

B501 COURSE OUTLINE

Computation Theory is to CS what Physics is to Engineering

Three pillars of Computation Theory

1. *Models of Computing:*

Their relations, unity, role of memory structure

2. *Computational Complexity:*

Resources (time, size), feasibility

3. *Computability Theory:*

Boundaries of computations.

Structure of the course

- 1: ***Data:***
Languages, sets, mappings, inductive data
(~ 2.5 weeks)

Structure of the course

- 1: ***Data:***
Languages, sets, mappings, inductive data
(~ 2.5 weeks)
- 2: ***Machine***-based computing from automata to Turing machines.
(~ 3 weeks)

Structure of the course

- 1: ***Data:***
Languages, sets, mappings, inductive data
(~ 2.5 weeks)
- 2: ***Machine***-based computing from automata to Turing machines.
(~ 3 weeks)
- 3: ***Symbolic*** computing
Grammars, context-free languages (~ 2.5 weeks)

Structure of the course

- 1: **Data:**
Languages, sets, mappings, inductive data
(~ 2.5 weeks)
- 2: **Machine**-based computing from automata to Turing machines.
(~ 3 weeks)
- 3: **Symbolic** computing
Grammars, context-free languages (~ 2.5 weeks)
- 4: **Unity** of computing, the Turing Thesis.
(~ 2 weeks)

Structure of the course

- 1: **Data:**
Languages, sets, mappings, inductive data
(~ 2.5 weeks)
- 2: **Machine**-based computing from automata to Turing machines.
(~ 3 weeks)
- 3: **Symbolic** computing
Grammars, context-free languages (~ 2.5 weeks)
- 4: **Unity** of computing, the Turing Thesis.
(~ 2 weeks)
- 5: **limits** of computability.
Decidable and semi-decidable problems (~ 2 weeks)

Structure of the course

- 1: **Data:**
Languages, sets, mappings, inductive data
(~ 2.5 weeks)
- 2: **Machine**-based computing from automata to Turing machines.
(~ 3 weeks)
- 3: **Symbolic** computing
Grammars, context-free languages (~ 2.5 weeks)
- 4: **Unity** of computing, the Turing Thesis.
(~ 2 weeks)
- 5: **limits** of computability.
Decidable and semi-decidable problems (~ 2 weeks)
- 6: Computational **complexity**.
(~ 3 weeks)