

slower.m

```
function A = slower(n)
% SLOWER(N) pinakas A, NxN me A(I,J) = 1/(I+J)

for i=1:n
    for j=1:n
        A(i,j) = 1 / (i+j);
    end
end
end
```

slow.m

```
function A = slow(n)
% SLOW(N) pinakas A, NxN me A(I,J) = 1/(I+J)

A = zeros(n);
for i=1:n
    for j=1:n
        A(i,j) = 1 / (i+j);
    end
end
end
```

faster.m

```
function A = faster(n)
% FASTEST(N) pinakas A, NxN me A(I,J) = 1/(I+J)

A = zeros(n);
tmp = 1:n;
for i=1:n
    A(i,:) = 1./(tmp + i);
end
end
```

fastest.m

```
function A = fastest(n)
% FASTEST(N) pinakas A, NxN me A(I,J) = 1/(I+J)

A = zeros(n);
tmp = (1:n)';
for i=1:n
    A(:,i) = 1./(tmp + i);
end
end
```

» n = 1000;

» tic; slower(n); toc

» tic; slow(n); toc

» tic; fastest(n); toc

» clear all

sxima2.m

```
x = 1:10;  
y = 10.^x;  
semilogy(x,y,'o-');  
title('x vs. log_{10}y')  
xlabel('x');  
ylabel('y');
```

Νόρμες Διανυσμάτων

» `x = rand(10,1)`

» `norm(x)`

» `norm(x,1)`

» `norm(x,inf)`

$$\|x\|_{\infty} = \max_{1 \leq i \leq n} |x_i|$$

Νόρμες Πινάκων

» `A = [1:3; 4:6; 7:9]`

» `norm(A)`

φασματική νόρμα

» `sqrt(max(abs(eig(A'*A))))`

» `norm(A,1)`

νόρμα αθροίσμ. στηλών

» `norm(A,inf)`

νόρμα αθροίσματος γραμμών

Δείκτης κατάστασης πίνακα A:



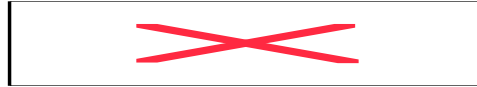
» `cond(A)` Δείκτης κατάστασης ως προς την $\|\cdot\|_2$

» `cond(A,1)` Δείκτης κατάστασης ως προς την $\|\cdot\|_1$

» `cond(A,inf)` Δείκτης κατάστασης ως προς την $\|\cdot\|_{\infty}$

Πίνακες Hilbert (κακής κατάστασης)

» `B = hilb(4)`



» `cond(B)`

» `inv(B)`

Υπολογίζει τον αντίστροφο του B λύνοντας το συστ. $BX=I$

» `invhilb(4)`

Υπολογίζει τον ακριβή αντίστροφο του πίνακα Hilbert

Επίλυση Γραμμικών συστημάτων

» `n=500;`

» `A = rand(n);`

» `b = rand(n,1);`

» `tic; x=A\b; t1 = toc;` --> Απαλοιφή Gauss

» `tic; y=inv(A)*b; t2 = toc;`

» `norm(x-y)`

» `t1`

» `t2`