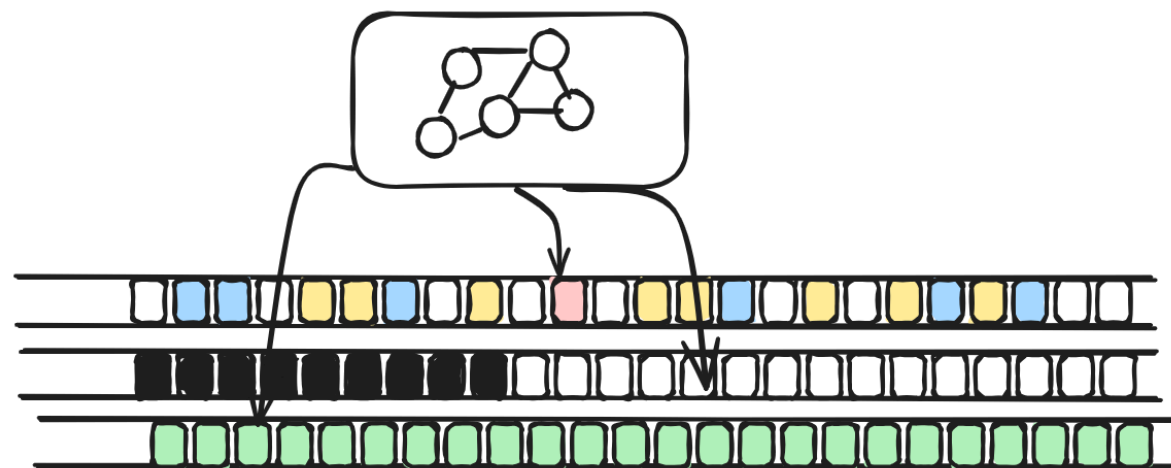


# Turingovi stroji

# Uroš Čibej



# Pregled

- Osnovni model Turingovega stroja
- Razširitve
  - večtračni
  - nedeterministični
- Ekvivalence teh modelov
- Church-Turingova teza

# Literatura

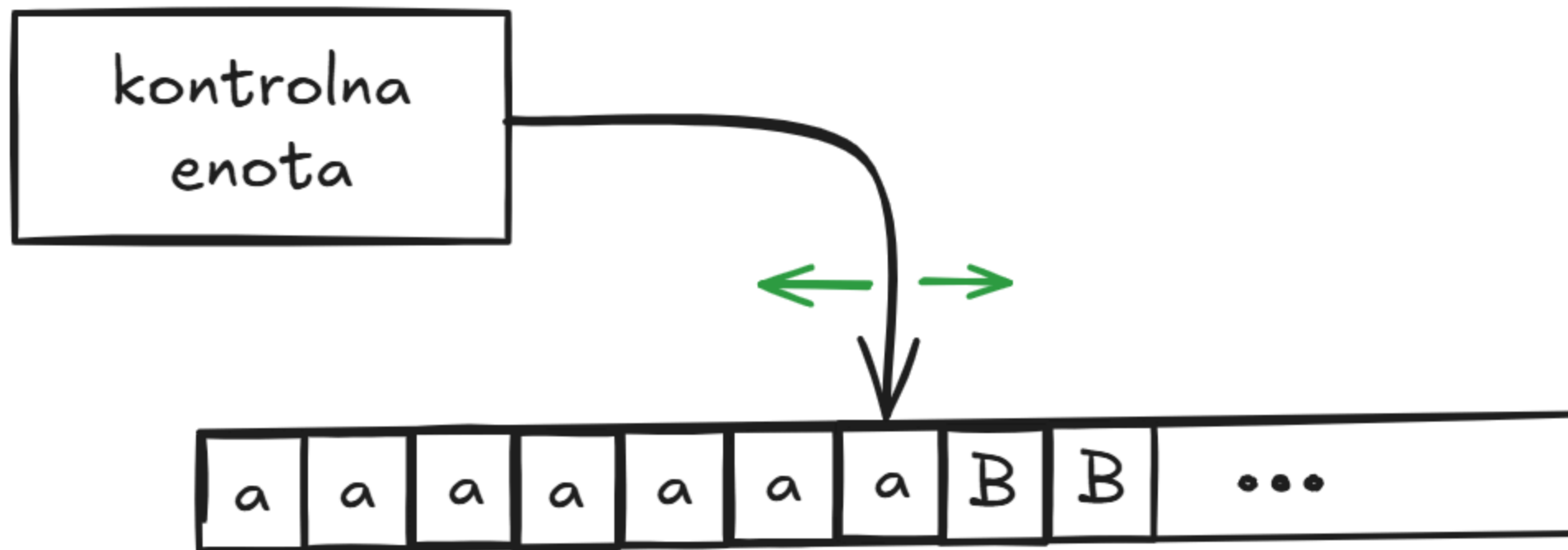
- Sipser poglavje 3.
- **dodatno:** [https://introtcs.org/public/lec\\_06\\_loops.html](https://introtcs.org/public/lec_06_loops.html)

# Ekspres zgodovina

- **David Hilbert** (1910) - potreba po formalni definiciji algoritma
- **Alan Turing**: On Computable Numbers, with an Application to the Entscheidungsproblem (1936)
- **Alonzo Church**: An Unsolvable Problem of Elementary Number Theory (1936)



## Shematski prikaz TS



# Formalna definicija TS

$$M = \langle Q, \Sigma, \Gamma, \delta, q_0, q_F, q_R \rangle$$

- $\Sigma \subseteq \Gamma$
- $B \notin \Sigma$  (blank) je privzeti simbol na traku
- $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$
- $q_F$  je stanje sprejetja, **stroj se takoj ustavi**
- $q_R$  je stanje zavrnitve, **stroj se takoj ustavi**

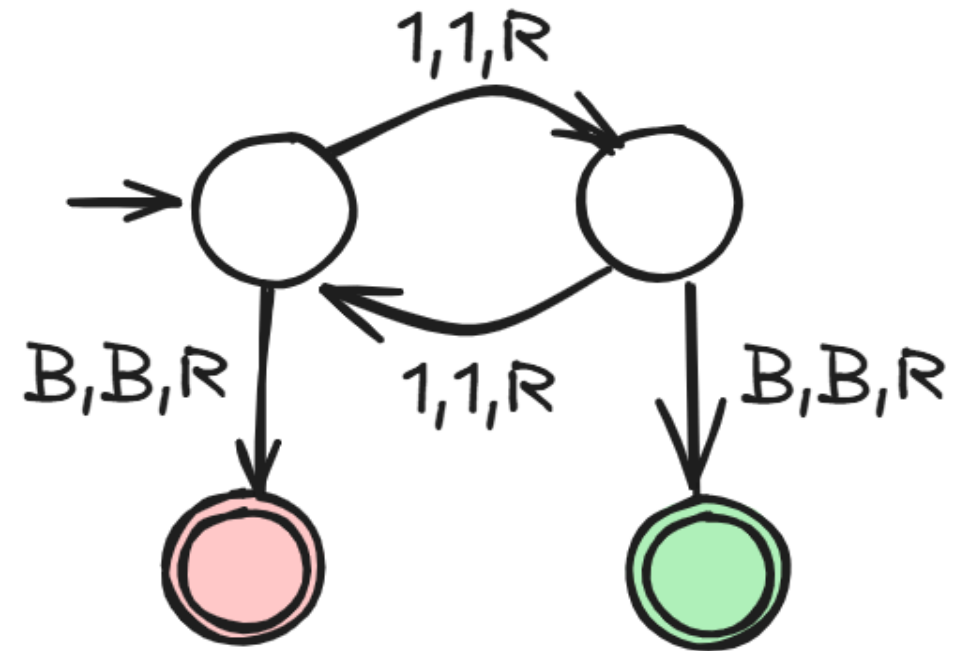
## Podajanje TS

$$\delta(q_0, 1) = (q_1, 1, R)$$

$$\delta(q_1, 1) = (q_0, 1, R)$$

$$\delta(q_0, B) = (q_R, B, R)$$

$$\delta(q_1, B) = (q_A, B, R)$$



# Primer reševanja problema s TS

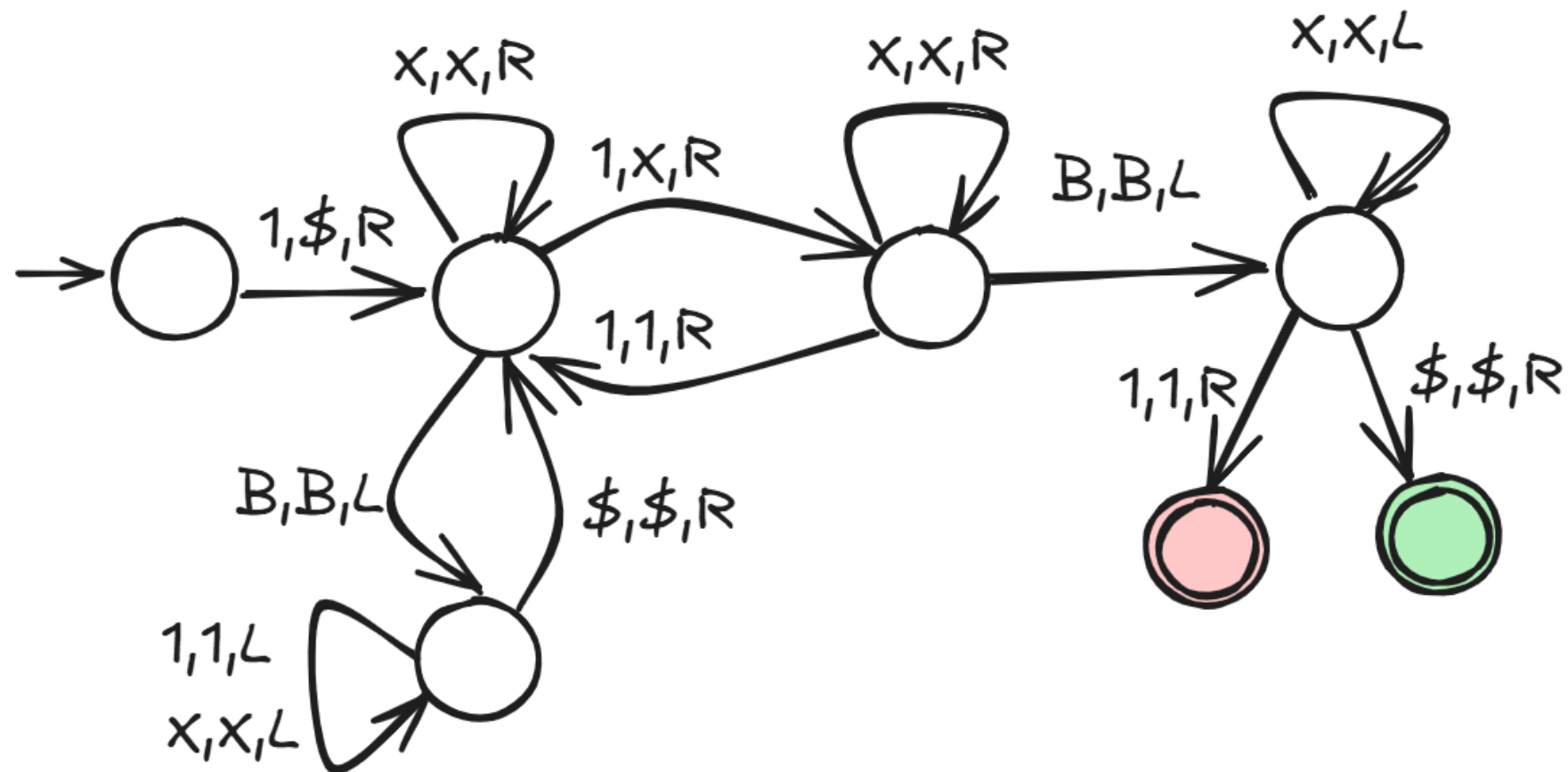
$$L = \{a^n b^n\}$$



# Številski problem

$$L = \{a^{2^n}\}$$

# Rešitev



# Trenutni opisi (konfiguracije) TS

$$\alpha q a \beta$$

- $\alpha$  - vsebina traku levo od glave
- $q$  trenutno stanje
- $a$  simbol trenutno pod glavo
- $\beta$  - vsebina traku desno od glave

## Prehodi med konfiguracijami

$$\alpha q a \beta \vdash \alpha b r \beta$$

Če obstaja  $\delta(q, a) = (r, b, R)$

(podobno, če bi imeli premik levo)

# Ključne konfiguracije

## Začetna

$q_0w$

## Ustavitveni konfiguraciji

1.  $\alpha q_F \beta$  - besedo sprejmemo
2.  $\alpha q_R \beta$  - besedo zavrnemo

# Jezik Turingovega stroja

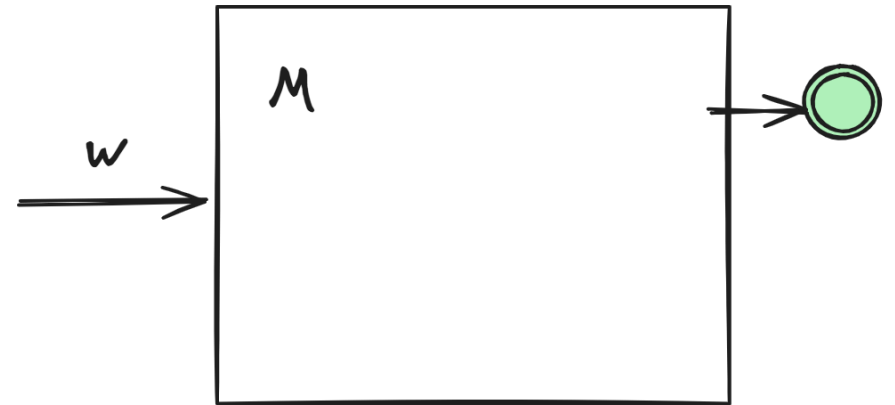
$$L = \{w \in \Sigma^* \mid q_0 w \vdash^* \alpha q_F \beta\}$$

# Polodločljivost jezika

Def Jezik  $L$  je polodločljiv, če

$$\exists M, L(M) = L, \forall w \in L : q_0 w \vdash^* \alpha q_F \beta$$

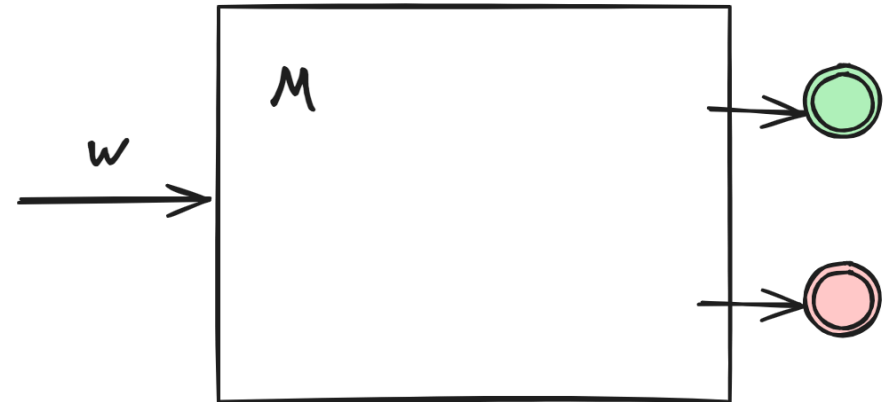
(zanima nas samo obnašanje za  
 $w \in L$ )



# Odločljivost jezika

Def Jezik  $L$  je odločljiv, če

$$\exists M, L(M) = L, \forall w \in L : q_0 w \stackrel{*}{\vdash} \alpha q_F \beta$$



# Razširitve in ekvivalence modelov

- Neskončen trak v obe smeri
- Več trakov
- Nedeterminizem



# Neskončnost traku v obe smeri

## Primer

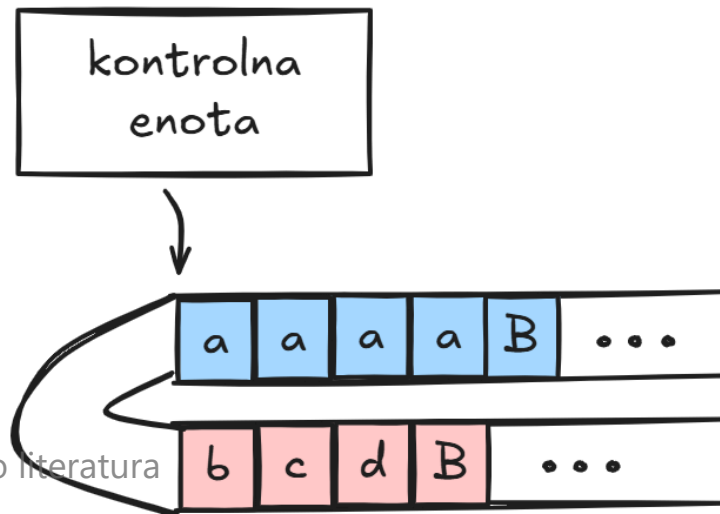
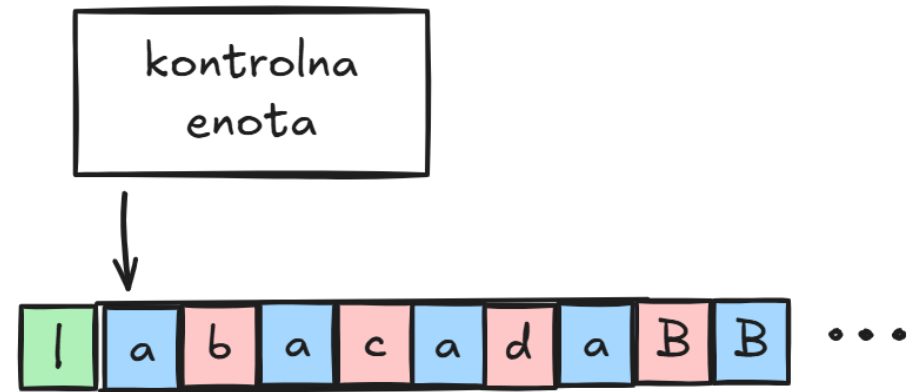
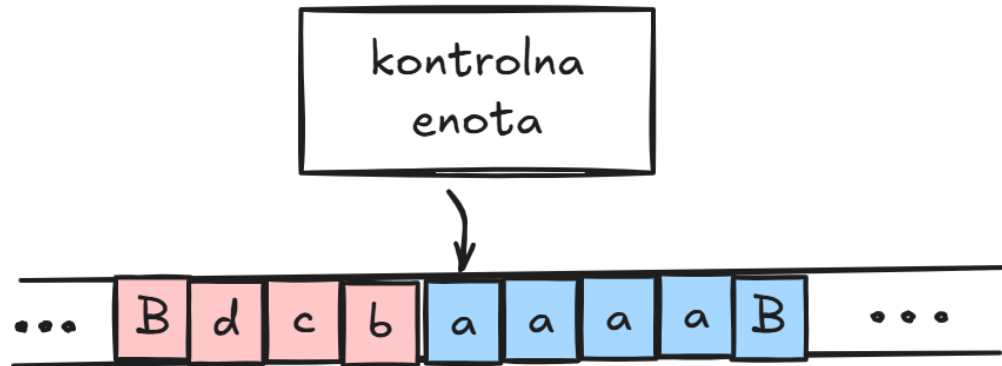
$$\delta(q_0, B) = (q_1, X, L)$$

$$\delta(q_1, B) = (q_0, X, R)$$

$$\delta(q_0, X) = (q_0, X, R)$$

$$\delta(q_1, X) = (q_1, X, L)$$

# Simulacija z enosmerno neskončnim trakom



## Večtračnost (definicija)

$k$ -tračni Turingov stroj  $M = \langle Q, \Sigma, \Gamma, \delta, q_0, q_F, q_R \rangle$

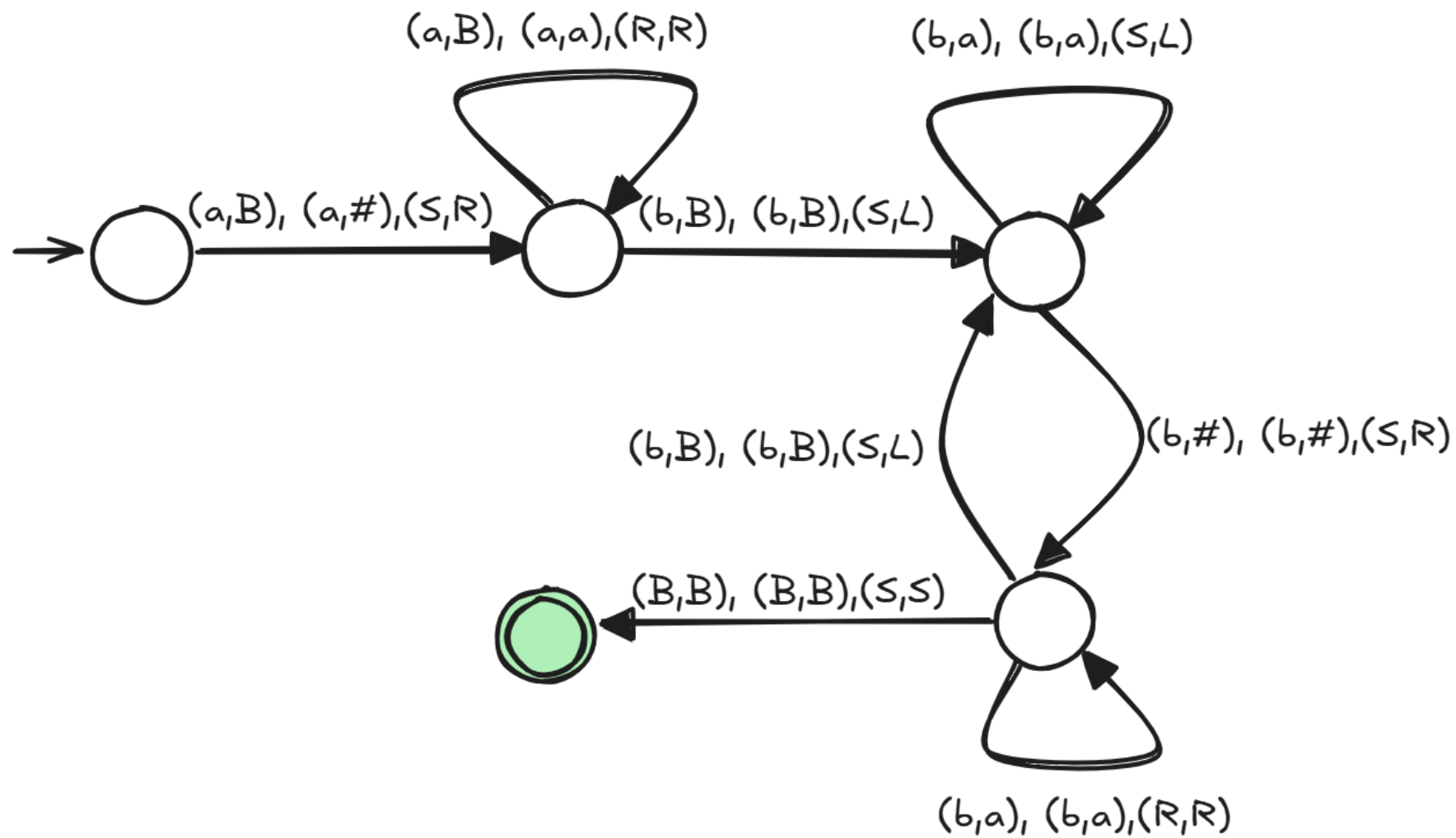
$$\delta : Q \times \Gamma^k \rightarrow Q \times \Gamma^k \times \{L, R, S\}^k$$

## Večtračnost (primer)

$$L = \{a^k b^m \mid k \text{ deli } m\}$$

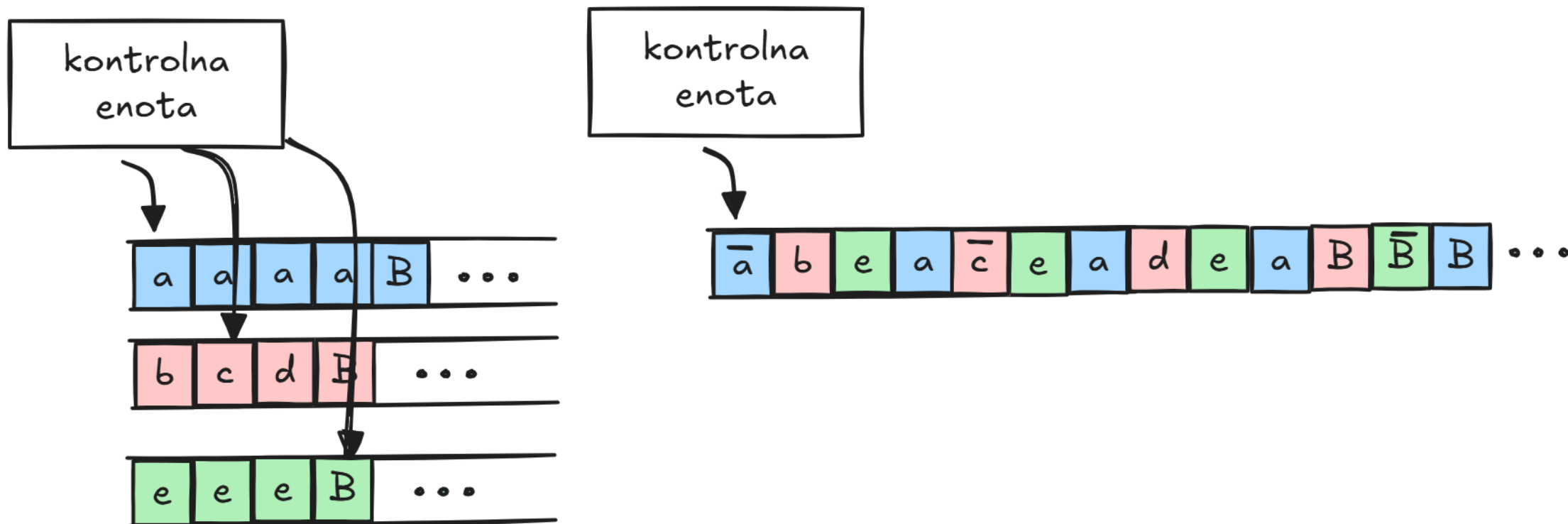
Ideja: dvotračni TS

- a-je prepišemo na drugi trak
- sprehajamo se hkrati čez a-je in čez b-je (drugi trak prevrtimo velikokrat nazaj)
- če na obeh trakovih **hkrati** pridemo na B,  $k$  deli  $m$



**Izrek** Vsak večtračni Turingov stroj ima ekvivalenten enotračni Turingov stroj.

# Simulacija z enim trakom



# Opis simulacije

En korak večtračnega stroja

1. poiščemo glavo (in znak) na vsakem traku (zapomnimo si s stanjem katere znake smo videli)
2. glede na te simbole naredimo ustrezne zapise na posamezen trak in ustrezne premike glave



## Nedeterminizem (definicija)

$$M = \langle Q, \Sigma, \Gamma, \delta, q_0, q_F, q_R \rangle$$

$$\delta : Q \times \Gamma \rightarrow 2^{Q \times \Gamma}$$

# Nedeterminizem (primer)

$$L = \{a^k \mid k \text{ je sestavljeno število} \}$$

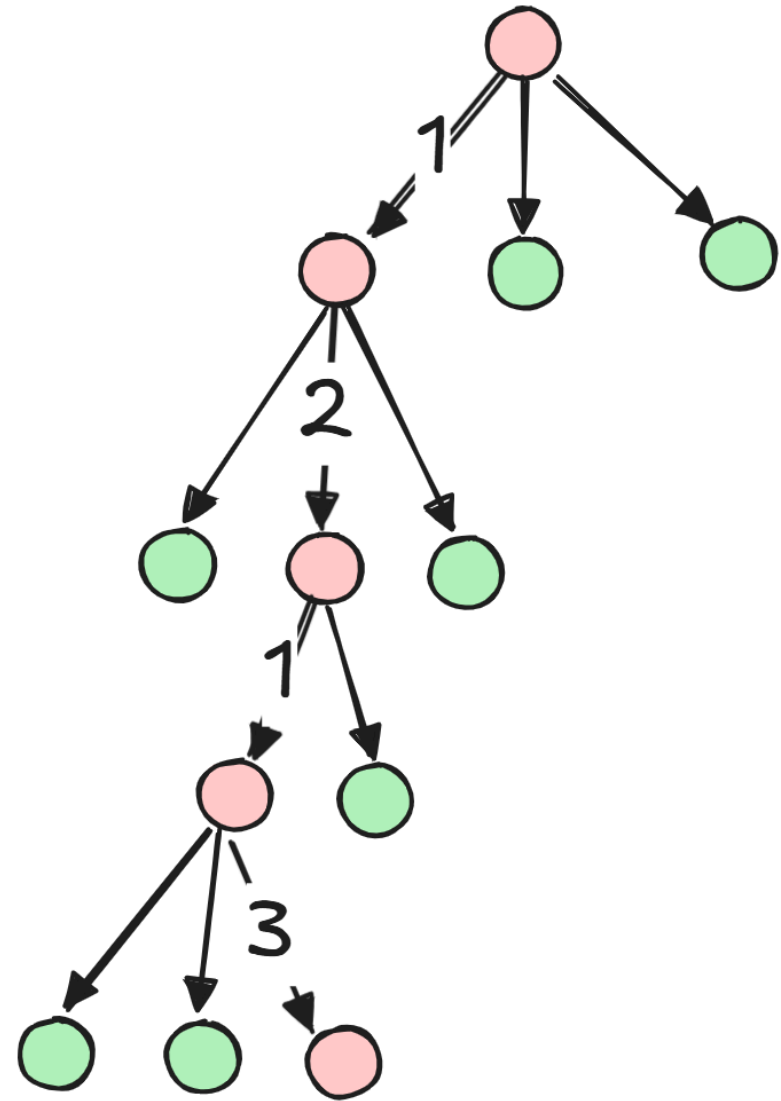
# Ekvivalenca determinizma in nedeterminizma

**Izrek:** Za vsak nedeterministični TS  $M$  obstaja determinističen TS  $M'$ , da velja  $L(M) = L(M')$

# Simulacija z determinizmom

1. Sled izvajanja nedeterminističnega stroja je drevo (potencialno neskončno)
2. Z determinizmom lahko **sistematično** preiskujemo to drevo (iščemo konfiguracijo, ki niz sprejme)

1. V drevesu izvajanja se osredinimo na eno pot
2. Pot je določena z zaporedjem števil, ki predstavljajo odločitev ob vsaki vejitvi v drevesu



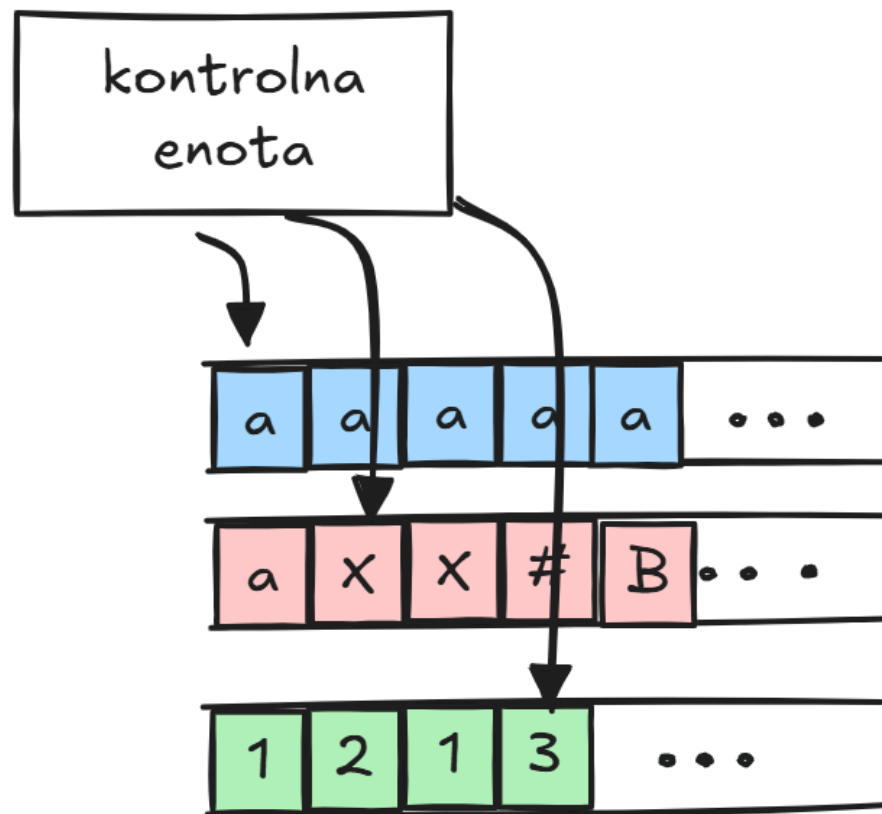
**Nedeterministično:**

$$\delta(q_0, a) = \{(q_0, a, R), (q_1, X, L)\}$$

**Deterministično:**

$$\delta(q_0, a, 1) = (q_0, a, 1, R, R)$$

$$\delta(q_0, a, 2) = (q_1, X, 2, L, R)$$



vhod  
simulacija  
vodilo

# Opis simulacije

1. Prvi trak vsebuje vhod  $w$  (trak vedno samo beremo)
2. Prekopiramo vsebino traku 1 na trak 2
3. Simuliramo stroj na drugem traku, za vsako nedeterministično odločitev uporabimo tretji trak, da postane deterministična
4. Ko zmanjka simbolov na 3. traku povečamo vsebino 3. traku za 1 in gremo na točko 2.
5. Če je kadarkoli na traku konfiguracija, ki vhod sprejme, besedo sprejmemo



## Church-Turingova teza

TS  $\longleftrightarrow$  Algoritem

# Zakaj zaupamo Church-Turingovi tezi?

- V teh modelih znamo izraziti vse kar intuitivno razumemo kot algoritem
- **OGROMNO** zelo različnih modelov računanja je ekvivalentnih TS
- teoretični rezultati, dobljeni iz teh modelov dobro odražajo praktične situacije