Math 322

Q's / 12.4 #13b

Regular Sets

Basis Step: $\{\ \}, \{\lambda\}, \{x\}$ $x \in I$

are all Regular

Inductive Step: A, B are regular

$\rightarrow AB, A \cup B, A^*$ are regular

Set is regular $\boxed{\text{iff}}$ #1 it is recognized by a F.S.A.

#2 $\boxed{\text{iff}}$ it is generated by a $\boxed{\text{Regular Grammar.}}$

productions $A \rightarrow a$

$A \rightarrow aB$

For #1 Make a F.S.A based on the inductive definition.

Basis:

$\{\ \}$ is regular

$M_{\{\}}$: start $\rightarrow S_0$

$\{\lambda\}$ is regular

$M_{\{\lambda\}}$: start $\rightarrow S_0$

$\{x\}$ $x \in I$ is regular

$M_{\{x\}}$: start $\rightarrow S_0 \xrightarrow{x} S_1$

Inductive

AB, $A \cup B$, A^*

final?

assume M_A: start $\rightarrow S_A \rightarrow \cdots \rightarrow \bigcirc$

assume M_B: start $\rightarrow S_B \rightarrow \cdots \rightarrow \bigcirc$

A, B are regular

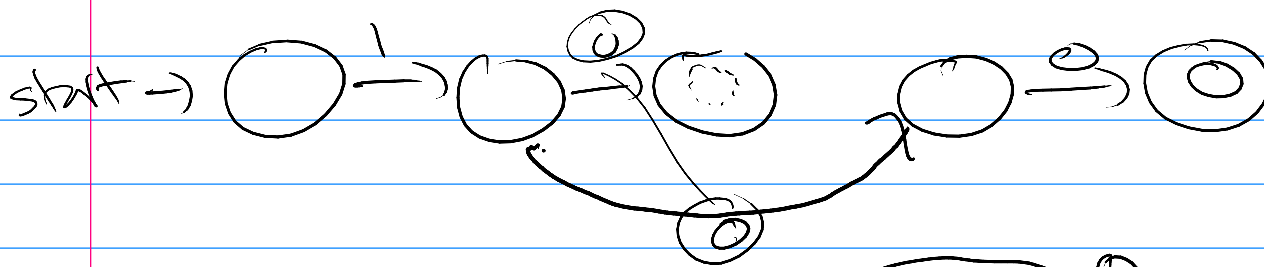
1. M_{AB}

ex M_A: start $\rightarrow \bigcirc \xrightarrow{1} \bigcirc \xrightarrow{0} \bigcirc$

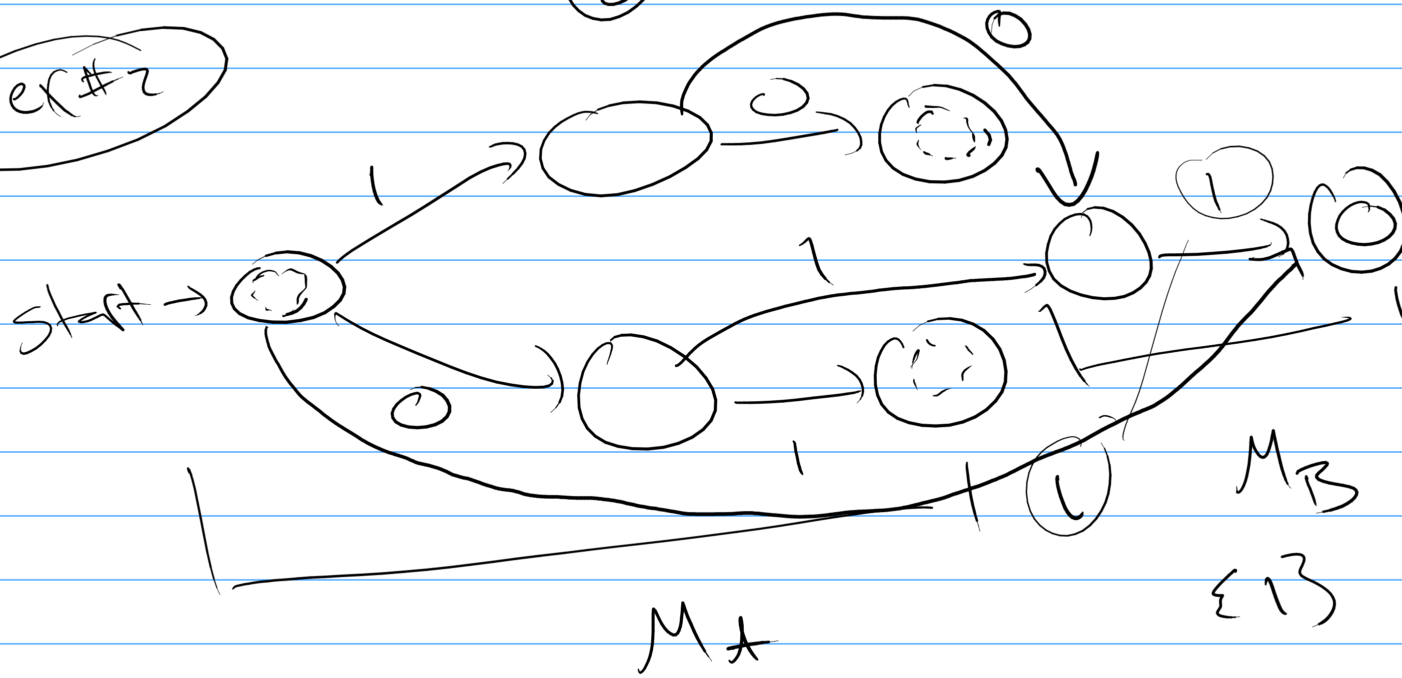
M_B: start $\rightarrow \bigcirc \xrightarrow{0} \bigcirc$

ex $L(M_A) = \{10\}$ $L(M_B) = \{0\}$

$L(M_{AB}) = \{100\}$

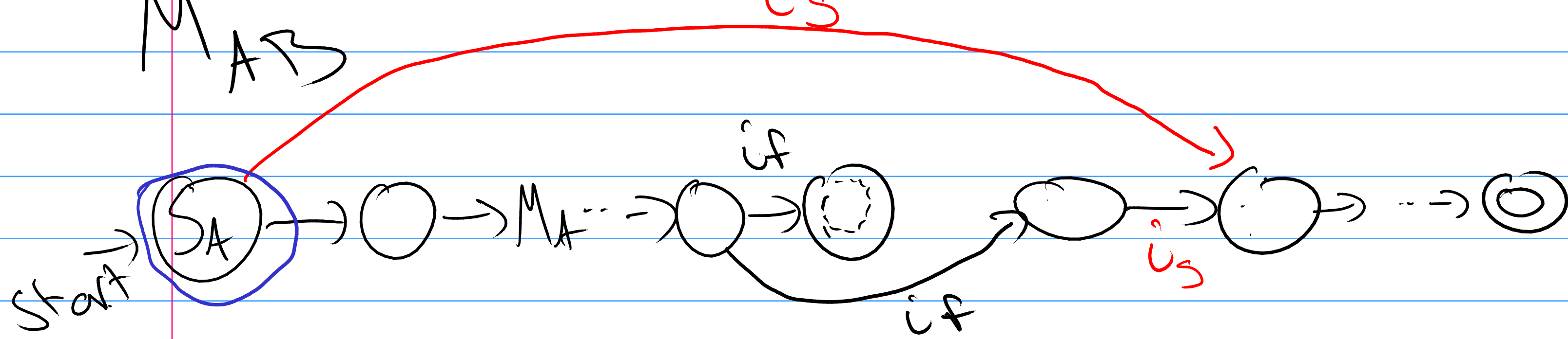


ex#2



$\{R, 10, 01\}$

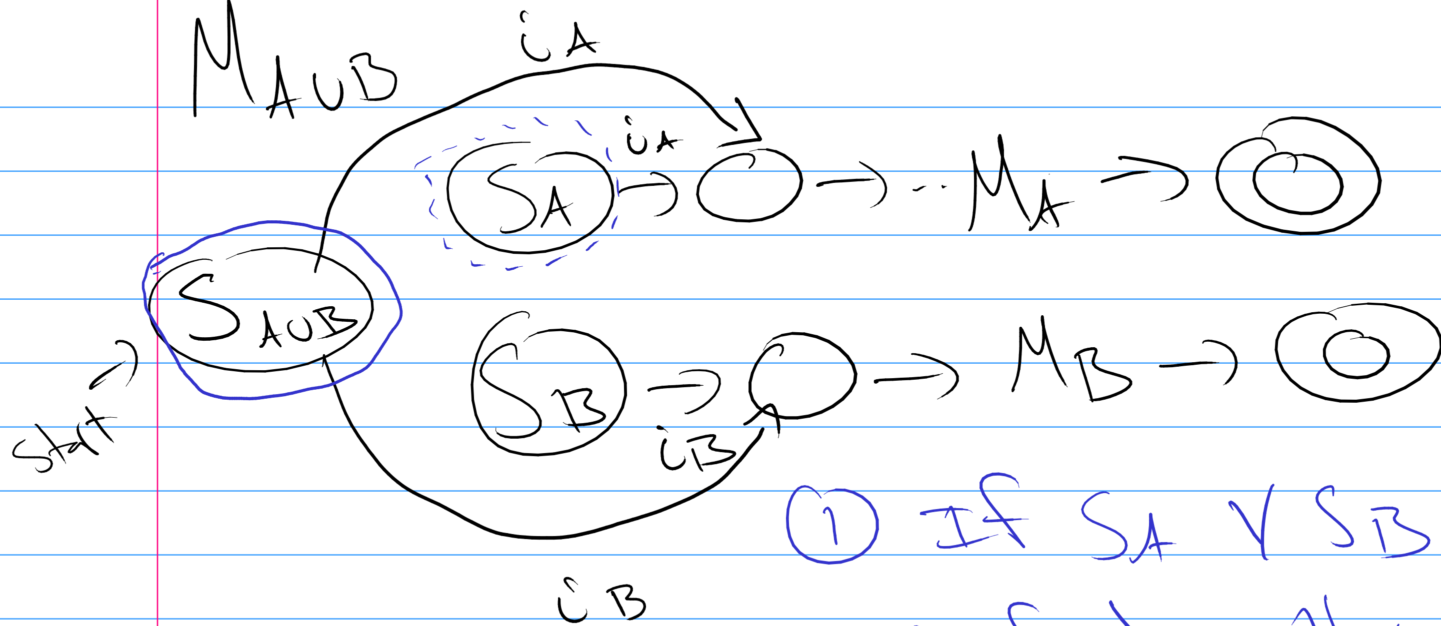
M_{AB}



① if S_A is final use red edge

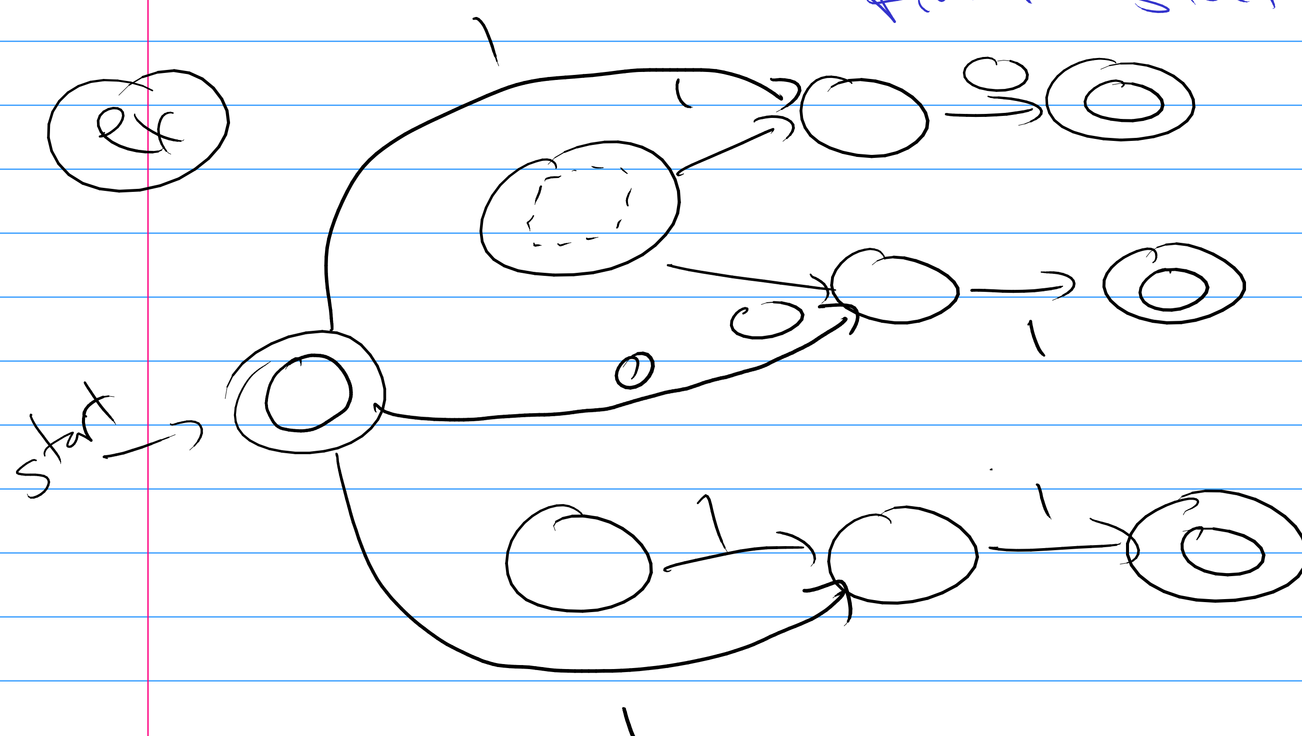
② if $S_A \wedge S_B$ are final S_{AB} is final

$M_{A\cup B}$

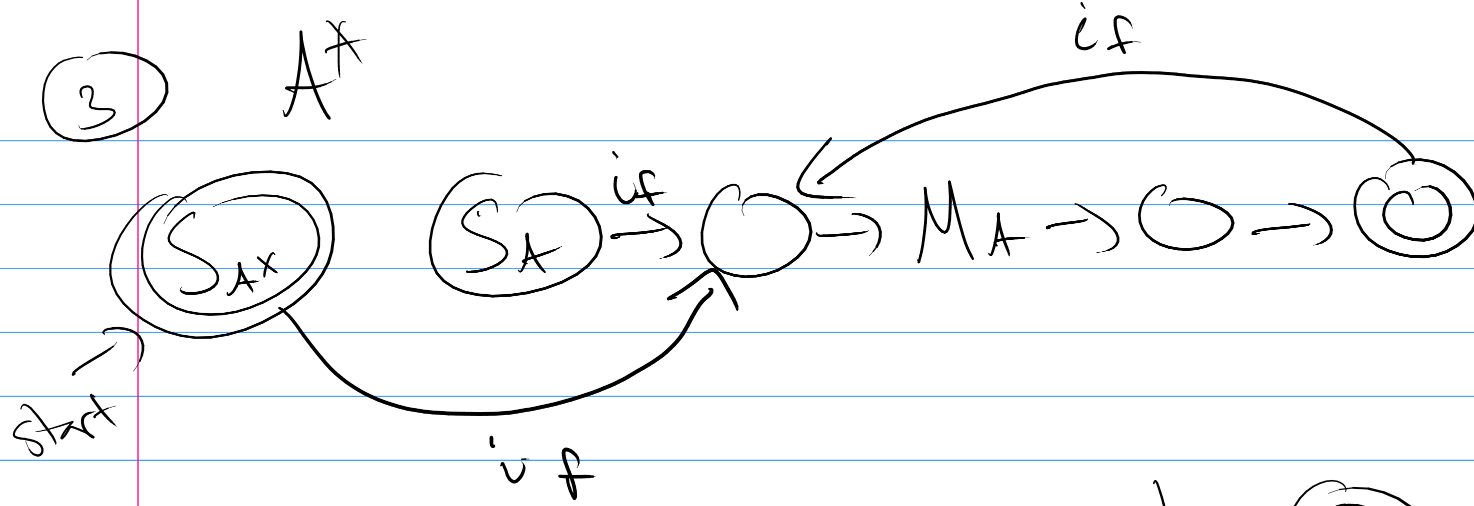


① If $S_A \vee S_B$ is final then $S_{A\cup B}$ is final (remove S_A or S_B final status)

ex



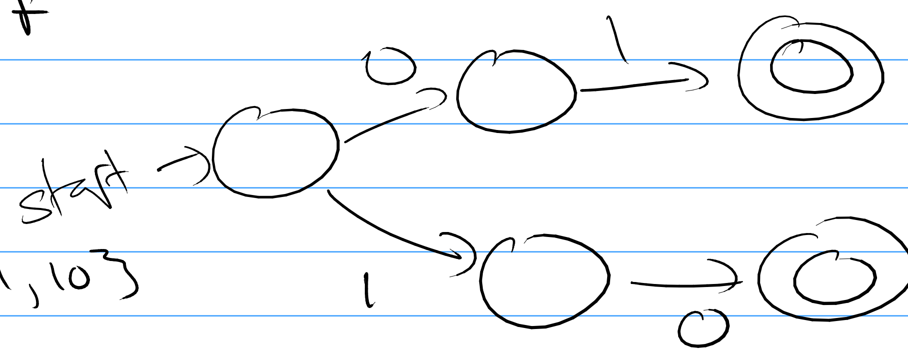
3 A^*



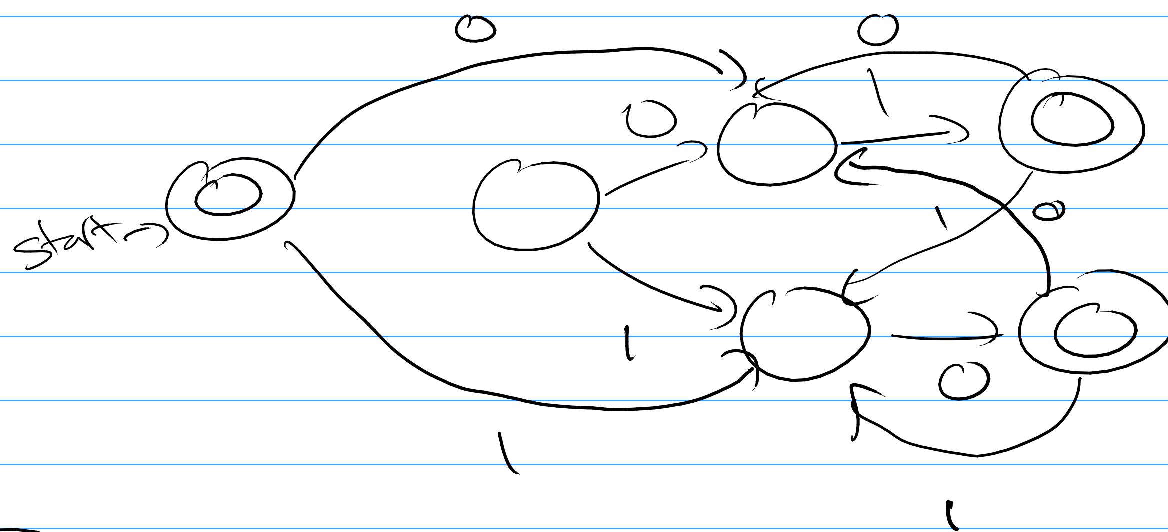
ex

M_A

$L_{M_A} = \{01, 10\}$



M_{A^*}

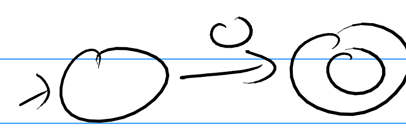


Q17b

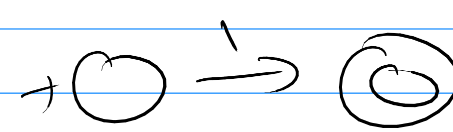
12.4 #13b

$(0 \cup 11)^*$

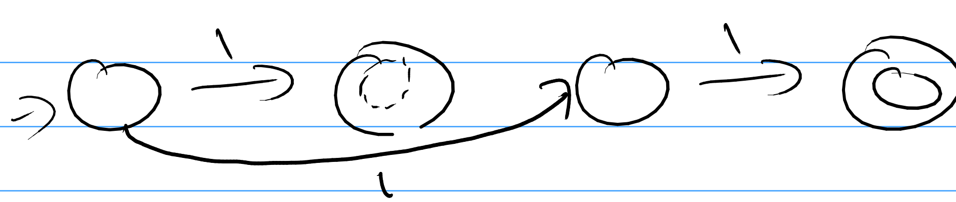
M_0:



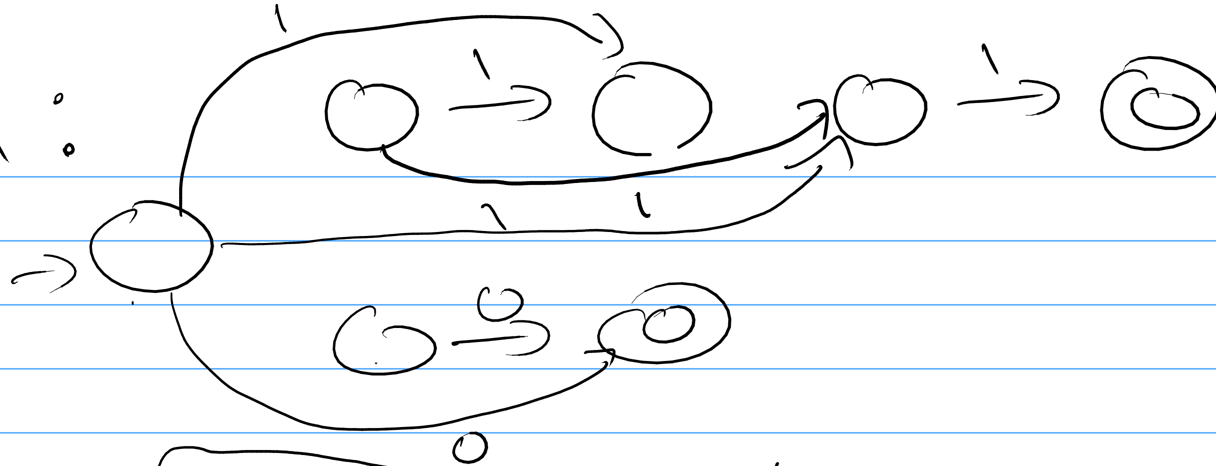
M_1:



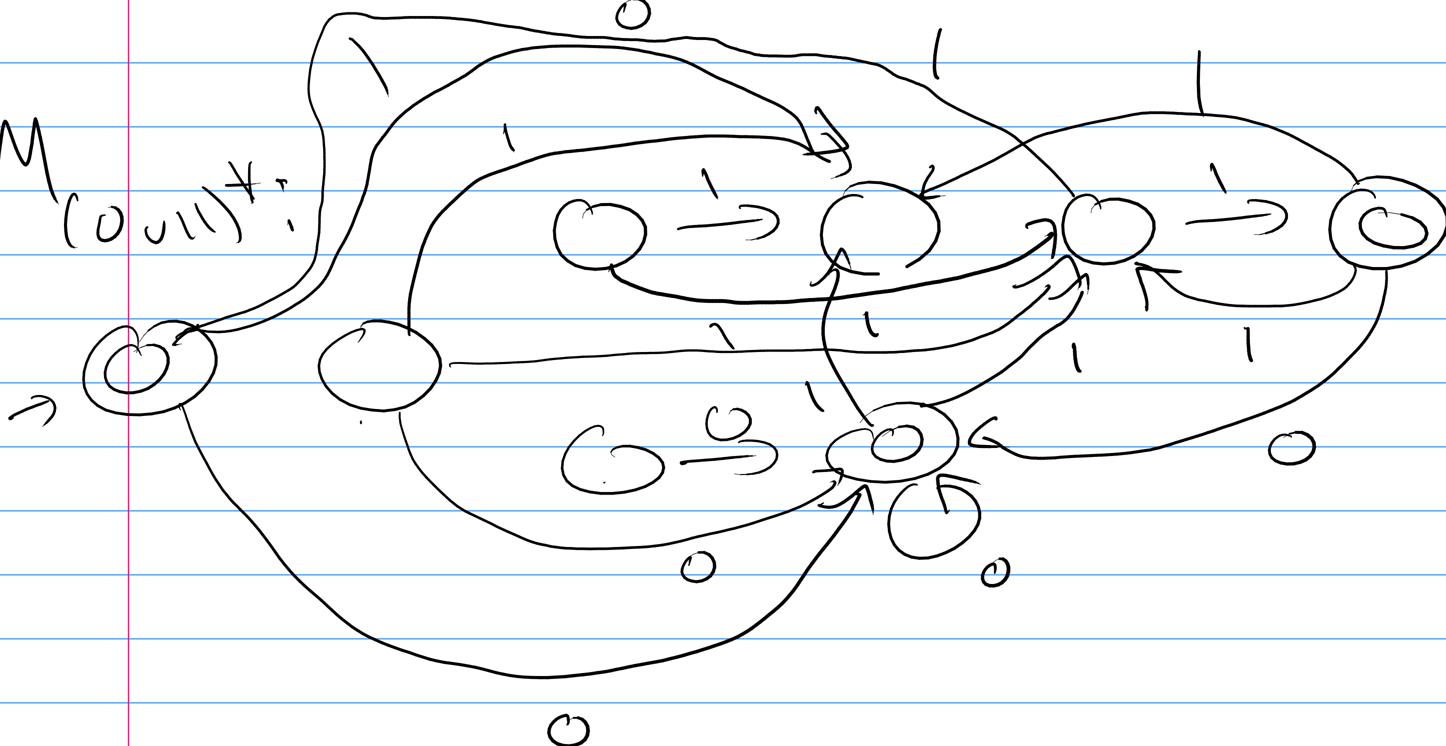
M_{11}:



$M_{0\cup 11}$:



$M_{(0\cup 11)^*}$:



$\boxed{\text{Regular Sets} = \text{F.S.A}}$ <u>iff</u> Regular Grammar.
(productions matter)

Regular Grammar → F.S.A

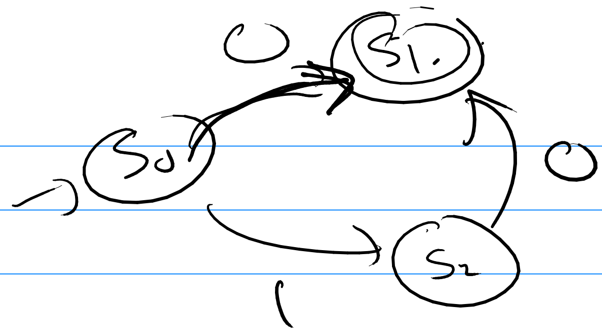
$S \to \lambda$

$A \to a$

$A \to aB$

state transitions. ?

Consider:



$L(M) = \{0, 10\}$

$$(S_0, 0) \to S_1$$

$$S_0 \to 0 S_1$$

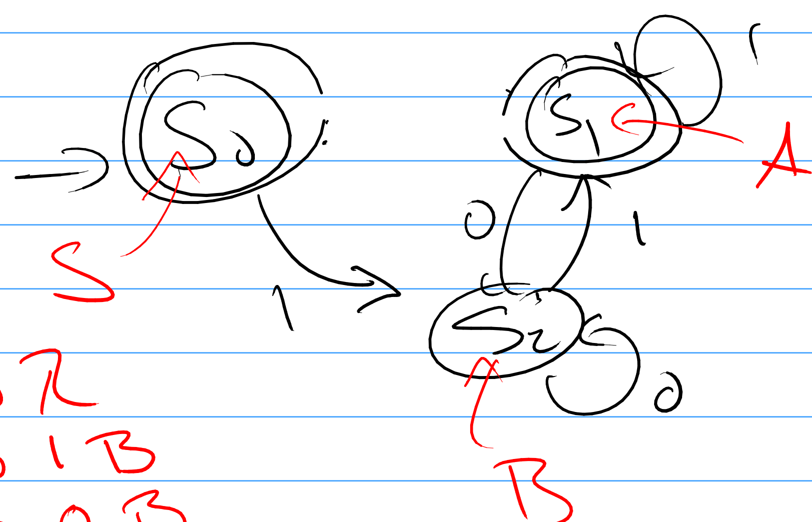
$$S_0 \to 0$$

States = NonTerm

inputs = term

---

ex



$S \to \lambda$

$S \to 1B$

$B \to 0B$

$B \to 1A$ } state trans.

$B \to 1$ } state trans. to final state

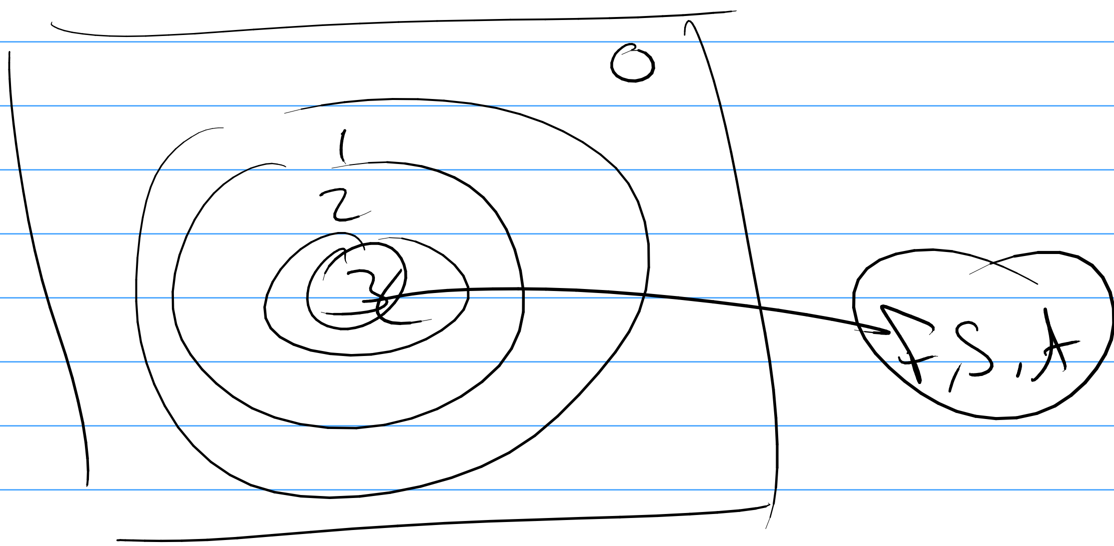
$A \to 0B$

$A \to 1A$

$A \to 1$

Regular Grammar iff F.S.A

	types		Machine
0	Phrase structure		?
1	Context sensitive		?
2	context free		?
3	Regular	iff	F.S.A



More powerful Machines.

$\boxed{\text{Type 2}}$ } F.S.A + push down memory (stack) to recognize a string in f.S.A (+) stack mem

1. empty memory
2. end in a final state

$\boxed{\text{type 1}}$ Context sensitive.

f.S.A (+) Linear bounded Memory.
(finite tape memory)

$\boxed{\text{type 0}}$ Phrase-Structure Grammar

f.S.A (+) Infinite Tape Memory.

(Turing Machines)

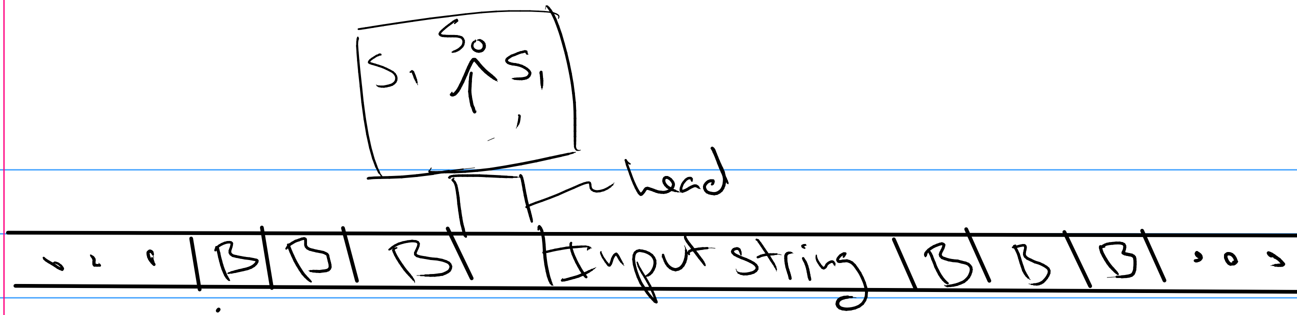
Def: $T = (S, I, f, S_0)$ is a Turing Machine

partial function $f : S \times I \to S \times I \times \{\text{left}, \text{right}\}$

set of (s, i, s', i', m) 5-tuples

list these to define T

plus $B \in I$
blank



1. get s, i pair
2. find 5-tuple that has that pair in front of it. (s, i, s', i', m)
3. change to new state s'
4. write i' to same location
5. move m one spot.

ex

$$B\ B\ \overbrace{1\ 1\ 1\ 1\ 1\ 1\ 1\ 1}^{6}\ B$$

$$B\ B\ \underbrace{1\ 1\ 1}_{2} * \underbrace{1\ 1\ 1\ 1\ 1}_{4}\ B\ B$$

$6 \to 7$