

Αλέξανδρος Βυθούλκας 2013021

Απόστολος Καμπλιώνης 2015121

Επεξεργασία Ομιλίας και Ήχου - Απαλλακτική Εργασία:

“Μετατροπή του γραπτού λόγου σε ομιλία (Text To Speech – Speech To Text) και αναγνώριση γλώσσας”

Παραδοτέο 1 / Παρουσίαση προόδου εργασίας(18/3/22)

Όσο αφορά το 1ο Παραδοτέο, αρχικά, εξοικειωθήκαμε περαιτέρω με το Matlab και τις δυνατότητες που προσφέρει, ερχόμενοι από απο τυπική μελέτη της χρήσης του Octave.

Στη συνέχεια, επηρεασμένοι από την θεωρητική ύλη του μαθήματος το ενδιαφέρον μας επικεντρώθηκε στο στήσιμο μιας εφαρμογής που θα συνδυάζει λειτουργίες όπως **Text-To-Speech(TTS)** και **Speech-To-Text(STT)**. Θεωρήσαμε πως θα αποβεί ένα ιδιαίτερα χρήσιμο εργαλείο για άτομα που παρουσιάζουν προβλήματα ομιλίας και όρασης.

Η έναρξη της υλοποίησης πραγματοποιήθηκε με την έρευνα και αναζήτηση του κατάλληλου υλικού. Ειδικότερα εντοπίσαμε σε ένα guide-thread πληροφορίες για το στήσιμο της συνάρτησης **TTS**, που κάνει χρήση του **Microsoft Win32 SAPI**. Ενδεικτικά, παραθέτουμε τον σχολιασμένο κώδικα μετά από την μελέτη που του κάναμε.

```
if ~ispc, errordlg('Microsoft Win32 SAPI is required.','Operating System Error'); end %error('Microsoft Win32 SAPI is required.');
```

```
end %ελεγχει εαν γπαρχει το Mirosoft Speech API dhladh ean to leitourgiko einai windows
```

```
if ~ischar(txt), errordlg('First input must be string.','Text input Error'); end %error('First input must be string.');
```

```
end %ελεγχει εαν to text input ths synarthshs einai ontws string alliws bgazei error.
```

```
SV = actxserver('SAPI.SpVoice');%Dhmiourgei ena Microsoft Speech API antikeimeno. Sygkekrimena dhmiourgei enan local OLE Automation server gia na mporesoyme na xrhsimopoihsoyme thn yphresia SAPI twn Windows kai to
```

```
% apotelesma ths synarthshs actserver ekxwreitai sthn metablth SV.
```

```
TK = invoke(SV,'GetVoices');% Epikaleitai thn methodo 'GetVoices' ths yphresias SAPI poy thn exoyme ekxwrhsei sto SV mesw ths synarthshs actserver kai thn ekxwrei sthn metablth TK.
```

```
if nargin > 1 %ελεγχει αν to number of function input arguments einai megalytero apo 1 tote
```

```
    % Select voice;
```

```
    for k = 0:TK.Count-1 %epanalhph for gia na dialexei fwnh to systhma apo 0 ews thn timh TK.Count-1
```

```
        if strcmpi(voice,TK.Item(k).GetDescription) %ελεγχει αν h symboloseira voice einai iidia me thn symboloseira toy sapi mesw ths TK metablthhs.
```

```
            SV.Voice = TK.Item(k); %ekxwreitai sto antikeimeno SV voice to antikeimeno TK gia thn sygkekrimenh loupα
```

```
            break; %termatizetai h ektelesh tou for xwris na ektelestoy oi parakatw entoles mesa se ayto
```

```
        elseif strcmpi(voice,'list') % alliws an h symboloseira voice einai idio me to 'list'
```

```
            disp(TK.Item(k).GetDescription); %emfanizei thn lista me ta diathesima voices pou mporei na parei to SAPI
```

```
        end
```

```
    end
```

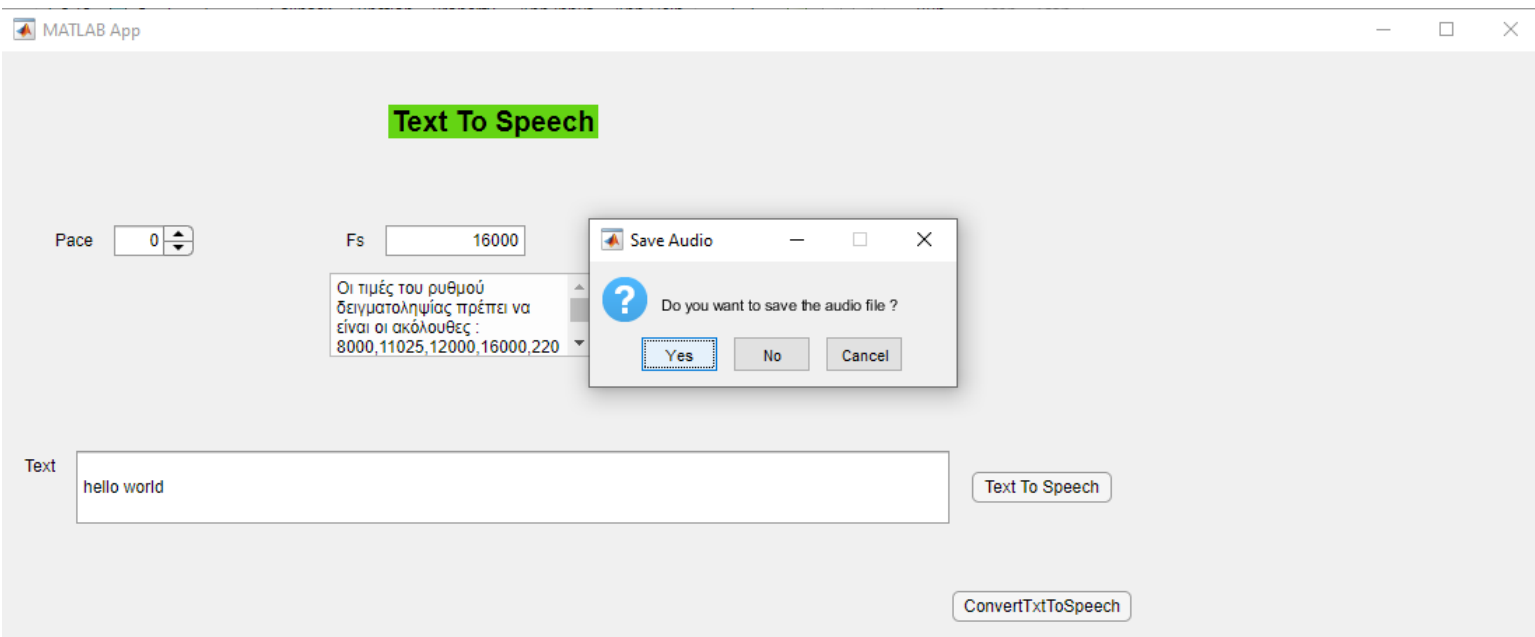
```

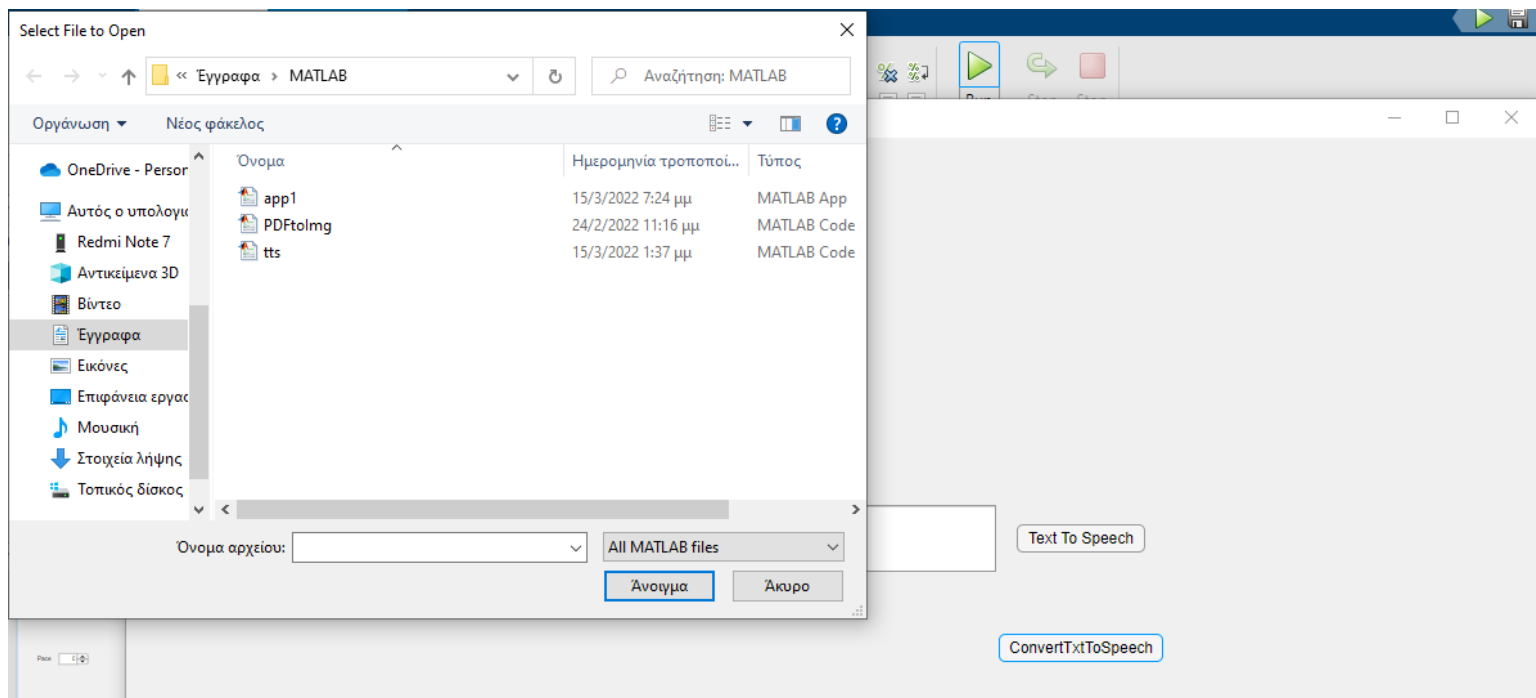
if nargin < 4 || ~ismember(fs,[8000,11025,12000,16000,22050,24000,32000,...
    44100,48000]), fs = 16000; end %elegxei an to number of function input arguments einaimikrotero apo 4 h an to fs einai diaforo apo tis times pou mporei
if nargin > 0 % elegxei ean to number of function output arguments einai megalytero tou 0 tote
    % Output variable;
    MS = actxserver('SAPI.SpMemoryStream'); %ekxwrei sthn metablhth MS thn leitoyrgia tou SAPI.spMemoryStream. To actserver kanei oti eipame kai pio panw.
    MS.Format.Type = sprintf('SAFT%dkHz16BitMono',fix(fs/1000)); %sto antikeimeno MS.Format.Type ekxwreitai h timh ths morfopoihmenhs timhs ths strogkylopaihshs
    SV.AudioOutputStream = MS; %ekxwrei sto antikeimeno SV.AudioOutputStream thn timh MS
end
invoke(SV,'Speak',txt); % Epikaleitai thn methodo 'Speak' ths yphresias SAPI pou thn exoyme ekxwrhsei sto SV mesw ths synarthshs actserver
if nargin > 0 % elegxei ean to number of function output arguments einai megalytero tou 0 tote
    % Convert uint8 to double precision; unsigned integer number stored
    % with 8 bits
    wav = reshape(double(invoke(MS,'GetData')),2,[]);
    wav = (wav(:,2)*256+wav(:,1))/32768;
    wav(wav >= 1) = wav(wav >= 1)-2;
    delete(MS);
    clear MS;
end
delete(SV);
clear SV TK;
pause(0.2);
end % TTS;

```

Υλοποίηση του TTS κομματιού με την δημιουργία του κατάλληλου UI

Συγκεκριμένα δημιουργήθηκαν 2 buttons, ένα που εκτελεί την λειτουργία TTS της αντίστοιχης συνάρτησης και ένα άλλο που να κάνει convert ολόκληρο txt file σε string και μετά να εκτελεί την TTS συνάρτηση. Επίσης, έγινε η αρχή δημιουργίας ενός dialog box που θα ζητάει από τον χρήστη εάν θέλει να αποθηκεύσει το παραγόμενο ηχητικό ή όχι. Παραθέτουμε screenshots από την μέχρι τώρα πρόοδο μαζί με τον κώδικα.





Αυτά είναι τα dialog boxes που έχουμε φτιάξει. Το ένα είναι όταν δημιουργείται το ηχητικό από τη συνάρτηση TTS να ζητάει να το αποθηκεύσει και το δεύτερο είναι από το convert button που ζητάει από τον χρήστη να κάνει upload ένα txt αρχείο για να το κάνει μετατροπή σε συμβολοσειρά.

Ο κώδικας που έχουμε γράψει μέχρι τώρα, δύο callback functions για το πάτημα των κουμπιών με τον εξής κώδικα:

```
%a = uigetfile();
%fid = fopen(a);
%txt = fscanf(fid,'%s');
%formatSpec = '%s'
%txt = extractFileText(a);
txt = app.TextEditField.Value; %pairnei thn timh poy exoyme dwsei sto plaisio toy gui g
pace = app.PaceSpinner.Value; %pairnei thn timh poy exoyme dwsei gia to pace mesw gui
fs = app.FsEditField.Value; %pairnei thn timh poy exoyme dwsei gia to fs mesw gui
tts(txt,'1',pace,fs); %ekteleitai h synarthsh tts gia thn metatroph tou keimenoy se omi
answer = questdlg('Do you want to save the audio file ?', 'Save Audio');
wav = tts(txt,'1',pace,fs);
if answer == 'Yes', audiowrite('sample.wav',wav,fs); end
%fclose(fid);
```

Το μόνο πράγμα που αλλάζει για το convert του txt είναι ότι αντί να πάρει σαν είσοδο το κείμενο που θα δώσει ο χρήστης απευθείας από το γραφικό περιβάλλον, θα πάρει σαν είσοδο το κείμενο που θα δώσει ο χρήστης από το dialog box και με την εντολή **fscanf** θα το μετατρέψει σε συμβολοσειρά.