```
%Homework 3 %involved m-files: specdens.m (compute the spectral density of a % function % noisekiller.m (low-band filter for hbn) % noisepanpan.m close all;clear all; tic if 1 %==============EXERCICE 1================= %initialization load('hw9.mat'); if 1 %1.1) plotting the signal and the spectral density========== figure subplot(1,2,1),plot(t,(hbn)) subplot(1,2,2),plot(t,10*log10(hbn)); %IMPORTANT: the spectral density should be plot in function of the %frequencies, and not in function of the time. However, the spectral density %and the fourier transform of the signal have the same scale, and we juste want to %have a look at the shape of the signal, and use filters before applying %inverse fourier transform on it. For this reason, the scale of the %signal is not important, and we use t as a frequency vector. figure subplot(1,2,1),plot(t,specdens(hbn)) subplot(1,2,2),plot(t,10*log10(specdens(hbn))); end %1.2)=== Filtering================================ %first, we calculate the fourier transform of hbn ft=fftshift(fft(hbn)); if 1 figure plot(t,ft) end %we can see the peaks. Now we want to use filters to clear the signal peak1=find(t>0.4757); peak2=find(t>0.5245); peak=[peak1(1) ; peak2(1)]; %1st filtering for various tolerance (tol=1 used for the rest of the %exercise) for i=1:10 clearedsign(:,i)=noisepanpan(ft,peak,i); clearedspec(:,i)=noisepanpan(specdens(hbn),peak,i); end %plotting the signal and its spectral density after aplying filter for for %given tolerance (tol=1) %filtering oscillation figure subplot(1,2,1),plot(t,ifft(ifftshift(clearedsign(:,1)))) subplot(1,2,2),plot(t,10*log10(ifft(ifftshift(clearedsign(:,1))))); figure subplot(1,2,1),plot(t,clearedspec(:,1)) subplot(1,2,2),plot(t,10*log10(clearedspec(:,1))); %2nd filtering peak1=find(t>0.4762); peak2=find(t>0.5241); peak=[peak1(1) ; peak2(1)]; for i=1:10 clearedsign(:,i)=noisekiller(clearedsign(:,i),peak); clearedspec(:,i)=noisekiller(clearedspec(:,i),peak); end %filtering oscillation+background noise figure subplot(1,2,1),plot(t,ifft(ifftshift(clearedsign(:,1)))) subplot(1,2,2),plot(t,10*log10(ifft(ifftshift(clearedsign(:,1))))); figure subplot(1,2,1),plot(t,clearedspec(:,1)) subplot(1,2,2),plot(t,10*log10(clearedspec(:,1))); %1.3)==========Original vs Filtered signal======= %plotting the old and the new signal in dB scale figure subplot(1,2,1),plot(t,10*log10(hbn)); subplot(1,2,2),plot(t,10*log10(ifft(ifftshift(clearedsign(:,1))))); end if 1 %==============EXERCICE 2================ signal=wavread('hw9.wav'); fsign=fftshift(fft(signal)); t=[1:1:length(signal)]'; if 1 %plotting the original signal subplot(1,2,1),plot(t,signal) subplot(1,2,2),plot(t,real(fsign)) end %decomposition of the fourier transorm of signal in 4 equals parts+central part for i=1:9975 part1(i)=fsign(i); part4(i)=fsign(length(fsign)-9975+i); end for i=1:9975 part2(i)=fsign(9975+i); part3(i)=fsign(19951+i); end central=fsign(19951); % for i=1:13000 % part1(i)=fsign(i); % part4(i)=fsign(length(fsign)-13000+i); % end % % for i=1:6950 % part2(i)=fsign(13000+i); % part3(i)=fsign(13001+6950+i); % end % central=fsign(19951); %recomposition of the fourier transform of the new signal for i=1:9975 fnew(i)=part3(i); end for i=9976:19950 fnew(i)=part4(i-9975); end fnew(19951)=central; for i=19952:29926 fnew(i)=part1(i-19951); end for i=29927:39901 fnew(i)=part2(i-29926); end % % for i=1:6950 % fnew(i)=part3(i); % end % for i=6951:19950 % fnew(i)=part4(i-6950); % end % fnew(19951)=central; % for i=19952:32951 % fnew(i)=part1(i-19951); % end % for i=32952:39901 % fnew(i)=part2(i-32951); % end if 1 %plotting the new signal and its fourier transform t1=(1:9975); t2=(9976:19950); tc=(19950:19952); t3=(19952:29926); t4=(29927:39901); central=[fnew(19950) fnew(19951), fnew(19952)]; figure subplot(1,2,1),plot(t1,part1,'g',t2,part2,'b',tc,central,'k',t3,part3,'r',t4,part4,'m'); subplot(1,2,2),plot(t1,part3,'r',t2,part4,'m',tc,central,'k',t3,part1,'g',t4,part2,'b'); new=ifft(ifftshift(fnew)); figure plot(t,new) end if 1 %comparison betwenn old and new sound
```

```
sound(wavread('hw9.wav')) sound(new) end end taratapouf=toc
```