

```
1 %Here we have created a function LSSNEqn
2 function [X]=LSSNEqn(A,b)
3 temp1=A.';    %temp1 is transpose of A
4
5 C=temp1*A;
6
7 D=chol(C);    %chol function is basically do a cholskey decomposition
8 temp3=D.';
9
10 %Solving the system of linear equation using backward and forward
11 %substitution
12
13 Y=FdSubs(temp3,temp1*b);
14 X=BdSubs(D,Y);
15
16 end
```

```
1 %Gram-Schmidt Method for QR decomposition
2
3 function [Q,R] = GramSelf(X)
4
5
6     [m,n] = size(X);
7     Q = zeros(m,n);
8     R = zeros(n,n);
9     Q(1:m,1) = X(1:m,1);
10    R(1,1) = norm(Q(1:m,1));
11    Q(1:m,1) = Q(1:m,1)/R(1,1);
12    for k = 2:n
13        Q(1:m,k) = X(1:m,k);
14
15        R(1:k-1,k) = Q(1:m,k-1)'*Q(1:m,k);
16        Q(1:m,k) = Q(1:m,k) - Q(1:m,1:k-1)*R(1:k-1,k);
17
18        R(k,k) = norm(Q(1:m,k));
19        Q(:,k) = Q(:,k)/R(k,k);
20    end
21
22 end
23
```

```
1
2 %Householder's Method for QR decomposition
3 function [Q,R] = HouseSelf(A)
4
5 [m,n] = size(A);
6 Q = eye(m);
7 R = A;
8
9 for i = 1:n
10 norm_val = norm(R(i:n,i));
11 s = -sign(R(i,i));
12 u1 = R(i,i) - s*norm_val;
13 w = R(i:n,i)/u1;
14 w(1) = 1;
15 val = -s*u1/norm_val;
16
17 R(i:n,:) = R(i:n,:)-(val*w)*(w.'*R(i:n,:));
18 Q(:,i:n) = Q(:,i:n)-(Q(:,i:n)*w)*(val*w).';
19
20 end
21
```

```
1 %Doing QR decomposition using GramSelf method
2
3 prompt = "Enter the matrix which you want to decompose \nX= ";
4 X = input(prompt);
5 disp("X=");
6 disp(X);
7 [Q,R]=GramSelf(X);
8 disp('Q=');
9 disp(Q);
10 disp('R=');
11 disp(R);
12
```

```
>> A3P2
```

```
Enter the matrix which you want to decompose
```

```
X= [1,1,0;1,0,1;0,1,1]
```

```
X=
```

1	1	0
1	0	1
0	1	1

```
Q=
```

0.7071	0.4082	-0.3818
0.7071	-0.4082	0.7361
0	0.8165	0.5589

```
R=
```

1.4142	0.7071	0.4082
0	1.2247	0.4082
0	0	1.1927

```
>>
```

```
1 %Doing QR decomposition using Householder's method
2
3 prompt = "Enter the matrix which you want to decompose \nX= ";
4 X = input(prompt);
5 disp("X=");
6 disp(X);
7 [Q,R] = HouseSelf(X);
8 disp('Q');
9 disp(Q);
10 disp('R');
11 disp(R);
12
```

```
>> A3P3
```

```
Enter the matrix which you want to decompose
```

```
X= [-1,-1,1;1,3,3;-1,-1,5]
```

```
X=
```

```
   -1   -1    1  
    1    3    3  
   -1   -1    5
```

```
Q
```

```
  -0.5774   -0.4082    0.7071  
   0.5774   -0.8165   -0.0000  
  -0.5774   -0.4082   -0.7071
```

```
R
```

```
   1.7321    2.8868   -1.7321  
  -0.0000   -1.6330   -4.8990  
   0.0000    0.0000   -2.8284
```

```
>>
```

```
1 %Basically we have already created a function LSSNEqn and now using it we are ↵
trying
2 %to solve a system of linear equation by least square method
3
4 prompt = "Enter the matrix A \nA=";
5 A = input(prompt);
6 prompt = "Enter the matrix b \nb=";
7 b = input(prompt);
8
9 X=LSSNEqn(A,b);
10
11 disp("A=");
12 disp(A);
13 disp("b=");
14 disp(b);
15 disp("Solution matrix x by least square method is ");
16 disp('X=');
17 disp(X);
18 disp("Solution matrix by A\b is ");
19 disp('X=');
20 disp(A\b);
```



```
>> A3P4
```

```
Enter the matrix A
```

```
A=[1,1;2,1;3,1]
```

```
Enter the matrix b
```

```
b=[1;2;2]
```

```
A=
```

```
    1    1  
    2    1  
    3    1
```

```
b=
```

```
    1  
    2  
    2
```

```
Solution matrix x by least square method is
```

```
X=
```

```
    0.5000  
    0.6667
```

```
Solution matrix by A\b is
```

```
X=
```

```
    0.5000  
    0.6667
```

```
>>
```