A linear bounded automaton is a multi-track non-deterministic Turing machine with a tape of some bounded finite length.

**Length** = function \( \text{LengthOfTheInitialInputString, constant} \)

Here,

**Memory information \( \leq c \times \text{Input information} \)**

The computation is restricted to the constant bounded area. The input alphabet contains two special symbols which serve as left end markers and right end markers which mean the transitions neither move to the left of the left end marker nor to the right of the right end marker of the tape.

A linear bounded automaton can be defined as an 8-tuple \((Q, X, \Sigma, q_0, M_L, M_R, \delta, F)\) where –

- \(Q\) is a finite set of states
- \(X\) is the tape alphabet
- \(\Sigma\) is the input alphabet
- \(q_0\) is the initial state
- \(M_L\) is the left end marker
- \(M_R\) is the right end marker where \(M_R \neq M_L\)
- \(\delta\) is a transition function which maps each pair \(\text{state, tapesymbol}\) to \((\text{state, tape symbol, Constant 'c')}\) where \(c\) can be 0 or +1 or -1
- \(F\) is the set of final states

A deterministic linear bounded automaton is always context-sensitive and the linear bounded automaton with empty language is undecidable.