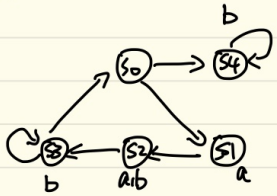
1. a) $AG(p \rightarrow (EX(q) \vee (EX(q) \wedge EXEX(q)) \vee (EX(q) \wedge EXEX(q) \wedge EXEXEX(q)) \vee (EX(q) \wedge EXEX(q) \wedge EXEXEX(q) \wedge EXEXEXEX(q))))$

b) $EF(AF(p) \wedge AF(q) \wedge AG(p \rightarrow \neg q \wedge q \rightarrow \neg p))$

2. a) $T = \{(\neg a \wedge \neg b \wedge a' \wedge \neg b') \vee (\neg a \wedge \neg b \wedge \neg a' \wedge b') \vee (\neg a \wedge b \wedge \neg a' \wedge b') \vee (a \wedge \neg b \wedge a' \wedge b') \vee (a \wedge b \wedge \neg a' \wedge b') \vee (\neg a \wedge b \wedge \neg a' \wedge b') \vee (\neg a \wedge b \wedge \neg a' \wedge \neg b')\}$




$A_b(true) = [[b]] \cap R_V(R_V(true)) = [[b]]$

$A_\varphi(z) = [[\varphi]] \cap R_V(R_V(z))$

$A_b^2(true) = [[b]] \cap R_V(R_V(b))$

$= b \wedge R_V(\forall a'b'(T \wedge b'))$

$= b \wedge R_V((\neg a \wedge \neg b \wedge \neg a' \wedge b') \vee (\neg a \wedge b \wedge \neg a' \wedge b') \vee (a \wedge \neg b \wedge a' \wedge b') \vee (a \wedge b \wedge \neg a' \wedge b'))$

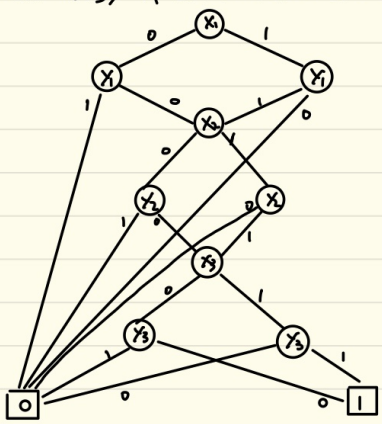
$= (\neg a \wedge b \wedge \neg a' \wedge b') \vee (a \wedge b \wedge \neg a' \wedge b')$

$A_b^3(true) = b \wedge R_V^2(\forall a'b'(T \wedge ((\neg a \wedge b \wedge \neg a' \wedge b') \vee (a \wedge b \wedge \neg a' \wedge b'))))$

$= b \wedge ((\neg a \wedge \neg b \wedge \neg a' \wedge b') \vee (a \wedge b \wedge \neg a' \wedge b'))$

$= (\neg a \wedge b \wedge \neg a' \wedge b') \vee (a \wedge b \wedge \neg a' \wedge b')$

3. $B = ((x_1 \wedge y_1) \vee (\neg x_1 \wedge \neg y_1)) \wedge ((x_2 \wedge y_2) \vee (\neg x_2 \wedge \neg y_2)) \wedge ((x_3 \wedge y_3) \vee (\neg x_3 \wedge \neg y_3))$




4. a)

Firstly, check if node $n_1$ and $n_2$ are both true or false, if not then return true. If they are same, check if their left children are both true or not. if not then check if their right children are both true or false. If their right children are not same, return true.

So, wit function is to check two nodes and their children are same things.

if structure $n_1$ is the copy of $n_2$ $\Rightarrow$ false

otherwise, true.

b) Runtime: $O(n)$