MACHINE LEARNING FOR SIGNAL PROCESSING

HOMEWORK 1

İLKER BALTACI

040160237

AHMET BEDRİ YORULMAZ

040160050

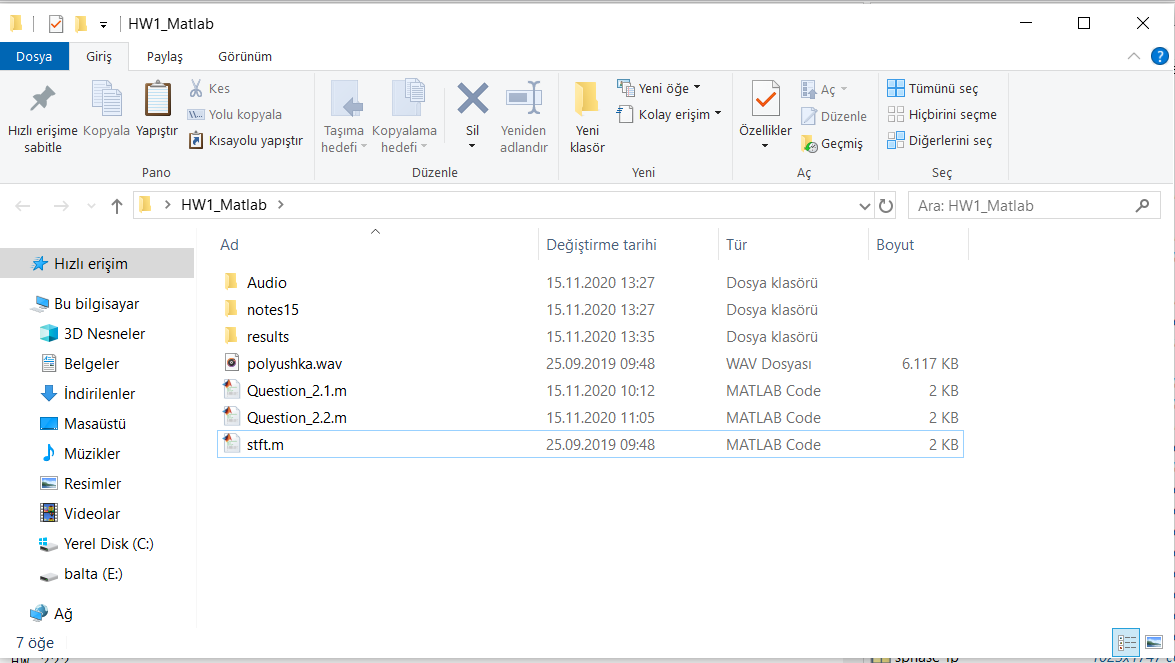
BURAK ŞİMŞEK

040160206

# Question 2.1

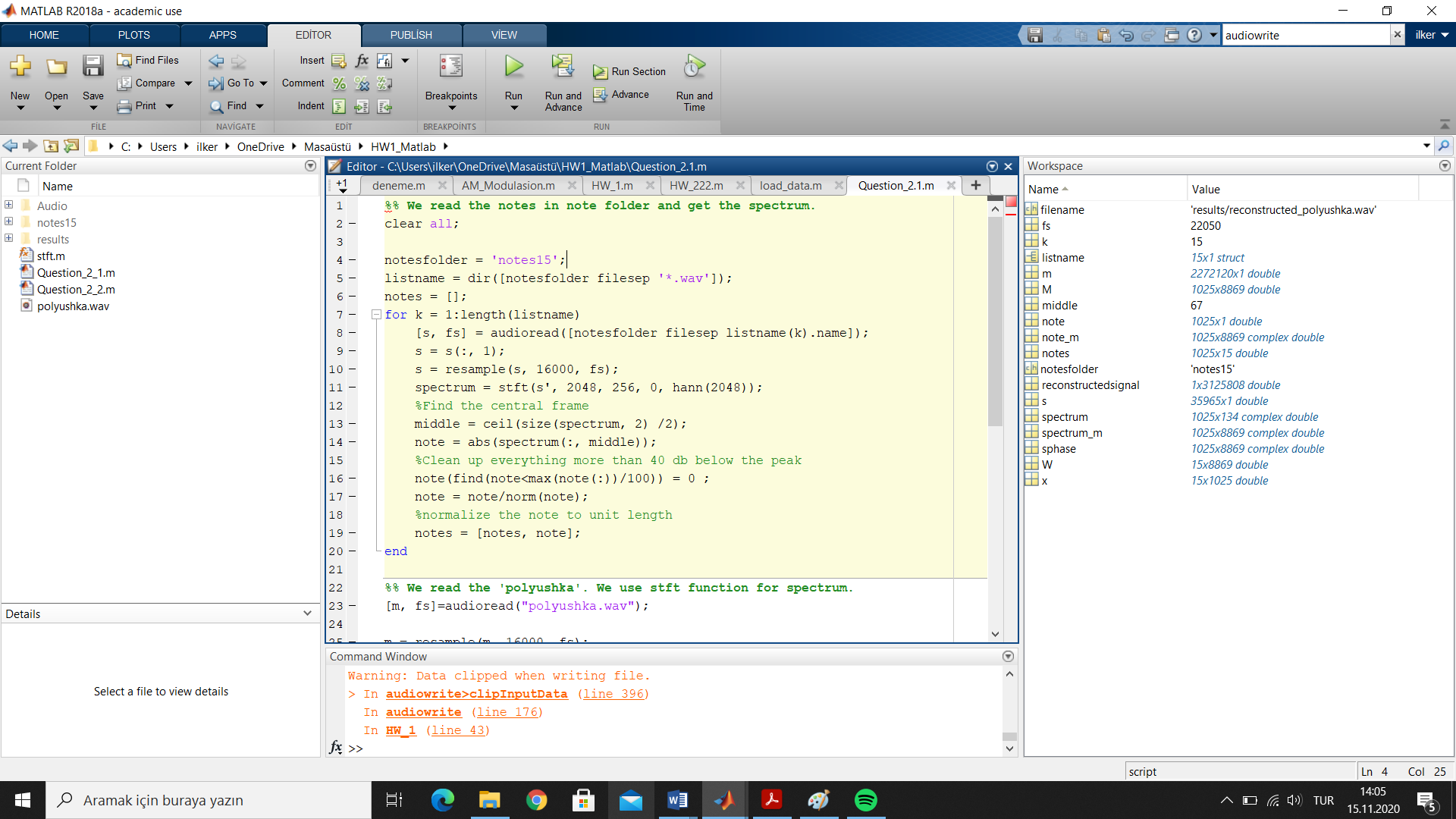
We created the matlab file called ‘Question\_2.1.m’. To achieve the answer of the question 2.1, we applied all steps in one matlab file.

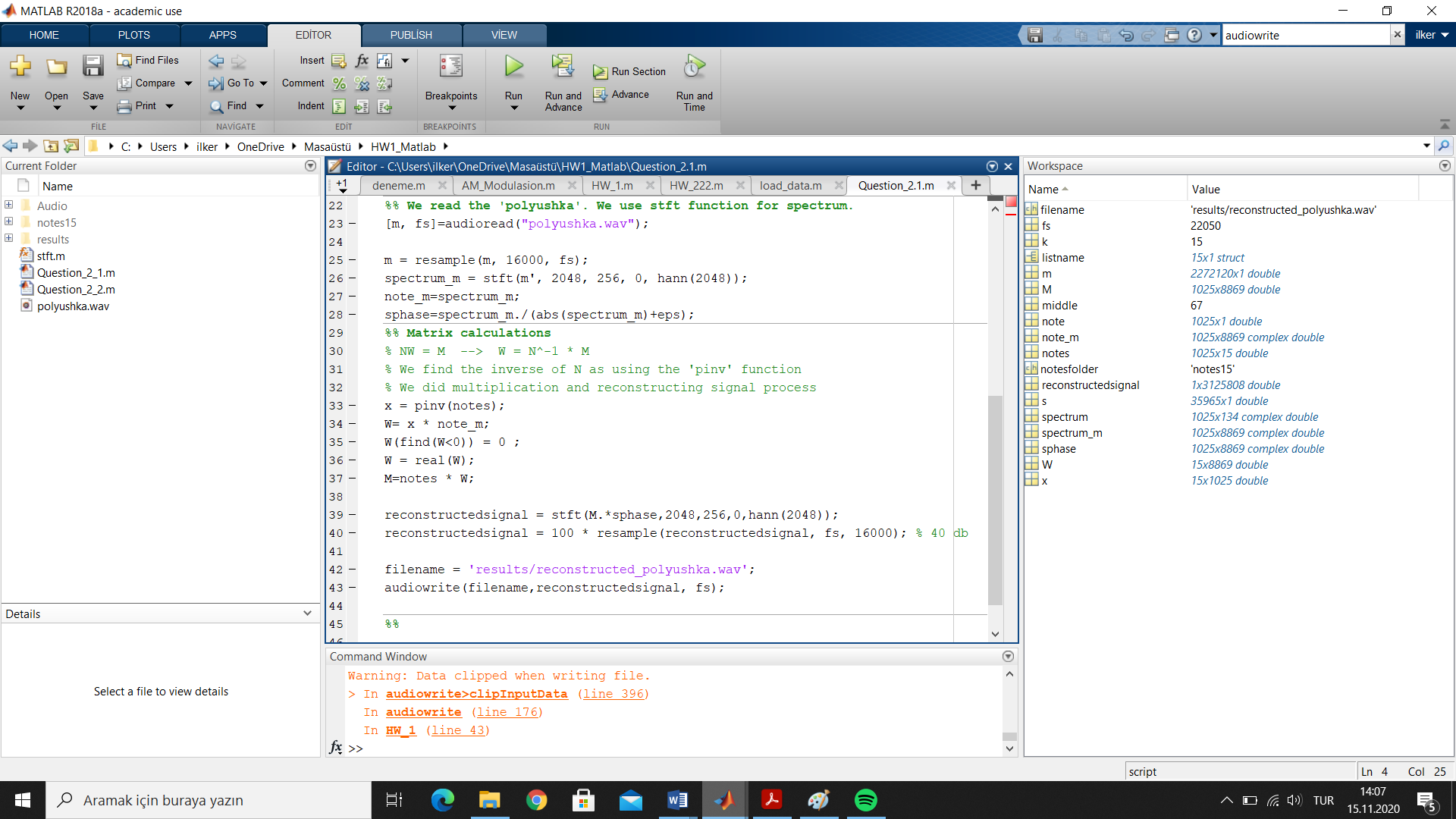
Firstly, when download the HW1\_Matlab file, you must see a file shown below.



If you run the Question\_2.1.m and Question\_2.2.m files, the answered music files for Q1 and Q2 will be saved to the file called ‘results’.

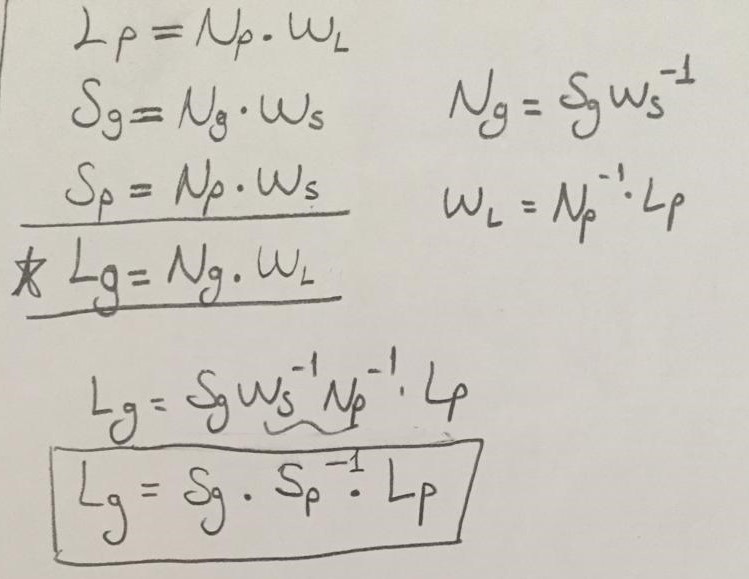
We read the notes in the note folder and get the spectrum. We applied the same steps for “polyushka.wav” file. After that, we did matrix operations to obtain the W matrix. To obtain inverse matrix of the notes matrix, we used pinv() function. We take real values in W matrix. Then, we created new music and saved to results file.



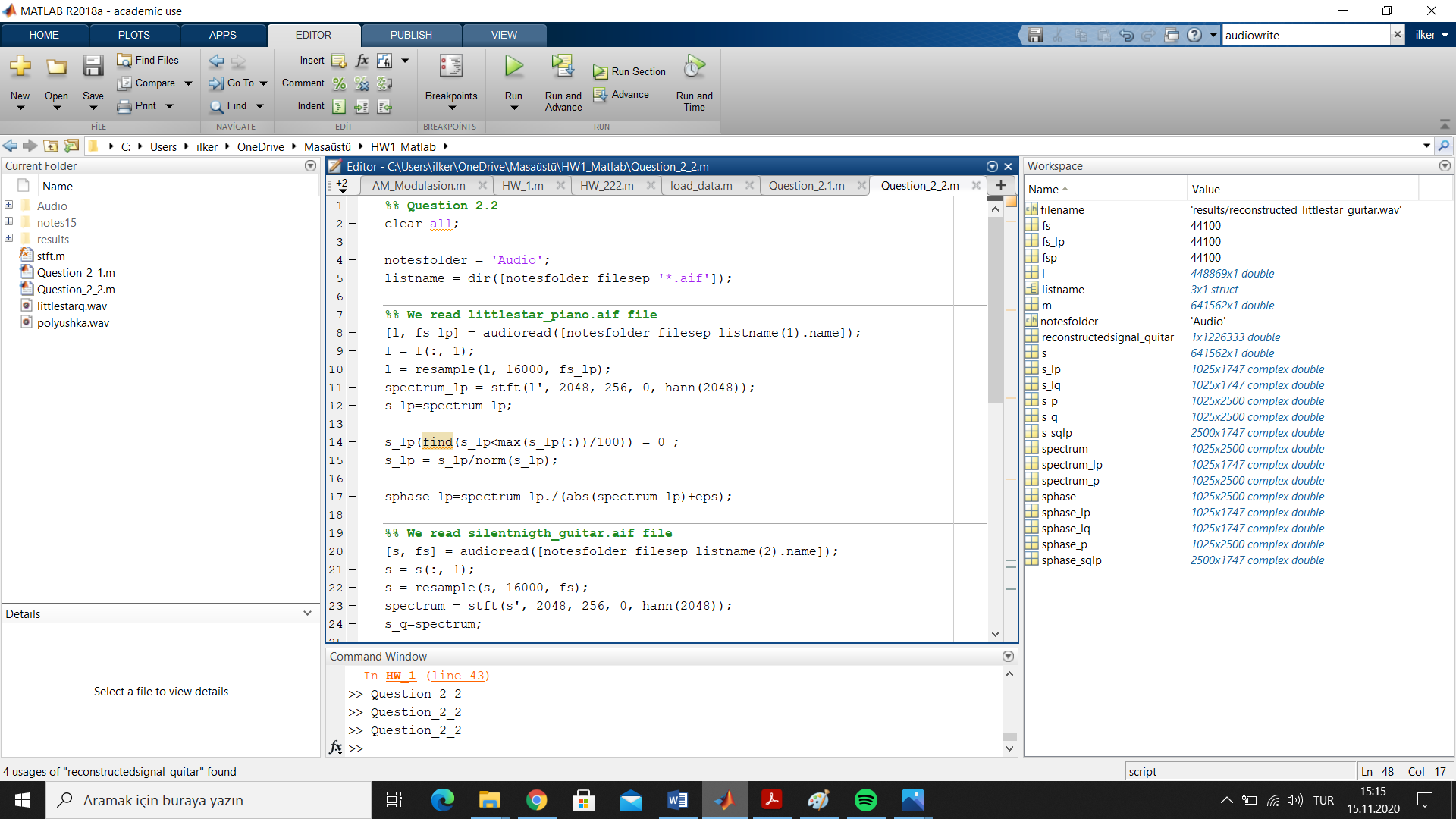


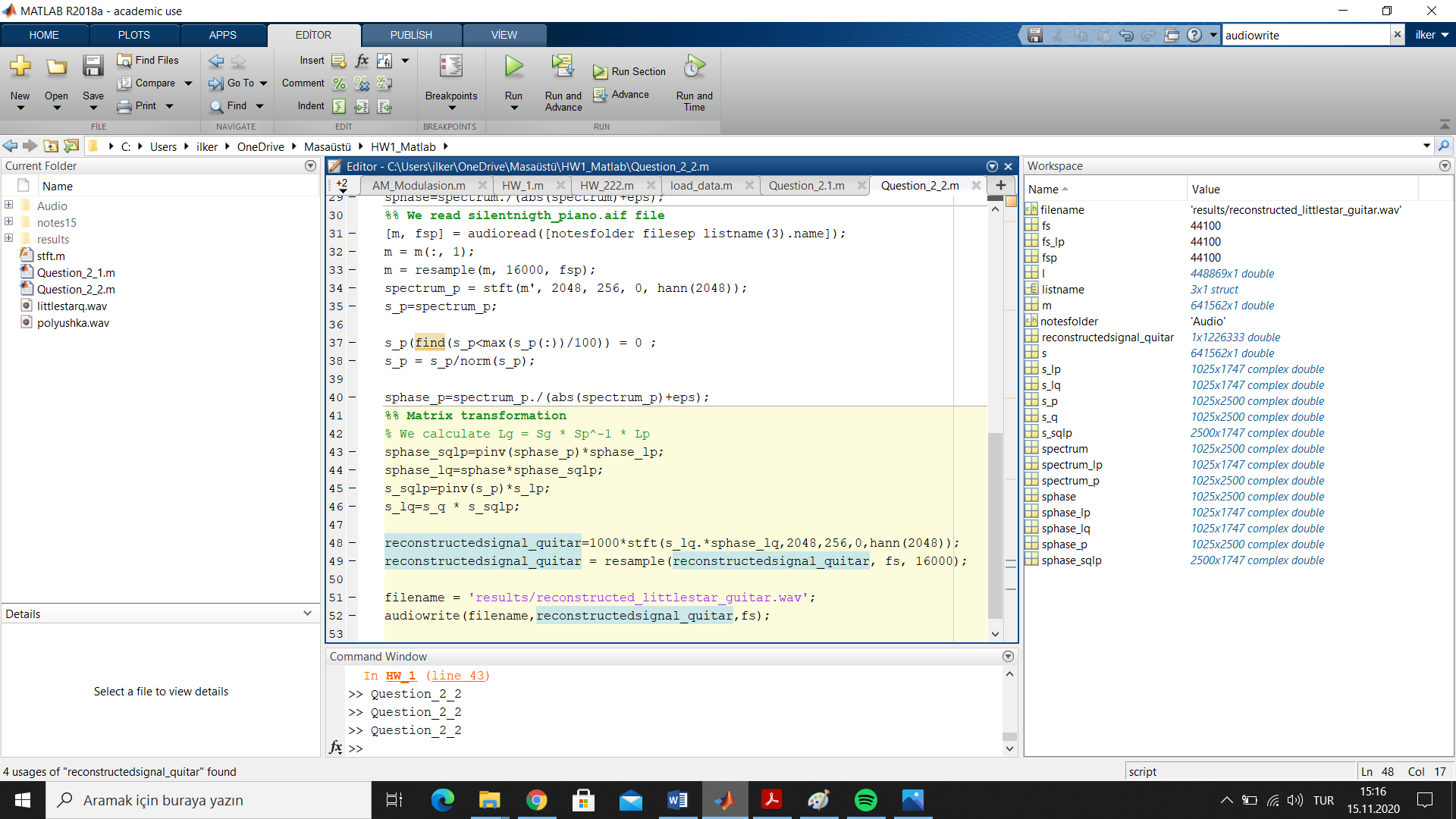
# Question 2.2

We solved matrix equations to obtain ‘littlestar\_guitar.wav’ file. You can see the matrix calculation steps below.



We solved the equation with matlab and recovered the signal. Then, we created guitar version of littlestar song.





# MATLAB CODES

Q2.1)

%% We read the notes in note folder and get the spectrum.

clear all;

notesfolder = 'notes15';

listname = dir([notesfolder filesep '\*.wav']);

notes = [];

for k = 1:length(listname)

[s, fs] = audioread([notesfolder filesep listname(k).name]);

s = s(:, 1);

s = resample(s, 16000, fs);

spectrum = stft(s', 2048, 256, 0, hann(2048));

%Find the central frame

middle = ceil(size(spectrum, 2) /2);

note = abs(spectrum(:, middle));

%Clean up everything more than 40 db below the peak

note(find(note<max(note(:))/100)) = 0 ;

note = note/norm(note);

%normalize the note to unit length

notes = [notes, note];

end

%% We read the 'polyushka'. We use stft function for spectrum.

[m, fs]=audioread("polyushka.wav");

m = resample(m, 16000, fs);

spectrum\_m = stft(m', 2048, 256, 0, hann(2048));

note\_m=spectrum\_m;

sphase=spectrum\_m./(abs(spectrum\_m)+eps);

%% Matrix calculations

% NW = M --> W = N^-1 \* M

% We find the inverse of N as using the 'pinv' function

% We did multiplication and reconstructing signal process

x = pinv(notes);

W= x \* note\_m;

W(find(W<0)) = 0 ;

W = real(W);

M=notes \* W;

reconstructedsignal = stft(M.\*sphase,2048,256,0,hann(2048));

reconstructedsignal = 100 \* resample(reconstructedsignal, fs, 16000); % 40 db

filename = 'results/reconstructed\_polyushka.wav';

audiowrite(filename,reconstructedsignal, fs);

%%

Q2.2)

%% Question 2.2

clear all;

notesfolder = 'Audio';

listname = dir([notesfolder filesep '\*.aif']);

%% We read littlestar\_piano.aif file

[l, fs\_lp] = audioread([notesfolder filesep listname(1).name]);

l = l(:, 1);

l = resample(l, 16000, fs\_lp);

spectrum\_lp = stft(l', 2048, 256, 0, hann(2048));

s\_lp=spectrum\_lp;

s\_lp(find(s\_lp<max(s\_lp(:))/100)) = 0 ;

s\_lp = s\_lp/norm(s\_lp);

sphase\_lp=spectrum\_lp./(abs(spectrum\_lp)+eps);

%% We read silentnigth\_guitar.aif file

[s, fs] = audioread([notesfolder filesep listname(2).name]);

s = s(:, 1);

s = resample(s, 16000, fs);

spectrum = stft(s', 2048, 256, 0, hann(2048));

s\_q=spectrum;

s\_q(find(s\_q<max(s\_q(:))/100)) = 0 ;

s\_q = s\_q/norm(s\_q);

sphase=spectrum./(abs(spectrum)+eps);

%% We read silentnigth\_piano.aif file

[m, fsp] = audioread([notesfolder filesep listname(3).name]);

m = m(:, 1);

m = resample(m, 16000, fsp);

spectrum\_p = stft(m', 2048, 256, 0, hann(2048));

s\_p=spectrum\_p;

s\_p(find(s\_p<max(s\_p(:))/100)) = 0 ;

s\_p = s\_p/norm(s\_p);

sphase\_p=spectrum\_p./(abs(spectrum\_p)+eps);

%% Matrix transformation

% We calculate Lg = Sg \* Sp^-1 \* Lp

sphase\_sqlp=pinv(sphase\_p)\*sphase\_lp;

sphase\_lq=sphase\*sphase\_sqlp;

s\_sqlp=pinv(s\_p)\*s\_lp;

s\_lq=s\_q \* s\_sqlp;

reconstructedsignal\_quitar=1000\*stft(s\_lq.\*sphase\_lq,2048,256,0,hann(2048));

reconstructedsignal\_quitar = resample(reconstructedsignal\_quitar, fs, 16000);

filename = 'results/reconstructed\_littlestar\_guitar.wav';

audiowrite(filename,reconstructedsignal\_quitar,fs);