Zadanie 20.05.2020

Dźwięk i operacje na wektorach i macierzach

W wykonaniu tego zadania pomocny będzie wykład z 3 kwietnia, najlepiej zacząć od przypomnienia sobie wiadomości tam zawartych.
Jako wstęp do wykonania tego ćwiczenia należy za pomocą funkcji ‘audioread’ wczytać sobie do Matlaba dowolny, krótki plik dźwiękowy (najlepiej nagrane wcześniej, krótkie zdanie).
Przykład, pokazujący odczyt pliku dźwiękowego i jego odtworzenie

```
>> [Y,Fs]=audioread('test.mp3');
>> p=audioplayer(Y,Fs);
>> p.play()
```

(Na uwagę zasługuje fakt, że funkcja ‘audioplayer’ zwraca obiekt audio który, posiada różne metody w tym metodę ‘play’, spotykamy się więc tutaj z elementem programowania obiektowego, znanym z C++). Oczywiście dokładny opis funkcji uzyskamy jak zawsze wpisując np.:

```
>> help audioread
```

Mając w przestrzeni roboczej Matlaba wektor (właściwie macierz) będący reprezentacją dźwięku, możemy wykonać sobie na nim dla testu jakieś proste przekształcenia, np.: zmienić nieco częstotliwość próbkowania, wyciąć fragment, wzmocnić fragment, przefiltrować używając wcześniej napisanej funkcji itp…
W dalszych działaniach najlepiej utworzyć sobie wektor, który będzie reprezentował dźwięk mono (domyślnie dźwięk zapisany jest w dwóch ścieżkach czyli stereo), najłatwiej po prostu wybrać sobie jeden z kanałów (czyli jedną z dwóch kolumn macierzy reprezentującej dźwięk).

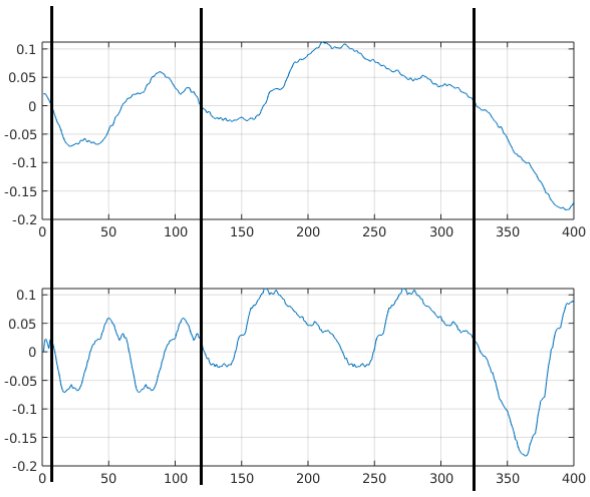
Zadaniem do wykonania będzie utworzenie trzech funkcji i przetestowanie ich działania.

Funkcja 1, pogłos: (5 punktów)

Funkcja ma dodawać pogłos, do przebiegu podanego na wejściu. Pogłos jest to kilkukrotne powtórzenie dźwięku z różnym opóźnieniem zsumowane oczywiście z dźwiękiem podstawowym.
Informację czym jest pogłos można oczywiście znaleźć w sieci. Proszę tak dobrać parametry efektu aby był on faktycznie słyszalny.

Funkcja 2, zniekształcenie , podwyższenie tonów bez przyspieszenia odtwarzania: (8 punktów)

Proszę utworzyć procedurę działającą według algorytmu:
każdy fragment przebiegu zawarty między przejściem sygnału przez zero należy „podwoić” to znaczy w sygnale wynikowym wpisać dwa razy. Aby czas trwania sygnału się nie zmienił można wybrać co drugą próbkę sygnału wejściowego, tak aby „podwojony” sygnał dalej zajmował tyle samo próbek. Efekt działania tej procedury powinien być taki, że otrzymany sygnał będzie trwał mniej więcej tyle samo co sygnał źródłowy, ale ton głosu mówiącego będzie zniekształcony, wyższy. Sposób działania procedury obrazuje rysunek:



Funkcja 3, spektrum: (3 punkty)
Dołączony skrypt pokazujący jak można użyć funkcji fft, proszę zamienić w funkcję, która rysuje wykres częstotliwościowy. Funkcja ma pobierać dwa argumenty: przebieg i Fs (częstotliwość próbkowania) i działać ogólnie dla dowolnych sygnałów (nie jedynie dla konkretnego)

Całość należy wypróbować za pomocą skryptu: (4 punkty)
Skrypt ma wczytywać sygnał z pliku audio, testować napisane wcześniej funkcje, w tym pokazywać spektrum sygnału przed zniekształceniem i po zniekształceniu

We wszystkich plikach proszę umieścić stosowne komentarze, nazwisko autora.