

OP KEY	Memory Opcodes	Generic Opcodes	Flow Control
Char Opcodes	Floating Opcodes	Integer Opcodes	

Opcode	<i>Parameters</i>	Description
iadd	<i>NULL</i>	Will pop stack twice and add the values and push the result back onto the stack.
isub	<i>NULL</i>	Will pop stack twice and subtract the values and push the result back onto the stack.
imul	<i>NULL</i>	Will pop stack twice and multiply the values and push the result back onto the stack.
idiv	<i>NULL</i>	Will pop stack twice and divide the values and push the result back onto the stack.
ilt	<i>NULL</i>	Will pop stack twice and do a less than operator then push the result back onto the stack.
igt	<i>NULL</i>	Will pop stack twice and do a greater than operator then push the result back onto the stack.
ilte	<i>NULL</i>	Will pop stack twice and do a less than or equal to operator then push the result back onto the stack.
igte	<i>NULL</i>	Will pop stack twice and do a greater than or equal to operator then push the result back onto the stack.
ieq	<i>NULL</i>	Will pop stack twice and check if they are equal then return the result.

iconst	<i>int</i>	Will push an int value onto the stack.
iconvert	<i>NULL</i>	Will convert float on stack into int and push.
jmp	<i>address (ptr)</i>	Basically a goto, will move the program counter to the specified ptr.
jmpt	<i>address (ptr)</i>	Same as jmp opcode but will only jump if the top of stack is true.
jmpn	<i>address (ptr)</i>	Same as jmp opcode but will only jump if the top of stack is false.
halt	<i>NULL</i>	Will terminate the program.
ret	<i>NULL</i>	Return back to the function.
call	<i>address, num_args</i>	Will call a function.
pop	<i>NULL</i>	Will pop the stack.
sys_write	<i>NULL</i>	Will pop stack and print value.
load	<i>address (ptr)</i>	Will load a local variable onto stack.
gload	<i>address (ptr)</i>	Will load a global variable onto stack.
store	<i>address (ptr)</i>	Takes the top of the stack and puts it into a local variable.
gstore	<i>address (ptr)</i>	Takes the top of the stack and puts it into a global variable.
charconst	<i>char</i>	Will push a char to the stack.
IMPORTANT NOTE	VM expects that there will be no need for conversions to occur during operations. This is something the compiler will have to take	**IMPORTANT NOTE**

	care of beforehand.	
--	----------------------------	--