LAK-ComputerMathematik

Projekt **GJ-2**

Mandelbrot-Fraktale

Stand: 19. Oktober 2026, 13:36

Konsultationen zum Projekt: nach Terminvereinbarung (E-mail)

Mandelbrot-Fraktale:

Das Ziel dieses Projekt ist eine Visualisierung der Mandelbrot-Menge bzw. Fraktale zu erzeugen. Dies erfolgt durch die Berechnung der Folge $(z_n)_{n\in\mathbb{N}} \subset \mathbb{C}$, die wie folgt definiert ist:

$$\begin{cases} z_0 = 0\,, \\ z_{n+1} = z_n^2 + c\,, \end{cases}$$

für verschiedene Werte der *komplexe* Zahl $c \in \mathbb{C}$. Die Mandelbrot-Menge M ist die Menge der Zahlen c, für welche die Folge $(z_n)_{n\in\mathbb{N}}$ begrenzt ist.

1. Schreiben Sie eine Funktion `iterate` mit Parametern `cfix` und `maxiter`, die die Zahl

   $$\min\{n \in \mathbb{N} : n > \texttt{maxiter} \vee |z_n| > 1000\}$$

   berechnet und diese zurückgibt, wobei die Folge $(z_n)_n$ mit $c = \texttt{cfix}$ zu berechnen ist.

2. Schreiben Sie eine Funktion `build_grid` mit folgenden Parametern: `min_real, max_real, min_imag, max_imag, n_real, n_imag` die ein regelmäßiges Gitter des Rechteckes liefert, *i.e.* eine `n_imag` $\times$ `n_real` Matrix G, wobei

   $$g_{k\ell} = (\texttt{min_real} + (\texttt{max_real} - \texttt{min_real})\ell) + (\texttt{min_imag} + (\texttt{max_imag} - \texttt{min_imag})k)\ i\,.$$

   Achtung: hier bezeichnet i die komplexe Zahl! Diese kann man mit `1i` in MATLAB eingeben. `build_grid` kann man entweder mit For-Schleifen oder mithilfe der Funktionen `linspace` und `repmat` geschrieben werden.

3. In der Hauptdatei:

   - Definieren Sie die Konstanten `maxiter, min_real, ...` und bauen Sie das entsprechende Gitter G mithilfe von `build_grid` auf.
   - Berechnen Sie die Matrix R (so groß wie G, hier sind `zeros` und `size` hilfreich), die durch $r_{k\ell} := \texttt{iterate}(g_{k\ell}, \texttt{maxiter})$ definiert ist.
   - Visualisieren Sie die Matrix R mithilfe der Funktion `imshow` von MATLAB für verschiedene Bereiche der komplexen Ebene. Eine Skalierung durch `imshow`s 2. Argument (der Form `[low high]`) wird hier notwendig. Siehe die `Colormap` Option für ein bunteres Ergebnis.
     Ein Vorschlag dafür: `imshow(m, [0, maxiter], 'Colormap', jet)`