HIMANSHU CHOUDHARY (200020059)

ASSIGNMENT-2.

PROBLEM NO. 1

PROBLEM STATEMENT

We are given the matrices A and B, (15x15) and (15x1) respectively. We have to find the matrix X such that

$$AX = B$$

where x is a matrix of size (15x1).

DESCRIPTION OF METHOD

We will use Gauss Elimination method to solve AX = B.

First we'll convert the matrix A to Row Echelon form in which A is an upper triangular matrix, and will apply the same set of operations on B. After that, we'll use Back substitution to find the matrix X.

Matrix x is our answer and we'll use plot function to plot X.

PSEUDO-CODE

initialize X and operations = 0.
Sort A and B
~~let c be from~~
let c = 1
while c ≤ n # Gauss elimination
    Sort A and B
    let r = m
    while r ≥ c+1
        A(r^th row) = A(r^th row) − (A(r,c) / A(c,c)) A(pivot row)
        B(r) = B(r) − (A(r,c) / A(c,c)) B(c)
        Increase operations to 2n+3
    decrease r by 1
    decrease c by 1
X(n) = B(n)/A(n,n)
let i = m−1
while i ≥ 1 # Back subs.
    sum = 0
    for j in [n : i+1]
        sum += $x_j A_{ij}$
    X(i) = (B(i) − sum)/A(i,i)

```
% Matrix A and B
A = load('A.txt');
B = load('B.txt');

dimensions = size(A); % Dimensions of A
m = dimensions(1); % No. of rows
n = dimensions(2); % No. of columns

X = gauss_elimination(A, B, m, n);
disp('X = ')
disp(X)
plot(X, 'LineWidth', 1.5)
```

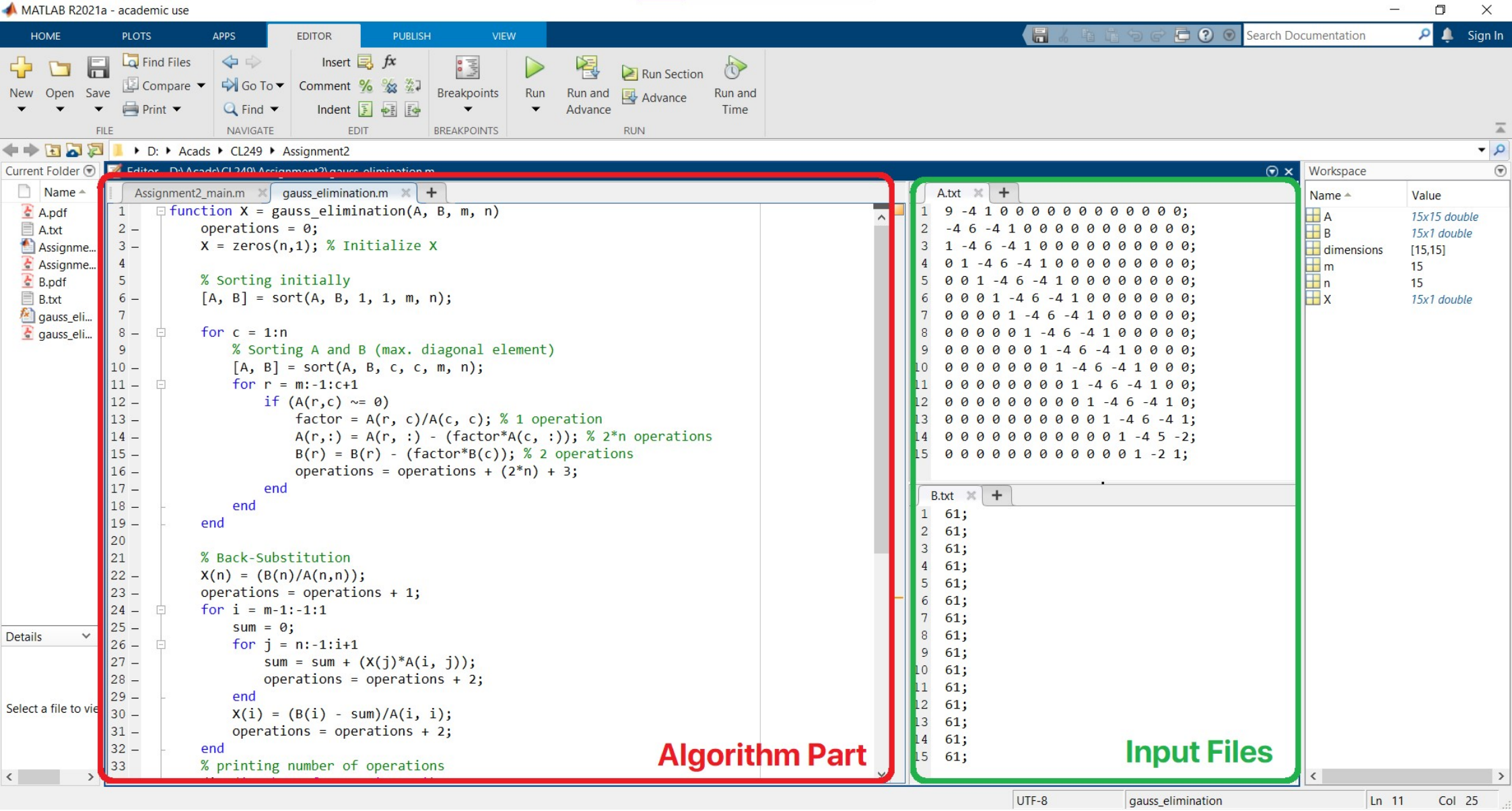
```
function X = gauss_elimination(A, B, m, n)
    operations = 0;
    X = zeros(n,1); % Initialize X

    % Sorting initially
    [A, B] = sort(A, B, 1, 1, m, n);

    for c = 1:n
        % Sorting A and B (max. diagonal element)
        [A, B] = sort(A, B, c, c, m, n);
        for r = m:-1:c+1
            if (A(r,c) ~= 0)
                factor = A(r, c)/A(c, c);  % 1 operation
                A(r,:) = A(r, :) - (factor*A(c, :));  % 2*n operations
                B(r) = B(r) - (factor*B(c));  % 2 operations
                operations = operations + (2*n) + 3;
            end
        end
    end

    % Back-Substitution
    X(n) = (B(n)/A(n,n));
    operations = operations + 1;
    for i = m-1:-1:1
        sum = 0;
        for j = n:-1:i+1
            sum = sum + (X(j)*A(i, j));
            operations = operations + 2;
        end
        X(i) = (B(i) - sum)/A(i, i);
        operations = operations + 2;
    end
    % printing number of operations
    disp('Number of Operations =')
    disp(operations)

end

% Sorting Function
function [mat1, mat2] = sort(A, B, rs, cs, m, n)
    for s = rs:m-1
        for r = rs:m-1
            if abs(A(r, cs)) < abs(A(r+1, cs))
                temp1 = A(r, :);
                A(r, :) = A(r+1, :);
                A(r+1, :) = temp1;
                temp2 = B(r);
                B(r) = B(r+1);
                B(r+1) = temp2;
            end
        end
    end
    mat1 = A;
    mat2 = B;
end
```

A.txt

```
9 -4 1 0 0 0 0 0 0 0 0 0 0 0 0;
-4 6 -4 1 0 0 0 0 0 0 0 0 0 0 0;
1 -4 6 -4 1 0 0 0 0 0 0 0 0 0 0;
0 1 -4 6 -4 1 0 0 0 0 0 0 0 0 0;
0 0 1 -4 6 -4 1 0 0 0 0 0 0 0 0;
0 0 0 1 -4 6 -4 1 0 0 0 0 0 0 0;
0 0 0 0 1 -4 6 -4 1 0 0 0 0 0 0;
0 0 0 0 0 1 -4 6 -4 1 0 0 0 0 0;
0 0 0 0 0 0 1 -4 6 -4 1 0 0 0 0;
0 0 0 0 0 0 0 1 -4 6 -4 1 0 0 0;
0 0 0 0 0 0 0 0 1 -4 6 -4 1 0 0;
0 0 0 0 0 0 0 0 0 1 -4 6 -4 1 0;
0 0 0 0 0 0 0 0 0 0 1 -4 6 -4 1;
0 0 0 0 0 0 0 0 0 0 0 1 -4 5 -2;
0 0 0 0 0 0 0 0 0 0 0 0 1 -2 1;
```

B.txt

```
61;
61;
61;
61;
61;
61;
61;
61;
61;
61;
61;
61;
61;
61;
61;
```


MATLAB R2021a - academic use
HOME
PLOTS
APPS
EDITOR
PUBLISH
VIEW
Search Documentation
Sign In
New
Open
Save
Find Files
Compare
Print
FILE
Go To
Find
NAVIGATE
Insert
Comment
Indent
EDIT
Breakpoints
BREAKPOINTS
Run
Run and Advance
Run Section
Advance
Run and Time
RUN
D: ▸ Acads ▸ CL249 ▸ Assignment2
Current Folder
Name
A.pdf
A.txt
Assignme...
Assignme...
B.pdf
B.txt
gauss_eli...
gauss_eli...
Details
Select a file to vie
Assignment2_main.m
gauss_elimination.m
function X = gauss_elimination(A, B, m, n)
operations = 0;
X = zeros(n,1); % Initialize X
% Sorting initially
[A, B] = sort(A, B, 1, 1, m, n);
for c = 1:n
% Sorting A and B (max. diagonal element)
[A, B] = sort(A, B, c, c, m, n);
for r = m:-1:c+1
if (A(r,c) ~= 0)
factor = A(r, c)/A(c, c); % 1 operation
A(r,:) = A(r, :) - (factor*A(c, :)); % 2*n operations
B(r) = B(r) - (factor*B(c)); % 2 operations
operations = operations + (2*n) + 3;
end
end
end
% Back-Substitution
X(n) = (B(n)/A(n,n));
operations = operations + 1;
for i = m-1:-1:1
sum = 0;
for j = n:-1:i+1
sum = sum + (X(j)*A(i, j));
operations = operations + 2;
end
X(i) = (B(i) - sum)/A(i, i);
operations = operations + 2;
end
% printing number of operations
Algorithm Part
A.txt
9 -4 1 0 0 0 0 0 0 0 0 0 0 0 0;
-4 6 -4 1 0 0 0 0 0 0 0 0 0 0 0;
1 -4 6 -4 1 0 0 0 0 0 0 0 0 0 0;
0 1 -4 6 -4 1 0 0 0 0 0 0 0 0 0;
0 0 1 -4 6 -4 1 0 0 0 0 0 0 0 0;
0 0 0 1 -4 6 -4 1 0 0 0 0 0 0 0;
0 0 0 0 1 -4 6 -4 1 0 0 0 0 0 0;
0 0 0 0 0 1 -4 6 -4 1 0 0 0 0 0;
0 0 0 0 0 0 1 -4 6 -4 1 0 0 0 0;
0 0 0 0 0 0 0 1 -4 6 -4 1 0 0 0;
0 0 0 0 0 0 0 0 1 -4 6 -4 1 0 0;
0 0 0 0 0 0 0 0 0 1 -4 6 -4 1 0;
0 0 0 0 0 0 0 0 0 0 1 -4 6 -4 1;
0 0 0 0 0 0 0 0 0 0 0 1 -4 5 -2;
0 0 0 0 0 0 0 0 0 0 0 0 1 -2 1;
B.txt
61;
61;
61;
61;
61;
61;
61;
61;
61;
61;
61;
61;
61;
61;
61;
Input Files
Workspace
Name
Value
A
15x15 double
B
15x1 double
dimensions
[15,15]
m
15
n
15
X
15x1 double
UTF-8
gauss_elimination
Ln 11
Col 25

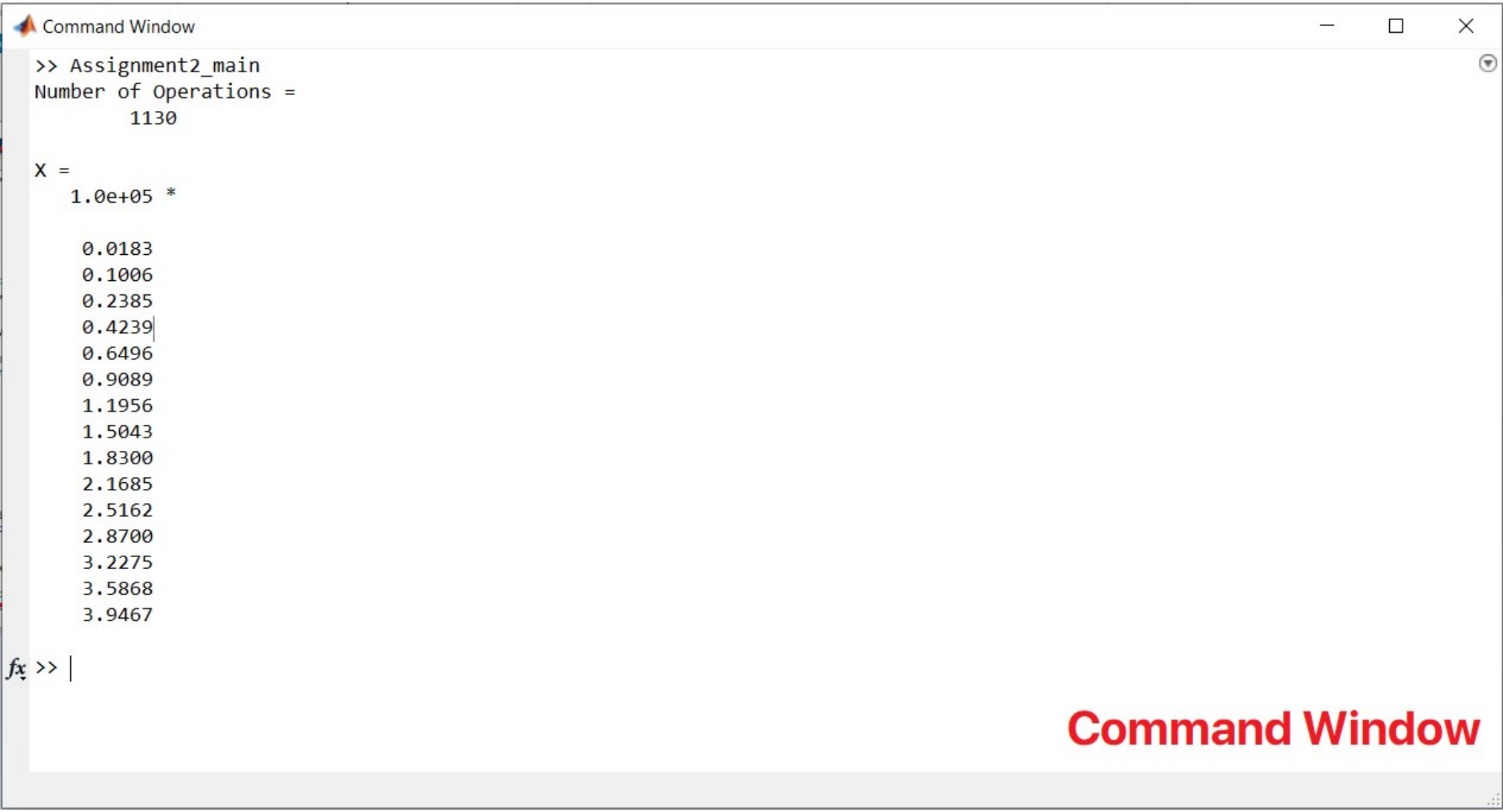
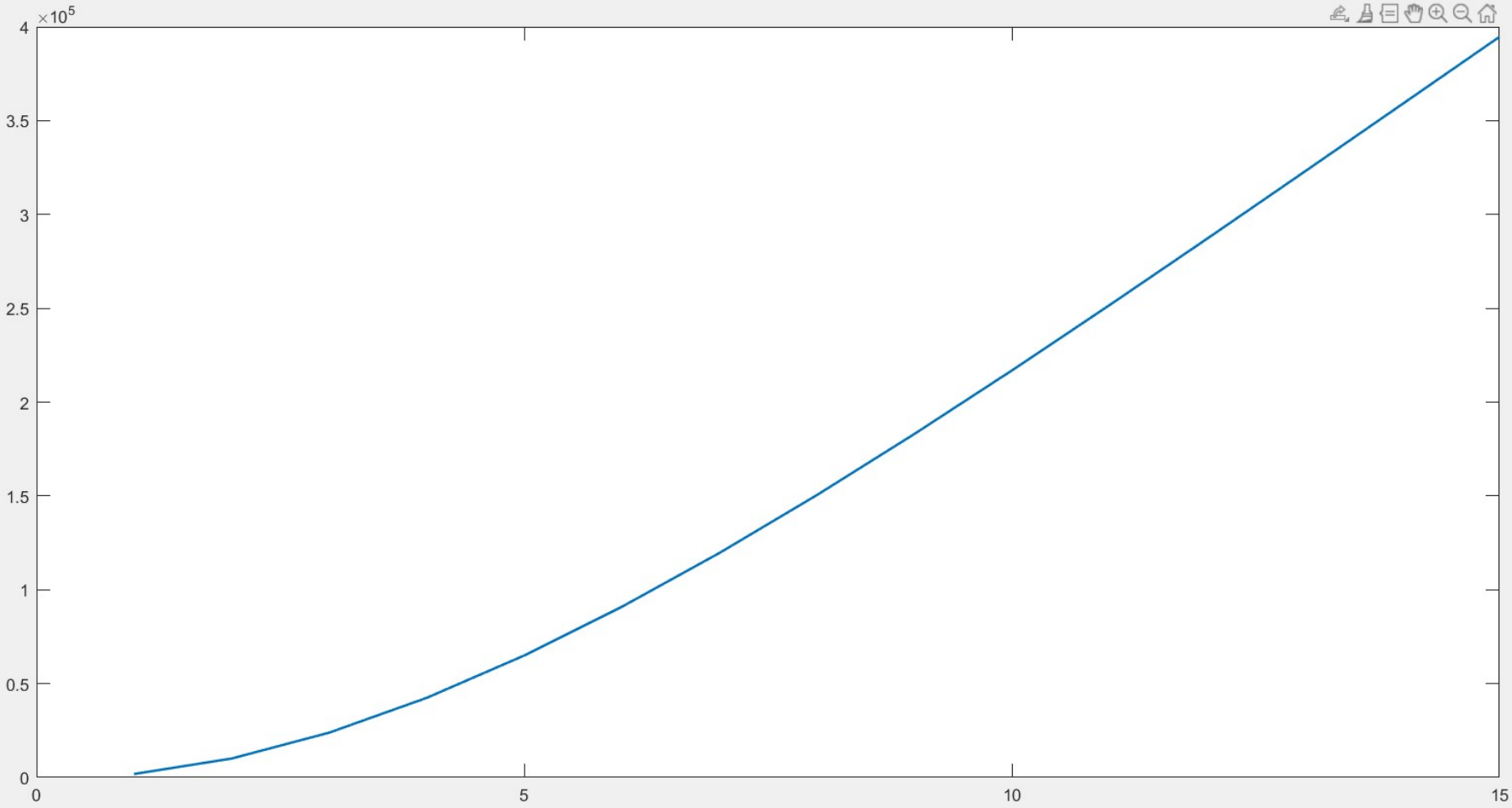

Command Window
>> Assignment2_main
Number of Operations =
1130
X =
1.0e+05 *
0.0183
0.1006
0.2385
0.4239
0.6496
0.9089
1.1956
1.5043
1.8300
2.1685
2.5162
2.8700
3.2275
3.5868
3.9467
>>
Command Window

Figure 1

File Edit View Insert Tools Desktop Window Help



Plot of X