5CCS2FC2: Foundations of Computing II

Tutorial Sheet 1

1.1 Over the alphabet $\Sigma = \{0, 1\}$, construct a Finite Automata (deterministic or non-deterministic) that accepts each of the following languages:

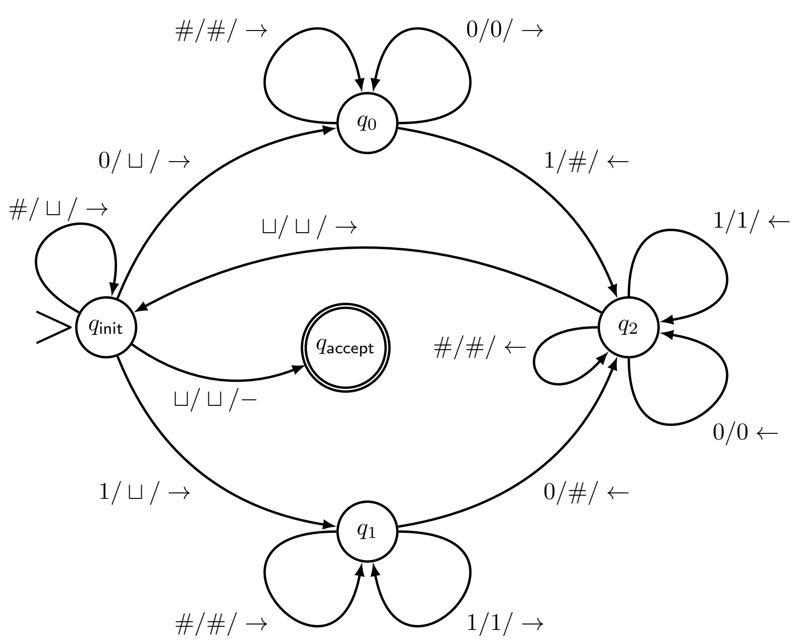
(i) The language of all binary strings that begin and end with a 1,

(ii) The language of all strings that contain exactly three 1s

(iii) The language of all strings that contain the substring 1010,

(iv) The regular language represented by the expression $1(01)^*1$,

1.2 Let $\Sigma = \{0, 1, \#\}$, and consider the following Turing Machine $\mathcal{T}$ on the language, whose transition diagram is depicted below:



(Code for `turingmachinesimulator.com` is available on KEATS.)

(i) Which of the following words are accepted by this Turing Machine:

- 1001,
- 00,
- 101,
- 0011.

(ii) What is the language that is accepted by this Turing Machine? *i.e.*, what is $L(\mathcal{T})$?

1.3 Consider the language of all *palindromes* over the binary alphabet $\Sigma = \{0, 1\}$,

$$L \;=\; \left\{w \in \{0,1\}^* \;:\; w = w^R\right\}$$

where w^R denote the reversal of $w \in \{0,1\}^*$ (*e.g.*, $(111001)^R = 100111$)

(i) Give 5 examples of words belonging to L,

(ii) Outline a pseudo-code program for a Turing machine that accepts the language L,

(iii) Convert your pseudo-code into a complete description of a Turing Machine. (Test that your machine works as intended using the `turingmachinesimulator.com`.)

1.4 Consider the language

$$L \;=\; \{w\#w \;:\; w \in \{0,1\}^*\}$$

comprising all those strings over the alphabet $\Sigma = \{0, 1, \#\}$, that consist of two copies of a binary string separated by a special character $\#$ that appears precisely once. Using the pigeon-hole principle, show that no DFA can accept the language L.