

UESC – Universidade Estadual de Santa Cruz

Relatório de Projeto

Relatório do Projeto2C da disciplina Conceitos de Linguagens de
Programação (CET087)

Giovanne Almeida Messias (201120035)
25/06/2015

Tabela de conteúdo

[Instruções para compilar o projeto \(usando GCC\)](#)

[Instruções para executar o projeto](#)

[Das amostras](#)

[Links pra download](#)

1. Instruções para compilação e execução (com GCC)

Para executar estes passos é preciso ter o GCC instalado. Abra o **Prompt de Comando** e construa o executável a partir do código fonte **pCodeMachine.c**.

```
> gcc pCodeMachine.c -o pCodeMachine.exe [Enter]
```

Para executar o programa basta chamá-lo passando um dos arquivos de amostra como parâmetro.

```
> pCodeMachine.exe <arquivo de amostra>.txt [Enter]
```

Ao término da análise um arquivo de resultados será criado no diretório do projeto.

2. Os arquivos de amostras

Os arquivos de amostra estão separados em 3 tipos:

- sample-arithmetic.txt - Amostras de expressões aritméticas

Neste arquivo de amostras existem operações de:

- Soma ($a+b$)
- Subtração ($a-b$)
- Multiplicação ($a*b$)
- Divisão (a/b)

- sample-logic.txt - Amostras de operações aritméticas de comparação

Neste arquivo de amostras existem operações de:

- Maior ($a>b$)
- Menor ($a<b$)
- Maior ou igual ($a\geq b$)
- Menor ou igual ($a\leq b$)
- Igual ($a=b$)
- Não igual ($a\neq b$)
- Ímpar ($a\text{ODD}$)

- sample-polish - Amostras de expressões aritméticas no formato polonês ($+ a b$)

Assim como nas amostras de expressões aritméticas as operações são de:

- Soma ($+ a b$)
- Subtração ($- a b$)
- Multiplicação ($* a b$)
- Divisão ($/ a b$)

Cada arquivo conta com 8 amostras de cada tipo.

3. Os arquivos de resultados

Após a análise de um arquivo de amostra, um arquivo de resultados é criado na pasta do projeto. Neste arquivo estão listadas as expressões analisadas e os passos executados pela máquina para realizar cada operação e a pilha de execução.

Por exemplo, uma análise da expressão **0+1** se apresenta no arquivo de resultados como:

Analyzing: 0+1 -----

1. LIT 0 0

2. STO 0 3

3. LIT 0 1

4. STO 0 4

5. LOD 0 3

6. LOD 0 4

7. OPR 0 2

s|1|1|0|0|1|0|

4. Referências

Wikipedia

<https://en.wikipedia.org/?title=P-code_machine>

Black Mesa Tech

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

5. Link para download do projeto

GitHub

<<https://github.com/giovannealmeida/CLP2C/archive/master.zip>>