Functional language compiler to WebAssembly - Supervisor meeting

The quasi final version of the compiler was presented.

Location: C10.15 HEIA-FR

Date: 03.07.2024 08:45 - 09:30

Agenda

- Demonstration of the compiler
- Feedback on the report
- Next steps

Participants

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

Information shared

- The code generation of the compiler for a functional programming language is more complex than for an imperative programming language. The main reason is that the functional programming language has more complex features like partial application, lazy evaluation, currying, and pattern matching. The code generation for these features is more complex than for the imperative programming language.
- The project should mention how it aims to complete at lest one of the 17 UNESCO sustainability goals.
- Some errors were discovered during the implementation of the standard library. These errors were due to unhandled edge cases. So the implementation of compiler is not yet finished. It should however be finished by tomorrow.
- It would be interesting to add a section that explains what the student would do differently if he had to start the project again. Especially considering the discovery of errors due to unhandled edge cases during the implementation of the standard library.

Decisions

- Add a section to the report about the challenges of code generation for a functional programming language compared to an imperative programming language.
- Add a section in the report that explains what the student would do differently if he had to start the project again. Especially considering the

discovery of errors due to unhandled edge cases during the implementation of the standard library.
- Add a section in the report that explains how the project aims to complete at least one of the 17 UNESCO sustainability goals.

Tasks

What?	Who?	Deadline
Finish implementation of compiler and the standard library	Mr. Godel	03.07.2024