Last Updated: 04/11/2025

Functional Requirements

1. **Memory Allocation:** The system shall provide a memory space of 250 words, where each word is stored as a signed six-digit decimal number.

2. **Program Loading:** The system shall allow users to load a BasicML program into memory beginning at location 00.

3. **READ Instruction:** When a READ (010) instruction is encountered, the system shall prompt the user for a valid signed six-digit decimal input and store the value at the specified memory location.

4. **WRITE Instruction:** When a WRITE (011) instruction is executed, the system shall retrieve the value from the specified memory location and display it on the screen.

5. **LOAD Instruction:** When a LOAD (020) instruction is executed, the system shall copy the value from the specified memory location into the accumulator.

6. **STORE Instruction:** When a STORE (021) instruction is executed, the system shall copy the value from the accumulator into the specified memory location.

7. **ADD Instruction:** When an ADD (030) instruction is executed, the system shall add the value from the specified memory location to the accumulator and update the accumulator with the result.

8. **SUBTRACT Instruction:** When a SUBTRACT (031) instruction is executed, the system shall subtract the value from the specified memory location from the accumulator and update the accumulator with the result.

9. **DIVIDE Instruction:** When a DIVIDE (032) instruction is executed, the system shall divide the accumulator by the value from the specified memory location, update the accumulator with the result, and if a division by zero is attempted, the system shall halt execution and display an appropriate error message.

10. **MULTIPLY Instruction:** When a MULTIPLY (033) instruction is executed, the system shall multiply the accumulator by the value from the specified memory location and update the accumulator with the result.

11. **BRANCH Instruction:** When a BRANCH (040) instruction is executed, the system shall set the execution pointer to the specified memory location.

12. **BRANCHNEG Instruction:** When a BRANCHNEG (041) instruction is executed, the system shall set the execution pointer to the specified memory location if the accumulator contains a negative value.

13. **BRANCHZERO Instruction:** When a BRANCHZERO (042) instruction is executed, the system shall set the execution pointer to the specified memory location if the accumulator contains a zero value.

14. **HALT Instruction:** When a HALT (043) instruction is executed, the system shall immediately stop program execution.

15. **Execution Logging:** The system shall log each executed instruction and all memory changes on a display panel, showing the instruction executed, and the resulting state of memory.

Non-Functional Requirements

1. **Performance:** The system shall execute a full BasicML program (up to 250 instructions) within 0.5 seconds from the moment the user clicks the "Run" button, under typical workload conditions.
2. **Usability:**
   The system's graphical user interface (GUI) shall:
   - Provide buttons with clear and unique labels.
   - Include instructions for each interactive element that accurately describe its functionality.
3. **Maintainability and Documentation**
   - Include documentation for all functions and classes.
   - Adhere to industry-standard coding conventions.
   - Ensure module independence by limiting each module to no more than 8 external dependencies and no more than 15 public functions, quantifying and ensuring low coupling for ease of maintenance and future enhancements.