Prof. Dr. Jasmin Blanchette
Dr. Martin Desharnais-Schäfer
Dr. Michael Kirsten

Ludwig-Maximilians-Universität München
Institut für Informatik
Discussion on 23.12.2026
Homework due on 13.01.2027 at 14:15

Exercise Sheet 9 in

Scientific and Technical English

WiSe 2026/27

The exercise sheets consist of in-class exercises and homework. The in-class exercises take place during the second half of the lecture time slots. The homework, which is optional and ungraded, can be submitted via the "Homework" section in Moodle. The homework is subject to peer review.

Unless indicated otherwise, generative artificial intelligence assistants such as ChatGPT may be used, as long as you acknowledge how you use them as specified by the Institute's policy on plagiarism.[1] However, you may not use such tools to generate peer reviews for you. In addition, we strongly recommend that you do not use them to generate entire solutions, since that would defeat the purpose of the exercises.

In-class exercise 9-1 ***Citations*** Check and, if necessary, correct the following passages to use accurate and stylistically correct citations and quotations.

a) Friedrich Gödel pointed out that for every finite subsystem to be satisfiable, it is necessary and sufficient that it is a countably infinite system of formulas. [**godelWrong**]

b) In computability theory, Rice's theorem states that all nontrivial semantic properties of programs are undecidable [**riceWiki**].

c) In the seminal work "On computable numbers, with an application to the entscheidungsproblem," Alan Turing defined a fundamental machine model that he called after himself as *Turing machine* [**turingMachineWrong**]. He also gave an influential definition of artificial intelligence within his paper "computing machinery and intelligence" by defining a so-called *animation game* [**turingTestWrong**].

[1] `https://www.medien.ifi.lmu.de/lehre/Plagiate-IfI.pdf`

In-class exercise 9-2 ***Over and Under*** The following passages both overcite and undercite. Remove all superfluous citations, and mark any parts that lack citations but should have them. (You are not asked to find the sources to cite.)

a) Abstract argumentation frameworks (AAFs) **[dung]** are formal systems that represent conflicting pieces of information as a set **[cantor]** of arguments, and a binary attack relation. In recent years, AAFs have been widely used for a variety of different tasks in knowledge representation and artificial intelligence systems **[karacapilidis]**. Several extensions that have been proposed for AAFs have been presented, among them Bipolar Argumentation Frameworks **[cayrol2005acceptability]**, which include a binary support relation in addition to the attack relation.

b) A refinement type is a kind of dependent type that "refines" an existing type by use of a predicate that needs to hold for every value to be admitted to the refined type. The concept was first introduced by Freeman and Pfenning in 1991 **[ML]**, who presented a refinement type system for a subset of Standard ML **[SMLExists]**. Since then, refinement type systems have been developed for languages such as Haskell **[Haskell]**, TypeScript **[TypeScript]**, Rust **[Rust, Rust2]**, and Scala.

Homework 9-3 ***Cite and Quote*** Write a paragraph of at most 200 words explaining generics in the Java programming language. Include at least one citation, one naming of authors, one reference, and one quotation. Potential reference material includes the following:

- *The Java® Language Specification*[2]
- *The Java™ Tutorials: Generics*[3]
- *Java® Platform, Standard Edition & Java Development Kit*[4]

Homework 9-4 ***Explaining a Topic*** For each of the following two papers, write a text of about 250 words that explains the topic using appropriate citations and quotations to acknowledge the paper as your source. You will probably not need to do a close reading of the papers.

a) **medieninf**

b) **mapreduce**

[2] `https://docs.oracle.com/javase/specs/jls/se21/html/index.html`
[3] `https://docs.oracle.com/javase/tutorial/extra/generics/index.html`
[4] `https://docs.oracle.com/en/java/javase/21/docs/api/index.html`