CS311 Formal Language and Automata

Tingting Chen
Computer Science
Cal Poly Pomona

Closure Properties of CFL

- Context-free languages are closed under:
  - Union. Constructing a CFG for the union is easy.
  - Concatenation.
  - Star operation.

- Context-free languages are not closed under:
  - Intersection. E.g., L1=$\{a^n b^n c^m\}$, L2=$\{a^n b^m c^m\}$
  - Complement.
- The intersection of a CFL and a regular language is context-free.

Automata

- A Finite Automaton has only its states to serve as memory - very limited in what languages it could recognize
- In a Pushdown Automaton we added a stack to a Finite Automaton - this gave it better memory, and allowed it to recognize more languages. But it was still limited by the nature of the stack.
- We could add another stack to a Pushdown Automaton. This would allow it to accept some non-context free languages (e.g., $a^nb^nc^n$).
- Or we could change the stack to a queue. This would allow other languages to be accepted (e.g., ww).

All of the automata we study in this class have a finite number of states.

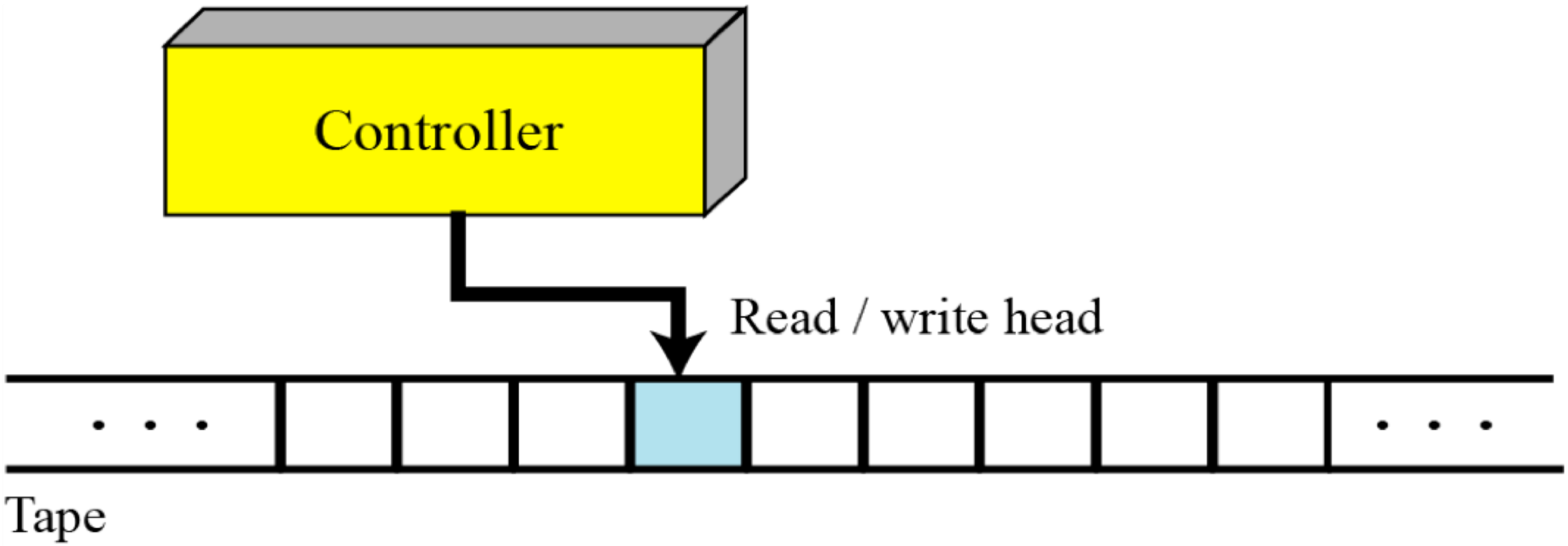
They differ in the "auxiliary memory" they have and how it is organized.

DFA NFA	finite memory
DPDA PDA	infinite stack memory
Turing machine	infinite tape memory (tape can be read forwards and backwards without erasing)

Standard Turing Machine

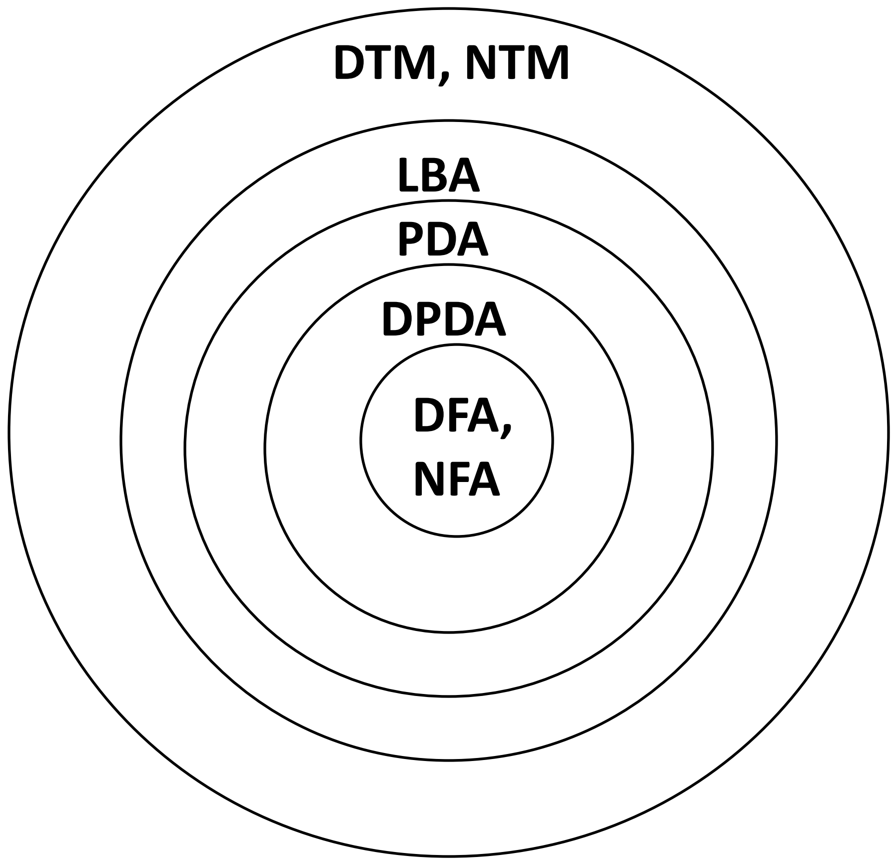
- The standard Turing machine described by our textbook has several features:
- The TM has a tape that is *unbounded in both directions*.
- There is no special input file; the input is copied onto the tape.
- There is no special output device; the tape contains the output, if any.

Components of Turing machine



1. A Tape
2. Controller
3. Read/Write Head

Hierarchy of automata



The Turing Machine

- Alan M. Turing, b. 1912, d. 1954. Contributed much to the foundations of computing theory.
- Published the Turing machine model in 1937.
- Church-Turing Thesis - “Any algorithmic procedure that can be carried out by a human, a team of humans, or a machine can be carried out by some Turing machine.”
- Unproveable, because we don’t have a precise definition of what “algorithmic procedure” means, but generally accepted as true.
- Puts a limit on what can be computed.

The Church-Turing Thesis

- No model of digital computation is more powerful than a Turing machine.
- By “more powerful,” we mean “can recognize languages that a TM cannot recognize.”
- This is not something that can be proved. But everybody believes it because no one has been able to devise a more powerful model of computation.