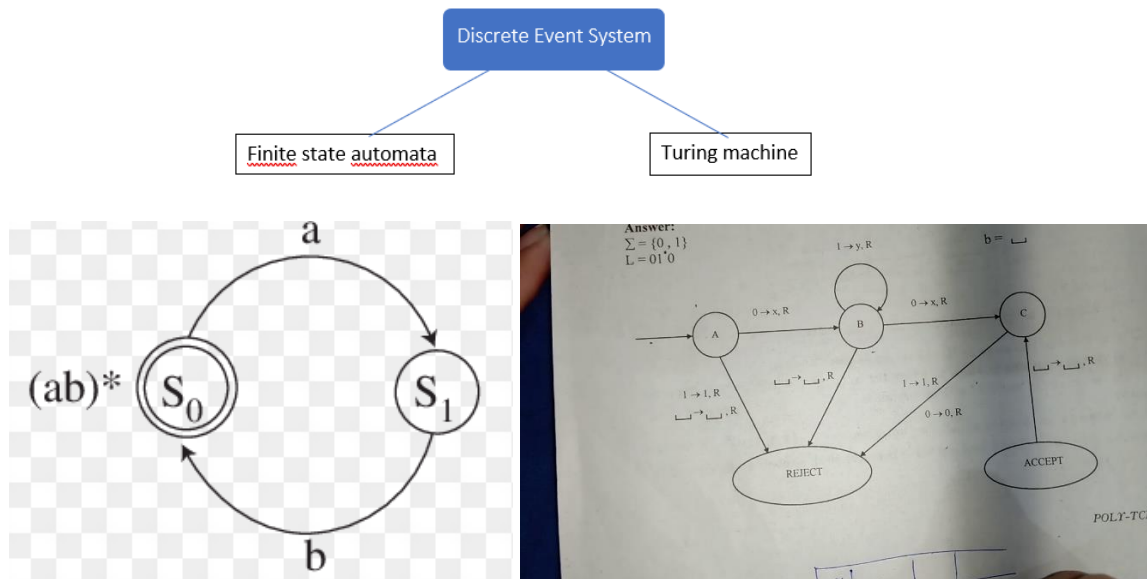


Theorievragen

1.



De taal $(ab)^*$ past bij deze machine omdat de machine altijd moeten eindige op state S_0 . En de taal bestaat uit alleen maar uit substring (ab) waardoor de machine altijd eindigt op S_0

De taal (01^*0) past bij deze machine omdat het eerste getal 0 moet zijn bij state A waarna die naar rechts gaat en state B. State B komt terug naar zichzelf als er een 1 staat en een naar rechts. Als er een 0 staat gaat die naar state C en een naar rechts waarna er geen getal meer mag staan. De taal klopt dus omdat de taal altijd begint met een 0 en altijd eindigt met een 0

Bronnen: canvas video,

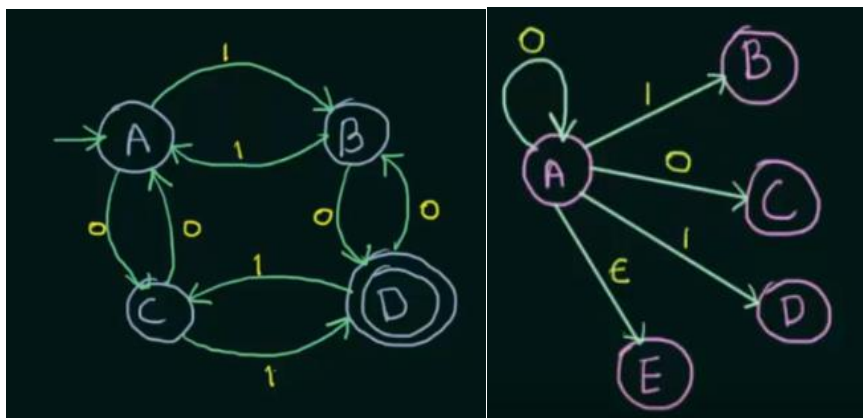
plaatjes van het internet

2. deterministisch automaton

- geven de momentele een state, weet je altijd wat de volgende state zal zijn.
- Het heeft altijd maar een unieke volgende state.
- Geen random keuzes.

Deterministisch

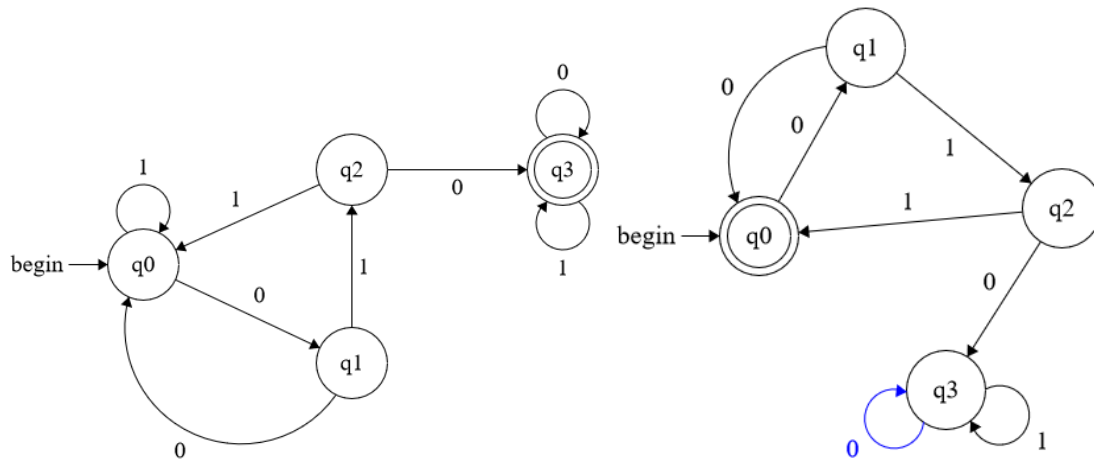
niet-deterministisch



Je ziet dat bij de deterministisch geen random keuzes worden gemaakt en je weet altijd wat de volgende state word omdat er alleen maar unieke states zijn. Bij de niet deterministische zie je dat er bij state A met 1 naar state B of naar state D. dus kan deze niet deterministisch zijn.

automata

1.

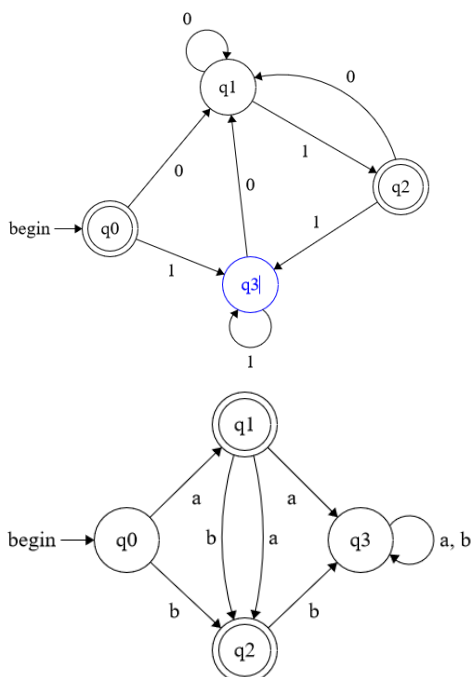


2. Automata 1: alfabet $\{a, b\}$. moet beginnen en eindigen met een a. en mag nooit 2x dezelfde getal achter elkaar hebben.

Automata 2: alfabet $\{a, b\}$. de taal moet alleen een b zijn of a^* plus een b.

3. Taal: een lege string of moet eindige met sub string (0 1).

(En soms de string (1) en de string(1 1)) maar deze tellen niet voor de deterministische automaton.



4.

4. De machine gaat naar de rechter kant van de string en verandert alle 1en naar een 0 tot de machine een 0 of een _ tegen komt en schrijft een 1. Wat dit doet in het binaire stelsel is 1 bij een getal optellen.

Aangepaste machine: in plaats van 1 optellen gaan we nu 1 aftrekken. Als het binaire getal 0 is word het binaire getal -1 gemaakt door aan het einde nog een 1 te schrijven.

Taal: $\{0, 1, _ \}$

