Ngoc Huynh
MATH 446
Project 4

<u>Question 1:</u> Implement the Matlab code for Gaussian Elimination of Section 2.1:

```
%Ngoc Huynh
for i=1:n-1
    for j=i+1:n
        m=a(j,i)/a(i,i);
        for k=i:n
            a(j,k)=a(j,k)-m*a(i,k);
        end
        b(j)=b(j)-m*b(i)
    end
end
for i=n:-1:1
    for j=i+1:n
        b(i)=b(i)-a(i,j)*x(j);
    end
    x(i)=b(i)/a(i,i);
end
x
```

Check that the code gives the correct solution for a small system of equations. I choose this system of equation, where:

a = [2 -2 -1; 4 1 -2; -2 1 -1]
b = [-2; 1; -3]

Running Matlab with the system above I obtained the following output:

```
>> a=[2 -2 -1;4 1 -2;-2 1 -1]

a =

     2    -2    -1
     4     1    -2
    -2     1    -1

>> b=[-2;1;-3]

b =

    -2
     1
    -3

>> aa = a;
>> bb = b

bb =

    -2
     1
    -3

>> elim
```

```
x =

     1     1     2

>> x = x'

x =

     1
     1
     2

>> aa*x

ans =

    -2
     1
    -3

>> b'

ans =

     2     5    -4

>> bb'

ans =

    -2     1    -3
```

Backward Error = ||Ax – b || = 0.000000000
The backward error is 0, which is very close to the machine epsilon.

<u>Question 2:</u> Let A be the n x n matrix whose (i, j) entry is Aij = tan(3i + 4j)

```
% Ngoc Huynh
n=8; %this n will be replaced every run
for i=1:n
    for j=1:n
        a(i,j)=tan(3*i+4*j);
    end
end
c=ones(n,1);
b=a*c;
aa=a
bb=b
for i=1:n-1
    for j=i+1:n
        m=a(j,i)/a(i,i);
        for k=i:n
            a(j,k)=a(j,k)-m*a(i,k);
        end
        b(j)=b(j)-m*b(i);
    end
end
```

```
x=zeros(n,1);
for i=n:-1:1
    for j=i+1:n
        b(i)=b(i)-a(i,j)*x(j);
    end
    x(i)=b(i)/a(i,i);
end
x
```

<u>n = 8:</u>

Output:

```
>> gel

aa =

   1.0e+02 *

   0.008714479827243  -2.259508464541951  -0.008559934009085   0.001515894706124   0.015881530833913  -0.032737038004281  -0.004416955680207   0.004738147204145
   0.006483608274591   0.072446066160948  -0.011373137123377   0.000088516560417   0.011787535542063  -0.064053311966463  -0.006234989627162   0.003103096609948
   0.004630211329365   0.034939156454748  -0.015274985276366  -0.001335264070215   0.008871428437982  -0.753130148000851  -0.008407712554028   0.001606566986806
   0.003006322420239   0.022371609442247  -0.021348966977217  -0.002814296045643   0.006610060414838   0.077504709056991  -0.011172149309239   0.000177046992787
   0.001515894706124   0.015881530833913  -0.032737038004281  -0.004416955680207   0.004738147204145   0.036145544071015  -0.014983873388552  -0.001245275681327
   0.000088516560417   0.011787535542063  -0.064053311966463  -0.006234989627162   0.003103096609948   0.022913879924375  -0.020866135311214  -0.002719006119976
  -0.001335264070215   0.008871428437982  -0.753130148000851  -0.008407712554028   0.001606566986806   0.016197751905439  -0.031729085521592  -0.004311581967196
  -0.002814296045643   0.006610060414838   0.077504709056991  -0.011172149309239   0.000177046992787   0.012001272431163  -0.060532723827928  -0.006112736881917


bb =

   1.0e+02 *

  -2.274372339664100
   0.012247384430965
  -0.728100746692558
   0.074334335995004
   0.004897974060829
  -0.055980414388013
  -0.772238044783654
   0.015661182831052

x =

   0.999999999902987
   0.999999999999999
   0.999999999999988
   0.999999999927758
   1.000000000011589
   1.000000000000001
   1.000000000001564
   1.000000000163843
```

Solve for Ax = b using Gaussian Elimination Code. Look at the result above for the x found, when n = 8.

<u>n = 12:</u>

Output:

```
aa =

   1.0e+02 *

  Columns 1 through 8

   0.008714479827243  -2.259508464541951  -0.008559934009085   0.001515894706124   0.015881530833913  -0.032737038004281  -0.004416955680207   0.004738147204145
   0.006483608274591   0.072446066160948  -0.011373137123377   0.000088516560417   0.011787535542063  -0.064053311966463  -0.006234989627162   0.003103096609948
   0.004630211329365   0.034939156454748  -0.015274985276366  -0.001335264070215   0.008871428437982  -0.753130148000851  -0.008407712554028   0.001606566986806
   0.003006322420239   0.022371609442247  -0.021348966977217  -0.002814296045643   0.006610060414838   0.077504709056991  -0.011172149309239   0.000177046992787
   0.001515894706124   0.015881530833913  -0.032737038004281  -0.004416955680207   0.004738147204145   0.036145544071015  -0.014983873388552  -0.001245275681327
   0.000088516560417   0.011787535542063  -0.064053311966463  -0.006234989627162   0.003103096609948   0.022913879924375  -0.020866135311214  -0.002719006119976
  -0.001335264070215   0.008871428437982  -0.753130148000851  -0.008407712554028   0.001606566986806   0.016197751905439  -0.031729085521592  -0.004311581967196
  -0.002814296045643   0.006610060414838   0.077504709056991  -0.011172149309239   0.000177046992787   0.012001272431163  -0.060532723827928  -0.006112736881917
  -0.004416955680207   0.004738147204145   0.036145544071015  -0.014983873388552  -0.001245275681327   0.009030861493754  -0.451830879105211  -0.008257740091968
  -0.006234989627162   0.003103096609948   0.022913879924375  -0.020866135311214  -0.002719006119976   0.006738001006481   0.083308568524905  -0.010975097786623
  -0.008407712554028   0.001606566986806   0.016197751905439  -0.031729085521592  -0.004311581967196   0.004846992267921   0.037431679442724  -0.014700382576632
  -0.011172149309239   0.000177046992787   0.012001272431163  -0.060532723827928  -0.006112736881917   0.003200403893796   0.023478603091954  -0.020400815980159

  Columns 9 through 12

   0.036145544071015  -0.014983873388552  -0.001245275681327   0.009030861493754
   0.022913879924375  -0.020866135311214  -0.002719006119976   0.006738001006481
   0.016197751905439  -0.031729085521592  -0.004311581967196   0.004846992267921
   0.012001272431163  -0.060532723827928  -0.006112736881917   0.003200403893796
   0.009030861493754  -0.451830879105211  -0.008257740091968   0.001697497520827
   0.006738001006481   0.083308568524905  -0.010975097786623   0.000265605177760
   0.004846992267921   0.037431679442724  -0.014700382576632  -0.001155485457945
   0.003200403893796   0.023478603091954  -0.020400815980159  -0.002624173775019
   0.001697497520827   0.016523172640102  -0.030776204031934  -0.004207009506211
   0.000265605177760   0.012219599181369  -0.057370225392790  -0.005991799983411
  -0.001155485457945   0.009192864044036  -0.322685757759344  -0.008109944158319
  -0.002624173775019   0.006867476893515   0.090036549456071  -0.010781838051640
```

```
bb =

   1.0e+02 *

  -2.245425083169210
   0.018314123930630
  -0.743096670007986
   0.022890551610117
  -0.444462286121769
   0.023356662534510
  -0.745815241107586
   0.019315200061623
  -0.447582714555567
   0.024391496203661
  -0.321824095348129
   0.024136914933382
```

```
x =

   1.000000031619747
   0.999999999999861
   0.999999999999881
   0.999999999991115
   1.000000002161472
   1.000000000000019
   0.999999999999908
   0.999999999749194
   0.999999999793060
   0.999999999999983
   0.999999999999225
   0.999999966613188
```

Solve for Ax = b using Gaussian Elimination Code. Look at the result above for the x found, when n = 12.

<u>n = 16:</u>

Output:

```
aa =

   1.0e+02 *

  Columns 1 through 8

   0.008714479827243  -2.259508464541951  -0.008559934009085   0.001515894706124   0.015881530833913  -0.032737038004281  -0.004416955680207   0.004738147204145
   0.006483608274591   0.072446066160948  -0.011373137123377   0.000088516560417   0.011787535542063  -0.064053311966463  -0.006234989627162   0.003103096609948
   0.004630211329365   0.034939156454748  -0.015274985276366  -0.001335264070215   0.008871428437982  -0.753130148000851  -0.008407712554028   0.001606566986806
   0.003006322420239   0.022371609442247  -0.021348966977217  -0.002814296045643   0.006610060414838   0.077504709056991  -0.011172149309239   0.000177046992787
   0.001515894706124   0.015881530833913  -0.032737038004281  -0.004416955680207   0.004738147204145   0.036145544071015  -0.014983873388552  -0.001245275681327
   0.000088516560417   0.011787535542063  -0.064053311966463  -0.006234989627162   0.003103096609948   0.022913879924375  -0.020866135311214  -0.002719006119976
  -0.001335264070215   0.008871428437982  -0.753130148000851  -0.008407712554028   0.001606566986806   0.016197751905439  -0.031729085521592  -0.004311581967196
  -0.002814296045643   0.006610060414838   0.077504709056991  -0.011172149309239   0.000177046992787   0.012001272431163  -0.060532723827928  -0.006112736881917
  -0.004416955680207   0.004738147204145   0.036145544071015  -0.014983873388552  -0.001245275681327   0.009030861493754  -0.451830879105211  -0.008257740091968
  -0.006234989627162   0.003103096609948   0.022913879924375  -0.020866135311214  -0.002719006119976   0.006738001006481   0.083308568524905  -0.010975097786623
  -0.008407712554028   0.001606566986806   0.016197751905439  -0.031729085521592  -0.004311581967196   0.004846992267921   0.037431679442724  -0.014700382576632
  -0.011172149309239   0.000177046992787   0.012001272431163  -0.060532723827928  -0.006112736881917   0.003200403893796   0.023478603091954  -0.020400815980159
  -0.014983873388552  -0.001245275681327   0.009030861493754  -0.451830879105211  -0.008257740091968   0.001697497520827   0.016523172640102  -0.030776204031934
  -0.020866135311214  -0.002719006119976   0.006738001006481   0.083308568524905  -0.010975097786623   0.000265605177760   0.012219599181369  -0.057370225392790
  -0.031729085521592  -0.004311581967196   0.004846992267921   0.037431679442724  -0.014700382576632  -0.001155485457945   0.009192864044036  -0.322685757759344
  -0.060532723827928  -0.006112736881917   0.003200403893796   0.023478603091954  -0.020400815980159  -0.002624173775019   0.006867476893515   0.090036549456071

  Columns 9 through 16

   0.036145544071015  -0.014983873388552  -0.001245275681327   0.009030861493754  -0.451830879105211  -0.008257740091968   0.001697497520827   0.016523172640102
   0.022913879924375  -0.020866135311214  -0.002719006119976   0.006738001006481   0.083308568524905  -0.010975097786623   0.000265605177760   0.012219599181369
   0.016197751905439  -0.031729085521592  -0.004311581967196   0.004846992267921   0.037431679442724  -0.014700382576632  -0.001155485457945   0.009192864044036
   0.012001272431163  -0.060532723827928  -0.006112736881917   0.003200403893796   0.023478603091954  -0.020400815980159  -0.002624173775019   0.006867476893515
   0.009030861493754  -0.451830879105211  -0.008257740091968   0.001697497520827   0.016523172640102  -0.030776204031934  -0.004207009506211   0.004956775331814
   0.006738001006481   0.083308568524905  -0.010975097786623   0.000265605177760   0.012219599181369  -0.057370225392790  -0.005991799983411   0.003298264065077
   0.004846992267921   0.037431679442724  -0.014700382576632  -0.001155485457945   0.009192864044036  -0.322685757759344  -0.008109944158319   0.001788701724388
   0.003200403893796   0.023478603091954  -0.020400815980159  -0.002624173775019   0.006867476893515   0.090036549456071  -0.010781838051640   0.000354205013394
   0.001697497520827   0.016523172640102  -0.030776204031934  -0.004207009506211   0.004956775331814   0.038805963103842  -0.014424174716642  -0.001065878721054
   0.000265605177760   0.012219599181369  -0.057370225392790  -0.005991799983411   0.003298264065077   0.024067297096422  -0.019952004122082  -0.002529780967614
  -0.001155485457945   0.009192864044036  -0.322685757759344  -0.008109944158319   0.001788701724388   0.016858253705060  -0.029873862594340  -0.004103212990482
  -0.002624173775019   0.006867476893515   0.090036549456071  -0.010781838051640   0.000354205013394   0.012442700581287  -0.054513401108232  -0.005872139151569
  -0.004207009506211   0.004956775331814   0.038805963103842  -0.014424174716642  -0.001065878721054   0.009357524720632  -0.250925349796765  -0.007964255049200
  -0.005991799983411   0.003298264065077   0.024067297096422  -0.019952004122082  -0.002529780967614   0.006998536538095   0.097929802635358  -0.010592232274910
  -0.008109944158319   0.001788701724388   0.016858253705060  -0.029873862594340  -0.004103212990482   0.005067526002248   0.040278017638844  -0.014154931063390
  -0.010781838051640   0.000354205013394   0.012442700581287  -0.054513401108232  -0.005872139151569   0.003396697316951   0.024681619615828  -0.019518769927439
```

```
bb =

   1.0e+02 *

  -2.687293032205460
   0.103132799028042
  -0.712327994555803
   0.030211641840408
  -0.457965551687998
  -0.024487499595245
  -1.065629377256825
   0.105791593372963
  -0.419310029557607
   0.029275272275464
  -0.337154215503504
  -0.023451719731738
  -0.705308845277893
   0.103829392266847
  -0.315360209264019
  -0.015898342841110


x =

   0.999999992273037
   0.999999999999163
   0.999999999999758
   1.000000000007490
   0.999999966184438
   1.000000000000015
   0.999999999999763
   0.999999999991853
   0.999999999575934
   0.999999999999987
   0.999999999999851
   1.000000009839256
   1.000000000002371
   0.999999999999931
   0.999999999993652
   1.000000032080072
```

Solve for Ax = b using Gaussian Elimination Code. Look at the result above for the x found, when n = 16.

Question: *How large would n have to be to lose half of the correct digits, i.e for RFE to exceed 0.5×10^{-8}?*

Matlab Code I used to obtain the following values:

```
RFE=max(abs(c-x))/max(abs(c))
RBE=max(abs(bb-aa*x))/max(abs(bb))
EMF = RFE/RBE
COND = cond(aa,inf)
```

Look at the table:

	RFE	RBE	EMF	COND(a,inf)
n = 8	**1.638429392158969e-10**	**6.597765079683392e-15**	**2.483309684978345e+04**	**1.503622444634095e+06**
n = 10	**8.173200893857313e-09**			
n = 12	**3.338681175968361e-08**	**1.682314705193813e-13**	**1.984575873741604e+05**	**2.138730163764400e+07**
n = 16	**3.381556168413624e-08**	**8.066570159183062e-13**	**4.192061931754263e+04**	**2.560665603425178e+06**

→ n has to be larger than n = 10 for RFE to exceed 0.5×10^{-8}

<u>Question 3:</u> Repeat step 2 for Aij = cos(sin(3i + 4j))

<u>n = 8</u>

Output:

```
aa =

   0.791836209014479   0.540310546745653   0.795909568622799   0.988789420040569   0.662817961369183   0.576485022196244   0.919481157301567   0.909721840267583
   0.855634354821367   0.548181994273030   0.731015566745341   0.999960827417674   0.723071068995164   0.550334409962843   0.863270440189559   0.956403346027634
   0.913020816562331   0.572374612843129   0.669949444253653   0.991254284859670   0.787759024788576   0.540376470931675   0.799977784713449   0.987445780836918
   0.958841320080304   0.611417804419412   0.617230638219365   0.963528898818160   0.851777946625377   0.547201825560528   0.735011178059940   0.999843325015127
   0.988789420040569   0.662817961369183   0.576485022196244   0.919481157301567   0.909721840267583   0.570406714334183   0.673565059665817   0.992374552663789
   0.999960827417674   0.723071068995164   0.550334409962843   0.863270440189559   0.956403346027634   0.608583948080286   0.620208114167993   0.965776724822844
   0.991254284859670   0.787759024788576   0.540376470931675   0.799977784713449   0.987445780836918   0.659304076308518   0.578626534946618   0.922640164050383
   0.963528898818160   0.851777946625377   0.547201825560528   0.735011178059940   0.999843325015127   0.719124567298015   0.551506155742815   0.867047474039951


bb =

   6.185351725558077
   6.227872008432612
   6.262158219789402
   6.284852936798213
   6.293641727838936
   6.287608879663997
   6.267384121435806
   6.235041371159915


x =

   1.000000101985430
   1.000000039024275
   1.000000035777242
   1.000000056929115
   0.999999942074150
   0.999999963747986
   0.999999961822339
   0.999999898638619
```

Solve for Ax = b using Gaussian Elimination Code. Look at the result above for the x found, when n = 8.

<u>n = 12</u>
Output:

```
aa =

  Columns 1 through 8

   0.791836209014479   0.540310546745653   0.795909568622799   0.988789420040569   0.662817961369183   0.576485022196244   0.919481157301567   0.909721840267583
   0.855634354821367   0.548181994273030   0.731015566745341   0.999960827417674   0.723071068995164   0.550334409962843   0.863270440189559   0.956403346027634
   0.913020816562331   0.572374612843129   0.669949444253653   0.991254284859670   0.787759024788576   0.540376470931675   0.799977784713449   0.987445780836918
   0.958841320080304   0.611417804419412   0.617230638219365   0.963528898818160   0.851777946625377   0.547201825560528   0.735011178059940   0.999843325015127
   0.988789420040569   0.662817961369183   0.576485022196244   0.919481157301567   0.909721840267583   0.570406714334183   0.673565059665817   0.992374552663789
   0.999960827417674   0.723071068995164   0.550334409962843   0.863270440189559   0.956403346027634   0.608583948080286   0.620208114167993   0.965776724822844
   0.991254284859670   0.787759024788576   0.540376470931675   0.799977784713449   0.987445780836918   0.659304076308518   0.578626534946618   0.922640164050383
   0.963528898818160   0.851777946625377   0.547201825560528   0.735011178059940   0.999843325015127   0.719124567298015   0.551506155742815   0.867047474039951
   0.919481157301567   0.909721840267583   0.570406714334183   0.673565059665817   0.992374552663789   0.783679330311021   0.540508304188228   0.804039534310583
   0.863270440189559   0.956403346027634   0.608583948080286   0.620208114167993   0.965776724822844   0.847897730184077   0.546285991536423   0.739020757161725
   0.799977784713449   0.987445780836918   0.659304076308518   0.578626534946618   0.922640164050383   0.906378587279118   0.568497659162202   0.677212579338368
   0.735011178059940   0.999843325015127   0.719124567298015   0.551506155742815   0.867047474039951   0.953903804653362   0.605799440406546   0.623231935836399

  Columns 9 through 12

   0.570406714334183   0.673565059665817   0.992374552663789   0.783679330311021
   0.608583948080286   0.620208114167993   0.965776724822844   0.847897730184077
   0.659304076308518   0.578626534946618   0.922640164050383   0.906378587279118
   0.719124567298015   0.551506155742815   0.867047474039951   0.953903804653362
   0.783679330311021   0.540508304188228   0.804039534310583   0.986028689462143
   0.847897730184077   0.546285991536423   0.739020757161725   0.999647538816758
   0.906378587279118   0.568497659162202   0.677212579338368   0.993419453220166
   0.953903804653362   0.605799440406546   0.623231935836399   0.967959427108159
   0.986028689462143   0.655826060031135   0.580824894812507   0.925750179880196
   0.999647538816758   0.715196802264638   0.552741233769289   0.870795414138629
   0.993419453220166   0.779598434963957   0.540706016282565   0.808093490829612
   0.967959427108159   0.843995036135214   0.545434704536177   0.743043086128806

bb =

   9.205377382532888
   9.270338525687812
   9.329107582374039
   9.376434938532356
   9.407897586110913
   9.420460897362981
   9.412892400435661
   9.385935979164381
   9.342206317228753
   9.285828041159856
   9.221900561931875
   9.155900134960509


x =

   0.800311319467813
   0.995241833380079
   1.003683727237312
   0.993371223564931
   0.987549472482774
   1.003815708490066
   0.996144671456617
   1.012540520150345
   1.006547660690635
   0.996329917906944
   1.004822625024931
   1.199641320153447
```

Solve for Ax = b using Gaussian Elimination Code. Look at the result above for the x found, when n = 12.

n = 16

Output:

```
aa =

  Columns 1 through 8

   0.791836209014479   0.540310546745653   0.795909568622799   0.988789420040569   0.662817961369183   0.576485022196244   0.919481157301567   0.909721840267583
   0.855634354821367   0.548181994273030   0.731015566745341   0.999960827417674   0.723071068995164   0.550334409962843   0.863270440189559   0.956403346027634
   0.913020816562331   0.572374612843129   0.669949444253653   0.991254284859670   0.787759024788576   0.540376470931675   0.799977784713449   0.987445780836918
   0.958841320080304   0.611417804419412   0.617230638219365   0.963528898818160   0.851777946625377   0.547201825560528   0.735011178059940   0.999843325015127
   0.988789420040569   0.662817961369183   0.576485022196244   0.919481157301567   0.909721840267583   0.570406714334183   0.673565059665817   0.992374552663789
   0.999960827417674   0.723071068995164   0.550334409962843   0.863270440189559   0.956403346027634   0.608583948080286   0.620208114167993   0.965776724822844
   0.991254284859670   0.787759024788576   0.540376470931675   0.799977784713449   0.987445780836918   0.659304076308518   0.578626534946618   0.922640164050383
   0.963528898818160   0.851777946625377   0.547201825560528   0.735011178059940   0.999843325015127   0.719124567298015   0.551506155742815   0.867047474039951
   0.919481157301567   0.909721840267583   0.570406714334183   0.673565059665817   0.992374552663789   0.783679330311021   0.540508304188228   0.804039534310583
   0.863270440189559   0.956403346027634   0.608583948080286   0.620208114167993   0.965776724822844   0.847897730184077   0.546285991536423   0.739020757161725
   0.799977784713449   0.987445780836918   0.659304076308518   0.578626534946618   0.922640164050383   0.906378587279118   0.568497659162202   0.677212579338368
   0.735011178059940   0.999843325015127   0.719124567298015   0.551506155742815   0.867047474039951   0.953903804653362   0.605799440406546   0.623231935836399
   0.673565059665817   0.992374552663789   0.783679330311021   0.540508304188228   0.804039534310583   0.986028689462143   0.655826060031135   0.580824894812507
   0.620208114167993   0.965776724822844   0.847897730184077   0.546285991536423   0.739020757161725   0.999647538816758   0.715196802264638   0.552741233769289
   0.578626534946618   0.922640164050383   0.906378587279118   0.568497659162202   0.677212579338368   0.993419453220166   0.779598434963957   0.540706016282565
   0.551506155742815   0.867047474039951   0.953903804653362   0.605799440406546   0.623231935836399   0.967959427108159   0.843995036135214   0.545434704536177

  Columns 9 through 16

   0.570406714334183   0.673565059665817   0.992374552663789   0.783679330311021   0.540508304188228   0.804039534310583   0.986028689462143   0.655826060031135
   0.608583948080286   0.620208114167993   0.965776724822844   0.847897730184077   0.546285991536423   0.739020757161725   0.999647538816758   0.715196802264638
   0.659304076308518   0.578626534946618   0.922640164050383   0.906378587279118   0.568497659162202   0.677212579338368   0.993419453220166   0.779598434963957
   0.719124567298015   0.551506155742815   0.867047474039951   0.953903804653362   0.605799440406546   0.623231935836399   0.967959427108159   0.843995036135214
   0.783679330311021   0.540508304188228   0.804039534310583   0.986028689462143   0.655826060031135   0.580824894812507   0.925750179880196   0.902992268754990
   0.847897730184077   0.546285991536423   0.739020757161725   0.999647538816758   0.715196802264638   0.552741233769289   0.870795414138629   0.951343639579128
   0.906378587279118   0.568497659162202   0.677212579338368   0.993419453220166   0.779598434963957   0.540706016282565   0.808093490829612   0.984538695111335
   0.953903804653362   0.605799440406546   0.623231935836399   0.967959427108159   0.843995036135214   0.545434704536177   0.743043086128806   0.999373545500257
   0.986028689462143   0.655826060031135   0.580824894812507   0.925750179880196   0.902992268754990   0.566647907146516   0.680890977137955   0.994388579550175
   0.999647538816758   0.715196802264638   0.552741233769289   0.870795414138629   0.951343639579128   0.603064993029241   0.626301313580225   0.970076173504100
   0.993419453220166   0.779598434963957   0.540706016282565   0.808093490829612   0.984538695111335   0.652384867913855   0.583079563418264   0.928810061750628
   0.967959427108159   0.843995036135214   0.545434704536177   0.743043086128806   0.999373545500257   0.711288938910725   0.554039355038984   0.874512951212444
   0.925750179880196   0.902992268754990   0.566647907146516   0.680890977137955   0.994388579550175   0.775517642705054   0.540969561861158   0.812138324552615
   0.870795414138629   0.951343639579128   0.603064993029241   0.626301313580225   0.970076173504100   0.840071197699607   0.544648161629352   0.747076937393342
   0.808093490829612   0.984538695111335   0.652384867913855   0.583079563418264   0.928810061750628   0.899564107720386   0.564857902451582   0.684599213260593
   0.743043086128806   0.999373545500257   0.711288938910725   0.554039355038984   0.874512951212444   0.948723815238889   0.600381301876152   0.629415442289963
```

```
bb =

  12.191779970524976
  12.270489615467357
  12.347835709058732
  12.417420778018673
  12.473290989589740
  12.510537987114665
  12.525829037623129
  12.517782351464835
  12.487126049818389
  12.436614160852551
  12.370713750125955
  12.295114925622919
  12.216141867033885
  12.140152723277373
  12.073007331699630
  12.019656414654841


x =

  18.873948372761841
 -23.513203011237287
 -13.041605417827951
  -7.252784114654585
  31.613749460716075
  -0.447097646415438
   0.146947886385413
   0.094165905859961
   3.784514701252850
   2.219866477817299
   1.690378177754307
 -18.119275293809576
  25.094497584700285
  14.886883377067559
   9.258091193774403
 -29.289077654144716
```

Solve for Ax = b using Gaussian Elimination Code. Look at the result above for the x found, when n = 16.

```
COND =

     8.060600085183927e+17
```

Question: *How large would n have to be to lose half of the correct digits, i.e for RFE to exceed 0.5×10^{-8}?*

Matlab Code I used to obtain the following values:

```
RFE=max(abs(c-x))/max(abs(c))
RBE=max(abs(bb-aa*x))/max(abs(bb))
EMF = RFE/RBE
COND = cond(aa,inf)
```

Look at the table:

	RFE	RBE	EMF	COND(a,inf)
n = 7	1.673975069671485e-09			
n = 8	1.019854303230261e-07	2.822462599901127e-16	3.613349219458168e+08	2.608082943790527e+09
n = 12	0.199688680532187	3.771273738628853e-16	5.294993001616178e+14	3.045893040675552e+15
n = 16	30.613749460716075	2.410863678491860e-15	1.269824989850393e+16	8.060600085183927e+17

→ n has to be larger than or equal n = 8 for RFE to exceed 0.5×10^{-8}.

<u>Question 4:</u> *Compare the results obtaining from the matrices A in step 2 and step 3, out of the 6 systems:*

	n	RFE
A_{ij} = tan(3i + 4j)	n = 8	1.638429392158969e-10
	n = 12	3.338681175968361e-08
	n = 16	3.381556168413624e-08
A_{ij} = cos(sin(3i + 4j))	n = 8	1.019854303230261e-07
	n = 12	0.199688680532187
	n = 16	30.613749460716075

- **For Aij = tan(3i + 4j):**
  With n = 8, the RFE predicts at least 10 correct digits.
  With n = 12, the RFE predicts at least 8 correct digits.
  With n = 16, the RFE predicts at least 8 correct digits.
- **For Aij = cos(sin(3i + 4j)):**
  With n = 8, the RFE predicts at least 7 correct digits.
  With n = 12, the RFE predicts at least 1 correct digit.
  With n = 16, the RFE predicts at least 0 correct digits.

→ When n = 12 and n = 16 for matrix Aij = cos(sin(3i + 4j) were total failures because they only have up to 1 and 0 correct digits, respectively.
→ When n = 12 and n = 16 for matrix Aij = tan(3i + 4j) and when n = 8 for matrix Aij = cos(sin(3i + 4j) could be solved with at least 8 and 7 correct digits, respectively.
→ When n = 8 for matrix Aij = tan(3i + 4j) was the best out of 6 systems because it could be solved with up to 10 correct digits.