Regular languages
DFAs/NFAs

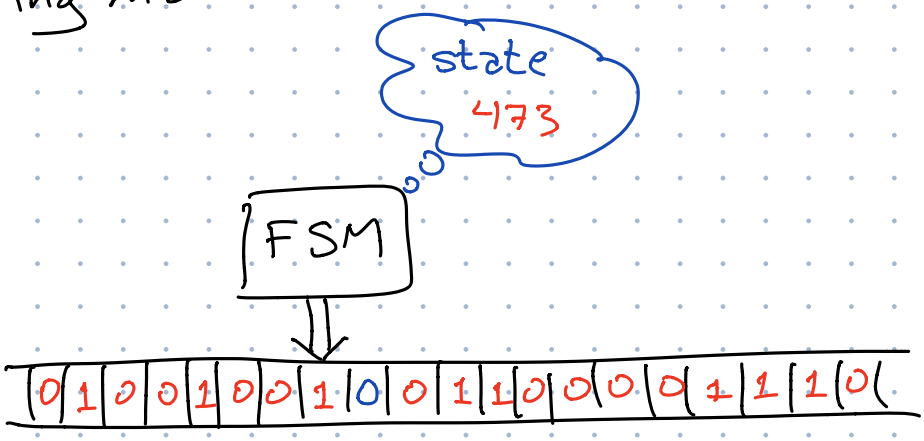
- single string
- concat/sequencing
- alternation/choice
- iteration/looping

Context-Free languages
CFG
Pushdown automata
Recursive automata

- recursion

Turing machines

- memory



$\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{\triangleleft, \triangleright\}$

Γ tape alphabet

Turing machine

Q — finite set of states

start — start state

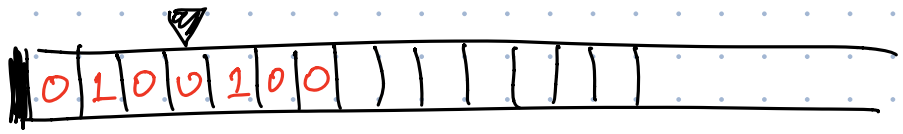
accept, reject > halt states

Σ — input alphabet $\{0,1\}$

Γ — tape alpabet $\Sigma \subseteq \Gamma$

$\square$ — blank $\in \Gamma$

$$\delta : (Q \setminus \{accept, reject\}) \times \Gamma \longrightarrow Q \times \Gamma \times \{-1, +1\}$$



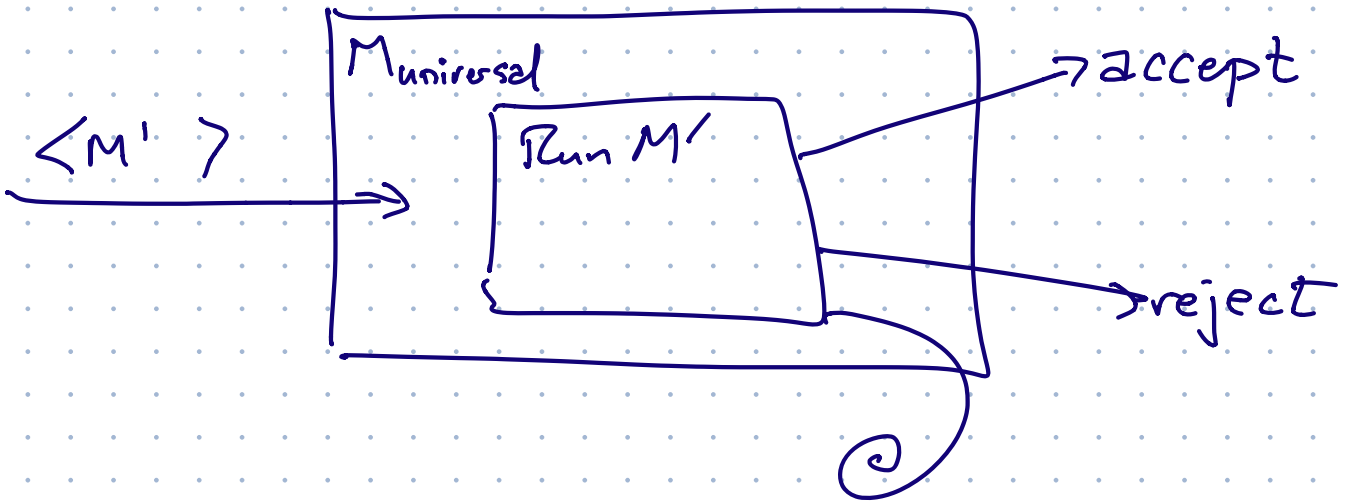
Entscheidungsproblem

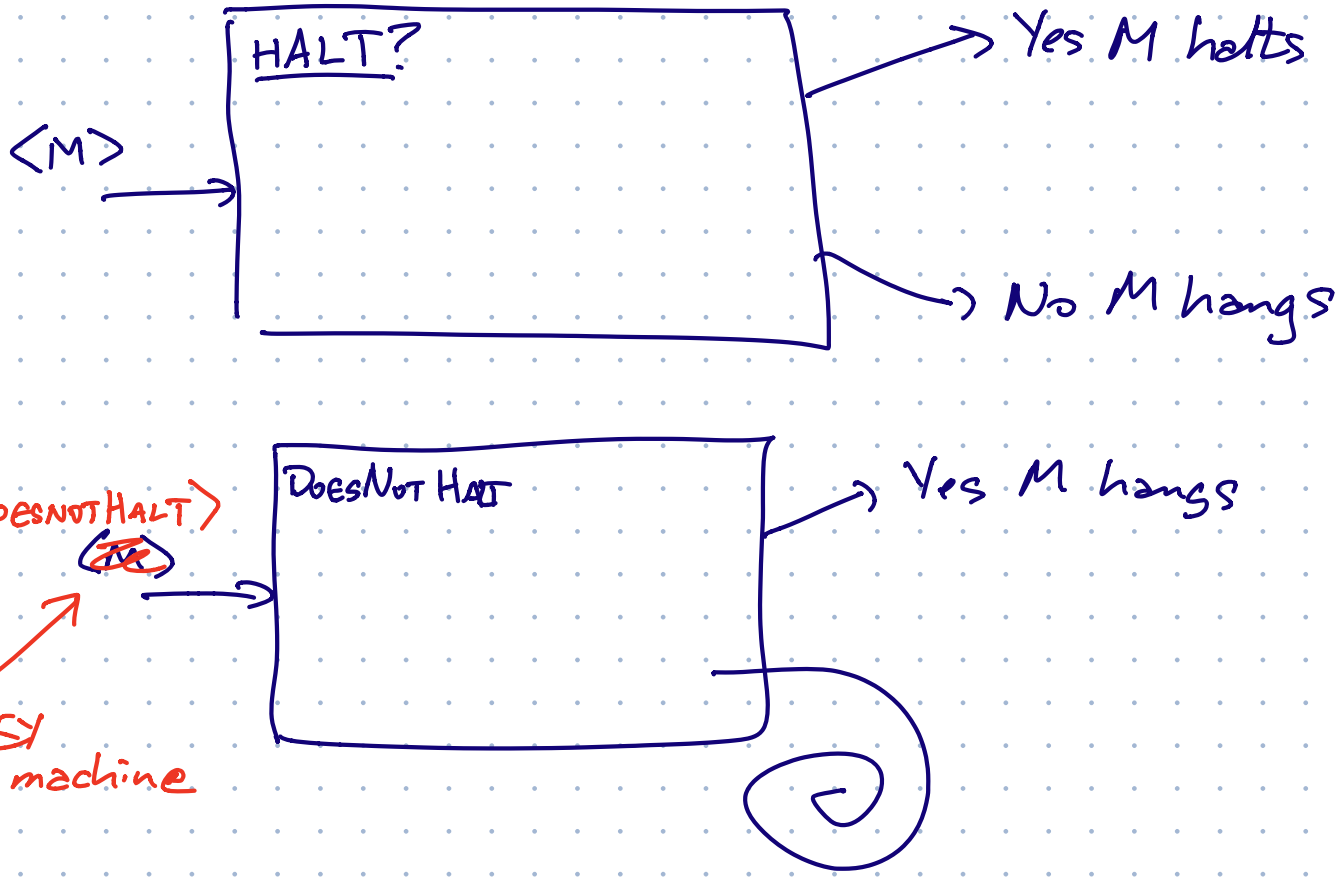
Gödel Church → Turing

λ-calculus

Turing machines have software

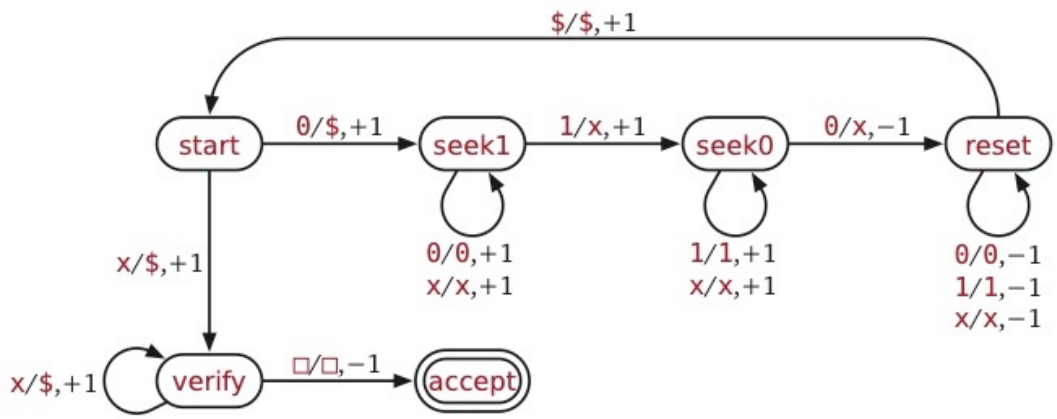
that can simulate other machines




HALT?
<M>
Yes M halts
No M hangs
<DoesNotHalt>
DoesNotHalt
Yes M hangs
arbitrary
Turing machine

$\delta(\ p\ , a) = (\ q\ , b, \delta\ )$	explanation
$\delta($ start , 0$) = ($ seek1 , $, +1)$	mark first 0 and scan right
$\delta($ start , x$) = ($ verify , $, +1)$	looks like we're done, but let's make sure
$\delta($seek1, 0$) = ($ seek1 , 0, +1$)$	scan rightward for 1
$\delta($seek1, x$) = ($ seek1 , x, +1$)$	
$\delta($seek1, 1$) = ($ seek0 , x, +1$)$	mark 1 and continue right
$\delta($seek0, 1$) = ($ seek0 , 1, +1$)$	scan rightward for 0
$\delta($seek0, x$) = ($ seek0 , x, +1$)$	
$\delta($seek0, 0$) = ($ reset , x, +1$)$	mark 0 and scan left
$\delta($ reset , 0$) = ($ reset , 0, −1$)$	scan leftward for $
$\delta($ reset , 1$) = ($ reset , 1, −1$)$	
$\delta($ reset , x$) = ($ reset , x, −1$)$	
$\delta($ reset , $$) = ($ start , $, +1$)$	step right and start over
$\delta($verify, x$) = ($ verify , $, +1$)$	scan right for any unmarked symbol
$\delta($verify, □$) = ($accept, □, −1$)$	success!

The transition function for a Turing machine that decides the language $\{0^n1^n0^n \mid n \geq 0\}$.



$\delta(p, a) = (q, b, \delta)$
$\delta($ start , 0$) = ($ seek1 , $, +1$)$
$\delta($ start , x$) = ($ verify , $, +1$)$
$\delta($seek1, 0$) = ($ seek1 , 0, +1$)$
$\delta($seek1, x$) = ($ seek1 , x, +1$)$
$\delta($seek1, 1$) = ($ seek0 , x, +1$)$
$\delta($seek0, 1$) = ($ seek0 , 1, +1$)$
$\delta($seek0, x$) = ($ seek0 , x, +1$)$
$\delta($seek0, 0$) = ($ reset , x, +1$)$
$\delta($ reset , 0$) = ($ reset , 0, −1$)$
$\delta($ reset , 1$) = ($ reset , 1, −1$)$
$\delta($ reset , x$) = ($ reset , x, −1$)$
$\delta($ reset , $$) = (start , $, +1$)$
$\delta($verify, x$) = ($ verify , $, +1$)$
$\delta($verify, □$) = ($accept, □, −1$)$

(start, 001100)
⇒ (seek1, $01100)
⇒ (seek1, $01100)
⇒ (seek0, $0x100)
⇒ (seek0, $0x100)
⇒ (reset, $0x1x0)
⇒ (reset, $0x1x0)
⇒ (reset, $0x1x0)
⇒ (reset, $0x1x0)
⇒ (start, $0x1x0)
⇒ (seek1, $$x1x0)
⇒ (seek1, $$x1x0)
⇒ (seek0, $$xxx0)
⇒ (seek0, $$xxx0)
⇒ (reset, $$xxxx)
⇒ (reset, $$xxxx)
⇒ (reset, $$xxxx)
⇒ (reset, $$xxxx)
⇒ (start, $$xxxx)
⇒ (verify, $$$xxx)
⇒ (verify, $$$$xx)
⇒ (verify, $$$$$x)
⇒ (verify, $$$$$$□)
⇒ (accept, $$$$$$) ⇒ **accept!**

(start, 00100)
⇒ (seek1, $0100)
⇒ (seek1, $0100)
⇒ (seek0, $0x00)
⇒ (reset, $0xx0)
⇒ (reset, $0xx0)
⇒ (reset, $0xx0)
⇒ (start, $0xx0)
⇒ (seek1, $$xx0)
⇒ (seek1, $$xx0)
⇒ (seek1, $$xx0) ⇒ **reject!**