

Σπαϊής Αλέξανδρος
Userid: alspais
AEM : 305

Κώδικας MatLab

lu run

```
function [xronos error] = lu_run(nx , ny, c1, c2, c)

%% 1. Δημιουργεί τα a, f
[a, f] = elliptic(nx, ny, c1, c2, c);

%% 2. Σχεδιάζει τη δομή του πίνακα αν nx, ny =< 5
nonzero(a);

%% 3. Λύνει το σύστημα με την lu
tic;
[l u] = lu(a);

%% 4.Μετρά το χρόνο εκτέλεσης και το σφάλμα προσέγγισης της λύσης
y = 1 \ f';
x_p = u \ y;
xr=toc;
sfalma=(a*x_p - f');

%% 5. Εμφανίζει με κατάλληλα σχόλια το χρόνο εκτέλεσης και το σφάλμα
xronos=xr;
error=norm(sfalma)

%% 6. Σχεδιάζει τη λύση
plotsolution(nx,ny,x_p);
```

luinc run

```
function [xronos error] = luinc_run(nx , ny, c1, c2, c)

%% 1. Δημιουργεί τα a, f
[a, f] = elliptic(nx, ny, c1, c2, c);

%% 2. Σχεδιάζει τη δομή του πίνακα αν nx, ny =< 5
nonzero(a);

%% 3. Λύνει το σύστημα με την lu
tic;
sa = sparse(a);
[l u] = lu(sa);

%% 4. Μετρά το χρόνο εκτέλεσης και το σφάλμα προσέγγισης της λύσης
y = 1 \ f';
x_p = u \ y;
xr=toc;
sfalma=(sa*x_p - f');

%% 5. Εμφανίζει με κατάλληλα σχόλια το χρόνο εκτέλεσης και το σφάλμα
xronos=xr;
error=norm(sfalma)

%% 6. Σχεδιάζει τη λύση
```

```
plotsolution(nx,ny,x_p);
```

Jacobi run

```
function [xronos error] = jacobi_run(nx , ny, c1, c2, c)

%% 1. Δημιουργεί τα a, f
[a, f] = elliptic(nx, ny, c1, c2, c);

%% 2. Σχεδιάζει τη δομή του πίνακα αν nx, ny =< 5
nonzero(a);

%% 3. Λύνει το σύστημα με την Jacobi
tic;
x0 = a.*[0]';
eps = .5e-5;
maxiter = (nx * ny);
[sol, err, it] = jacobi(a , x0, f', eps, maxiter);

%% 4. Μετρά το χρόνο εκτέλεσης και το σφάλμα προσέγγισης της λύσης
xr=toc;

%% 5. Εμφανίζει με κατάλληλα σχόλια το χρόνο εκτέλεσης και το σφάλμα
xronos=xr;
error=norm(err);

%% 6. Σχεδιάζει τη λύση
plotsolution(nx, ny, sol);
```

Gauss - Seidel

```
function [xronos error] = gs_run(nx , ny, c1, c2, c)

%% 1. Δημιουργεί τα a, f
[a, f] = elliptic(nx, ny, c1, c2, c);

%% 2. Σχεδιάζει τη δομή του πίνακα αν nx, ny =< 5
nonzero(a);

%% 3. Λύνει το σύστημα με την gs
tic;
x0 = a*0;
eps = .5e-5
maxiter = nx * ny;
[sol, err, it] = gs(a , x0, f', eps, maxiter)

%% 4. Μετρά το χρόνο εκτέλεσης και το σφάλμα προσέγγισης της λύσης
xr=toc;

%% 5. Εμφανίζει με κατάλληλα σχόλια το χρόνο εκτέλεσης και το σφάλμα
xronos=xr;
error=norm(err);

%% 6. Σχεδιάζει τη λύση
plotsolution(nx,ny,sol);
```

Conjugate Gradients (Συζυγών Κλίσεων)

```
function [xronos error] = cg_run(nx , ny, c1, c2, c)

%% 1. Δημιουργεί τα a, f
[a, f] = elliptic(nx, ny, c1, c2, c);

%% 2. Σχεδιάζει τη δομή του πίνακα αν nx, ny =< 5
nonzero(a);

%% 3. Λύνει το σύστημα με την cg
tic;
eps = .5e-5;
maxiter = nx * ny;
[sol, err, it] = cg(a, f', eps, maxiter)

%% 4. Μετρά το χρόνο εκτέλεσης και το σφάλμα προσέγγισης της λύσης
xr=toc;
sfalma=(a*sol - f');

%% 5. Εμφανίζει με κατάλληλα σχόλια το χρόνο εκτέλεσης και το σφάλμα
xronos=xr;
error=norm(sfalma)

%% 6. Σχεδιάζει τη λύση
plotsolution(nx,ny,sol);
```