|  |
| --- |
| Proposal for the Development of a Differential Calculator and Accompanying Extensible Language |
| The Final Project of Neale & Thomas |
| Neale Petrillo & Thomas Tharp |

Contents

[Introduction 3](#_Toc244661956)

[Language Features and Structure 3](#_Toc244661957)

[Features 3](#_Toc244661958)

[Structure 3](#_Toc244661959)

[Examples 3](#_Toc244661960)

# Introduction

The authors propose to develop a calculator for differential calculus and an accompanying extensible language based on the Scheme programming language.

# Language Features and Structure

## Features

The language accompanying the actual calculus calculator must be robust, simple, and easily extensible to all areas on mathematics related to calculus. It should, therefore, have the following features:

* Recognizable – the language must be similar in style to common ways of inputting mathematical expressions into computers. There should be little to have to learn in order to use the proposed language and associated calculator.
* Extensible – the calculus can be applied to virtually all areas of mathematics and “hard” sciences. There are many, many rules, functions, and processes the majority of which the authors could not hope to implement. Thus, it is necessary that the user be able to implement new rule sets, functions (and their derivatives), and operators.

## Structure

For the sake of simplicity it will be necessary to implement the language using prefix notation (Polish Notation). This will ease programming requirements (virtually eliminating string manipulation).

The language will support the following items natively:

* Variables – characters used to represent variables within functions.
* Operators – binary and unary operators such as +, -, \*, /, ^ (power), and !.
* Functions – functions such as sin, cos, tan, etc.
* Key words – special phrases for calling built in functions such as d for derivative.

A list of all natively supported functions and operators is included in Appendix A

### Examples

(der x (+ x 1)) – the first derivative of x + 1 in regards to x

(der y (+ x 1)) – the first derivative of x + 1 in regards to y

(der 2 x (+ x 1)) – the second derivative of x + 1 in regards to x

These can be expanded to things like:

(der x (x + y)) – the first derivative of x + y in regards to x

(der x (sin x)) – the first derivative of sin(x) in regards to x

(der x ((sin x)/(x))) – the first derivative of (sin x)/x in regards to x

# Appendix A

## Supported Operators

|  |  |
| --- | --- |
| Operator | Description |
| + | Addition |
| - | Subtraction |
| \* | Multiplication |
| / | Division |
| ^ | Power |
| ! | Factorial |

## Supported Functions

|  |  |
| --- | --- |
| Function | Description |
| sin | The sin function |
| cos | The cos function |
| tan | The tan function |

## Supported Keywords

|  |  |  |
| --- | --- | --- |
| Keyword | Arguments | Description |
| der | 1. A variable to operate in regards to 2. An expression to take the derivative of | Takes in a variable to differentiate with respect to as well as an expression to differentiate. |
| der | 1. The number of the derivative (1 -> first derivative, 2 -> second derivative, etc) 2. A variable to operate in regards to 3. An expression to take the derivative of | Takes in the number of the derivative (1 for first derivative, 2 for second, etc), a variable to operate in regards to, and an expression to differentiate. |
| e | None | The constant e |
| pi | None | The constant pi |