Serie 4

1.

a) Finden Sie ein Algorithmus von der IDFT für die reelle Folge (vgl. Aufgabe 4, Serie 3).

b) Implementieren Sie die IDFT für die reelle Folge aus der Aufgabe 6 in der Serie 2.

2.

Beweisen Sie die folgenden Probleme:

a) Sei $f : \mathbb{R} \to \mathbb{C}$ eine 2π-periodische Funktion in $\mathcal{C}^1([0, 2\pi])$ mit der Reihe $\sum_{k\in\mathbb{Z}} \hat{f}(k)$, die gleichmässig konvergiert. Dann hat man

$$|f(x) - p_N(x)| \leq 2 \sum_{|k|\geq N/2}{}' |\hat{f}(k)|$$

für das trigonometrische Polynom $p_N(x)$.

b) Falls die maximale Frequenz von der Funktion f M ist, dann ist das trigonometrische Polynom $p_N(x)$ die exakte Lösung für $N > 2M$.

3.

a) Schreiben Sie 2 Matlab-Funktionen `myfft` und `myifft`, um die **FFT** bzw. **IFFT** zu berechnen.

b) Als *Testbeispiel* nehmen Sie die folgenden Funktionen $f_1(x) = \sin(10x)$ bzw. $f_2(x) = \cos(60x)$ mit $N = 8, 16, 32, 64, 128,$ und 256.

c) Plotten Sie das Amplitudenspektrum mit Hilfe der **FFT**.

4.

Laden Sie die Daten `filtre.data` auf der Webseite der Vorlesung herunter. Mit Hilfe von den Matlab-Befehlen `fopen` und `fscanf` plotten Sie die Daten. Benutzen Sie Ihr Programm `myfft`, um die kleinen Frequenzen zu löschen.

5.

Laden Sie die Daten `haendel.data` auf der Webseite der Vorlesung herunter. Benutzen Sie die Matlab-Funktion `fft`, um mit den Frequenzen zu spielen; man kann z.B. die Frequenzen löschen, die kleiner als 98 sind.
Mit dem Matlab-Befehl `wavwrite` kann man ein `-.wav` - File herstellen. Benutzen Sie z.B. ein Programm *playwave* auf der Konsole, um mit dem File zu spielen.