

Motivace: sečítat velmi dlouhá čísla napsaná na velmi dlouhé zdi. Jak budete postupovat?

Kódování vstupu:

abeceda ... množina znaků

Př:  $\Sigma_1 = \{0, 1\}$  ... binární abeceda

$\Sigma_2 = \{a, b, c, d, \dots, z\}$  ... latinky

$\Sigma_3 = \{0, 1, 2, 3, \dots, 9\}$  ... dekadické číslice

$\Sigma_4 = \{0, 1, 2, \dots, 9, +, -, =, *, ', ', \{, \}, <, >, (, ), \}$

vstup ... řetězec znaků

$\Sigma^* = \{ \text{množina všech konečných řetězců nad abecedou } \Sigma \}$

Př:  $\Sigma^* = \{ \epsilon, 0, 1, 00, 01, 10, 11, 000, 001, \dots \}$   
 $\uparrow$   
 prázdné slovo

$\forall w \in \Sigma^* \quad |w| = \text{délka slova}$

zajímají nás

• funkce

$f: \Sigma^* \rightarrow \Sigma^*$

$f: \Sigma^* \rightarrow \{0, 1\}$

booleanští fce

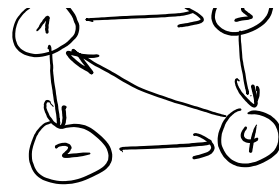
$L = \{ w \in \Sigma^* \mid F(w) = 1 \}$

$\uparrow$   
 jazyk  $L \subseteq \Sigma^*$

Př: graf  $G = (V, E)$   
 $\dots, \uparrow, \downarrow, \dots$



Př: graf  $G = (V, E)$   
 ↑                      ↑  
 vrchol                      hrany



matku sousednosti

$$A = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

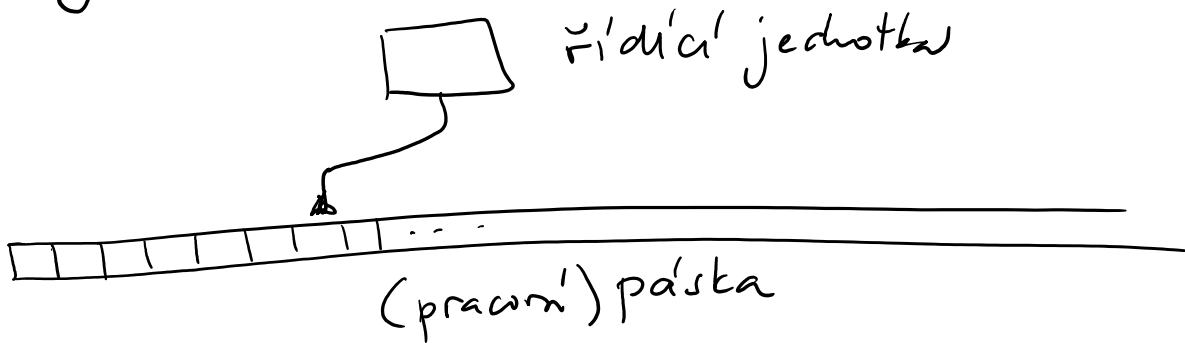
→ 0110 # 0001 # 0001 # 1000

nebo:  $(\{1, 2, 3, 4\}, \{(1, 2), (1, 3), (2, 4), (3, 4), (4, 1)\})$

• bitární abeceda postačuje pro všechno

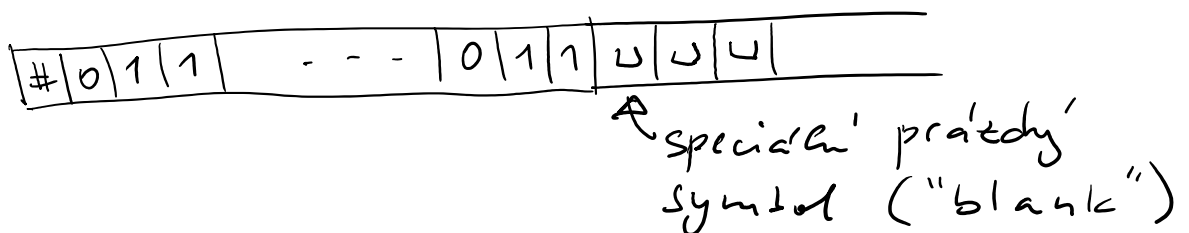
0 → 00                      00 11 11 00 10 00 00 ...  
 1 → 11                      0 1 1 0 # 0 0 :-  
 # → 10

## Thuringův stroj



• program zabudován v řídící jednotce

Př: zvyš číslo zapsaného na páse binárně 0 1.

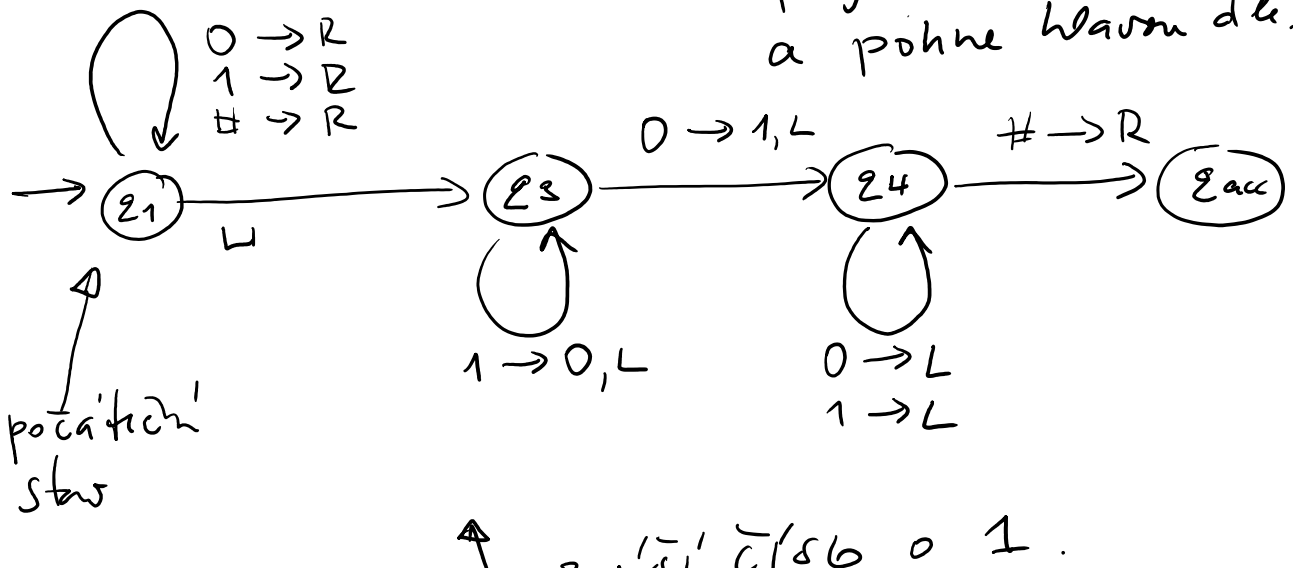


- alg:
1. jdi dokud nenarazíš na 'L'
  2. udělej krok zpět
  3. dokud vidíš 1, přepiš na nulu a jdi ulvo  
jakmile uvidíš 0, přepiš na jednu, jdi 4.
  4. jed' dleeva dokud nenarazíš na #.

## Formální definice Turingova stroje.

- $\Sigma$  ... vstupní abeceda, neobsahuje #,  $\sqcup$   
 $\Gamma$  ... pracovní abeceda,  $\Sigma \subseteq \Gamma$ , #,  $\sqcup \in \Gamma$ .  
 $Q$  ... množina stavů  $\approx$  řádky programu  
 $\delta: Q \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$  ... přechodová fce  
 $q_0$  ... počáteční stav  $q_0, q_{acc}, q_{rej} \in Q$   
 $q_{acc}$  ... přijímací stav } koncoví stavy  
 $q_{rej}$  ... odmítací stav }

$\delta(s, a) = (s', a', d)$  ... když TS je ve stavu  $s$  a vidí  $a$  přijde do stavu  $s'$ , zapíše  $a'$  a pohne hlavou dle  $d$ .



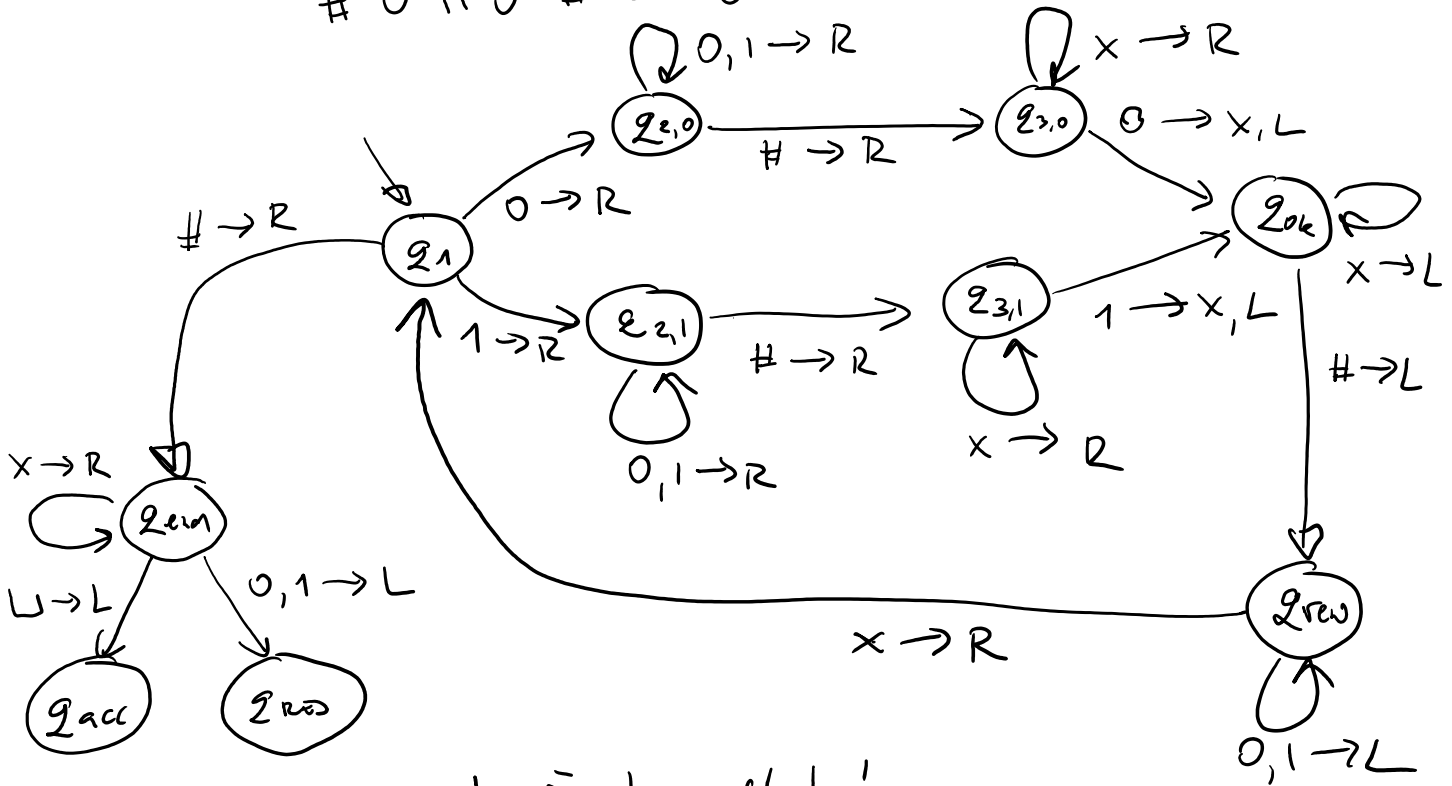
Star

↑  
zvyšší číslo o 1.

Př: Na vstupu dostanu řetězec 0 a 1 oddělený #  
a chci zjistit, zda jsou stejné

# 0 1 1 0 # 0 1 0 1  
# 0 1 1 0 # 0 1 1 0

≠  
=



+ všechny ostatní  
přechody veden do qrej

D.ú. - posunout pásmo o jedna doprava

#u → #0u

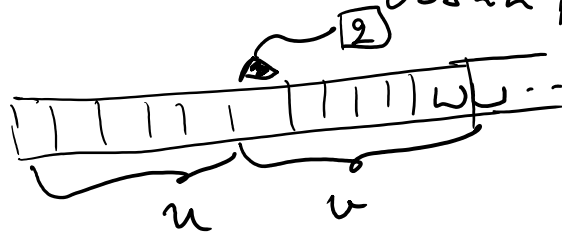
zkopírovat řetězec

#u → #u#u

průběh výpočtu lze zaznamenat pomocí konfigurací!

konfigurace :

- aktuální stav
- aktuální pozice hlavy
- obsah pásky



→  $uqv$

na páse je  $uv$   
jsem ve stavu  $q$   
hlava je na první  
znaku  $v$

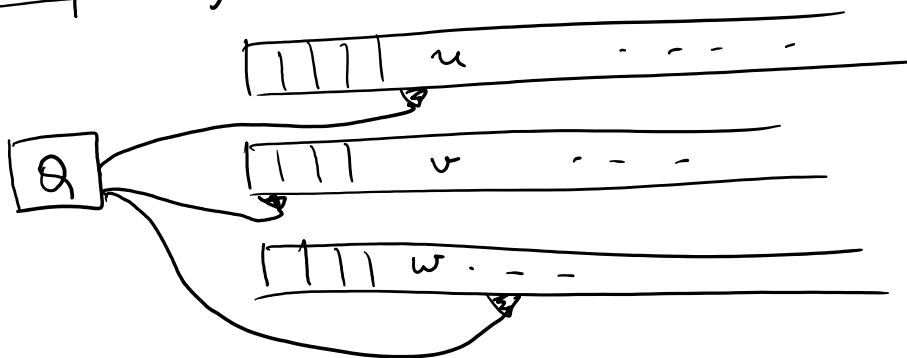
výpočet : postupnost konfigurací

$q_0 w \rightarrow \dots$   
↑  
vstup

→  $q_{acc} v$   
↑  
výstup

Varianty TS :

• vícepáskový



lze simulovat na jednopáskovém :

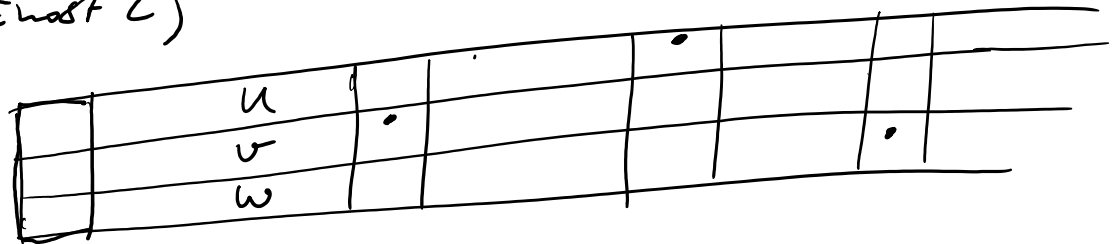
možnost 1) 

$u$		#		$v$		#		$w$		#		...
-----	--	---	--	-----	--	---	--	-----	--	---	--	-----

  
 tečka označuje symbol, kde je zrovna hlava

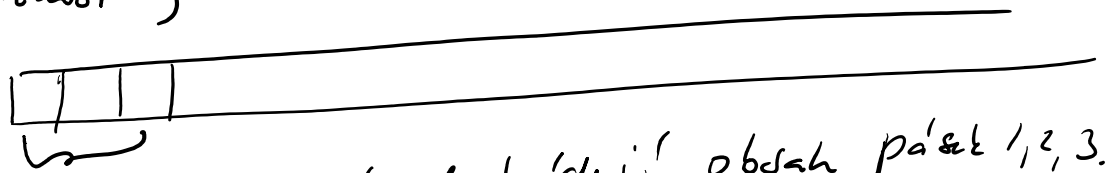
(pro každý znak  $a$  zavodu otevřený  $\underline{a}$   
 $\rightarrow$  součást pracovní abecedy)

- jeden krok  $M$  simulují tím, že si projdou páskou, zjistím, co vidí hlava  $M$ , rozhodnu se, co by udělal a to provedu (adsimulují možnost 2)



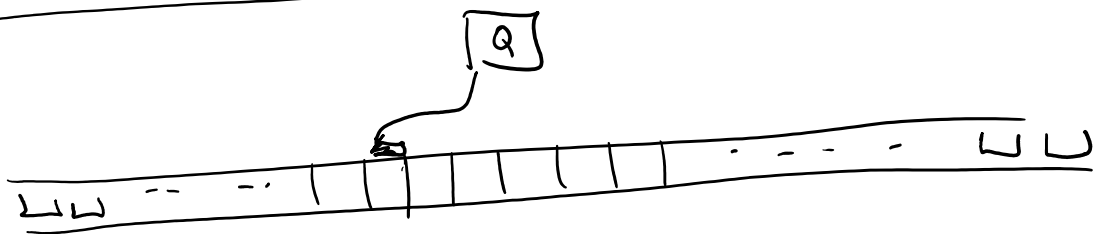
$\hookrightarrow$  jeden symbol kóduje tři symboly

možnost 3)



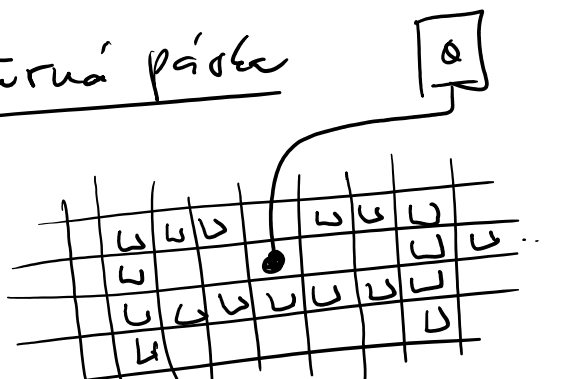
tři; po sobě jdoucí pole kódují obsah pásek 1, 2, 3.

- obousměrná nekonečná páska



- dvourozměrná páska

(Důl)





znaku. → automaty, ...)

• Simulace RAM



registry (A, B, C, D, E, F)	instrukce :	MOV	=
		ADD	+
		SUB	-
		MUL	·
		DIV	/
		JMP	J

Church - Turingova Thesis

• Turingovy stroje dokážou simulovat každý myslitelný algoritmus.

• TS  $M$  nad abecedou  $\Sigma$   
 •  $L(M) = \{ w \in \Sigma^* \mid M \text{ přijímá } w \}$   
 ... jazyk přijímaný TS  $M$ .

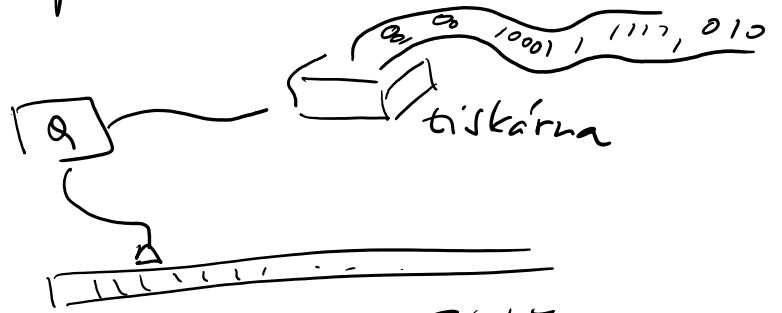
•  $M(w)$  ... výsledek výpočtu  $M$  nad  $w$   
 např.:  $M(w) = \text{ACCEPT} / 1$  přijímá  
 $M(w) = \text{REJECT} / 0$  nepřijímá  
 $M(w) = \uparrow$  nezastaví se na  $w$

•  $M$  rozhoduje  $L \subseteq \Sigma^* : M \text{ se zastaví na všech vstupích } \& L(M) = L$ .





- elektrický počítač



- tiskne ke tiskárně libovolné řetězce