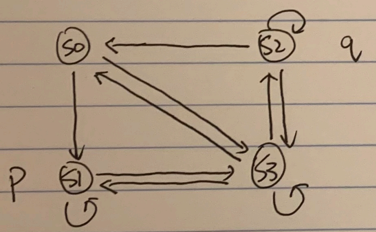
1. (a) $EX(p) = \{s_1, s_2\}$ (e) $AG(p) = \{\ \}$ (h) $AG(EX(p)) = \{\ \}$
(b) $EX(q) = \{s_1, s_2\}$ (f) $EG(p) = \{s_2\}$ (i) $AG(AF(p)) = \{\ \}$
(c) $AX(p) = \{s_1\}$ (g) $AF(p) = \{s_2\}$
(d) $AX(q) = \{\ \}$

2. (a) $AG(\text{invalid login} \rightarrow A\, \neg\, \text{withdraw-money})$

(b) $A(\text{open}\ U\ \text{request-to-move})$

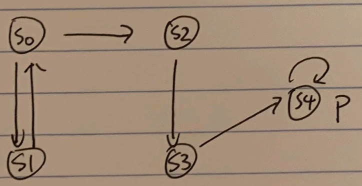
(c) $AG(\text{request-site-update} \rightarrow (\neg AG\ \text{updating-site}) \vee (EX\ \text{updating-site}) \vee$
$(EF\ \text{updating-site}) \rightarrow (EX\ \text{update-complete}) \vee (EF\ \text{update-complete})$

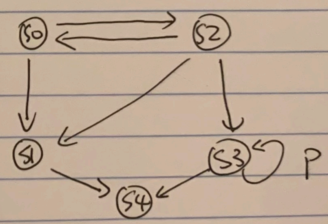
3. Disprove $AG(p) \Rightarrow AF(q)$ and $AG(p \Rightarrow AF(q))$



In this kripke structure $AG(p \Rightarrow AF(q))$ is satisfied but $AG(p) \Rightarrow AF(q)$ not. So, they are not equivalent

4. Prove ~~and~~ or disprove $EF(AG(p))$ is stronger than $EF(EG(p))$





In this case, $EF(AG(p))$ is stronger

In this case, $EF(EG(p))$ is stronger

5. (a) $A(p \circ \text{false})$: it's easy to implement "all state in the path satisfy $\varphi 1$ and not $\varphi 2$, so, it holds at least one.

(b) $A(\text{false} \circ p)$: it also can be expresed in CTL. start state is P, because $\varphi 2$ appears in first state. $A(\varphi_1 \circ \varphi_2)$ holds true.