CS 440: Intro to Artificial Intelligence — Assignment 2B
Fall 2022 — Due December 14, 2022, 11:59 p.m.

---

**Name**: Will Chen (whc27, Section 2)
**Name**: Noel Declaro (nd680, Section 1)
**Name**: Afsana Rahman (ar1641, Section 2)

**Overview**: This document contains answers to all questions, Problems 5-7, for Assignment 2. The source code for Problems 5 and 7 should be in the zip folder.

---

**Question 5 (35 points):** Please refer to the `README.txt` for implementation details.

(a) **3x3 grid.** For the actions $Right, Right, Down, Down$ and the readings $N, N, H, H$, we have the following probability distributions after each step:

- After step 1 $(Right, N)$:

$$\begin{matrix} 0.00132 & 0.01315 & 0.02500 \\ 0.02368 & 0.23684 & 0.45000 \\ 0.23684 & 0.00000 & 0.01315 \end{matrix}$$

- After step 2$(Right, N)$:

$$\begin{matrix} 7.7 * 10^{-}6 & 0.00001 & 0.00215 \\ 0.00249 & 0.04735 & 0.69784 \\ 0.24923 & 0.00000 & 0.00008 \end{matrix}$$

- After step 3$(Down, H)$:

$$\begin{matrix} 1.1 * 10^{-}6 & 2.2 * 10^{-}5 & 1.8 * 10^{-}5 \\ 2.2 * 10^{-}5 & 0.00406 & 0.00614 \\ 0.02151 & 0.00000 & 0.96822 \end{matrix}$$

- After step 4$(Down, H)$:

$$\begin{matrix} 1.2 * 10^{-}7 & 2.3 * 10^{-}6 & 1.0 * 10^{-}7 \\ 1.9 * 10^{-}7 & 0.00002 & 3.6 * 10^{-}5 \\ 0.00122 & 0.00000 & 0.99850 \end{matrix}$$

(b) **100x50 grids.** Please refer to the video demonstration folder in the assignment zip folder.

(c) **Heat maps.** As above, implementation is shown in the videos provided in the implementation zip folder.

---

**Question 6 (10 points):**

(a) **Expected net gain.** The expected net gain of buying $C_1$ would be equal to the 70% chance of a \$1000 net gain (if $C_1$ is in good shape) plus the the 30% chance of a \$400 loss (if $C_1$ is in bad shape), which is \$1000*0.7 + (-\$400)*0.3 = \$580.

(b) **Bayes' Theorem.** The following probabilities are calculated using Bayes' Theorem:

- $P(q^+|pass) = \frac{P(pass|q^+)*P(q^+)}{P(pass|q^+)*P(q+)+P(pass|q^-)*P(q-)} = \frac{0.8*0.7}{0.8*0.7+0.35*0.3} = 0.842$
- $P(q^-|pass) = \frac{P(pass|q^-)*P(q^-)}{P(pass|q^+)*P(q+)+P(pass|q^-)*P(q-)} = \frac{0.35*0.3}{0.8*0.7+0.35*0.3} = 0.158$
- $P(q^+|\neg pass) = \frac{P(\neg pass|q^+)*P(q^+)}{P(\neg pass|q^+)*P(q+)+P(\neg pass|q^-)*P(q-)} = \frac{0.2*0.7}{0.2*0.7+0.65*0.3} = 0.418$
- $P(q^-|\neg pass) = \frac{P(\neg pass|q^-)*P(q^-)}{P(\neg pass|q^+)*P(q+)+P(\neg pass|q^-)*P(q-)} = \frac{0.65*0.3}{0.2*0.7+0.65*0.3} = 0.582$

(c) **Expected Utility.** If given a "pass" from the mechanic, then the expected utility of buying the car would be $\$900 * P(q^+|pass) + (-\$500) * P(q^-|pass) = \$678.80$; since it is a positive expected utility, it would be best to buy the car. If given a "fail" from the mechanic, then the expected utility of buying the car would be $\$900 * P(q^+|\neg pass) + (-\$500) * P(q^-|\neg pass) = \$85.20$; again, since it is a positive expected utility, it would still be best to buy the car as there is still some expected net gain.

(d) **Decision.** The value of optimal information would be

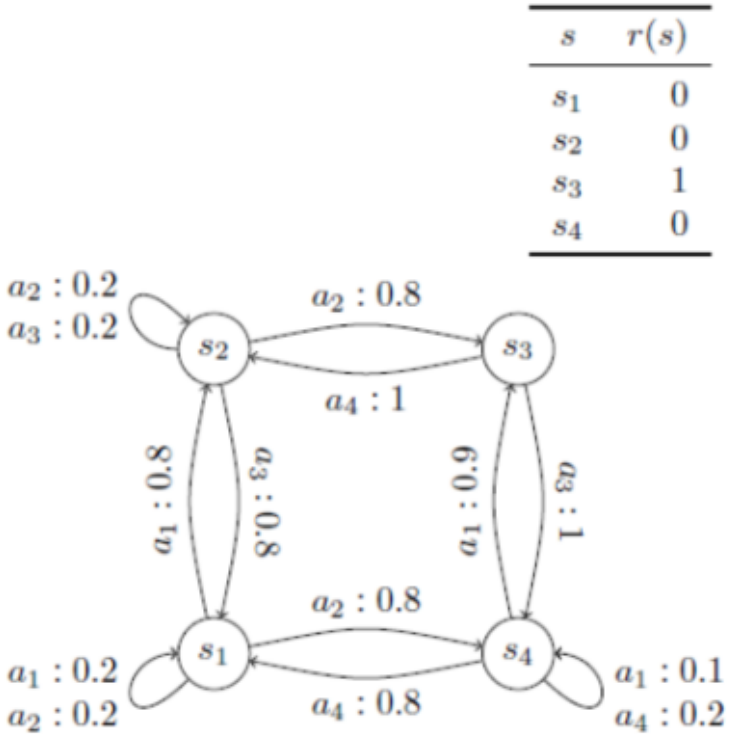
$$P(pass) * expected\_util(pass) + P(\neg pass) * expected\_util(\neg pass) \quad (1)$$

which gives us (0.8*0.7 + 0.35*0.3)(678.8) + (0.2*0.7 + 0.65*0.3)(85.2) = \$479.94. Since this value is less than the initial expected net gain of \$580 found in part (a), it would be better *not* to take the car to the mechanic.

---

**Question 7 (15 points):**

The process started by assigning the reward values of each state as their respective values (i.e., s1, s2, and s4 = 0, s3 = 1). The gamma value was set to 0.9 and yielded 178 iterations in 0.0100 seconds. In the implementation, the `valuePolicy(double gamma)` function helps compute the Bellman equation, where the maximums are saved into the `value` and `policy` instance variables of each state. We check/terminate for convergence when the previous states' values equal the current states' values. To run the code, run `javac State.java`, then `java State`, which will then print each iteration's policies and values (as shown in Figure 2) and the time and number of iterations at the end.




| $s$ | $r(s)$ |
|---|---|
| $s_1$ | 0 |
| $s_2$ | 0 |
| $s_3$ | 1 |
| $s_4$ | 0 |

| $s_i$ | $a$ | $s_j$ | $T(s_i, a, s_j)$ |
|---|---|---|---|
| $s_1$ | $a_1$ | $s_1$ | 0.2 |
| $s_1$ | $a_1$ | $s_2$ | 0.8 |
| $s_1$ | $a_2$ | $s_1$ | 0.2 |
| $s_1$ | $a_2$ | $s_4$ | 0.8 |
| $s_2$ | $a_2$ | $s_2$ | 0.2 |
| $s_2$ | $a_2$ | $s_3$ | 0.8 |
| $s_2$ | $a_3$ | $s_2$ | 0.2 |
| $s_2$ | $a_3$ | $s_1$ | 0.8 |
| $s_3$ | $a_4$ | $s_2$ | 1 |
| $s_3$ | $a_3$ | $s_4$ | 1 |
| $s_4$ | $a_1$ | $s_4$ | 0.1 |
| $s_4$ | $a_1$ | $s_3$ | 0.9 |
| $s_4$ | $a_4$ | $s_4$ | 0.2 |
| $s_4$ | $a_4$ | $s_1$ | 0.8 |

Figure 1: Problem 7 network and corresponding probability/action table.

```
--- ITERATION 1:---
POLICY: a2, VALUE: 0.0
POLICY: a2, VALUE: 0.7200000000000001
POLICY: a4, VALUE: 1.6480000000000001
POLICY: a1, VALUE: 1.33488
--- ITERATION 2:---
POLICY: a2, VALUE: 0.9611136000000002
POLICY: a2, VALUE: 1.3161600000000002
POLICY: a3, VALUE: 2.2013920000000002
POLICY: a1, VALUE: 1.9032667200000004
--- ITERATION 3:---
POLICY: a2, VALUE: 1.5433524864000003
POLICY: a2, VALUE: 1.8219110400000003
POLICY: a3, VALUE: 2.712940048
POLICY: a1, VALUE: 2.36877544368
--- ITERATION 4:---
POLICY: a2, VALUE: 1.9833217670016001
POLICY: a2, VALUE: 2.28126082176
POLICY: a3, VALUE: 3.1318978993120004
POLICY: a1, VALUE: 2.750027088373921
--- ITERATION 5:---
POLICY: a2, VALUE: 2.337017421689511
POLICY: a2, VALUE: 2.6655934354214406
POLICY: a3, VALUE: 3.4750243795365288
POLICY: a1, VALUE: 3.0622721853782413
```

(a)

```
--- ITERATION 24:---
POLICY: a2, VALUE: 3.893552976160534
POLICY: a2, VALUE: 4.375118521109137
POLICY: a3, VALUE: 4.9926737850453335
POLICY: a1, VALUE: 4.443333144391254
--- ITERATION 25:---
POLICY: a2, VALUE: 3.900039399670599
POLICY: a2, VALUE: 4.382246459032285
POLICY: a3, VALUE: 4.998999829952128
POLICY: a1, VALUE: 4.449089845256437
--- ITERATION 26:---
POLICY: a2, VALUE: 3.905351780525343
POLICY: a2, VALUE: 4.388084240191343
POLICY: a3, VALUE: 5.004180860730793
POLICY: a1, VALUE: 4.453804583265022
--- ITERATION 27:---
POLICY: a2, VALUE: 3.9097026204453775
POLICY: a2, VALUE: 4.392865382960613
POLICY: a3, VALUE: 5.00842412493852
POLICY: a1, VALUE: 4.457665953694053
--- ITERATION 28:---
POLICY: a2, VALUE: 3.9132659583398866
POLICY: a2, VALUE: 4.396781138888644
POLICY: a3, VALUE: 5.011899358324648
POLICY: a1, VALUE: 4.4608284160754295
```

(b)

```
--- ITERATION 174:---
POLICY: a2, VALUE: 3.929389570138794
POLICY: a2, VALUE: 4.414499393612719
POLICY: a3, VALUE: 5.027624309392263
POLICY: a1, VALUE: 4.47513812154696
--- ITERATION 175:---
POLICY: a2, VALUE: 3.929389570138795
POLICY: a2, VALUE: 4.4144993936127195
POLICY: a3, VALUE: 5.027624309392264
POLICY: a1, VALUE: 4.475138121546961
--- ITERATION 176:---
POLICY: a2, VALUE: 3.9293895701387953
POLICY: a2, VALUE: 4.41449939361272
POLICY: a3, VALUE: 5.027624309392265
POLICY: a1, VALUE: 4.475138121546961
--- ITERATION 177:---
POLICY: a2, VALUE: 3.9293895701387953
POLICY: a2, VALUE: 4.414499393612721
POLICY: a3, VALUE: 5.027624309392265
POLICY: a1, VALUE: 4.475138121546961
--- ITERATION 178:---
POLICY: a2, VALUE: 3.9293895701387953
POLICY: a2, VALUE: 4.414499393612721
POLICY: a3, VALUE: 5.027624309392265
POLICY: a1, VALUE: 4.475138121546961
```

(c)

Figure 2: Iteration results.