Logical and Theoretical Foundations of Computer Science

LaTFoCS

Pamela Fleischmann

fpa@informatik.uni-kiel.de

Winter Semester 2019

Kiel University

Dependable Systems Group

Why logic?

$\lambda o \gamma \iota \kappa \eta \quad \tau \varepsilon \chi \nu \eta$

Why logic?

$\lambda o \gamma \iota \kappa \eta \quad \tau \varepsilon \chi \nu \eta$

$\lambda o \gamma \iota \kappa \eta$ = thinking, reasoning (older: spoken word)

Why logic?

$\lambda o \gamma \iota \kappa \eta \quad \tau \varepsilon \chi \nu \eta$

$\lambda o \gamma \iota \kappa \eta$ = thinking, reasoning (older: spoken word)

$\tau \varepsilon \chi \nu \eta$ = technique, science, art

Why logic?

$\lambda o \gamma \iota \kappa \eta \quad \tau \varepsilon \chi \nu \eta$

$\lambda o \gamma \iota \kappa \eta$ = thinking, reasoning (older: spoken word)

$\tau \varepsilon \chi \nu \eta$ = technique, science, art

The Art of Thinking

Convincing not Persuading

Rhetorica ad Herennium (~ 80 B.C)

- Exordium: use relevant generalities, connect them to the specific topic

Convincing not Persuading

Rhetorica ad Herennium (~ 80 B.C)

- Exordium: use relevant generalities, connect them to the specific topic
- Narratio: state the argument that has to be proven

Convincing not Persuading

Rhetorica ad Herennium (~ 80 B.C)

- Exordium: use relevant generalities, connect them to the specific topic
- Narratio: state the argument that has to be proven
- Divisio: divide into main points

Convincing not Persuading

Rhetorica ad Herennium (~ 80 B.C)

- Exordium: use relevant generalities, connect them to the specific topic
- Narratio: state the argument that has to be proven
- Divisio: divide into main points
- Confirmatio: set out arguments as well as evidence

Convincing not Persuading

Rhetorica ad Herennium (~ 80 B.C)

- Exordium: use relevant generalities, connect them to the specific topic
- Narratio: state the argument that has to be proven
- Divisio: divide into main points
- Confirmatio: set out arguments as well as evidence
- Refutatio: refute opposing arguments

Convincing not Persuading

Rhetorica ad Herennium (~ 80 B.C)

- Exordium: use relevant generalities, connect them to the specific topic
- Narratio: state the argument that has to be proven
- Divisio: divide into main points
- Confirmatio: set out arguments as well as evidence
- Refutatio: refute opposing arguments
- Conclusio: summary of the argument

Syllogisms

The mental health of the clones in the BBC TV-Series *Orphan Black* is tested with questions of the following kind:

- Some bags are pockets.
  No pocket is a pouch.
  Conclusion: All bags are not pouches.

Syllogisms

The mental health of the clones in the BBC TV-Series *Orphan Black* is tested with questions of the following kind:

- Some bags are pockets.
  No pocket is a pouch.
  Conclusion: All bags are not pouches. false

Syllogisms

The mental health of the clones in the BBC TV-Series *Orphan Black* is tested with questions of the following kind:

- Some bags are pockets.
  No pocket is a pouch.
  Conclusion: All bags are not pouches. false
- Some pigs are predators.
  No predator is a pet.
  Conclusion: Some pigs are not pets.

Syllogisms

The mental health of the clones in the BBC TV-Series *Orphan Black* is tested with questions of the following kind:

- Some bags are pockets.
  No pocket is a pouch.
  Conclusion: All bags are not pouches. false
- Some pigs are predators.
  No predator is a pet.
  Conclusion: Some pigs are not pets. true

Syllogisms

The mental health of the clones in the BBC TV-Series *Orphan Black* is tested with questions of the following kind:

- Some bags are pockets.
  No pocket is a pouch.
  Conclusion: All bags are not pouches. false
- Some pigs are predators.
  No predator is a pet.
  Conclusion: Some pigs are not pets. true
- Some maggots are flies.
  No fly is welcome.
  Conclusion: No maggots are welcome.

Syllogisms

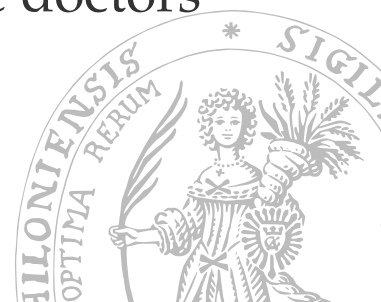
The mental health of the clones in the BBC TV-Series *Orphan Black* is tested with questions of the following kind:

- Some bags are pockets.
  No pocket is a pouch.
  Conclusion: All bags are not pouches. false
- Some pigs are predators.
  No predator is a pet.
  Conclusion: Some pigs are not pets. true
- Some maggots are flies.
  No fly is welcome.
  Conclusion: No maggots are welcome. false

Syllogisms

The mental health of the clones in the BBC TV-Series *Orphan Black* is tested with questions of the following kind:

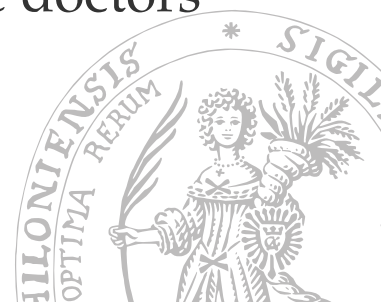
- Some bags are pockets.
  No pocket is a pouch.
  Conclusion: All bags are not pouches. false
- Some pigs are predators.
  No predator is a pet.
  Conclusion: Some pigs are not pets. true
- Some maggots are flies.
  No fly is welcome.
  Conclusion: No maggots are welcome. false
- Some doctors are fools.
  All fools are rich.
  Conclusion: Some doctors are rich.



Syllogisms

The mental health of the clones in the BBC TV-Series *Orphan Black* is tested with questions of the following kind:

- Some bags are pockets.
  No pocket is a pouch.
  Conclusion: All bags are not pouches. false
- Some pigs are predators.
  No predator is a pet.
  Conclusion: Some pigs are not pets. true
- Some maggots are flies.
  No fly is welcome.
  Conclusion: No maggots are welcome. false
- Some doctors are fools.
  All fools are rich.
  Conclusion: Some doctors are rich. true



... and why in Computer Science?

automated conclusions in

- autonomous driving
- meteorology
- washing machines
- medicine

How to transform our knowledge into Computers?

1. we need a formalism to describe it (formulae)
2. we need a formalism how to interprete it (interpretation, models)
3. we need some ensurance that everything works as intended
   - either formula or negation are true (Consistency)
   - everything provable is really true (Soundness)
   - everything true is provable (Completeness)
   - the axioms are independent of each other (Indepence)

What are we talking about?

- We know what we have to take care of in transforming *thinking* to computers?

What are we talking about?

- We know what we have to take care of in transforming *thinking* to computers?
- But what are computers?

What are we talking about?

- We know what we have to take care of in transforming *thinking* to computers?
- But what are computers?
- What is computation?

What are we talking about?

- We know what we have to take care of in transforming *thinking* to computers?
- But what are computers?
- What is computation?
- As decently as we look into logic, we have to look into the definition of computation!

What is computation?

Definition (Computation (Oxford Dictionary))

1.
   - (mass noun) The action of mathematical calculation.
   - (count noun) ‘statistical computations’
2. The use of computers, especially as a subject of research or study.

Is everything computable?

```
i=1;
while TRUE
   i++;
```

```
i=1;
while i<5
   i++;
```

Is everything computable?

```
i=1;
while TRUE
  i++;
```

```
i=1;
while i<5
  i++;
```

Goal: Program P_1 with input P_2 that decides whether P_2 terminates.

Is everything computable?

```
i=1;
while TRUE
  i++;
```

```
i=1;
while i<5
  i++;
```

Goal: Program P_1 with input P_2 that decides whether P_2 terminates. $\hat{=}$ Halt-Problem

Is everything computable?

```
i=1;
while TRUE
  i++;
```

```
i=1;
while i<5
  i++;
```

Goal: Program P_1 with input P_2 that decides whether P_2 terminates. $\hat{=}$ Halt-Problem

- Turing: Halt-Problem is undecidable!

Theory of Computation

We have to investigate what computation means in detail!

- Which kinds of problems do we have?
- How to encode problems?
- What is the definition of computable/decidable?

Problems and Encoding

Different Kinds of Problems

Holidays: car, petrol and now?

Different Kinds of Problems

Holidays: car, petrol and now?

- Can we reach Hamburg?

Different Kinds of Problems

Holidays: car, petrol and now?

- Can we reach Hamburg? Decision-Problem

Different Kinds of Problems

Holidays: car, petrol and now?

- Can we reach Hamburg? Decision-Problem
- How many kilometers can we drive?

Different Kinds of Problems

Holidays: car, petrol and now?

- Can we reach Hamburg? Decision-Problem
- How many kilometers can we drive? Optimisation-Problem
- Where can we drive?

Different Kinds of Problems

Holidays: car, petrol and now?

- Can we reach Hamburg? Decision-Problem
- How many kilometers can we drive? Optimisation-Problem
- Where can we drive? Search-Problem

Different Kinds of Problems

Holidays: car, petrol and now?

- Can we reach Hamburg? Decision-Problem
- How many kilometers can we drive? Optimisation-Problem
- Where can we drive? Search-Problem

Here only Decision-Problems!

Decision-Problem

informal:

- Output: **yes**, **no** (resp. 0/1 or true/false)

Decision-Problem

informal:

- Output: **yes**, **no** (resp. 0/1 or true/false)
- Input: Instances of the problem (somehow coded)

Decision-Problem

informal:

- Output: **yes**, **no** (resp. 0/1 or true/false)
- Input: Instances of the problem (somehow coded)

Definition (Decision Problem)

tuple (A, B, Σ) with

Decision-Problem

informal:

- Output: **yes**, **no** (resp. 0/1 or true/false)
- Input: Instances of the problem (somehow coded)

Definition (Decision Problem)

tuple (A, B, Σ) with

- alphabet for encoding Σ

Decision-Problem

informal:

- Output: **yes**, **no** (resp. 0/1 or true/false)
- Input: Instances of the problem (somehow coded)

Definition (Decision Problem)

tuple (A, B, Σ) with

- alphabet for encoding Σ
- set of all possible inputs $A \subseteq \Sigma^*$

Decision-Problem

informal:

- Output: **yes**, **no** (resp. 0/1 or true/false)
- Input: Instances of the problem (somehow coded)

Definition (Decision Problem)

tuple (A, B, Σ) with

- alphabet for encoding Σ
- set of all possible inputs $A \subseteq \Sigma^*$
- set of all *yes*-instances $B \subseteq A$

Course Outline

- Propositional Logic
- Application to *real world* proofs.
- Theory of Computation: Chomsky Hierarchy
- Predicate Logic
- Application to *real world* proofs.
- Complexity Theory: P, NP, NPC