3.16) b) - Let $L_1$ be Turing recognizable
- Let $L_2$ be turing reccognizable

- Let $M_1$ be ~~decide~~ recognize $L_1$
- Let $M_2$ ~~decide~~ reccognize $L_2$
- Let $M$ recognize $L_1 \circ L_2$

1) $w$ is a ~~string~~ word from $L_1 \circ L_2$
2) divide each string of $L_1 \circ L_2$ into $w_1$ and $w_2$ nondeterministicly
3) run $w_1$ on $L_1$
   - If halting, then reject
4) run $w_2$ on $L_2$
   - If halting then reject
   - If accept, then accept

Therefore Turing recognizable languages are closed under concatenation

3.16) c) Let $L_1$ be a Turing Recognizable language

Let $L_1^*$ be obtained by *

Let $M_1$ be a machine that recognizes $L_1^*$

1) For an input string $\omega$ divide it into n parts:
   $\omega_1 \omega_2 \dots \omega_n$

2) For each part run $M_1$ on all divided parts.
   - If all parts parts are accepted, then accept
   - ~~If~~ Else, then reject

~~Therefore the star is clo~~

Therefore Turing recognizable languages are closed under *.

3.16) a)

- Let $L_1$ and $L_2$ be Turing recognizable
- Let $M_1$ and $M_2$ recognize $L_1$ and $L_2$

1) Assume s is an input string of $L_1 \cup L_2$

2) run s on $M_1$ :
   - If it accepts runs s on $L_2$
   - Else rejet

3) If $M_2$ accepts $L_2$ then accept
   - Else then reject

Turing recognizable languages are closed under intersection.