Q.1.

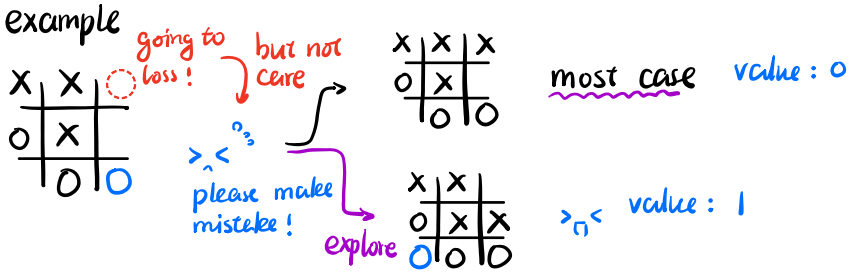
a) Yes, it will learn a different policy.

Given all positions being explored by enough time, the agents should result in a policy that play optimally to play with itself by construction.

Note that as both players will explore sometimes, all states should be visited for enough time. Given both players being able to perform optimal at each state against with self-playing, any occational exploration would be leveraged to win. Also, since value of loss and draw are identical (0), both players would not avoid losses.

The simulation result in programming part also verifies this. The X players tend to win O players at a high probability, while O players is not mad at all. Instead, he/she is trying to wait any exploration ("mistake") of X-player so that he/she can win, even if this may cause loss. (The last sentence is imagination).

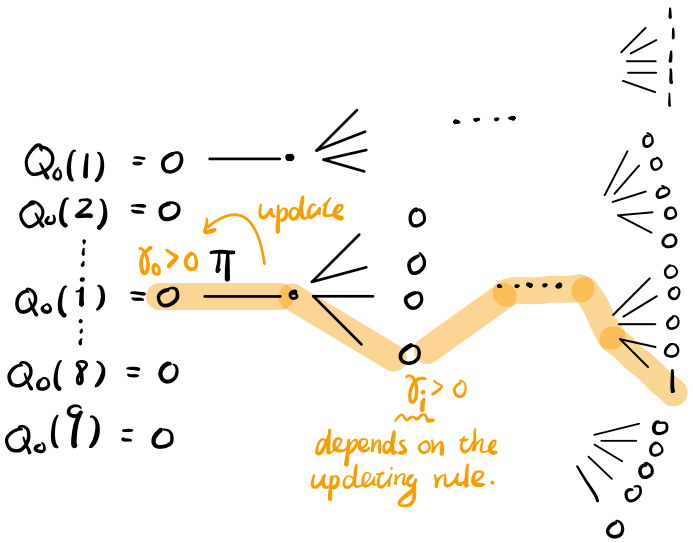
For example



Instead, if we change the loss value to be -0.1, (Not presented in the notebook, though), the self-play will tend to converge to nearly always draw.

b)

In most case, greedy player should learn worse than ε-greedy player. But it depends on how the value-table is set up, as well. In the setting of this specific question, To elastrate, take an extreme case with the opponent π player silly if $A_0=1$, but very smart for any other cases. (for example, optimal self-play policy, learn except $A_0=1$ case).



Clearly, $A_0=1$ is the "optimal" action in this example. But suppose the first action selected by the greedy player is $A_0=i$. And by accident, (maybe the opponent is exploring) it wins. This will update the value of $Q_0(i) > 0$ and keeps on positive all the way. This implies that the $A_0 = i$ is always greedy and always be selected.

Clearly, sometimes greedy is better if the value table has already been "optimal", for example. But in general, greedy-player should perform worse than ε-greedy player, as it is easy to suck in a "local optimal state."

c) Notice that symmetries result from rotations and flips. To be more concise, denote operators "pretty-print" $P : \mathbb{R}^9 \to \mathbb{R}^{3\times 3}$ as

$$P(S) = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \qquad S = (a_{11}, a_{12}, a_{13}, a_{21}, a_{22}, a_{23}, a_{31}, a_{32}, a_{33})^T$$

which clearly is bijective, and "rotation $90°\circlearrowright$" $R : \mathbb{R}^{3\times 3} \to \mathbb{R}^{3\times 3}$ as well as "horizontal flip $-\!\!\updownarrow\!\!-$" : $F : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}^3 \times \mathbb{R}^3$ s.t.

$$R \circ P(S) = \begin{bmatrix} a_{13} & a_{23} & a_{33} \\ a_{12} & a_{22} & a_{32} \\ a_{11} & a_{21} & a_{31} \end{bmatrix} \qquad F \circ P(S) = \begin{bmatrix} a_{31} & a_{32} & a_{33} \\ a_{21} & a_{22} & a_{23} \\ a_{11} & a_{12} & a_{13} \end{bmatrix}$$

both of which are also bijective. Note that the operations of rotation causing symmetries are $\{I, R, R^2, R^3\}$, i.e, rotate by $90°, 180°, 270°$. and the flips causing symmetries are $\{I, F_1, F_2, F_3, F_4\}$, i.e, flip by horizontal ⋯, vertical ⋮, diagonal ⋱, ⋰ respectively. Futher, note that

$$\{I, F_1, F_2, F_3, F_4\} = \{I, F, FoR^2, FoR^3, FoR\}$$

Hence, all operations resulting in symmetries are

$$\mathcal{O} = \{I, R, R^2, R^3, F, FoR, FoR^2, FoR^3\}$$

That is, $\forall\ S = (a_{11}, a_{12}, a_{13}, a_{21}, a_{22}, a_{23}, a_{31}, a_{32}, a_{33})^T$, $S' = (a'_{11}, a'_{12}, a'_{13}, a'_{21}, a'_{22}, a'_{23}, a'_{31}, a'_{32}, a'_{33})^T$

$S \sim S'$, i.e, S is equivalent S' due to symmetry iff $\exists\ O \in \mathcal{O}$ s.t.

$$P^{-1} \circ O \circ P(S) = S' \in \mathbb{R}^9$$

which is as follows (Here I only represent to $P(S')$).

$$I \circ P(S) = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \qquad R \circ P(S) = \begin{bmatrix} a_{13} & a_{23} & a_{33} \\ a_{12} & a_{22} & a_{32} \\ a_{11} & a_{21} & a_{31} \end{bmatrix} \circlearrowright$$

$$R^2 \circ P(S) = \begin{bmatrix} a_{33} & a_{32} & a_{31} \\ a_{23} & a_{22} & a_{21} \\ a_{13} & a_{12} & a_{11} \end{bmatrix} \quad R^3 \circ P(S) = \begin{bmatrix} a_{31} & a_{21} & a_{11} \\ a_{32} & a_{22} & a_{12} \\ a_{33} & a_{23} & a_{13} \end{bmatrix}$$

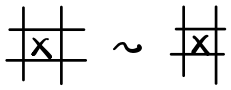
$$F \circ P(S) = \begin{bmatrix} a_{13} & a_{12} & a_{11} \\ a_{23} & a_{22} & a_{21} \\ a_{33} & a_{32} & a_{31} \end{bmatrix} \quad F \circ R \circ P(S) = \begin{bmatrix} a_{11} & a_{21} & a_{33} \\ a_{12} & a_{22} & a_{32} \\ a_{13} & a_{23} & a_{33} \end{bmatrix}$$

$$F \circ R^2(S) = \begin{bmatrix} a_{31} & a_{32} & a_{33} \\ a_{21} & a_{22} & a_{23} \\ a_{11} & a_{12} & a_{13} \end{bmatrix} \quad F \circ R^3 \circ P(S) = \begin{bmatrix} a_{33} & a_{23} & a_{13} \\ a_{32} & a_{22} & a_{12} \\ a_{31} & a_{21} & a_{11} \end{bmatrix}$$

The above analysis ensures that the above S' consist of ALL states equivalent to S. And we have

$$[S] = \{ S' : S' = P^{-1} \circ O \circ P(S), \text{ for some } O \in \mathcal{O} \}. \qquad \square$$

d) By applying $o \in \mathcal{O}$ to each state S into its equivalent class form $[S]$, the set number $|S'|$, where $S' = \{[S]\}$, is much less than S, where $S = \{S\}$ is the total identical states without considering equivalent relation.

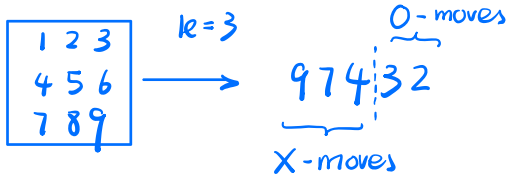
Note that $|S'| > \frac{1}{8}|S|$ as, for example,  only, as well as  only. In fact, by wiki,

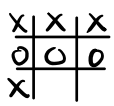
denoted → $|S''|$

$$|S| = 5478 \leq \boxed{\sum_{k=1}^{5} \binom{9}{k}\binom{9-k}{k-1} + \sum_{k=0}^{4} \binom{9}{k}\binom{9-k}{k}} = 6047$$

$$|S'| = 765 \approx \frac{1}{7.16}|S|$$

e.g.



S'' is how I implemented my value table. The exact $|S| = 5478$ is time-consuming as I need to eleminate impossible states close to case by case. for example,  , which is not quite necessary as $6047 \approx 5478$.

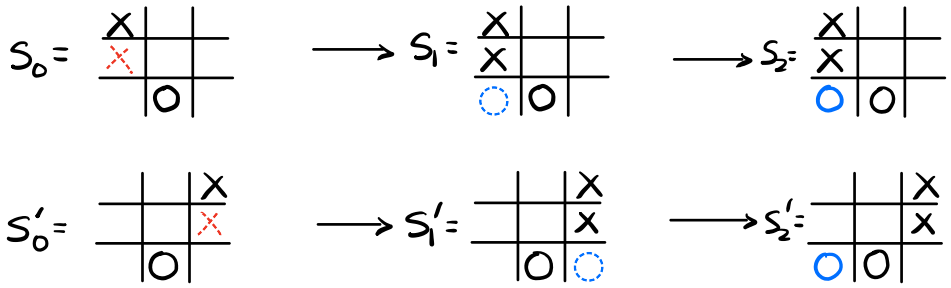
At least, it is much smaller than $512 \times 512 = 262144$ which is the state space of the author.

e) Under proper condition (as we'll see in Q.1(f)), taking this advantages means we have a mush smaller state space, hence reducing the memory requirements (value-table), and the time to learn (less game to play).

f)

It depends on the opponent's policy. In general, we should not, because using equivalent class $[S]$ instead of $S' \in [S]$ means that our policy views S and S' the same. This requires that our policy <u>assuming that our opponent act symmetrically over $S' \in [S]$</u>.

For example, suppose we are "X" and act firstly,



where $[S_0] = [S_0']$ implies that for us, the value of S_1 and S_1' should be the same. This is <u>under the assumption</u> that our opponent value a_{31} and a_{13} the same, i.e, $\pi(a_{31}|S_1) = \pi(a_{13}|S_1')$.

However, the opponent in this silly example view

$$\pi(a_{31}|S_1) = \pi(a_{31}|S_1') = 1$$
$$\pi(a_{13}|S_1) = \pi(a_{13}|S_1') = 0$$

and results in S_2 and S_2' respectively. In this example, it is clearly that, <u>if we know our opponent is silly</u>, we should not view S_0, S_0' the same.

Now suppose our opponent realizes it and slightly modifies his policy by (P_n), $(Q_n) \in [0, 1]$ each games as

$$\pi(a_{31}|S_1) = P_n \qquad \pi(a_{13}|S_1) = Q_n$$

we still need to value S_0, S_0' differently <u>until $P_n = Q_n$</u>.

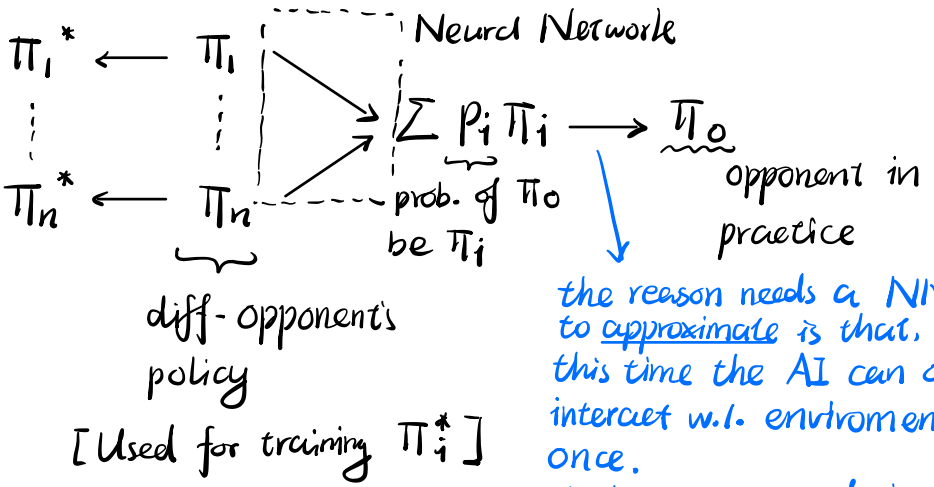
That is, in general, unless our opponent, though not realizing the

advantage of symmetries, having a symmetry policy by accident, we should not take the advantage of symmetries as well. Instead, it intimates that we should take the advantage of our opponents of not realizing it to defeat it.

A good solution, as discussed in the booked office-hour, is to carry up two value tables —— one symmetric $V_S'(\cdot)$ and one non-symmetric $V_S(\cdot)$.
Then, the value of each position is estimated by the average of $V_S'(\cdot)$ and $V_S(\cdot)$, and both tables update for each possition as well.
Note that although the advantage of memory saving is not obtained (as we have even one extra table), the advantage of learning speed is kept.
Unfortunately, given the limited time, I did not apply this into my program. I should really team up if I can predict this situation. The time I finish this sentence is Sunday 21:04. and my program has not yet been well-wrapped.

g) We may consider substitute $V(\cdot)$ by $V(\cdot|\theta)$. i.e, using generalising fne to approximate $V(\cdot)$, for example, neural network, so that the speed of learning may be improved.

h) As we discussed in the office hour, suppose our opponents are not fixed. Then, to win a general opponent as much as possible,

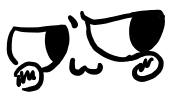


$$\Rightarrow \quad \pi_o^* = \sum p_i \pi_i^*$$

Just a whimsical imagination.

i) As the rewards are the same while addition is easyer to teach it encourages the agent only teachs addition and students never learn subtraction.

* This implies that Prof. Paul should only consider an easy final for his students as his optimal action, as his rewards are not related to the hardness level as well.



Q.2.

a)

Denote "□" as the greedy-action list, "■" as the actually selected action.

$Q_1(a^1) = 0$	$Q_1(a^2) = 0$	$Q_1(a^3) = 0$	$Q_1(a^4) = 0$

→ can be the ε-case.

$Q_2(a^1) = 1$	$Q_2(a^2) = 0$	$Q_2(a^3) = 0$	$Q_2(a^4) = 0$

→ must be the ε-case.

$Q_3(a^1) = 1$	$Q_3(a^2) = 1$	$Q_3(a^3) = 0$	$Q_3(a^4) = 0$

→ can be the ε-case.

$Q_4(a^1) = 1$	$Q_4(a^2) = 3/2$	$Q_4(a^3) = 0$	$Q_4(a^4) = 0$

→ can be the ε-case.

$Q_5(a^1) = 1$	$Q_5(a^2) = 5/3$	$Q_5(a^3) = 0$	$Q_5(a^4) = 0$

→ must be the ε-case.

b)

$$Q_{t+1}(a) = \frac{1}{N_{t+1}(a)} \sum_{i=1}^{t} R_i \cdot \mathbb{1}_{A_i = a}$$

$$= \frac{1}{N_{t+1}(a)} \left(R_t \mathbb{1}_{A_t = a} + \sum_{i=1}^{t-1} R_i \cdot \mathbb{1}_{A_i = a} \right)$$

$$= \frac{1}{N_{t+1}(a)} \left(R_t \mathbb{1}_{A_t = a} + N_t(a) \cdot \frac{1}{N_t(a)} \sum_{i=1}^{t-1} R_i \cdot \mathbb{1}_{A_i = a} \right)$$

$$= \frac{1}{N_{t+1}(a)} \left(R_t \mathbb{1}_{A_t = a} + (N_{t+1}(a) - \mathbb{1}_{A_t = a}) Q_t(a) \right)$$

$$= \frac{1}{N_{t+1}(a)} \left(R_t \mathbb{1}_{A_t = a} + N_{t+1}(a) Q_t(a) - Q_t(a) \mathbb{1}_{A_t = a} \right)$$

$$= Q_t(a) + \frac{1}{N_{t+1}(a)} \left(R_t \mathbb{1}_{A_t = a} - Q_t(a) \mathbb{1}_{A_t = a} \right)$$

$$= Q_t(a) + \frac{\mathbb{1}_{A_t = a}}{N_t(a) + \mathbb{1}_{A_t = a}} \left(R_t - Q_t(a) \right)$$

Follow the question such that action a occurs, we have

$$Q_{t+1}(a) = Q_t(a) + \frac{1}{N_{t+1}(a)} \left(R_t - Q_t(a) \right)$$

where

$$N_{t+1}(a) = N_t(a) + \mathbb{1}_{A_t = a} = N_t(a) + 1.$$ □

c)

Given

$$Q_{n+1} = Q_n + \alpha_n (R_n - Q_n)$$

$$= \alpha_n R_n + (1 - \alpha_n) Q_n$$

$$= \alpha_n R_n + (1 - \alpha_n) [\alpha_{n-1} R_{n-1} + (1 - \alpha_{n-1}) Q_{n-1}]$$

$$= \alpha_n R_n + \alpha_{n-1} R_{n-1} (1 - \alpha_n) + Q_{n-1} (1 - \alpha_n)(1 - \alpha_{n-1})$$

$$= \cdots$$

$$= \alpha_n R_n + \sum_{i=1}^{n-1} \alpha_{n-i} R_{n-i} \prod_{j=0}^{i-1} (1 - \alpha_{n-j}) + Q_1 \prod_{i=1}^{n} (1 - \alpha_{n-j})$$

Check that as $\alpha_n \equiv \alpha$, $\forall n \in \mathbb{N}$. we have

$$Q_{n+1} = \alpha R_n + \sum_{i=1}^{n-1} \alpha R_{n-i} (1 - \alpha)^i + Q_1 (1 - \alpha)^n$$

$$= \sum_{i=0}^{n-1} \alpha R_{n-i} (1-\alpha)^i + Q_1 (1-\alpha)^n$$
$$= Q_1 (1-\alpha)^n + \sum_{i=1}^{n} \alpha R_i (1-\alpha)^{n-i}$$ □

d)

Denote that action set $\mathcal{A} = \{a_1, a_2\}$, then the soft-max is

$$\mathbb{P}(A_t = a_1) = \frac{e^{H_t(a_1)}}{e^{H_t(a_1)} + e^{H_t(a_2)}} = \frac{1}{1 + \exp\{H_t(a_2) - H_t(a_1)\}}$$
$$\mathbb{P}(A_t = a_2) = \frac{e^{H_t(a_2)}}{e^{H_t(a_1)} + e^{H_t(a_2)}} = \frac{1}{1 + \exp\{H_t(a_1) - H_t(a_2)\}}$$

where as the sigmoid fne on $\mathcal{A}$, although with a little modification, is

$$\text{sigmoid} \circ f(a) = \frac{1}{1 + \exp\{H_t(a_1) + H_t(a_2) - 2H_t(a)\}}$$

where $f(a) = 2H_t(a) - H_t(a_1) - H_t(a_2)$. So that

$$\text{sigmoid} \circ f(a_1) = \frac{1}{1 + \exp\{H_t(a_2) - H_t(a_1)\}}$$
$$\text{sigmoid} \circ f(a_2) = \frac{1}{1 + \exp\{H_t(a_1) - H_t(a_2)\}}$$

as required. □