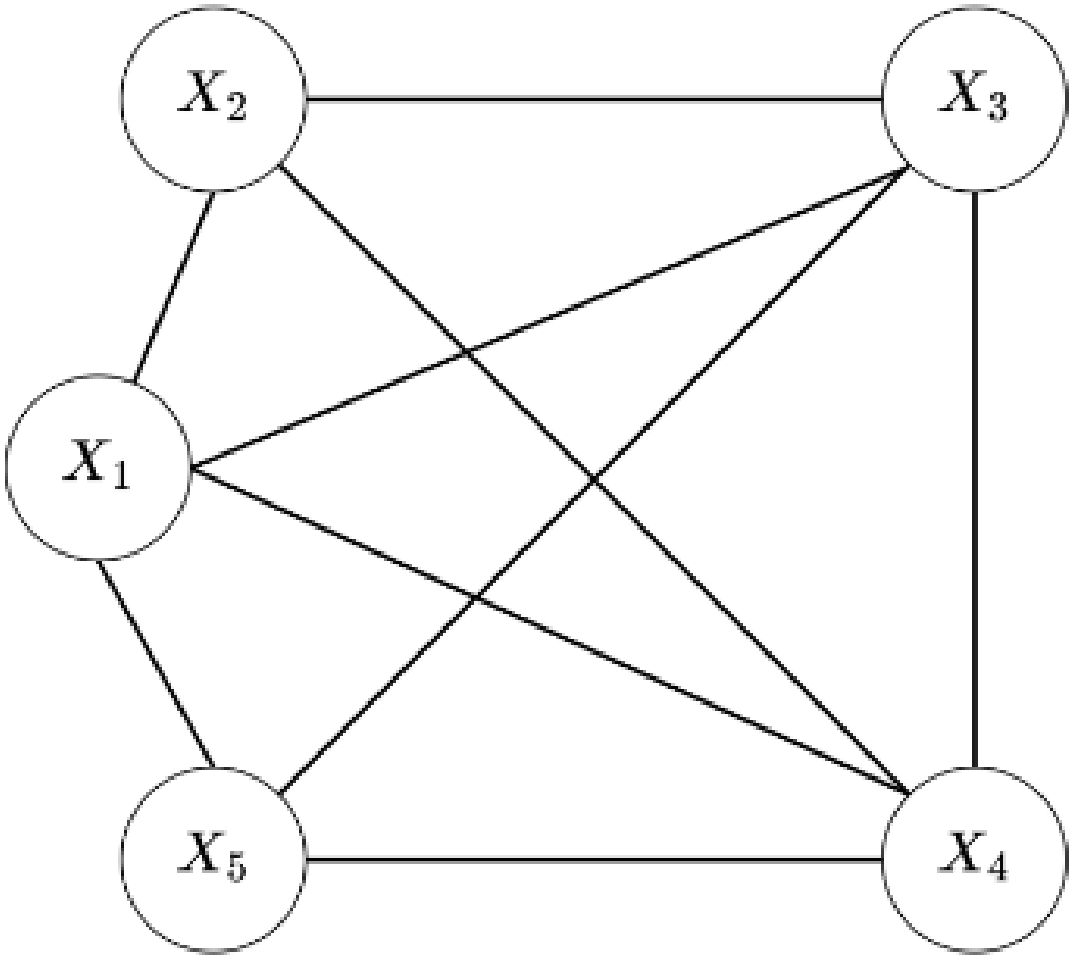


Figure 1: Constraint graph

D(X1) = [1, 2, 3, 4, 5]
D(X2) = [1, 2, 3, 5]
D(X3) = [1, 2, 4, 5]
D(X4) = [1, 3, 4, 5]
D(X5) = [1, 3, 4, 5]

a. What are the neighbors of X4 on the constraint graph?
X1, X2, X3, and X5

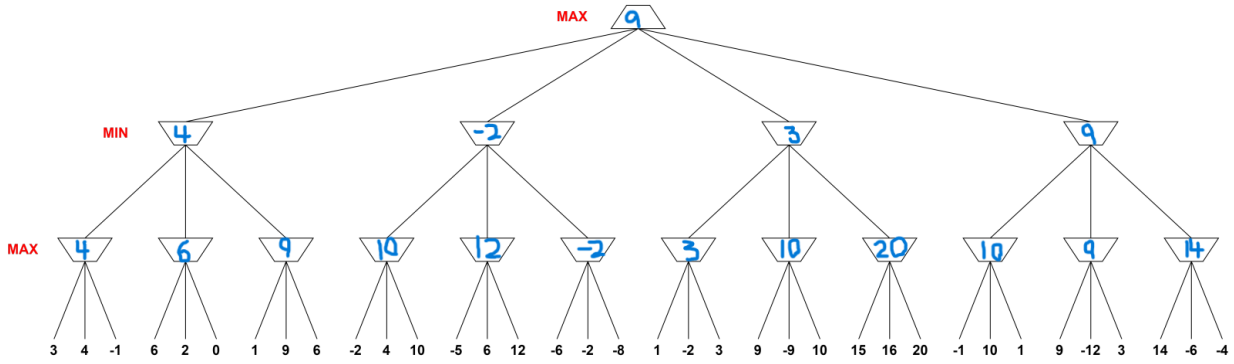
b. Assume we are doing backtracking with *assign*(X1 = 1, X2 = 2). If the next chosen variable is X4, what are the possible values X4 that could be assigned?
[3,4,5]

c. Assume we are doing backtracking with forward checking and *assign*(X1 = 1, X2 = 2), what are the reduced domains of all the variables?
X1 = [1], X2 = [2];
X3 = [4,5], X4 = [3,4,5], X5 = [2,3,4,5];

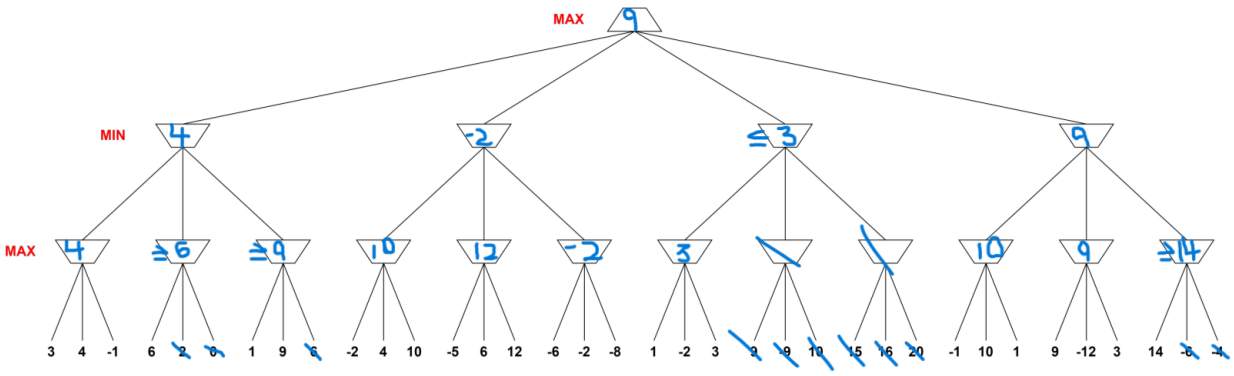
d. Assume we are given the reduced domain D(X3) = [2], D(X5) = [1], the assignment is provided as *assign*(X3 = 2, X5 = 1) as they are the only values these variables could take. If we apply the arc consistency on all arcs, what are the reduced domains of the other variables?
X3 = [2], X5 = [1];
X1 = [3,4,5], X2 = [1,3,5], X4 = [3,4,5];

e. Assume we are doing backtracking and the latest assignment is *assign*(X1 = 1, X2 = 2, X4 = 4, X5 = 5), the next chosen variable is of course X3, what should we do next (go back/backtrack or stop)? Why?
Before: X3 = [1,2,4,5]; After: X3 = [] (as it has every other node as a neighbor), so we must backtrack as the set is left empty and does not satisfy arc consistency.

2. MinMax and Alpha-Beta Pruning
   a. Fill in each blank trapezoid with the proper minimax search value. Process the tree from left to right.



   b. Fill in each blank trapezoid and cross out the branches pruned by minimax + alpha-beta pruning. Process the tree from left to right.



3. Treasure Hunting

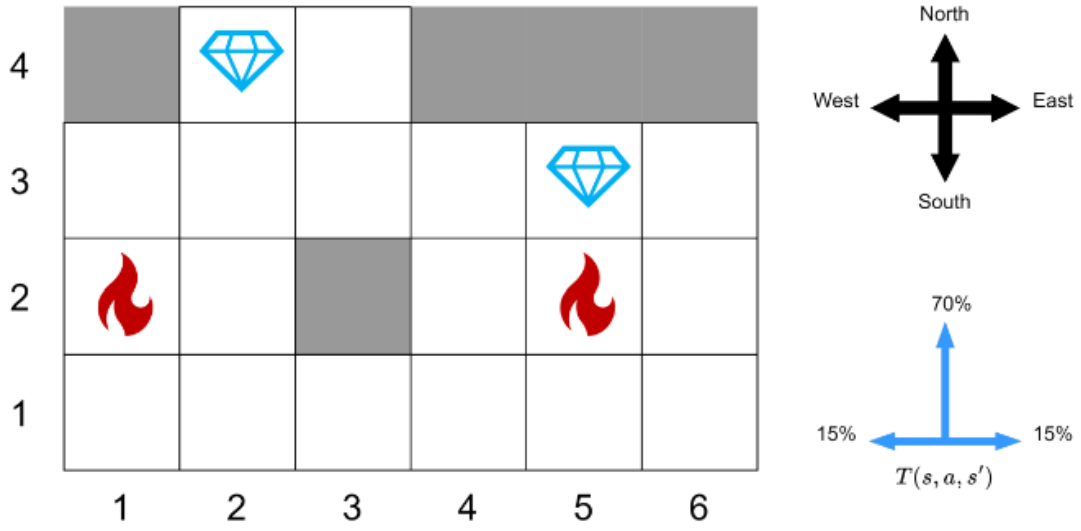


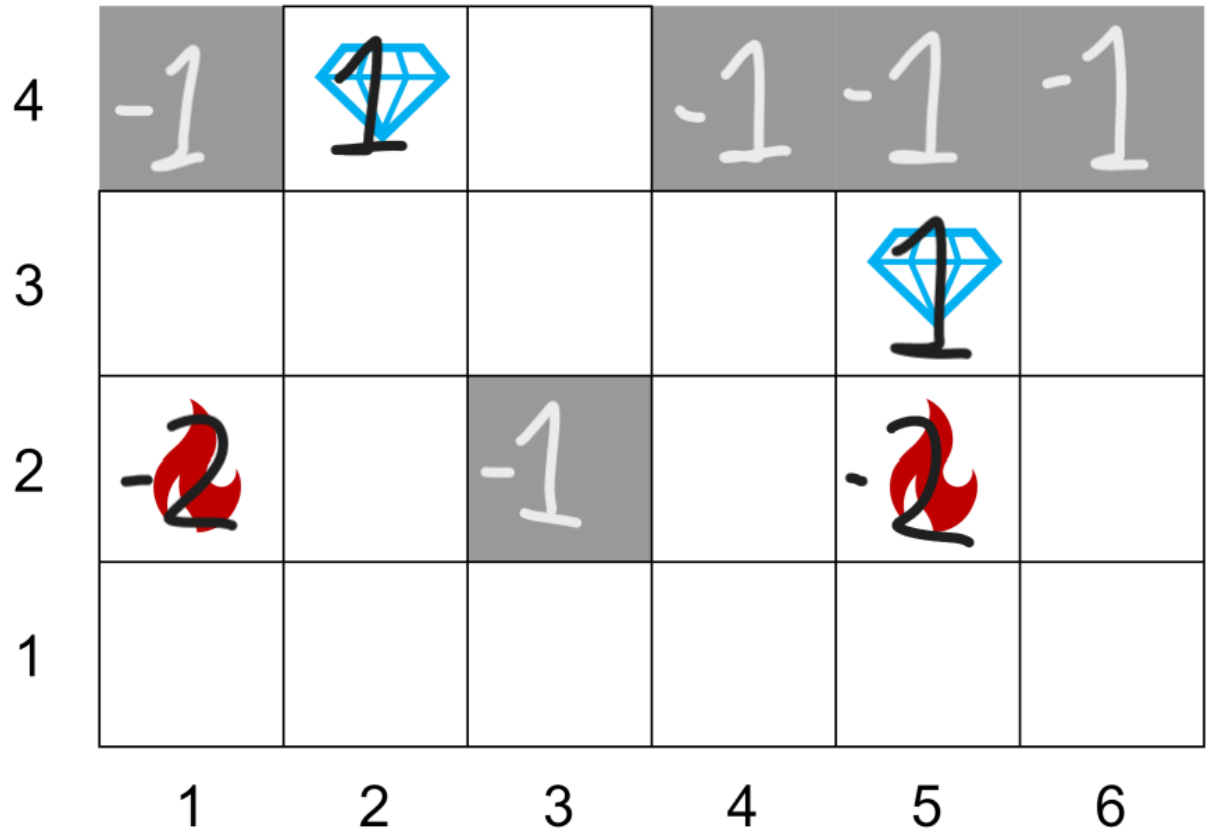
Figure 2: Treasure Map

$$R(s,a,s') = \begin{cases} 1 & s' = \text{End}, s \in \{(2,4),(5,3)\} \\ -2 & s' = \text{End}, s \in \{(1,2),(5,2)\} \\ -1 & s' \in \text{wall states} \\ -1 & s' \text{ outside the map} \\ 0 & \textit{otherwise} \end{cases}$$

The value iteration is defined as:

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

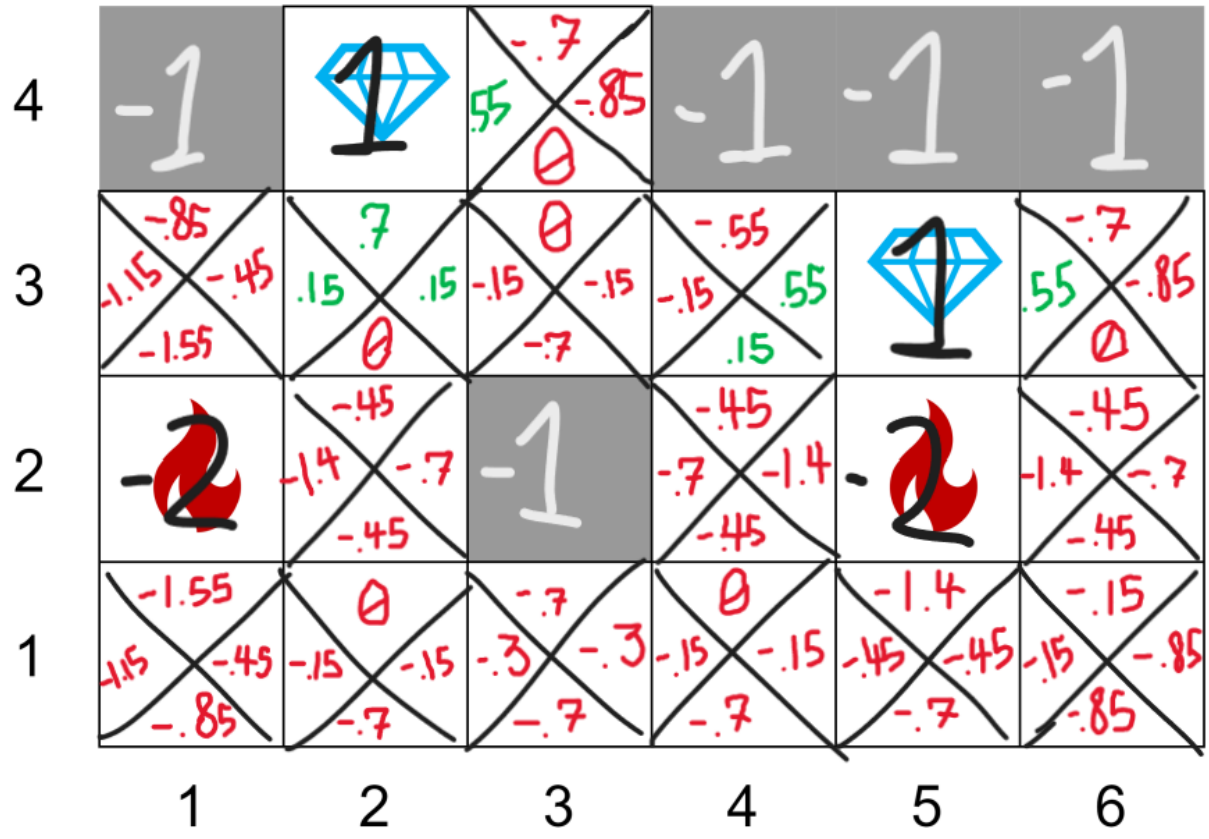
Where gamma, the discount factor, is equivalent to 1.0 (no discount), and all initial values are 0. Known spot values are shown below:



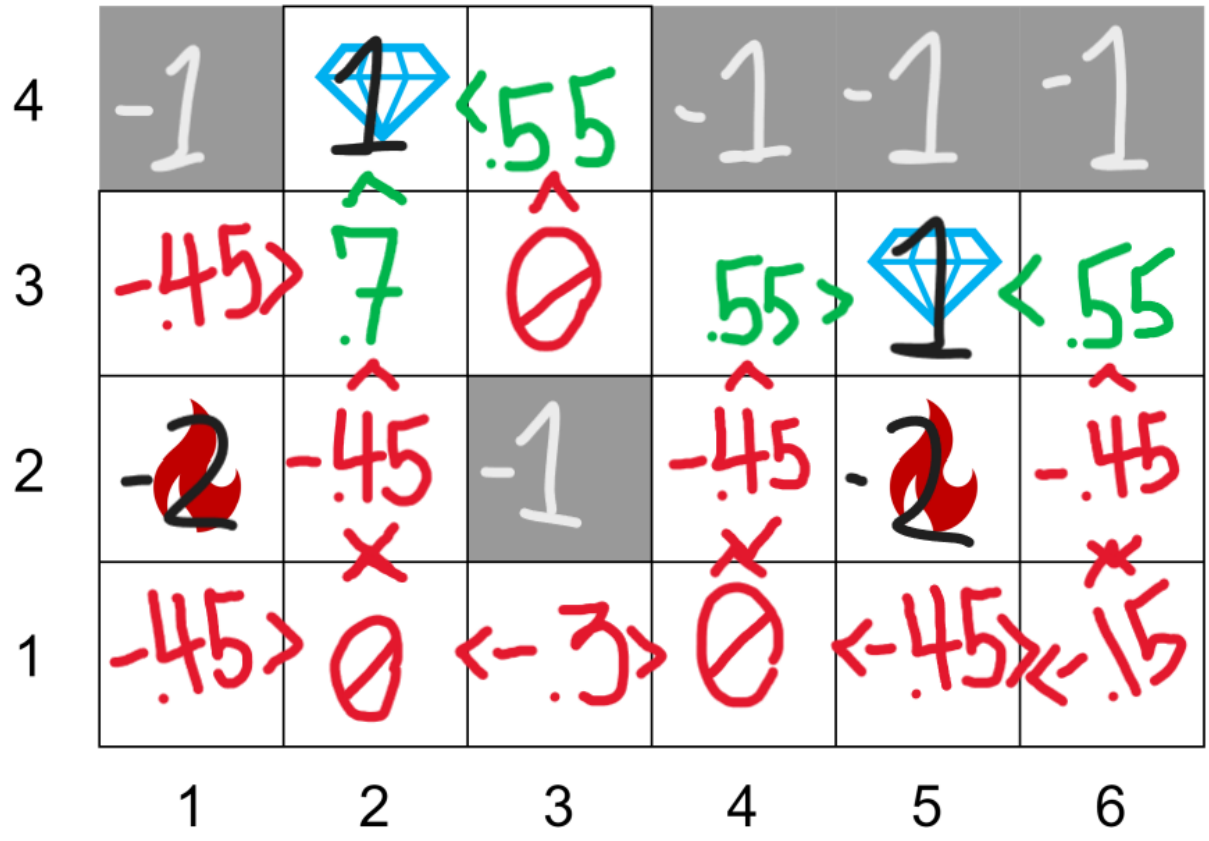
Where all outside grid spaces are given the value of -1.

a. What are the Q values after the first update (k=1)?

Q Values:

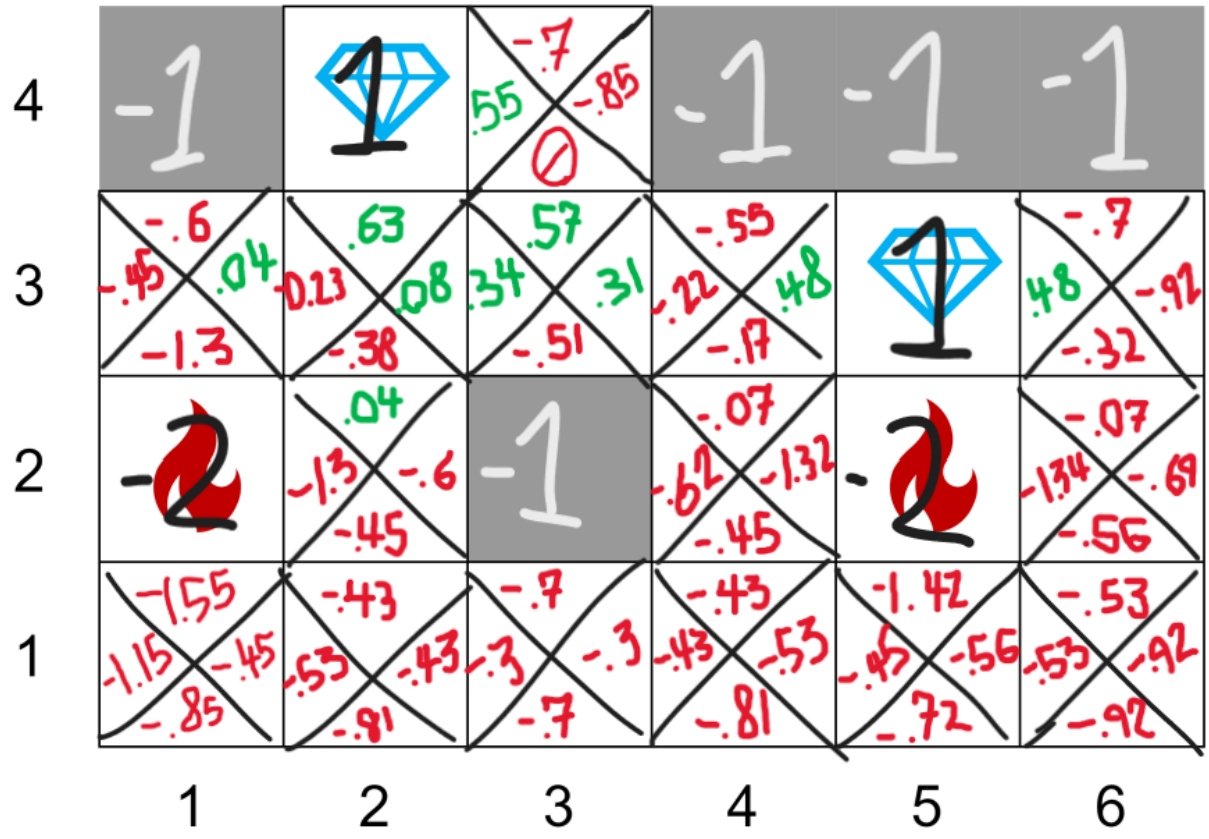

4
3
2
1
1
2
3
4
5
6
-1
1
-.7
.55
-.85
0
-1
-1
-1
-.85
-1.15
-.45
-1.55
.7
.15
.15
0
0
-.15
-.15
-.7
-.55
-.15
.55
.15
1
-.7
.55
-.85
0
-2
-.45
-1.4
-.7
-.45
-1
-.45
-.7
-1.4
-.45
-2
-.45
-1.4
-.7
-.45
-1.55
-1.15
-.45
-.85
0
-.15
-.15
-.7
-.7
-.3
-.3
-.7
0
-.15
-.15
-.7
-1.4
-.45
-.45
-.7
-.15
-.15
-.85
-.85

Policy Map:

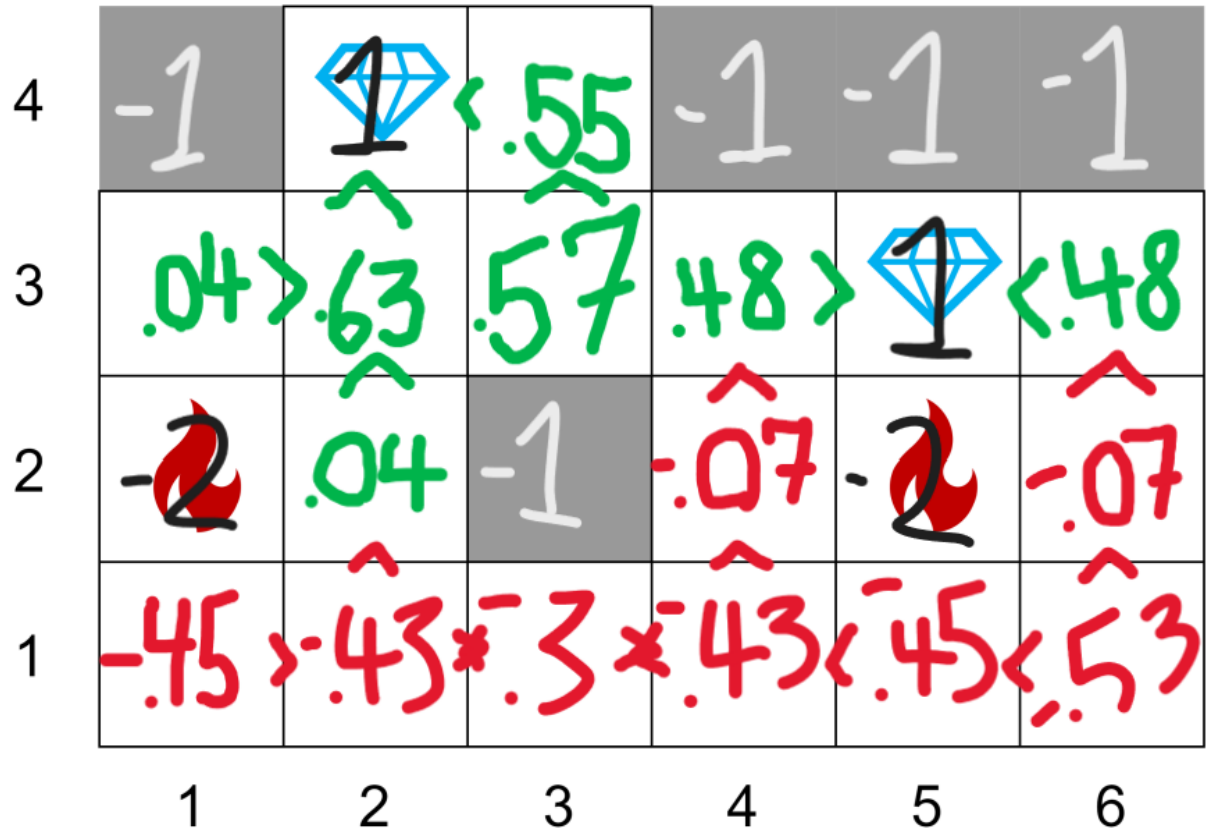


b. What are the Q values after the second update (k=2)?

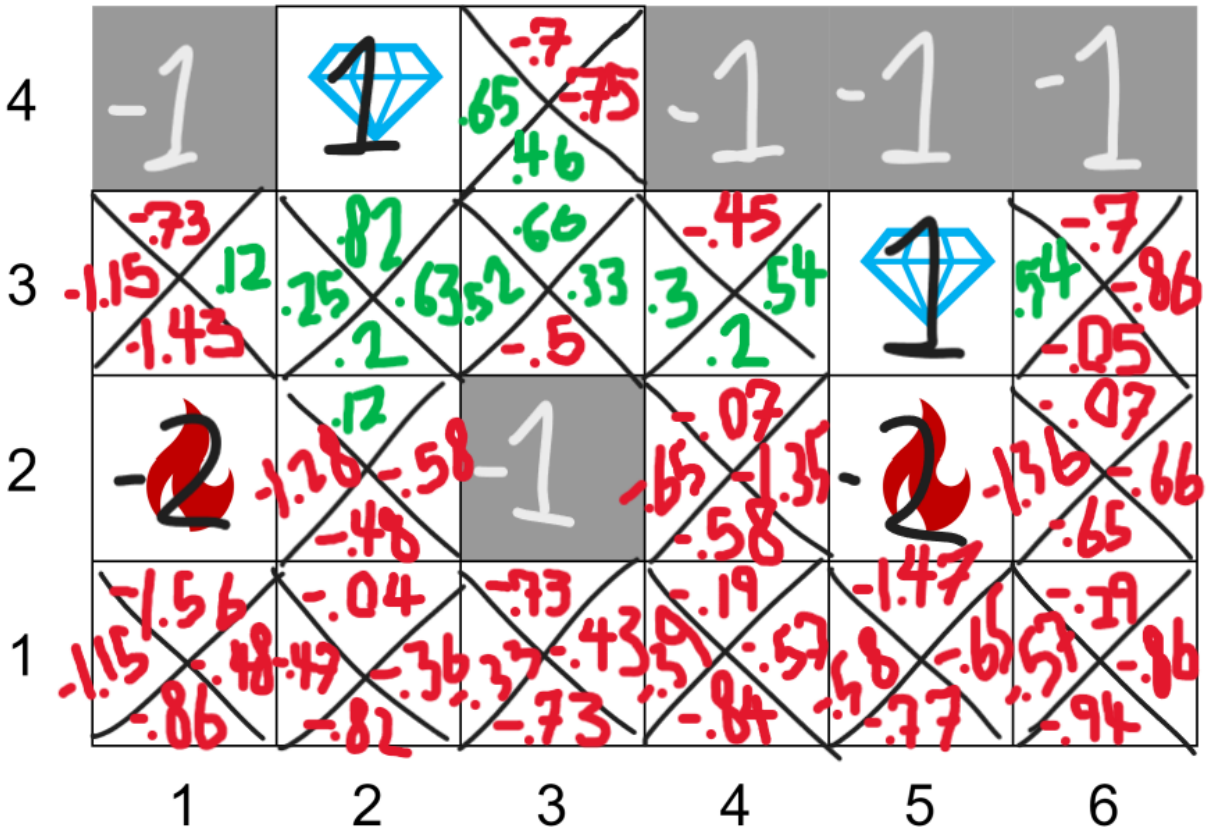
Q-Values:


4
3
2
1
1
2
3
4
5
6
-1
-1
-1
-1
-1
-1
-.7
.55
-.85
0
-.6
-.45
.04
-1.3
.63
-0.23
.08
-.38
.57
.34
.31
-.51
-.55
-.22
.48
-.17
-.7
.48
-.92
-.32
-2
.04
-1.3
-.6
-.45
-1
-.07
-.62
-1.32
-.45
-2
-.07
-1.34
-.69
-.56
-1.55
-1.15
-.45
-.85
-.43
-.53
-.43
-.81
-.7
-.3
-.3
-.7
-.43
-.43
-.53
-.81
-1.42
-.45
-.56
-.72
-.53
-.53
-.92
-.92

Policy Map:

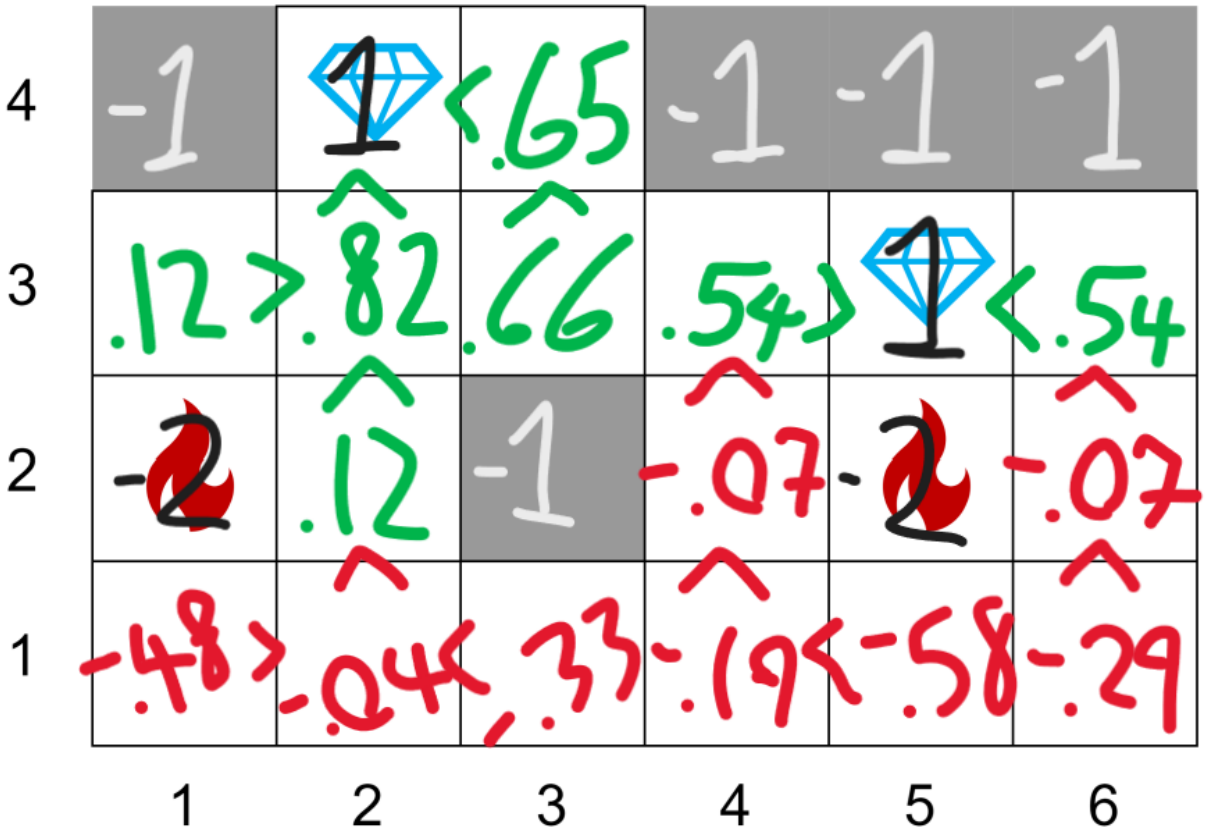


c. What are the converged Q values?



d. There is no D?

e. From part c, what are the V values derived from converged Q values?



f. What is the optimal policy map from converged Q values?
Please refer to the previous picture!

g. If we are dropped at state (3,1), what is the path suggested by the optimal policy map?
Left, up, up, up