

2. Übungsblatt zu Algorithmen der Numerischen Mathematik

Aufgabe 4: Sei f stetig und 2π -periodisch mit absolut summierbaren Fourierkoeffizienten $(\widehat{f}(n))_{n \in \mathbb{Z}}$.
Deren Approximation durch die Mittelpunktsregel lautet

$$\widetilde{f}_N(n) = \frac{1}{N} \sum_{j=0}^{N-1} f(t_j) e^{-int_j} \quad \text{mit} \quad t_j = \frac{2j+1}{2} \cdot \frac{2\pi}{N}.$$

Zeigen Sie die Aliasing-Formel

$$\widetilde{f}_N(n) = \sum_{l=-\infty}^{\infty} (-1)^l \widehat{f}(n + lN).$$

Aufgabe 5: (Sortierproblem)

Eine Datei von $N = 2^L$ Namen soll alphabetisch geordnet werden. Geben Sie einen Algorithmus an, der dies in $\mathcal{O}(N \log N)$ Operationen durchführt. Wir gehen dabei davon aus, dass Namen in $\mathcal{O}(1)$ verglichen werden können.

Hinweis: Divide et impera!

Programmieraufgabe 1: Implementieren Sie die schnelle Fourier-Transformation (ohne Verwendung von `fft` und `ifft`). Sie dürfen annehmen, dass die Länge des Eingabevektors eine Zweierpotenz ist.

Hinweis: Implementieren Sie die schnelle Fourier-Transformation rekursiv (d.h. Ihre Funktion ruft sich selbst wieder auf).

Programmieraufgabe 2: Die zweidimensionale diskrete Fouriertransformation $\mathcal{F}(X) = \hat{X}$ von $X = (x_{m,n}) \in \mathbb{C}^{M \times N}$ ist durch

$$\hat{x}_{k,l} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x_{m,n} w_M^{mk} w_N^{nl}$$

und die Faltung $X * Y \in \mathbb{C}^{M \times N}$ durch

$$(X * Y)_{k,l} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x_{m,n} y_{k-m, l-n}$$

($k = 0, \dots, M-1, l = 0, \dots, N-1$) gegeben. Wie im eindimensionalen Fall gilt

$$X * Y = \mathcal{F}^{-1}(\mathcal{F}(X) \cdot \mathcal{F}(Y)),$$

wobei die Multiplikation auf der rechten Seite wiederum komponentenweise zu verstehen ist und

$$\mathcal{F}^{-1}(X) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x_{m,n} w_M^{-mk} w_N^{-nl}$$

gilt. Realisieren Sie die zweidimensionale FFT, indem Sie Ihren Code aus PA1 (oder die Matlab-Funktionen `fft` bzw. `ifft`) zunächst auf die Zeilen und dann auf die Spalten von X anwenden und berechnen Sie effizient die Faltung der Matrix `double(imread('Ausgangsmatrix.jpg'))` und der Matrix, welche in der Datei `Faltungsmatrix.asc` gespeichert ist. Das Ergebnis C läßt sich mit dem Befehl `imwrite(C/255, 'Ergebnismatrix.jpg')`; wieder als Bilddatei speichern. Die Bilddatei und die Faltungsmatrix finden Sie auf der Übungshomepage.

wichtiger Hinweis: Die in Matlab eingebauten Funktionen `fft` und `ifft` weichen etwas von der in der Vorlesung verwendeten Konvention ab. Informieren Sie sich hierüber vor Verwendung der eingebauten Funktionen in der ausführlichen Product-Help!