

Relatório do 2º Projeto de Estrutura de Dados

Integrantes:

Nome:	RA:
Guilherme Salim de Barros	18188
Maria Eduarda Elias Rocha	18190

Objetivos do Projeto:

Com base no prévio conhecimento adquirido em sala de aula sobre **Estrutura de Dados**, fazer uma **calculadora** na qual, em seu sistema, implementa a estrutura de dados pilha e tenha o cálculo de apresentação de resultados baseados na conversão deles para pós-fixa.

Desenvolvimento:

Início do Projeto → Dia 5 de Abril de 2019

Término Completo do Projeto → Dia 08 de Maio de 2019

05/04

Criação do Projeto da Calculadora usando o Visual Studio 2017 como a IDE (Ambiente de Desenvolvimento Integrado) e adição das classes futuramente usadas, como "ListaSimples", "PilhaLista", "NoLista" e entre outras.

13/04

Desenvolvimento completo da interface com o usuário, ou seja, com o **Windows Form**, visamos criar um design semelhante ao apresentado na proposta do trabalho, com duas “TextBox” para a entrada de valores e resultado, botões para todos os números e outros caracteres usados para operações matemáticas (como “ + ”, “ * ”, “ - ”, “ / ”), além de um botão com o símbolo “ = ” para calcular o resultado da expressão. A formatação criada pode ser vista na imagem abaixo:

The image shows a Windows Form application designed as a calculator. The background is a vertical gradient from dark purple at the top to bright yellow at the bottom. The interface includes the following elements:

- Original:** A text label followed by a single-line text box.
- Resultado:** A text label followed by a single-line text box.
- Pósfixa:** A text label.
- Infixa:** A text label.
- Buttons:** A grid of buttons for calculator functions:
 - Row 1: "CE", "(", and ")"
 - Row 2: "7", "8", "9", "+", and "/"
 - Row 3: "4", "5", "6", "-", and "*"
 - Row 4: "1", "2", "3", "^", and a tall "=" button
 - Row 5: "0", ",", and "C"

Não só fizemos a interface, mas demos início a entrada de dados no programa. A todos os botões foram atribuídas funções e o visor dos dados digitados já é lido.

16/04

Demandamos uma grande quantia de tempo para programar essa parte do projeto, inicializamos criando uma classe chamada “Elemento”, a qual armazena uma preferência e uma string, esta seria uma parte da expressão matemática. De acordo com a preferência do elemento digitado nós atribuímos uma posição para ele no resultado em formato pós-fixado. Nós também codificamos o método de cálculo juntamente com a parte de conversão, com esse não houve tantas dificuldades e, conseqüentemente, conseguimos torná-lo eficiente.

05/05

A codificação da calculadora não se resume a somente o cálculo e a conversão das expressões matemáticas, mas assim como impedir e/ou alertar o usuário caso insira os dados de maneira errônea. Levando isso em consideração, codificamos o seguinte: caso o visor de entrada de dados esteja vazio, nenhum operador, com exceção do menos (por conta da ocasião em que a expressão seguinte a ele seja um número negativo) poderá ser digitado, somente números e parênteses. O mesmo vale para quando é digitado qualquer símbolo de uma operação ou no começo de uma expressão entre parênteses, nenhum outro desses poderá ser digitado à seguida. Além disso, caso o número de “abre” e “fecha” parênteses seja distinto, o sistema alertará o indivíduo que cometeu o erro e abortará a operação.

Conclusão:

Para esse projeto, a lógica foi estudada a fundo. O método de conversão para pós fixa foi bem complexo para entender, mas, com o tempo tudo foi esclarecido.

O tempo e o uso da memória foram pensamentos recorrentes durante o desenvolvimento. Ter certeza de que o projeto está orientado a objetos e com equilíbrio entre o tempo gasto e o uso de memória foi um dos pontos principais.

No geral, tivemos um bom controle da proposta pedida.