

Ex 1

I will present my TM through both a high and low level description

Firstly, the higher level description, through which I'm going to offer a clear explanation of computation for my TM and a formal description of computation.

My TM decides $A = \{ (ab)^i \mid i \geq 0 \}$ the language consisting of all strings that have 'a' on odd positions and 'b' on even positions.

TM = "On input string w

1. Locate the head of the first tape on the first character of the string, and the head of the second tape on the second character of the string
2. Sweep left to right across both tapes, moving in both cases a step equal to 2.
3. If the head of the first tape lands on a different character than 'a' or the head of the second tape lands on a different character than 'b', reject
4. If we find a blank space character in the second tape, accept."

Ex 1

My TM computes as follows:

- Initially, TM receives its input w on the leftmost squares of the tapes, and the rest of the tapes are blank (filled with blank symbols)
- The first head starts on the leftmost square of the first tape, while the second head starts on the second leftmost square of the second tape. Note that Σ does not contain the blank space character, so the first blank appearing on the tapes marks the end of the input.
- Once TM has started, the computation proceeds according to the rules described by the transition functions
- If TM tries to move its head to the left off the left-hand of the tapes, the head stays in the same place for that move, even though the transition function indicates L
- The computation continues until it enters either the accept or reject states, at which point it halts.
~~It neither accepts nor rejects, it goes on forever.~~
- As TM computes, changes occur in the current state and the current head location

Ex 1

Obs:

- By using a step=2 and having offsetted heads, we traverse the whole string without missing characters.
- The TM stops checking the string after finding a blank space character in the second tape because we can safely assume that the blank space character is at the end of the string as our language A doesn't contain any blank space characters.
- The first delta doesn't have an accept state because the A language can't end with an 'a'.

Wan

Ex 1

Now, for the lower level description, I'll give the formal definition for my two-tape T.M.

$$TM = (Q_1, Q_2, \delta_1, \delta_2, q_1, q_2, q_a, q_r, \Sigma, \Gamma)$$

sets of states

$$Q_1 = \{q_1, q_3, q_4\}$$

$$Q_2 = \{q_2, q_5, q_a, q_r\}$$

$q_1, q_2 = \text{start state}$

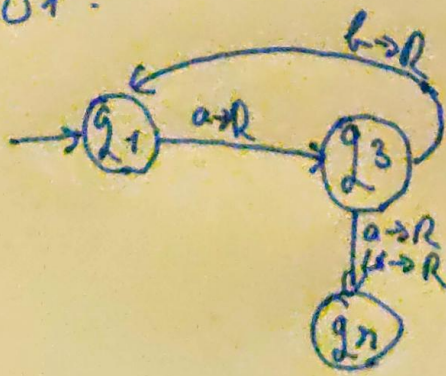
$$\Sigma = \{a, b\}$$

$q_a = \text{accept state}$

$$\Gamma = \{a, b, \sqcup\}$$

$q_r = \text{reject state}$

δ_1 :



δ_2 :

