Projeto de Métodos Numéricos - Parte 1

Autor: Filipe Barbosa Lima - fbl@cin.ufpe.br
Monitor chefe: Victor Crisóstomo Mellia - vcm@cin.ufpe.br
Professor: Ricardo Martins de Abreu Silva - rmas@cin.ufpe.br
Site da disciplina: https://sites.google.com/a/cin.ufpe.br/if816ec/home
ver. 2.01.2019.1 - 26/02/2019
ver. 1.01.2018.2 - 19/08/2018
Centro de Informática - UFPE www.cin.ufpe.br

Objetivo

Implementar um programa que receba um arquivo de entrada, calcule os métodos especificados nele e gere um arquivo de saída com as respectivas respostas.

Funcionalidades

Seu programa deve calcular os seguintes métodos:

- Euler
- Euler Inverso
- Euler Aprimorado
- Runge-Kutta
- Adam-Bashforth
- Adam-Multon
- Fórmula Inversa

Euler

Recebe como entrada os valores y(0),t(0), h, quantidade de passos, a função. E calcula cada passo do método.

Bônus: gere o gráfico

Euler Inverso

Recebe como entrada os valores y(0),t(0), h, quantidade de passos, a função. E calcule cada passo do método.

Dica: use o método de previsão a partir do método de euler.

Bônus: resolva sem usar o método de previsão

Bônus: gere o gráfico

Euler Aprimorado (Euler Modificado)

Recebe como entrada os valores y(0),t(0), h, quantidade de passos, a função. E calcule cada passo do método.

Dica: use o método de previsão a partir do método de euler.
Bônus: resolva sem usar o método de previsão
Bônus: gere o gráfico

Runge-Kutta

Recebe como entrada os valores y(0),t(0), h, quantidade de passos, a função. E calcule cada passo do método. Utilize Runge-Kutta de quarta ordem.
Bônus: faça runge-kutta para ordens mais altas.
Bônus: gere o gráfico

Adam-Bashforth

- Adam-Bashforth por lista de valores iniciais

Recebe como entrada a lista de valores de y,t(0), h, quantidade de passos, a função, a ordem (de 2 a 8). E calcule cada passo do método.

- Adam-Bashforth obtendo os valores iniciais por métodos anteriores

Recebe como entrada o tipo de método anterior, t(0), h, quantidade de passos, a função, a ordem (de 2 a 8). E calcule cada passo do método.

Dica: consulte o material de apoio para saber os coeficientes de cada ordem
Dica: em alguns materiais o conceito de ordem vária, utilize o conceito dado em sala
Dica: a quantidade de y iniciais necessários é igual a ordem -1
Bônus: resolva sem usar o método de previsão
Bônus: calcule para ordem n
Bônus: gere o gráfico

Adam-Multon

- Adam-Multon por lista de valores iniciais

Recebe como entrada a lista de valores de y,t(0), h, quantidade de passos, a função, a ordem (de 2 a 8). E calcule cada passo do método.

- Adam-Multon obtendo os valores iniciais por métodos anteriores

Recebe como entrada o tipo de método anterior, t(0), h, quantidade de passos, a função, a ordem (de 2 a 8). E calcule cada passo do método.

Dica: consulte o material de apoio para saber os coeficientes de cada ordem
Dica: em alguns materiais o conceito de ordem vária, utilize o conceito dado em sala
Dica: a quantidade de y iniciais necessários é igual a ordem -1
Dica: use o método de previsão a partir do método de bashforth.

Bônus: utilize o método de previsão a partir de outros métodos além de bashforth
Bônus: resolva sem usar o método de previsão
Bônus: calcule para ordem n
Bônus: gere o gráfico

Fórmula Inversa

- Fórmula Inversa por lista de valores iniciais

Recebe como entrada a lista de valores de y,t(0), h, quantidade de passos, a função, a ordem (de 2 a 6). E calcule cada passo do método.

- Fórmula Inversa obtendo os valores iniciais por métodos anteriores

Recebe como entrada o tipo de método anterior, t(0), h, quantidade de passos, a função, a ordem (de 2 a 6). E calcule cada passo do método.

Dica: consulte o material de apoio para saber os coeficientes de cada ordem
Dica: em alguns materiais o conceito de ordem vária, utilize o conceito dado em sala
Dica: a quantidade de y iniciais necessários é igual a ordem -1
Dica: use o método de previsão a partir do método de bashforth.
Bônus: resolva sem usar o método de previsão
Bônus: calcule para ordem n
Bônus: gere o gráfico

Bônus

Implementem uma função que calcula para todos os métodos.
Implementem uma função que compara todos os métodos e com os valores exatos.

Linguagem

Preferencialmente deverá ser implementado em Python ou Julia. Confirme com o professor para outras linguagem.
Sugestão: utilize a biblioteca **sympy** para ler as funções e expressões matemáticas.
Sugestão: se for usar Python, utilizem python3
Sugestão: utilizem o ambiente linux

Sugestão de Instalação

- Pré-Requisitos:
  1. um terminal capaz de executar shell scripts
  2. python versão 3.5 ou superior instalado e atendendo por "python3"
  3. virtualenv instalado e capaz de criar ambientes para python3

- Para rodar o projeto:
  1. coloque no mesmo diretório os arquivos: requirements.txt e RUNME
  2. navegue pelo terminal até o diretório onde o projeto estiver.
  3. comando no terminal: "source RUNME"   (instala a biblioteca necessária para interpretar as strings funções)
  4. comando no terminal: "python metodos.py" // ou ”python nome_do_projeto.py”

Sugestão: caso necessário, edite o arquivo requirements.txt para adicionar uma biblioteca de plotar gráficos

- Links
  requirements.txt
  https://drive.google.com/file/d/1wbVrx8G-0yfpeJLV2Y3oxeCRsQHNZ--d/view?usp=sharing
  RUNME
  https://drive.google.com/file/d/1pgDz84WIA6JOAQFQYRqEPF-fN2LOuYhr/view?usp=sharing

Material de Apoio

Coefficients and error constants for Adams–Bashforth methods

k	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	C
1	1								$-\frac{1}{2}$
2	$\frac{3}{2}$	$-\frac{1}{2}$							$\frac{5}{12}$
3	$\frac{23}{12}$	$-\frac{4}{3}$	$\frac{5}{12}$						$-\frac{3}{8}$
4	$\frac{55}{24}$	$-\frac{59}{24}$	$\frac{37}{24}$	$-\frac{3}{8}$					$\frac{251}{720}$
5	$\frac{1901}{720}$	$-\frac{1387}{360}$	$\frac{109}{30}$	$-\frac{637}{360}$	$\frac{251}{720}$				$-\frac{95}{288}$
6	$\frac{4277}{1440}$	$-\frac{2641}{480}$	$\frac{4991}{720}$	$-\frac{3649}{720}$	$\frac{959}{480}$	$-\frac{95}{288}$			$\frac{19087}{60480}$
7	$\frac{198721}{60480}$	$-\frac{18637}{2520}$	$\frac{235183}{20160}$	$-\frac{10754}{945}$	$\frac{135713}{20160}$	$-\frac{5603}{2520}$	$\frac{19087}{60480}$		$-\frac{5257}{17280}$
8	$\frac{16083}{4480}$	$-\frac{1152169}{120960}$	$\frac{242653}{13440}$	$-\frac{296053}{13440}$	$\frac{2102243}{120960}$	$-\frac{115747}{13440}$	$\frac{32863}{13440}$	$-\frac{5257}{17280}$	$\frac{1070017}{3628800}$

Coefficients and error constants for Adams–Moulton methods

k	β_0	β_1	β_2	β_3	β_4	β_5	β_6	β_7	C
0	1								$\frac{1}{2}$
1	$\frac{1}{2}$	$\frac{1}{2}$							$-\frac{1}{12}$
2	$\frac{5}{12}$	$\frac{2}{3}$	$-\frac{1}{12}$						$\frac{1}{24}$
3	$\frac{3}{8}$	$\frac{19}{24}$	$-\frac{5}{24}$	$\frac{1}{24}$					$-\frac{19}{720}$
4	$\frac{251}{720}$	$\frac{323}{360}$	$-\frac{11}{30}$	$\frac{53}{360}$	$-\frac{19}{720}$				$\frac{3}{160}$
5	$\frac{95}{288}$	$\frac{1427}{1440}$	$-\frac{133}{240}$	$\frac{241}{720}$	$-\frac{173}{1440}$	$\frac{3}{160}$			$-\frac{863}{60480}$
6	$\frac{19087}{60480}$	$\frac{2713}{2520}$	$-\frac{15487}{20160}$	$\frac{586}{945}$	$-\frac{6737}{20160}$	$\frac{263}{2520}$	$-\frac{863}{60480}$		$\frac{275}{24192}$
7	$\frac{5257}{17280}$	$\frac{139849}{120960}$	$-\frac{4511}{4480}$	$\frac{123133}{120960}$	$-\frac{88547}{120960}$	$\frac{1537}{4480}$	$-\frac{11351}{120960}$	$\frac{275}{24192}$	$-\frac{33953}{3628800}$

Atenção: O conceito de ordem pode variar, utilizem o conceito de ordem dada pelo professor.

Order	Formula	LTE
1	$y_{n+1} = y_n + hf_{n+1}$	$-\frac{h^2}{2}y''(\eta)$
2	$y_{n+2} - \frac{4}{3}y_{n+1} + \frac{1}{3}y_n = \frac{2h}{3}f_{n+2}$	$-\frac{2h^3}{9}y'''(\eta)$
3	$y_{n+3} - \frac{18}{11}y_{n+2} + \frac{9}{11}y_{n+1} - \frac{2}{11}y_n = \frac{6h}{11}f_{n+3}$	$-\frac{3h^4}{22}y^{(4)}(\eta)$
4	$y_{n+4} - \frac{48}{25}y_{n+3} + \frac{36}{25}y_{n+2} - \frac{16}{25}y_{n+1} + \frac{3}{25}y_n = \frac{12h}{25}f_{n+4}$	$-\frac{12h^5}{125}y^{(5)}(\eta)$
5	$y_{n+5} - \frac{300}{137}y_{n+4} + \frac{300}{137}y_{n+3} - \frac{200}{137}y_{n+2} + \frac{75}{137}y_{n+1} - \frac{12}{137}y_n = \frac{60h}{137}f_{n+5}$	$-\frac{10h^6}{137}y^{(6)}(\eta)$
6	$y_{n+6} - \frac{360}{147}y_{n+5} + \frac{450}{147}y_{n+4} - \frac{400}{147}y_{n+3} + \frac{225}{147}y_{n+2} - \frac{72}{147}y_{n+1} + \frac{10}{147}y_n = \frac{60h}{147}f_{n+6}$	$-\frac{20h^7}{343}y^{(7)}(\eta)$

Atenção

- O uso da biblioteca **sympy** é restrita para computar as funções de entrada e para resolver os métodos implicitamente sem usar fator de correção (a parte bônus). Não utilize **sympy** para implementar os métodos.
- Para os métodos implícitos, implemente o método com o fator de correção (vocês estarão estudando para a prova), não será aceito apenas implementar implicitamente com o sympy.
- O formato do arquivo gerado poderá ter algumas diferenças (a correção não será feito um merge de arquivos, por exemplo em algoritmos). Mas siga o modelo e adapte caso realize funções bônus.
- Os valores calculados poderão ter pequenas diferenças, principalmente quando o método utilizar fatores de correção. Não necessariamente os valores serão

exatamente iguais ao exemplo. Mas verifique que seu código está consistente e que não tenha resultados discrepantes.

Formato de Arquivos

- Entrada
  Nome: “entrada.txt”
  Cada linha do arquivo terá um método a ser calculado e suas respectivas entradas. Seu programa deverá ler e executar todos os métodos citados no arquivo.

- Saída
  Nome: “saida.txt”
  Para cada método calculado, deverá ter:
  Nome do Método
  y(valor_do_t(0)) = valor_do_y(0)
  h = valor_do_h
  numero_do_passo_0 valor_y_0
  numero_do_passo_1 valor_y_1
  numero_do_passo_2 valor_y_2
  …
  Pule uma linha entre cada método

- Códigos
  - euler
  - euler_inverso
  - euler_aprimorado
  - runge_kutta
  - adam_bashforth
  - adam_bashforth_by_euler
  - adam_bashforth_by_euler_inverso
  - adam_bashforth_by_euler_aprimorado
  - adam_bashforth_by_runge_kutta
  - adam_multon
  - adam_multon_by_euler
  - adam_multon_by_euler_inverso
  - adam_multon_by_euler_aprimorado
  - adam_multon_by_runge_kutta
  - formula_inversa
  - formula_inversa_by_euler
  - formula_inversa_by_euler_inverso
  - formula_inversa_by_euler_aprimorado
  - formula_inversa_by_runge_kutta

- Exemplo de Arquivo de Entrada: “entrada.txt”
  ///
  euler 0 0 20 1-t+4*y
  euler_inverso 0 0 0.1 20 1-t+4*y

euler_aprimorado 0 0 0.1 20 1-t+4*y
runge_kutta 0 0 0.1 20 1-t+4*y
adam_bashforth 0.0 0.1 0.23 0.402 0.6328 0 0.1 20 1-t+4*y 5
adam_bashforth_by_euler 0 0 0.1 20 1-t+4*y 6
adam_bashforth_by_euler_inverso 0 0 0.1 20 1-t+4*y 6
adam_bashforth_by_euler_aprimorado 0 0 0.1 20 1-t+4*y 6
adam_bashforth_by_runge_kutta 0 0 0.1 20 1-t+4*y 6
adam_multon 0.0 0.1 0.23 0.402 0.6328 0 0.1 20 1-t+4*y 6
adam_multon_by_euler 0 0 0.1 20 1-t+4*y 6
adam_multon_by_euler_inverso 0 0 0.1 20 1-t+4*y 6
adam_multon_by_euler_aprimorado 0 0 0.1 20 1-t+4*y 6
adam_multon_by_runge_kutta 0 0 0.1 20 1-t+4*y 6
formula_inversa 0.0 0.1 0.23 0.402 0.6328 0 0.1 20 1-t+4*y 6
formula_inversa_by_euler 0 0 0.1 20 1-t+4*y 6
formula_inversa_by_euler_inverso 0 0 0.1 20 1-t+4*y 6
formula_inversa_by_euler_aprimorado 0 0 0.1 20 1-t+4*y 6
formula_inversa_by_runge_kutta 0 0 0.1 20 1-t+4*y 6
///

- Exemplo de Arquivo de Saída: “saida.txt”
  ///
  Metodo de Euler
  y(0.0) = 0.0
  h = 0.1
  0 0.0
  1 0.1
  2 0.23
  3 0.402
  4 0.6328
  5 0.9459200000000001
  6 1.3742880000000002
  7 1.9640032000000003
  8 2.7796044800000006
  9 3.911446272000001
  10 5.486024780800001
  11 7.680434693120002
  12 10.742608570368002
  13 15.019651998515204
  14 20.99751279792129
  15 29.356517917089803
  16 41.04912508392572
  17 57.40877511749601
  18 80.30228516449442
  19 112.34319923029219
  20 157.19047892240906

Metodo de Euler Inverso
y(0.0) = 0.0
h = 0.1
0 0.0
1 0.13
2 0.31880000000000003
3 0.5993280000000001
4 1.0229516800000003
5 1.6698046208000004
6 2.6648952084480007
7 4.203236525178881
8 6.589048979279054
9 10.296916407675326
10 16.06718959597351
11 25.054815769718676
12 39.061512600761134
13 60.89795965718737
14 94.9488170652123
15 148.0541546217312
16 230.88448120990066
17 360.085790687445
18 561.6258334724142
19 876.0143002169661
20 1366.4463083384671

Metodo de Euler Aprimorado
y(0.0) = 0.0
h = 0.1
0 0.0
1 0.11499999999999999
2 0.2732
3 0.495336
4 0.8120972800000001
5 1.2689039744000001
6 1.9329778821120003
7 2.9038072655257605
8 4.328634752978125
9 6.425379434407626
10 9.516561562923286
11 14.079511113126465
12 20.820676447427168
13 30.78560114219221
14 45.521689690444475
15 67.31910074185782
16 99.56726909794958
17 147.28255826496536

18 217.88918623214875
19 322.37499562358016
20 477.00199352289866

Metodo de Runge-Kutta
y(0.0) = 0.0
h = 0.1
0 0.0
1 0.1172
2 0.27973781333333336
3 0.5099075540764445
4 0.8409660953343014
5 1.322523823280022
6 2.028586204647585
7 3.0695496610129576
8 4.610096214321729
9 6.895887526110867
10 10.293385285617118
11 15.349252610064577
12 22.87896509352033
13 34.09899486217406
14 50.82399393573378
15 75.7609392203986
16 112.94791839970927
17 168.4086814741263
18 251.12905711100336
19 374.5135054610541
20 558.5579065464365

Metodo Adan-Bashforth por Euler
y(0.0) = 0.0
h = 0.1
0 0.0
1 0.1
2 0.23
3 0.402
4 0.6328
5 0.9459200000000001
6 1.4503504000000005
7 2.2072179280000013
8 3.318772899904446
9 4.931521006056477
10 7.350452015794431
11 10.973404262924845
12 16.326989348813566
13 24.28362943392663

14 36.18497327461336
15 53.92464439436779
16 80.3224224273389
17 119.70045024267056
18 178.468455832467
19 266.0758803692256
20 396.69620254792426

Metodo Adan-Bashforth por Euler Inverso
y(0.0) = 0.0
h = 0.1
0 0.0
1 0.13
2 0.31880000000000003
3 0.5993280000000001
4 1.0229516800000003
5 1.6698046208000004
6 2.566695703722668
7 3.805477937051404
8 5.7211929638180035
9 8.584308015571505
10 12.751192709513163
11 18.969331691784014
12 28.324500507174854
13 42.21624204626003
14 62.84263142677515
15 93.67742736390062
16 139.71366685993195
17 208.2579068626411
18 310.45963784606005
19 463.0085179382395
20 690.5078275840287

Metodo Adan-Bashforth por Euler Aprimorado
y(0.0) = 0.0
h = 0.1
0 0.0
1 0.11499999999999999
2 0.2732
3 0.495336
4 0.8120972800000001
5 1.2689039744000001
6 1.9456653358080005
7 2.920644161652339
8 4.39066188909708
9 6.556352627519578

10 9.755144563025334
11 14.53808125656135
12 21.67150537914415
13 32.26873608511426
14 48.06073367154866
15 71.63570450622579
16 106.77652638004642
17 159.1464865088791
18 237.26778123036024
19 353.80213780930626
20 527.5725058378443

Metodo Adan-Bashforth por Runge-Kutta (ordem = 6)
y(0.0) = 0.0
h = 0.1
0 0.0
1 0.1172
2 0.27973781333333336
3 0.5099075540764445
4 0.8409660953343014
5 1.322523823280022
6 2.0283613404798517
7 3.039020819458135
8 4.56861136284053
9 6.8269353219418925
10 10.155188501236871
11 15.130343809019017
12 22.560192668558326
13 33.59705591467788
14 50.03531435289132
15 74.58027063477684
16 111.17577967161537
17 165.7061352881605
18 247.04463916792673
19 368.3893689613675
20 549.3357916341132

Metodo Adan-Multon por Euler
y(0.0) = 0.0
h = 0.1
0 0.0
1 0.1
2 0.23
3 0.402
4 0.6328
5 0.9459200000000001

6 1.4750378000000004
7 2.2670810195598
8 3.438631555313039
9 5.200368573045594
10 7.848193604038299
11 11.8299848542377
12 17.829453387140337
13 26.874331806967994
14 40.51569568895171
15 61.096954239367825
16 92.15508561714024
17 139.0294901815327
18 209.78125959860066
19 316.5797915100637
20 477.7966798139911

Metodo Adan-Multon por Euler Inverso
y(0.0) = 0.0
h = 0.1
0 0.0
1 0.13
2 0.31880000000000003
3 0.5993280000000001
4 1.0229516800000003
5 1.6698046208000004
6 2.581116616843852
7 3.920908589896288
8 5.946061794985466
9 8.985528899947635
10 13.558122868563895
11 20.452239633883423
12 30.846597293284564
13 46.524832895653944
14 70.18166757321569
15 105.88270122112027
16 159.7661972888257
17 241.0994497045787
18 363.8724191038611
19 549.2052707474983
20 828.9824648890799

Metodo Adan-Multon por Euler Aprimorado
y(0.0) = 0.0
h = 0.1
0 0.0
1 0.11499999999999999

2 0.2732
3 0.495336
4 0.8120972800000001
5 1.2689039744000001
6 1.9675750763448892
7 3.0046440863266755
8 4.556202545156505
9 6.887384872499946
10 10.393364499671515
11 15.673179670181906
12 23.631540043279866
13 35.63314747054581
14 53.73868563021887
15 81.05925646166563
16 122.29132684247982
17 184.5250406475373
18 278.4641687013116
19 420.2677326883821
20 634.3303970317119

Metodo Adan-Multon por Runge-Kutta (ordem = 6)
y(0.0) = 0.0
h = 0.1
0 0.0
1 0.1172
2 0.27973781333333336
3 0.5099075540764445
4 0.8409660953343014
5 1.322523823280022
6 2.0495082169541137
7 3.127147337903094
8 4.741938006649038
9 7.1677665361907605
10 10.816320013366724
11 16.311862655734682
12 24.595769149278148
13 37.088734136117765
14 55.93615609611062
15 84.3767057549708
16 127.2995369680666
17 192.08574766213243
18 289.8782822894515
19 437.4991796432457
20 660.3440531769239

Metodo Formula Inversa de Diferenciacao por Euler

y(0.0) = 0.0
h = 0.1
0 0.0
1 0.1
2 0.23
3 0.402
4 0.6328
5 0.9459200000000001
6 1.4437388408163268
7 2.2479649747479296
8 3.4554338862398115
9 5.212067269227025
10 7.804896017282374
11 11.682418743024849
12 17.47059530486844
13 26.080385082783675
14 38.89795451071046
15 58.01391624461245
16 86.53192547437257
17 129.06212736136024
18 192.48904719320763
19 287.0996135619689
20 428.2388015587404

Metodo Formula Inversa de Diferenciacao por Euler Inverso
y(0.0) = 0.0
h = 0.1
0 0.0
1 0.13
2 0.31880000000000003
3 0.5993280000000001
4 1.0229516800000003
5 1.6698046208000004
6 2.620158533179211
7 3.9964780938182143
8 6.023736210040799
9 9.048861220877665
10 13.56022586470148
11 20.270274592114035
12 30.25881928973648
13 45.151823474303704
14 67.36558072307787
15 100.49171087605492
16 149.89248955501705
17 223.5782025436108
18 333.4982346326962

19 497.471299143187
20 742.0790563835598

Metodo Formula Inversa de Diferenciacao por Euler Aprimorado
y(0.0) = 0.0
h = 0.1
0 0.0
1 0.11499999999999999
2 0.2732
3 0.495336
4 0.8120972800000001
5 1.2689039744000001
6 1.9619185227441631
7 3.018057101661196
8 4.589349106466106
9 6.9061229704436125
10 10.34392024861073
11 15.470563989540665
12 23.11303694159662
13 34.495809303986896
14 51.45856948704324
15 76.75544326498611
16 114.48763374209419
17 170.76479762302017
18 254.70528748057745
19 379.91952486220595
20 566.7110540797345

Metodo Formula Inversa de Diferenciação por Runge-Kutta (ordem = 6)
y(0.0) = 0.0
h = 0.1
0 0.0
1 0.1172
2 0.27973781333333336
3 0.5099075540764445
4 0.8409660953343014
5 1.322523823280022
6 2.0490810835541944
7 3.1476001796258033
8 4.7796200954679415
9 7.190374814386867
10 10.770315722217395
11 16.1068102083762
12 24.06046846463799
13 35.908741225499604
14 53.567634770370596

15 79.90247273326928
16 119.18179032226195
17 177.76715294849487
18 265.15215214544355
19 395.5052094276906
20 589.9623529200103
///