

# test

## Contents

MGS and CGS	1
MGS function	1
Test Code	2

## MGS and CGS

```
%15307130224

%%CGS function

%function [Q,R] = MGS(A)

%[m, n] = size(A);

%Q = zeros(m,n);
%R = zeros(n);

%for k = 1:n
% R(k, k) = normest(A(:,k));
% Q(:,k) = A(:,k)/R(k,k);
%
% for j = k+1:n
%   R(k,j) = dot(A(:,j),Q(:,k));
%   A(:,j) = A(:,j) - R(k,j)*Q(:,k);
% end
%end
```

## MGS function

```
%function [Q,R] = MGS(A)

%[m, n] = size(A);
```

```

%Q = zeros(m,n);
%R = zeros(n);

%for k = 1:n
% R(k, k) = normest(A(:,k));
% Q(:,k) = A(:,k)/R(k,k);

% for j = k+1:n
%   R(k,j) = dot(A(:,j),Q(:,k));
%   A(:,j) = A(:,j) - R(k,j)*Q(:,k);
% end
%end

```

## Test Code

```

SIZE = 100
COUNT = 30
error_cgs = zeros(1,COUNT)
error_mgs = zeros(1,COUNT)
range = 1:1:COUNT;
for i = 1:COUNT
    A = randn(SIZE);
    [q,r] = CGS(A);
    error_cgs_matrix = eye(SIZE) - q'*q;
    error_cgs(1,i) = norm(error_cgs_matrix,p="fro");
    [Q,R] = MGS(A);
    error_mgs_matrix = eye(SIZE) - Q'*Q;
    error_mgs(1,i) = norm(error_mgs_matrix, p="fro");
end

semilogy(range,error_cgs,'o;CGS;', range,error_mgs, '*;
MGS;')

```

```

SIZE = 100
COUNT = 30
error_cgs =
Columns 1 through 20:
    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
           0    0    0    0    0    0
Columns 21 through 30:
    0    0    0    0    0    0    0    0    0    0
error_mgs =

```

Columns 1 through 20:

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
0 0 0 0 0 0 0 0

Columns 21 through 30:

0 0 0 0 0 0 0 0 0 0

