UDESC
UNIVERSIDADE
DO ESTADO DE
SANTA CATARINA

W. S. Rebozzo

CONCURSO PÚBLICO – 01/2022

Área de Conhecimento: ENGENHARIAS/ENGENHARIA QUÍMICA OU CIÊNCIAS EXATAS E DA TERRA/MATEMÁTICA OU CIÊNCIA DA COMPUTAÇÃO
PROVA ESCRITA – PADRÃO DE GABARITO

1) Considere o problema de valor inicial (PVI):

$$\begin{cases} \dfrac{d^2y}{dt^2} + 4y = t^2e^{2t}, \quad t \in \mathbb{R}_+ \\ y(0) = 1 \\ \dfrac{dy}{dt}(0) = 1 \end{cases} \qquad (01)$$

a) (20% da questão) Use Transformadas de Laplace para encontrar uma solução analítica para o PVI (01).
b) (40% da questão) Obtenha a partir da expansão em série de Taylor, o método de Euler (ou Runge-Kutta de primeira ordem) para a aproximação da solução do problema $\frac{dy}{dx} = f(x,y)$, com valor inicial $y(x_0) = y_0$. Extenda o resultado para o método de Runge-Kutta de segunda ordem e faça uma interpretação geométrica para cada uma das duas situações expostas (Runge-Kutta de primeira e de segunda ordem). Discuta também o erro de truncamento resultante de cada um desses métodos.
c) (20% da questão) Apresente o procedimento do método de Runge-Kutta de quarta ordem para o problema dado no item anterior e estenda o resultado para a solução de um PVI de segunda ordem.
d) (20% da questão) Desenvolva um algoritmo para a solução aproximada do PVI (01) no intervalo $[0,1]$, usando o método de Runge-Kutta de quarta ordem.

Gabarito:

a) Seja $Y(s) = \mathcal{L}\{y(t\}$, a transformada de Laplace (TL) da função y.
Aplicando a TL na equação $\frac{d^2y}{dt^2} + 4y = t^2e^{2t}$, temos:

$$\mathcal{L}\left\{\frac{d^2y}{dt^2}\right\} + 4\mathcal{L}\{y\} = \mathcal{L}\{t^2e^{2t}\}$$

$$s^2Y(s) - sy(0) - y'(0) + 4Y(s) = \frac{2}{(s-2)^3}$$

$$[s^2+4]Y(s) = \frac{2}{(s-2)^3} + s + 1$$

$$Y(s) = \frac{2}{(s-2)^3(s^2+4)} + \frac{s+1}{(s^2+4)}$$

$$Y(s) = \frac{1}{32}\left[\frac{1}{s-2} - 4\frac{1}{(s-2)^2} + 4\frac{2}{(s-2)^3} + 31\frac{s}{s^2+4} + 17\frac{2}{s^2+4}\right]$$

Aplicando a transformada inversa, segue que:

$$y(t) = \frac{1}{32}[e^{2t} - 4t^{2t} + 4t^2e^{2t} + 31\cos(2t) + 17sen(2t)]$$

Pauta de correção:

- Aplicar corretamente as transformadas de Laplace: 30%
- Resolver a equação algébrica no "espaço transformado", obtendo corretamente as frações parciais: 30%
- Aplicar corretamente as transformadas inversas e obter a solução final do PVI: 40%

b) Seja $y = y(x)$ uma função com derivadas contínuas em um intervalo aberto que contém os números reais a e x. Então a expansão de Taylor de $y(x)$ em torno de a é dada por:

$$y(x) = y(a) + y'(a)\frac{x-a}{1!} + y''(a)\frac{(x-a)^2}{2!} + y'''(a)\frac{(x-a)^3}{3!} + \cdots + y^{(k)}(a)\frac{(x-a)^k}{k!} + \cdots \quad \text{(b.1)}$$

Pelo Teorema de Taylor, dado $k \in \mathbb{N}$, existe $c \in [a, x]$ tal que:

$$y(x) = P_k(x) + R_k(x) \quad \text{(b.2)}$$

onde

$$P_k(x) = y(a) + y'(a)\frac{x-a}{1!} + \cdots + y^{(k)}(a)\frac{(x-a)^k}{k!} \quad \text{(b.3)}$$

e

$$R_k(x) = y^{(k+1)}(c)\frac{(x-a)^{k+1}}{(k+1)!}. \quad \text{(b.4)}$$

P_k é chamado de polinômio de Taylor e R_k de resto ou erro de truncamento da série.
Note que se tomarmos $M = \max_{\xi\in[0,x]}\{y^{(k+1)}(\xi)\}$, então
$R_k(x) \le M\frac{(x-a)^{k+1}}{(k+1)!}$, o que implica em considerar que quando x tende a a, o erro de truncamento tende a zero.
Para aproximar a solução $y(x)$ de um PVI, em um intervalo $[\alpha, \beta]$, particionado em N subintervalos de comprimento $h = \frac{\beta-\alpha}{N}$, se considerarmos $k = 1$ em (b.2), temos:

$$y(x) = P_1(x) + R_1(x) \approx P_1(x).$$

Dessa forma, ao aproximarmos a solução $y(x) = P_1(x)$, temos um erro de truncamento $R_1(x)$. A aproximação $y(x) = P_1(x)$ nos leva a:

$$y(x) = y(a) + y'(a)h$$

Tomando $a = x_n$ e $x = x_{n+1} = x_n + h$, e denotando $y(x_n) = y_n$, e considerando que $y'(x) = f(x, y)$, segue que

$$y(x_{n+1}) = y_{n+1} = y_n + h\, f(x_n, y_n), \quad \text{(b.5)}$$

que consiste no método de Euler (ou Runge-Kutta de primeira ordem).
Note que o erro de truncamento local (em cada interação) é dado por:

$$R_1(x) \le \frac{M}{2}\, h^2 = O(h^2).$$

Considerando o erro acumulado em cada um dos N subintervalos, temos o erro global na ordem $O(h^1)$, o que caracteriza o método como de primeira ordem do ponto de vista do erro global $\mathcal{E}$ de truncamento, pois, sendo $N = \frac{\beta-\alpha}{h}$, exite um número real positivo $\bar{M}$, tal que:

$$\mathcal{E} \le \frac{\bar{M}}{h}\, h^2 = \bar{M}h = O(h).$$

A exemplo do caso anterior, a aproximação

$$y_{n+1} = y_n + \frac{1}{2}h[f(x_n, y_n) + f(x_{n+1}, y_i + hf(x_i, y_i))] \quad \text{(b.6)}$$

satisfaz o polinômio de Taylor de segunda ordem (P_2), sendo o erro de truncamento local (de forma análoga) de ordem $O(h^3)$ e o global de ordem $O(h^2)$. Note que a expressão (b.6) pode ser escrita da forma:

W S Rodrigo

$$y(x_{n+1}) = y_{n+1} = y_n + \frac{1}{2}[k_1 + k_2], \qquad com \tag{b.7}$$
$$k_1 = hf(x_n, y_n)$$
$$k_2 = hf(x_{n+1}, y_n + k_1)$$

Observe que tanto em (b.5) como em (b.7), conhecendo a condição inicial, $y(x_0) = y_0$, podemos obter o valor de y_1 e assim sucessivamente os demais valores y_n, $n = 1, 2, \dots, N$.
Do ponto de vista geométrico, o método de Euler (ou Runge-Kutta de primeira ordem) consiste em obter a reta tangente à curva $y = y(x)$ no ponto conhecido $P_0(x_0, y_0)$. Note que o coeficiente angular dessa reta tangente é dado por $\frac{dy}{dx}(x_0) = f(x_0, y_0)$. A aproximação y_1 é obtida, calculando-se a ordenada do ponto P_1 sobre esta reta que tenha abscissa x_1. Na segunda iteração, obtém-se a reta passando por P_1 com coeficiente angular $\frac{dy}{dx}(x_1) = f(x_1, y_1)$ e assim sucessivamente, até chegar no $N - ésimo$ ponto, conforme ilustrado na Fig. 1.

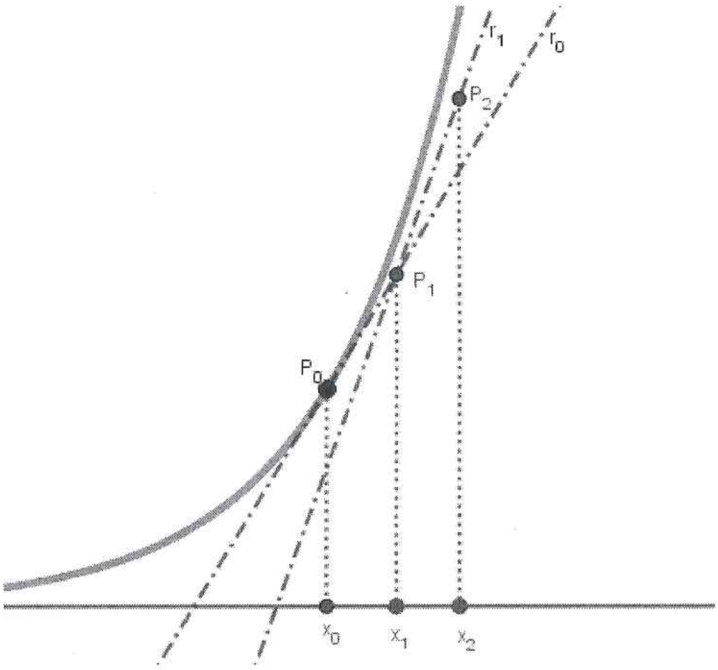


Fig. 01: Esquema da interpretação geométrica do método de Euler

Já o método de Runge-Kutta de segunda ordem consiste em um melhoramento do método de Euler e é obtido ao realizar uma correção na inclinação em um ponto intermediário de forma que (no caso do Runge-Kutta clássico de segunda ordem, esse ponto é obtido com abcissa igual ao valor médio de x_n e x_{n+1}), conforme mostrado na Fig. 2.

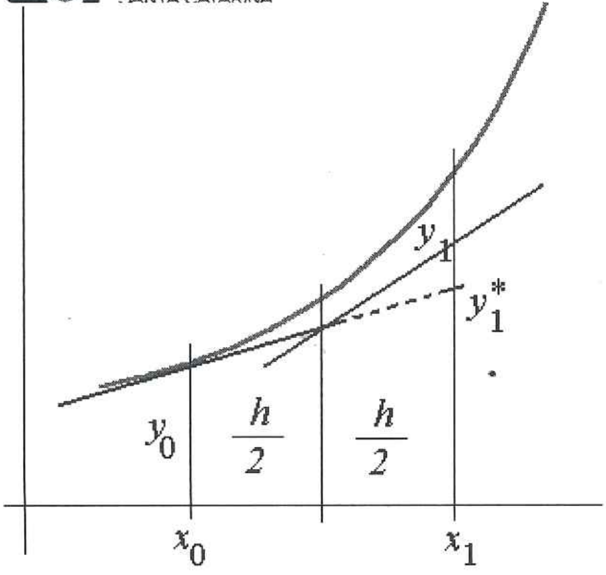


Fig. 02: Esquema da interpretação geométrica do método de Runge-Kutta clássico de segunda ordem.

Pauta de correção:

- Desenvolvimento correto da série de Taylor e obtenção do método de Euler pelo seu truncamento: 25%;
- Extensão do resultado para o método de Runge-Kutta de segunda ordem: 15%;
- Análise do erro de truncamento local e global do método de Euler a partir do resto no truncamento da série de Taylor: 15%;
- Análise do erro de truncamento local e global do método de Runge-Kutta de segunda ordem: 15%;
- Interpretação geométrica do método de Euler:15%;
- Interpretação geométrica do método de Runge-Kutta de segunda ordem: 15%.

c) O método de Runge-Kutta de quarta ordem pode ser escrito da forma:

$$y_{n+1} = y_n + \frac{1}{6}\,[k_1 + 2k_2 + 2k_3 + k_4], \qquad com \tag{c.1}$$
$$k_1 = hf(x_n, y_n)$$
$$k_2 = hf\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right)$$
$$k_3 = hf\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}\right)$$
$$k_4 = hf(x_n + h, y_n + k_3)$$

Para estender o método para um PVI de segunda ordem, consideremos o seguinte problema:

$$\begin{cases} y'' = f(x, y, y') \\ y(x_0) = y_0 \\ y'(x_0) = u_0 \end{cases} \tag{c.2}$$

Fazendo a substituição $y' = u$ na equação diferencial do problema (c.2) temos um sistema de duas equações de primeira ordem:

W S Rebozza

$$\begin{cases} y' = u \\ u' = f(x,y,u) \end{cases} \qquad (c.3)$$

Com condições iniciais $y(x_0) = y_0$ e $u(x_0) = u_0$. Assim, o método de Runge-Kutta aplicado a este sistema fica: (c.4)

$$y_{n+1} = y_n + \frac{1}{6}\,[m_1 + 2m_2 + 2m_3 + m_4]$$

$$u_{n+1} = u_n + \frac{1}{6}\,[k_1 + 2k_2 + 2k_3 + k_4]$$

Onde

$$m_1 = h\,u_n \qquad k_1 = h\,f(x_n, y_n, u_n)$$

$$m_2 = h\left(u_n + \frac{k_1}{2}\right) \qquad k_2 = hf\left(x_n + \frac{h}{2}, y_n + \frac{m_1}{2}, u_n + \frac{k_1}{2}\right)$$

$$m_3 = h\left(u_n + \frac{k_2}{2}\right) \qquad k_3 = hf\left(x_n + \frac{h}{2}, y_n + \frac{m_2}{2}, u_n + \frac{k_2}{2}\right).$$

$$m_4 = h(u_n + k_3) \qquad k_3 = hf(x_n + h, y_n + m_3, u_n + k_3)$$

Pauta de correção:

- Apresentação correta do método de Runge-Kutta de quarta ordem para o PVI de primeira ordem: 25%;
- Redução da equação de segunda ordem para um sistema de duas equações de primeira ordem, aplicando corretamente as condições iniciais: 25%
- Descrição correto do método de Runge-Kutta de quarta ordem para a resolução do sistema equivalente ao PVI de segunda ordem: 50%

d) Considerando o problema de valor inicial (PVI) dado em (01), podemos escrever, usando a notação $\frac{dy}{dt} = y'(t)$:

$$\begin{cases} y'' + 4y = t^2 e^{2t}, \quad t \in [0,1] \\ y(0) = 1 \\ y'(0) = 1 \end{cases} \qquad (d.1)$$

Fazendo a substituição $y' = u$, temos:

$$\begin{cases} y' = u \\ u' = -4y + t^2 e^{2t} \end{cases} \qquad (c.3)$$

com $y(0) = 1$ e $u(0) = 1$.

Algoritmo:

Defina o valor de N (número de partições do intervalo);

Calcule $h = \frac{1}{N}$;

Informe os valores de t_0, y_0 e u_0;

Para $n = 1$ até N, calcule:

$t_n = t_0 + n * h;$

$m_1 = h * u_{n-1};$

$k_1 = h * (-4 * y_{n-1} + (t_{n-1})^2 * e^{2*t_{n-1}});$

$m_2 = h * \left(u_{n-1} + \frac{k_1}{2}\right);$

$k_2 = h * \left(-4 * \left(y_{n-1} + \frac{m_1}{2}\right) + \left(t_{n-1} + \frac{h}{2}\right)^2 * e^{2*\left(t_{n-1} + \frac{h}{2}\right)}\right);$

WSRdozza

$m_3 = h * \left(u_{n-1} + \frac{k_2}{2}\right);$

$k_3 = h * \left(-4 * \left(y_{n-1} + \frac{m_2}{2}\right) + \left(t_{n-1} + \frac{h}{2}\right)^2 * e^{2*\left(t_{n-1}+\frac{h}{2}\right)}\right);$

$m_4 = h * (u_{n-1} + k_3);$

$k_4 = h * \left(-4 * (y_{n-1} + m_3) + (t_{n-1} + h)^2 * e^{2*(t_{n-1}+h)}\right);$

$y_n = y_{n-1} + \frac{1}{6} * (m_1 + 2 * m_2 + 2 * m_3 + m_4);$

$u_n = u_{n-1} + \frac{1}{6} * (k_1 + 2 * k_2 + 2 * k_3 + k_4);$

Fim.

Pauta de correção:

- Escrever o PVI (01) na forma de um sistema de duas equações de primeira ordem, aplicando corretamente as condições iniciais: 30%
- Desenvolvimento de um algoritmo que consiga resolver o PVI (01) usando o método de Runge-Kutta de quarta ordem: 70%.

Obs.: O gabarito apresenta de forma resumida discussões mínimas esperadas em cada item. No gabarito aqui apresentado foi usada a abordagem apresentada em algumas das referências bibliográficas sugeridas. Outras abordagens serão consideradas, desde que respondam corretamente às questões propostas.

2) Considere a equação $e^{-0,5x} = x^2$

a) (20% da questão) Esboce os gráficos das duas funções associadas e aponte o número de raízes da equação.

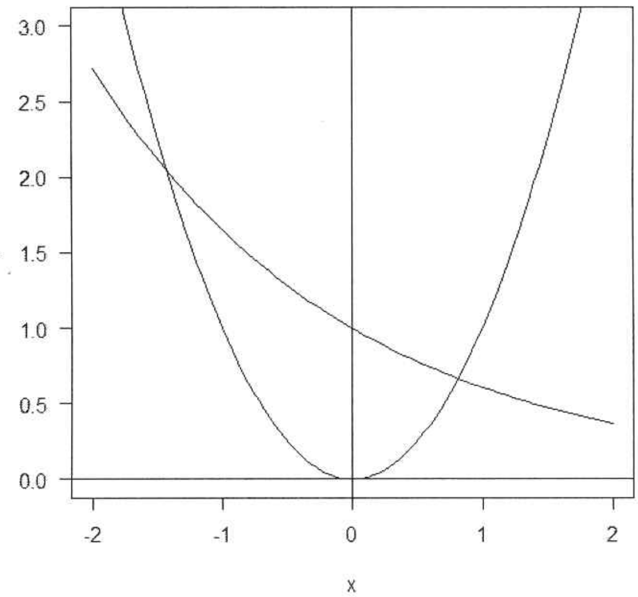
b) (20% da questão) Descreva geometricamente o procedimento de resolução da equação usando os métodos numéricos da bissecção, secante e Newton-Raphson.

c) (30% da questão) Aponte, justificando, estimativas e/ou intervalos iniciais adequados para encontrar cada raiz da equação usando os métodos numéricos da bissecção, secante e Newton-Raphson.

d) (30% da questão) Estime a(s) raiz(es) da equação com precisão de uma casa decimal usando qualquer método numérico.

Gabarito:

a) A equação tem duas raízes como pode ser observado pelos dois pontos de intersecção dos gráficos das duas funções: $f(x) = e^{-0,5x}$ e $g(x) = x^2$.



b) O método da bissecção consiste na estimativa de um intervalo inicial que contém a raiz desejada de uma dada função. No caso, para se estimar o intervalo, deve-se usar a função $f(x) = e^{-0,5x} - x^2$ e selecionar o intervalo (a,b) a partir de pontos a e b para os quais $f(a)f(b) < 0$. Em seguida, o intervalo é dividido pela metade, sendo que as novas extremidades passam pelo mesmo teste. O procedimento é repetido até que o intervalo fique tão pequeno quanto a precisão desejada. Trata-se de um método cuja convergência para a solução da equação é lenta. O método de Newton-Raphson, consiste na obtenção de uma estimativa inicial (x_0) para a raiz da função através de análise gráfica ou considerações sobre o problema considerado. Em seguida, um valor x_1 é estimado através da equação

$$\boldsymbol{x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}}$$

Nesta equação, x_1 consiste no ponto no eixo das abscissas interceptado pela reta tangente à curva $f(x)$, a qual é originada no ponto x_0. O procedimento é repetido sucessivamente até que se atinja a precisão desejada. Portanto, a expressão geral da solução é dada por:

$$\boldsymbol{x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}}$$

Em geral, o método de Newton-Raphson converge bem mais rapidamente para a solução que o

W S Robozza — [illegible]

método da bissecção. O método da secante consiste no uso de uma sequência de raízes de linhas secants para se aproximar o valor da raiz desejada. Nesse caso, são necessárias duas estimativas iniciais e a equação iterativa usada é:

$$x_{n+1} = x_n - \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})} f(x_n)$$

c) Como observado nos gráficos das funções associadas, a equação tem duas raízes, sendo uma positiva e uma negativa. Para a raiz negativa, observa-se que um intervalo adequado é (-2,-1) e para a raiz positiva, observa-se que um interval adequado é (0,1). Para o método de Newton-Raphson, uma estimativa inicial razoável pode ser os valores médios dos intervalos apresentados e para o método da secante, quaisquer dois valores localizados dentro dos dois intervalos são apropriados.

d) Raiz negativa:

Usando o valor $x_0 = -1{,}5$ e pelo método de Newton-Raphson:

$$x_1 = -1{,}5 - \frac{e^{0{,}75} - 2{,}25}{-0{,}5e^{0{,}75} + 3} = -1{,}4$$

Repetindo-se o cálculo usando $x_1 = -1{,}4$, obtém-se o mesmo resultado. Portanto, -1,4 é a aproximação da raiz negativa com uma casa decimal de precisão.

Usando o valor $x_0 = 0{,}5$ e pelo método de Newton-Raphson:

$$x_1 = 0{,}5 - \frac{e^{-0{,}25} - 0{,}25}{-0{,}5e^{-0{,}25} - 1} = 0{,}9$$

$$x_2 = 0{,}9 - \frac{e^{-0{,}45} - 0{,}81}{-0{,}5e^{-0{,}45} - 1{,}8} = 0{,}8$$

Ao se repetir o cálculo, obtém-se o mesmo resultado, Portanto, 0,8 é a aproximação da raiz positiva com uma casa decimal de precisão.

WSRolozze

3) Utilizando algoritmo do tipo pseudo-linguagem (pseudocódigo), escreva as seguintes funções: (observação: assuma que a manipulação de arrays - vetores e/ou matrizes - ocorrerá apenas por acesso a elementos individuais, através de seus respectivos índices).

a) (25% da questão) Uma função que recebe dois números naturais N, M como argumento, e solicita que o usuário informe os coeficientes de uma matriz N×M, retornando a matriz como resultado.

b) (20% da questão) Uma função que recebe uma matriz A e efetua a soma das linhas da matriz.

c) (25% da questão) Uma função que recebe uma matriz A e retorna o maior elemento, em módulo, de cada coluna da matriz.

d) (30% da questão) Escreva uma função que recebe como argumento um vetor qualquer com *n* elementos, e retorna outro vetor com *n*–1 elementos, em que cada elemento é dado pela média de dois elementos consecutivos do vetor original.

Gabarito:

Item a:

```
função numérico cria_matriz(N,M)
        declare A[1:N,1:M] numérico
        declare N,M,i,j numérico
        i <- 1
        repita
            se i>N
               então interrompa
            fim se
            j <- 1
            repita
                se j>M
                   então interrompa
                fim se
                leia A[i,j]
                j <- j + 1
            fim repita
            i <- i + 1
        fim repita
        cria_matriz <- A
fim função
```

WSRobazza

Item b:

```
função numérico somas(A,N,M)
        declare somas[1:N] numérico
        declare N,M,i,j numérico
        declare A[1:N,1:M] numérico
        i <- 1
        repita
            se i>N
               então interrompa
            fim se
            somas[i] <-0
            j <- 1
            repita
                se j>M
                   então interrompa
                fim se
                somas[i] <- somas[i] + A[i,j]
                j <- j + 1
            fim repita
            i <- i + 1
        fim repita
fim função
```

Item c:

```
função maiores(A,N,M)
        declare maiores[1:M] numérico
        declare N,M,i,j numérico
        declare A[1:N,1:M] numérico
        j <- 1
        repita
           se j>M
              então interrompa
           fim se
           maiores[j] <- A[1,j]
           i <- 2
           repita
               se i>N
                  então interrompa
               fim se
               se A[i,j]^2 > maiores[j]^2
                  então maiores[j] <- A[i,j]
               fim se
               i <- i + 1
           fim repita
           j <- j + 1
        fim repita
        retorna maiores
fim função
```

UDESC
UNIVERSIDADE
DO ESTADO DE
SANTA CATARINA

Item d:

```
função média(v,N)
        declare v2[1:N-1] numérico
        declare N,i numérico
        declare v[1:N] numérico
        i <- 1
        repita
            se i>N-1
                então interrompa
            fim se
            v2[i] <- (v[i]+v[i+1])/2
            i <- i + 1
        fim repita
        média <- v2
fim função
```

Membros da Banca:

Avaliador 1 (nome e assinatura)
WEBER DA SILVA ROBAZZA

Avaliador 2 (nome e assinatura)
TONY JEFFERSON LOPES

Avaliador 3 (nome e assinatura)
VITOR JOSÉ PETRY

WEBER DA SILVA ROBAZZA
Presidente da Banca (nome e assinatura)