A Computer-Checked Library of Category Theory

Functors and F-Coalgebras

INTRODUCTION

CONDENSED ABSTRACT

This project develops a computer-checked library of category theory in Lean, focusing on functional programming concepts. It explores final coalgebras for understanding infinite data structures. The formalized library enhances our understanding of functional programming and provides a rigorous foundation for program reasoning. It benefits researchers and practitioners in the field by offering a comprehensible resource for studying these topics.

BACKGROUND

Category theory is an area of mathematics that abstractly describes general concepts and phenomena across various fields of application, including computer science. It is used as a way to understand concepts abstractly, without being overwhelmed by "implementation details".

Method

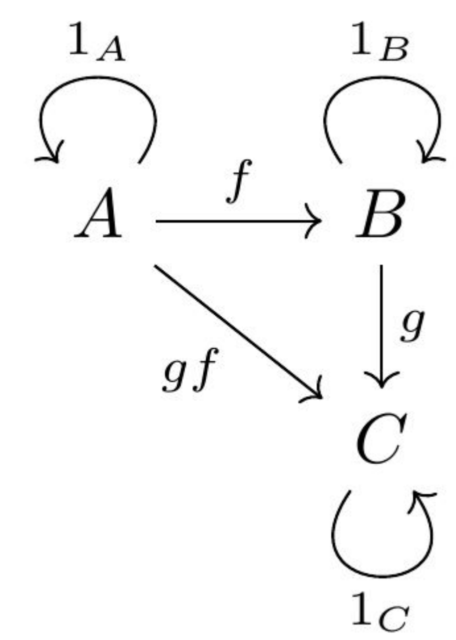
Proof assistants are crucial for formal verification, ensuring the correctness of complex systems. They provide an interactive environment to construct and verify proofs, define specifications, and establish properties of mathematical objects or programs.

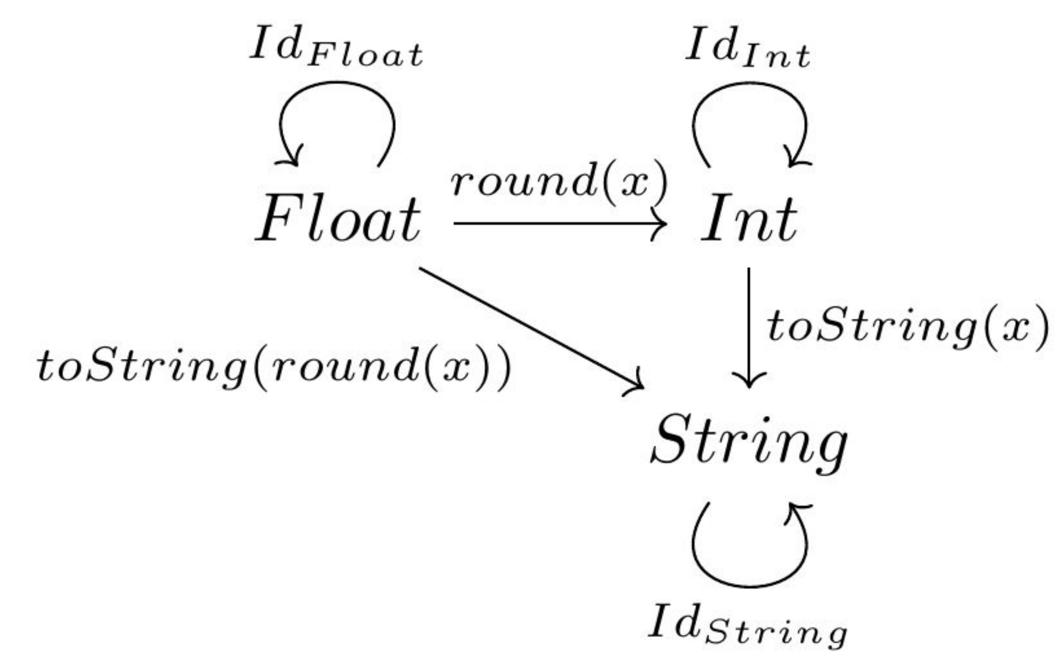
RESEARCH

Categories

A category consists of:

- Objects
- Morphisms
- Composition
  - Identities
  - Associativity

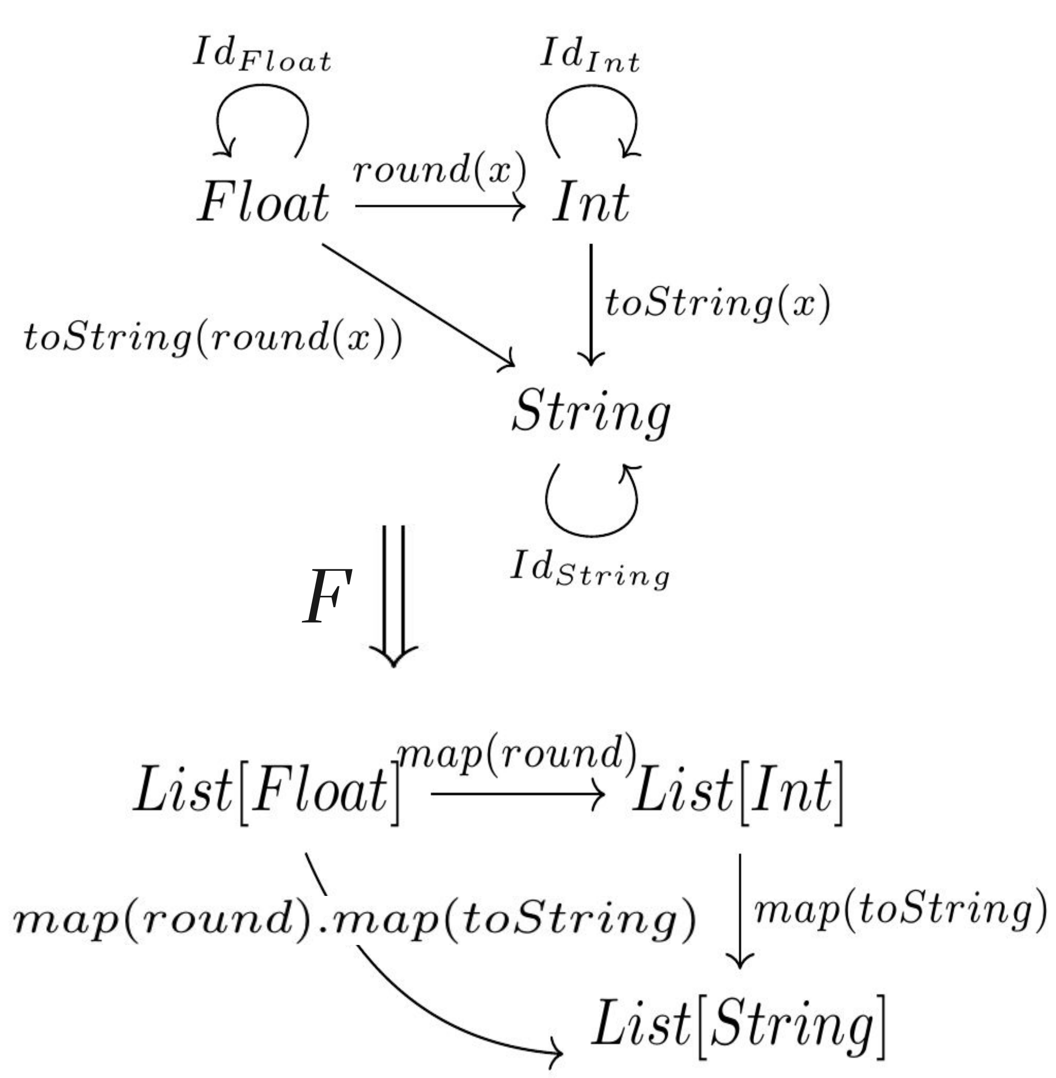




Functors

A functors consists of:

- A mapping of all objects
- A mapping of all morphisms
- Preserve:
  - Identities
  - Compositions

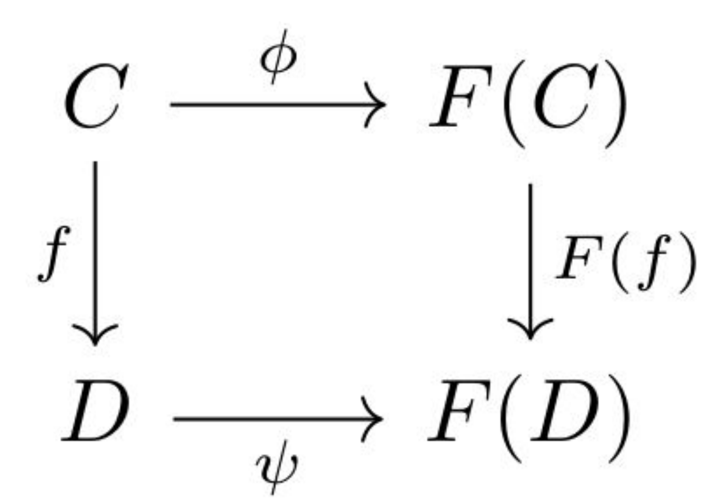


F-Coalgebras

An F -coalgebra consists of data:

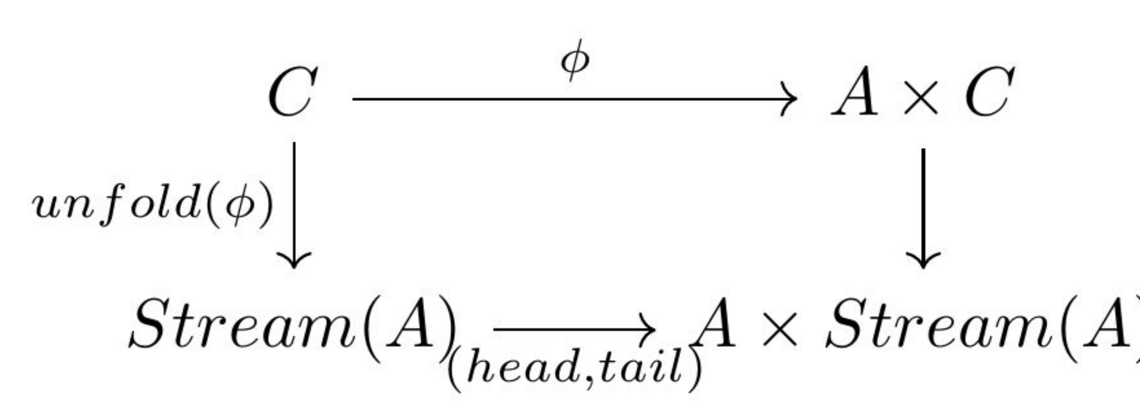
- An object C ∈ C.
- A morphism ϕ ∈ C(C, F (C)).

A homomorphism from (C, ϕ) to (D, ψ) is a morphism f such that the following diagram commutes:



The final F-coalgebra is the terminal object in CoAlg(F). The unique morphism from an arbitrary F-coalgebra to the final coalgebra is called anamorphism.

Stream is the final coalgebra of Types and the unfold function is the anamorphism.



CONCLUSION

CONCLUSION

In conclusion, this research project successfully developed a computer-checked library of category theory within the Lean proof assistant.

We achieved the formalization of streams within our computer-checked library. Using coalgebras, we defined streams as final coalgebras in the category of sets. By establishing structure maps and proving their properties, we created a robust foundation for representing and manipulating streams .

Our library facilitates the exploration and understanding of category theory concepts in functional programming. With formal definitions, theorems, and programming examples, it serves as an educational resource for programmers and students. By leveraging the rigor of the Lean proof assistant, the library ensures accurate and valid definitions and theorems. Its emphasis on understandability and pedagogy makes it accessible for learners of category theory and its practical implications.

INFO

Author: Pedro H. Brandão de Araujo
Email: phbrandao1206@gmail.com
Supervisors: Benedikt Ahrens, Lucas Escot