**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS   
DEPARTAMENTO DE COMPUTAÇÃO  
CURSO DE ENGENHARIA DE COMPUTAÇÃO**

****

Ítalo Fernandes Gonçalves

**AED 8 - Método de Runge Kutta de Quarta Ordem**

Dr. Clarimar José Coelho

GOIÂNIA,

2018

Ítalo Fernandes Gonçalves

**Método de Runge Kutta de Quarta Ordem**

Relatório apresentado como requisito parcial para obtenção de aprovação na disciplina CMP1058 - Fundamentos da computação IV, no curso de Engenharia da Computação, na Pontifícia Universidade Católica de Goiás.

Dr. Clarimar José Coelho

GOIÂNIA,

2018

**SUMÁRIO**

[**Resumo**](https://docs.google.com/document/d/1n2C7-9IBOV3S9HmVCrdMEH040__WmtRSZdwoe9GBnmM/edit#heading=h.30j0zll)

1. [**Introdução**](#_1fob9te) **4**
2. [**Objetivo**](#_3znysh7) **4**
3. [**Desenvolvimento**](#_2et92p0) **5**
4. [**Resultados**](#_tyjcwt) **6**
5. [**Conclusão**](#_3dy6vkm) **7**
6. [**Referências**](#_1t3h5sf) **7**

# **Introdução**

Frequentemente na áreas da ciências exatas e áreas afins, é necessário o cálculo de EDOs, em especifico, problemas de valores iniciais. Tais cálculos muitas das vezes precisam ser realista e não analitico (calculo simplificado) que ainda sim possuem solução matemática complexa.

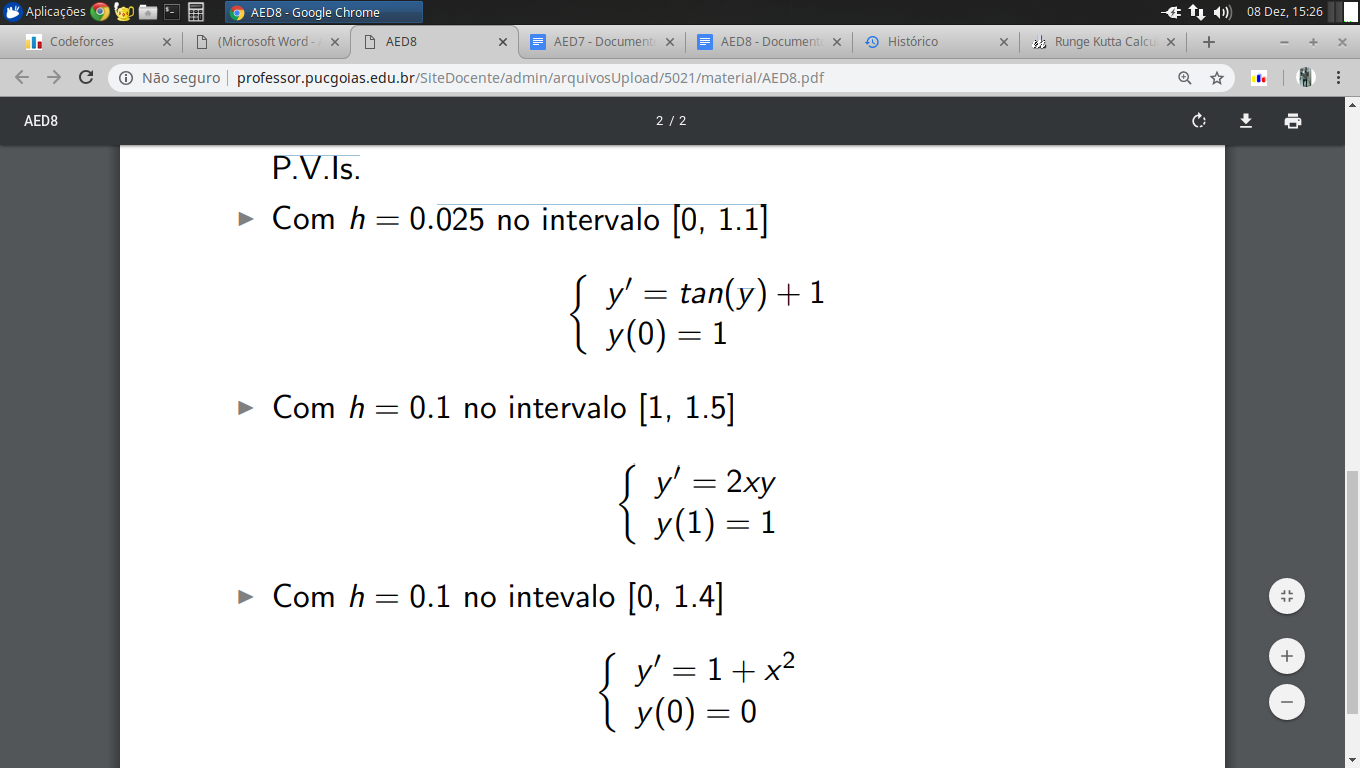
Métodos Iterativos são frequentemente usados sob essas circunstâncias. Este trabalho apresenta o método de Runge kutta de ordem 4, sendo este um método voltado para problemas de valores iniciais (PIV).

# **Objetivo**

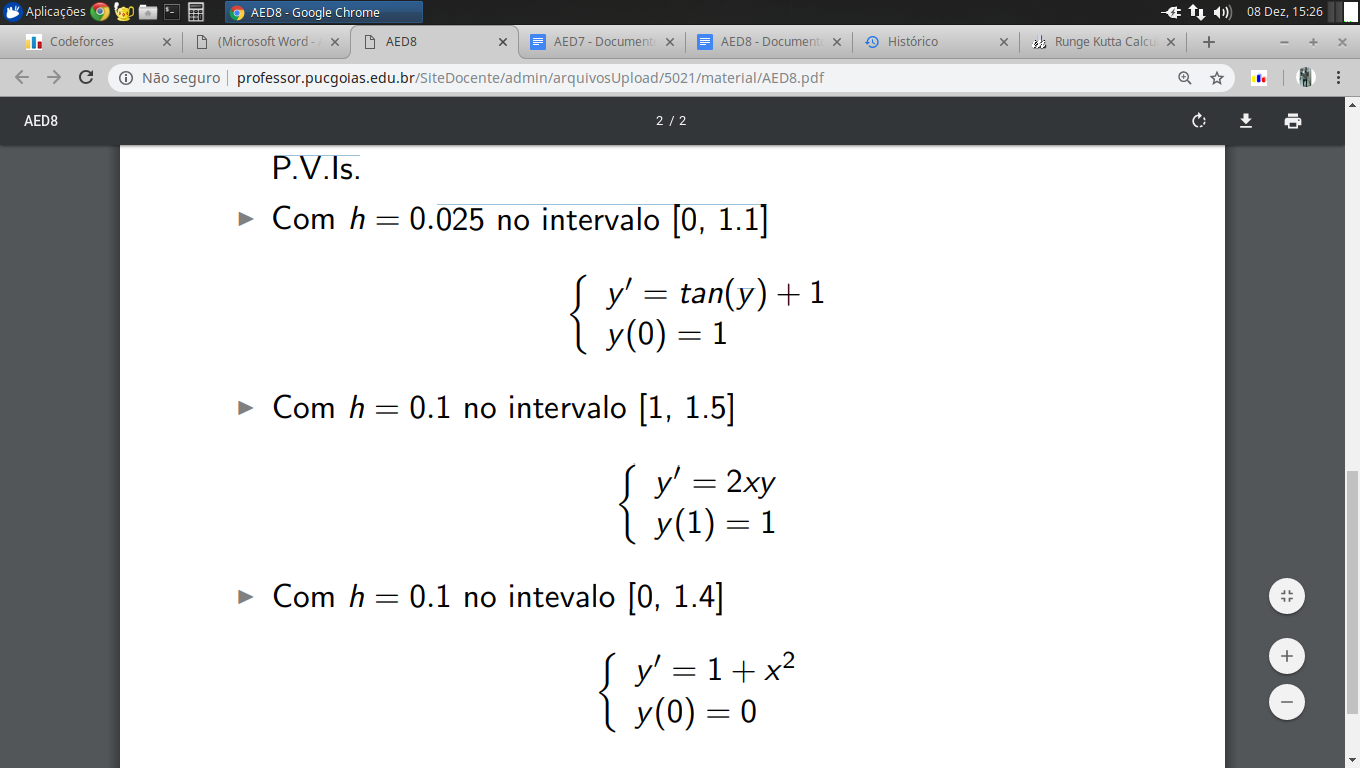
Implementar o método Runge kutta de quarta ordem.

Realizar os seguintes cálculos:

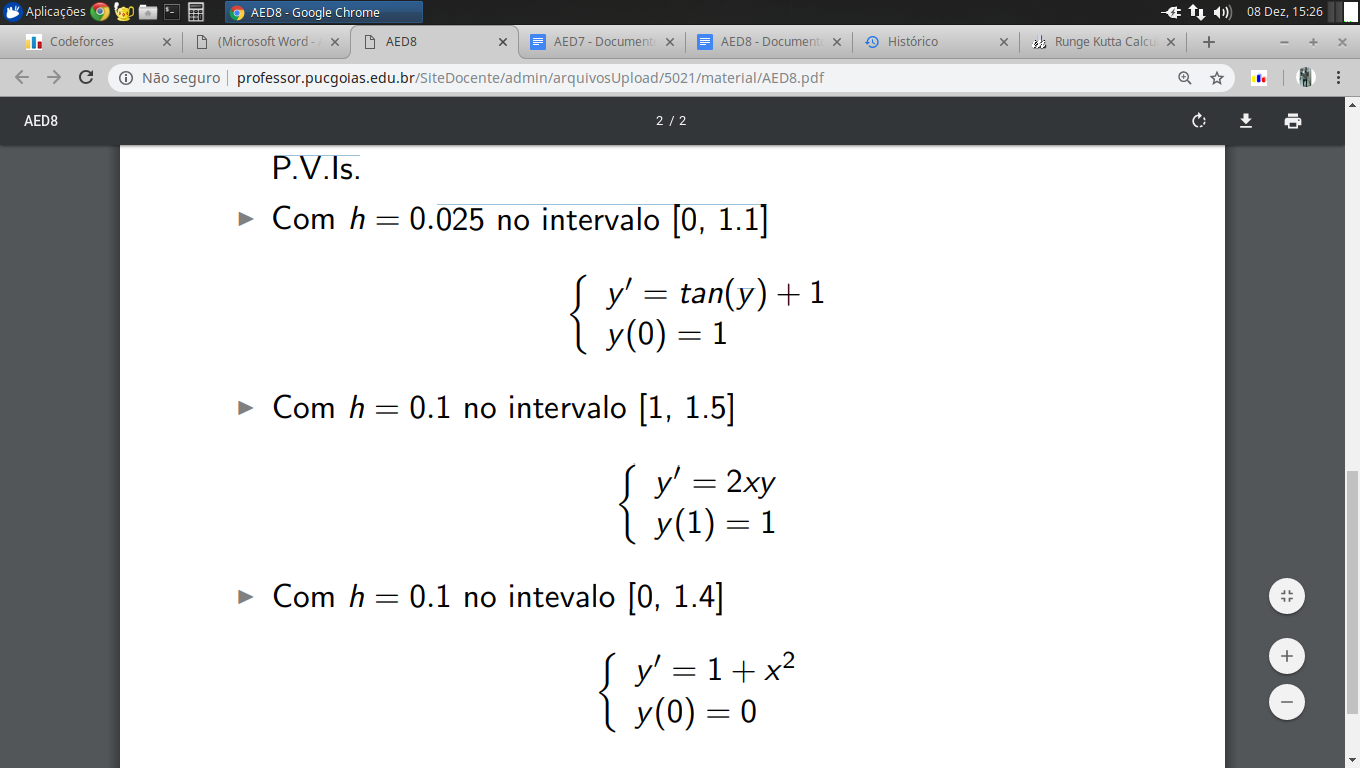
Fazendo h = 0.025 no intervalo de [0, 1.1]



Fazendo h = 0.1 no intervalo [1, 1.5]



Fazendo h = 0.1 no intervalo [0, 1.4]



# **Desenvolvimento**

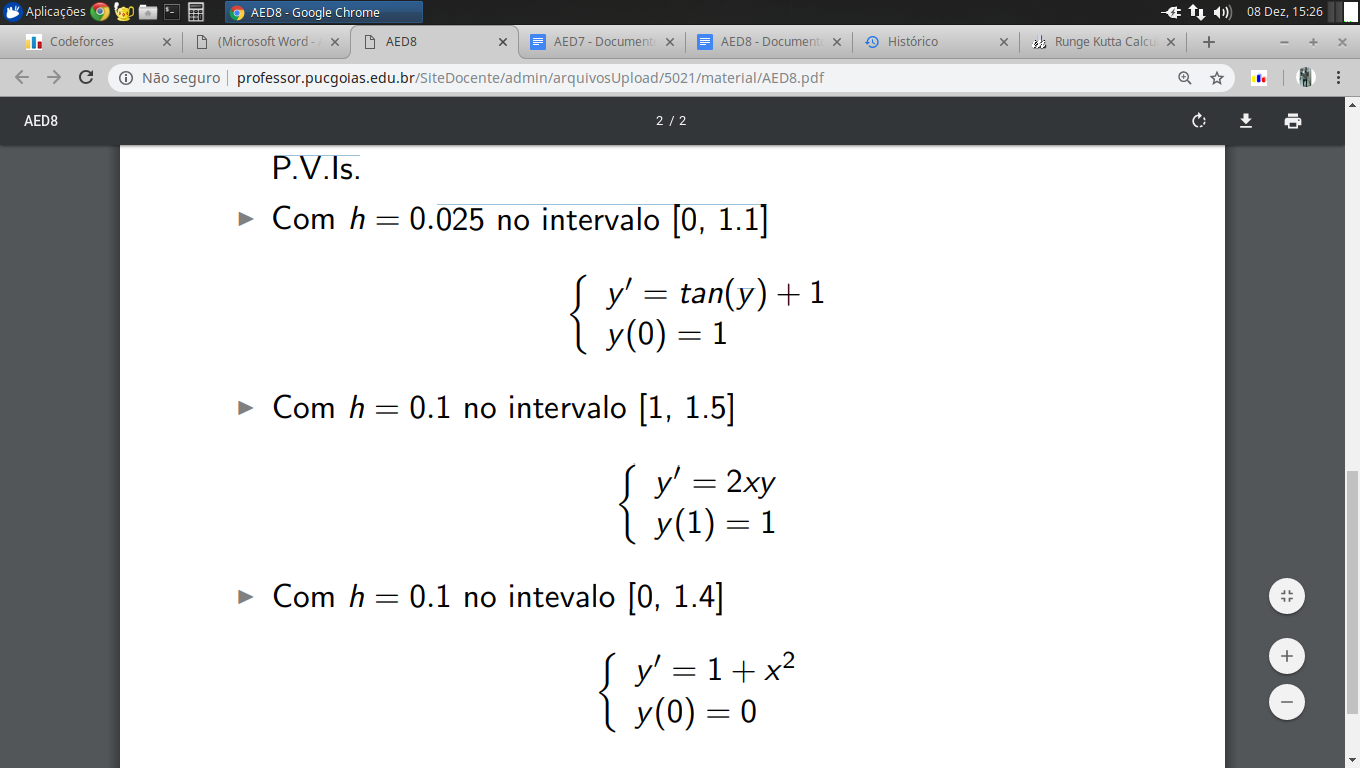
O código que segue foi desenvolvido no software Octave.

|  |
| --- |
| function [Yret] = RungeKutta4(l,r,h,valorInicial) H = (r-l)/h; Yanterior = valorInicial; Xanterior = l; Yatual = 0; for i = 1 : H  k1 = yderivada(Xanterior,Yanterior);  k2 = yderivada(Xanterior+h/2.,Yanterior+k1\*h/2.);  k3 = yderivada(Xanterior+h/2.,Yanterior+k2\*h/2.);  k4 = yderivada(Xanterior+h,Yanterior + h\*k3);  Yatual = Yanterior + h/6\*(k1 + 2\*k2 + 2\*k3 + k4);  Yanterior = Yatual;  Xanterior = Xanterior + h; endfor Yret = Yatual; endfunction  function [yderivada] = yderivada(x,y)  %yderivada = tan(y)+1;   %yderivada = 2\*x\*y ;  yderivada = 1+x\*x; endfunction |

# **Resultados**

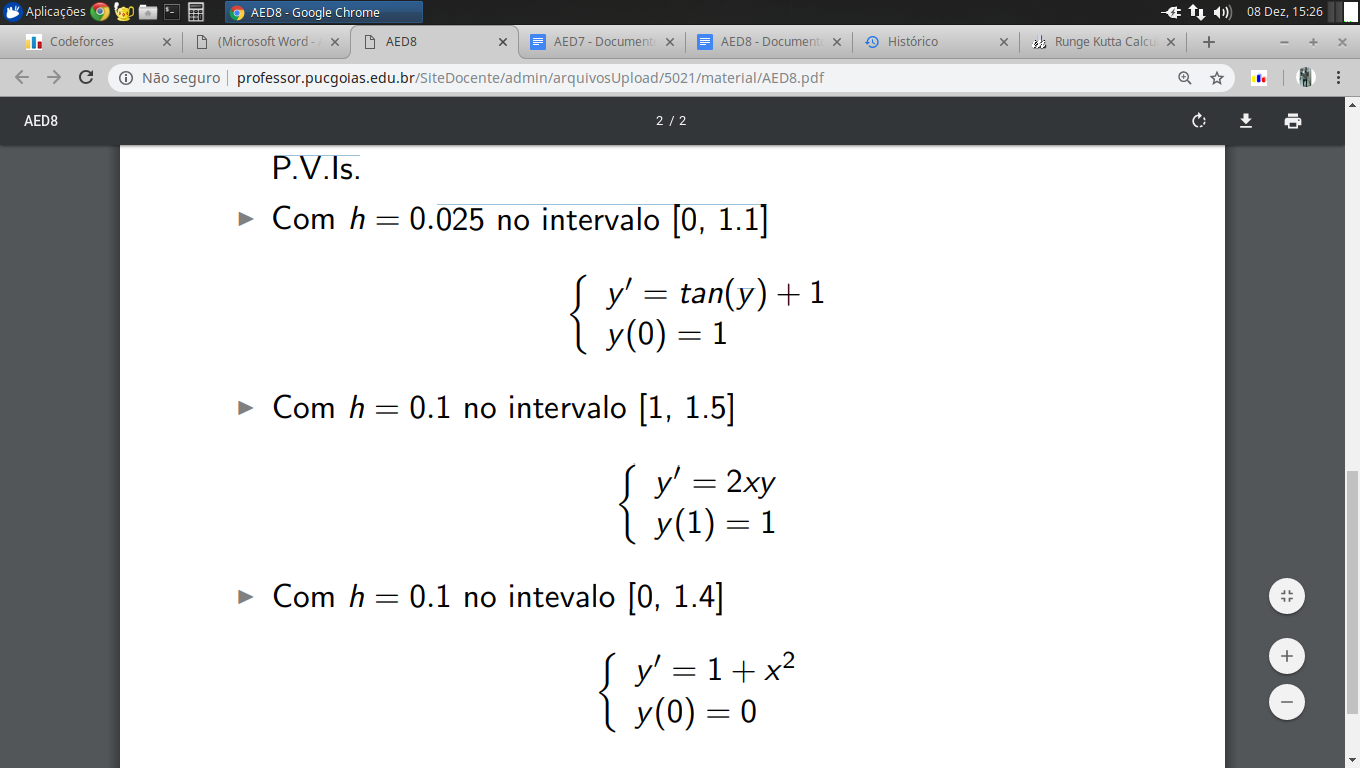
O Resultado obtido para a chamada de função para cada situação descrita inicialmente:

Fazendo h = 0.025 no intervalo de [0, 1.1]



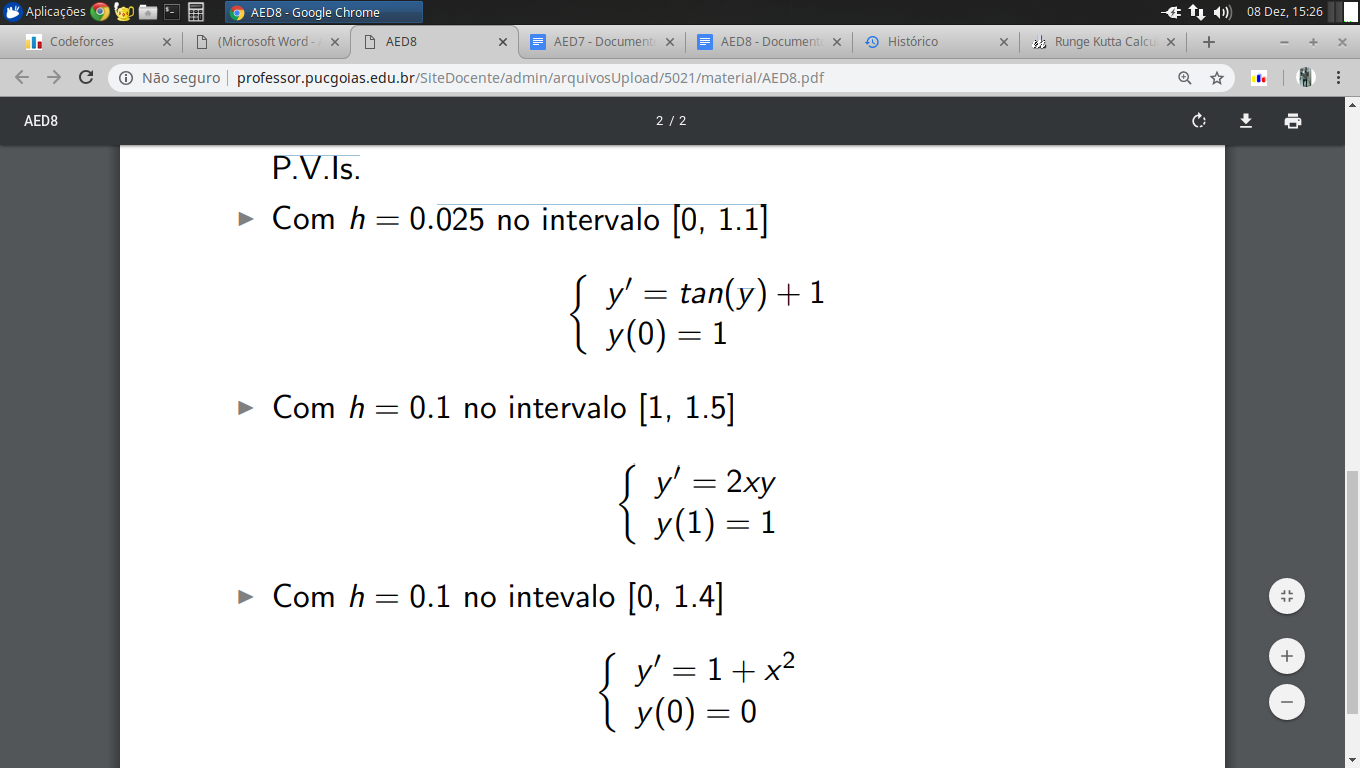
|  |
| --- |
| ans = 2.89054809354232 |

Fazendo h = 0.1 no intervalo [1, 1.5]



|  |
| --- |
| ans = 3.49021063637295 |

Fazendo h = 0.1 no intervalo [0, 1.4]



|  |
| --- |
| ans = 2.31466666666667 |

# **Conclusão**

Dessa forma, concluímos que EDOs com solução matemática complexas também podem ser resolvidas computacionalmente por métodos iterativos.

# **Referências**

<http://www.mathstools.com/section/main/runge_kutta_calculator#.XAv6fHVKhqO>

<http://professor.pucgoias.edu.br/SiteDocente/admin/arquivosUpload/5021/material/AED7%20(1).pdf>

<https://homepages.dcc.ufmg.br/~assuncao/an/Integracao01.pdf>

[https://www.wolframalpha.com/input/?i=integral+%5B1,6%5D+2\*sin(2\*sqrt(x))](https://www.wolframalpha.com/input/?i=integral+%5B1,6%5D+2*sin(2*sqrt(x)))

<https://www1.univap.br/spilling/CN/CN_Capt6.pdf>

<http://www.decom.ufop.br/marcone/Disciplinas/MetodosNumericoseEstatisticos/Integracao.pdf>

<http://www.facom.ufms.br/~montera/integracao_parte2.pdf>

<https://www.math.tecnico.ulisboa.pt/~calves/courses/integra/capiii33.html>