Matrix Computations MA423 Lab 01

Name**: Naman Goyal**
Roll No: **180123029**

Ques.**1(a)**

- To generate **Wilkinson's** Matrix without any for loops. Let's say the matrix is W.

```
function WilkinsonMat = Wilkinson(n)
    WilkinsonMat = 2*eye(n) - tril(ones(n));
    WilkinsonMat(:,n) = ones(1, n);
end
```

- Output for **N = 5 :**

```
Input Wilkinson no:
5
     1     0     0     0     1
    -1     1     0     0     1
    -1    -1     1     0     1
    -1    -1    -1     1     1
    -1    -1    -1    -1     1
```

1(b)

- To generate **Hamiltonian** Matrix:

```
function HamiltonMat = Hamiltonian(n)
    H_11 = randn(floor(n));
    H_12_lower = tril(randn(n));
    H_12 = H_12_lower + H_12_lower';
    H_21_lower = tril(randn(n));
    H_21 = H_21_lower + H_21_lower';
    HamiltonMat = [H_11 H_12; H_21 -H_11'];
end
```

- H_12_lower is a lower triangular matrix, hence H_12_lower' would be an upper triangular matrix, with the same entries as the lower triangular matrix, hence H_12 = H_12_lower + H_12_lower' is such that H_12 = H_12' similarly for H21 it would be the same.

- Output for **N = 4 :**

```
Question 1 (b)
Input Hamiltonian no:
4
   -0.3013   -0.7118   -0.2340   -0.8327    3.5704    2.1258   -0.8185   -1.1772
    1.0980   -1.1056   -1.8419    0.0643    2.1258    1.3545    0.4275   -0.7226
    0.9187    0.4106    1.6348    2.4744   -0.8185    0.4275   -0.5115   -0.4085
   -1.0304    1.5867    0.7065    0.6270   -1.1772   -0.7226   -0.4085    2.4510
    0.4864    0.1469    0.4874    1.9923    0.3013   -1.0980   -0.9187    1.0304
    0.1469   -1.1085    0.0699   -0.7802    0.7118    1.1056   -0.4106   -1.5867
    0.4874    0.0699   -1.4136    0.6361    0.2340    1.8419   -1.6348   -0.7065
    1.9923   -0.7802    0.6361   -0.9558    0.8327   -0.0643   -2.4744   -0.6270
```

Ques.2

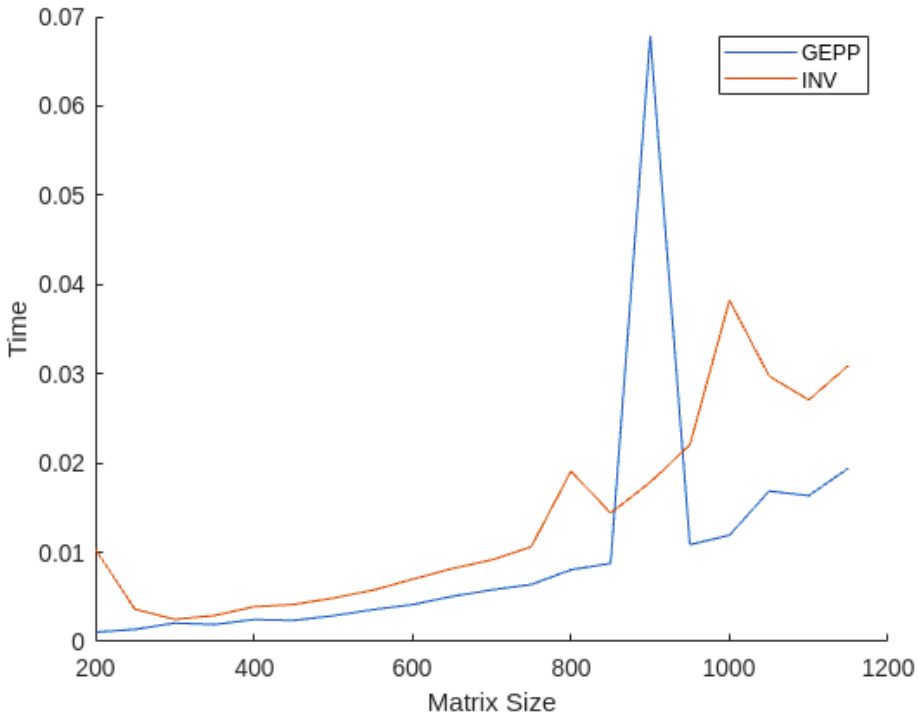
- Value of m = 1000 & k = 5 is taken. The output :
- Where **s** = Sum of 1/n for n = 1,2,3..1000, **scf** = Sum of 1/n upto 5-digit arithmetic, **scb** = Sum of 1/n in 5-digit arithmetic in reverse order.

```
Question 2

s : 7.48547
scf : 7.48460
scb : 7.48490
(c) is more closer to (a) than (b)
>>
```

Ques.3

- The graph is plotted for **Time vs Matrix** size.



Ques.4 (a)

```
Question 4 a
Input size of Upper Triangular Matrix(n):
3
For U =
   -0.0703   -0.4012   -1.2241
         0   -0.0975   -0.1943
         0         0   -0.0604

For b =
   -0.2416
    1.6786
    1.0342

Solution x using Column Oriented Back substitution =

  205.0127
   16.8838
  -17.1111
```

4 (b)

```
Input size of Lower Triangular Matrix(n):
3
For L =
    0.9907         0         0
    0.4314    0.8741         0
   -0.0658    0.7450    0.3328

For b =
    0.8366
   -0.0569
    2.3304

Solution x  using Row Oriented Forward substitution =

    0.8444
   -0.4818
    8.2475
```

Ques.5

- The Matrix A and it's **LU factorization**:

```
Input size of matrix A:
3
Matrix A
A =
   -0.2374    0.1261   -1.0443
   -1.6389   -0.3334    0.9893
   -0.2001    0.8681    0.3450


L =
    1.0000         0         0
    6.9047    1.0000         0
    0.8429   -0.6325    1.0000

U =
   -0.2374    0.1261   -1.0443
         0   -1.2044    8.1998
         0         0    6.4118
```

- For verification purposes **L*U should be the same as A.**

```
L*U =
   -0.2374    0.1261   -1.0443
   -1.6389   -0.3334    0.9893
   -0.2001    0.8681    0.3450
```
