AI REPORT

1.1)

```
Iteration:  3
3.3          -           -0.701          33.0
-3.103          -3.62           -3.397          -0.701
-3.62           -3.661          -            -3.424
-3.661          -3.663          -6.6            -3.661


POLICY :
-   -   E   -
N   W   E   N
N   N   -   N
N   W   -   E
```

With Gamma = 0.1, and Step Cost as -X/10 = -3.3
This took 3 iteration of VI algorithm. Since the gamma is 0.1, it finishes sooner. We can see that since the discount factor is so steep, it tries to reach a possible end state as soon as possible. This is the expected behaviour.

1.2)

```
Iteration:  16
3.3             -               27.983          33.0
13.025          18.99           24.01           27.983
9.705           14.093          -               23.519
5.699           7.762           -6.6            16.286


POLICY :
-   -   E   -
E   E   E   N
E   N   -   N
N   N   -   N
```

With Gamma = 0.99, and Step Cost as -X/10 = -3.3
This took 16 iterations to finish. Since step cost isn't as high, the MDP tries to reach the state with maximum reward

2.1)

```
Iteration:  70
3.3          -            1678.508            33.0
1678.508             1678.508             1678.508             1678.471
1678.508             1678.508             -            1678.469
1678.508             1678.508             -6.6             1678.429


POLICY :
-    -    W    -
S    N    W    S
N    N    -    N
N    W    -    E
```

With Gamma = 0.99 and Step Cost as X = 33
This tries to avoid hitting a goals state as much as possible since merely traversing through the states indefinitely gives maximum reward

2.2)

```
Iteration:  13
3.3              -                23.423           33.0
-2.269           6.276            15.838           23.423
-9.66            -2.883           -                14.901
-17.107          -11.258          -6.6             5.047


POLICY :
-    -    E    -
E    E    E    N
N    N    -    N
N    N    -    N
```

With Gamma = 0.99 and Step Cost as -X/5 = -6.6. Since the step cost isn't as rewarding, we try to reach the Goal state with +X reward as soon as possible. We can see a difference from the previous case where going to an end state was totally avoided in that case

2.3)

```
Iteration:  15
3.3          -           21.142          33.0
-6.259          -0.028          11.751          21.142
-15.859         -10.924         -           10.591
-24.874         -16.159         -6.6            -0.572


POLICY :
-   -   E   -
N   E   E   N
N   N   -   N
N   E   -   N
```

With Gamma = 0.99 and Step Cost as -X/4 = -8.25
Since the step cost is more penalising, we see a difference in the policy of the cell (3,1) which now points to the terminal state on the left as opposed to traversing all the way to the most rewarding end state.

2.4)

```
Iteration:  7
3.3             -               -13.039         33.0
-42.913         -84.207         -49.475         -13.039
-84.207         -91.739         -               -53.956
-91.739         -52.437         -6.6            -48.336


POLICY :
-   -   E   -
N   W   E   N
N   S   -   N
E   E   -   W
```

Gamma = 0.99 and Step Cost = -X = -33
This is very very penalising for traversing the states, hence it tries to reach an end state as soon as possible. Even the negative reward end state is preferred over traversing the MDP