

GUI.PY

EXECUTEPROGRAM.PY

MAIN.PY

MEMORY.PY

UVSIM.PY

PROCESS\_PROGRAM.PY

purpose of the class,  
purpose of each function,  
description of any input  
parameters for the function,  
description of return value  
(if any), pre- and post-  
conditions for the function

Brandon Woodruff

GUI class  
Function: creates a simple GUI using tkinter  
Input: UVsim class  
Methods:  
select\_file():  
    purpose: search directories and choose a txt file.  
    input: n/a  
load\_file():  
    purpose: load program from the user-selected file into the sim  
    input: n/a  
operations\_output():  
    purpose: output accumulator value in GUI.  
    input: machine language command, function's name, and the operand.  
final\_output():  
    purpose: output accumulator value in GUI.  
    input: n/a  
read():  
    purpose: Read a word from the keyboard into memory.  
    input: value from keyboard  
submit():  
    purpose: command for GUI button, causes GUI to proceed  
    input: n/a  
write():  
    purpose: write a word from memory to gui  
    input: n/a  
too\_long():  
    purpose: GUI error message if sim pc exceeds available memory  
    input: n/a

SIMPLEGUI CLASS
-main: GUI window -sim: object -label: object -file_button: object -operations_text: object -program: list
select_file()
load_file()
operations_output()
final_output()
read()
submit()
write()
too_long()

EXECUTE CLASS
execute_program()

execute\_program class  
Function: to execute the program  
Input: UVSim class  
Methods:  
execute():  
    purpose: To actually execute, To call the other classes as needed.  
    input: GUI

Brandon Woodruff

MAIN
my_Sim = UVSim(100)
my_gui = SimpleGUI(my_sim)
my_gui.main.mainlo op()

Main:  
Function: to run the code  
Input: Program: n/a  
Methods: n/a

Brandon Woodruff

MEMORY CLASS
-self_registers: int
constructor()
load()
store()
load_program()
truncate()
len()

UVSIM CLASS
-_memory: int -_accumulator: int -_pc: int -_operand: int -_op: int -_program: list
Constructor()
load_ml_program()
add()
subtract()
divide()
multiply()
branch()
branch_neg()
branch_zero()
halt()

UVSim class  
Function: UVSim is a simulation that can interpret a machine language called BasicML.  
Input: n/a  
Methods:  
  
load\_ml\_program:  
    function: to load the program into memory.  
    input: a program list  
add:  
    function: to add two numbers  
    input: two values to add together  
subtract:  
    function: to subtract two numbers  
    input: two values to subtract  
divide:  
    function: to divide a number from another  
    input: a value, and a value to divide that number by  
multiply:  
    function: to multiply two numbers  
    input: two values to multiply  
branch:  
    function: to move to a specific location in memory  
    input: a value  
branchNeg:  
    function: to move to a specific location if the number is negative.  
    input: accumulator value, operand value, and pc value  
branchZero:  
    function: to move to a specific location if the number is negative  
    input:  
halt:  
    function: to end the program immediately  
    input: n/a

Brandon Woodruff

PROCESS
read_txt()

Process class  
Function: This class reads a program from a data source. It also loads the program into memory  
Input: n/a  
Methods:  
read\_txt():  
    purpose: read a program from a file and return it as a list of instructions and data values  
    input: file\_path

Brandon Woodruff