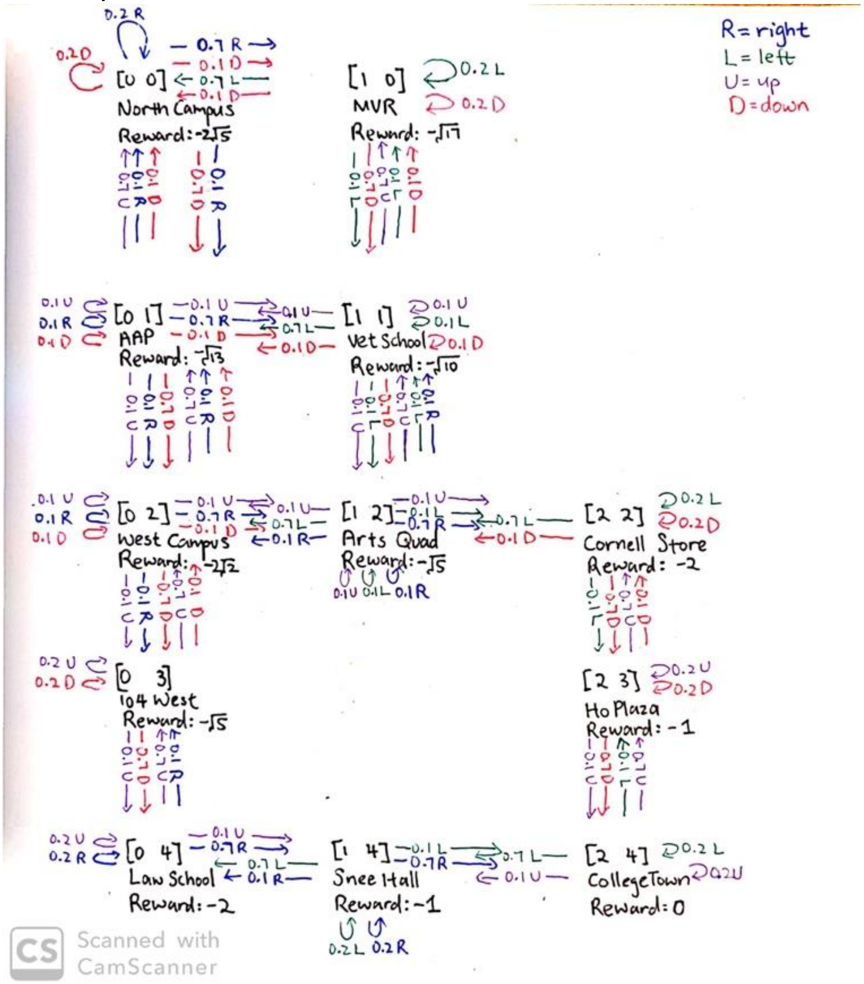
1. MPD and Utility Function



   a.
   b. Let $N = U^{\pi}(North\ Campus)$, $V = U^{\pi}(Vet\ School)$, $H = U^{\pi}(Ho\ Plaza)$, $W = U^{\pi}(104\ West)$

$$N = 0.7\left(-\sqrt{13} + 0.5U^{\pi}([0\ 1])\right) + 0.1\left(-\sqrt{17} + 0.5U^{\pi}([1\ 0])\right) + 0.2(-2\sqrt{5} + 0.5N)$$

$$V = 0.7\left(-\sqrt{5} + 0.5U^{\pi}([1\ 2])\right) + 0.1\left(-\sqrt{17} + 0.5U^{\pi}([1\ 0])\right) + 0.1(-\sqrt{13} + 0.5U^{\pi}([0\ 1]) + 0.1(-\sqrt{10} + 0.5U^{\pi}V)$$

$$H = 0.7(0 + 0.5U^{\pi}([2\ 4])) + 0.1(-2 + 0.5U^{\pi}([2\ 2])) + 0.2(-1 + 0.5H)$$

$$W = 0.7(-2 + 0.5U^{\pi}([0\ 4])) + 0.1\left(-2\sqrt{2} + 0.5U^{\pi}([0\ 2])\right) + 0.2(-\sqrt{5} + 0.5W)$$

2. Discount Rewards
   a. 1) If the environment does not contain a terminal state or if the agent never reaches one, then all environment histories will be infinitely long, and utilities will be infinite, so you can't compare state sequences then.
      2) Discounting favors near-term rewards, which is what we want to do, but you can't do that if you're just adding up all future rewards as is.

b. Lower bound is when $R(s_t, a_t, s_{t+1}) = R_{\text{min.}}$
   $\sum_{t=0}^{\infty} \gamma^t R_{\text{min.}} = \frac{R_{\text{min.}}}{1-\gamma}$
   Upper bound is when $R(s_t, a_t, s_{t+1}) = R_{\text{max.}}$
   $\sum_{t=0}^{\infty} \gamma^t R_{\text{max.}} = \frac{R_{max}}{1-\gamma}$
   Anything between the two bounds is also finite.

c. $\gamma^5(1) = 2\gamma^{10}$
   $\frac{1}{2} = \gamma^5$
   $\boldsymbol{\gamma = 0.871}$

d. You should **decrease** γ because now you have $0.871 > \gamma$

3. Policy Iteration
   a. There are 2 states and 3 actions in this environment. $2 \times 3 = \mathbf{6}$
   b. $s \times a$
   c. :

$$\hat{\pi}_0(low) = searching$$
$$\hat{\pi}_0(high) = searching$$
$$\widehat{U}_0(low) = 0$$
$$\widehat{U}_0(high) = 0$$

$i = 1$

$$\widehat{U}_1(low) = 0.4\left(-10 + 0.5\widehat{U}_0(high)\right) + 0.6\left(4 + 0.5\widehat{U}_0(low)\right) = -1.6$$
$$\widehat{U}_1(high) = 0.7\left(4 + 0.5\widehat{U}_0(high)\right) + 0.3\left(4 + 0.5\widehat{U}_0(low)\right) = 4$$
$$\hat{\pi}_1(low) = \underset{a\in A}{\operatorname{argmax}}(\widehat{U}_1(search), \widehat{U}_1(wait), \widehat{U}_1(\boldsymbol{recharge}))$$
$$= \underset{a\in A}{\operatorname{argmax}}(-1.28, 0.2, 2)$$
$$\hat{\pi}_1(high) = \underset{a\in A}{\operatorname{argmax}}(\widehat{U}_1(\boldsymbol{search}), \widehat{U}_1(wait), \widehat{U}_1(recharge)) = \underset{a\in A}{\operatorname{argmax}}(4, 3, 2)$$

$i = 2$

$$\widehat{U}_2(low) = 1\left(0 + 0.5\widehat{U}_1(high)\right) = 2$$
$$\widehat{U}_2(high) = 0.7\left(4 + 0.5\widehat{U}_1(high)\right) + 0.3\left(4 + 0.5\widehat{U}_1(low)\right) = 5.16$$
$$\hat{\pi}_2(low) = \underset{a\in A}{\operatorname{argmax}}(\widehat{U}_2(search), \widehat{U}_2(wait), \widehat{U}_2(\boldsymbol{recharge}))$$
$$= \underset{a\in A}{\operatorname{argmax}}(-0.6, 2, 2.58)$$
$$\hat{\pi}_2(high) = \underset{a\in A}{\operatorname{argmax}}(\widehat{U}_2(\boldsymbol{search}), \widehat{U}_2(wait), \widehat{U}_2(recharge))$$
$$= \underset{a\in A}{\operatorname{argmax}}(5.16, 3.58, 2.58)$$

Converge!

4. Q-learning
   a. The Q-value is the expected reward from taking an action and the expected utility of the state we end up in from taking that action.

b. If $\alpha = 1$, then $Q(s,a) = R(s,a,s') + \gamma \max_{a'} Q(s',a')$, i.e. the Q-value we learned for the state-action pair will be gone; the Q-function will also be deterministic.
   If $\alpha = 0$, then $Q(s,a)$ is not changing.
   If $\alpha > 1$, then we are overestimating the reward.
   If $\alpha < 0$, then reward will be negative
c. It ensures we always take the best action, thus learning the best experience
d. $Q([0\ 0], right) = 0.7\left(-\sqrt{17} + 0.5U^*([1\ 0])\right) + 0.1\left(-\sqrt{13} + 0.5U^*([0\ 1])\right) + 0.2\left(-2\sqrt{5} + 0.5U^*([0\ 0])\right)$

   $Q([0\ 0], down) = 0.7\left(-\sqrt{13} + 0.5U^*([0\ 1])\right) + 0.1\left(-\sqrt{17} + 0.5U^*([1\ 0])\right) + 0.2\left(-2\sqrt{5} + 0.5U^*([0\ 0])\right)$