EAFIT UNIVERSITY
DEPARTMENT OF INFORMATICS AND SYSTEMS
PROJECT CHOICE

Third Report

May 17, 2022

Course

Numerical analysis

Teacher

Edwar Samir Posada Murillo

Semester

2022-1

Project's name

Numerical Algorithms

Repository

This project has a GitHub repository where the evidence related with it will be. https://github.com/DanielHernandezO/NumericalMethodsProject

Members

1. Jose Miguel Blanco Velez
2. Neller Pellegrino Baquero
3. Samuel David Villegas Bedoya
4. Daniel Andres Hernandez Oyola

Project's description

Webpage used to calculate data using different types of numerical methods with the option of visualising them in a 2d graph.

Added values

1. The project will be done in english
2. The project will have its documentation in latex
3. The numerical algorithms can be found in multiple programming languages
4. The project will have extra numerical methods

1 Crout - Matlab

Input data

$$A = \begin{pmatrix} 4 & -1 & 0 & 3 \\ 1 & 15.5 & 3 & 8 \\ 0 & -1.3 & -4 & -1.1 \\ 14 & 5 & -2 & 30 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Result

```
x =

    0.4853
    0.2078
   -0.2503
   -0.2445


L =

    4.0000         0         0         0
    1.0000   15.7500         0         0
         0   -1.3000   -3.7524         0
   14.0000    8.5000   -3.6190   16.0711


U =

    1.0000   -0.2500         0    0.7500
         0    1.0000    0.1905    0.4603
         0         0    1.0000    0.1337
         0         0         0    1.0000
```

2 Crout - JavaScript

Input data

$$A = \begin{pmatrix} 4 & -1 & 0 & 3 \\ 1 & 15.5 & 3 & 8 \\ 0 & -1.3 & -4 & -1.1 \\ 14 & 5 & -2 & 30 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Result

```
 L:
 4.0000         0         0         0
 1.0000   15.7500         0         0
 0.0000   -1.3000   -3.7523         0
14.0000    8.5000   -3.6193   16.0714

 U:
1   -0.2500   0.0000   0.7500
0         1   0.1905   0.4603
0         0        1   0.1337
0         0        0        1

 X:
 0.4853
 0.2078
-0.2503
-0.2445
```

3 Doolittle - Matlab

Input data

$$A = \begin{pmatrix} 4 & -1 & 0 & 3 \\ 1 & 15.5 & 3 & 8 \\ 0 & -1.3 & -4 & -1.1 \\ 14 & 5 & -2 & 30 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Result

```
x =

    0.4853
    0.2078
   -0.2503
   -0.2445


L =

    1.0000         0         0         0
    0.2500    1.0000         0         0
         0   -0.0825    1.0000         0
    3.5000    0.5397    0.9645    1.0000


U =

    4.0000   -1.0000         0    3.0000
         0   15.7500    3.0000    7.2500
         0         0   -3.7524   -0.5016
         0         0         0   16.0711
```

4 Doolittle - Javascript

Input data

$$A = \begin{pmatrix} 4 & -1 & 0 & 3 \\ 1 & 15.5 & 3 & 8 \\ 0 & -1.3 & -4 & -1.1 \\ 14 & 5 & -2 & 30 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Result

```
 L:
    1         0         0   0
0.2500         1         0   0
0.0000   -0.0825         1   0
3.5000    0.5397    0.9645   1

 U:
4.0000   -1.0000    0.0000    3.0000
    0   15.7500    3.0000    7.2500
    0         0   -3.7525   -0.5019
    0         0         0   16.0713

 X:
 0.4853
 0.2078
-0.2503
-0.2445
```

5 Cholesky - Matlab

Input data

$$A = \begin{pmatrix} 4 & -1 & 0 & 3 \\ 1 & 15.5 & 3 & 8 \\ 0 & -1.3 & -4 & -1.1 \\ 14 & 5 & -2 & 30 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Result

```
x =

    0.4853
    0.2078
   -0.2503
   -0.2445


L =

   2.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i
   0.5000 + 0.0000i   3.9686 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i
   0.0000 + 0.0000i  -0.3276 + 0.0000i   0.0000 + 1.9371i   0.0000 + 0.0000i
   7.0000 + 0.0000i   2.1418 + 0.0000i   0.0000 + 1.8683i   4.0089 + 0.0000i


U =

   2.0000 + 0.0000i  -0.5000 + 0.0000i   0.0000 + 0.0000i   1.5000 + 0.0000i
   0.0000 + 0.0000i   3.9686 + 0.0000i   0.7559 + 0.0000i   1.8268 + 0.0000i
   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 1.9371i   0.0000 + 0.2589i
   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i   4.0089 + 0.0000i
```

6 Cholesky - Javascript

Input data

$$A = \begin{pmatrix} 4 & -1 & 0 & 3 \\ 1 & 15.5 & 3 & 8 \\ 0 & -1.3 & -4 & -1.1 \\ 14 & 5 & -2 & 30 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Result

```
 L:
  2                     0     0     0
0.5     3.905124837953327     0     0
  0   -0.33289588782552954   NaN     0
  7     0.3841106397986879   NaN   NaN

 U:
2                   0.5                             0                     7
0   3.905124837953327   -0.33289588782552954   0.3841106397986879
0                   0                   NaN                   NaN
0                   0                     0                   NaN

 X:
NaN
NaN
NaN
NaN
```

7 Heun - Matlab

Input data

$$f(t, y) = (y - t^2 + 1)$$

left end point
right end point=2
number of subintervals= 10
alpah = 1 **Result**

```
0.0000   1.00000000
0.2000   1.43991111
0.4000   1.95980255
0.6000   2.55966996
0.8000   3.23950802
1.0000   3.99931024
1.2000   4.83906868
1.4000   5.75877366
1.6000   6.75841334
1.8000   7.83797327
2.0000   8.99743580
```

8 Linear - Matlab

Input data

$$\text{Tabla} = \begin{array}{|c|c|c|c|c|} \hline \text{x} & -1 & 0 & 3 & 4 \\ \hline \text{y} & 15.5 & 3 & 8 & 1 \\ \hline \end{array}$$

Result

```
ans =

  -12.5000    3.0000
    1.6667    3.0000
   -7.0000   29.0000
```

9 Linear - Javascript

Input data

$$\text{Tabla} = \begin{array}{|c|c|c|c|c|} \hline \text{x} & -1 & 0 & 3 & 4 \\ \hline \text{y} & 15.5 & 3 & 8 & 1 \\ \hline \end{array}$$

Result

```
C:\Users\Jose\Desktop\analisis\NumericalMethodsProject>node index.js
[ [ [ -12.5 ], [ 3 ] ], [ [ 1.6666666666666665 ], [ 3 ] ] ]
```

10 Cuadratica - Matlab

Input data

Tabla =

x	-1	0	3	4
y	15.5	3	8	1

Result

```
ans =

         0  -12.5000    3.0000
    4.7222  -12.5000    3.0000
  -22.8333  152.8333 -245.0000
```

11 Cuadratica - Javascript

Input data

Tabla =

x	-1	0	3	4
y	15.5	3	8	1

Result

```
[
  [ [ 0 ], [ -12.500000000000001 ], [ 3 ] ],
  [ [ 4.722222222222221 ], [ -12.500000000000001 ], [ 3 ] ],
  [
    [ -22.833333333333336 ],
    [ 152.83333333333337 ],
    [ -245.0000000000001 ]
  ]
]
```

12 Cubica - Matlab

Input data

Tabla =

x	-1	0	3	4
y	15.5	3	8	1

Result

```
ans =

    2.5333    7.6000   -7.4333    3.0000
   -1.5222    7.6000   -7.4333    3.0000
    2.0333  -24.4000   88.5667  -93.0000
```

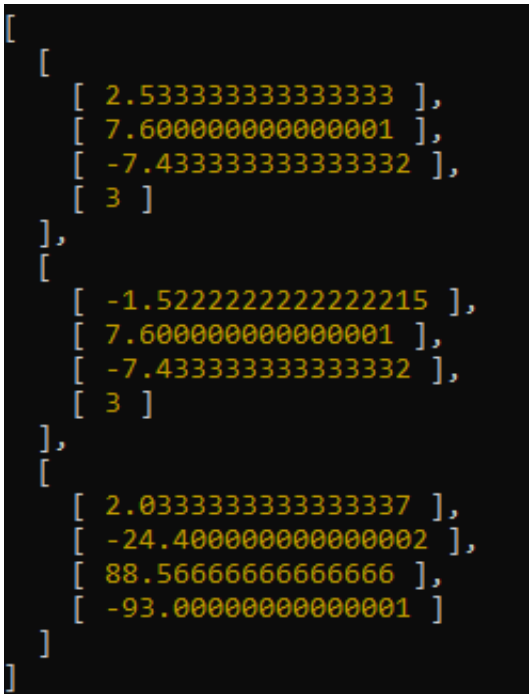
13 Cubica - Javascript

Input data

Tabla =

x	-1	0	3	4
y	15.5	3	8	1

Result



14 Newton - Matlab

Input data

Tabla =

x	-1	0	3	4
y	15.5	3	8	1

Result

```
ans =

   15.5000  -12.5000    3.5417   -1.1417
```

15 Newton - Javascript

Input data

$$\text{Tabla} = \begin{array}{|c|c|c|c|c|} \hline x & -1 & 0 & 3 & 4 \\ \hline y & 15.5 & 3 & 8 & 1 \\ \hline \end{array}$$

Result

```
C:\Users\Jose\Desktop\analisis\NumericalMethodsProject>node index.js
[ 15.5, -12.5, 3.5416666666666665, -1.1416666666666666 ]
```

16 Lagrange - Javascript

Input data

$$\text{Tabla} = \begin{array}{|c|c|c|c|c|} \hline x & -1 & 0 & 3 & 4 \\ \hline y & 15.5 & 3 & 8 & 1 \\ \hline \end{array}$$

Result

```
C:\Users\Jose\Desktop\analisis\NumericalMethodsProject>node index.js
[ -1.1416666666666666, 5.825, -5.533333333333335, 3 ]
```

17 Lagrange - Matlab

Input data

$$\text{Tabla} = \begin{array}{|c|c|c|c|c|} \hline x & -1 & 0 & 3 & 4 \\ \hline y & 15.5 & 3 & 8 & 1 \\ \hline \end{array}$$

Result

```
   -1.1417    5.8250   -5.5333    3.0000
```

18 Simple LU - JavaScript

Input data

$$A = \begin{pmatrix} 4 & -1 & 0 & 3 \\ 1 & 15.5 & 3 & 8 \\ 0 & -1.3 & -4 & -1.1 \\ 14 & 5 & -2 & 30 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Result

```
L =

   1                     0                     0   0
0.25                     1                     0   0
   0   -0.08253968253968254                     1   0
 3.5     0.5396825396825397   0.9644670050761421   1
U =

4      -1                     0                     3
0   15.75                     3                  7.25
0       0   -3.7523809523809524   -0.5015873015873017
0       0                     0    16.071065989847714
Z =

                1
             0.75
  1.061904761904762
-3.9289340101522843
x =  [
  0.48531269740998106,
  0.2078332280480101,
  -0.2503158559696778,
  -0.24447252053063806
]
```

19 Simple LU - Matlab

Input data

$$A = \begin{pmatrix} 4 & -1 & 0 & 3 \\ 1 & 15.5 & 3 & 8 \\ 0 & -1.3 & -4 & -1.1 \\ 14 & 5 & -2 & 30 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Result

```
L =

   1.000000000000000                   0                   0                   0
   0.250000000000000   1.000000000000000                   0                   0
                   0  -0.082539682539683   1.000000000000000                   0
   3.500000000000000   0.539682539682540   0.964467005076142   1.000000000000000


U =

   4.000000000000000  -1.000000000000000                   0   3.000000000000000
                   0  15.750000000000000   3.000000000000000   7.250000000000000
                   0                   0  -3.752380952380952  -0.501587301587302
                   0                   0                   0  16.071065989847714


z =

   1.000000000000000
   0.750000000000000
   1.061904761904762
  -3.928934010152284


x =

   0.485312697409981   0.207833228048010  -0.250315855969678  -0.244472520530638
```

20 Pivot LU - JavaScript

Input data

$$A = \begin{pmatrix} 4 & -1 & 0 & 3 \\ 1 & 15.5 & 3 & 8 \\ 0 & -1.3 & -4 & -1.1 \\ 14 & 5 & -2 & 30 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Result

```
L =

                  1                     0                     0   0
0.07142857142857142                     1                     0   0
                  0   -0.0858490566037736                     1   0
 0.2857142857142857   -0.16037735849056603   -0.28831562974203334   1

U =

14                   5                  -2                  30
 0   15.142857142857142   3.142857142857143   5.857142857142858
 0                   0   -3.730188679245283   -0.5971698113207548
 0                   0                   0   -4.804248861911987

P =

0   0   0   1
0   1   0   0
0   0   1   0
1   0   0   0

Bn =

1
1
1
1

z =

                  1
0.9285714285714286
1.0797169811320755
1.1745068285280729

x =  [
  0.4853126974099811,
  0.20783322804801013,
  -0.2503158559696778,
  -0.24447252053063806
]
```

21 Pivot LU - Matlab

Input data

$$A = \begin{pmatrix} 4 & -1 & 0 & 3 \\ 1 & 15.5 & 3 & 8 \\ 0 & -1.3 & -4 & -1.1 \\ 14 & 5 & -2 & 30 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Result

```
L =

   1.000000000000000                   0                   0                   0
   0.071428571428571   1.000000000000000                   0                   0
                   0  -0.085849056603774   1.000000000000000                   0
   0.285714285714286  -0.160377358490566  -0.288315629742033   1.000000000000000


U =

  14.000000000000000   5.000000000000000  -2.000000000000000  30.000000000000000
                   0  15.142857142857142   3.142857142857143   5.857142857142858
                   0                   0  -3.730188679245283  -0.597169811320755
                   0                   0                   0  -4.804248861911987


P =

     0     0     0     1
     0     1     0     0
     0     0     1     0
     1     0     0     0


Bn =

     1
     1
     1
     1


z =

   1.000000000000000
   0.928571428571429
   1.079716981132075
   1.174506828528073


x =

   0.485312697409981   0.207833228048010  -0.250315855969678  -0.244472520530638
```

22 Trapezoid compound - JavaScript

Input data

$$a = 1$$

$$b = 2$$

$$n = 10$$

$$f(x) = e^x - 2x$$

Result

(index)	xi	fxi	A
0	1	0.7182818284590451	0.7182818284590451
1	1.1	1.6083320478928655	2.3266138763519106
2	1.2	1.8402338454730947	4.166847721825006
3	1.3	2.138593335238488	6.305441057063494
4	1.4	2.5103999336893477	8.815840990752841
5	1.5	2.963378140676129	11.77921913142897
6	1.6	3.5060648487902295	15.2852839802192
7	1.7000000000000002	4.147894783454401	19.433178763673602
8	1.8	4.899294928825891	24.332473692499494
9	1.9	5.7717888845585374	30.10426257705803
10	2	3.3890560989306495	33.49331867598868

A = 1.6746659337994343

23 Trapezoid compound - Matlab

Input data

$$a = 1$$

$$b = 2$$

$$n = 10$$

$$f(x) = e^x - 2x$$

Result

```
    Counter     Xi          Fxi      A
    1.0000    1.0000    0.7183    0.7183
    2.0000    1.1000    1.6083    2.3266
    3.0000    1.2000    1.8402    4.1668
    4.0000    1.3000    2.1386    6.3054
    5.0000    1.4000    2.5104    8.8158
    6.0000    1.5000    2.9634   11.7792
    7.0000    1.6000    3.5061   15.2853
    8.0000    1.7000    4.1479   19.4332
    9.0000    1.8000    4.8993   24.3325
   10.0000    1.9000    5.7718   30.1043
   11.0000    2.0000    3.3891   33.4933


A =

    1.6747
```

24 Simpson - JavaScript

Input data

$$a = 1$$

$$b = 2$$

$$n = 10$$

$$f(x) = e^x - 2x$$

Result

(index)	xi	fxi	A
0	1	0.7182818284590451	0.7182818284590451
1	1.1	3.216664095785731	3.934945924244776
2	1.2	1.8402338454730947	5.775179769717871
3	1.3	4.277186670476976	10.052366440194847
4	1.4	2.5103999336893477	12.562766373884195
5	1.5	5.926756281352258	18.489522655236453
6	1.6	3.5060648487902295	21.99558750402668
7	1.7000000000000002	8.295789566908802	30.291377070935482
8	1.8	4.899294928825891	35.19067199976137
9	1.9	11.543577769117075	46.73424976887844
10	2	3.3890560989306495	50.123305867809094

A = 1.6707768622603032

25 Simpson - Matlab

Input data

$$a = 1$$

$$b = 2$$

$$n = 10$$

$$f(x) = e^x - 2x$$

Result

```
    Counter     Xi          Fxi      A
    1.0000    1.0000    0.7183    0.7183
    2.0000    1.1000    3.2167    3.9349
    3.0000    1.2000    1.8402    5.7752
    4.0000    1.3000    4.2772   10.0524
    5.0000    1.4000    2.5104   12.5628
    6.0000    1.5000    5.9268   18.4895
    7.0000    1.6000    3.5061   21.9956
    8.0000    1.7000    8.2958   30.2914
    9.0000    1.8000    4.8993   35.1907
   10.0000    1.9000   11.5436   46.7342
   11.0000    2.0000    3.3891   50.1233


A =

    1.6708
```

26 Simpson Simple - JavaScript

Input data

$$a = 1$$

$$b = 2$$

$$n = 9$$

$$f(x) = e^x - 2x$$

Result

(index)	xi	fxi	A
0	1	0.7182818284590451	0.7182818284590451
1	1.1111111111111112	2.44652866588578	3.164810494344825
2	1.2222222222222223	2.8508362279633763	6.015646722308201
3	1.3333333333333333	2.254002456033021	8.269649178341222
4	1.4444444444444444	4.051821971680084	12.321471150021306
5	1.5555555555555556	4.879820245595901	17.201291395617208
6	1.6666666666666665	3.9223134342733896	21.123604829890596
7	1.7777777777777777	7.0834141053263195	28.207018935216915
8	1.8888888888888888	8.50272040107713	36.70973933629405
9	2	3.3890560989306495	40.0987954352247

A = 1.6707831431343625

27 Simpson Simple - Matlab

Input data

$$a = 1$$

$$b = 2$$

$$n = 9$$

$$f(x) = e^x - 2x$$

Result

```
    Counter     Xi          Fxi      A
    1.0000    1.0000    0.7183    0.7183
    2.0000    1.0833    2.3635    3.0818
    3.0000    1.1667    2.6338    5.7156
    4.0000    1.2500    1.9807    7.6963
    5.0000    1.3333    3.3810   11.0773
    6.0000    1.4167    3.8701   14.9474
    7.0000    1.5000    2.9634   17.9108
    8.0000    1.5833    5.1135   23.0243
    9.0000    1.6667    5.8835   28.9077
   10.0000    1.7500    4.5092   33.4169
   11.0000    1.8333    7.7641   41.1810
   12.0000    1.9167    8.8948   50.0758
   13.0000    2.0000    3.3891   53.4649


A =

    1.6708
```

28 jacobi - Matlab

Input data

$$A = \begin{pmatrix} 4 & -1 & 0 & 3 \\ 1 & 15.5 & 3 & 8 \\ 0 & -1.3 & -4 & -1.1 \\ 14 & 5 & -2 & 30 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

$$x0 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Result

20.000000000000000	0.523560004785802	0.254518224580359	-0.409622194215019	0.000769598898146
21.000000000000000	0.523943421238336	0.254751901431433	-0.409833506856140	0.000581980955112
22.000000000000000	0.524229864816185	0.254924984100144	-0.409993060766602	0.000437345449321
23.000000000000000	0.524447106480190	0.255057107247651	-0.410113101999437	0.000330218043530
24.000000000000000	0.524610005548426	0.255155697163931	-0.410203660392208	0.000248442296050
25.000000000000000	0.524733205064977	0.255230535734461	-0.410271837862075	0.000187422179934
26.000000000000000	0.524825781040666	0.255286615763212	-0.410323244719618	0.000141102408853
27.000000000000000	0.524895684573456	0.255329050313967	-0.410361961355017	0.000106393026870
28.000000000000000	0.524948275149107	0.255360924746978	-0.410391145961265	0.000080129148558
29.000000000000000	0.524987950144451	0.255384999889262	-0.410413130827260	0.000060401258846
30.000000000000000	0.525017819165938	0.255403108413074	-0.410429700335005	0.000045500530962
31.000000000000000	0.525040341181347	0.255416771957328	-0.410442183729544	0.000034292711698
32.000000000000000	0.525057303265795	0.255427057179971	-0.410451591320835	0.000025835995875
33.000000000000000	0.525070089389608	0.255434813228340	-0.410458679451516	0.000019470213684
34.000000000000000	0.525079721163651	0.255440654157096	-0.410464020846618	0.000014669798856
35.000000000000000	0.525086980451755	0.255445057318882	-0.410468045468966	0.000011054701396
36.000000000000000	0.525092449548966	0.255448374083469	-0.410471078209027	0.000008329473338
37.000000000000000	0.525096571117290	0.255450873942363	-0.410473363362482	0.000006276644366
38.000000000000000	0.525099676498614	0.255452757275854	-0.410475085302710	0.000004729418638
39.000000000000000	0.525102016620934	0.255454176599820	-0.410476382792042	0.000003563776166
40.000000000000000	0.525103779849087	0.255455245970705	-0.410477360484623	0.000002685320789
41.000000000000000	0.525105108524583	0.255456051824416	-0.410478097185512	0.000002023460236
42.000000000000000	0.525106109673854	0.255456659012126	-0.410478652306110	0.000001524697071
43.000000000000000	0.525106864073963	0.255457116559210	-0.410479070596616	0.000001148893313
44.000000000000000	0.525107432517472	0.255457461316841	-0.410479385786889	0.000000865705897
45.000000000000000	0.525107860854829	0.255457721104288	-0.410479623287367	0.000000652326719
46.000000000000000	0.525108183611137	0.255457916854707	-0.410479802248417	0.000000491537725
47.000000000000000	0.525108426815271	0.255458064358137	-0.410479937098365	0.000000370382862
48.000000000000000	0.525108610072691	0.255458175503237	-0.410480038710219	0.000000279089274
49.000000000000000	0.525108748160840	0.255458259253653	-0.410480115276396	0.000000210298774
50.000000000000000	0.525108852212271	0.255458322360613	-0.410480172970352	0.000000158463532
51.000000000000000	0.525108930616973	0.255458369913002	-0.410480216443700	0.000000119405045
52.000000000000000	0.525108989696140	0.255458405744405	-0.410480249201618	0.000000089973674

29 gauss-seidel - Matlab

Input data

$$A = \begin{pmatrix} 4 & -1 & 0 & 3 \\ 1 & 15.5 & 3 & 8 \\ 0 & -1.3 & -4 & -1.1 \\ 14 & 5 & -2 & 30 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

$$x0 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Result

14.000000000000000	0.523771039468388	0.254652177903460	-0.409850937440877	0.001245991200282
15.000000000000000	0.524306977399777	0.254975125985672	-0.410103024684111	0.000746956331826
16.000000000000000	0.524628265568755	0.255168729371301	-0.410254147872196	0.000447791093152
17.000000000000000	0.524820873856912	0.255284792209243	-0.410344744356502	0.000268445228405
18.000000000000000	0.524936340146210	0.255354370446767	-0.410399055829629	0.000160929598099
19.000000000000000	0.525005560760193	0.255396081740605	-0.410431614886814	0.000096475305962
20.000000000000000	0.525047057663135	0.255421087145966	-0.410451133639366	0.000057835754084
21.000000000000000	0.525071934543803	0.255436077577567	-0.410462834890390	0.000034671820080
22.000000000000000	0.525086847926438	0.255445064156122	-0.410469849645823	0.000020785327808
23.000000000000000	0.525095788315090	0.255450451498947	-0.410474054905046	0.000012460547242
24.000000000000000	0.525101147967639	0.255453681144417	-0.410476575906000	0.000007469944136
25.000000000000000	0.525104361013130	0.255455617276977	-0.410478087214927	0.000004478139227
26.000000000000000	0.525106287194208	0.255456777964528	-0.410478993225958	0.000002684589144
27.000000000000000	0.525107441915969	0.255457473782376	-0.410479536368379	0.000001609378027
28.000000000000000	0.525108134157399	0.255457890916580	-0.410479861975550	0.000000964802245
29.000000000000000	0.525108549147585	0.255458140983382	-0.410480057173027	0.000000578387027
30.000000000000000	0.525108797929074	0.255458290895341	-0.410480174191503	0.000000346735877
31.000000000000000	0.525108947070504	0.255458380765708	-0.410480244342634	0.000000207863875
32.000000000000000	0.525109036478949	0.255458434641883	-0.410480286397370	0.000000124611825
33.000000000000000	0.525109090078206	0.255458466939984	-0.410480311608665	0.000000074703250

30 s.o.r - Matlab

Input data

$$A = \begin{pmatrix} 4 & -1 & 0 & 3 \\ 1 & 15.5 & 3 & 8 \\ 0 & -1.3 & -4 & -1.1 \\ 14 & 5 & -2 & 30 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

$$x0 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Result

16.000000000000000	0.525048838054285	0.255562087609626	-0.410492655431864	0.000267368334562
17.000000000000000	0.525153096855929	0.255388792096719	-0.410431010542021	0.000211426878513
18.000000000000000	0.525083032330798	0.255496702359134	-0.410531691160648	0.000163371507257
19.000000000000000	0.525121506112613	0.255442774363417	-0.410441485182732	0.000111917733452
20.000000000000000	0.525105541319758	0.255461263177453	-0.410504216337229	0.000067319451869
21.000000000000000	0.525108394432681	0.255461653797037	-0.410468618237813	0.000035714388124
22.000000000000000	0.525111497554948	0.255453839646213	-0.410484215061002	0.000017718668531
23.000000000000000	0.525106824710537	0.255462595617494	-0.410480615160611	0.000010557546628
24.000000000000000	0.525110917873488	0.255455725721314	-0.410478506228640	0.000008270250935
25.000000000000000	0.525108112407864	0.255460070529084	-0.410482345141617	0.000006440904033
26.000000000000000	0.525109680330846	0.255457853352381	-0.410478807970147	0.000004459353879
27.000000000000000	0.525109010211074	0.255458650387605	-0.410481311714314	0.000002711652727
28.000000000000000	0.525109149635733	0.255458624953548	-0.410479866117868	0.000001452527181
29.000000000000000	0.525109257661341	0.255458336405102	-0.410480516916483	0.000000720047620
30.000000000000000	0.525109078935774	0.255458675557160	-0.410480350801202	0.000000417805257
31.000000000000000	0.525109239686291	0.255458403801543	-0.410480280920073	0.000000323380915
32.000000000000000	0.525109127597958	0.255458578544970	-0.410480426786329	0.000000253724307
33.000000000000000	0.525109191360249	0.255458487567299	-0.410480288304344	0.000000177538239
34.000000000000000	0.525109163313918	0.255458521682970	-0.410480388092254	0.000000109124253
35.000000000000000	0.525109169889426	0.255458519030165	-0.410480329483345	0.000000059036251

31 Vandermonde - Matlab

Input data

$$\text{Tabla} = \begin{array}{|c|c|c|c|c|} \hline \text{x} & -1 & 0 & 3 & 4 \\ \hline \text{y} & 15.5 & 3 & 8 & 1 \\ \hline \end{array}$$

Result

```
Ux =

-(137*x^3)/120


Px =

- (137*x^3)/120 + (233*x^2)/40 - (83*x)/15 + 3
```

32 euler - Matlab

Input data

$$x = 2$$

$$xf = 4$$

$$y = 4$$

$$y = 0.5$$

$$f(x) = 0.1 * \sqrt{y} + 0.4 * x^2$$

Result

```
       Resolucion de ecuaciones diferenciales ordinarias por el metodo de euler
 Ingrese la derivada:
0.1*sqrt(y) + 0.4*x^2

 Ingrese el valor inicial de X:
2

 Ingrese el valor final de X:
4

 Ingrese el valor incial de Y:
4

 Ingrese el paso:
0.5
x     y

2.0 4.0000
2.5 4.9000
3.0 6.2607
3.5 8.1858
4.0 10.7788
```

Members signatures

Jose Miguel
Daniel Hernandez
Neller Pellegrino
Samuel Villegas