

# Homework: language implementation

Programming Languages in Software Engineering at Constructor University Bremen

Maximum points: 20

Deadline: 19 October, 23:59

In this homework, you will define the basic components of your language:

- Parser
- Pretty-printer
- Type inference
- Evaluator
- Standard library
- Test suite

Note that the test suite is essential: each part will be graded based on the tests you write for it.

If you would like to deviate from the requirements, reach out to me and we can discuss it.

**Exercise 1 (6 pts):** Implement a parser and pretty-printer for your language. There are three requirements:

- Valid ASTs can be pretty-printed.
- Pretty-printing an AST and then parsing it gives the same result.

**Exercise 2 (4 pts):** Write a type inference engine that annotates every expression with its type.

**Exercise 3 (6 pts):** Write an evaluator that takes a fully typed AST and runs it.

**Exercise 4 (4 pts):** Implement the following standard library functions:

- `read_char : () -> Int`
- `write_char : (Int) -> Unit`

And the following operators:

- Arithmetic operators for integers and floating point numbers.
- Comparison operators for integers, floating point numbers, and booleans.
- Logical operators for booleans.
- A “size of outermost dimension” operator for arrays.
- A concatenation operator for arrays.
  - You may want to consider several versions of this, for different kinds of gluing.

This assignment is in some ways an experiment to measure the speed increase of coding with modern tools. If you find that you cannot complete the homework by the deadline, reach out to me and we’ll discuss how to restructure it. However, I would prefer to move on past the basics as soon as possible.