

# Discrete Mathematics in Computer Science

## E1. Syntax and Semantics of Propositional Logic

Malte Helmert, Gabriele Röger

University of Basel

# Discrete Mathematics in Computer Science

## — E1. Syntax and Semantics of Propositional Logic

### E1.1 Introduction to Formal Logic

### E1.2 Syntax of Propositional Logic

### E1.3 Semantics of Propositional Logic

# E1.1 Introduction to Formal Logic

# Why Logic?

- ▶ formalizing mathematics
  - ▶ What is a true statement?
  - ▶ What is a valid proof?
- ▶ basis of many tools in computer science
  - ▶ design of digital circuits
  - ▶ semantics of databases; query optimization
  - ▶ meaning of programming languages
  - ▶ verification of safety-critical hardware/software
  - ▶ knowledge representation in artificial intelligence
  - ▶ logic-based programming languages (e.g. Prolog)
  - ▶ ...

# Application: Logic Programming I

**Declarative approach:** Describe **what** to accomplish,  
**not how** to accomplish it.

## Example (Map Coloring)

Color each region in a map with a limited number of colors so that no two adjacent regions have the same color.



CC BY-SA 3.0 Wikimedia Commons (TUBS)

This is a hard problem!

# Application: Logic Programming II

## Prolog program

```
color(red). color(blue). color(green). color(yellow).

neighbor(StateAColor, StateBColor) :-
    color(StateAColor), color(StateBColor),
    StateAColor \= StateBColor.

switzerland(AG, AI, AR, BE, BL, BS, FR, GE, GL, GR,
            JU, LU, NE, NW, OW, SG, SH, SO, SZ, TG,
            TI, UR, VD, VS, ZG, ZH) :-
    neighbor(AG, BE), neighbor(AG, BL), neighbor(AG, LU),
    ...
    neighbor(UR, VS), neighbor(VD, VS), neighbor(ZH, ZG).
```

# What Logic is About

## General Question:

- ▶ Given some knowledge about the world (a **knowledge base**)
- ▶ what can we **derive** from it?
- ▶ And on what basis may we argue?

↪ **logic**

## Goal: “mechanical” proofs

- ▶ formal “game with letters”
- ▶ detached from a concrete meaning

# Task

What's the secret of your long life?



I am on a strict diet: If I don't drink beer to a meal, then I always eat fish. Whenever I have fish and beer with the same meal, I abstain from ice cream. When I eat ice cream or don't drink beer, then I never touch fish.

Simplify this advice!

Exercise from U. Schöning: Logik für Informatiker  
Picture courtesy of graur razvan ionut / FreeDigitalPhotos.net



# Propositional Logic

**Propositional logic** is a simple logic without numbers or objects.

Building blocks of propositional logic:

- ▶ **propositions** are statements that can be either true or false
- ▶ **atomic propositions** cannot be split into sub-propositions
- ▶ **logical connectives** connect propositions to form new ones

**German:** Aussagenlogik, Aussage, atomare Aussage, Junktoren

# Examples for Building Blocks



If I don't **drink beer** to a meal, then I always **eat fish**. Whenever I **have fish** and **beer** with the same meal, I abstain from **ice cream**. When I **eat ice cream** or don't **drink beer**, then I never touch **fish**.

- ▶ Every sentence is a proposition that consists of sub-propositions (e. g., “eat ice cream or don't drink beer”).
- ▶ atomic propositions “**drink beer**”, “**eat fish**”, “**eat ice cream**”
- ▶ logical connectives “and”, “or”, negation, “if, then”

Exercise from U. Schöning: Logik für Informatiker  
Picture courtesy of graur razvan ionut / FreeDigitalPhotos.net

# Examples for Building Blocks



If I don't drink beer to a meal, then I always eat fish. Whenever I have fish and beer with the same meal, I abstain from ice cream. When I eat ice cream or don't drink beer, then I never touch fish.

- ▶ Every sentence is a proposition that consists of sub-propositions (e. g., “eat ice cream or don't drink beer”).
- ▶ atomic propositions “drink beer”, “eat fish”, “eat ice cream”
- ▶ logical connectives “and”, “or”, negation, “if, then”

Exercise from U. Schöning: Logik für Informatiker  
Picture courtesy of graur razvan ionut / FreeDigitalPhotos.net

# Problems with Natural Language



If I don't drink beer **to a meal**, then I **always** eat fish.

Whenever I have fish and beer **with the same meal**, I abstain from ice cream.

When I eat ice cream or don't drink beer, then I never touch fish.

► “irrelevant” information

Exercise from U. Schöning: Logik für Informatiker  
Picture courtesy of graur razvan ionut / FreeDigitalPhotos.net

# Problems with Natural Language



If I **don't** drink beer, then I eat fish.  
Whenever I have fish and beer, I **abstain**  
from ice cream.  
When I eat ice cream or **don't** drink  
beer, then I **never** touch fish.

- ▶ “irrelevant” information
- ▶ **different formulations for the same connective/proposition**

Exercise from U. Schöning: Logik für Informatiker  
Picture courtesy of graur razvan ionut / FreeDigitalPhotos.net

# Problems with Natural Language



If not DrinkBeer, then EatFish.  
If EatFish and DrinkBeer,  
then not EatIceCream.  
If EatIceCream or not DrinkBeer,  
then not EatFish.

- ▶ “irrelevant” information
- ▶ different formulations for the same connective/proposition

Exercise from U. Schöning: Logik für Informatiker  
Picture courtesy of graur razvan ionut / FreeDigitalPhotos.net

# What is Next?

- ▶ What are meaningful (well-defined) sequences of atomic propositions and connectives?  
“if then EatIceCream not or DrinkBeer and” not meaningful  
→ **syntax**
- ▶ What does it mean if we say that a statement is true?  
Is “DrinkBeer and EatFish” true?  
→ **semantics**
- ▶ When does a statement logically follow from another?  
Does “EatFish” follow from “if DrinkBeer, then EatFish”?  
→ **logical entailment**

**German:** Syntax, Semantik, logische Folgerung

## E1.2 Syntax of Propositional Logic



# Syntax of Propositional Logic

## Definition (Syntax of Propositional Logic)

Let  $A$  be a set of **atomic propositions**. The set of **propositional formulas** (over  $A$ ) is inductively defined as follows:

- ▶ Every **atom**  $a \in A$  is a propositional formula over  $A$ .
- ▶ If  $\varphi$  is a propositional formula over  $A$ , then so is its **negation**  $\neg\varphi$ .
- ▶ If  $\varphi$  and  $\psi$  are propositional formulas over  $A$ , then so is the **conjunction**  $(\varphi \wedge \psi)$ .
- ▶ If  $\varphi$  and  $\psi$  are propositional formulas over  $A$ , then so is the **disjunction**  $(\varphi \vee \psi)$ .

The **implication**  $(\varphi \rightarrow \psi)$  is an abbreviation for  $(\neg\varphi \vee \psi)$ .

The **biconditional**  $(\varphi \leftrightarrow \psi)$  is an abbrev. for  $((\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi))$ .

**German:** atomare Aussage, aussagenlogische Formel, Atom, Negation, Konjunktion, Disjunktion, Implikation, Bikonditional

## Syntax: Examples

Which of the following sequences of symbols are propositional formulas over the set of all possible letter sequences? Which kinds of formula are they (atom, conjunction, ...)?

- ▶  $(A \wedge (B \vee C))$
- ▶  $((\text{EatFish} \wedge \text{DrinkBeer}) \rightarrow \neg \text{EatIceCream})$
- ▶  $\neg(\wedge \text{Rain} \vee \text{StreetWet})$
- ▶  $\neg(\text{Rain} \vee \text{StreetWet})$
- ▶  $\neg(A = B)$
- ▶  $(A \wedge \neg(B \leftrightarrow)C)$
- ▶  $(A \vee \neg(B \leftrightarrow C))$
- ▶  $((A \leq B) \wedge C)$
- ▶  $((A_1 \wedge A_2) \vee \neg(A_3 \leftrightarrow A_2))$

# E1.3 Semantics of Propositional Logic

# Meaning of Propositional Formulas?

So far propositional formulas are only symbol sequences without any meaning.

For example, what does this mean:

$((\text{EatFish} \wedge \text{DrinkBeer}) \rightarrow \neg \text{EatIceCream})?$

▷ **We need semantics!**

# Semantics of Propositional Logic

## Definition (Semantics of Propositional Logic)

A **truth assignment** (or **interpretation**) for a set of atomic propositions  $A$  is a function  $\mathcal{I} : A \rightarrow \{0, 1\}$ .

A propositional **formula**  $\varphi$  (over  $A$ ) **holds under**  $\mathcal{I}$  (written as  $\mathcal{I} \models \varphi$ ) according to the following definition:

|   |     |  |                  |
|---|-----|--|------------------|
| $\mathcal{I} \models a$                     | iff | $\mathcal{I}(a) = 1$   | (for $a \in A$ ) |
| $\mathcal{I} \models \neg\varphi$           | iff | not $\mathcal{I} \models \varphi$                            |                  |
| $\mathcal{I} \models (\varphi \wedge \psi)$ | iff | $\mathcal{I} \models \varphi$ and $\mathcal{I} \models \psi$ |                  |
| $\mathcal{I} \models (\varphi \vee \psi)$   | iff | $\mathcal{I} \models \varphi$ or $\mathcal{I} \models \psi$  |                  |

**Question:** should we define semantics of  $(\varphi \rightarrow \psi)$  and  $(\varphi \leftrightarrow \psi)$ ?

**German:** Wahrheitsbelegung/Interpretation,  $\varphi$  gilt unter  $\mathcal{I}$

# Semantics of Propositional Logic: Terminology

- ▶ For  $\mathcal{I} \models \varphi$  we also say  $\mathcal{I}$  is a model of  $\varphi$  and that  $\varphi$  is true under  $\mathcal{I}$ .
- ▶ If  $\varphi$  does not hold under  $\mathcal{I}$ , we write this as  $\mathcal{I} \not\models \varphi$  and say that  $\mathcal{I}$  is no model of  $\varphi$  and that  $\varphi$  is false under  $\mathcal{I}$ .
- ▶ **Note:**  $\models$  is not part of the formula but part of the meta language (speaking about a formula).

**German:**  $\mathcal{I}$  ist ein/kein Modell von  $\varphi$ ;  $\varphi$  ist wahr/falsch unter  $\mathcal{I}$ ;  
**Metasprache**

## Exercise

Consider set  $A = \{X, Y, Z\}$  of atomic propositions and formula  $\varphi = (X \wedge \neg Y)$ .

Specify an interpretation  $\mathcal{I}$  for  $A$  with  $\mathcal{I} \models \varphi$ .

## Semantics: Example (1)

$$A = \{\text{DrinkBeer}, \text{EatFish}, \text{EatIceCream}\}$$

$$\mathcal{I} = \{\text{DrinkBeer} \mapsto 1, \text{EatFish} \mapsto 0, \text{EatIceCream} \mapsto 1\}$$

$$\varphi = (\neg \text{DrinkBeer} \rightarrow \text{EatFish})$$

Do we have  $\mathcal{I} \models \varphi$ ?



## Semantics: Example (2)

**Goal:** prove  $\mathcal{I} \models \varphi$ .

Let us use the definitions we have seen:

$$\begin{aligned}\mathcal{I} \models \varphi &\text{ iff } \mathcal{I} \models (\neg\text{DrinkBeer} \rightarrow \text{EatFish}) \\ &\text{ iff } \mathcal{I} \models (\neg\neg\text{DrinkBeer} \vee \text{EatFish}) \\ &\text{ iff } \mathcal{I} \models \neg\neg\text{DrinkBeer} \text{ or } \mathcal{I} \models \text{EatFish}\end{aligned}$$

This means that if we want to prove  $\mathcal{I} \models \varphi$ , it is sufficient to prove

$$\mathcal{I} \models \neg\neg\text{DrinkBeer}$$

or to prove

$$\mathcal{I} \models \text{EatFish}.$$

We attempt to prove the first of these statements.

## Semantics: Example (3)

**New goal:** prove  $\mathcal{I} \models \neg\neg\text{DrinkBeer}$ .

We again use the definitions:

$$\begin{aligned}\mathcal{I} \models \neg\neg\text{DrinkBeer} &\text{ iff not } \mathcal{I} \models \neg\text{DrinkBeer} \\ &\text{ iff not not } \mathcal{I} \models \text{DrinkBeer} \\ &\text{ iff } \mathcal{I} \models \text{DrinkBeer} \\ &\text{ iff } \mathcal{I}(\text{DrinkBeer}) = 1\end{aligned}$$

The last statement is true for our interpretation  $\mathcal{I}$ .

To write this up as a **proof** of  $\mathcal{I} \models \varphi$ ,  
we can go through this line of reasoning back-to-front,  
starting from our assumptions and ending with the conclusion  
we want to show.

## Semantics: Example (4)

Let  $\mathcal{I} = \{\text{DrinkBeer} \mapsto 1, \text{EatFish} \mapsto 0, \text{EatIceCream} \mapsto 1\}$ .

**Proof** that  $\mathcal{I} \models (\neg\text{DrinkBeer} \rightarrow \text{EatFish})$ :

- ① We have  $\mathcal{I} \models \text{DrinkBeer}$   
(uses defn. of  $\models$  for atomic props. and fact  $\mathcal{I}(\text{DrinkBeer}) = 1$ ).
- ② From (1), we get  $\mathcal{I} \not\models \neg\text{DrinkBeer}$   
(uses defn. of  $\models$  for negations).
- ③ From (2), we get  $\mathcal{I} \models \neg\neg\text{DrinkBeer}$   
(uses defn. of  $\models$  for negations).
- ④ From (3), we get  $\mathcal{I} \models (\neg\neg\text{DrinkBeer} \vee \psi)$  for all formulas  $\psi$ ,  
in particular  $\mathcal{I} \models (\neg\neg\text{DrinkBeer} \vee \text{EatFish})$   
(uses defn. of  $\models$  for disjunctions).
- ⑤ From (4), we get  $\mathcal{I} \models (\neg\text{DrinkBeer} \rightarrow \text{EatFish})$   
(uses defn. of “ $\rightarrow$ ”). □

# Summary

- ▶ **propositional logic** based on atomic propositions
- ▶ **syntax** defines what well-formed formulas are
- ▶ **semantics** defines when a formula is true
- ▶ **interpretations** are the basis of semantics