

Máquina Virtual

Avaliação de Expressões

o valor lógico verdadeiro é avaliado como 1 e falso como 0.

Instruções

LDC	k	(Carregar constante): $S := s + 1$; $M[s] := k$
LDV	n	(Carregar valor): $S := s + 1$; $M[s] := M[n]$
ADD		(Somar): $M[s-1] := M[s-1] + M[s]$; $s := s - 1$
SUB		(Subtrair): $M[s-1] := M[s-1] - M[s]$; $s := s - 1$
MULT		(Multiplicar): $M[s-1] := M[s-1] * M[s]$; $s := s - 1$
DIVI		(Dividir): $M[s-1] := M[s-1] \text{ div } M[s]$; $s := s - 1$
INV		(Inverter sinal): $M[s] := -M[s]$
AND		(Conjunção): se $M[s-1] = 1$ e $M[s] = 1$ então $M[s-1] := 1$ senão $M[s-1] := 0$; $s := s - 1$
OR		(Disjunção): se $M[s-1] = 1$ ou $M[s] = 1$ então $M[s-1] := 1$ senão $M[s-1] := 0$; $s := s - 1$
NEG		(Negação): $M[s] := 1 - M[s]$
CME		(Comparar menor): se $M[s-1] < M[s]$ então $M[s-1] := 1$ senão $M[s-1] := 0$; $s := s - 1$
CMA		(Comparar maior): se $M[s-1] > M[s]$ então $M[s-1] := 1$ senão $M[s-1] := 0$; $s := s - 1$
CEQ		(Comparar igual): se $M[s-1] = M[s]$ então $M[s-1] := 1$ senão $M[s-1] := 0$; $s := s - 1$
CDIF		(Comparar desigual): se $M[s-1] \neq M[s]$ então $M[s-1] := 1$ senão $M[s-1] := 0$; $s := s - 1$
CMEQ		(Comparar menor ou igual): se $M[s-1] \leq M[s]$ então $M[s-1] := 1$ senão $M[s-1] := 0$; $s := s - 1$
CMAQ		(Comparar maior ou igual): se $M[s-1] \geq M[s]$ então $M[s-1] := 1$ senão $M[s-1] := 0$; $s := s - 1$
START		(Iniciar programa principal): $S := -1$
HLT		(Parar): “Pára a execução da MVD”

Atribuição

STR n (Armazenar valor):

$M[n] := M[s]; s := s - 1$

Desvios (não há o incremento implícito sobre i)

JMP t (Desviar sempre):

$i := t$

JMPF t (Desviar se falso):

se $M[s] = 0$ então $i := t$ senão $i := i + 1$;

$s := s - 1$

Operação Nula

NULL (Nada)

Entrada

RD (Leitura):

$S := s + 1; M[s] := \text{"próximo valor de entrada"}$.

Saída

PRN (Impressão):

"Imprimir $M[s]$ "; $s := s - 1$

Alocação e Desalocação de Variáveis

ALLOC m, n (Alocar memória):

Para $k := 0$ até $n - 1$ **faça**

$\{ s := s + 1; M[s] := M[m + k] \}$

DALLOC m, n (Desalocar memória):

Para $k := n - 1$ até 0 **faça**

$\{ M[m + k] := M[s]; s := s - 1 \}$

Chamada de Rotina

CALL t (Chamar procedimento ou função):

$S := s + 1; M[s] := i + 1; i := t$

RETURN (Retornar de procedimento):

$i := M[s]; s := s - 1$

Exemplo:

```
programa exemplo6;  
  var x, y: integer;  
  
  procedimento p;  
    var z: integer;  
  inicio  
    z:= x; x:=x-1;  
    se z>1 entao p (1)  
      senao y:=1;  
    y:=y*z  
  fim { p };  
inicio  
  leia(x);  
  p;  
  escreva (y);  
  escreva (x)  
fim.
```

A pilha *M* está invertida!

Supondo que o valor lido em x seja igual a 2:

	START		programa								
	ALLOC	0,2	var x,y		x		x		x		x
	JMP	L1									
L2	NULL		procedimento p								
	ALLOC	2,1	var z		y		y		y		y
	LDV	0									
	STR	2	z:=x								
	LDV	0				z1	z	z2	z	z3	z
	LDC	1									
	SUB							*	*	*	
	STR	0	x:=x-1					*	*	*	
	LDV	2						*	*	*	
	LDC	1									
	CMA		se z>1					z1	z1	z1	
	JMPF	L3	entao								
	CALL	L2	p						*	*	
	JMP	L4							*	*	
L3	NULL		senao						*	*	
	LDC	1									
	STR	1	y:=1						z2	z2	
L4	NULL										
	LDV	1									
	LDV	2								*	
	MULT									*	
	STR	1	y:=y*z							*	
	DALLOC	2,1	fim								
	RETURN									z3	
L1	NULL										
	RD										
	STR	0	leia(x)								
	CALL	L2	p								
	LDV	0									
	PRN		escreva (x)								
	LDV	1									
	PRN		escreva (y)								
	DALLOC	0,2	fim.								
	HLT										

Exemplo de interface para a Máquina Virtual:

