



QUANTUM  
CODING

# Die Python-Bibliothek SymPy

# Inhalt des Vortrags



1. Was ist Computeralgebra?
2. Das Computeralgebrasystem SymPy
3. Drei kleine Beispiele
4. Kurvendiskussion
5. Fazit

# Was ist Computeralgebra?

The screenshot shows the German Wikipedia page for 'Computeralgebra'. The browser address bar displays 'https://de.wikipedia.org/wiki/Computeralgebra'. The page features the standard Wikipedia layout with a left sidebar containing navigation links like 'Hauptseite', 'Themenportale', and 'Zufälliger Artikel'. The main content area includes a 'Wikipedia Loves Folklore' banner, the article title 'Computeralgebra', and a brief introductory paragraph: 'Die **Computeralgebra** ist das Teilgebiet der **Mathematik** und **Informatik**, das sich mit der automatisierten **symbolischen Manipulation algebraischer** Ausdrücke beschäftigt.' Below this is a table of contents with sections: 1 Überblick, 2 Zugrundeliegende Strukturen (with sub-sections 2.1 Gruppen, 2.2 Berechenbare Ringe, 2.3 Beispiele, 2.4 Formale Objekte), 3 Anwendungen, 4 Komplexitätsbetrachtungen (with sub-sections 4.1 Effiziente exakte Arithmetik mit ganzen Zahlen, 4.2 Effiziente exakte Arithmetik mit rationalen Zahlen, 4.3 Effiziente exakte Arithmetik mit Polynomen in  $\mathbb{Q}[x]$ ), 5 Siehe auch, 6 Literatur, and 7 Weblinks. The 'Überblick' section is currently expanded, showing the introductory text: 'Hauptziel der Computeralgebra ist es, durch konservative Rechnungen algebraische Ausdrücke umzuformen und eine möglichst kompakte Darstellung zu erzielen. Die Wertigkeit und Präzision der Gleichung wird dabei nicht angefasst. Rundungen oder Näherungen werden nicht zugelassen. Eine Nebenbedingung ist hierbei die verwendeten Algorithmen und Methoden effizient zu gestalten.'





# Das Computeralgebrasystem SymPy

- Voraussetzung: Python (unter Windows z. B. Anaconda)
- `pip install sympy`
- Features: <https://www.sympy.org/en/features.html>
  - Kernfunktionalitäten
  - Rechnen mit Polynomen
  - Ableitungen, Integrale, Grenzwerte
  - Gleichungslöser
  - ...

# Drei kleine Beispiele

- Aus dem Abstract:

## Beispiele für mit SymPy lösbare Aufgaben

- Löse nach  $x$  auf:  $(x - 1)(3x + \frac{5}{6}) - (2x - \frac{2}{3})(\frac{1}{2} + 1,5x) - \frac{7}{6} = 0$
- Löse nach  $v$  auf:  $r = a \frac{c+v}{c-v}$
- Bestimme die Lösungen  $(x, y, z) \in \mathbb{R}^3$  für  $x^2 + y^2 + z^2 - 1 = 0$  und  $x^2 - y + z^2 = 0$  und  $x - z = 0$
- Ist 2023 eine Primzahl?
- Größter gemeinsamer Teiler von Polynomen

# Und was kann ChatGPT?

- $r = a * (c + v) / (c - v)$

# Kurvendiskussion

- $x$  ist Nullstelle von  $f(x)$  gdw.  $f(x) = 0$
- Bei  $x$  ist ein Minimum von  $f(x)$  gdw.  $f'(x) = 0$  und  $f''(x) > 0$  (linksgekrümmt)
- Bei  $x$  ist ein Maximum von  $f(x)$  gdw.  $f'(x) = 0$  und  $f''(x) < 0$  (rechtsgekrümmt)

Wir betrachten z. B.  $f(x) = a \cdot (x+4) \cdot x \cdot (x-4)$  mit  $a = 1 / \sqrt{27}$



# Fazit (1/2)

- Plus points of SymPy:

- Open-source and free
- Symbolic mathematics capabilities
- Integration with Python
- Extensive functionality
- Educational tool

- Minus points of SymPy:

- Performance limitations for certain computations
- Learning curve, especially for beginners
- Documentation may be limited



# Fazit (2/2)

- Overall, SymPy is a powerful and versatile tool for symbolic mathematics, offering a wide range of functionalities for mathematical computation and exploration. It is particularly well-suited for tasks that involve symbolic manipulation and analysis, as well as educational purposes. While SymPy may have some limitations in terms of performance and learning curve, its open-source nature, integration with Python, and extensive functionality make it a valuable asset for mathematicians, scientists, engineers, educators, and anyone else interested in mathematical computing. With practice and familiarity, users can leverage SymPy to tackle a variety of mathematical problems and gain deeper insights into mathematical concepts.

(Khaled Saifullah, Data Scientist of Quantum Coding)