

Calculadora Básica e de Raízes

Alfredo Gabriel de Sousa Oliveira¹, Marcus Maciel Oliveira¹

¹Instituto de Ciências Exatas e Naturais - Faculdade de Computação
Universidade Federal do Pará (UFPA) - Belém, PA - Brasil

{marcus.oliveira, alfredo.oliveira}@icen.ufpa.br

Abstract. *This article describes the elaboration of two calculators: the basic one, which is capable of execute the five basic operations and the module operation; and the roots one, capable of calculate the two real roots based on the parameters of the quadratic equation.*

Resumo. *Este trabalho descreve a elaboração de duas calculadoras: a básica, capaz de realizar as cinco operações básicas e a operação de resto; e a de raízes, capaz de calcular as duas raízes reais a partir dos parâmetros da equação do segundo grau.*

1. Introdução

Na contemporaniedade, as calculadoras são utilizadas nos mais diversos ramos. Através da utilização de circuitos lógicos configurados com softwares para realizar cálculos binários, são capazes de resolver rapidamente os mais diversos cálculos e retornar ao usuário comum a resposta desses cálculos em formato decimal. Nesse sentido, a finalidade desse projeto é de construir duas calculadoras: uma que consiga realizar as cinco operações básicas incluindo a operação de resto de uma divisão exata; e uma que consiga calcular as raízes reais de uma equação do segundo grau a partir dos parâmetros mutáveis dessa equação.

Dessa maneira, a calculadora básica deve ser capaz de realizar as seguintes operações binárias: soma, subtração, multiplicação, divisão, exponenciação e resto da divisão exata; enquanto a calculadora de raízes do segundo grau deve ser capaz de calcular as duas raízes reais da seguinte equação: $ax^2 + bx + c = 0$, retornando as raízes caso as mesmas existam ou retornando Not a Number (NaN) caso não existam. As equações necessárias para os cálculos dessas raízes são as seguintes:

$$\Delta = b^2 + 4ac \quad (1)$$

$$r' = \frac{-b + \sqrt{\Delta}}{2a} \quad (2)$$

$$r'' = \frac{-b - \sqrt{\Delta}}{2a} \quad (3)$$

Diante disso, para a melhor compreensão deste trabalho, ele será dividido nas seguintes seções: a modelagem do software mediante uso de Diagramas UML, de forma que seja feito uma abstração dos objetivos do projeto e seja possível expressar os mesmos em diagramas; as ferramentas utilizadas, que explicitam as ferramentas externas que foram utilizadas para a elaboração deste projeto; os testes computacionais, que comprovam a funcionalidade dos recursos implementados no software da calculadora básica e da calculadora de raízes; e a conclusão do trabalho, que apresentará as considerações finais adquiridas no decorrer do trabalho. Ademais, com fito de complementar as ideias construídas nesse projeto, haverá um apêndice, de tal forma que seja explicado os pontos fundamentais do ponto de vista da codificação do software.

2. Modelagem do Software (Diagramas UML)

Nesta seção, serão evidenciados os três diagramas que foram realizados com fito de simplificar a implementação do projeto da perspectiva do programador: o diagrama de casos de uso, responsável por facilitar o levantamento de requisitos e a comunicação entre o programador e o cliente; o diagrama de classes, responsável pelos aspectos necessários na classe do projeto em questão, como os atributos e os métodos; e o diagrama de atividades, responsável por evidenciar o encadeamento lógico que o programa deve possuir para fluidez do mesmo.

2.1. Diagrama de casos de uso

Inicialmente, será explicitado o modelo do diagrama de casos de uso. Pelo fato de ser um diagrama mais informal, sua representação é mais simples: apenas indica o que deve ser colocado obrigatoriamente ou opcionalmente no programa, de forma que qualquer sujeito externo que o veja consiga compreender os componentes fundamentais e o relacionamento dos mesmos presentes no software. Pelo fato de que este diagrama necessita ser de fácil entendimento, este tipo de diagrama não contém os detalhes de implementação, mas ainda consegue ser de extrema importância para a comunicação; um dos cinco pilares da Engenharia de Software.

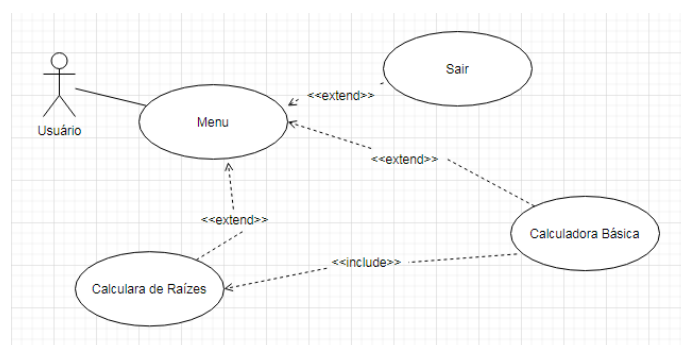


Figura 1. Diagrama de casos de uso. Fonte: acervo próprio

Conforme evidenciado acima na Figura 1, o usuário terá acesso ao menu; sendo que este possuirá três funcionalidades: acessar a calculadora básica, acessar a calculadora de raízes ou a opção de sair do programa. Como a calculadora de raízes utiliza operações básicas, conforme evidenciado na equação (1), (2) e (3), ela será atrelada à

calculadora básica. Dessa forma, a visualização dos componentes fundamentais do software relacionamento entre os mesmos ficam claramente visíveis, seja pro cliente, seja pro programador.

2.2. Diagrama de classes

Em seguida, será explicado sobre o diagrama de classes: em suma, é um diagrama que abstrai os objetos; escolhe apenas os pontos fundamentais para o projeto, e os insere como atributos ou métodos dentro de uma visibilidade definida. Além disso, enquanto no diagrama de casos de uso o relacionamento dos componentes do software foram definidos, no diagrama de classes esse conceito será mais especializado, isto é, será explicitado o tipo de relacionamento que as classes terão entre elas, como o relacionamento por composição, herança e associação. Dessa forma, o programador, antes de fazer a implementação do programa, já possui todos os elementos do mesmo definido, de tal forma que se torna necessário definir apenas os blocos de código que as funcionalidades terão.

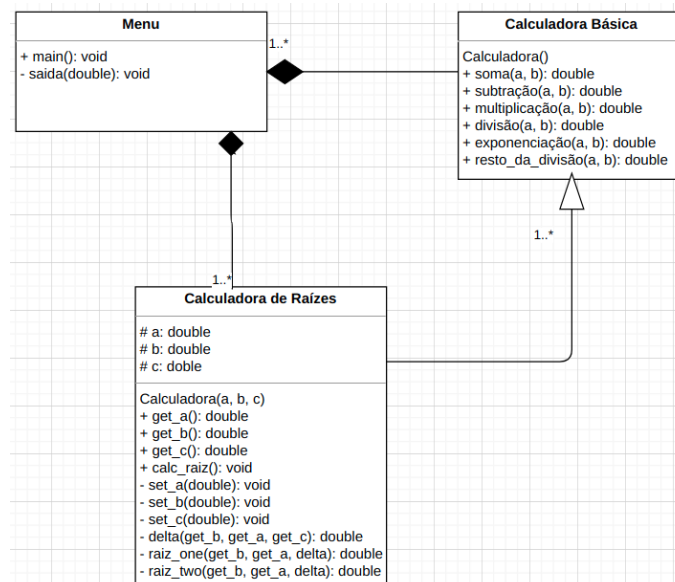


Figura 2. Diagrama de classes. Fonte: acervo próprio.

Em conformidade com a Figura 2, o diagrama de classes desse projeto é composto por três tabelas: a tabela referente à classe Calculadora Básica, a tabela referente à classe Calculadora de Raízes e a tabela referente à classe Menu.

A calculadora básica possui seis métodos, que são as quatro operações aritméticas básicas e as operações de exponenciação e de resto, no qual tem relacionamento de um para muitas classes, representado por "1..*", haja em vista que ele apesar de ser independente, existem classes que dependem dele para realizar suas atribuições.

A calculadora de raízes possui três atributos: a, b e c, que são as variáveis da equação quadrática; e os métodos relativos às equações (1), (2) e (3), os métodos de encapsulamento e os métodos adquiridos pelo relacionamento de herança, representado pela seta branca, com a classe Calculadora Básica e de forma semelhante pode ter relacionamento com varias classes.

A classe Menu é responsável por instanciar iterativamente as classes de acordo com a intenção do usuário, ou seja, as entradas de dados definirão qual comportamento o programa irá assumir. Ademais, é mister enfatizar que a classe Menu será estendida, seta preta, com os métodos da classe Calculadora Básica ou da classe Calculadora de Raízes em conformidade com a escolha do usuário e de forma análoga pode ter diversos relacionamentos com no mínimo 1.

2.3. Diagrama de atividades

Por fim, o diagrama de atividades permite ao programador elaborar o programa principal com mais facilidade. Através de uma visualização do fluxo lógico, permite ao programador elaborar um circuito lógico mais conciso; e portanto que atenda os requisitos funcionais do projeto.

Ademais, este diagrama serve de base para a estruturação das bibliotecas do programa, uma vez que, seguindo tal fluxo, em paralelo com o diagrama de casos de uso, é possível modificar tal fluxo e adicionar posteriormente novas funcionalidades de acordo com os requisitos do cliente, mantendo assim um código limpo e de fácil manutenibilidade.

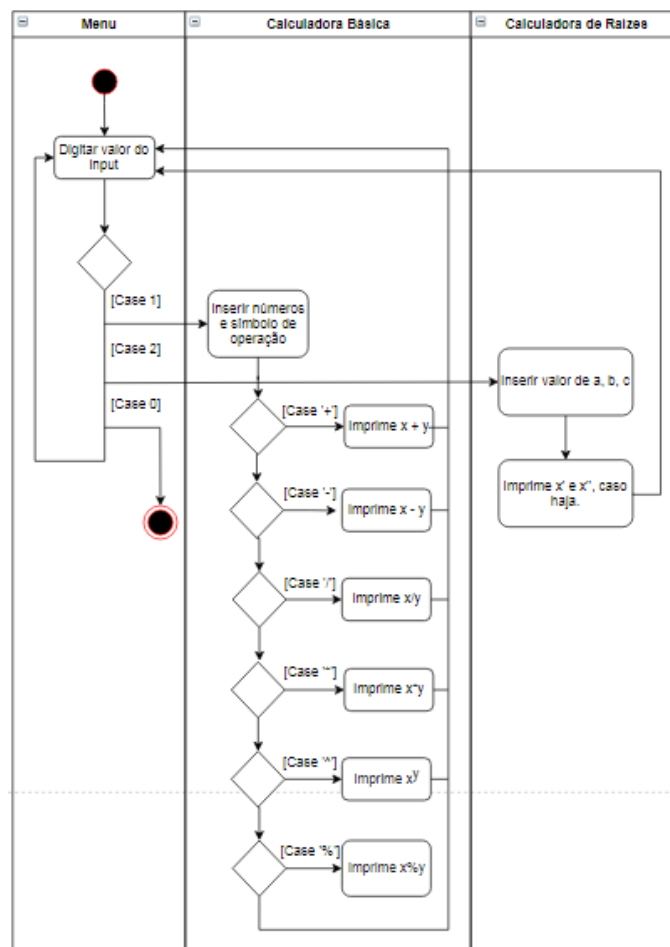


Figura 3. Diagrama de atividades. Fonte: acervo próprio.

Conforme exposto na Figura 3, a execução do programa começa em um menu, que através de uma entrada de dados, permite acessar a calculadora básica, a calculadora

de raízes, o fechamento do programa ou a execução indefinidamente do programa. Desse modo, caso o usuário acesse a calculadora básica através da entrada 1, ele terá acesso às cinco operações básicas e a operação de resto da divisão exata, e após a realização de qualquer uma das operações ou da entrada de uma operação inválida, o programa irá retornar ao início do menu. De forma similar, caso o usuário acesse a calculadora de raízes mediante entrada 2, ele necessitará inserir os parâmetros mutáveis de uma equação quadrática, ou seja, dos valores de a , b e c , de forma que os cálculos das raízes sejam feitos internamente; e, pois, sejam retornados os valores das raízes, caso existam. De forma análogo, caso o usuário digite 0, o programa será fechado automaticamente; e caso digite qualquer outra coisa, o programa continua indefinidamente até que 1, 2 ou 0 seja digitado. Dessa forma, o programador, através da compreensão do fluxo lógico que um programa deve ter, sem escrever uma linha de código, consegue modelar toda a execução de um programa.

3. Ferramentas Utilizadas

3.1. Github

Plataforma de versionamento de código com possibilidade de programar cooperativamente

3.2. Visual Studio Code

IDE utilizada para a escrita dos códigos presentes nesse trabalho, como o programa que executa a calculadora e a própria calculadora.

3.3. Draw.io

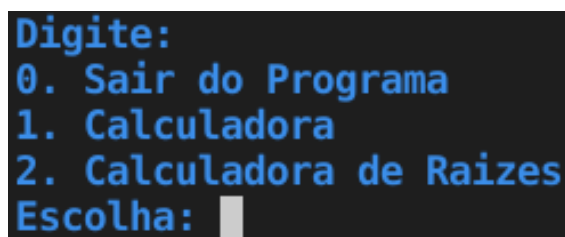
Programa utilizado para o desenvolvimento dos diagramas UML.

3.4. Overleaf

Editor de texto online usado para a construção do relatório.

4. Testes computacionais

Nesta seção, serão demonstrados os testes das funcionalidades da calculadora, no qual serão abordados todos suas operações. No apêndice, serão incluídos os testes automatizados, responsáveis por realizem testes pré-definidos a fim de verificar o comportamento do programa nas mais diversas situações, substituindo o trabalho manual de executar o programa e ir testando caso por caso. Contudo, inicialmente, expõe-se o funcionamento do programa com os expostos a seguir:



```
Digite:
0. Sair do Programa
1. Calculadora
2. Calculadora de Raizes
Escolha: █
```

Figura 4. Menu do Projeto. Fonte: acervo próprio.

Na Figura 4. temos o comprimento de um dos requisitos do projeto, isto é, o menu de apresentação/utilização do software. A seguir será demonstrado as funcionalidades da Calculadora, responsável pelas operações básicas, e as funções nela inclusa.

```
Escolha: 1
Digite:
+. Soma
-. Subtração
/. Divisão
*. Multiplicação
^. Exponenciação a^b
%. Resto da Divisão
Escolha: |
```

Figura 5. Menu da Calculadora. Fonte: acervo próprio.

Na Figura 5. após selecionarmos a calculadora básica, temos a demonstração do funcionamento do menu interativo, isto é, a demonstração do menu das operações aritméticas.

```
Escolha: +
Digite o primeiro valor: 5
Digite o segundo valor: 5
Você instanciou uma calculadora basica!!
O resultado é: 10,00
```

Figura 6. Função Soma. Fonte: acervo próprio.

Na Figura 6. há a utilização do método soma acessível pelo input "+", no qual será solicitado ao usuário dois valores e em seguida será apresentado o resultado da soma entre eles.

```
Escolha: -
Digite o primeiro valor: 5
Digite o segundo valor: 5
Você instanciou uma calculadora basica!!
O resultado é: 0,00
```

Figura 7. Função Subtração. Fonte: acervo próprio.

Na Figura 7. há a utilização do método subtração acessível pelo input "-" e a apresentação do seu resultado, obtido a partir da entrada do usuário, conforme solicitado pelo cliente. Além disso, vale ressaltar que o método retorna o valor, caso seja necessário integrar a calculadora em outro programa.

```
Escolha: /  
Digite o primeiro valor: 5  
Digite o segundo valor: 5  
Você instanciou uma calculadora basica!!  
O resultado é: 1,00
```

Figura 8. Função Divisão. Fonte: acervo próprio.

Na Figura 8. há a utilização do método divisão acessível pelo input "/", no qual o primeiro valor corresponde ao número a ser dividido e o segundo corresponde ao divisor, em seguida é apresentado o resultado para o usuário.

```
Escolha: *  
Digite o primeiro valor: 5  
Digite o segundo valor: 3  
Você instanciou uma calculadora basica!!  
O resultado é: 15,00
```

Figura 9. Função Multiplicação. Fonte: acervo próprio.

Na Figura 9. há a utilização do método responsável pela multiplicação, acessível pelo input "*", no qual o usuário irá inserir dois valores a fim de ter o resultado da multiplicação entre eles.

```
Escolha: ^  
Digite o primeiro valor: 2  
Digite o segundo valor: 3  
Você instanciou uma calculadora basica!!  
O resultado é: 8,00
```

Figura 10. Função Exponenciação. Fonte: acervo próprio.

Na Figura 10. há a utilização do método da exponenciação, acessível pelo input "^", no qual o primeiro valor corresponde a base e o segundo ao expoente.

```
Escolha: %  
Digite o primeiro valor: 5  
Digite o segundo valor: 4  
Você instanciou uma calculadora basica!!  
O resultado é: 1,00
```

Figura 11. Função Resto. Fonte: acervo próprio.

Na Figura 11. há a utilização do método que calcula o resto da divisão, acessível pelo input "%", no qual semelhante a divisão entraremos com o divisor e o dividendo e retornaremos o valor após a virgula, isto é, o que não foi possível dividir.

```
Digite:
0. Sair do Programa
1. Calculadora
2. Calculadora de Raizes
Escolha: 2
Digite o valor de a: 1
Digite o valor de b: 3
Digite o valor de c: 2
Você instanciou uma calculadora de raizes!
a = 1,00 b = 3,00 c = 2,00
Duas raizes reais e diferentes
A primeira raiz é: -1,00
A segunda raiz é: -2,00
```

Figura 12. Função Quadrática. Fonte: acervo próprio.

Na Figura 12. há a utilização do método que é responsável pela realização do cálculo das raízes, isto é, a partir das entradas a, b, c, referentes expressão matemática, a função utilizara os as equações (1), (2) e (3) para resolve-la, além dos métodos já desenvolvidos.

5. Conclusão

Tendo em vista os fatos apresentados, é possível afirmar que, na implementação das funcionalidades de um programa, o diagrama UML foi de suma importância para a simplificação do mesmo. Por mais que seja trabalhoso a construção do mesmo, foi fundamental para que se evitasse erros no design do software, e deixou explícito o objetivo do projeto e a forma do mesmo ser condizido.

Contudo, por mais que sejam de extrema importância para a modelagem do projeto, é necessário que o projetista consiga balancear a quantidade de informações no projeto UML, uma vez que esta ferramenta pode atrapalhar os programadores dependendo do tamanho do projeto em razão do excesso de funções desnecessárias presentes no mesmo. Dessa forma, é necessário a harmonia entre bom senso e a necessidade a fim de melhorar o desenvolvimento do projeto.

Destarte, é imprescindível que todas as ideias construídas e implementadas, neste trabalho, sejam registradas e vistas pelos membros da equipe a fim de possuir uma maior qualidade. Dessa forma, foi utilizado a ferramenta GitHub e o Draw.io, a fim de partilhar todos os diagramas UML e os códigos para a implementação da calculadora, de tal forma que qualquer indivíduo que tiver interesse consiga facilmente visualizar e compreender os mesmos. Todos os arquivos referentes à esse projeto estão contidos no seguinte repositório: @alverad-katsuro/atividades_prog_two.