

INIZIO FUNCTION

PRDGRAM<- function(y1,n1,m1) {

VALORI DEL PARAMETRO ak

a0=c(); k=0; a0=0;

for(t in 1:n1){a0=a0+y1[t]*cos(2*pi*t*k/n1)}

a0

a0=a0*2/n1;a0=a0/2

a0

a=c();a[1:m1]=0;

for(k in 1:m1) {

for(t in 1:n1){

a[k]=a[k]+y1[t]*cos(2*pi*t*k/n1)}

a=2*a/n1

vALORI DEL PARAMETRO bk

b=c();b[1:m1]=0;b0=0;k=0

for(k in 1:m1) {

for(t in 1:n1){

b[k]=b[k]+y1[t]*sin(2*pi*t*k/n1)}

a <- as.vector(a)

for(i in 1:m1){

if (abs(a[i]) < 1e-10) a[i]=0 else a[i]=a[i]}

a

for(i in 1:m1){

if (abs(b[i]) < 1e-10) b[i]=0 else b[i]=b[i]}

b=2*b/n1

b

AMPIEZZE

#ro[1:m1]=0

ro <- sqrt(a^2 +b^2)

for(i in 1:m1){

if (abs(ro[i]) < 1e-10) ro[i]=0 else ro[i]=ro[i]}

CALCOLO DELLA FASE DI OGNI ARMONICA

RIPORTANDO IL VALORE AL QUADRANTE GIUSTO

f2=c()

f2[1:m1]=0

for(i in 1:m1){

f2[i] <- abs(a[i]/b[i])

f2[i] <- atan(f2[i])*180/pi}

f2 =as.vector(f2)

f2

```

#f2[1:m1]=0 un f2[1:m1] di troppo!
phi <- c()
for(i in 1:m1){
# f2 <- abs(a[i]/b[i]);
# f2 <- atan(f2)*180/pi;
if(b[i]>0 & a[i]>0) phi[i] = f2[i];
if(b[i]<0 & a[i]>0) phi[i] = 180-f2[i];
if(b[i]<0 & a[i]<0) phi[i] = 180+f2[i];
if(b[i]>0 & a[i]<0) phi[i] = 360-f2[i];
if(b[i]==0 & a[i]==0) phi[i] = 0;
if((b[i]<0 & b[i]>0) | a[i]==0) phi[i]=0;
if(b[i]==0 & a[i]>0) phi[i]=90;
if(b[i]==0 & a[i]<0) phi[i]=360-90
}

```

PHI FASE ARMONICHE

```

phi=as.vector(phi)
phi
param_a <-a
param_b <-b
ampiezza <- ro
fase <- phi

```

```

# Qui, al termine della function si pone il valore di un'unica
# variabile che esce o, se escono più variabili, si usa
# un data.frame: data=data.frame(x1,x2,...).
# Ogni chiamata alla function permette di includere l'unica
# variabile o i data nel nome della chiamata:
# es. periodxx=nome.function(x1,x2,...)

```

```

data <-data.frame(a,b,ro, phi)
data
# questa matrice esce dalla function e viene 'raccolta' nella variabile nomexx (es.,periodxx)

}
FINE FUNCTION

```

#Per richiamare la function:

nomexx = PRDGRAM(Nome_var_vettore dati, numerosità del campione, numero di armoniche da cercare)