Functional Requirements

1. The system shall allow loading and saving BasicML programs from/to .txt files.
2. The system shall validate all instructions for correct syntax before execution.
3. The system shall provide a Run button to execute programs and a Reset button to return to initial state.
4. The system shall display memory contents and program instructions in separate scrollable sections.
5. The system shall visually indicate execution status (running/waiting for input).
6. The system shall execute all BasicML operations:
   1. READ (10) and WRITE (11) for I/O
   2. LOAD (20) and STORE (21) for memory access
   3. Arithmetic operations (30-33)
   4. Branch operations (40-42)
   5. HALT (43)
7. The system shall handle user input via dialog prompts for READ operations.
8. The system shall display output from WRITE operations in the console.
9. The system shall maintain and display accumulator state during execution.
10. The system shall highlight the currently executing instruction.
11. The system shall properly handle arithmetic overflow conditions.
12. The system shall report all errors without crashing.
13. The system shall allow editing of loaded programs.
14. The system shall support branch operations (BRANCH, BRANCHNEG, BRANCHZERO).
15. The system shall display memory contents in a table format that updates during execution.

Non-Functional Requirements

1. The system shall execute programs in less than 10 seconds.
2. The system shall provide an intuitive user interface requiring minimal training.
3. The system shall be compatible with Windows (.NET 6.0+).
4. The system shall include a tutorial for new users.
5. The system shall respond to user input within 1 second.