

**Ano Letivo:** 2022/2023

**Atividade 03-Métodos Numéricos para EDO/PVI**

**Relatório**

Análise Matemática II

**Docente:** Arménio Correia

**Trabalho realizado por:**

Martim Alexandre Vieira Antunes

**nº:** 2022141890 **Curso:** LEI

Pedro Lino Neves Faneca

**nº:** 2022134142 **Curso:** LEI

**Índice**

1.Introdução-----------------------------------------------------------------------------------------------3

1.1-Equação diferencial e propriedades---------------------------------------------------3

1.2-Definição de PVI----------------------------------------------------------------------------4

2.Métodos numéricos para a resolução de PVI---------------------------------------------------5

2.1-Método de Euler----------------------------------------------------------------------------6

2.1.1-Fórmulas-------------------------------------------------------------------------6

2.1.2-Algoritmo/Função-------------------------------------------------------------6

2.2-Método de Euler Melhorado ou Modificado----------------------------------------7

2.2.1-Fórmulas-------------------------------------------------------------------------7

2.2.2-Algoritmo/Função-------------------------------------------------------------8

2.3-Método de RK2-----------------------------------------------------------------------------9

2.3.1-Fórmulas-------------------------------------------------------------------------9

2.3.2-Algoritmo/Função-------------------------------------------------------------10

2.4-Método de RK4----------------------------------------------------------------------------11

2.4.1-Fórmulas------------------------------------------------------------------------11

2.4.2-Algoritmo/Função------------------------------------------------------------12

2.5-Função ODE45 do Matlab---------------------------------------------------------------13

2.5.1-Fórmulas------------------------------------------------------------------------13

2.5.2-Algoritmo/Função------------------------------------------------------------13

2.6-Método do Ponto Médio----------------------------------------------------------------14

2.6.1-Fórmulas------------------------------------------------------------------------14

2.6.2-Algoritmo/Função------------------------------------------------------------15

3.Exemplos de aplicação e teste dos métodos---------------------------------------------------16

3.1-Exercício 3 do Teste Farol---------------------------------------------------------------16

3.1.1-PVI-Equação Diferencial de 1ªordem e Condições Iniciais---------16

3.1.2-Exemplos de output-App com gráfico e tabela-----------------------17

3.2-Problemas de aplicação do livro------------------------------------------------------21

3.2.1-Modelação matemática do problema-----------------------------------21

3.2.2-Resolução através da App desenvolvida--------------------------------22

3.3-Problemas de aplicação da alínea 2.b do teste Farol----------------------------25

3.3.1- Modelação matemática do problema----------------------------------25

3.3.2- Resolução através da App desenvolvida-------------------------------26

4.Conclusão------------------------------------------------------------------------------------------------27

5.Bibliografia----------------------------------------------------------------------------------------------28

6.Autoavaliação e heteroavaliação------------------------------------------------------------------29

**1.Introdução**

Este trabalho surge do âmbito da unidade curricular de Análise Matemática 2, do curso de Engenharia Informática do Instituto Superior de Engenharia de Coimbra.

O seu foco consiste no estudo de Métodos Numéricos para a resolução de Equações Diferenciais Ordinárias (EDOs) e de Problemas de Valor inicial (PVIs), e na implementação desses métodos através do desenvolvimento de uma app em linguagem de programação MATLAB.

Para uma melhor familiarização com estes conteúdos, incluímos exemplos de aplicação e testes dos métodos numéricos analisados.

**1.1-Equação diferencial: definição e propriedades**

As equações diferenciais são aquelas que contêm as derivadas de uma ou mais variáveis dependentes em relação a uma ou mais variáveis independentes.

As equações diferenciais são classificadas quanto ao **tipo**, **ordem** e **linearidade**.

* Quanto ao tipo as equações diferenciais são classificadas em: ordinárias e parciais. Equações diferenciais ordinárias (EDO) são aquelas que contêm uma ou mais derivadas de variáveis dependentes em relação a uma variável independente. As equações diferenciais parciais (EDP) são aquelas que envolve as derivadas parciais de uma ou mais variáveis dependentes em relação a uma ou mais variáveis independentes.
* Quanto a ordem uma equação diferencial pode ser de 1ª, 2ª,…,n-ésima ordem dependendo da derivada de maior ordem presente na equação. Uma equação ordinária de ordem n pode ser escrita na forma:

{F(t,y,y^{\prime},y^{\prime \prime}...y^{(n)})=0}

* Quanto a linearidade de uma equação diferencial ela pode ser linear e não linear. Ela é linear se as incógnitas e suas derivadas aparecem de forma linear. Por exemplo uma equação diferencial ordinária de ordem n é uma equação que pode ser escrita como:

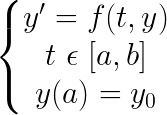


As equações diferenciais ordinárias que não podem ser escritas nessa forma são não lineares.

**1.2-Definição de PVI**

Um PVI significa Problema de Valor Inicial e é um conceito da matemática aplicada e da análise numérica. Em termos simples, um PVI é uma equação diferencial que descreve o comportamento de uma função desconhecida, juntamente com uma condição inicial que especifica o valor da função em um ponto específico. O objetivo do problema é encontrar uma solução que satisfaça a equação diferencial e a condição inicial. Os PVIs podem ser resolvidos de forma exata ou aproximada.

Um problema de valor inicial (PVI) é uma equação diferencial da forma:



**2.Métodos Numéricos para a resolução de PVI**

Existem vários métodos numéricos para a resolução de PVI entre eles:

* Método de Euler: Este é o método mais simples e direto para resolver um PVI. Ele usa uma aproximação linear para a solução e é fácil de implementar. No entanto, a precisão deste método é limitada e pode gerar erros significativos em soluções complexas.
* Método de Runge-Kutta: Este é um método iterativo que usa várias estimativas de ordem superior para melhorar a precisão da solução. O método de Runge-Kutta é mais preciso do que o método de Euler, mas é mais complexo de implementar.
* Método de Adams-Bashforth: Este método é uma abordagem de múltiplas etapas que usa as soluções anteriores para estimar a solução atual. Ele tem uma precisão moderada e é fácil de implementar.
* O método de Heun, também conhecido como método de Euler moderado, é um método numérico para resolver problemas de valor inicial (PVI) de equações diferenciais ordinárias (EDOs). Ele é uma melhoria do método de Euler, que consiste em aproximar a solução da EDO por meio de uma reta tangente ao ponto inicial.

Esses são apenas alguns dos métodos disponíveis para resolver PVI. A escolha do método a ser usado depende da complexidade da equação diferencial e das condições iniciais, bem como dos requisitos de precisão e eficiência computacional.

Neste trabalho iremos abordar os métodos de **Euler**, **Euler Moderado**, **Runge-Kutta de ordem 2 e 4**, a **função ODE45** e ainda o **Ponto Médio**.

**2.1-Método de Euler**

O método de Euler é um dos métodos numéricos mais simples e amplamente utilizados para resolver problemas de valor inicial (PVI) de equações diferenciais ordinárias (EDOs). Ele consiste em aproximar a solução da EDO por meio de uma reta tangente ao ponto inicial, no entanto, é importante lembrar que ele tem limitações em relação à precisão e pode não ser a melhor escolha em todos os casos.

**2.1.1-Fórmulas**

A fórmula do Método de Euler para resolver um PVI é:

yi+1= yi+h\*f(ti,yi) ,i=0,1,…,n-1

**Legenda:**

* *yi+1* = Próximo valor aproximado da solução do problema original (na abcissa *ti+1*);
* *yi* = Valor aproximado da solução do problema original na abcissa atual;
* *h* = Valor de cada subintervalo (passo);
* *f(ti ,yi)*  = Valor da equação em *ti* e *yi*;

**2.1.2-Algoritmo/Função**

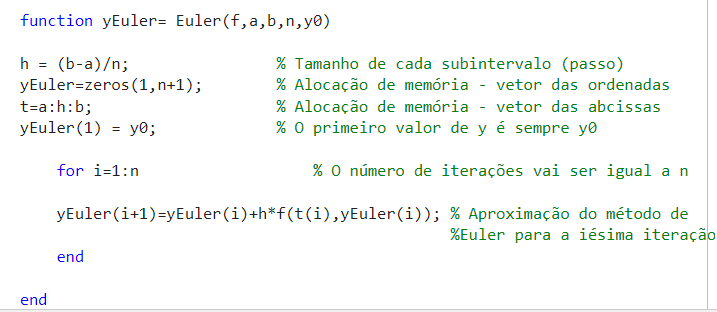
**Algoritmo:**

* Ler f, a, b, y0, n;
* Calcular h (h =(b-a)/n);
* t = a:h:b;
* y = y0;
* Para i de 1 até n fazer

y=y+h\*f(t,y);

* Escrever y;

**Função (MATLAB):**

****

**2.2 Método de Euler Moderado**

O método de Euler moderado, ou método de Heun, é um método numérico para resolver equações diferenciais ordinárias (EDOs) de primeira ordem. Ele é uma melhoria do método de Euler simples, que é mais simples de implementar, mas pode produzir soluções menos precisas.

**2.2.1-Fórmulas**

**Fórmula Geral**

yi+1= yi + h/2\*(k1+k2) ,i=0,1,…,n-1

**Legenda:**

* yi+1 = Próximo valor de y(valor aproximado da solução ao problema) na abcissa *ti+1* ;
* *yi* = Valor aproximado da solução do problema na abcissa atual(abcissa *ti)*;
* *h* = Valor de cada subintervalo (passo);
* *k1 =* Inclinação no início do intervalo;
* *k2 =*  Inclinação no fim do intervalo;

**Fórmula para calcular k1**

k1 = f(ti,yi)

**Legenda:**

* *k1* = Inclinação no início do intervalo
* *f(ti ,yi)* = Valor da equação em *ti* e *yi*;

**Fórmula para calcular k2**

k2 = f(tt+1, yi + k1\*h)

**Legenda:**

* *k2* = Inclinação no fim do intervalo;
* *ti+1* = Valor da abcissa seguinte;
* yi = Valor aproximado da solução do problema original na abcissa atual;
* *k1* = Inclinação no início do intervalo;
* *h* = Valor de cada subintervalo (passo);

**2.2.2-Algoritmo/Função**

**Algoritmo:**

* Ler f, a, b, n, y0;
* Calcular h (h =(b-a)/n);
* t = a:h:b;
* y = y0;
* Para i de 1 até n fazer:

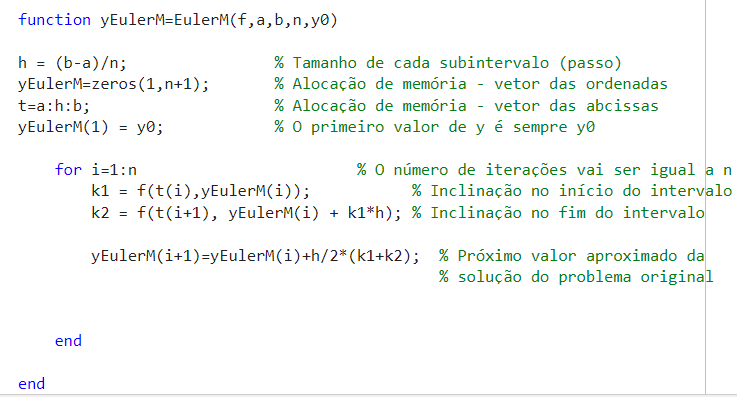
k1 = f(ti,yi);

k2 = f(tt+1, yi + k1\*h);

yi+1= yi + h/2\*(k1+k2);

* Escrever yi+1

**Função (MATLAB):**

****

**2.3 Método de Runge-Kutta de 2ª Ordem**

O método de Runge-Kutta de 2ª ordem é um algoritmo numérico utilizado para aproximar soluções de equações diferenciais ordinárias de primeira ordem. Ele utiliza duas estimativas para calcular a solução em cada passo de tempo, resultando numa precisão maior do que a do método de Euler.

**2.3.1-Fórmulas**

**Fórmula Geral**

yi+1= yi + \*(k1+k2) ,i=0,1,…,n-1

**Legenda:**

* yi+1 = Próximo valor de y(valor aproximado da solução ao problema) na abcissa *ti+1* ;
* *yi =* Valor aproximado da solução do problema na abcissa atual(abcissa *ti)*;
* *k1* = Inclinação no início do intervalo;
* *k2* = Inclinação no fim do intervalo;

**Fórmula para calcular k1**

k1 =h\* f(ti,yi)

**Legenda:**

* *k1* = Inclinação no início do intervalo
* *f(ti ,yi)* = Valor da equação em *ti* e *yi*;
* *h* = Valor de cada subintervalo (passo);

**Fórmula para calcular k2**

k2 = h\*f(tt+1, yi + k1)

**Legenda:**

* k2 = Inclinação no fim do intervalo;
* ti+1 = Valor da abcissa seguinte;
* yi = Valor aproximado da solução do problema original na abcissa atual;
* k1 = Inclinação no início do intervalo;
* h = Valor de cada subintervalo (passo);

**2.3.2-Algoritmo/Função**

**Algoritmo:**

* Ler f, a, b, n, y0;
* Calcular h (h =(b-a)/n);
* t= a:h:b;
* y = y0;
* Para i de 1 até n fazer:

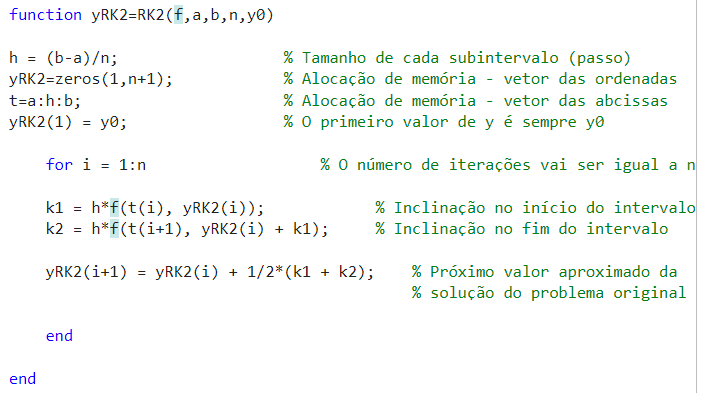
k1 = h\*f(ti,yi);

k2 =h\* f(tt+1, yi + k1);

yi+1= yi + \*(k1+k2);

* Escrever yi+1

**Função (MATLAB):**

****

**2.4-Método de Runge-Kutta de 4ª Ordem**

O método de Runge-Kutta de 4ª ordem é um algoritmo numérico utilizado para aproximar soluções de equações diferenciais ordinárias, que utiliza quatro estimativas intercaladas para calcular a solução em cada passo de tempo. O método de Runge-Kutta de 4ª ordem é mais preciso do que o método de Runge-Kutta de 2ª ordem para a aproximação da solução numérica de equações diferenciais ordinárias.

**2.4.1-Fórmulas**

**Fórmula Geral**

yi+1= yi + \*(k1+2\*k2+2\*k3+k4) ,i=0,1,…,n-1

**Legenda:**

* yi+1 = Próximo valor de y(valor aproximado da solução ao problema) na abcissa *ti+1* ;
* *yi =* Valor aproximado da solução do problema na abcissa atual(abcissa *ti)*;
* k1 = Inclinação no início do intervalo;
* k2 = Inclinação no ponto médio do intervalo;
* k3 = Inclinação no ponto médio do intervalo;
* k4 = Inclinação no final do intervalo

**Fórmulas para calcular k1,k2,k3,k4**

k1 =h\* f(ti,yi)

k2 = h\*f(ti+ ,yi +)

k3 = h\*f(ti+ ,yi +)

k4 = h\*f(ti+h ,yi +k3)

**2.4.2-Algoritmo/Função**

**Algoritmo:**

* Ler f, a, b, n, y0;
* Calcular h (h =(b-a)/n);
* t=a:h:b;
* y= y0;
* Para i de 1 até n fazer:

k1 = h\*f(ti,yi);

k2 =h\* f (ti+ ,yi +);

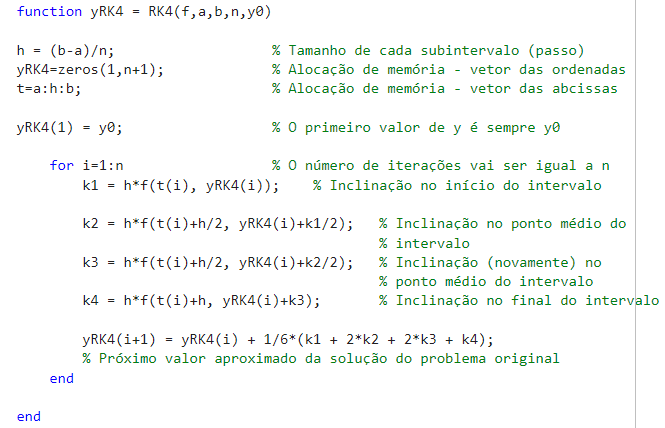
k3 = h\*f(ti+ ,yi +);

k4 = h\*f(ti+h ,yi +k3);

yi+1= yi + \*(k1+2\*k2+2\*k3+k4);

* Escrever yi+1

**Função (MATLAB):**

****

**2.5-Função ODE45 do Matlab**

A função ode45 nativa do Matlab é um método numérico utilizado para resolver equações diferenciais ordinárias de primeira ordem e segunda ordem, que é um método de passo variável baseado num método de Runge-Kutta.

**2.5.1-Fórmulas**

[t, y] = ode45(f, t, y0)

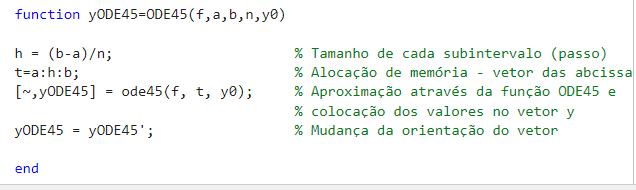
**Legenda:**

* *t*  Vetor das abcissas;
* *f*  Equação diferencial de *t* e y;
* *y0*  Condição inicial do PVI (valor inicial de y);

**2.5.2-Algoritmo/Função**

**Algoritmo:**

* Ler f, a, b, n, y0;
* Calcular h (h=(b-a)/n);
* t= a:h:b
* y= y0;
* Aproximar as soluções através da função ODE45;

**Função (MATLAB):**

**2.6-Método do Ponto Médio**

O método do Ponto Médio é um método numérico para resolver Equações Diferenciais Ordinárias (ODE).

**2.6.1-Fórmulas**

**Formula Geral**

yi+1= yi + h\* f(ti+h/2,yi+h\*k1), i=0,1,…,n-1

**Fórmula para calcular k1**

k1 = f(ti,yi)

**2.6.2-Algoritmo/Função**

**Algoritmo:**

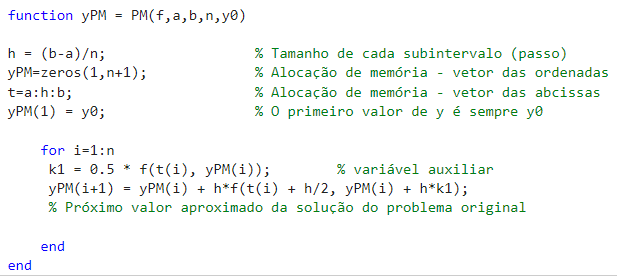
* Ler f, a, b, n, y0;
* Calcular h (h =(b-a)/n);
* t= a:h:b;
* y= y0;
* Para i de 1 até n fazer:

k1 =f(ti,yi);

yi+1= yi + h\*f(ti+h/2, yi +h\*k1);

* Escrever yi+1

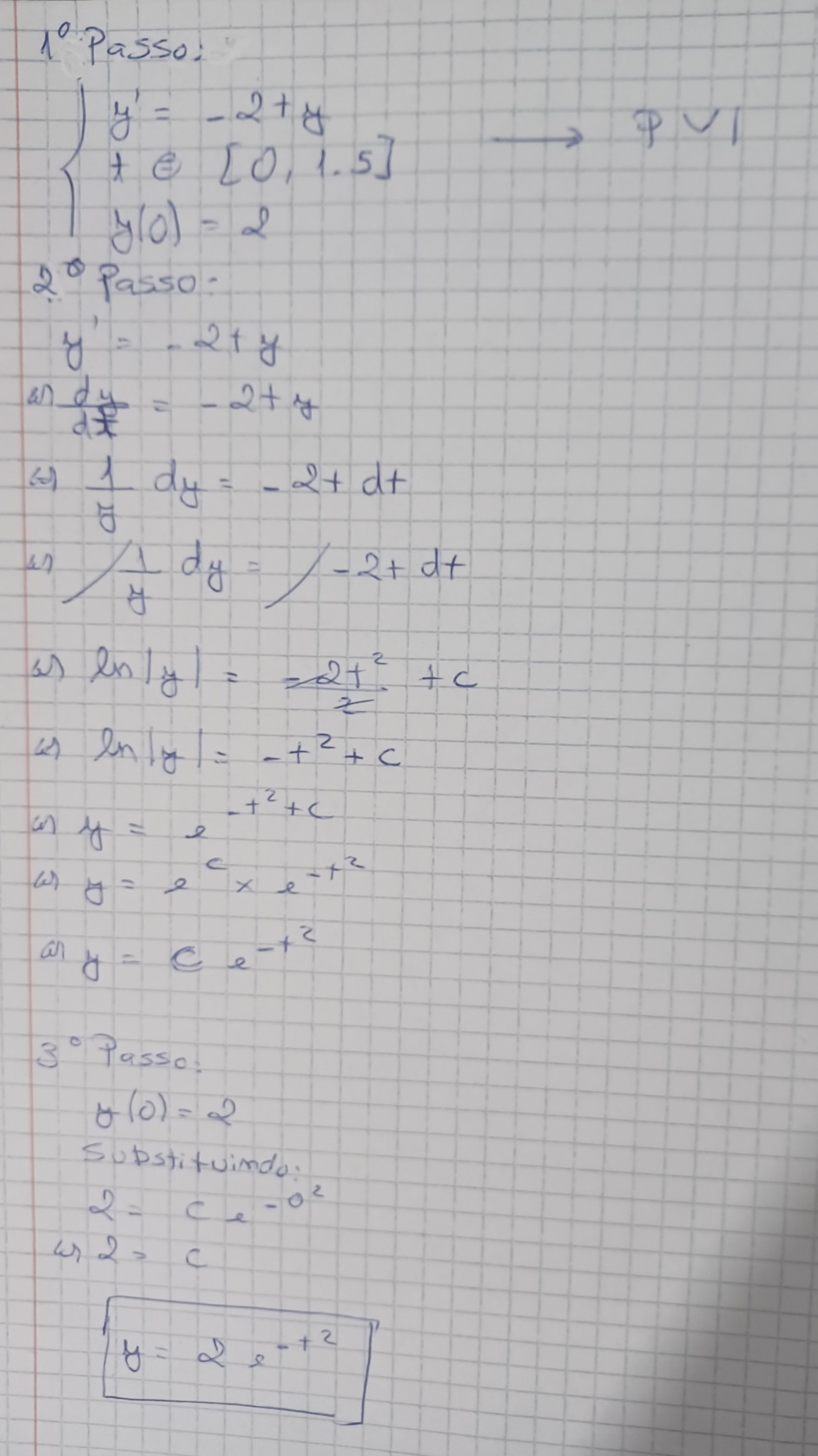
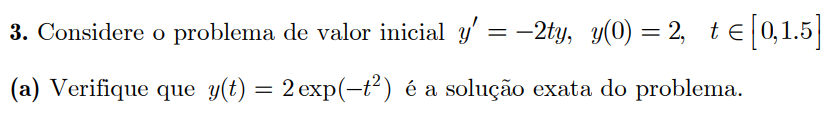
**Função:**

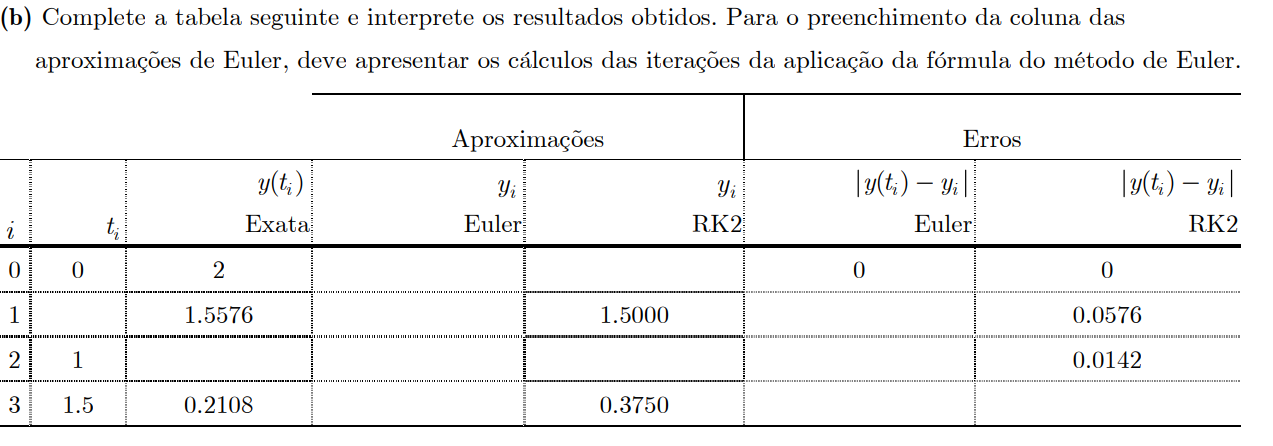
****

**3.Exemplos de aplicação e teste dos métodos**

**3.1-Exercício 3 do Teste Farol**

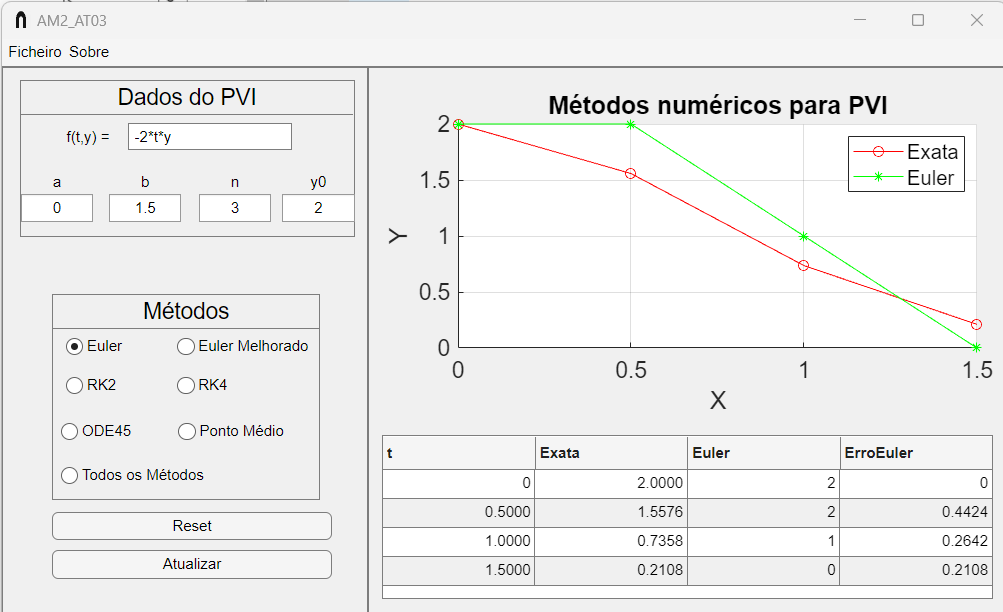
**3.1.1-PVI - Equação Diferencial de 1ª ordem e Condições Iniciais**

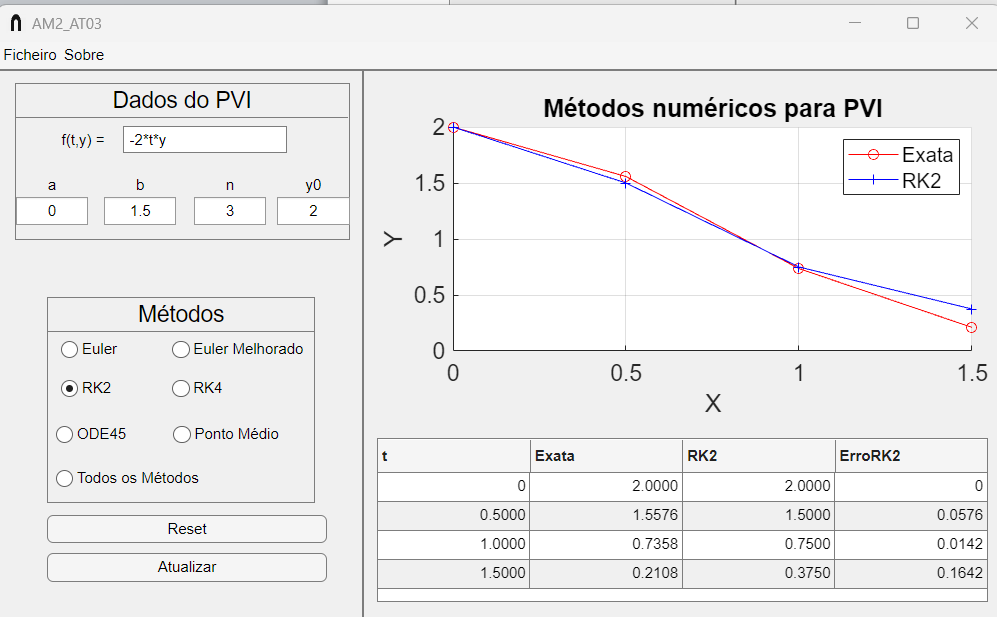
****

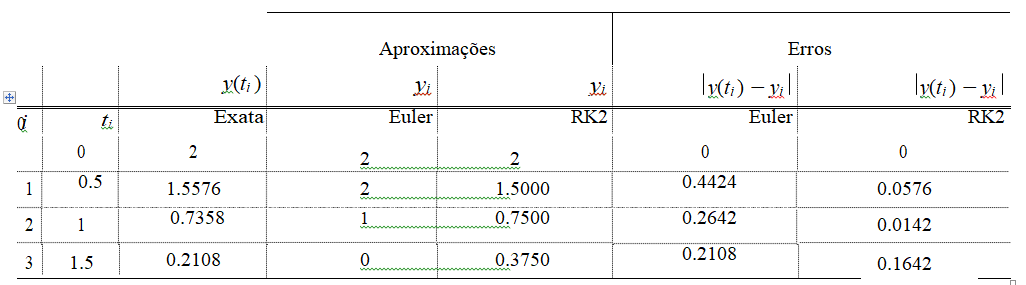
**3.1.2-Exemplos de output - App com gráfico e tabela**

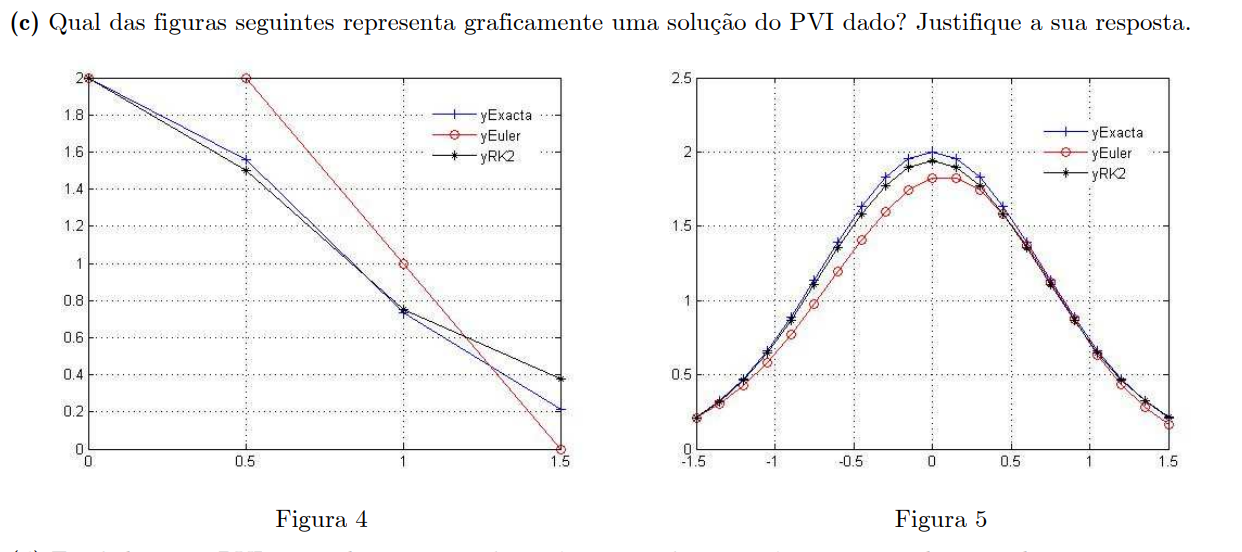
Pela tabela verificamos que h = 0.5, a = 0 e b = 1.5.

Como h = (b-a)/n, logo n = 3

**Euler**

**RK2**

**Tabela preenchida**

****

Como podemos ver pelos gráficos obtidos na nossa aplicação na alínea anterior, a figura 4 é a que representa corretamente uma solução do PVI dado.

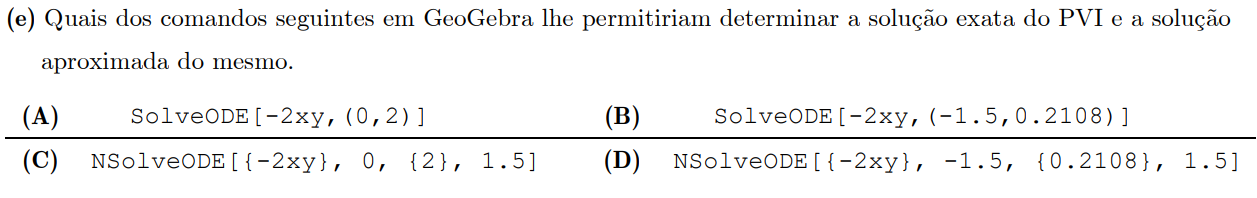


**PVI**

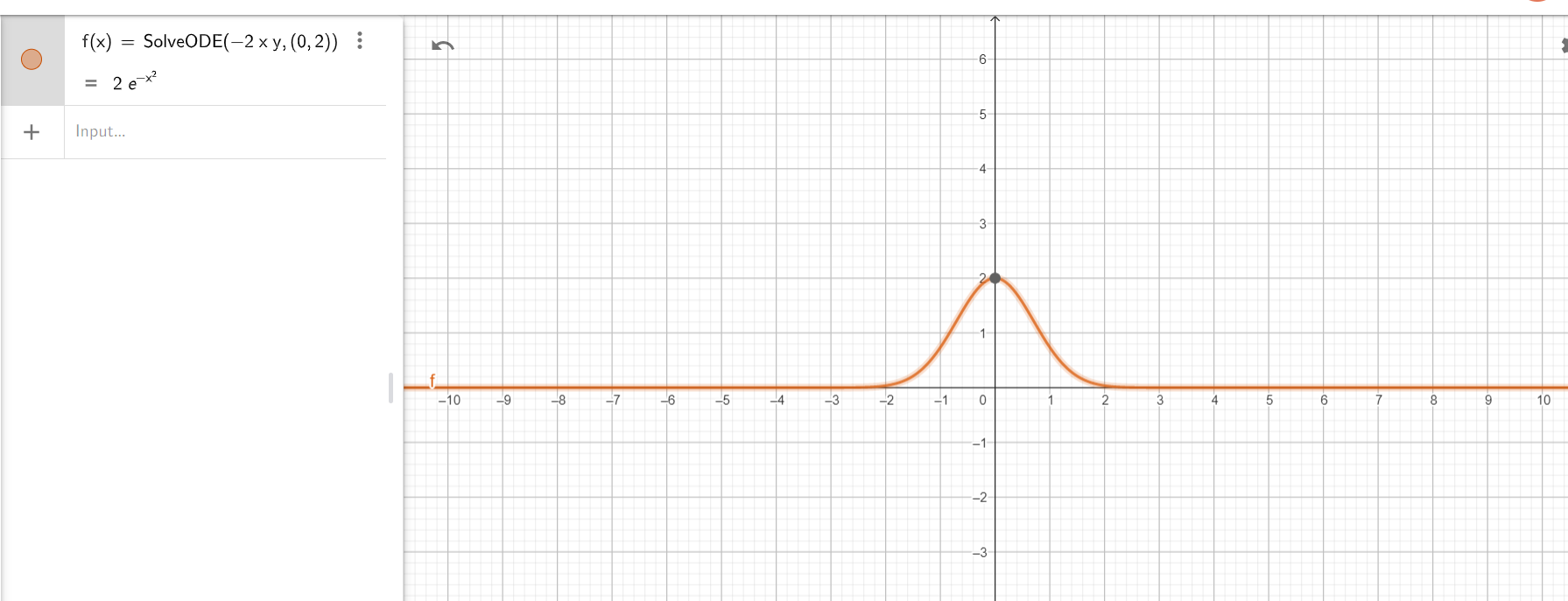
y’ = -2\*t\*y

t

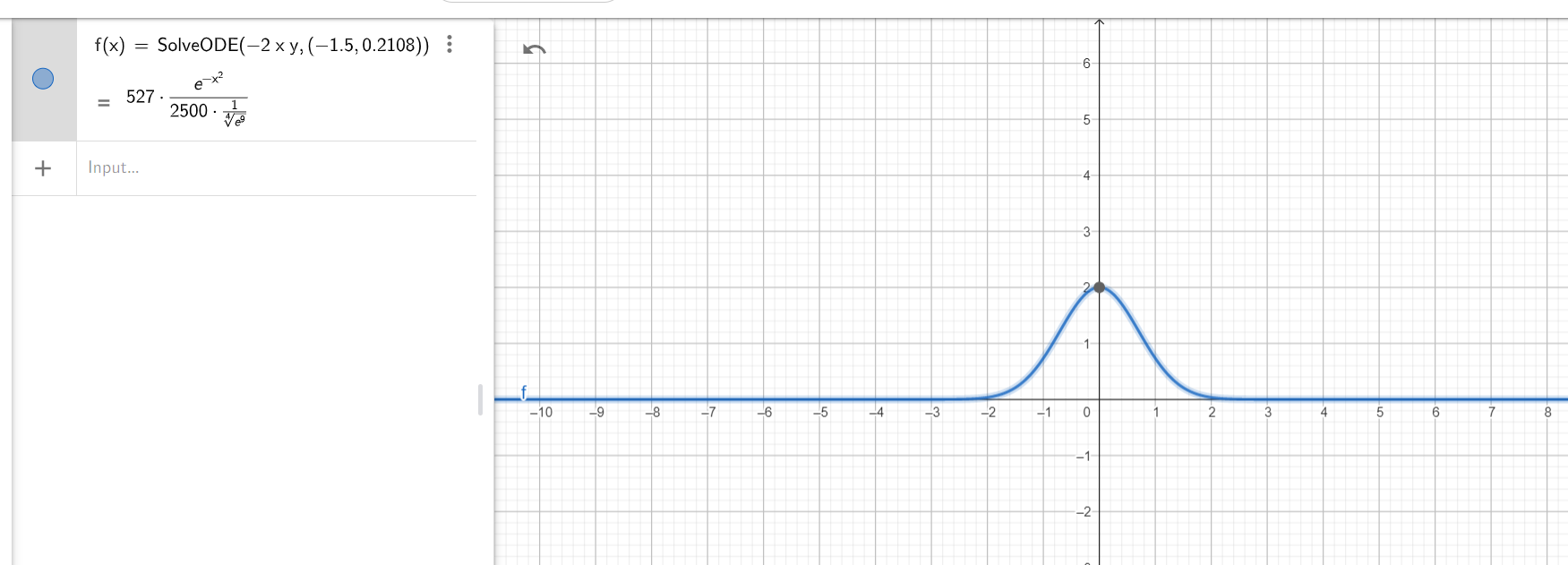
y(0) = 0.2108



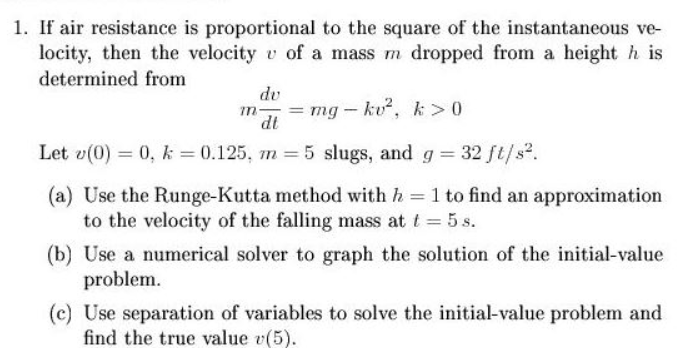
(A) - Solução exata



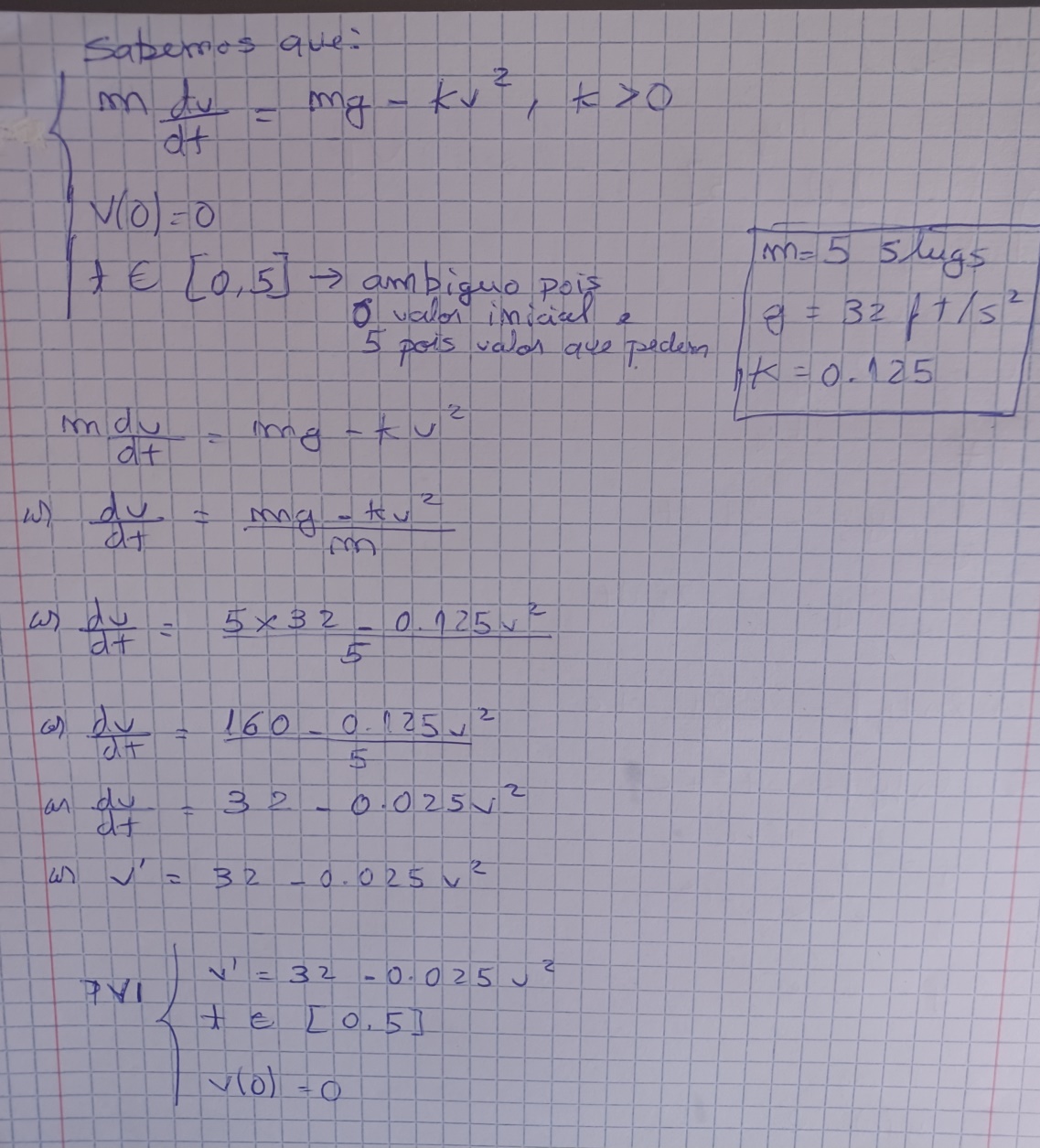
1. - Solução aproximada



**3.2-Problemas de aplicação do livro “Differential Equations with Modeling Applications”**

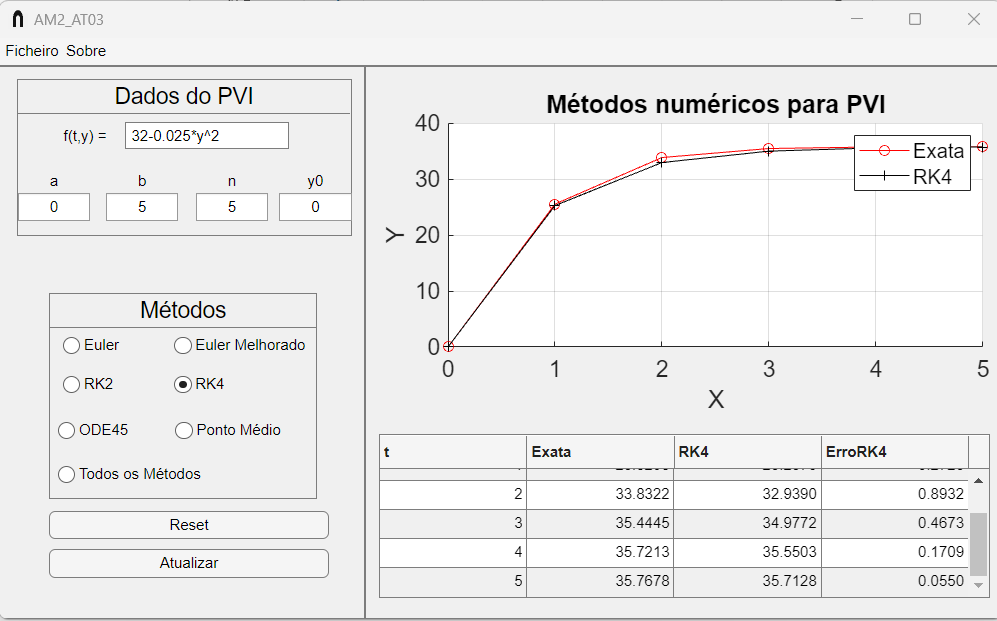
****

**3.2.1-Modelação Matemática do problema**



**3.2.2-Resolução através da App Desenvolvida**

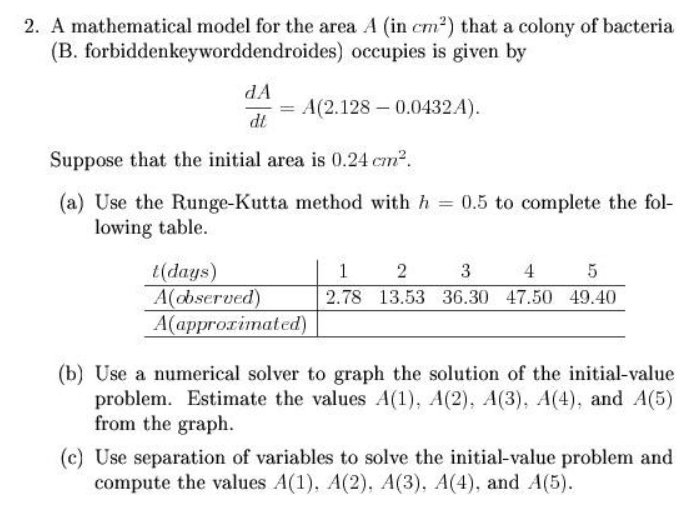
Usámos o método de Runge-Kutta de ordem 4 (RK4) em vez do Rk2 pois é mais preciso, o que conduzes a um erro menor.



Resposta a)- Gráfico da solução do PVI

Resposta b)- aproximação pelo método de Runge-Kutta da velocidade da massa em queda em ***t*** = 5s

Resposta c)- Valor real de v(5)

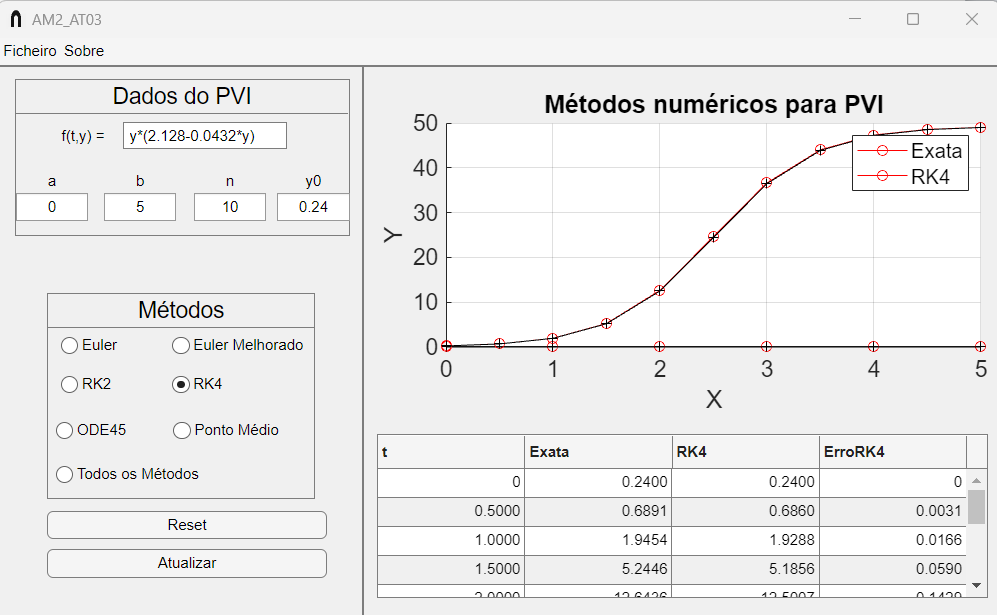


PVI

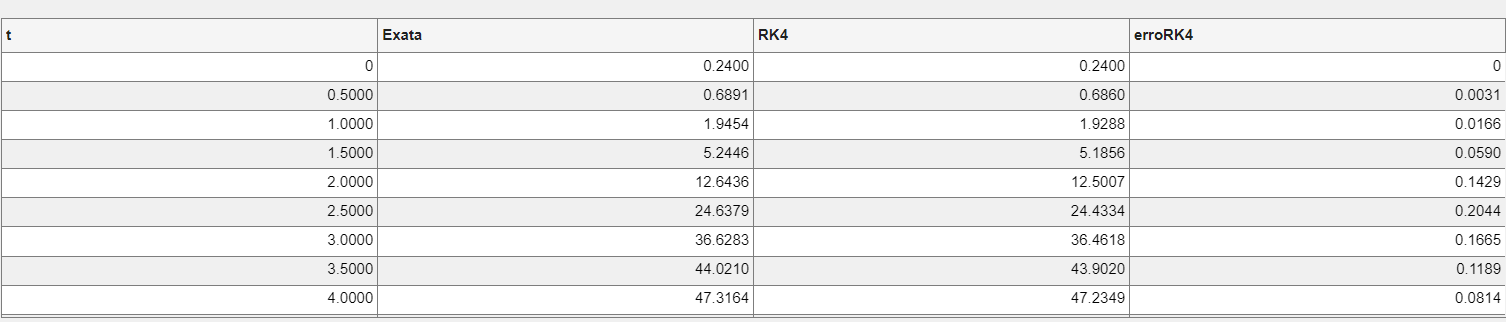
y’ = y\*(2.128-0.0432\*y)

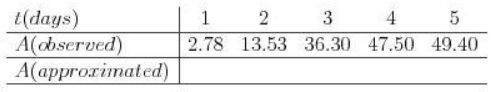
t

y(0) = 0.24

Como h=0.5, a=0, b=5 , pela formula h = (b-a)/n, o valor de **n = 10**

Pela app podemos retirar os valores que precisamos para preencher a tabela



****

1.93

49.00

47.23

36.46

12.50

49.02

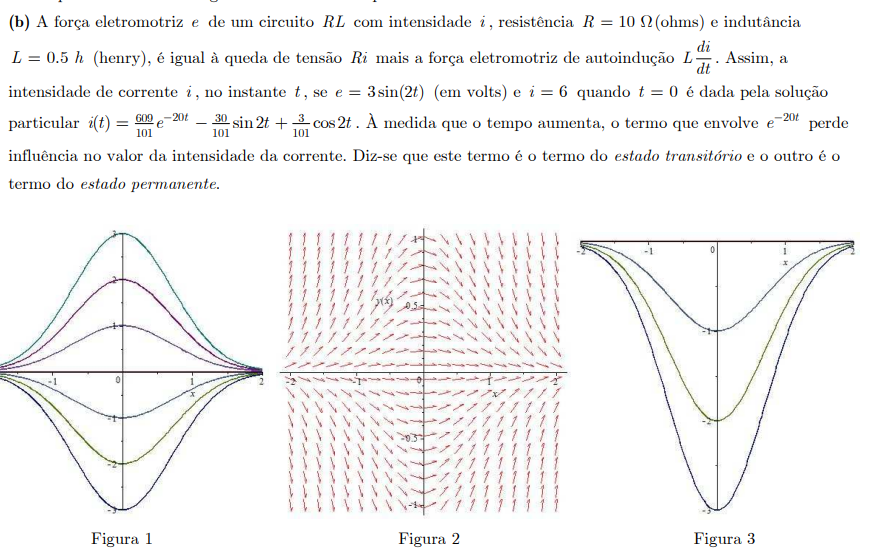
47.32

36.63

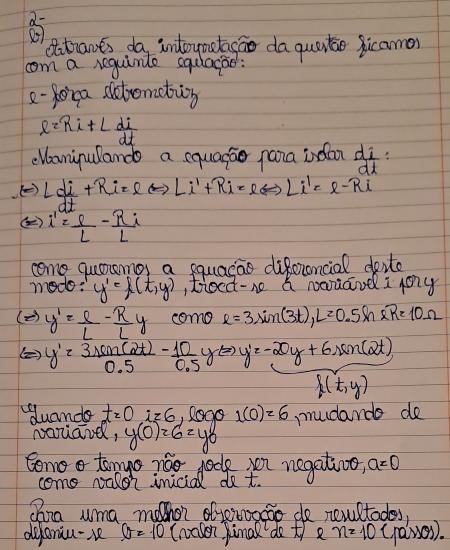
12.64

1.95

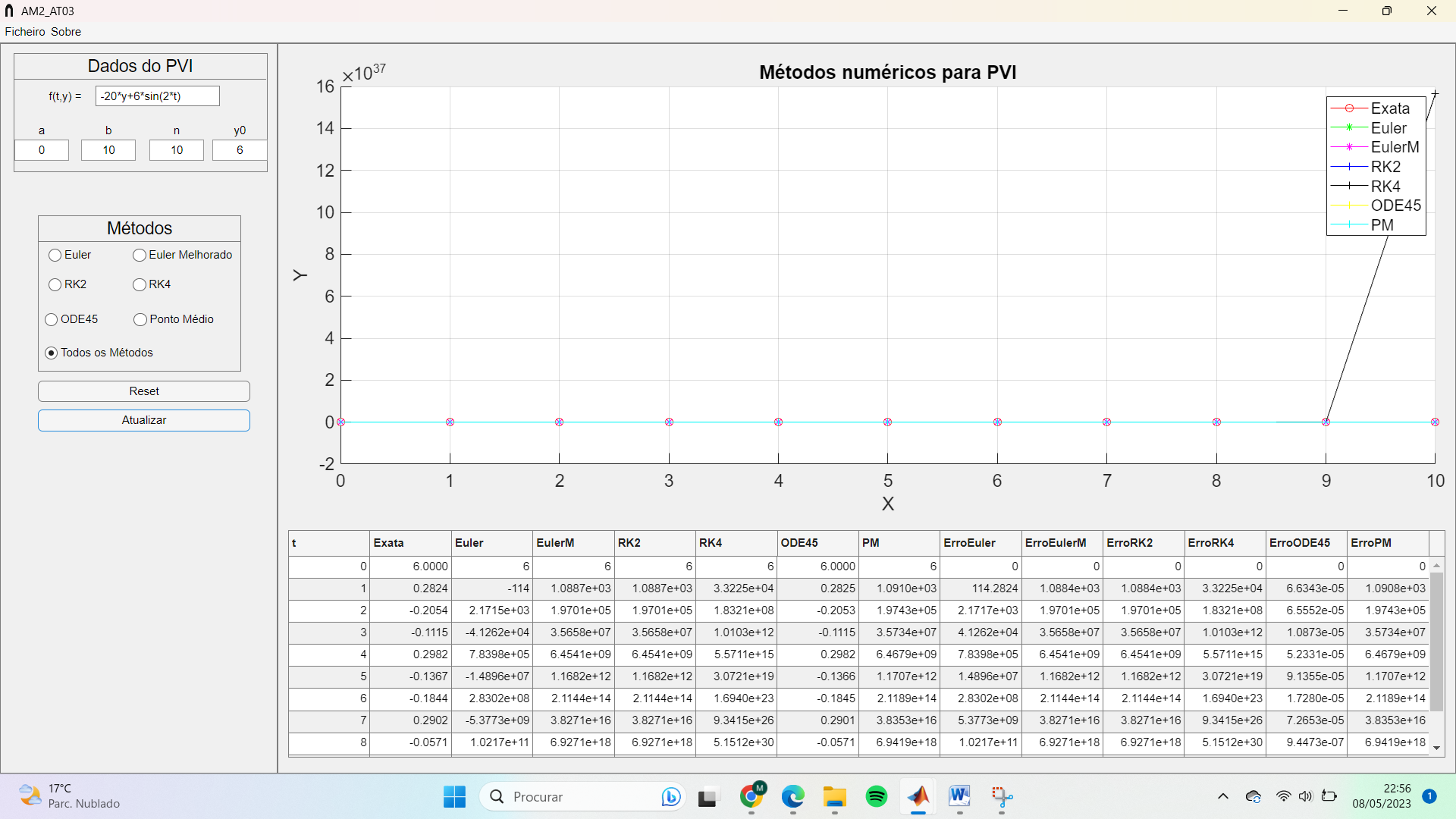
A (exact)

**3.3 Problemas de aplicação da alínea 2.b do teste Farol**

**3.3.1-Modelação matemática do problema**



**3.3.2-Resolução através da app desenvolvida**

Com a informação retirada e explicada anteriormente, utilizámos a nossa app com todos os métodos implementados.

**4.Conclusão**

Em suma, os métodos numéricos são uma ferramenta poderosa e indispensável para a solução de problemas matemáticos e científicos na atualidade. Porém, é importante ressaltar que a utilização dos métodos numéricos requer um conhecimento sólido de matemática e programação, além de cuidados na escolha do método e dos parâmetros a serem utilizados para evitar erros e imprecisões nos resultados obtidos.

Comparando os diferentes métodos, observamos que os que verificam menor erro e, consequentemente, melhor aproximação ao valor exato, são o método de Runge-Kutta de ordem 4 e o método usando a função ODE45 do MATLAB, que muitas vezes apresentaram erros muito pequenos (milésimas). Em contrapartida, temos o método de Euler, cujo erro é especialmente grande comparado com todos os outros métodos implementados.

Com este trabalho, adquirimos várias técnicas, não só de programação de Matlab como também ficamos a entender melhor ainda, a matéria dos métodos numéricos lecionada nas aulas.

**5.Bibliografia**

* Ficheiros de suporte disponibilizados pelo professor
* Formulário da cadeira
* Métodos Numéricos (Visualizado a 17 de abril de 2023)

**Disponível em:** https://paginas.fe.up.pt/~faf/mnum/mnum-faf-handout.pdf

* Midpoint method-Wikipedia (Visualizado a 17 de abril de 2023)

**Disponível em:** <https://en.wikipedia.org/wiki/Midpoint_method>

* Equações Diferencias-Método de Heun (Visualizado a 15 de abril de 2023

**Disponível em:** <https://cn.ect.ufrn.br/index.php?r=conteudo%2Fedo-heun>

* Problema de valor inicial-Wikipedia (Visualizado a 23 de abril de 2023)

**Disponivel em:** <https://pt.wikipedia.org/wiki/Problema_de_valor_inicial>

* Introdução as equações diferenciais-Luso Academia (Visualizado a 20 de abril de 2023)

Disponível em: <https://lusoacademia.org/2015/11/19/1-introducao-as-equacoes-diferenciais/>

**6.Autoavalição e heteroavaliação**

Chegando ao fim do trabalho, estamos contentes pelo resultado final, tanto a nível da app quanto ao nível da LiveScript implementada.

Pelo esforço e trabalho aplicados a esta atividade, achamos que merecemos numa escala de 0 a 5 valores, um 4.5.

A nível de grupo, não houve quaisquer problemas e ambos trabalhámos bem. O aluno Martim Antunes foi quem distribui as tarefas, explicou como secalhar ficava melhor e quem preocupava-se de sempre colocar um “dedinho” dele no final, mesmo não lhe tivesse sido atribuído aquela tarefa, aperfeiçoar ainda mais.

Por isso achamos que o aluno Martim Antunes mereça um 5 e o aluno Pedro Faneca um 4.