Theory of Computation

Basics/DFA

Lecture 2

Gaurav Raj

Symbol : anything {क, ख, a, b, 2, @, ∈, +, −, ...} Ex {a, b}

Alphabet Σ, Δ, Γ Ex Σ^2, Σ^3, Σ^* {Power}

Σ^2 ↓ Length 2

String : $\in$ or λ, a, b (1), aa, ab, ba, bb (2), ...

Epsilon — 0 length String

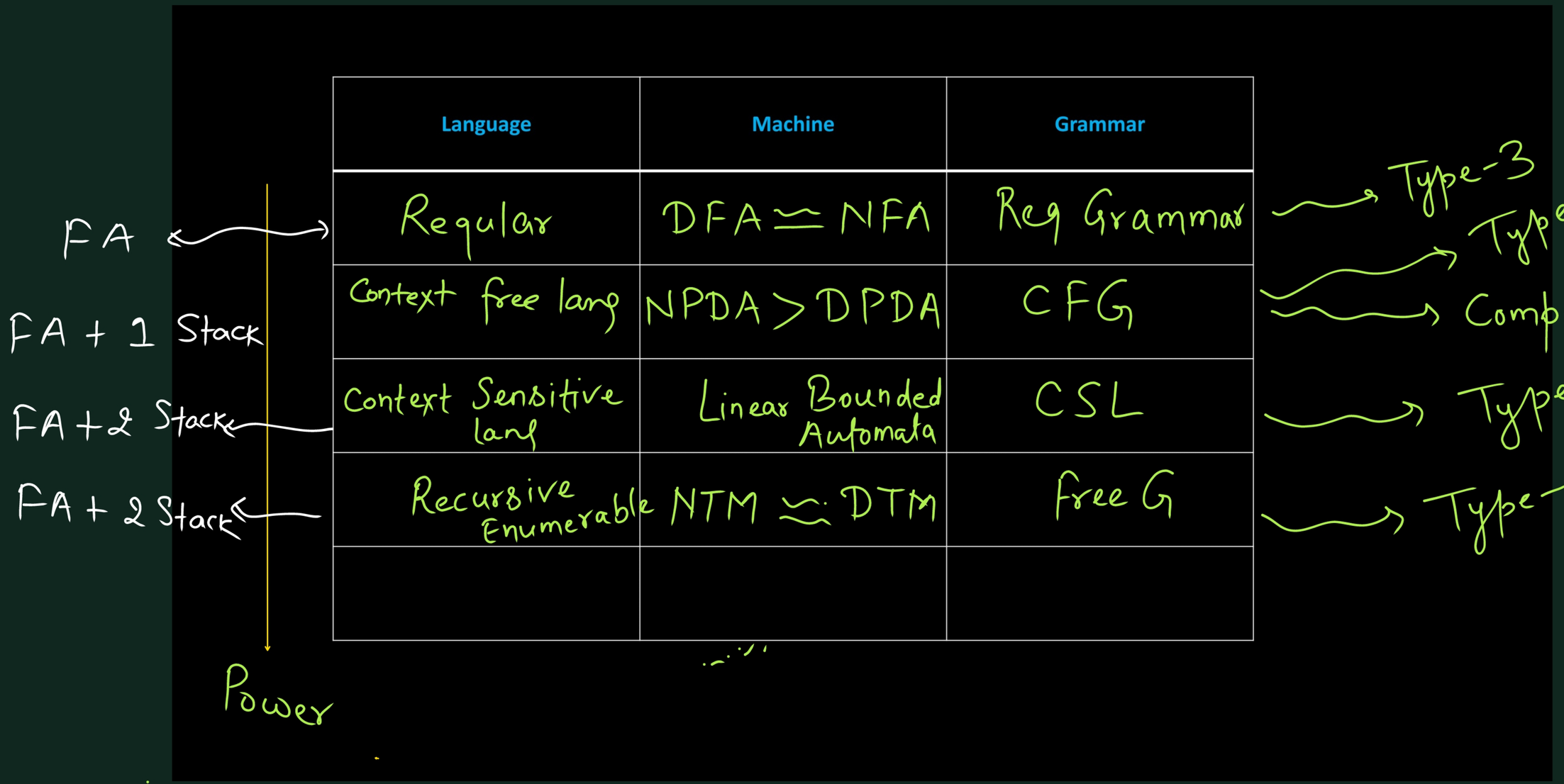
Language :

L1: aba

L2: $a^* b^*$

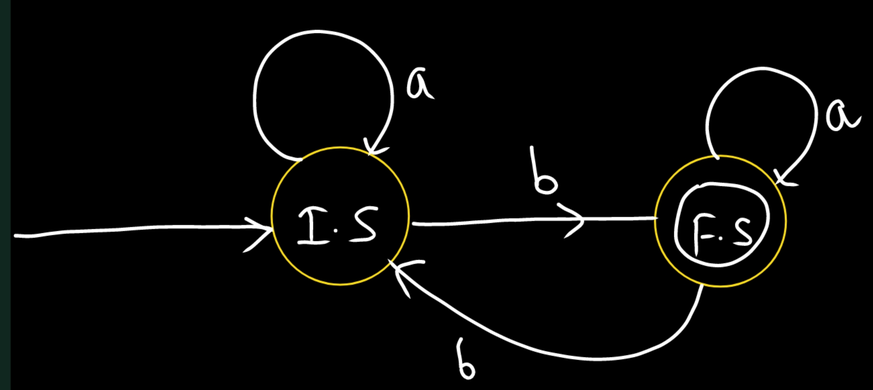
{∈, a, b, ab, ~~ba~~ ...}

a^* : {∈, a, aa, ..., aaa}

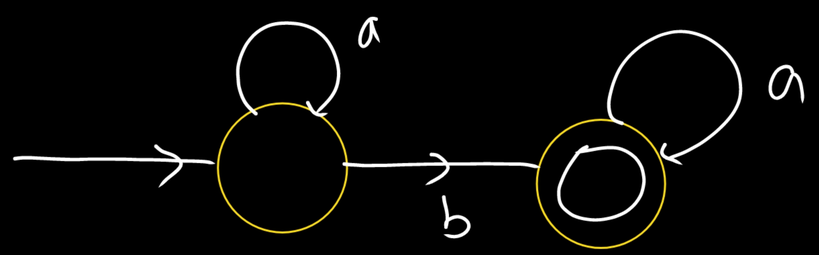

Language
Machine
Grammar
Regular
DFA ≃ NFA
Reg Grammar
Context free lang
NPDA > DPDA
CFG
Context Sensitive lang
Linear Bounded Automata
CSL
Recursive Enumerable
NTM ≃ DTM
Free G
FA
FA + 1 Stack
FA + 2 Stack
FA + 2 Stack
Power
Type-3
Type-2
Compiler
Type-1
Type-0

DFA vs NFA : Transition Table

$\Sigma = \{a, b\}$



DFA



NFA

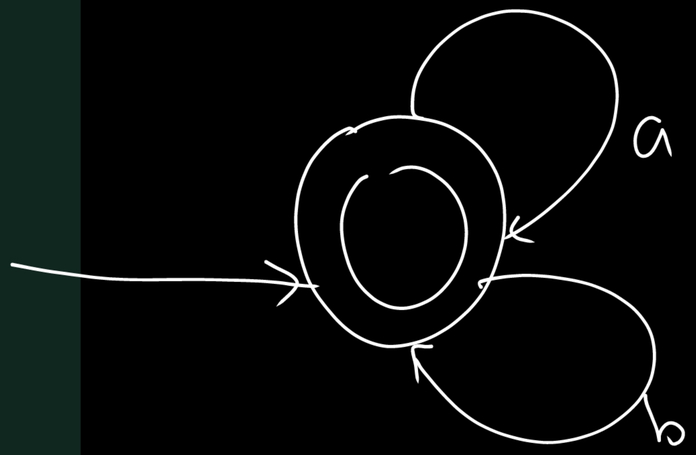
Transition Table

	a	b
I.S	I.S	F.S
F.S	F.S	I.S

	a	b
IS	IS	FS
FS	F.S	—

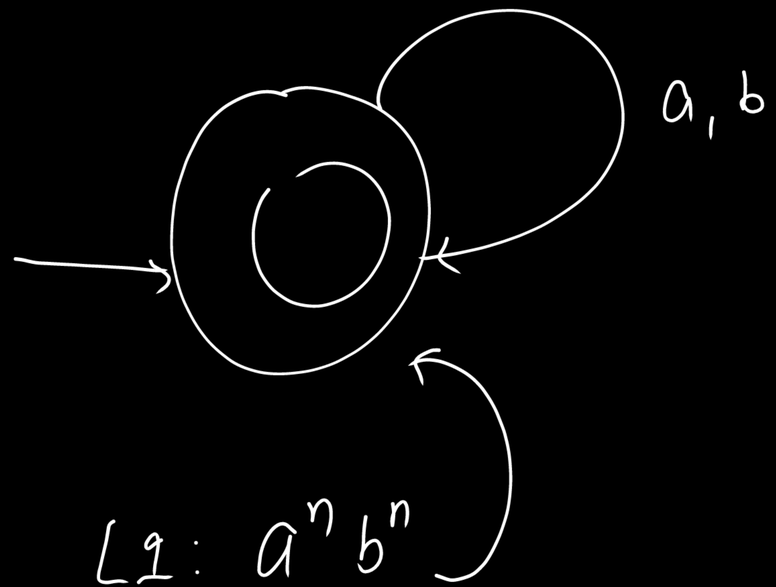
1. Construct minimal Finite Automata that accepts all strings of a's and b's including ϵ

$L1: \{ \epsilon,$



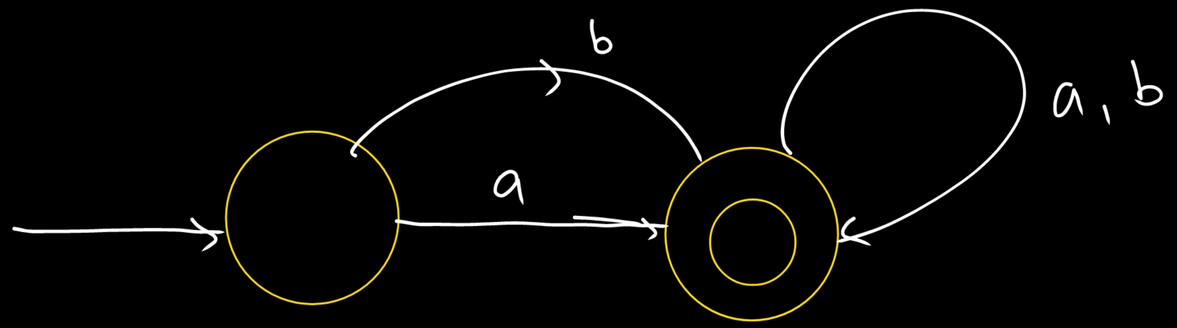
Universal lang.

$\equiv$



$L1: a^n b^n$

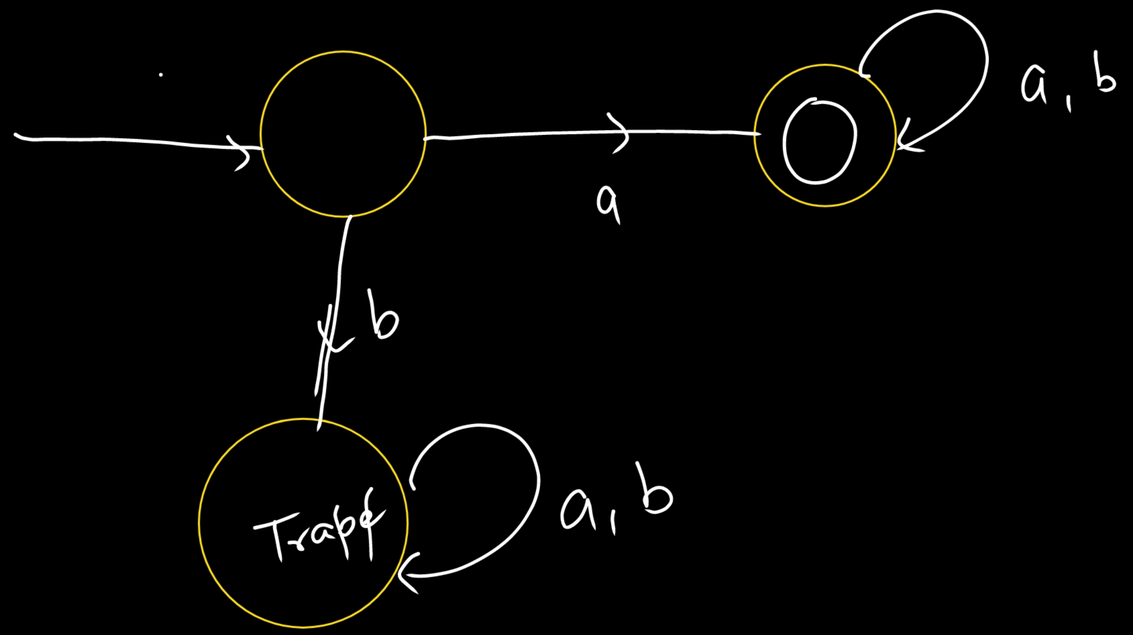
2. Construct minimal Finite Automata that accepts all strings of a's and b's excluding ϵ



L1: { a, b, ab, ba, aa, bb ... }

3. Construct minimal Finite Automata that accepts all strings of a's and b's, where each string starts with a.

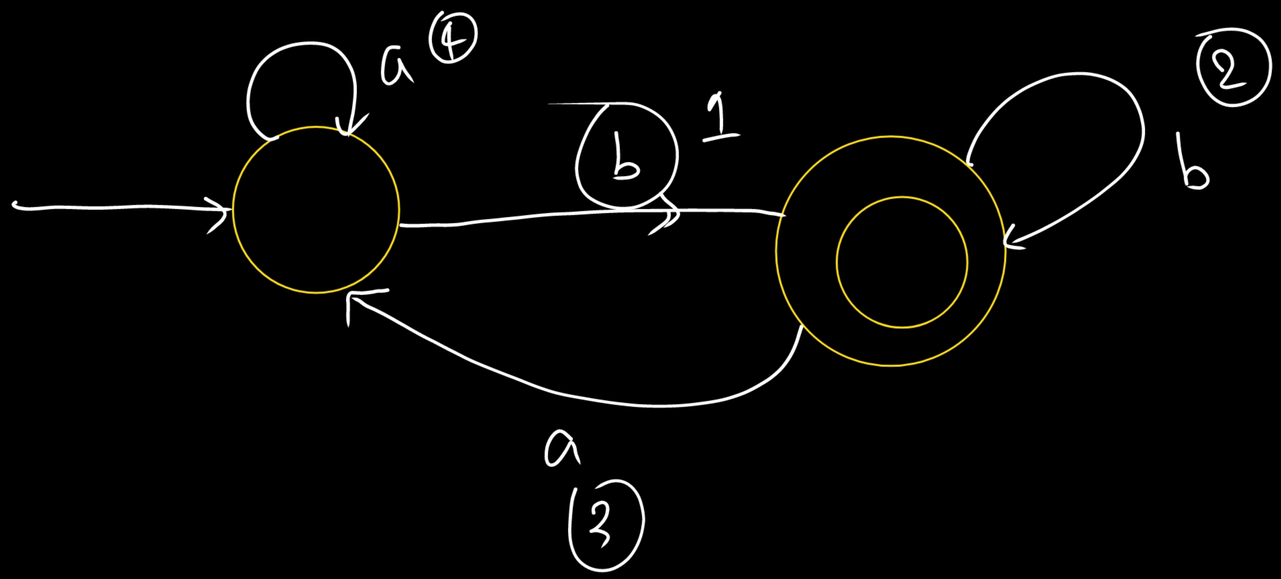
L1 : { a, b, aa, ab, ba, bb, -- ... }
(b, ba, bb are crossed out)



	a	b
I.S	F.S	T.S
F.S	F.S	F.S
T.S	T.S	T.S

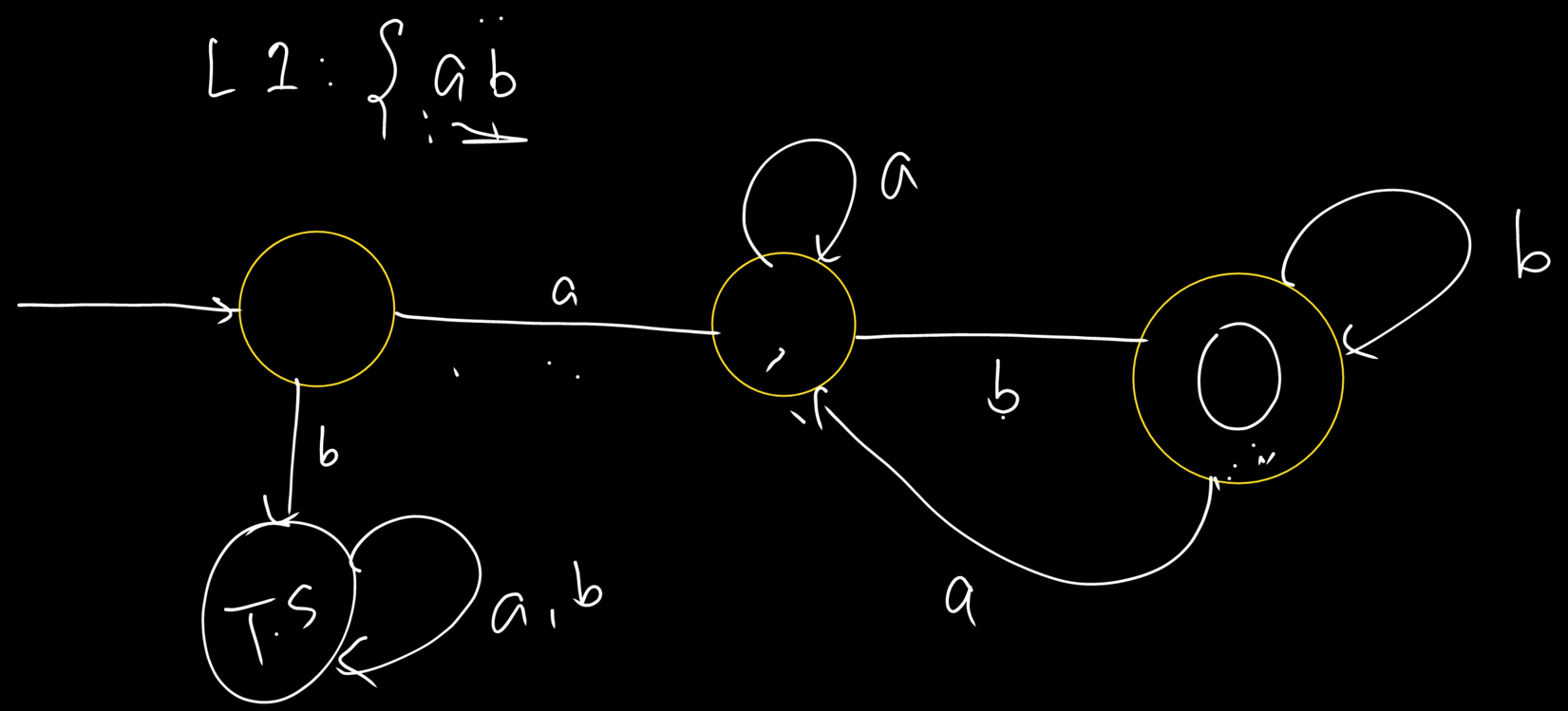
4. Construct minimal Finite Automata that accepts all strings of a's and b's, where each string ends with b.

L1: { a, b, aa, ab, ba, bb, - - - aaaaaab infinite }
(a, ab, ba, bb, ... aaaaaab)



b bbb a b

5. Construct minimal Finite Automata that accepts all strings of a's and b's, where each string starts with a and ends with b.



6. Construct minimal Finite Automata that accepts all strings of a’s and b’s, where each string contain ab as substring.

7. Construct minimal Finite Automata that accepts all strings of a’s and b’s, where each string starts with the substring ab.

Thank You