Note: You have one week to solve the tutorials. The theoretical part should be submitted before the lecture. The practical part has to be send to your tutor in a common compressed file format until Monday. There will be about 8 tutorials in two blocks (1-4) and (4-8). You can get 12 points for each tutorial. You need half of the maximal accessible points in each block in order to participate in the final exam. The code should be well structured and documented. Do not use any Matlab-Toolbox if not mentioned that you could use it. You should be able to present your work in the Mechatronik-Cip-Pool.

(6P) Exercise 1

In this exercise you will analyse the linear predictive coding (LPC) algorithm.

(1P) Subtask 1.1
Read the documentation of the Matlab command lpc and explain the parameters.

(2P) Subtask 1.2
Implement a Matlab function which computes the estimated signal

$$\hat{x}(n) = \sum_{k=1}^{P} a_k x(n - k)$$

and the error-signal

$$e(n) = x(n) - \hat{x}(n)$$

based on the LPC-coefficients $a_k$.

(2P) Subtask 1.3
There are 4 wav-files on our homepage which contains the phonems s, p, l and i.

1. Perform an LPC-analysis on these files by setting $P = 4$ and using the first 512 samples of each signal.
   Generate a plot of $x$, $\hat{x}$ and $e$ for each phonem. (4 plots)

2. Explain the different results for the different phonems.

---

Questions: alikanso91@hotmail.com and dhhjx880713@163.com
(1P) Subtask 1.4
In this part you will analyse the influence of the LPC-length on the estimation error.

1. Create a plot for each phonem, which shows the squared error \( e(n)^2 \) versus the LPC-length \( P = [1, \cdots, 200] \).

2. Explain your results.

(6P) Exercise 2
In this exercise, we ask you to implement Levinson-Durbin recursion.

(1P) Subtask 2.1
State the goal and preconditions of Levinson-Durbin recursion. What is the running time of the Levinson-Durbin algorithm explored in the lecture?

(1P) Subtask 2.2
State the four Levinson-Durbin formulae for \( K_P, a_P, a_i, \) and \( E_P \) that were given in the lecture.

(3P) Subtask 2.3
Implement Levinson-Durbin recursion in Matlab.

(1P) Subtask 2.4
To verify the correctness of your algorithm, compute the LPC coefficients for a rectangular signal of width 6, i.e.:

\[
x[n] = \begin{cases} 1, & n = 0...M - 1 \\ 0, & otherwise \end{cases}
\]

with \( M = 6 \) and \( P = 4 \).

The matrix form of the modified Yule-Walker equation is then:

\[
\begin{pmatrix}
6 & 5 & 4 & 3 \\
5 & 6 & 5 & 4 \\
4 & 5 & 6 & 5 \\
3 & 4 & 5 & 6
\end{pmatrix}
\begin{pmatrix}
a_1 \\
a_2 \\
a_3 \\
a_4
\end{pmatrix}
= 
\begin{pmatrix}
5 \\
4 \\
3 \\
2
\end{pmatrix}
\]

(4)

Compare the results of your algorithm with those obtained in the lecture. Alternatively, you can also use results obtained with the Matlab function \texttt{levinson} for comparison. Read the Matlab reference manual to find out how to use \texttt{levinson}.

Questions: alikanso91@hotmail.com and dhhjx880713@163.com