Small Machine Little Language.

Small Machine Little Language is a small virtual machine that can run its own assembly language using special instructions that works like a real CPU

I made this project to get a better understanding of how computers and languages work, this project includes it own assembler that breaks down the source file into tokens and then assembles them into a machine-like code. The next part is the interpreter or VM that takes the generated code and executes and preforms action defined by the instructions.

This VM can allows you to preform mathematical operations decision-making using Jump instructions and control flow I included a small set of examples that shows the project in action.

Here is a list of the instructions and their purpose.

|  |  |  |
| --- | --- | --- |
| **Opcode** | **Args** | **Description** |
| PUSH | 1 | Push a value on the stack |
| POP | 0 | Remove the stacks top item |
| ADD | 0 | Add the contents of the two stack items |
| SUB | 0 | Subtract the top two items on the stack |
| MUL | 0 | Multiply the top two stack items |
| DIV | 0 | Divide the top two stack items |
| AND | 0 | Bitwise and the two top stack items |
| OR | 0 | Bitwise or the top two stack items |
| XOR | 0 | XOR the top two stack items |
| MOD | 0 | Push the remainder of the top two stack items |
| NOT | 0 | Invert the stacks top item |
| LT | 0 | Compare the two top stack items for less then |
| LTE | 0 | Compare the two top stack items for less than equal |
| GT | 0 | Compare the two top stack items for greater than |
| GTE | 0 | Compare the two top stack items for greater than equal |
| EQ | 0 | Compares of the top two stack items are equal |
| DUP | 0 | Duplicate the last top stack item |
| INT | 0 | Converts the top stacks item to integer |
| STORE | 1 | Stores the top stacks item into the memory address |
| LOAD | 1 | Loads contents of the memory address to the top of the stack |
| JMP | 1 | Jumps to a location in the code using a label name |
| JIF | 1 | Jumps to a label location if the top stack value is equal to 1 |
| JIFZ | 1 | Jumps to a label location if the top value of the stack is 0 |
| PRTD | 0 | Output the top of the stack to a double |
| PRTI | 0 | Output the top of the stack to an integer |
| PRTC | 0 | Output the top of the stack to a char |
| NEG | 0 | Flip the top of the stack between positive and negative |
| NOP | 0 | It does nothing |
| INC | 0 | Increment the top of the stack value by 1 |
| DEC | 0 | Decrement the top of the stack value by 1 |
| INP | 0 | Allow the user to input from the keyboard |