|  |  |
| --- | --- |
|  | UNIVERSIDADE ESTADUAL DE SANTA CRUZ – UESCDEPARTAMENTO DE CIÊNCIAS EXATAS E TECNOLOGICAS - DCETCONCEITOS DE LINGUAGEM DE PROGRAMAÇÃO - CET087 |

**P- CODE MACHINE : Representação Vetorial para gerar notação polonesa reversa**

Discente : Philiphe Kramer

Matrícula : 201410879

Docente : César Bravo Pariente

**Sumário**

[1. Definições 3](#_Toc449298288)

[1.1 Sobre o relatório 3](#_Toc449298289)

[1.2 Sobre a entrada 3](#_Toc449298290)

[1.3 Sobre a saída 3](#_Toc449298291)

[2. Detalhes de Implementação 3](#_Toc449298292)

[2.1 Arvore binária vetorial 3](#_Toc449298293)

[3. Compilação e execução 4](#_Toc449298294)

[3.1 Linha de comando de compilação do projeto 4](#_Toc449298295)

[3.2 Linha de comando de execução do projeto 4](#_Toc449298296)

[3.3 Teste realizados 4](#_Toc449298297)

[4. Link para download 5](#_Toc449298298)

**Relatório**

# Definições

## Sobre o relatório

Neste relátório consta o Projeto 1c sobre o ambiente da P-code machine.Séra abordado o método de conversão de uma árvore de análise sintatica para a notação polonesa reversa através do percurso pós ordem da árvore e por fim essa notação polonesa reversa alimentará a P-code machine com instruçoes geradas via código originando o resultado númerico da expressão contida na árvore.

## Sobre a entrada

A entrada da árvore binária em representação vetorialé feita via arquivo de texto e considera-se que ela está formatada corretamente.

As posiçoes que não estão ocupadas na árvore/vetor são representadas pelo caracter ‘X’ ( maíusculo) para facilitar o processo.

O tamanho máximo que esta implementação aceita é de 30 caracteres.

A notação polonesa reversa é gerada a partir de um é arquivo de texto e considera-se que ela está formatada corretamente.

A entrada das intruções P code é feita dentro do próprio código a partir da notação polonesa reversa de entrada.

## Sobre a saída

Para visualizar o resultado na execução do código fonte, basta executá-lo e ter certeza que os arquivos de entrada estejam presentes.

Na saída das instruçoes p**-**code não é considerado o estado nulo/zero em que a pilha está totalmente vazia.

A notação polonesa reversa que está sendo feita a análise na p**-**code é impressa logo no inicio da execução e seu resultado estará sempre no topo da pilha final.

# Detalhes de Implementação

Os apontadores respresentados no setor f da estrutura de code ( code.f ) são enumerados de INT, STO, LIT, LOD, OPR, CAL, FCH, JMP, JPC em ordem crescente.

A demonstração de resultados é feita a cada instrução executada da pilha mostrando qual foi a intrução, os apontadores f,a,l do vetor code , t,p e b e a pilha resultante das operaçoes feitas.

## Arvore binária em representação vetorial

A árvore binária foi implementada usando um vetor/array em que a raiz ocupa a posição [0] e para cada “nó”, seu filho a esquerda está na posição *2\*i + 1*e o seu filho a direita está na posição *2\*i + 2***.**

Para gerar a notação polonesa reversa a partir da árvore binária , é feito o percurso pós ordem do vetor,onde *i*representa a posição atual no vetor,baseado na recursividade a seguir :

*void Post\_order****(****int i***) {**

*if* **(***arvore***[***i***] ==** *NULL***)**

*return;*

*Post\_order***(***2\*i***+1)***;*

*Post\_order***(***2\*i***+2)***;*

**}**

*Obs: Esta é a base da recursivdade, no código fonte modificações foram feitas para se adequar ao problema.*

# Compilação e execução

## Linha de comando de compilação do projeto

gcc p-code\_machine\_kramer.c -o prog1c

## Linha de comando de execução do projeto

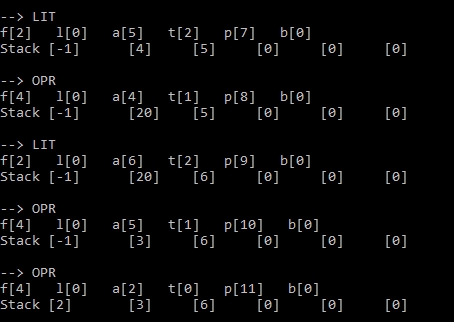
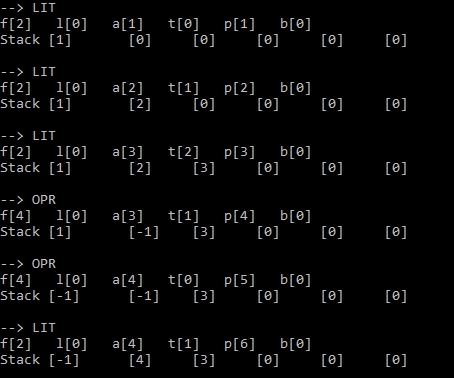
./prog1c

## Teste realizados

Expressão aritimética: ((1\*(2-3))+((4\*5)/6))

Notação polonesa reversa gerada : 123-\*45\*6/+

Saída:



O resultado da expressão polonesa reversa é apresentado no valor do item no topo da pilha na ultima instrução, que corresponde ao resultado da expressão aritimetica inicial ((1\*(2-3))+((4\*5)/6)) = 2.

Instruçoes em P code geradas :

|  |  |  |
| --- | --- | --- |
| i | Code[i] | Descrição |
| 0 | (LIT, 0, 1) | Push 1 |
| 1 | (LIT, 0, 2) | Push 2 |
| 2 | (LIT, 0, 3) | Push 3 |
| 3 | (OPR, 0, 3) | Faz a subtração de 2 - 3 |
| 4 | (OPR, 0, 4) | Multiplica o resultado por 1 |
| 5 | (LIT, 0, 4) | Push 4 |
| 6 | (LIT, 0, 5) | Push 5 |
| 7 | (OPR, 0, 4) | Multiplica 4\*5 |
| 8 | (LIT, 0, 3) | Push 6 |
| 9 | (OPR, 0, 5) | Divide o resultado por 6 |
| 10 | (LIT, 0, 2) | Soma os dois lados |

# Link para download

<https://onedrive.live.com/redir?resid=59B4855138F9B960!374&authkey=!AKwvw95yyCFVTos&ithint=folder%2c>