

# Haskelinterpreters: Definitional Interpreters in Haskell

Xurxo Riesco (xr2154)

COMS E6998 Types, Languages & Compilers

Final Project Proposal

## **Interpreters**

The goal of this project is to study interpreters following Josh Reynolds's **Definitional interpreters for higher-order programming languages**, in which the study of the interpreters is divided upon two basic characteristics, the use of high order functions and the order-of-application dependence, both of which can be combined to achieve four different kinds of interpreters, as shown in the image below taken from the paper itself.

Order-of-application dependence:	Use of higher-order functions:	
	yes	no
yes	direct interpreter for GEDANKEN	McCarthy's definition of LISP
no	Morris-Wadsworth method	SECD machine, Vienna definition

## **Implementation**

My hope is to implement the four interpreters separately in Haskell and study the differences between them and how they derive into each other. Another goal is to compare my own implementations with Eric Campbell's implementation of the interpreters derived from the same paper.

## **Continuation**

The paper was written in 1972 and discusses the limitations the approach suggested carries in the implementation of functional languages. Another goal of the project is to study if the implementation of functional languages has evolved beyond the approach presented and if said limitations, relating to important issues such as type checking, are still present with more modern approaches.

## **References**

1. <https://dl.acm.org/doi/10.1145/800194.805852>
2. <https://github.com/ericthewry/interpreters>