Functional language compiler to WebAssembly - Supervisor meeting

This session was a catch-up on yesterday's session, which didn't take place. The working version of the compiler was shown and discussed to the supervisor.

Location: C10.05 HEIA-FR

Date: 20.06.2024 10:00 - 11:00

Agenda

- Discuss yesterday's absence
- Show working version of the compiler
- Discuss the next steps

Participants

- Jacques Supcik (supervisor)
- Noah Godel (student)

Information shared

- From a cursory overview of the report, the structure of the document is good. However, the supervisor will provide more detailed feedback in the next meeting.
- Most computer scientists are not familiar with functional programming languages, so the important concepts should be explained in the analysis section.
- The design section should contain code examples of how the language can be used and how the different features are used.
- Since the standard library will be implemented in the language itself, the tests could be on compiled code, which would be a good way to test the compiler since all the language features would be used.
- The `wasm-merge` tool is a good way to merge the different WebAssembly modules into a single file and its use in the compiler is appropriate.

Decisions

- The glossary elements should not be underlined in the report since it interferes with the readability.
- The analysis section should contain a chapter on functional language features, such as lazy evaluation (and its implementation with the call graph), immutability, and higher-order functions since not all readers might be familiar with these concepts.
- The analysis section should also mention other projects that compile functional languages to WebAssembly, such as Asterius and Wisp.

- The testing of the compiler should be done on the compiled code of the standard library, which will be written in the language itself.
- Next week, the compiler should have the final set of features implemented, after which the focus will be on testing, the standard library, and writing the report.

Tasks

What?	Who?	Deadline
Implement the final set of features	Mr. Godel	26.06.2024
Make corrections to the report	Mr. Godel	26.06.2024
Write the standard library	Mr. Godel	28.06.2024
Write the tests on the compiled code	Mr. Godel	01.07.2024