

Symbolisches Rechnen

Ein symbolischer Ausdruck in MuPAD kann unbestimmte Größen (Bezeichner oder Variablen) enthalten, mit denen gerechnet werden kann.

Lösen von Gleichungen

- `solve(2*x + y = 3, x);` löst die Gleichung $2x + y = 3$ nach x auf
- `solve(2*x + y = 3, y);` löst die Gleichung $2x + y = 3$ nach y auf

Da es umständlich ist, größere Gleichungen jeweils neu einzugeben (Nutzen Sie hier das Kopieren!), kann man die Gleichung auch erst definieren:

- `Gleichung := 2 * x + y = 3;` Dem Bezeichner „Gleichung“ wird $2x + y = 3$ zugewiesen.
- `Gleichung;` liefert die Gleichung $2x + y = 3$
- `solve(Gleichung, x), solve(Gleichung, y);` liefert die schon bekannten Auflösungen

Enthält eine Gleichung nur eine Unbekannte, so kann auf die Angabe der Variablen, nach der die Gleichung aufzulösen ist, verzichtet werden:

- `solve(2 * x^2 + 3 * x - 25 = 0);` liefert die Nullstellen von $y = 2x^2 + 3x - 25$

Dagegen muss man bei mehreren Unbekannten die Variable, nach der aufgelöst werden soll, angeben:

- `solve(x^2 + p*x + q = 0, x);` liefert die Nullstellen von $y = x^2 + px + q$

Will man die numerischen Näherungswerte wissen, so nutze man den schon bekannten Befehl `float`:

- `float(solve(2 * x^2 + 3 * x - 25 = 0));` Nullstellen von $y = 2x^2 + 3x - 25$ numerisch

MuPAD stören auch komplexe Lösungen nicht:

- `solve(x^2 - 2 * x + 2 = 0);`

Wie sieht es mit Polynomen höheren Grades aus?

- `solve(x^3 - 8 * x^2 + x + 42 = 0);`
- `solve(x^4 + 12 * x^3 - 159 * x^2 + 62 * x + 840 = 0);`
- `solve(x^5 - 41 * x^4 - 795 * x^3 - 8489 * x^2 - 2446 * x - 44520 = 0);`

Für Polynome 3. Grades gibt es noch komplizierte Lösungsformeln für die Berechnung der Nullstellen. Ab 4. Grades muss die Mathematik im allgemeinen Fall passen. Trotzdem findet MuPAD für viele Gleichungen noch exakte Lösungen. Die Gleichung 5. Grades kann MuPAD nicht mehr exakt lösen. Es gibt jedoch Näherungsverfahren, die weiterhelfen.

Lösen von Gleichungssystemen

- `Gleichungen1:={x + y = a, x - a * y = b};` Dem Bezeichner „*Gleichungen1*“ werden $x + y = a$ und $x - ay = b$ zugewiesen.
- `solve(Gleichungen1, {x, y});` löst das Gleichungssystem nach x und y auf

Benötigt man die Gleichungen später nicht mehr, genügt es, die Gleichungen direkt bei `solve` anzugeben:

- `solve({x + y = 2, x - 2 * y = 3}, {x, y});` liefert $x = \frac{7}{3}$ und $y = -\frac{1}{3}$ als Lösungen
- `Gleichungen2 := {x^2 + y = 1, x - y = 2}; solve(Gleichungen2, {x, y});`

Liefert die beiden Lösungen $\left(y = x - 2; x = \frac{1}{2}\sqrt{13} - \frac{1}{2} \right)$ und $\left(y = x - 2; x = -\frac{1}{2}\sqrt{13} - \frac{1}{2} \right)$.

Das letzte Beispiel befriedigt natürlich noch nicht, da y jeweils noch nicht vollständig berechnet wurde, also noch von der Variablen x abhängig ist. Deshalb übergibt man die Option `BackSubstitution = TRUE`, damit MuPAD die Lösungen ineinander einsetzt:

- `Loesungen := solve(Gleichungen2, {x, y}, BackSubstitution = TRUE);` Beachte Loesungen!

Nun wird die Lösung für x auch sofort in y eingesetzt und MuPAD liefert $\left(y = \frac{1}{2}\sqrt{13} - \frac{5}{2}; x = \frac{1}{2}\sqrt{13} - \frac{1}{2} \right)$

und $\left(y = -\frac{1}{2}\sqrt{13} - \frac{5}{2}; x = -\frac{1}{2}\sqrt{13} - \frac{1}{2} \right)$.

Überprüfen wir die Lösungen! Dazu müssen wir die Ergebnisse in die Gleichungen $x^2 + y = 1$ und $x - y = 2$ einsetzen. Fangen wir mit der ersten Lösung $\left(y = \frac{1}{2}\sqrt{13} - \frac{5}{2}; x = \frac{1}{2}\sqrt{13} - \frac{1}{2}\right)$ an!

- **subs(Gleichungen2, op(Loesungen, 1));** (**op(Loesungen, 1)** greift auf die 1. Lösung zu.)

setzt in den Gleichungen2 die 1. Lösung ein und liefert $2 = 2$ und $\left(-\frac{1}{2}\sqrt{13} - \frac{1}{2}\right)^2 - \frac{1}{2}\sqrt{13} - \frac{5}{2} = 1$.

Die erste Bestätigung ist zufriedenstellend.

Die Probe in der 2. Gleichung sollte MuPAD aber auch alleine auflösen.

- **simplify(subs(Gleichungen2, op(Loesungen, 1)));** (**simplify** vereinfacht Ausdrücke)

Wir erhalten $1 = 1$ und $2 = 2$, was die Gültigkeit der 1. Lösung bestätigt.

Nun noch die Überprüfung der 2. Lösung:

- **simplify(subs(Gleichungen2, op(Loesungen, 2)));**

Will man die numerischen Näherungslösungen wissen, so nutze man wieder float. Die Operation float allein würde jedoch nur die symbolische Lösung von **solve** weiterverarbeiten, was bei mehreren Lösungen MuPAD Schwierigkeiten bereitet. Mit der Operation **hold** wird die symbolische Rechnung unterdrückt und ausschließlich numerisch gerechnet:

- **float(hold(solve)(Gleichungen2, {x, y}));**

liefert $(y = -4,3\dots; x = -2,3\dots)$ und $(y = -0,697\dots; x = 1,302\dots)$

Aufgaben:

1. Berechnen Sie die Nullstellen der Gleichung $y = 5x^2 + 4x - 32$ exakt und numerisch.
2. Lösen Sie das folgende Gleichungssystem allgemein:

$$a + b + c + d + e = 1$$

$$a + 2b + 3c + 4d + 5e = 2$$

$$a - 2b - 3c - 4d - 5e = 2$$

$$a - b - c - d - e = 3$$

(Die Lösung enthält freie Parameter. Wie viele?)

Entscheiden sie selbst, wie viel Übung Sie benötigen.

3. Lösen Sie S. 63/A151 a, b, d exakt und geben Sie für b) eine Näherungslösung an.
4. Lösen Sie S. 63/A152 exakt und ermitteln Sie Näherungslösungen (Variieren Sie auch die Stellenzahl der Genauigkeit.).
5. Lösen Sie S. 64/A159 exakt und ermitteln Sie Näherungslösungen.

Speichern Sie die Arbeit unter **Gleichungen.txt** ab.