The aim of today's work is to see:

(a) For any finite set of words, there is a finite state machine which accepts exactly that set of words and rejects all other words.

(b) There is a set of words (the set is infinite) and no finite state machine exists which accepts exactly the given set of words and rejects all other words.

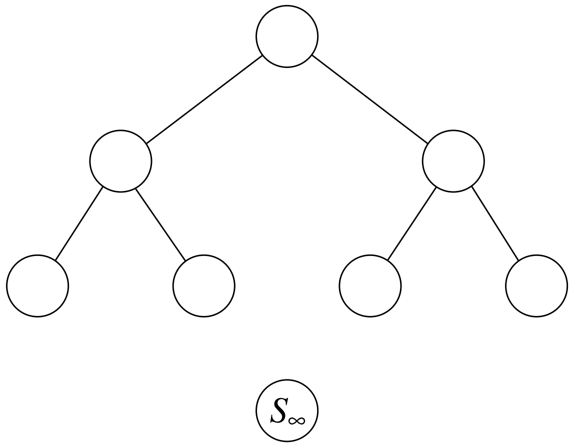
Creating a finite state machine for the set of words: θ, 1, 01.

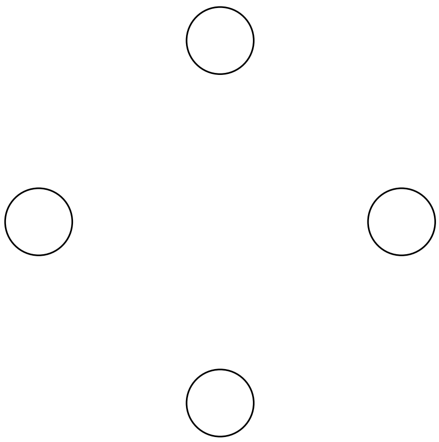
On the right is a complete binary tree of height 2.

(i) Label each left edge with 0 and each right edge with 1.

(ii) Label the root S_θ and label every other vertex via the path which leads to it from the root.

(iii) Add arrows, extra edges, an initial state and accepting states so as to yield the state diagram of an FSM which accepts the words θ, 1 and 01 and rejects all other words.

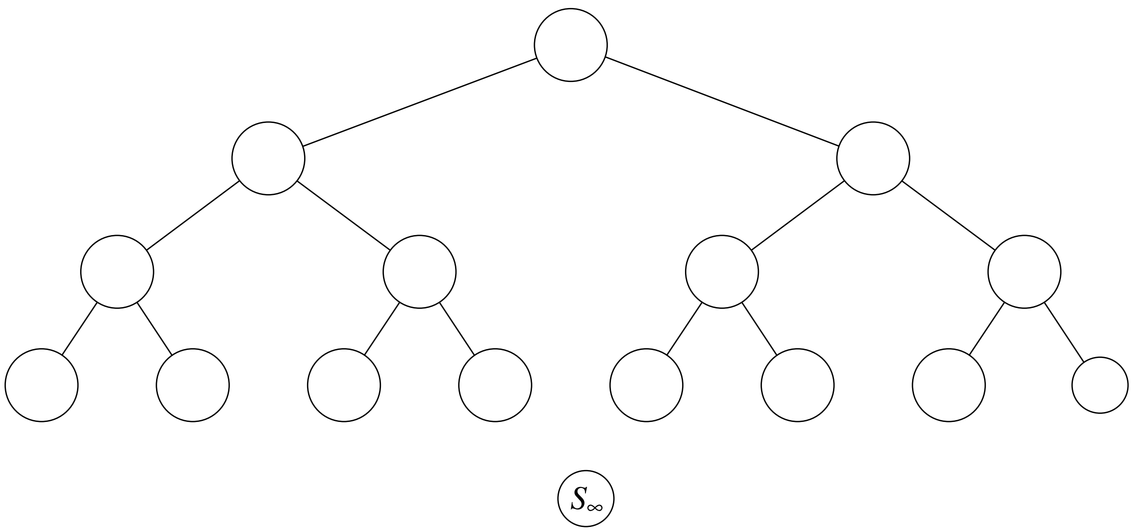




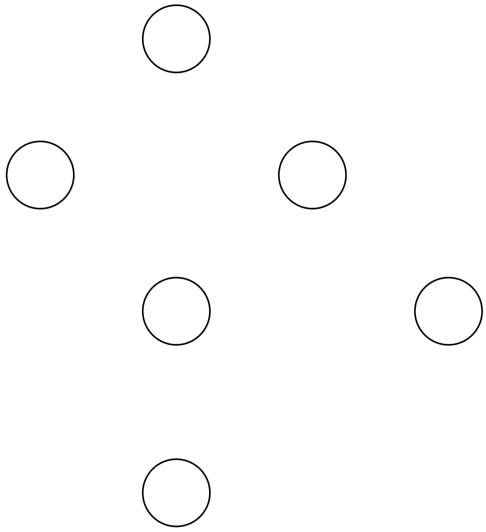
(iv) Now try to create an equivalent machine with only 4 states. The idea is to use the least possible number of states. This will mean using only a single dump state.

Two machines are called **equivalent** if they accept precisely the same words.

(v) Use the complete binary tree below to produce the state diagram of an FSM which accepts the words: 1, 11, 000, 100 and no others.



(vi) Try to produce an equivalent machine with only 6 states. (For the language 1, 11, 000, 100.)



A general method of creating the required FSM for any given finite set of words is as follows:

Let $F = \{w_1, \ldots, w_k\}$ be a finite set of binary words. Assume that the longest word in F has length n. The following steps will produce the state diagram of the required machine:

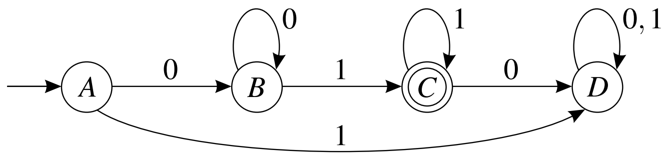
(1) Use a complete binary tree of height n. Put an arrow on each edge pointing away from the root. Label each left arrow 0 and each right arrow 1.

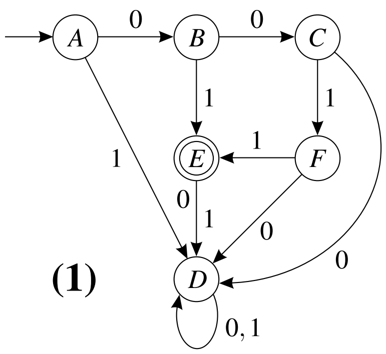
(2) Label a vertex S_w if the unique path from the root to the vertex is given by the binary word w.

(3) Make the root the initial state and make S_w an accepting state if w is in F.

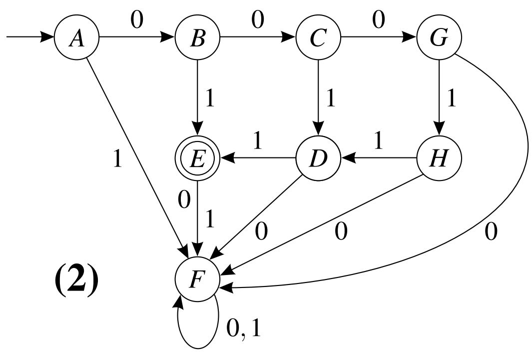
(4) Add a black hole and add an edge labelled 0,1 from each leaf of the binary tree to the black hole.

On the right is the state graph of an FSM whose language is $\{(0)^m(1)^n \mid m, n \geqslant 1\}$. [This can be done with 4 states, but can't be done with fewer.]





(1)



(2)

Describe the language of FSM **(1)**, FSM **(2)**and FSM **(3)**

Language of **(1)**

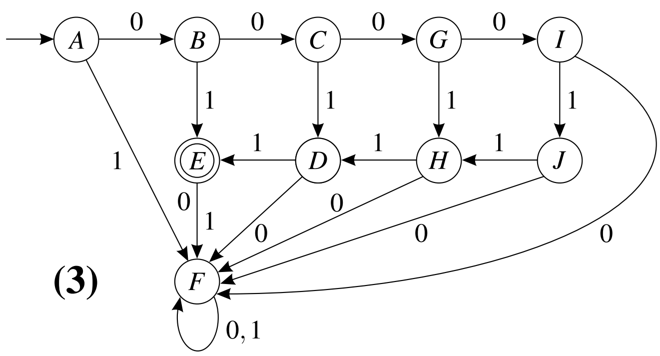
....

Language of **(2)**

....

Language of **(3)**

....

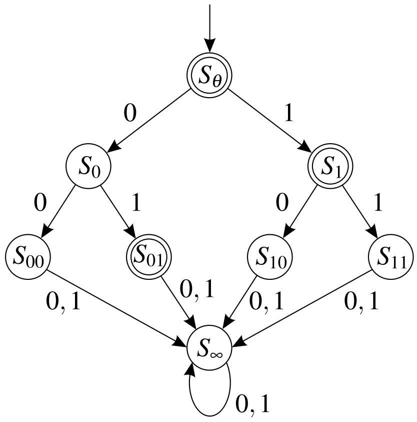


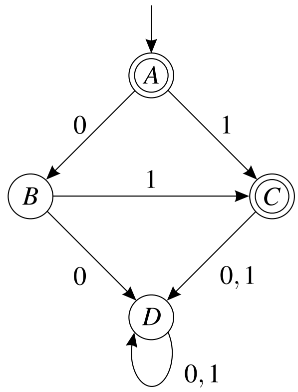
(3)

(vii) Can you see why an FSM which accepts the word 0000011111 but does not accept $(0)^m(1)^n$ for $m \neq n$ must have at least 6 states.

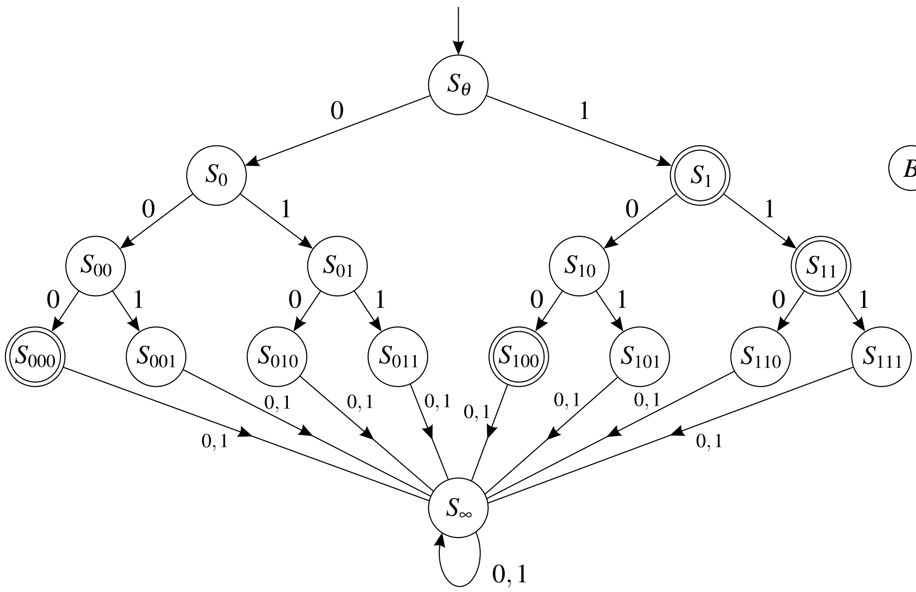
(viii) Can you see why there is no FSM whose language is $\{(0)^m(1)^m \mid m \geqslant 1\}$.

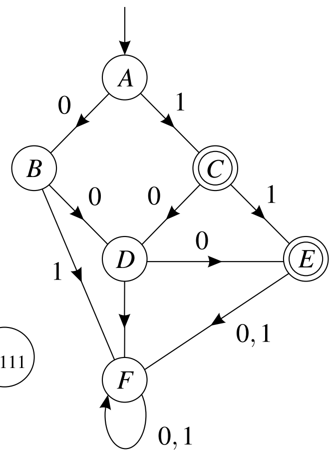
Here are the first two completed Finite State Machines:





Here are the next two completed Finite State Machines:





Language of **(1)**: 01, 0011

Language of **(2)**: 01, 0011, 000111

Language of **(3)**: 01, 0011, 000111, 00001111