Department of Computer Engineering

University of Puerto Rico

Mayaguez Campus

CASOLUS – Calculus Solver Programming Language

Final Phase

Team Members:

Kevin Santiago Ortiz

Carlos J. Ojeda Malavé

Cristian Meléndez González

Christian J. Cancel Ramos

Introduction

Due to the difficulty of the Calculus courses, most students drop out or end up failing. Two reasons for this are that it is too expensive to pay for a solutions manual or that it is difficult to find the solutions on the web. There are a lot of books that provide the solutions to some of the problems, but they are only for the odd numbered problems. These are the reasons why our team is motivated to create a tool that gives students the support they need when solving those mathematical problems. By creating a new programming language, we will be providing an irreplaceable tool for students, allowing them to place their equations, in a simple way, to receive a quick solution and thus ensuring that every student receives the correct answers to their problems.

Language tutorial

To see a language tutorial please enter this link:
https://www.youtube.com/watch?v=aEUEii59L4g

Language reference manual

I. **Assignments**

In order to assign values to a variable you have to first declare the name of the variable and then use the equal sign to assign the value.

Cases:

1. Assigning an integration variable : CASOLUS > var = x^2
2. Assigning numeric values : CASOLUS > var1 = 8 + 9 / 12
3. Assigning the value of a variable to a new variable:
   CASOLUS > var2 = var1

II. Basic Arithmetic

The program handles basic arithmetic. It currently supports addition, subtraction, division, multiplication, and powers.

Cases:

```
CASOLUS > 8 + 8
16
CASOLUS > 8 ^ 5
32768.0
CASOLUS > 15 / 3
5
CASOLUS > 8*8
64
CASOLUS > var = x
CASOLUS > var = var + x^3
CASOLUS > var
x+x^3
CASOLUS >
```

III. Derivation

One of the simplest calculus operations are the derivatives. Our program handles derivatives by using a special keyword named “derivation”. The structure to accomplish a successful derivative is as follows: “derivation of ” and then any expression or variable.

```
CASOLUS > derivation of x^3 + 4*x
3*x^2 + 4
CASOLUS > var = x^60
CASOLUS > derivation of var
60*x^59
CASOLUS > derivation of x^20 + x^2 + 4*x
20*x^19 + 2*x + 4
CASOLUS > derivation of x*sin(x)
x*cos(x) + sin(x)
CASOLUS > derivation of x*cos(2*x)
-2*x*sin(2*x) + cos(2*x)
```

IV. Integration

Our program handles integrals by using a special keyword named “integration”.

The structure to accomplish a successful integration is as follows: “integration of ” and then any expression or variable.

1. Indefinite integrals:

```
CASOLUS > integration of x^3
x^4/4
CASOLUS > integration of x^7 + 2*
x^8/8 + x^2
CASOLUS > integration of sin(2*x)
-x*cos(2*x)/2 + sin(2*x)/4
CASOLUS > var = sin(x)*cos(x)
CASOLUS > integration of var
sin(x)^2/2
```

2. Definite integrals

```
CASOLUS > integration from 5 to 8 of x^3
3471/4
CASOLUS > integration from 5 to 90 of x^7 + 2*x
4304672099673975/8
CASOLUS > integration from 0 to 90 of sin(x)
-cos(90) + 1
CASOLUS > var = sin(x)*cos(x)
CASOLUS > integration from 0 to 10 of var
sin(10)^2/2
```

V. Limits

Our program handles limits by using a special keyword named “limit when x -> N of ”. The function also needs to know where x tends to.The structure to calculate a limit is as follows: “limit when x -> [number] of” and then any expression or variable

```
CASOLUS > limit when x -> 0 of (cos(2*x)-1)/x
0
CASOLUS > var = sin(x)
CASOLUS > limit when x -> oo of var
<-1, 1>
CASOLUS > var = tan(x)
CASOLUS > limit when x -> oo of var
<-oo, oo>
```

VI. Summation

Our program handles limits by using a special keyword named “summation”.

Summation from 0 to 5 of X^3 ←- example

```
CASOLUS > summation from 0 to  10 of x^3 + 4*x
3245
CASOLUS > summation from 0 to 10 of sin(x)
sin(5) + sin(4) + sin(10) + sin(6) + sin(3) + sin(9) + sin(7) + sin(1) + sin(2) + sin(8)
CASOLUS > summation from 0 to 5 of x^3
225
```

VII. Product Notation

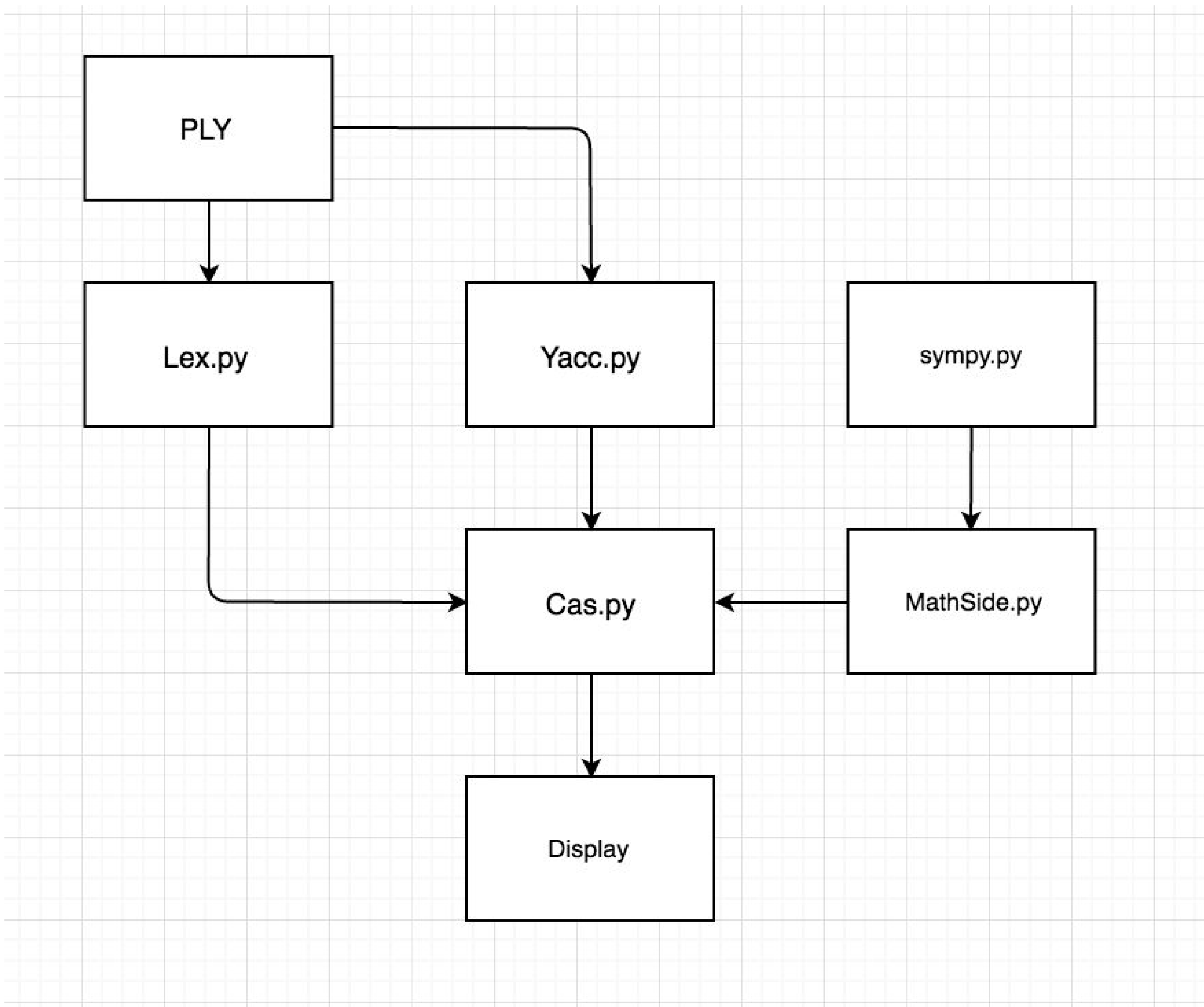
Our program handles limits by using a special keyword named "product".

product from 1 to 5 of X^3 (it has to be from 1 to n) ← example

```
CASOLUS > product from 2 to 20 of x^6
207370846870580355477536510520385395135770192365451496526593079415848450859519113691136
CASOLUS > product from 1 to 5 of x^3
1728000
CASOLUS > product from 8 to 9 of sin(x)*cos(x)
sin(8)*sin(9)*cos(8)*cos(9)
```

Language development

1. **Translator architecture**



2. **Interfaces between the modules**

The lexer and parser are both in our CAS.py file. This file also contains the main program which communicates with the NewMathSide.py file, which contains the intermediate code, and solves the calculus equations. NewMathSide communicates with the sympy library which contains the implementations for the complex calculus functions. After solving the problem, CAS.py is in charge of displaying the result to the user.

3. **Software development environment used to create the Translator**

The integrated development environment used to implement our program was Pycharm, this program allow us to write the code, compile it using Anaconda, run it and test it. Pycharm include a diverse amount of package and libraries that we used to developed the program including sympy and PLY which were fundamentals for this program. To test our program we first run CAS.py in Pycharm and send text files as input to test our lexical and sintax analyzer untill we pass all the test cases. After we succeed in passing all the test cases for the lexical and syntax analyzer we start testing the program manually looking for logical errors in the program and finally we test all the functions that we used in the program.

Conclusions

CASOLUS aims to be a tool that helps individuals solve calculus equations instead of having to buy extra book to look for solutions. Our current version of the programming language proves that CASOLUS will be a good tool for studying. It currently solves many of the calculus equations one encounter when starting to learn calculus. Even though we reached our initial objectives when creating the language there is an opportunity to increase it’s functionality. For example, CASOLUS can be programmed to accept double, and triple integrals as well as optimization problems. This will broaden the community for which this programming language will impact. For a programming language course, this project helped us understand the inner workings of a programming language as well as learning how the lexer and parser work to help create a new programming language.