**Overview**

The **UVSim** (University Virtual Simulator) is a software-based simulator that allows students to execute machine language programs using a virtual CPU, register, and memory. The simulator interprets BasicML, a simplified machine language, enabling students to learn the fundamental concepts of computer architecture and low-level programming.

**System Components**

* **CPU**: Manages program execution by fetching, decoding, and executing instructions.
* **Accumulator Register**: A special register used for arithmetic and memory operations.
* **Memory**: Contains 100 words, each holding a four-digit signed decimal number (instruction or data).
* **Instruction Set**: Implements BasicML operations for input/output, arithmetic, memory access, and control flow.
* **Instruction Pointer**: Tracks the current instruction being executed.
* **User Interface**: A command-line interface that loads and executes programs from an input file.

**User Stories**

As a coder, I want to print some stored data to the screen, so that I can test whether my "BasicML" code segment worked.  
  
As a coder, I want to implement branching functionality in BasicML, so that I can control the flow of execution in my program.

**Use Cases**

**1. Load a Program into Memory**

* **Actor: User**
* **Description: The user loads a BasicML program into the UVSim.**

**2. Execute a Program**

* **Actor: User**
* **Description: UVSim executes the loaded BasicML program.**

**3. Handle I/O Operations**

* **Actor: User**
* **Description: The simulator processes READ (10) and WRITE (11) instructions.**

**4. Perform Arithmetic Operations**

* **Actor: User**
* **Description: UVSim supports addition, subtraction, multiplication, and division using the accumulator.**

**5. Store and Load Data**

* **Actor: User**
* **Description: UVSim can store values in memory (STORE 21) and retrieve them (LOAD 20).**

**6. Implement Control Flow**

* **Actor: User**
* **Description: UVSim executes branching instructions (BRANCH 40, BRANCHNEG 41, BRANCHZERO 42).**

**7. Halt Execution**

* **Actor: User**
* **Description: UVSim stops execution when encountering a HALT 43 instruction.**

**8. Detect and Report Errors**

* **Actor: User**
* **Description: UVSim detects invalid instructions and reports errors.**

**9. Handle Memory Limits**

* **Actor: System**
* **Description: UVSim ensures programs do not exceed 100 memory locations.**

**10. Read from an Input File**

* **Actor: System**
* **Description: UVSim loads and processes instructions from a file.**