SVM's "bytecode" is the following:

Valid opcodes are:

**PUSHC <val>**: pushes the constant value <val> on the execution stack



**PUSHV <id>**: pushes the value of the identifier <id> on the execution stack



**SET <id>**: pops a value from the stack and sets <id> accordingly



**PRINT**: pops a value from the stack and prints it.

**ADD, SUB, MUL, DIV**: pops two values from the stack and pushes their

sum, difference, product, quotient respectively.

**USUB**: Changes the sign of the number on the top of the stack.

**JMP <tag>**: jumps to <tag>

**JIZ, JINZ <tag>**: if the top of the stack is (not) zero, jumps to <tag>

**Example**: this would be a valid "bytecode" file to print the numbers from 0 to 9:

PUSHC 0.0

SET a

JMP cond1

body1: PUSHV a

PRINT

PUSHV a

PUSHC 1.0

ADD

SET a

cond1: PUSHV a

PUSHC 10.0

SUB

JINZ body1