**High level functionality:**

This program, UVsim, will read in a text file containing a basicML program written one command per line. Each basicML command will be a four-digit signed word. UVsim will convert the first two digits into one of 12 operations. The second two will be an operand. The operations include: 10-read, 11-write, 20-load, 21-store, 30-add, 31-subtract, 32-divide, 33-multiply, 40-branch, 41-branchneg, 42-branchzero, and 43-halt.

**User Stories:**

1. “As a student, I want to be able to write programs in basicML and have a virtual machine interpret my program, so that I can learn basic programming skills in basicML.”
2. “ As a teacher, I want to have a tool that I can use to teach basicML and computer architecture ideas, so that students can have a better understanding of coding and the underlying machine language.”

**Use cases:**

1. Have up to 100 functional registers.
2. Read program written in basicML, one command per line, into registers.
3. Run program.
4. READ function takes a terminal input and stores to memory.
5. WRITE function prints word stored in memory to terminal.
6. LOAD function stores a word in the accumulator from specified memory location.
7. STORE function stores word in accumulator in specified memory location.
8. ADD function adds a word in specified memory location to word in accumulator. Stores result in accumulator.
9. SUBTRACT function subtracts a word in specified memory location from word in accumulator. Stores result in accumulator.
10. DIVIDE function divides the word in the accumulator by a word from a specified memory location. Stores result in accumulator.
11. MULTIPLY function multiplies a word in specified memory location with word in accumulator. Stores result in accumulator.
12. BRANCH function branches to specified memory location.
13. BRANCHNEG function if accumulator is negative, branches to specified memory location.
14. BRANCHZERO function if accumulator is zero, branches to specified memory location.
15. HALT function stops running the program.