```
% This worksheet demonstrates the use of Matlab to illustrate the computational time
% needed to find the inverse of a matrix using two different methods:
% LU Decomposition and Naive Gaussian Elimination.
% Developed by Fabian Farelo and Autar Kaw
% December 2006
% University os South Florida
% kaw@eng.usf.edu
clear all
% Enter three values of size of matrix you want to compare
n = 10;
n2 = 200;
n3 = 2000;
disp(sprintf('Comparison of computational time to find inverse of a matrix using'))
disp(sprintf('Gaussian elimination and LU Decomposition'))
disp(sprintf('©2007 Fabian Farelo, Autar Kaw'))
disp(sprintf('University of South Florida'))
disp(sprintf('United States of America'))
disp(sprintf('kaw@eng.usf.edu'))

disp(sprintf('\n\nNOTE: This worksheet demonstrates the use of Matlab to illustrate'))
disp(sprintf('comparison of computational time to find inverse of a matrix using'))
disp(sprintf('Gaussian Elimination and LU DEcomposition'))
%--------------------------------------------------------------------------
disp(sprintf('\n\n**************************** Background↙
*****************************'))
%The number of FLOPS used in finding the inverse of a matrix by Naive Gaussian Elimination↙
are
%Forward Elimination: FE = sum(n*(n+2)-k*(2*n+2)+k^2, k = 1 .. n-1)
%Back substitution: BS = Sum(i, i = 1 .. n)
%Flops used in Naive Gaussian: NG = n*(FE + BS)
disp(sprintf('\n\nThe number of FLOPS used in finding the inverse of a matrix by Naive↙
Gaussian Elimination are'))
disp(sprintf('Forward Elimination: FE = sum(n*(n+2)-k*(2*n+2)+k^2, k = 1 .. n-1'))
disp(sprintf('Back substitution: BS = Sum(i, i = 1 .. n)'))
disp(sprintf('Flops used in Naive Gaussian: NG = n*(FE + BS)'))
FE = 0;
BS = 0;
FE2=0;
BS2=0;
FE3=0;
BS3=0;
for k=1:1:n-1
   FE =FE + n*(n+2)-k*(2*n+2)+k^2;
end

for k=1:1:n
    BS = BS + k;
end

for k=1:1:n2-1
   FE2 =FE2 + n2*(n2+2)-k*(2*n2+2)+k^2;
end
```

```
for k=1:1:n2
    BS2 = BS2 + k;
end

for k=1:1:n3-1
   FE3 =FE3 + n3*(n3+2)-k*(2*n3+2)+k^2;
end

for k=1:1:n3
    BS3 = BS3 + k;
end
NG = n*(FE + BS);
NG2 = n2*(FE2 + BS2);
NG3 = n3*(FE3 + BS3);

% The number of FLOPS used in finding the inverse of a matrix by LU Decomposition are
% Forward Elimination: FEL = sum(n*(n+2)-k*(2*n+2)+k^2, k = 1 .. n-1)
% Forward substitution: FSL = Sum(i, i = 1 .. n)
% Backward substitution: BSL = Sum(i, i = 1 .. n)
% Flops used in LU Decomposition: LU = FEL + n*(FSL +BSL)
disp(sprintf('\nThe number of FLOPS used in finding the inverse of a matrix by LU↙
Decomposition are'))
disp(sprintf('Forward Elimination: FEL = Sum(n*(n+2)-k*(2*n+2)+k^2, k = 1 .. n-1'))
disp(sprintf('Forward substitution: FSL = Sum(i, i = 1 .. n)'))
disp(sprintf('Backward substitution: BSL = Sum(i, i = 1 .. n)'))
disp(sprintf('Flops used in LU Decomposition: LU = FEL + n*(FSL +BSL)'))
FEL = 0;
FSL = 0;
BSL = 0;
FEL2 = 0;
FSL2 = 0;
BSL2 = 0;
FEL3 = 0;
FSL3 = 0;
BSL3 = 0;
for k=1:1:n-1
   FEL =FEL + n*(n+2)-k*(2*n+2)+k^2;
end

for k=1:1:n
    FSL = FSL + k;
    BSL = BSL + k;
end
LU = FEL + n*(FSL +BSL);



for k=1:1:n2-1
   FEL2 =FEL2 + n2*(n2+2)-k*(2*n2+2)+k^2;
end
```

```
for k=1:1:n2
    FSL2 = FSL2 + k;
    BSL2 = BSL2 + k;
end
LU2 = FEL2 + n*(FSL2 +BSL2);


for k=1:1:n3-1
   FEL3 =FEL3 + n3*(n3+2)-k*(2*n3+2)+k^2;
end

for k=1:1:n3
    FSL3 = FSL3 + k;
    BSL3 = BSL3 + k;
end
LU3 = FEL3 + n3*(FSL3 +BSL3);
disp(sprintf('\n\n*************************** Results ******************************'))
format short g
X=[NG,NG2,NG3];
Y=[LU,LU2,LU3];
Comp=[NG/LU,NG2/LU2,NG3/LU3];
Z=[n,n2,n3];
%-----------------------------------------------------------------------
fprintf(' Relative time of computation between finding the inverse using LU Decomposition↙
and')
fprintf('\n Gaussian Elimination for a matrix of size %d ',n)
fprintf('is %d ',Comp(1))
fprintf('\n\n Relative time of computation between finding the inverse using LU↙
Decomposition and')
fprintf('\n Gaussian Elimination for a matrix of size %d ',n2)
fprintf('is =%d ',Comp(2))
fprintf('\n\n Relative time of computation between finding the inverse using LU↙
Decomposition and')
fprintf('\n Gaussian Elimination for a matrix of size %d ',n3)
fprintf('is =%d ',Comp(3))


%-------Plotting the results ------------------
% Ratio of time taken by Gaussian Elimination to LU Decomposition
% as a function of size of matrix
plot(Z,Comp)
title(['Comparison of Computational Time to Find Inverse of a Matrix'])
xlabel('Size of Matrix')
ylabel('Ratio of Time, Gaussian Elimination/LU Decomposition')
hold on
hold off
```