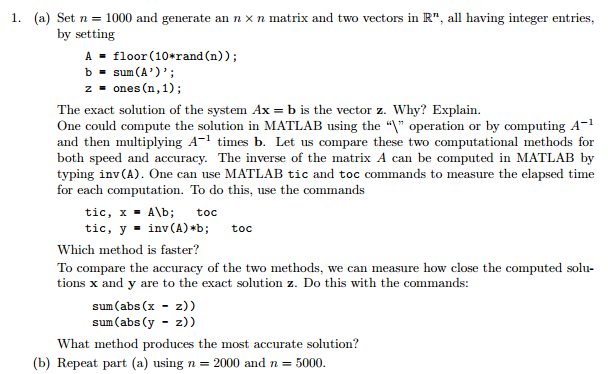
**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**

****

%EXERCISES GROUP 1

%Set n = 1000 and generate an n x n matrix.

n = 1000;A=floor(10\*rand(n));b=sum(A')';z=ones(n,1);

% The exact solution of the system Ax = b is the vector z.

% The operation A\b takes each element of b, which is a 1000x1

% vertical vector and divides to matrix A's sums of columns,

% which is a 1x1000 vertical vector. The net effect is a

% 1000x1 vertical vector of 1's.

% Elapse times for '\' and inv() operations

tic, x = A\b; toc

elapsed\_time =

**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**

0.2930

tic, y = inv(A)\*b; toc

elapsed\_time =

0.7440

% The data showed a faster execution time using the '\'

% operation.

% Compare the accuracy of the two methods.

sum(abs(x - z))

ans =

1.4653e-009

sum(abs(y - z))

ans =

1.8085e-008

% As shown above, the '\' operation was more accurate

% at least by an order of magnitude (i.e smaller error band).

**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**

% n = 2000

n = 2000;A=floor(10\*rand(n));b=sum(A')';z=ones(n,1);

tic, x = A\b; toc

elapsed\_time =

1.7640

tic, y = inv(A)\*b; toc

elapsed\_time =

5.1530

% n = 5000

n = 5000;A=floor(10\*rand(n));b=sum(A')';z=ones(n,1);

??? Error using ==> \*

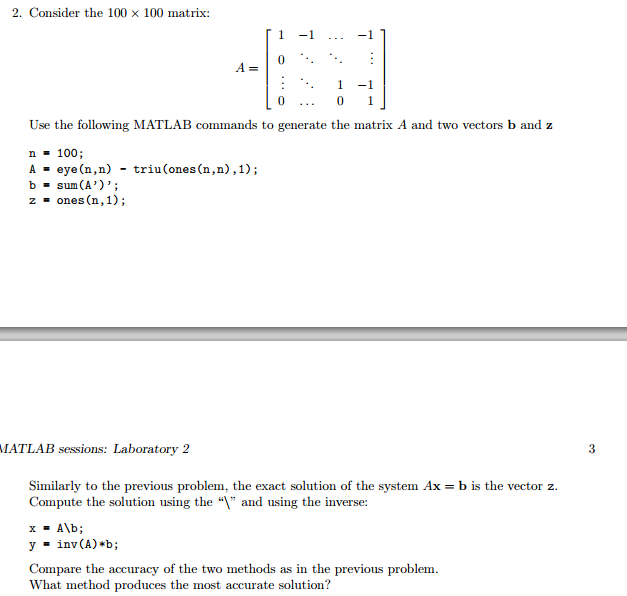
Out of memory. Type HELP MEMORY for your options.

% It seemed that MatLab was not capable of handling a 5000x5000

% square matrix, but I expected a similar result as n = 2000.

% (In my laptop only).

**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**

****

% PART 2

n = 100;A = eye(n,n) - triu(ones(n,n),1);

n = 100;A = eye(n,n) - triu(ones(n,n),1);b = sum(A')';z = ones(n,1);

**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**

x = A\b;

y = inv(A)\*b;

Warning: Matrix is close to singular or badly scaled.

Results may be inaccurate. RCOND = 1.577722e-032.

sum(abs(x - z))

ans =

0

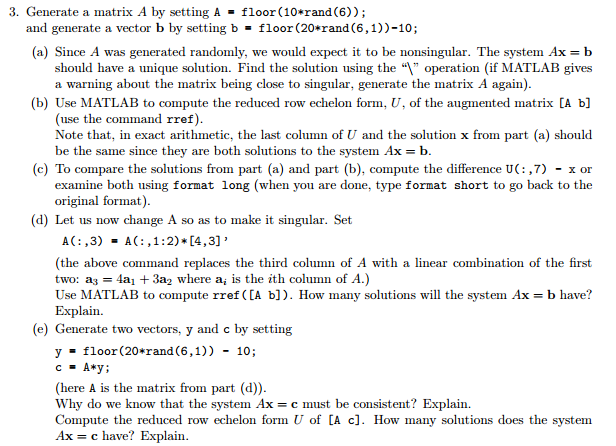
sum(abs(y - z))

ans =

45

% The '\' method is still more accurate than the inv() method.

**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**

****

% PART 3

A = floor(10\*rand(6));

A = floor(10\*rand(6));

A = floor(10\*rand(6));

b = floor(20\*rand(6,1))-10;

x=A\b;

ar=rref([A b]);

ar

**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**

ar =

Columns 1 through 6

1.0000 0 0 0 0 0

0 1.0000 0 0 0 0

0 0 1.0000 0 0 0

0 0 0 1.0000 0 0

0 0 0 0 1.0000 0

0 0 0 0 0 1.0000

Column 7

0.0413

-2.5688

-1.3899

7.7202

-0.0092

-3.9862

x

x =

0.0413

-2.5688

**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**

-1.3899

7.7202

-0.0092

-3.9862

% rref([A b]) = A\b

ar(:,7)-x

ans =

1.0e-014 \*

0.0756

-0.1332

-0.0888

-0.0888

0.0673

0.1332

% Observation: Very small error band.

A(:,3) = A(:,1:2)\*[4,3]'

A =

0 8 24 7 6 9

8 6 50 5 3 5

**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**

9 8 60 7 1 7

7 8 52 8 4 7

0 6 18 9 8 9

8 9 59 5 8 3

ar=rref([A b])

ar =

1 0 4 0 0 0 0

0 1 3 0 0 0 0

0 0 0 1 0 0 0

0 0 0 0 1 0 0

0 0 0 0 0 1 0

0 0 0 0 0 0 1

% The last row of all zeros in the coeefficient side means

% a mztrix with inconsistent solution set.

% In this case, no solution.

y = floor(20\*rand(6,1)) - 10;

c = A\*y;

A\c

Warning: Matrix is close to singular or badly scaled.

Results may be inaccurate. RCOND = 3.610904e-018.

ans =

**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**

24.6250

21.2187

1.5938

-6.0000

2.0000

1.0000

c

c =

187

385

454

392

127

471

A

A =

0 8 24 7 6 9

8 6 50 5 3 5

9 8 60 7 1 7

**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**

7 8 52 8 4 7

0 6 18 9 8 9

8 9 59 5 8 3

y

y =

7

8

6

-6

2

1

U=rref([A c])

U =

1 0 4 0 0 0 31

0 1 3 0 0 0 26

0 0 0 1 0 0 -6

0 0 0 0 1 0 2

0 0 0 0 0 1 1

0 0 0 0 0 0 0

**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**

% Ax = c is consistent because with many solutions

% because the last row is [0,0,0,...,0] and we don't

% have a pivot on the third column.

% U of [A c] has many solutions due to a 'free' term

% in column 3.

myrowproduct.m file

%------------------------------------------------------------------------------------------

%Write a function M-file that takes as input a matrix A and a vector x, and as output %gives the

% product y = Ax by row, as defined above (Hint: use a for loop to define each entry of %the vector

% y.) The M-file should perform a check on the dimensions of the input variables A and %x and return

% a message if the dimensions do not match. Call the file myrowproduct.m.

function [y] = myrowproduct(A,x)

[rA,cA] = size(A); % determine the dimension of A

[rx,cx] = size(x); % determine the dimension of x

if(cA == rx) % check the dimensions, columns of A

% must equal rows of x

y = zeros(rA,1); % initialize the vector y, a 6x1 vector

for i = 1:rA

y(i) = A(i,:)\*x; % implement the formula

end % End for loop

**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**

else % if dimension not agreeable

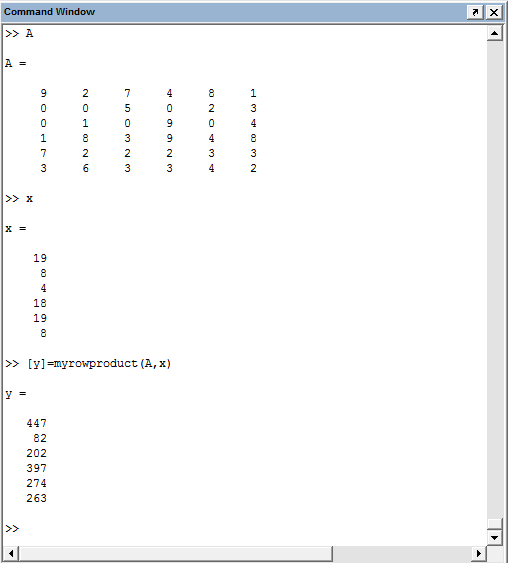
disp('dimensions do not match')% throw an error message

y = []; % return an empty matrix

end % end if

end % end function

%-----------------------------------------------------------------------------------



**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**

rowproduct.m

%---------------------------------------------------------------------------------------------

% Write a function M-file that takes as input two matrices A and B, and

% as output produces the product by rows of the two matrices

function [C] = rowproduct(A,B)

[rA,cA] = size(A); % determine the dimension of A

[rB,cB] = size(B); % determine the dimension of B

if(cA == rB) % check the dimensions, columns of A

% must equal rows of B

C = zeros(rA,cB); % initialize C, a mxq matrix

for i = 1:rA

C(i,:) = A(i,:)\*B; % implement the formula

end % End for loop

else % if dimension not agreeable

disp('dimensions do not match')% throw an error message

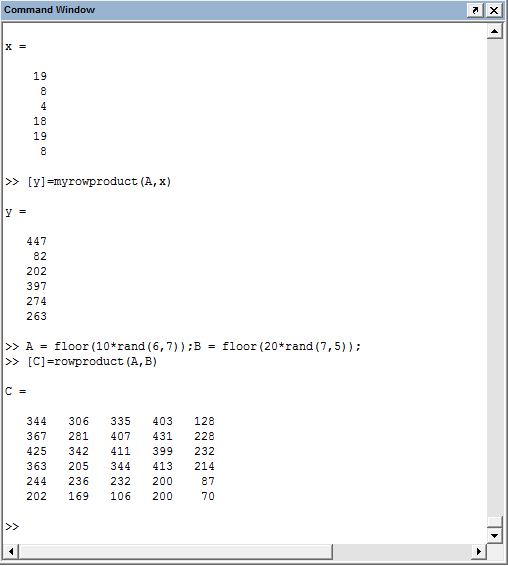
C = []; % return an empty matrix

end % end if

end % end function

%---------------------------------------------------------------------------------------------

**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**



columnproduct.m

%---------------------------------------------------------------------------------------------

% Write a function M-file that takes as input two matrices A and B,

% and as output produces the product by columns of the two matrix.

**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**

function [C] = rowproduct(A,B)

[rA,cA] = size(A); % determine the dimension of A

[rB,cB] = size(B); % determine the dimension of B

if(cA == rB) % check the dimensions, columns of A

% must equal rows of B

C = zeros(rA,cB); % initialize C, a mxq matrix

for i = 1:cB

C(:,i) = A\*B(:,i); % implement the formula

end % End for loop

else % if dimension not agreeable

disp('dimensions do not match')% throw an error message

C = []; % return an empty matrix

end % end if

end % end function

%---------------------------------------------------------------------------------------------

**MAT 343 MATLAB LAB 2 NAME: \_Hieu Pham\_\_\_\_**

