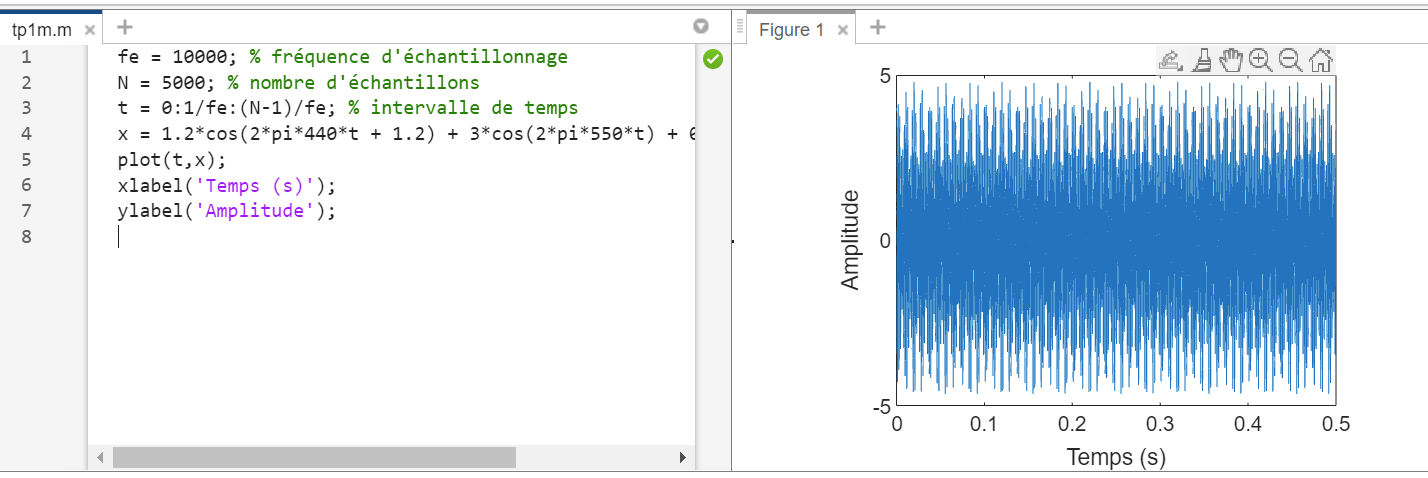
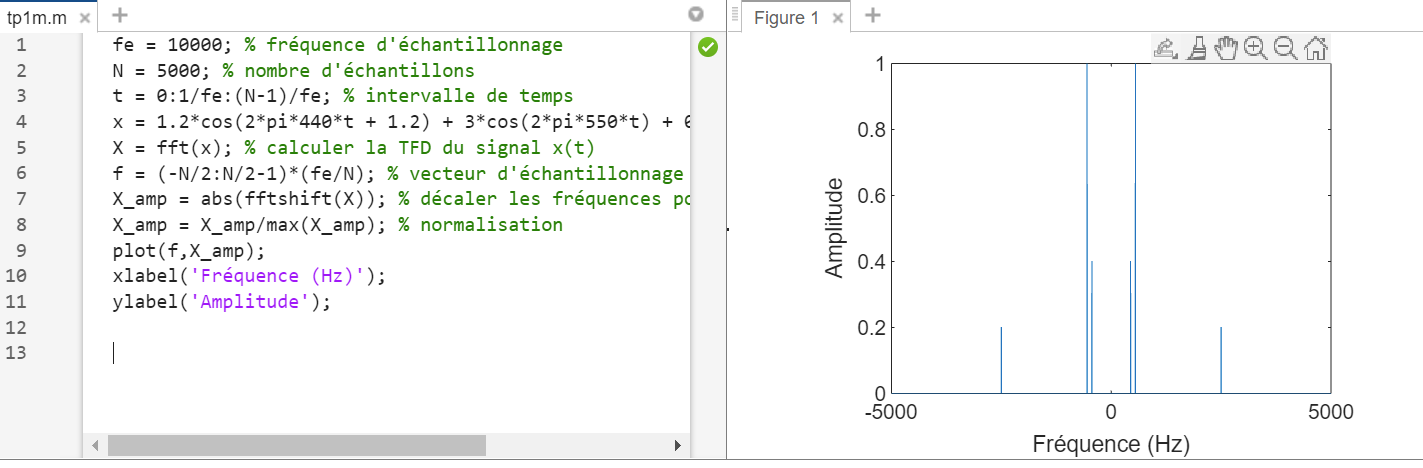
**TP1 Traitement de signal :**

BELMOUBARIK Merouane :

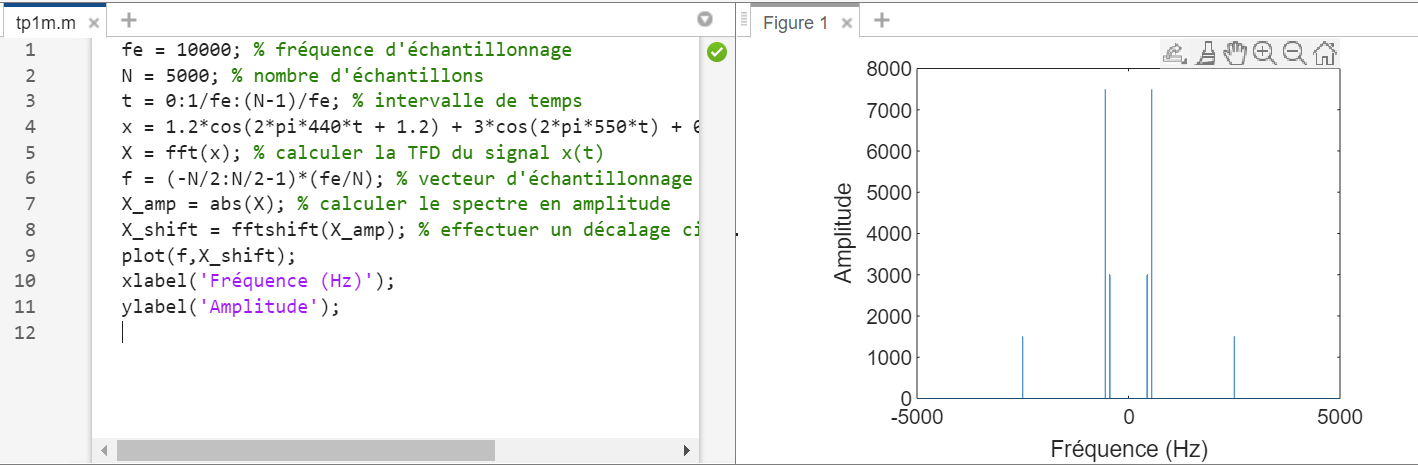
1. Tracer le signal x(t). Fréquence d’échantillonnage : fe = 10000Hz, Intervalle : Nombre d’échantillons : N = 5000.



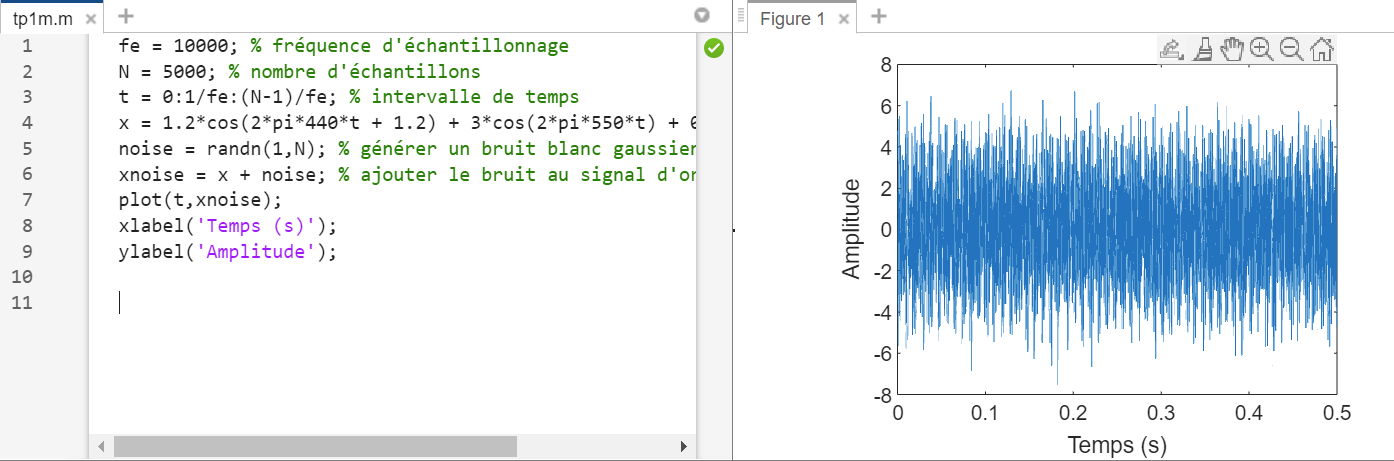
1. - Calculer la TFD du signal x(t) en utilisant la commande fft, puis tracer son spectre en amplitude après avoir créé le vecteur f qui correspond à l'échantillonnage du signal dans l'espace fréquentiel. Utiliser la commande abs pour afficher le spectre d’amplitude :



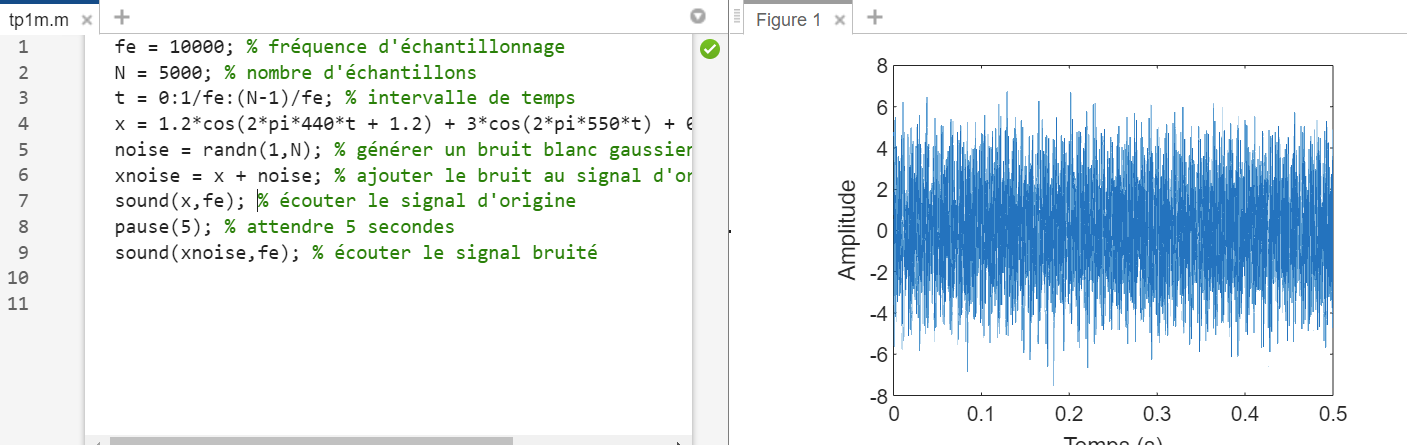
1. Pour mieux visualiser le contenu fréquentiel du signal, utiliser la fonction fftshift, qui effectue un décalage circulaire centré sur zéro du spectre en amplitude obtenu par la commande fft.



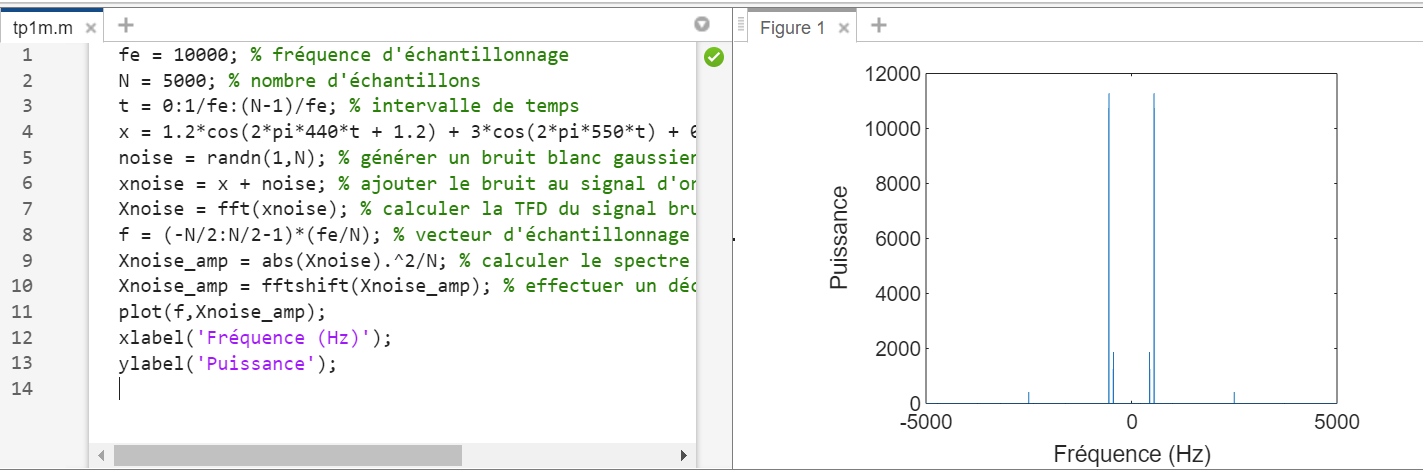
1. Créer un nouveau signal xnoise, en introduisant un bruit blanc gaussien dans le signal d’origine x(t), puis visualisez-le. Utiliser la commande randn pour générer ce bruit. Il est à noter qu’un bruit blanc est une réalisation d'un processus aléatoire dans lequel la densité spectrale de puissance est la même pour toutes les fréquences de la bande passante. Ce bruit suit une loi normale de moyenne 0 et d’écart type 1.



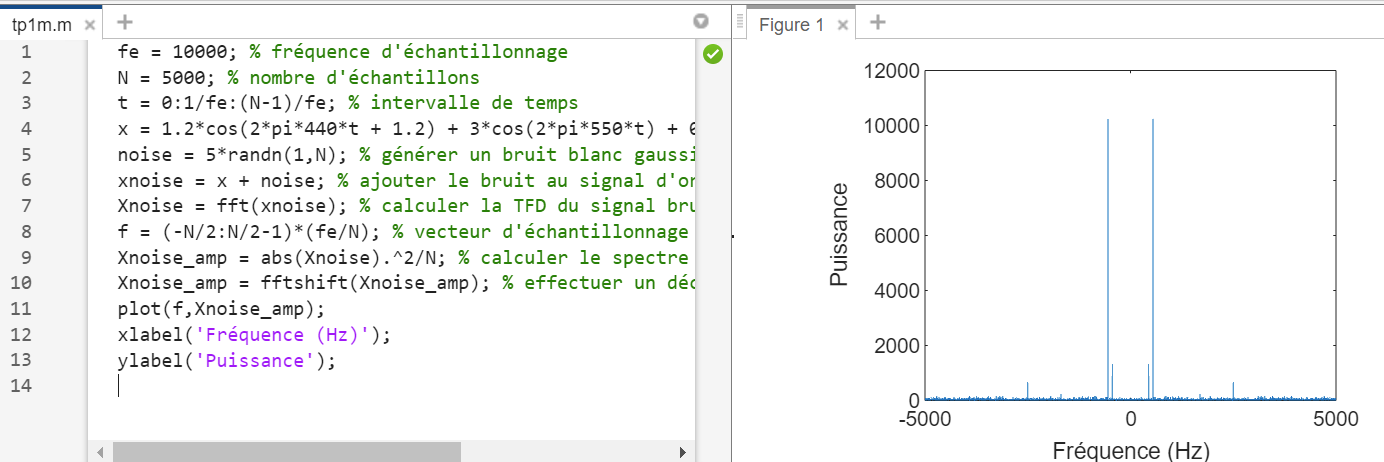
1. Utiliser la commande sound pour écouter le signal et puis le signal bruité.



1. Calculez puis tracer le spectre de puissance du signal bruité centré à la fréquence zéro.



1. Augmenter l’intensité de bruit puis afficher le spectre. Interpréter le résultat obtenu.



En augmentant l'intensité de bruit, l'effet sur le spectre de puissance sera d'avoir des valeurs plus élevées dans les fréquences aléatoires (c'est-à-dire, les fréquences qui correspondent au bruit) par rapport aux fréquences du signal d'origine. Cela signifie que la répartition de l'énergie du signal est plus importante dans les fréquences du bruit que dans les fréquences du signal d'origine, ce qui rend plus difficile de détecter les caractéristiques du signal d'origine.

**Partie 2 : bluewhale**

**Code :**

clear all

close all

clc

[x,fe]=audioread("bluewhale.au");

chant = x(2.45e4:3.10e4);

% sound(chant,fe)

N = length(chant);

te = 1/fe;

t = (0:N-1)\*(10\*te);

% plot(t,chant)

y = abs(fft(chant)).^2/N;

f = (0:floor(N/2))\*(fe/N)/10;

plot(f,y(1:floor(N/2)+1));

**Plot :**

