Raghav Aggarwal
1215935292

Q1) a)

```
>> n=1500

n =

        1500

>>  A = floor(15*rand(n));
>>  z = ones(n,1);

>> b = A*z;
```

(i)

```
>>tic, x = A\b;   toc
Elapsed time is 0.131731 seconds.

>> tic, y = inv(A)*b; toc
Elapsed time is 0.190335 seconds.
```

x = A\b is faster than inv(A)*b

(ii)

```
>> sum(abs(x - z))

ans =

   9.9041e-10

>> sum(abs(y - z))

ans =

   1.0541e-08
```

sum(abs(x - z)) is more accurate as it is smaller in value

Q1 b) Repeat part (a) using n = 3000 and n = 6000.

```
>> n=3000

n =

        3000

>> A = floor(15*rand(n));
>> z = ones(n,1);
>> b = A*z;

>> tic, x = A\b;   toc
Elapsed time is 0.412962 seconds.
```

Raghav Aggarwal
1215935292

```
>> tic, y = inv(A)*b; toc
Elapsed time is 0.880246 seconds.
```

A\b; is faster than inv(A)*b

```
>> sum(abs(x - z))

ans =

  4.6847e-09

>> sum(abs(y - z))

ans =

  5.6461e-08
```

sum(abs(x - z)) is more accurate as it is smaller in value

```
n =

     6000

>> A = floor(15*rand(n));
>> z = ones(n,1);
>> b = A*z;
>> tic, x = A\b;   toc
Elapsed time is 2.777828 seconds.
>> tic, y = inv(A)*b; toc
Elapsed time is 6.126905 seconds.
```

A\b; is faster than inv(A)*b

```
>> sum(abs(x - z))

ans =

  2.2096e-07

>> sum(abs(y - z))

ans =

  8.6134e-07
```

sum(abs(x - z)) is more accurate as it is smaller in value

Q1 C) Explain why the exact solution of the system Ax = b is the vector z.

The product of Ax is usually a diagonal matrix with the value same of that of the vector z. Hence the exact solution of the system Ax = b is the vector z.

Raghav Aggarwal
1215935292

Q2)

```
n =

   70

>> B = eye(n) - triu(ones(n),1);
>> A=B'*B;
>> z = ones(n,1);
>> b= A*z;
>> x = A\b;
>> y = inv(A)*b;

RCOND = 1.832605e-45

>> sum(abs(x - z))

ans =

     0

>> sum(abs(y - z))

ans =

  1.5032e+27
```

sum(abs(x - z)) is more accurate.

Q3)

Part A:

```
 >> A = floor(30*rand(6));
>> b = floor(60*rand(6,1))-30;
>> x = A\b;
>> x

x =

  -1.8673
   0.9673
   0.4398
  -3.5880
   2.2856
   3.0653
```

Part B:

```
>> U = rref([A, b])

U =
```

Raghav Aggarwal
1215935292

```
    1.0000         0         0         0         0         0   -1.8673
         0    1.0000         0         0         0         0    0.9673
         0         0    1.0000         0         0         0    0.4398
         0         0         0    1.0000         0         0   -3.5880
         0         0         0         0    1.0000         0    2.2856
         0         0         0         0         0    1.0000    3.0653
```

This is same as Answer from part A.

PART C:

```
>> U(:,7) - x

ans =

   1.0e-04 *

    0.0163
   -0.1013
    0.0476
    0.0333
    0.0023
   -0.0427
```

There is a very negligible gap between answer from part a and b. Which will work.

Part D:

```
>> A(:,6) = 2*A(:,4)+4*A(:,1);
>> U = rref([A, b])

U =

     1     0     0     0     0     4     0
     0     1     0     0     0     0     0
     0     0     1     0     0     0     0
     0     0     0     1     0     2     0
     0     0     0     0     1     0     0
     0     0     0     0     0     0     1
```

There are 5 solutions to this problem.

Part e)

```
>> y = floor(60*rand(6,1)) - 30;
>> c = A*y;
>> U = rref([A, c])

U =

     1     0     0     0     0     4   -35
     0     1     0     0     0     0    -3
     0     0     1     0     0     0    23
```

```
     0     0     0     1     0     2   -34
     0     0     0     0     1     0    -9
     0     0     0     0     0     0     0
```

The reason vector of the matrix will always have a solution is because the matrix becomes a diagonal matrix after multiplication so will have value equal to the vector and will defiantly have at least one value.

Q4)
myrowproduct.m

```
function y = myrowproduct(A,x)
[m,n] = size(A);
[p,q] = size(x);
if (q==1 && p==n)
    y = zeros(m,1);
    for i = 1:m
        y = [y; A(i,:)*x]
    end

    else
        disp('dimensions do not match')
        y = [];
    end
end
```

```
>> A=rand(4,6);
>> x = rand(6,1);
>> y = myrowproduct(A,x)

y =

    1.8769
    1.6381
    2.3084
    1.7965

>> A*x

ans =

    1.8769
    1.6381
    2.3084
    1.7965
```

Answer of A*x is same as that of mrrowproduct.m

```
>> A = rand(5,3);
>> x = rand(3,1);
>> y = myrowproduct(A,x)
```

Raghav Aggarwal
1215935292

```
y =

   0.6383
   0.3500
   0.9246
   0.5101
   0.4332

>> A*x

ans =

   0.6383
   0.3500
   0.9246
   0.5101
   0.4332
```

Answer of A*x is same as that of mrrowproduct.m

```
>> A = rand(5,3);
>> x = rand(1,3);
>> y = myrowproduct(A,x)
```
dimensions do not match

```
y =

    []
```

Q 5)
Part A)

```
function y = columnproduct(A,B)
[m,n] = size(A);
[p,q] = size(B);
C=[];
if (p==n)
   for i = 1:q
      C = [C,A*B(:,i)];
   end
   else
      disp('dimensions do not match')
end
   disp('C=');
   disp(C);
end
```

```
>> A = rand (3,5);
>> B = rand(5,3);
>> y = columnproduct(A,B)
```

```
C =

    1.1240    2.0322    1.3930
    0.6190    1.0938    0.5791
    0.4918    0.8714    0.5801

>> A*B

ans =

    1.1240    2.0322    1.3930
    0.6190    1.0938    0.5791
    0.4918    0.8714    0.5801
```

A*B is the same as result produced by columnproduct.m

```
>> A = rand (4,6);
>> B = rand(6,2);
>> y = columnproduct(A,B)
C=
    1.5812    0.9591
    1.3101    0.4351
    1.7868    0.8589
    2.0226    1.1596

>> A*B

ans =

    1.5812    0.9591
    1.3101    0.4351
    1.7868    0.8589
    2.0226    1.1596
```

A*B is the same as result produced by columnproduct.m

Part iii)

```
>> A = rand (4,6);
>> B = rand(2,6);
>> y = columnproduct(A,B)
```

dimensions do not match

```
>> A*B
Error using  *
Inner matrix dimensions must agree.
```

Q5
Part B)
Rowproduct.m

Raghav Aggarwal
1215935292

```
function y = rowproduct(A,B)
[m,n] = size(A);
[p,q] = size(B);
C=[];
if (p==n)
   for i = 1:m
      C = [C;A(i,:)*B]
   end
   else
      disp('dimensions do not match')
end
end
```

```
>> A = rand (3,5);
>> B = rand(5,3);
>> y = columnproduct(A,B)
C=
   1.0583    2.4312    1.1918
   0.8867    1.1261    0.8290
   0.7342    1.2216    0.7261

>> A*B

ans =

   1.0583    2.4312    1.1918
   0.8867    1.1261    0.8290
   0.7342    1.2216    0.7261
```

A*B is the same as result produced by rowproduct.m

```
>> A = rand (4,6);

>> B = rand(6,2);


  >> y = columnproduct(A,B)
C=
   1.1486    0.7353
   0.7385    0.7071
   1.5282    1.2789
   1.5053    0.9240

>> A*B

ans =

   1.1486    0.7353
   0.7385    0.7071
   1.5282    1.2789
   1.5053    0.9240
```

MAT 343 LAB 2

Raghav Aggarwal
1215935292

A*B is the same as result produced by rowproduct.m

>> A = rand (4,6);

>> B = rand(2,6);

>> y = columnproduct(A,B)
dimensions do not match
C=

>> A*B
Error using *
Inner matrix dimensions must agree.