4.1 a) Yes ; M accepts 0100

b) No ; M doesnt accept 011

c) No ; only has a single component

d) No ; not a regular expression

e) No ; language isnt empty

f) Yes ; accept same language as itself

4.2 $L = \langle R, S\rangle \mid R$ is deterministic FA

- Assume T is a Turing machine that decides L
- F = " $L = \langle R, S \rangle$ on input

- Convert R to a DFA
- Operate TM as F on input $\langle R_S, D_S \rangle$
- If F accept, accept L
- If F reject, reject L

4.3) $ALL_{DFA} = \{ \langle A \rangle | A$ is a DFA and $L(A) = \Sigma^* \}$

- A is a DFA that accepts every possible permutation and combination

- Mark initial state A
- Repeat until no new state is marked
- The state that has coming into it will be marked
- Accept when all states are marked
  - ~~a~~ Else, reject

4.4) Proof by constraction

- A_{ECFG} is decidable
- G derives $T\langle (G) \rangle$
- G does not derive $T \langle (G) \rangle$

- $S \to \varepsilon$
- If GFG derives ε G derives $L(g) = L(G')$
- G' Does not contain rule $S \to \varepsilon$
- Convert G in ~~to~~ CFG G'
- ↑ G' contains $S \to \varepsilon$ then accept
  - otherwise Reject