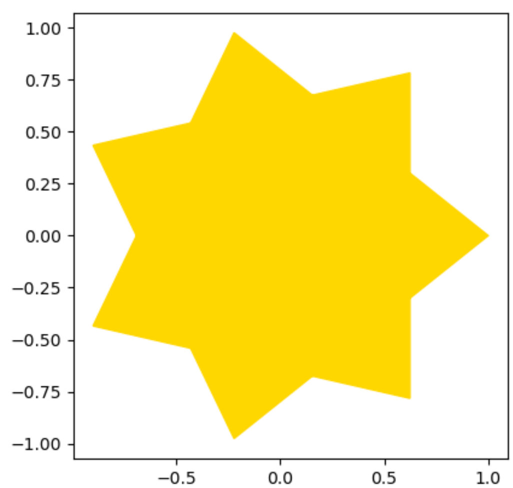
Computergestützte Mathematik zur Analysis – 7. Übungsblatt

Aufgabe 25:

Zeichnen Sie den folgenden regelmäßigen fünfstrahligen Stern:



Aufgabe 26:

Sei $f : [a, b] \to \mathbb{R}$ eine Funktion, die auf dem abgeschlossenen Intervall $[a, b]$ stetig und auf dem offenen Intervall (a, b) differenzierbar ist. Nach dem Mittelwertsatz gibt es dann ein $x_0 \in (a, b)$, so dass

$$f'(x_0) = \frac{f(b) - f(a)}{b - a}$$

gilt. Schreiben Sie eine Python-Funktion `MWS`, welche die folgenden Schritte ausführt:

- Übergabe der Funktion f und zusätzlich der Intervallgrenzen $a, b \in \mathbb{R}$ als `tuple`.
- Berechnung von $x_0 \in (a, b)$ und der Tangente g an den Graphen der Funktion im Punkt x_0.
- Zeichnen der Funktion f und der Tangente g über dem Intervall $[a, b]$.
- Rückgabe von x_0, der Tangente g und einem Plot p.

Wenden Sie Ihre Funktion auf $f(x) = 2x^2 - 4x + 1$ und das Intervall $[2, 7]$ an.

Allgemeiner Hinweis: `solveset` findet ggf. nicht alle Lösungen. Das ist für diese Aufgabe jedoch nicht relevant.

bitte wenden

Aufgabe 27:

Für eine natürliche Zahl n, $n \geq 1$, soll eine Liste oder ein Array aller Primzahlen bis einschließlich n erzeugt werden. Dazu kann man das Sieb des Eratosthenes verwenden.

(a) Recherchieren Sie den Algorithmus *Sieb des Eratosthenes / Sieve of Eratosthenes* und geben Sie zwei Quellen an.

(b) Auf der Homepage finden Sie zwei ineffiziente Implementierungen des Verfahrens, eine nutzt nur Python, die andere auch NumPy. Verstehen Sie was diese Implementierungen tun und was zurückgegeben wird.

(c) Kopieren Sie eine der Implementierungen, bennen Sie die Funktion um, und modifizieren Sie sie, sodass zusätzlich gezählt wird, wie oft eine Zahl als `False`, d.h. als zerlegbar, markiert wird.

(d) Implementieren Sie eine effizientere, schnellere Version des Verfahrens, bei der ein Teil der Mehrfachmarkierungen vermieden wird. Überlegen Sie sich dazu, ob man die Schleife(n) auch später beginnen lassen kann und ob man gegebenenfalls die Schleife(n) früher abbrechen kann.

Sie können für diese Aufgaben gerne ein KI Werkzeug verwenden.

Aufgabe 28:

(a) Schreiben Sie eine PYTHON-Funktion `ggTpoly`, welche den größten gemeinsamen Teiler zweier Polynome $p, q \in \mathbb{Q}[X]$ berechnet. Verwenden Sie dabei den Euklidischen Algorithmus und verwenden Sie nicht die modulo-Funktion `%`. Normalisieren Sie das Resultat, das heißt der führende Koeffizient soll 1 sein.

(b) Berechnen Sie ggTpoly$(x^6 - 3x^4 - 2x^2 + 8, x^5 + 3x^4 - x^3 - 7x^2 - 2x + 2)$ und ggTpoly$(x^5 + x^3 + 1, x^3 - 1)$.

Hinweis: Die Sympy-Funktion `gcd` darf hierbei nicht verwendet werden. Auf der Homepage finden Sie Implementierungen des Euklidischen Algorithmus für den Ring der ganzen Zahlen.

Aus den ersten sieben Blättern sollten Sie jetzt 22 Punkte für die Zulassung zur Klausur haben.

Besprechung in den Übungen vom 03.-05. Dezember 2025.