Formal Methods - End Sems

- ✅ Chapter 0 - 6:00 - 7:00
- ✅ Chapter 1 - 7:00 - 12:00
- ✅ Chapter 2 - 12:00 - 6:00

Chapter 1

- Finite Automata **< Q, ∑ , ∂ , q , F >**
- *(Regular)* Language
- Proof of Union, Concatenation, Star - Languages
- DFA vs NFA
- NFA to DFA
- Regular Expressions - start **->** concat **->** union
- Regular Expression to NFA
- NFA to GNFA to Regular Expression
  - GNFA+ 3 rules
- DFA -> GNFA -> Regular Expression
- Non-regular Languages
- Pumping Lemma for Regular Languages

Chapter 2

- CFG is a 4 tuple : **< V , ∑ , R , S >**
  - **V** is a set of variables
  - **∑** is a set of terminals *DISJOINT from V*
  - **R** is a set of rules.
  - **S** is the start state.
- Rule of Unions
- Rule of Memory
- CFG Construction
- DFA to CFG
- Ambiguity
- Leftmost Derivation
- Inherently Ambiguous Grammar
- Chomsky Normal Form
- CFG to CNF - Rules
  - Add new start state

    - Remove e rules unless start
    - Remove unit rules
    - Further Conversion

- Pushdown Automata is a 6 tuple **< Q , ∑ , tau , ∂ , q , F>**
    - Q is a set of states
    - ∑ is the set of alphabets
    - Tau is the set of stack
    - ∂ is the transition table
    - F is the set of accepted states
- Nondeterministic PDA
- Deterministic PDA
- PDA Construction

- Pumping Lemma for CFL
- Proof of PL for CFL