

Relatório final do primeiro trabalho de Estrutura de Dados

Carolina Estrella Machado

19 de outubro de 2018

Universidade de Brasília - Instituto de Ciências Exatas
Departamento de Ciência da Computação - CIC 116319 - Estrutura de Dados
2018.2 - Turma A - Professor Marcos Fagundes Caetano
carolina@host.com.br

1 Descrição do funcionamento da Notação Polonesa Reversa

A manipulação de expressões matemáticas da forma infixa, que é a usual, para a Notação Polonesa Reversa, também conhecida como notação pós-fixa é um exemplo da aplicação da estrutura de dados do tipo pilha. Na forma tradicional (infixa), o operador aparece entre os operandos, dificultando assim a resolução do problema, já que existe uma prioridade que deve ser seguida a fim de alcançar o resultado correto. Já na notação pós-fixa, os operandos e operadores são colocados na pilha de acordo com sua prioridade e ordem de execução. Dessa forma, a expressão ' $A + B$ ' é alterada para ' $A B +$ ', que está disposta na ordem em que será executada (da esquerda para a direita).

2 Arquitetura do sistema desenvolvido

Para a realização do trabalho foi necessário a criação de quatro arquivos, sendo estes `pilha.c/.h` e `calculadora.c/.h`.

Os arquivos `pilha.c/.h` contêm as assinaturas e implementação de uma Pilha. Para a resolução do problema, foi necessário criar dois tipos de pilha, uma que recebe um caractere e a outra um número real do tipo `double`.

O arquivo `calculadora.c` contém todas as funções necessárias para resolver o problema apresentado, sendo estas criadas a partir do pseudocódigo descrito na especificação do trabalho. No modo expressão, há três funções principais: A de validação da expressão, a que transforma a expressão da forma infixa para a pós-fixa e a que resolve a expressão. Além dessas funções, outras auxiliares foram criadas, como por exemplo a que verifica se o caractere é um operador ou operando e a função que converte uma expressão do tipo `char` para `double`.

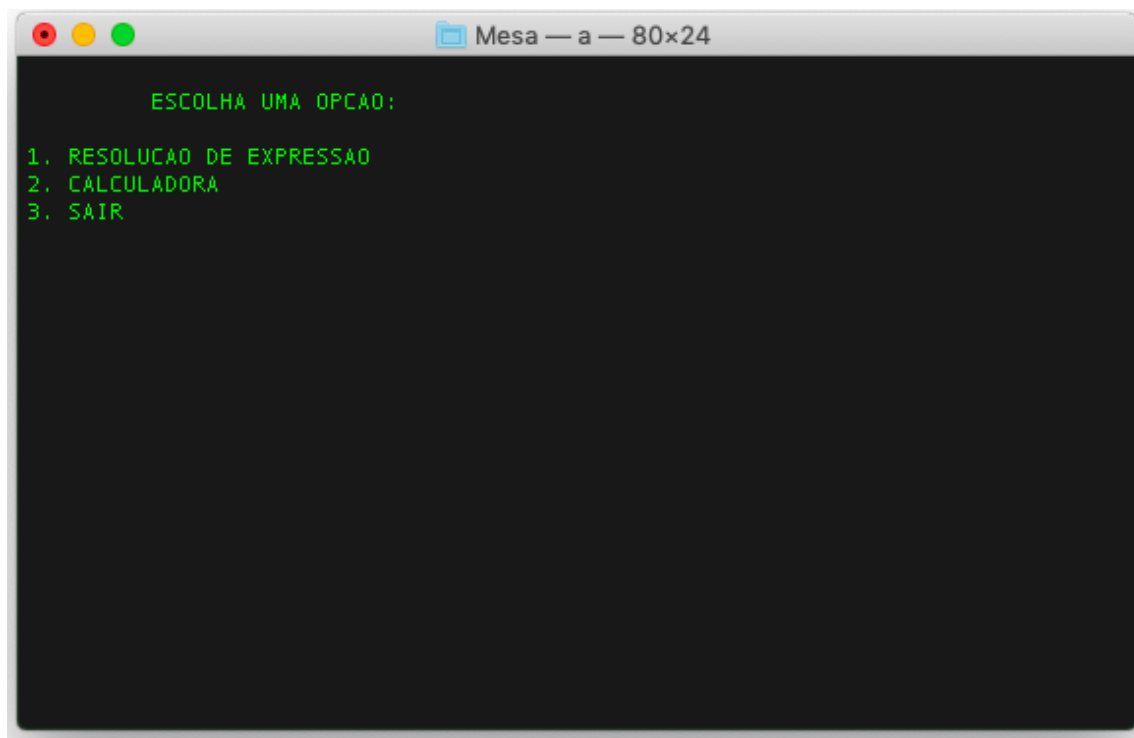


Figura 1: Menu principal.

Já no modo calculadora, também implementado no arquivo `calculadora.c`, as funções foram criadas com base em instruções do roteiro. As principais funções desse modo são as `void modonormal(char op, pilha* Pilha);` e `void modorepeticao(char op, pilha* Pilha);`. A primeira recebe números reais ou operadores, sendo que os números são empilhados até que um operador seja digitado. Quando digitado, a operação desejada é realizada com os dois primeiros elementos do topo e o resultado é novamente empilhado. No modorepetição, o usuário pode escolher operações especiais "!" ou "c", sendo o "!" acompanhado de algum operador que vai realizar a operação escolhida com todos os elementos da pilha até que esteja vazia. Quando escolhido o "c", o número no topo da pilha vai indicar quantas vezes o segundo elemento da pilha deve ser empilhado em uma nova pilha.

3 *Screenshots* e explicação do funcionamento de todas as funcionalidades implementadas



```
Mesa — a — 80x24
MODULO RESOLUCAO DE EXPRESSAO

Digite uma expressao valida - ex: (2*3)+5 ou ( 2 * 3 ) + 5
3,25-1*2+3,25-1
Equacao valida!
forma infixa = 3,25-1*2+3,25-1
forma posfixa = 3,25 1 2 *- 3,25 + 1 -
Resultado = 3.500000
```

Figura 2: Validação, transformação da forma infixa para a pós-fixa e resultado da expressão.



```
Mesa — a — 80x24
MODULO RESOLUCAO DE EXPRESSAO

Digite uma expressao valida - ex: (2*3)+5 ou ( 2 * 3 ) + 5
(2+3)*2
Equacao valida!
forma infixa = (2+3)*2
forma posfixa = 2 3 + 2 *
Resultado = 10.000000
```

Figura 3: Exemplo de expressão com parênteses.

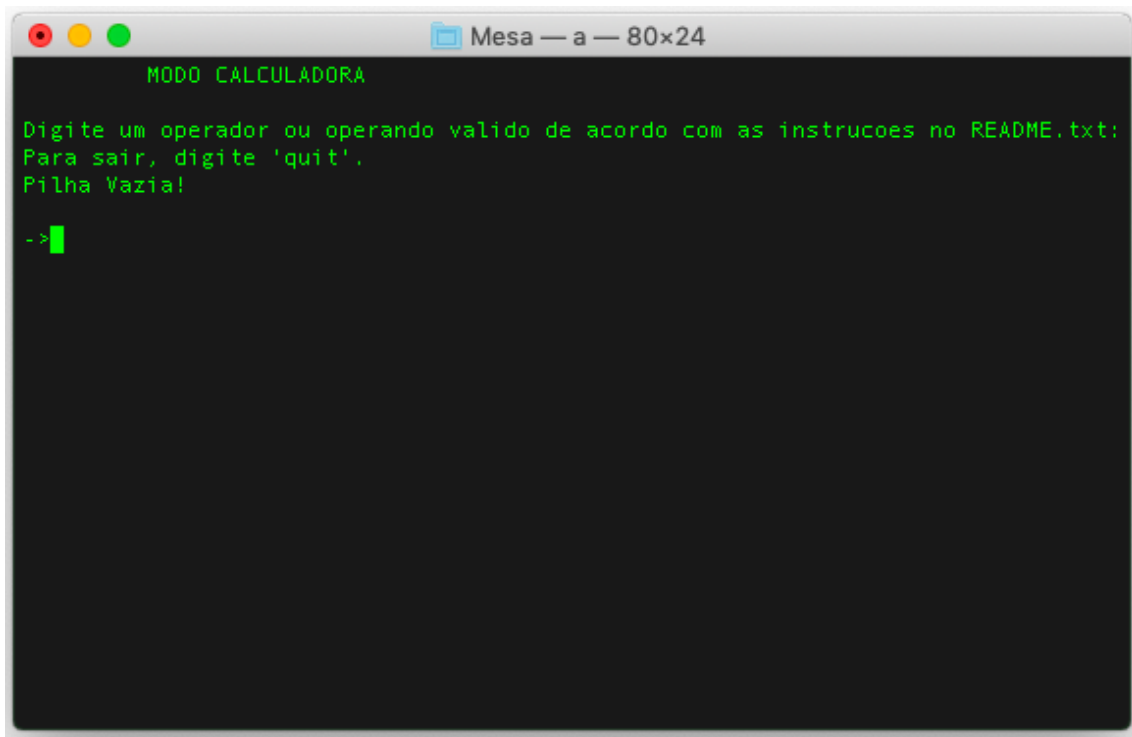


Figura 4: Modo Calculadora.

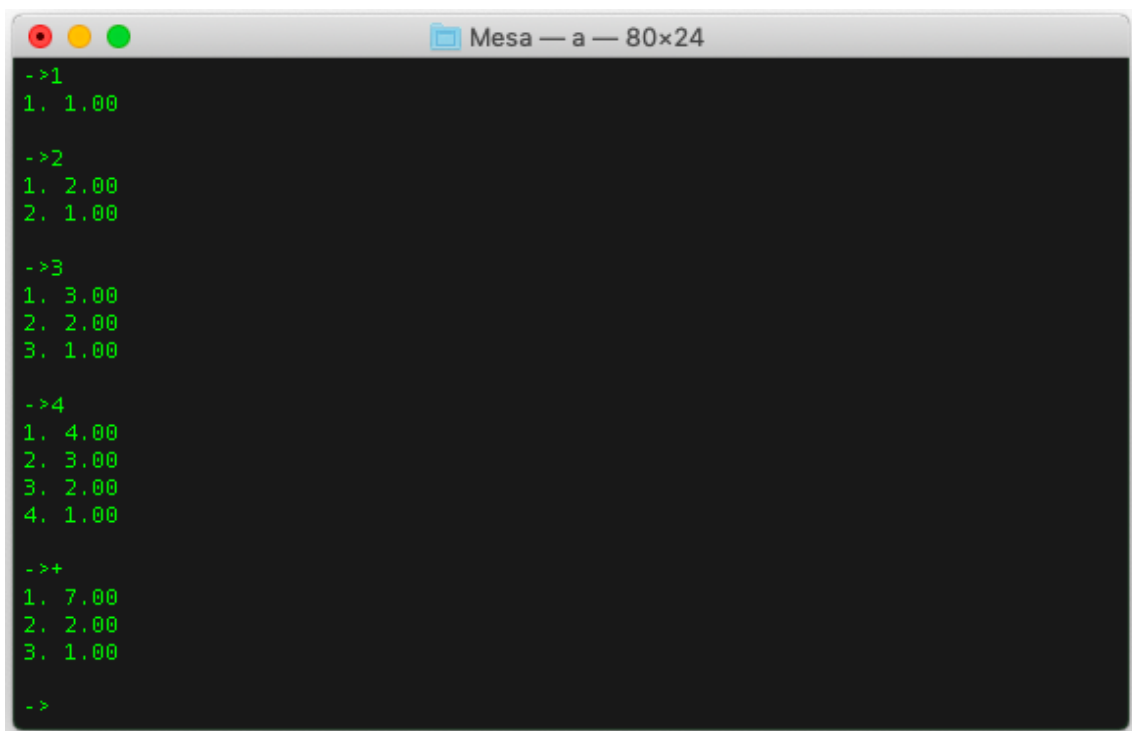
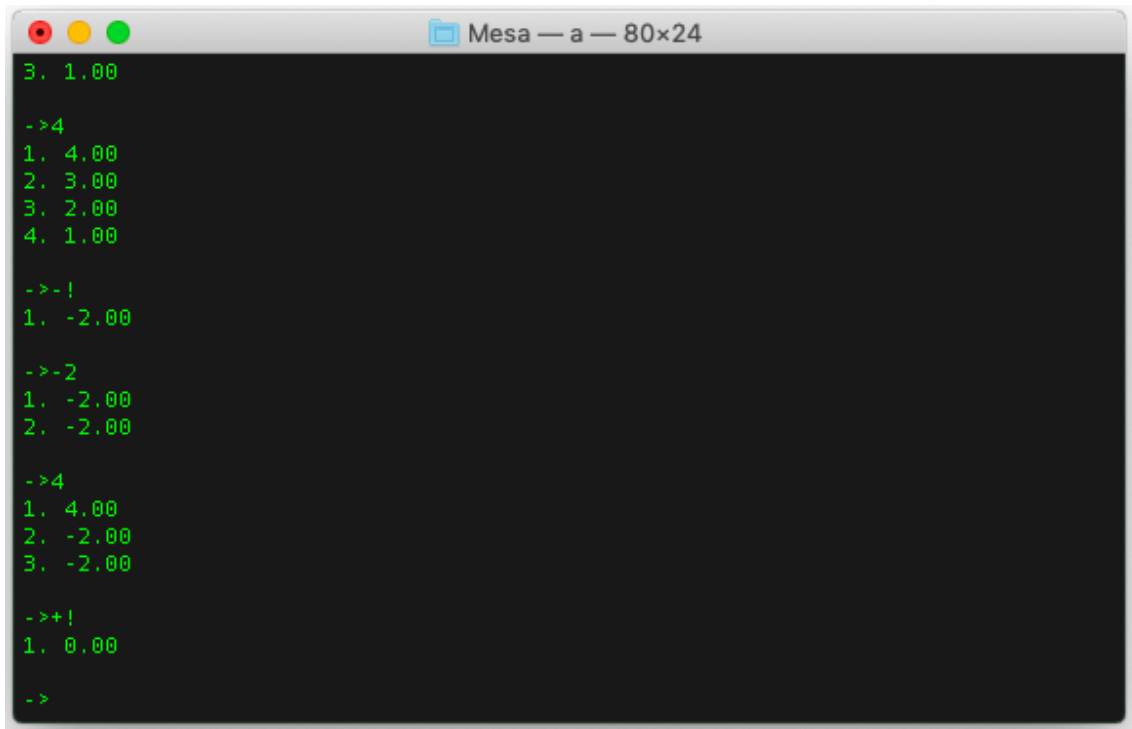


Figura 5: Exemplo de utilização do modo calculadora com a operação '+'.



```
Mesa — a — 80x24
3. 1.00

->4
1. 4.00
2. 3.00
3. 2.00
4. 1.00

->-!
1. -2.00

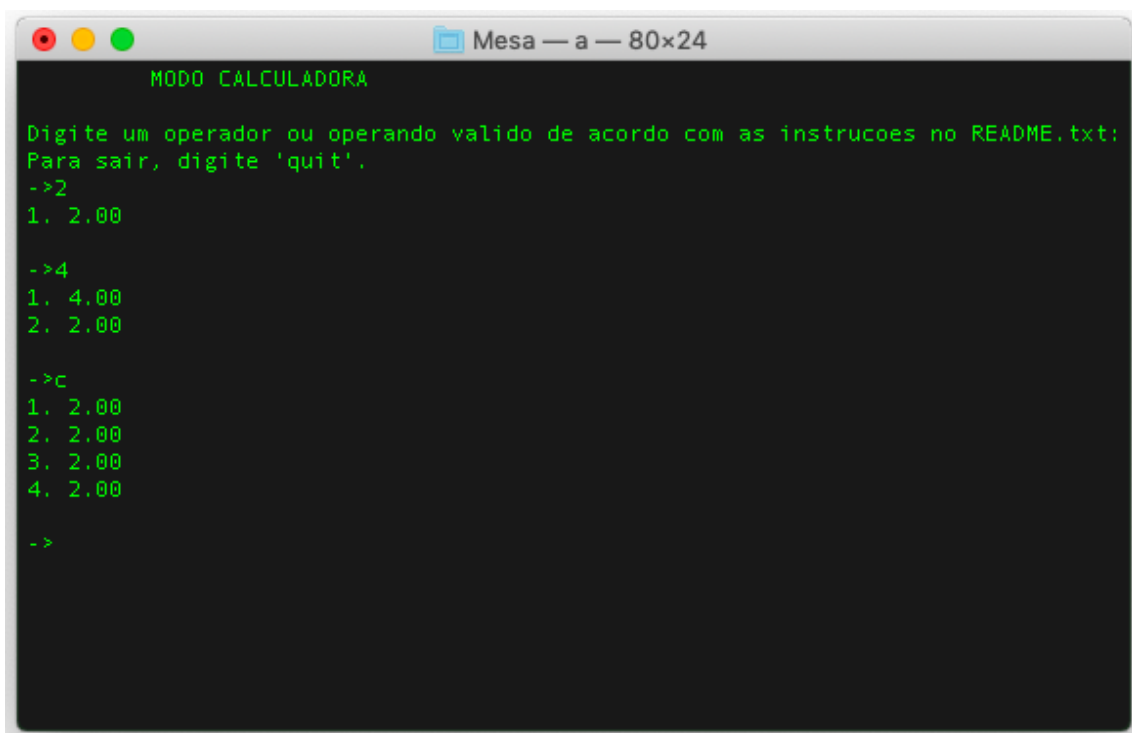
->-2
1. -2.00
2. -2.00

->4
1. 4.00
2. -2.00
3. -2.00

->+!
1. 0.00

->
```

Figura 6: Exemplo de utilização do operador '!' associado a outros.



```
Mesa — a — 80x24
MODO CALCULADORA

Digite um operador ou operando valido de acordo com as instrucoes no README.txt:
Para sair, digite 'quit'.
->2
1. 2.00

->4
1. 4.00
2. 2.00

->c
1. 2.00
2. 2.00
3. 2.00
4. 2.00

->
```

Figura 7: Exemplo de utilização da função cópia de elemento 'c'.