

```
% Dopo aver scaricato nella directory corrente il file lab2.m da Moodle
% diamo il comando
help lab2
% che restituisce
function B=lab2(A,f)
```

Esercizio:

cosa fa questo algoritmo?
scrivere l'help

```
% Notiamo che l'help di una function coincide con il gruppo di righe
% commentato appena sotto all'intestazione del file
```

```
edit lab2.m
```

```
% Esaminiamo l'algoritmo nel file lab2.m ed inseriamo via via i
```

```
% commenti
```

```
% Per comprendere la riga
```

```
% A(m:N,m:N)=A(m:N,m:N)-A(m:N,m-1)*A(m-1,m:N);
```

```
% eseguiamo un prodotto di vettori riga per vettori colonna
```

```
w=1:3
```

```
w =
```

```
1    2    3
```

```
v=(4:6)'
```

```
v =
```

```
4
```

```
5
```

```
6
```

```
v*w
```

```
ans =
```

```
4    8   12
```

```
5   10   15
```

```
6   12   18
```

```
% In questo modo il risultato è una matrice in cui ogni riga è
```

```
% multipla di w (ed i fattori moltiplicativi sono gli elementi di v)
```

```
w*v
```

```
ans =
```

```
32
```

```
% In questo ordine invece calcolo il prodotto scalare fra v e w
```

```
% lab2.m esegue la decomposizione A=LU della matrice in ingresso e
```

```
% restituisce gli elementi di U e i moltiplicatori in un'unica
```

```
% matrice B
% Per commenti più dettagliati si veda il file lunopivot.m su Moodle
% Ad esempio creiamo una matrice 4x4
A = rand(4)
A =
    0.7577    0.1712    0.0462    0.3171
    0.7431    0.7060    0.0971    0.9502
    0.3922    0.0318    0.8235    0.0344
    0.6555    0.2769    0.6948    0.4387
B = lunopivot(A)
B =
    0.7577    0.1712    0.0462    0.3171
    0.9807    0.5382    0.0519    0.6392
    0.5176   -0.1055    0.8050   -0.0623
    0.8650    0.2394    0.7981    0.0611
% Per estrarre dalla B i fattori L ed U si possono usare i seguenti
% comandi (si consulti 'doc triu', 'doc tril' e 'doc eye')
U=triu(B)
U =
    0.7577    0.1712    0.0462    0.3171
         0    0.5382    0.0519    0.6392
         0         0    0.8050   -0.0623
         0         0         0    0.0611
>> L=tril(B,-1)+eye(4)
L =
    1.0000         0         0         0
    0.9807    1.0000         0         0
    0.5176   -0.1055    1.0000         0
    0.8650    0.2394    0.7981    1.0000
% Verifichiamo che la decomposizione sia corretta
A-L*U
ans =
         0         0         0         0
         0         0         0         0
         0         0         0         0
         0         0         0         0
% In generale potranno esserci delle lievi differenze fra A e L*U
% dovute agli errori di arrotondamento durante l'algoritmo

% Il prossimo obiettivo è programmare una
```

```
% function x = lusolve(B,b)
% che risolva un sistema lineare  $A*x=b$  usando la decomposizione
% resitutita da  $B=lunopivot(A)$  ed ovviamente il termine noto b
%
% Si veda la il file lusolve.m su Moodle
```

```
% Proviamo a risolvere un sistema lineare
% Scegliamo una matrice
```

```
A=rand(3)
```

```
A =
```

```
    0.8147    0.9134    0.2785
    0.9058    0.6324    0.5469
    0.1270    0.0975    0.9575
```

```
% Costruiamo b tale che la soluzione esatta sia [1;1;1]
```

```
b=A*ones(3,1)
```

```
b =
```

```
    2.0066
    2.0850
    1.1820
```

```
B=lunopivot(A)
```

```
B =
```

```
    0.8147    0.9134    0.2785
    1.1118   -0.3831    0.2373
    0.1559    0.1170    0.8863
```

```
lusolve(B,b)
```

```
ans =
```

```
    1.0000
    1.0000
    1.0000
```

```
lusolve(B,b)-ones(3,1)
```

```
ans =
```

```
    1.0e-14 *
   -0.1554
    0.1554
         0
```

```
% La funzione ludecomp.m (su Moodle!) calcola la decomposizione  $PA=LU$ 
% con pivoting specificato dal secondo argomento. Si veda 'help ludecomp'
% e si consulti il sorgente per vedere come è realizzato lo scambio
% Si noti che
```

```
v=0:2:6
v =
    0    2    4    6
v([1,3,2,4])
ans =
    0    4    2    6
% Calcoliamo la decomposizione con pivoting della matrice 3x3
[B,p]=ludecomp(A,2)
B =
    0.9058    0.6324    0.5469
    0.8995    0.3446   -0.2134
    0.1402    0.0258    0.8863
p =
     2     1     3
% Sono state scambiate la prima e la seconda riga
% quindi me ne devo ricordare quando risolvo!
lusolve(B, b(p)) % b(p) scambia gli elementi!
ans =
    1.0000
    1.0000
    1.0000
lusolve(B, b(p)) - ones(3,1)
ans =
    1.0e-15 *
   -0.6661
    0.4441
     0
% Usando il pivoting l'errore è diminuito!
```