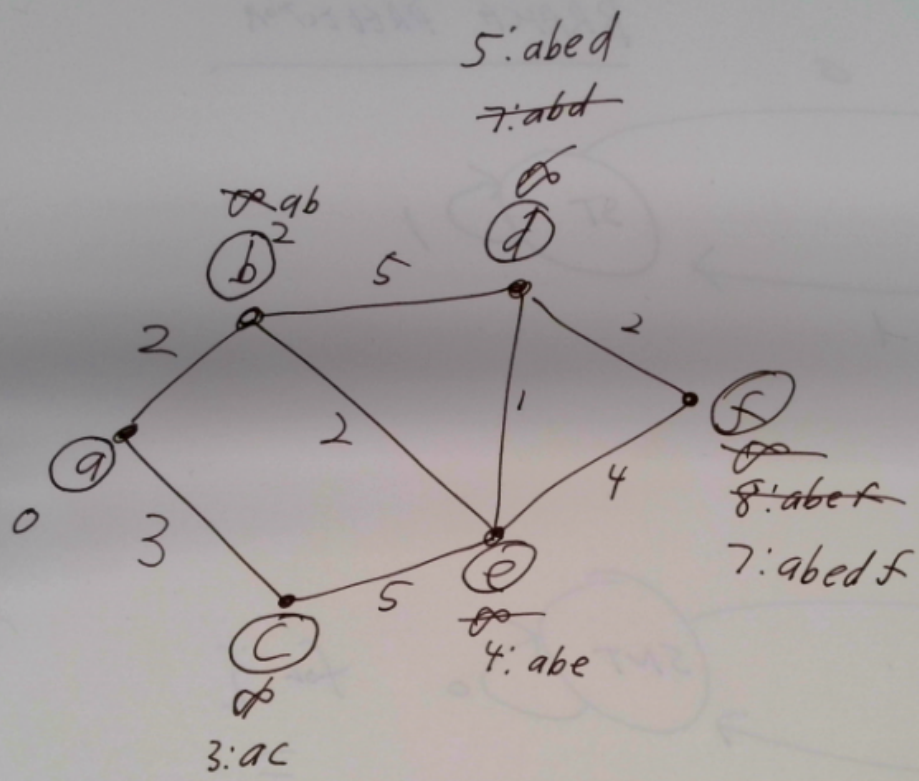
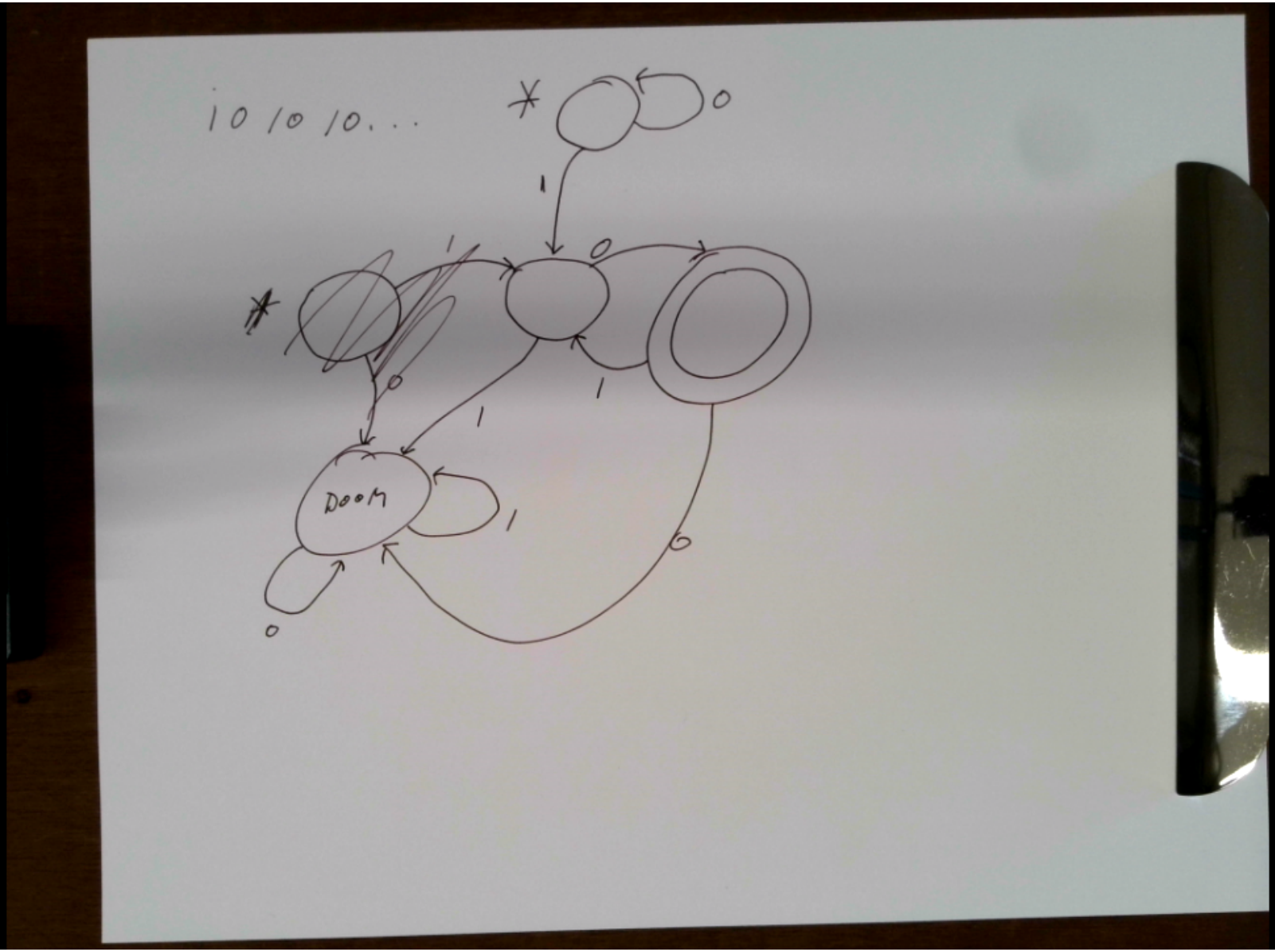
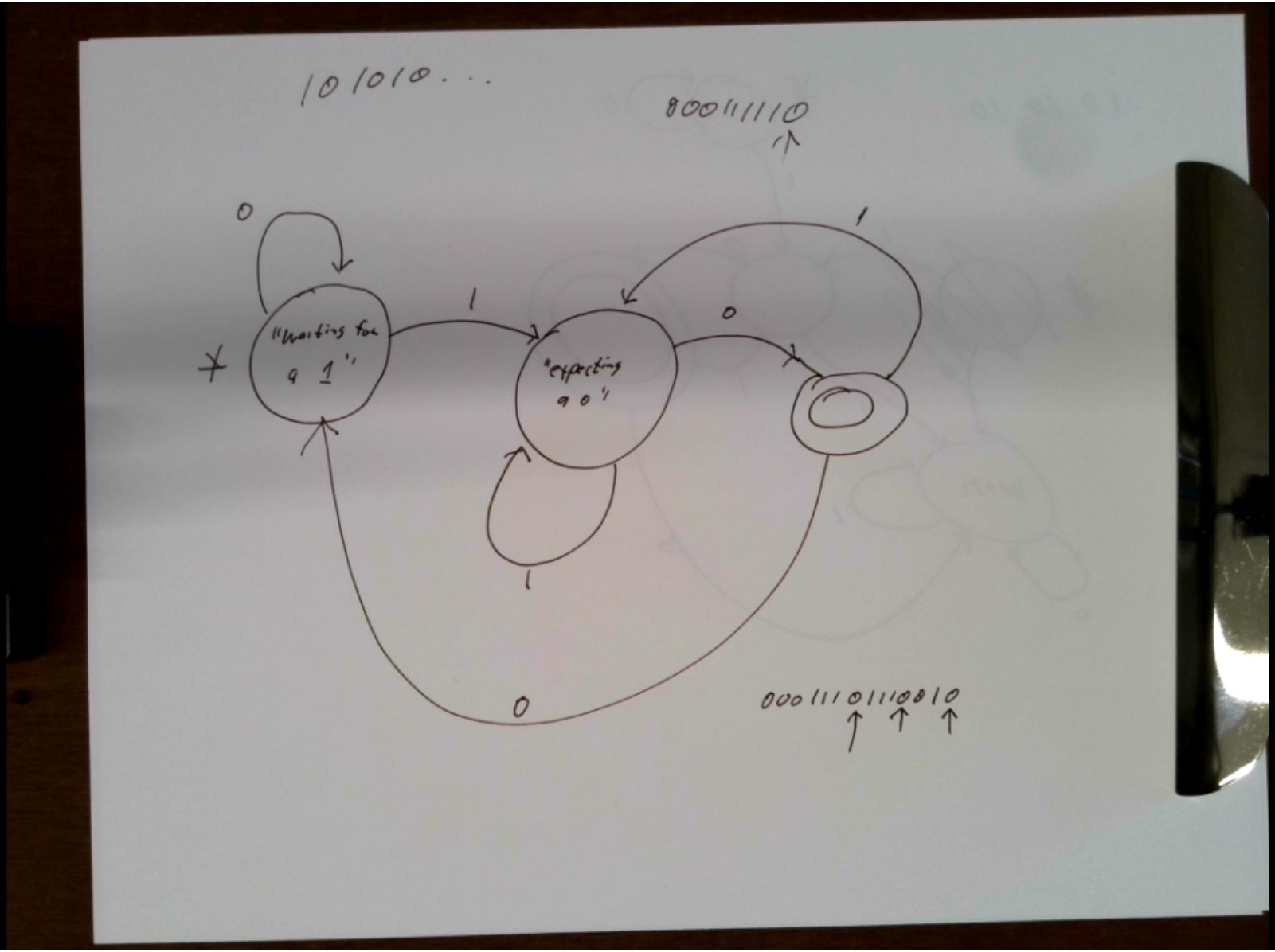
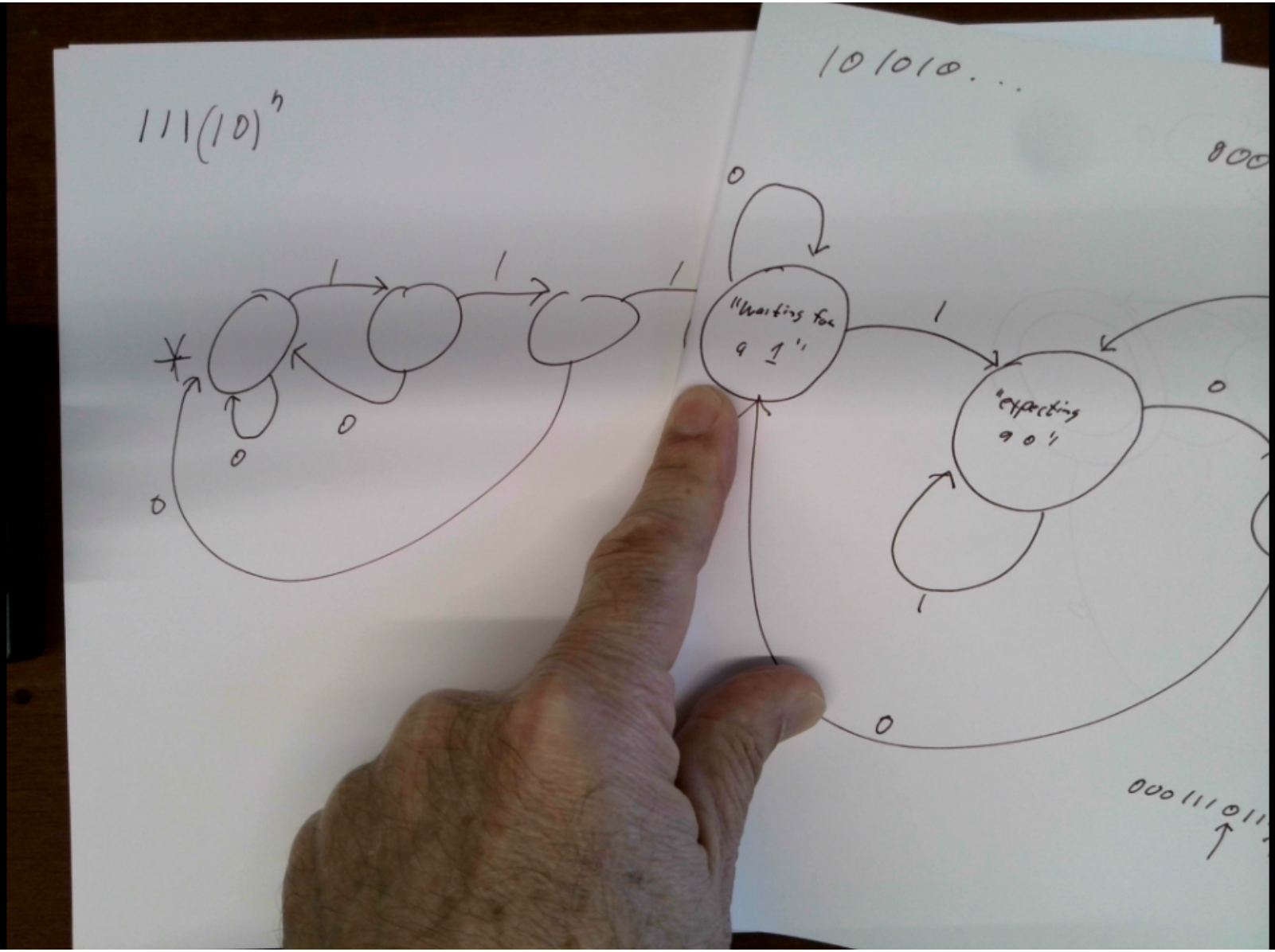
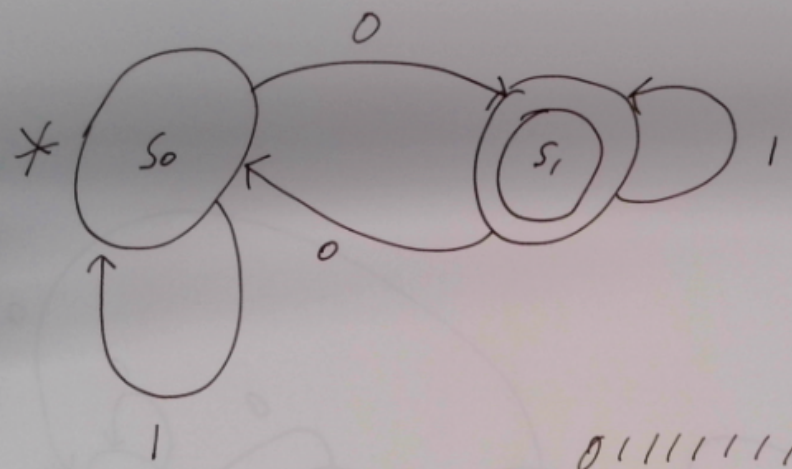

5: abed
~~7: abd~~
~~∞~~
~~∞~~ ab
2
b
d
5
2
2
1
2
a
f
4
~~8: abef~~
0
3
7: abedf
e
5
~~∞~~
4: abe
c
~~∞~~
3: ac


10 10 10...
DOOM


101010...
800111110
0
1
"Waiting for a 1"
1
"expecting a 0"
0
1
0
000111011100010


111(10)^n
101010...
"Waiting for a 1"
"Expecting a 0"
0
1
000111011

What Strings does this recognize?



0 1 1 1 1 1 1 1 1 1

0 0 0 1 1 1 1 1 1

1 1 1 0 1 1 1

BNF

Backus-Naur form

AS: = <VAR> = <EXP>

EXP: = <EXP> <OP> <EXP> | (<EXP>) | <VAR> | <NUM>

VAR: = <Letter> <VAR2> | <Letter>

VAR2: = <Symbol> <VAR2> | <SYMBOL>

NUM: = - <UNUM> | <UNUM>

UNUM: = <digit> <UNUM> | <digit>

digit: = 0 | 1 | 2 | ... | 9

letter: = A | B | C | ... | Y | Z

SYMBOL: = <letter> | <digit> | _

BNF

Backus-Naur form

AS: = <VAR> = <EXP>

EXP: = <EXP> <OP> <EXP> | (<EXP>) | <VAR> | <NUM>

VAR: = <Letter> <VAR2> | <Letter>

VAR2: = <Symbol> <VAR2> | <SYMBOL>

NUM: = - <UNUM> | <UNUM>

UNUM: = <digit> <UNUM> | <digit>

digit: = 0 | 1 | 2 | ... | 9

letter: = A | B | C | ... | Y | Z

SYMBOL: = <letter> | <digit> | _

OP: = + | - | * | /

BNF

Backus-Naur form

AS: = <VAR> = <EXP>

EXP: = <EXP> <OP> <EXP> | (<EXP>) | <VAR> | <NUM>

VAR: = <Letter> <VAR2> | <Letter>

VAR2: = <Symbol> <VAR2> | <SYMBOL>

NUM: = - <UNUM> | <UNUM>

UNUM: = <digit> <UNUM> | <digit>

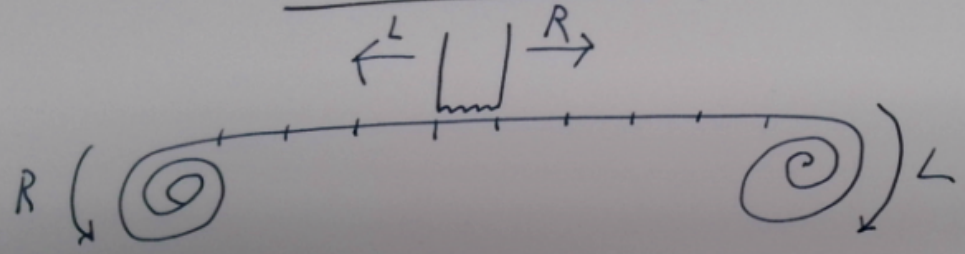
digit: = 0 | 1 | 2 | ... | 9

letter: = A | B | C | ... | Y | Z

SYMBOL: = <letter> | <digit> | _

OP: = + | - | * | /

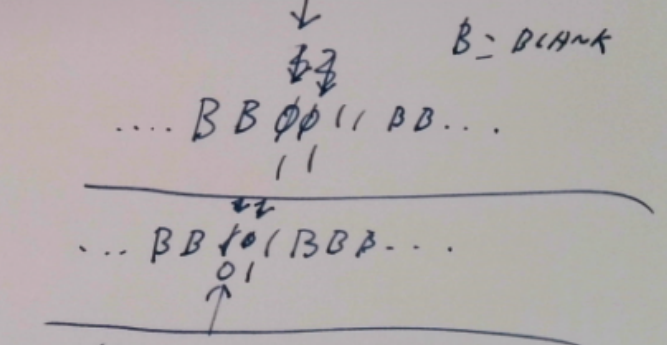
Turing Machines



(Present state, Symbol, Next state, new Symbol, L/R) S_0 = init. state.

Ex:
$(S_0, 0, S_1, 1, R)$
$(S_0, 1, S_1, 0, R)$
$(S_0, B, S_1, 0, R)$
$(S_1, 0, S_2, 1, L)$
$(S_1, 1, S_1, 0, R)$
$(S_1, B, S_2, 0, L)$

B = BLANK



STATE: ~~S_0~~ ~~S_1~~ ~~S_2~~
~~S_0~~ ~~S_1~~ ~~S_2~~
S_0 ~~S_1~~ ~~S_2~~

$(S_0, 0, S_0, 0, L)$

$(S_0, 1, S_1, 1, L)$

$(S_1, 0, S_1, 1, L)$

$(S_1, 1, S_1, 0, L)$

BB... BB 0 0 1 0 1 0 0 BB...
1 1 0 1

State = S_0 S_1

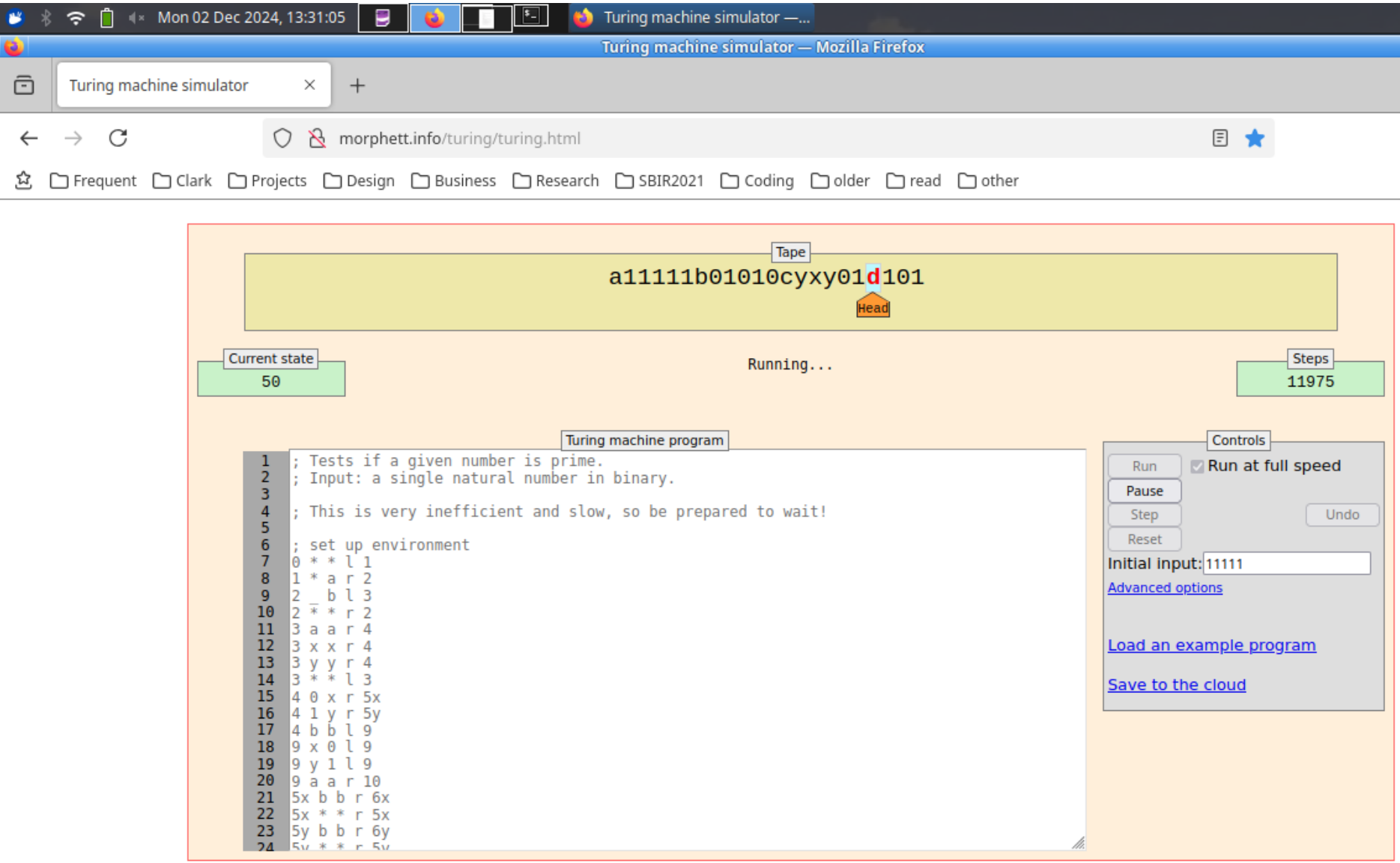
in: 0 0 1 0 1 0 0

out: 1 1 0 1 1 0 0

$(S_{10}, 0, S_{10}, 0, R)$

$(S_{10}, 1, S_{10}, 1, R)$

(S_{10}, B, S_0, B, L)


Mon 02 Dec 2024, 13:31:05
Turing machine simulator —...
Turing machine simulator — Mozilla Firefox
Turing machine simulator
morphett.info/turing/turing.html
Frequent
Clark
Projects
Design
Business
Research
SBIR2021
Coding
older
read
other
Tape
a11111b01010cyxy01d101
Head
Current state
50
Running...
Steps
11975
Turing machine program
; Tests if a given number is prime.
; Input: a single natural number in binary.
; This is very inefficient and slow, so be prepared to wait!
; set up environment
0 * * l 1
1 * a r 2
2 _ b l 3
2 * * r 2
3 a a r 4
3 x x r 4
3 y y r 4
3 * * l 3
4 0 x r 5x
4 1 y r 5y
4 b b l 9
9 x 0 l 9
9 y 1 l 9
9 a a r 10
5x b b r 6x
5x * * r 5x
5y b b r 6y
Controls
Run
Run at full speed
Pause
Step
Undo
Reset
Initial input: 11111
Advanced options
Load an example program
Save to the cloud

Syntax:

$(S_0, 0, S_0, 0, L)$

$(S_0, 1, S_1, 1, L)$

$(S_1, 0, S_1, 1, L)$

$(S_1, 1, S_1, 0, L)$

BB... BB 0 0 1 0 1 0 0 BB...

1101

State = S_0 S_1

in: 0 0 1 0 1 0 0

out: 1 1 0 1 1 0 0

$(S_{10}, 0, S_{10}, 0, R)$

$(S_{10}, 1, S_{10}, 1, R)$

(S_{10}, B, S_0, B, L)

DETERMINISTIC TURING MACHINE (DTM)

NON-DETERMINISTIC TURING MACHINE

(NDTM)

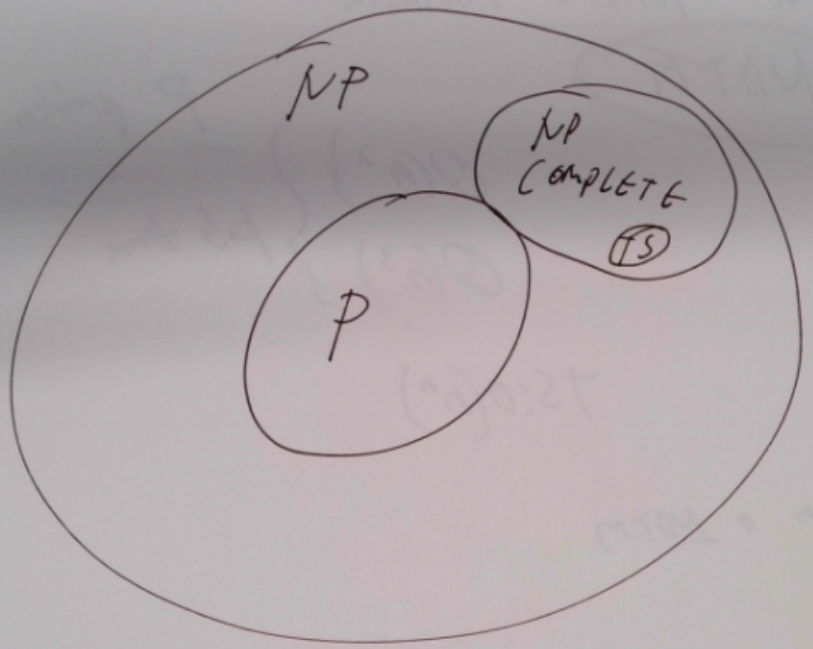
$O(n^2)$, $O(n^3)$ } poly time — P problem

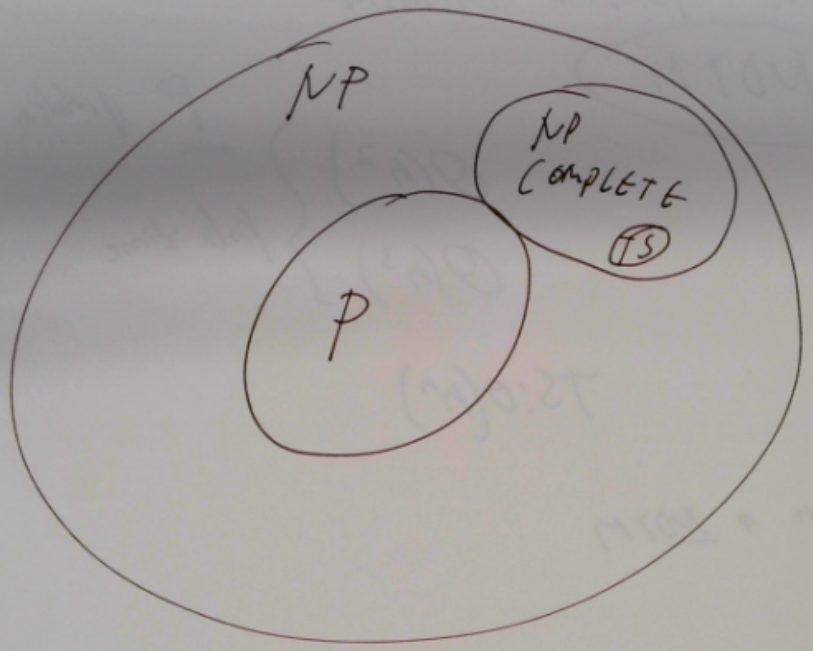
TS: $O(n^n)$

TS: Poly time on a NDTM

NP problem

Q: Can TS be solved in poly time on a DTM


NP
NP
COMPLETE
(TS)
P


NP
NP
COMPLETE
TS
P

Halting Problem

$(S_0, 0, S_0, 0, L)$
$(S_0, 1, S_0, 1, L)$
(S_0, B, S_0, B, L)

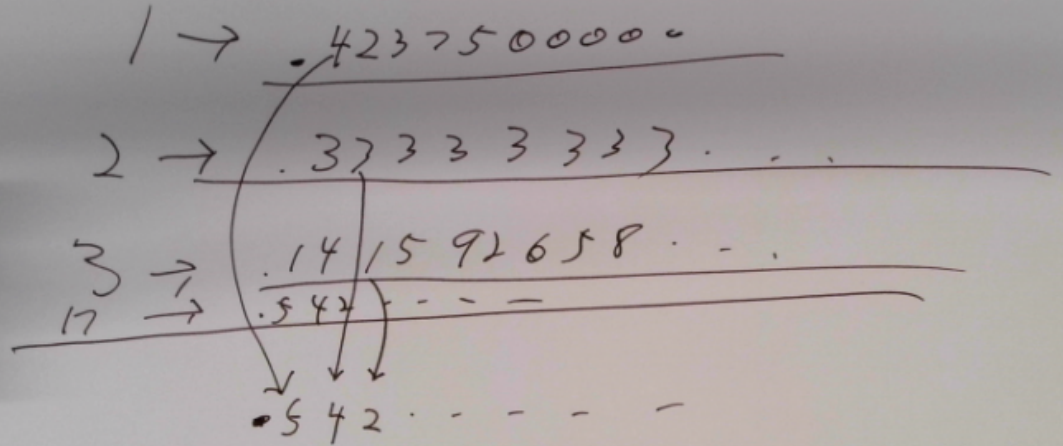
$|\mathbb{Z}| = |2\mathbb{Z}| = |\mathbb{Z} > 100|$

1-1 corr.

$x \longrightarrow 2x$

$|\mathbb{R}| > |\mathbb{Z}|$

Suppose $f: |\mathbb{N}| \to [0,1] \subseteq \mathbb{R}$



Suppose $f: |\mathbb{Z}^+| \to [0,1] \subseteq \mathbb{R}$