Jason Vessella

3/19/16

Program 4 Report’

**Introduction**

The objective of this project was to improve upon a previously written program, with the new knowledge of functions. The original program was a calculator that could handle very simple expressions, such as addition, subtraction, division, and multiplication of two numbers. This project was to use functions to improve upon this code, so that the program can handle more complicated expressions, like those with parentheses and multiple steps.

**Method**

This program uses a while loop to stay running, until the user inputs the word “end”. It calls upon several functions within itself to determine the validity of the input, and if valid, returns the result of the entered expression.

The function simpleExpressionIsValid reads the user input and determines if it is a valid and solvable expression. It tries to get the integers from the user input, returning True if it was able to, and False if it encounters a value error.

The function lex\_complex\_expr reads the input from the user, checking for open parentheses and close parentheses. If it finds them, it checks the input between them with the simpleExpressionIsValid function. If this check returns true, it returns the result of the expressions.

The function evaluateSimpleExpression uses integers from float values of the input expressions, and then returns the solved expression using python operators. The function evaluateComplexExpression checks the length of the input, and if it equals 1 it returns the evaluateSimpleExpression function.

The functions addToMemory and viewLastOfMemory both pertain to the calculator memory. addToMemory function appends the last expression to the memory string. viewLastOfMemory prints the specified memory of the calculator, returning an error if the user attempts to call memory that does not exist.

lex\_mem is a function that is called to attempt to find any numbers entered after a space in the memory, so that all of the memory can be called. lex\_expr breaks the simple expression into three parts, the integer, sign, and integer.

**Discussion**

This program admittedly took a lot longer than initially thought. The concepts did not seem too difficult until actually trying to implement them all together to tie the program together. The hardest part of the program was getting all of the pieces to actually work together without encountering unexpected errors. After testing, I have not found any bugs with the end result, but as always it is possible that some have eluded me.