

STOP	stop the program
LOAD v	$\text{stack}[\text{sp}] \leftarrow v$
PUSH v	$\text{stack}[\text{sp} + 1] \leftarrow v \parallel \text{sp} \leftarrow \text{sp} + 1$
DUPL	$\text{stack}[\text{sp} + 1] \leftarrow \text{stack}[\text{sp}] \parallel \text{sp} \leftarrow \text{sp} + 1$
SWAP	$\text{stack}[\text{sp}] \leftarrow \text{stack}[\text{sp} - 1] \parallel \text{stack}[\text{sp} - 1] \leftarrow \text{stack}[\text{sp}]$
ROT3	$\text{stack}[\text{sp}] \leftarrow \text{stack}[\text{sp} - 1] \parallel \text{stack}[\text{sp} - 1] \leftarrow \text{stack}[\text{sp} - 2] \parallel \text{stack}[\text{sp} - 2] \leftarrow \text{stack}[\text{sp}]$
IROT3	$\text{stack}[\text{sp}] \leftarrow \text{stack}[\text{sp} - 2] \parallel \text{stack}[\text{sp} - 1] \leftarrow \text{stack}[\text{sp}] \parallel \text{stack}[\text{sp} - 2] \leftarrow \text{stack}[\text{sp} - 1]$
FST	$\text{stack}[\text{sp}] \leftarrow \text{mem}[\text{stack}[\text{sp}]]$
SND	$\text{stack}[\text{sp}] \leftarrow \text{mem}[\text{stack}[\text{sp}] + 1]$
SET_FST	$\text{mem}[\text{stack}[\text{sp} - 1]] \leftarrow \text{stack}[\text{sp}] \parallel \text{sp} \leftarrow \text{sp} - 1$
SET_SND	$\text{mem}[\text{stack}[\text{sp} - 1] + 1] \leftarrow \text{stack}[\text{sp}] \parallel \text{sp} \leftarrow \text{sp} - 1$
CONS	$\text{mem}[\text{mc}] \leftarrow \text{stack}[\text{sp}] \parallel \text{mem}[\text{mc} + 1] \leftarrow \text{stack}[\text{sp} - 1] \parallel \text{stack}[\text{sp} - 1] \leftarrow \text{mc} \parallel \text{mc} \leftarrow \text{mc} + 2 \parallel \text{sp} \leftarrow \text{sp} - 1$
SPLIT	$\text{stack}[\text{sp} + 1] \leftarrow \text{mem}[\text{stack}[\text{sp}]] \parallel \text{stack}[\text{sp}] \leftarrow \text{mem}[\text{stack}[\text{sp}] + 1] \parallel \text{sp} \leftarrow \text{sp} + 1$
ADD	$\text{stack}[\text{sp} - 1] \leftarrow (\text{stack}[\text{sp}] + \text{stack}[\text{sp} - 1]) \parallel \text{sp} \leftarrow \text{sp} - 1$
SUB	$\text{stack}[\text{sp} - 1] \leftarrow (\text{stack}[\text{sp}] - \text{stack}[\text{sp} - 1]) \parallel \text{sp} \leftarrow \text{sp} - 1$
MUL	$\text{stack}[\text{sp} - 1] \leftarrow (\text{stack}[\text{sp}] \times \text{stack}[\text{sp} - 1]) \parallel \text{sp} \leftarrow \text{sp} - 1$
EQ	$\text{stack}[\text{sp} - 1] \leftarrow (\text{stack}[\text{sp}] = \text{stack}[\text{sp} - 1]) \parallel \text{sp} \leftarrow \text{sp} - 1$
LT	$\text{stack}[\text{sp} - 1] \leftarrow (\text{stack}[\text{sp}] < \text{stack}[\text{sp} - 1]) \parallel \text{sp} \leftarrow \text{sp} - 1$
CALL	$\text{stack}[\text{sp}] \leftarrow \text{stack}[\text{sp} - 1] \parallel \text{stack}[\text{sp} - 1] \leftarrow \text{pc} + 1 \parallel \text{pc} \leftarrow \text{stack}[\text{sp}]$
RETURN	$\text{pc} \leftarrow \text{stack}[\text{sp} - 1] \parallel \text{stack}[\text{sp} - 1] \leftarrow \text{stack}[\text{sp}] \parallel \text{sp} \leftarrow \text{sp} - 1$
BRANCH @1 @2	$\text{stack}[\text{sp}] \leftarrow \text{if } \text{stack}[\text{sp}] \text{ then } @1 \text{ else } @2$
JUMP @	$\text{pc} \leftarrow @$