

# Cheatsheet MVaP (version 3.1)

## Opérations sur la pile

Opcode	Pile	sp	pc	Condition
<b>PUSHI</b> <i>n</i>	$P[sp] := n$	sp+1	pc+2	
<b>POP</b>		sp-1	pc+1	$1 \leq sp$
<b>DUP</b>	$P[sp] := P[sp-1]$	sp+1	pc+1	$1 \leq sp$
<b>ADD</b> (SUB, MUL, DIV)	$P[sp-2] := P[sp-2] + P[sp-1]$	sp-1	pc+1	$2 \leq sp$
<b>SUP</b> (SUPEQ, INF, INFEQ, EQUAL, NEQ)	$P[sp-2] := 1$ si $P[sp-2] > P[sp-1]$ , 0 sinon	sp-1	pc+1	$2 \leq sp$

## Données

Opcode	Pile	sp	pc	Condition
<b>PUSHG</b> <i>n</i>	$P[sp] := P[n]$	sp+1	pc+2	$0 \leq n < sp$
<b>PUSHL</b> <i>n</i>	$P[sp] := P[fp+n]$	sp+1	pc+2	$0 \leq fp + n < sp$
<b>STOREG</b> <i>n</i>	$P[n] := P[sp-1]$	sp-1	pc+2	$1 \leq sp$ et $0 \leq n < sp$
<b>STOREL</b> <i>n</i>	$P[fp+n] := P[sp-1]$	sp-1	pc+2	$1 \leq sp$ et $0 \leq fp + n < sp$
<b>READ</b>	$P[sp] := \text{entier lu}$	sp+1	pc+1	un entier sur l'entrée standard
<b>WRITE</b>		sp	pc+1	$1 \leq sp$

## Contrôle de flot

Opcode	Pile	sp	pc	fp	Condition
<b>JUMP</b> <i>Label</i>		sp	<b>instr</b> ( <i>Label</i> )		
<b>JUMPF</b> <i>Label</i>		sp-1	<b>instr</b> ( <i>Label</i> ) si $P[sp-1]=0$ , pc+2 sinon		$1 \leq sp$
<b>CALL</b> <i>Label</i>	$P[sp] := pc+2$ , $P[sp+1] := fp$	sp+2	<b>instr</b> ( <i>Label</i> )	sp+2	
<b>RETURN</b>		fp-2	$P[fp-2]$	$P[fp-1]$	$2 \leq sp$

**Note:** avant assemblage, les *Label* font référence à des marques **LABEL** *Label* présentes dans le code. Lors de l'assemblage, ils sont remplacés par l'adresse **instr**(*Label*) dans le code.

## Flottants

Opcode	Pile	sp	pc	Condition
PUSHF $f$	$(P[sp], P[sp+1]) := f$	sp+2	pc+3	
FADD (FSUB,FMUL,FDIV)	$(P[sp-4], P[sp-3]) := (P[sp-4], P[sp-3]) + (P[sp-2], P[sp-1])$	sp-2	pc+1	$4 \leq sp$
FSUP (FSUPEQ,FINF,FINFEQ,FEQUAL,FNEQ)	$P[sp-4] := 1$ si $(P[sp-4], P[sp-3]) > (P[sp-2], P[sp-1])$ , 0 sinon	sp-3	pc+1	$4 \leq sp$
READF	$(P[sp], P[sp+1]) := \text{réel lu}$	sp+2	pc+1	un nombre flottant sur l'entrée standard
WRITEF		sp	pc+1	$2 \leq sp$
ITOF	$(P[sp-1], P[sp]) := \text{double}(P[sp-1])$	sp+1	pc+1	$1 \leq sp$
FTOI	$P[sp-2] := \text{int}(P[sp-2], P[sp-1])$	sp-1	pc+1	$2 \leq sp$

**Note:** Les nombres flottants sont stockés sur deux mots mémoire.

## Opérations supplémentaires

Opcode	Pile	sp	pc	Condition
FREE $n$		sp- $n$	pc+2	$n \leq sp$
ALLOC $n$	$P[x] := 0$ pour $sp \leq x < sp+n$	sp+ $n$	pc+2	
PUSHR $n$	$P[sp-1] := P[P[sp-1] + n]$	sp	pc+2	$1 \leq sp$ et $0 < P[sp-1] + n < sp$
STORER $n$	$P[P[sp-2] + n] := P[sp-1]$	sp-2	pc+2	$1 \leq sp$ et $0 < P[sp-1] + n < sp$
JUMPR $Label$		sp-1	<b>instr</b> ( $Label$ )+ $P[sp-1]$	$1 \leq sp$
PUSHSP	$P[sp] := sp$	sp+1	pc+1	
PUSHFP	$P[sp] := fp$	sp+1	pc+1	