



Universidade Estadual de Santa Cruz – UESC

Relatórios de Implementações de p-code Machine

Docente César Alberto Bravo Pariente

Discente Matheus Miranda Brandão

Matrícula 201820065

Disciplina Compiladores

Curso Ciência da Computação

Semestre 2022.2

**Ilhéus – BA
2022**

Índice

P-Code Machine	3
Comandos válidos:	3
Tabela de operações:	4
Compilando e Executando	5
Exercícios e Testes	6
Soma de dois números inteiros	6
Soma dos números naturais de 1 até 10	7
Soma dos números naturais de 1 até 100 (iterativamente)	8
Soma dos quadrados dos números naturais de 1 até 100 (iterativamente)	9
Soma dos cubos dos números naturais de 1 até 100 (iterativamente)	11
Link para download	13
Referências	14

P-Code Machine

O projeto consiste na implementação em C de um algoritmo feito em Pascal code. A execução do código segue a regra dos comandos da p-code machine e sua tabela de operações.

Comandos válidos:

LIT 0, a : carrega uma constante a.

OPR 0, a : executa uma operação delimitada entre os intervalos [0,13]..

LOD l, a : Carrega uma variável para o nível l

STO l, a : Armazena uma variável no nível l

CAL l, a : Chama um procedimento no nível l;

INT 0, a : Incrementa o registrador t em a;

JMP 0, a : Pula para a instrução a;

JPC 0, a : Pulo condicional para a instrução a (Se '0' pular, senão ignorar).

Tabela de operações:

Foi considerada a seguinte codificação de operações

Código	Símbolo	Semântica
0	Return	Realiza o retorno de uma subrotina
1	Negate	$x = \text{pop}(); \text{push}(-x)$
2	Add	$x = \text{pop}(); y = \text{pop}(); \text{push}(y+x).$
3	Subtract	$x = \text{pop}(); y = \text{pop}(); \text{push}(y-x).$
4	Multiply	$x = \text{pop}(); y = \text{pop}(); \text{push}(y*x).$
5	Divide	$x = \text{pop}(); y = \text{pop}(); \text{push}(y/x).$
6	Odd?	Testa se o valor no topo da pilha é ímpar.
7	Equal?	$x = \text{pop}(); y = \text{pop}(); \text{push}(y==x).$
8	Not equal?	$x = \text{pop}(); y = \text{pop}(); \text{push}(y!=x).$
9	Less then?	$x = \text{pop}(); y = \text{pop}(); \text{push}(y<x).$
10	Bigger or equal then?	$x = \text{pop}(); y = \text{pop}(); \text{push}(y>=x).$
11	Bigger then?	$x = \text{pop}(); y = \text{pop}(); \text{push}(y>x)$
12	Less or equal then?	$x = \text{pop}(); y = \text{pop}(); \text{push}(y<=x)$

Compilando e Executando

Para a execução não é necessário o uso de nenhuma dependência, basta compilá-lo normalmente.

```
$ gcc p-code.c -o p-code
```

Ao executar é necessário digitar o nome do arquivo destino contendo as instruções, caso contrário resultará em erro.

Exemplo:

```
$ ./p-code arquivo.txt
```

Exercícios e Testes

Soma de dois números inteiros

Para exemplificação foi utilizada como entradas os números inteiros 1 e 2:

INT 0 2

LIT 0 1

STO 0 0

LIT 0 2

STO 0 1

OPR 0 2

OPR 0 0

Output: 3

Soma dos números naturais de 1 até 10

Entrada Utilizada:

INT 0 3

LIT 0 1

STO 0 0

LIT 0 2

STO 0 1

LOD 0 0

LOD 0 1

OPR 0 2

STO 0 2

LOD 0 0

LOD 0 1

OPR 0 2

STO 0 1

LOD 0 1

LOD 0 2

OPR 0 2

STO 0 2

LOD 0 1

LIT 0 10

OPR 0 7

JPC 0 9

OPR 0 0

Output: 55

Soma dos números naturais de 1 até 100 (iterativamente)

Entrada Utilizada:

INT 0 3

LIT 0 1

STO 0 0

LIT 0 2

STO 0 1

LOD 0 0

LOD 0 1

OPR 0 2

STO 0 2

LOD 0 0

LOD 0 1

OPR 0 2

STO 0 1

LOD 0 1

LOD 0 2

OPR 0 2

STO 0 2

LOD 0 1

LIT 0 100

OPR 0 7

JPC 0 9

OPR 0 0

Output: 5050

Soma dos quadrados dos números naturais de 1 até 100 (iterativamente)

Entrada Utilizada:

INT 0 3

LIT 0 1

STO 0 0

LIT 0 2

STO 0 1

LOD 0 1

LOD 0 1

OPR 0 4

LOD 0 0

OPR 0 2

STO 0 2

LOD 0 0

LOD 0 1

OPR 0 2

STO 0 1

LOD 0 1

LOD 0 1

OPR 0 4

LOD 0 2

OPR 0 2

STO 0 2

LOD 0 1

LIT 0 100

OPR 0 7

JPC 0 11

OPR 0 0

Output: 338350

Soma dos cubos dos números naturais de 1 até 100 (iterativamente)

Entrada Utilizada:

INT 0 3

LIT 0 1

STO 0 0

LIT 0 2

STO 0 1

LOD 0 1

LOD 0 1

OPR 0 4

LIT 0 2

OPR 0 4

LOD 0 0

OPR 0 2

STO 0 2

LOD 0 0

LOD 0 1

OPR 0 2

STO 0 1

LOD 0 1

LOD 0 1

OPR 0 4

LOD 0 1

OPR 0 4

LOD 0 2

OPR 0 2

STO 0 2

LOD 0 1

LIT 0 100

OPR 0 7

JPC 0 13

OPR 0 0

Output: 25502500

Link para download

Código fonte e exemplos encontram-se para download no seguinte link:

<https://github.com/MatBrands/Compiladores/tree/master/Atividade%202>

Referências

https://en.wikipedia.org/wiki/P-code_machine

<https://homepages.cwi.nl/~steven/pascal/book/10pcode.html>

<https://blackmesatech.com/2011/12/pl0/pl0.xhtml>