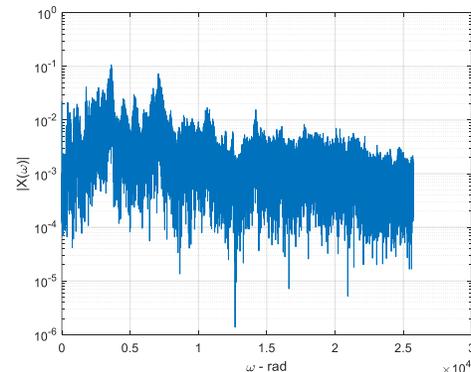
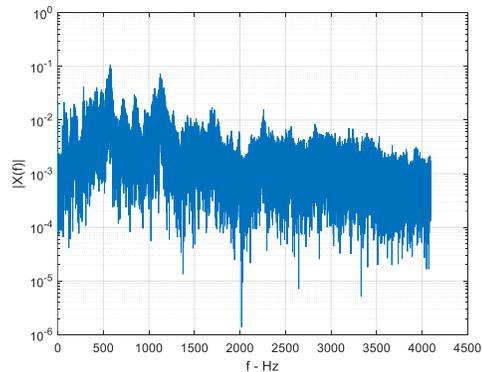
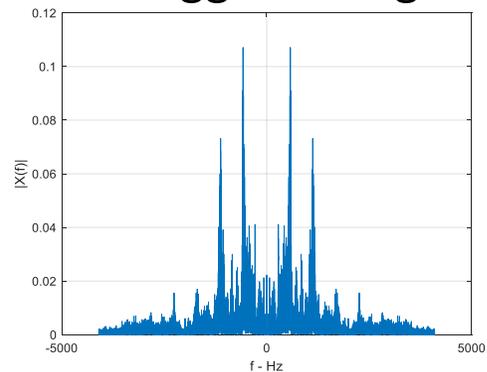


Esempio Nel file `handel.mat` si trova un segnale audio campionato. I campioni si trovano nella variabile `y`, mentre la frequenza di campionamento è in `Fs`.

Usando la traccia fornita, scrivere uno script che , riproduce il suono e mostra lo spettro in frequenza del segnale tempo continuo da cui sono stati tratti i campioni. Siccome il segnale da analizzare è reale, il suo spettro di ampiezza è pari, quindi se ne può visualizzare anche solo la parte corrispondente alle frequenza positive.

Per lo zero padding, prendere come valore di M la più piccola potenza di 2 maggiore o uguale a $8N$, dove N è il numero di campioni.



```
clearvars; close all;
% Carica un suono campionato e la freq. camp.
load("handel.mat");
whos % Mostra le variabili caricate
%sound(y,Fs); %Rimuovere il commento per ascoltare
% Zero-padding
M = 2^(nextpow2(numel(y))+3); % M>=8N
TC = ??? Determinare il per. camp. dalla freq. camp.
%-----
Y = ??? Usare TC, la fft e fftshift ???
F = ??? Freq iniziale : step : freq finale ???
%-----
figure(1);
plot(F,abs(Y)); grid minor
xlabel('f - Hz'); ylabel('|X(f)|');
% Per un segnale reale basta tracciare |X|
% per frequenze positive e prendere la 2da metà di Y
Fpos = 0:Fs/M:(Fs/2-Fs/M); Ypos = Y(M/2+1:end);
figure(2);
% Spesso si usa una scala log sulle ordinate:
semilogy(Fpos,abs(Ypos)); grid
xlabel('f - Hz'); ylabel('|X(f)|');
%
% Grafico di |X(omega)|
figure(3);
semilogy(2*pi*Fpos,abs(Ypos)); grid
xlabel('\omega - rad'); ylabel('|X(\omega)|');
```





Soluzione esercizio 1

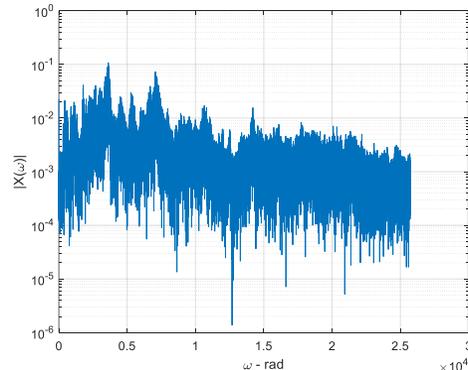
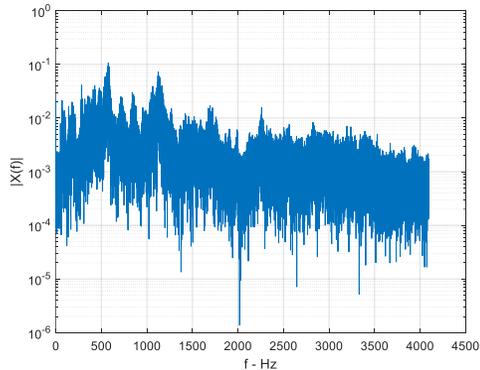
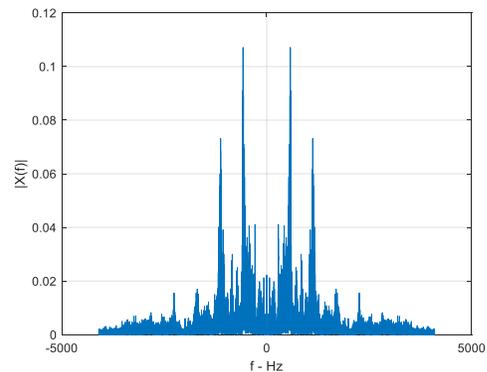
Il periodo di campionamento si calcola come $TC = 1/F_s$;

Con il comando $Y = TC*fftshift(fft(y,M))$;

si calcolano i campioni dello spettro del segnale y nell'intervallo di frequenze $\left(-\frac{F_s}{2}, \frac{F_s}{2} - \frac{F_s}{M}\right)$ con passo di campionamento $\frac{F_s}{M}$

Quindi il comando $F = -F_s/2 : F_s/M : F_s/2 - F_s/M$;
calcola i valori dell'asse delle ascisse.

Per prendere solo le frequenze positive basta partire da 0 nei valori di frequenza. I corrispondenti valori dello spettro sono nella seconda metà del vettore Y . Se non avessimo fatto **fftshift**, sarebbero stati nella prima metà



```
clearvars; close all;
% Carica un suono campionato e la freq. camp.
load("handel.mat");
whos % Mostra le variabili caricate
%sound(y,Fs); %Rimuovere il commento per ascoltare
% Zero-padding
M = 2^(nextpow2(numel(y))+3); % M>=8N
TC = 1/Fs;
%-----
Y = TC*fftshift(fft(y,M));
F = -Fs/2 : Fs/M : Fs/2-Fs/M;
%-----
figure(1);
plot(F,abs(Y)); grid minor
xlabel('f - Hz'); ylabel('|X(f)|');
% Per un segnale reale basta tracciare |X|
% per frequenze positive e prendere la 2da metà di Y
Fpos = 0:Fs/M:(Fs/2-Fs/M); Ypos = Y(M/2+1:end);
figure(2);
% Spesso si usa una scala log sulle ordinate:
semilogy(Fpos,abs(Ypos)); grid
xlabel('f - Hz'); ylabel('|X(f)|');
%
% Grafico di |X(omega)|
figure(3);
semilogy(2*pi*Fpos,abs(Ypos)); grid
xlabel('\omega - rad'); ylabel('|X(\omega)|');
```