Risc-V Simulator

**Team Members:**

Ahmed ElAswar - 900211265 - ahmed-farid@aucegypt.edu

Reem Hamada -

John Ibrahim - 900226449 - johnashraf@aucegypt.edu

**Brief Explanation:**

Initially we have the names of all supported 42 instructions, register names, and 5 halting instructions in the constants file.

registers.py has the logic for collaborating with the memories' registers.

All instructions are defined in functions.py which reads from registers.py to define

how registers will be utilized within each instruction.

in parse.py, the assembly code file is read and cleaned into data structures that then populates the memory

within the same file.

The overall connection and running of the program then happens in the main file.

**Decisions/Assumptions:**

Brief Description of Implementation:

1- pc is initialized to zero.

2- We write into the memory starting from the first address which is initialized to zero.

3- We are using 2 separate data structures

-byte arrays for the data memory.

-a list for the instructions memory.

4- The size of our data memory is 128 bytes.

**Bugs/Issues:**

No known bugs or issues.

**User Guide:**

Start by running main.py where a prompt will then be given asking for the name of the assembly input file you want to run. Once given, The final output is displayed in the terminal.

Example on testcase0