1. For the following FSM, $M$, document its parts by listing each of the following:
   a) alphabet b) states c) starting state d) accepting states e) transitions.
   List three strings (from $\Sigma^*$) that $M$ accepts.
   List three strings (from $\Sigma^*$) that $M$ does not accept.

   ![Finite State Machine Diagram]

2. Design an FSM to recognise each of the following languages:
   a) Words using only the letters \{a, b, c\} that contain no c.
   b) Words using only the letters \{a, b\} such that the last two symbols are the same.
   c) Words using only the letters \{a, b\} of the form $(ab)^n$ for $n > 0$ (that is ab repeated $n$ times).
   d) Words using only the letters \{a, b\} that contain exactly 2 bs.

3. Define a divisible by 3 checker as a machine that accepts a string from the alphabet 0, 1 if and only if the sum of its digits is divisible by 3. Design an FSA to implement a divisible by 3 checker.
4. Design a *nondeterministic* FSM (NFSM) to recognise any binary string that ends in 01. Hint: you only need 3 states.

5. Convert your NFSM from the previous question into a deterministic FSM for the same language.

6. Devise an FSM to solve the farmer, wolf, goat and cabbage problem and find the solutions. This problem involves a farmer, a wolf, a goat and a cabbage all on one side of a river. There is a boat but the farmer can carry only one passenger at a time. The farmer wants to get them all to the other side of the river. However, left alone, the wolf will eat the goat, and the goat will eat the cabbage. How do they all get to the other side?

7. *Source: Sipser 1.12*

   Let $D = \{ w \mid w$ contains an even number of $a$s and an odd number of $b$s and does not contain the substring $ab \}$

   Give a FSM with 5 states that recognises $D$.

   Hint: describe $D$ more simply.