

TI Voyage 200 - Befehlsreferenz

1 Basics

1.1 Tipps / Tricks

<i>ON</i>	Bricht die aktuelle Aktion ab.
$\diamond + G$	Aktiviert griechisches Alphabet. (für das nächste Zeichen) $\diamond + G \rightarrow D = \delta$ $\diamond + G \rightarrow \uparrow + D = \Delta$

1.2 Basic Funtions

<i>solve</i> ($x + 1 = 5, x$)	Löse die Gleichung nach x
<i>solve</i> ($x + 1 = y$ and $y + 2 = x, \{x, y\}$)	Löse das Gleichungssystem nach x und y
<i>abs</i> (a)	Betrag $ a $ einer (komplexen) Zahl a
<i>root</i> (a, n)	$\sqrt[n]{a}$ bzw. \sqrt{a} , falls n weggelassen wird.
<i>limit</i> ($n + 1, n, \infty$)	$\lim_{n \rightarrow \infty} (n + 1)$
$\sum(n + 1, n, a, b)$	$\sum_{n=a}^b n + 1$
$\int(x + 1, x, a, b)$	$\int_a^b x + 1 dx$ (a und b optional)
$\delta(x + 1, x, n)$	$\frac{\delta^n}{\delta x^n} x + 1$
<i>expand</i> (($x + 1$)($x + 2$))	Multipliziert den Term aus. Führt auch PBZ durch.
<i>factor</i> ($x^2 + x$)	Zerlegt den Term in Faktoren.
<i>factor</i> ($x^2 + x, x$)	Faktorisiert den Term nach der Variable x
<i>arcLen</i> (<i>cos</i> (x), x, a, b)	Gibt die Bogenlänge der Funktion zwischen a und b and.
<i>gcd</i> (a, b)	Gibt den grössten gemeinsamen Teiler (ggT) von a und b zurück.
<i>lcm</i> (a, b)	Gibt das kleinste gemeinsame Vielfache (kgV) von a und b zurück.

1.3 Advanced Functions

<i>deSolve</i> ($y'' + 2 * y' = 2 * x, x, y$)	Löst die DGL (1. oder 2. Ordnung) und gibt eine allgemeine
<i>deSolve</i> ($y' = x$ and $y(0) = 0, x, y$)	Lösung aus. Werden Anfangs-/Randbedingungen angegeben wird eine
<i>deSolve</i> ($y'' + y' + y = \sin(x)$ and $y(0) = 0$ and $y'(0) = 1, x, y$)	spezielle Lösung ausgegeben. Generelle Eingabe: <i>deSolve</i> (DGL and Anfangsbedingung ,unabhängige Variable, abhängige Variable)
<i>impDif</i> ($x^2 + y^2 = 100, x, y$)	Berechnet die implizite Ableitung der Gleichung, wenn eine Variable implizit durch die Andere gegeben ist. Resultat: $-x/y$
<i>nDeriv</i> ($x^2, x, [h]$)	Berechnet die numerische Ableitung nach x . Der optionale Parameter h gibt die Schrittweite an. Wenn statt x^2 eine Liste oder Matrix verwendet wird, wird die Ableitung über entsprechenden Werte gebildet.
<i>fMax</i> ($-(x - a)^2, x$)	Gibt Werte für x an, so dass der Term maximal wird.
<i>fMax</i> ($-(x - a)^2, x x > 3$)	... mit eingeschränktem Lösungsintervall
<i>fMin</i> ($(x - a)^2, x$)	Gibt Werte für x an, so dass der Term minimal wird.
<i>exp</i> ► <i>list</i> ($x = 2$ or $x = 1, x$)	Gibt durch <i>or</i> getrennte Werte als Liste zurück ($\{2, 1\}$)

2 Zahlensysteme

Der TI Voyage kennt folgende Zahlensysteme und Umrechnungsfunktionen:

... ► *bin* 0b... Binärsystem

... ► *hex* 0h... Hexadezimalsystem

... ► *dez* ... Dezimalsystem

Unter *MODE* ► *BASE* wird das Standard-Zahlensystem festgelegt. Hinweis: Nur die Ausgabe wird verändert. Die Eingabe muss weiterhin mit z.B. 0b... erfolgen.

3 Vektoren / Matrizen

Vektoren und Matrizen werden im TI Voyage 200 folgendermassen eingegeben:

$[a, b]$	$\begin{bmatrix} a & b \end{bmatrix}$	Zeilenvektoren
$\begin{bmatrix} a \\ b \end{bmatrix}$	$\begin{bmatrix} a \\ b \end{bmatrix}$	Spaltenvektoren
$\begin{bmatrix} a, b, c, d \\ [a, b][c, d] \end{bmatrix}$	$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$	Matrizen

3.1 Vektor-Funktionen

$crossP(\vec{a}, \vec{b})$	Kreuzprodukt $\vec{a} \times \vec{b}$
$dotP(\vec{a}, \vec{b})$	Skalarprodukt $\vec{a} \circ \vec{b}$

3.2 Matrix-Funktionen

$det(A)$	Determinante der Matrix A
$rref(A)$	Gibt die reduzierte Zeilenstufenform der Matrix an. $\begin{bmatrix} 1 & 0 & a \\ 0 & 1 & b \end{bmatrix}$
$eigVc(A)$	Ergibt eine Matrix, welche die Eigenvektoren der Matrix A enthält.
$eigVl(A)$	Gibt eine Liste der Eigenwerte der Matrix A zurück.
$identity(n)$	Gibt eine Einheitsmatrix der Dimension n zurück.
$diag(a)$	Erzeugt eine Matrix mit den Werten der Liste / des Vektors a in der Diagonale
$max(A)$	Gibt einen Zeilenvektor zurück, der das Maximum jeder Spalte enthält.
$min(A)$	Gibt einen Zeilenvektor zurück, der das Minimum jeder Spalte enthält.

3.3 Weitere Funktionen

$list \blacktriangleright mat(\{...\})$	Gibt einen Zeilenvektor mit den Elementen der Liste zurück.
$list \blacktriangleright mat(\{...\}, a)$	Gibt eine Matrix mit a Elementen pro Zeile zurück.
$mat \blacktriangleright list([...])$	Gibt eine Liste mit dem Inhalt der Matrix zurück (Zeile für Zeile).
$[x, y, z] \blacktriangleright cylind$	Gibt den dreidimensionalen Zeilen- oder Spaltenvektor in der Form $[r, \angle\theta, z]$ zurück.

4 Komplexe Zahlen

Komplexe Zahlen können im TI Voyage 200 in der Form $a + bi$ (Rectangular) oder $r \angle \phi$ (Polar) geschrieben werden. Unter **MODE** ► **Complex Format** kann der Standard-Modus ausgewählt werden. Der Modus *Real* zeigt nur komplexe Werte an, wenn auch die Eingabe komplex war. Die Eingabe und Umrechnung geschieht folgendermassen:

$25 + 3 * i$ | ... ► *Rect*

$(2 \angle 30)$ | ... ► *Polar*

4.1 Funktionen

$cFactor(x^2 + a^2, x)$	Komplexe Faktorzerlegung nach x
$cSolve(x^2 + x + 1, x)$	Lösen der komplexen Gleichung nach x
$cSolve(x = 2 * y \text{ and } y^2 = -1)$	Lösen komplexer Gleichungssysteme nach x und y
$cZeros(x^2 + 1, x)$	Bestimmen der (komplexen) Nullstellen
$conj(z)$	Konjugiert-komplexe Zahl \bar{z}
$abs(z)$	Betrag $ z $
$angle(z)$	Winkel $\arg(z)$
$real(z)$	Realteil $\Re(z)$
$imag(z)$	Imaginärteil $\Im(z)$

4.2 Umrechnungen

$P \blacktriangleright Rx(r, \theta)$	Gibt die X -Koordinate des Paares (r, ϕ) zurück.
$P \blacktriangleright Rx(\{r1, r2\}, \{\theta1, \theta2\})$	Funktioniert auch für Listen...
$P \blacktriangleright Rx([r1, r2; r3, r4], [\theta1, \theta2; \theta3, \theta4])$.. und Vektoren / Matrizen
$P \blacktriangleright Ry(r, \phi)$	Gibt die Y -Koordinate des Paares (r, ϕ) zurück.
$R \blacktriangleright Pr(x, y)$	Gibt die r -Koordinate des Paares (x, y) zurück.
$R \blacktriangleright P\theta(x, y)$	Gibt die θ -Koordinate des Paares (x, y) zurück.

5 Statistik / Wahrscheinlichkeit

5.1 Funktionen

$mean(\{...\})$	Berechnet das arithmetische Mittel der Elemente der Liste.
$mean(\{...\}, \{...\})$	Mit einer zweiten Liste lassen sich die Elemente einzeln gewichten.
$mean(A)$	Gibt einen Zeilenvektor mit den arith. Mitteln der Spalten zurück.
$mean(A, B)$	Mit einer Matrix B lassen sich die Elemente von A gewichten.
$median(\{...\})$	Berechnet den Median der Elemente der Liste.
$median(A)$	Gibt einen Zeilenvektor mit den Medianwerten der Spalten zurück.
$nCr(n, k)$	Binominalkoeffizient $\binom{n}{k}$ - funktioniert auch für Listen und Matrizen
$nPr(n, k)$	Anzahl Möglichkeiten unter Berücksichtigung der Reihenfolge k Elemente aus n auszuwählen.
$OneVarL1, [L2], [L3], [L4]$	Berechnet die Statistiken der Liste L1. Die Statistik wird mit <i>ShowStat</i> eingeblendet. Folgende Werte werden berechnet: \bar{x} , $\sum x$, $\sum x^2$, σx , ... Optionale Listen: L2: Häufigkeit, L3: Klassencodes, L4: Klassenliste

5.2 Regression

Zur Berechnung einer Regression muss eine Liste ($\{\dots\}$) die x-Werte enthalten und eine zweite Liste die y-Werte. Der Befehl *LinReg* $L1, L2$ berechnet die lineare Regression. Mit *ShowStat* werden die berechneten Werte angezeigt. Es ist auch möglich, die Datenpunkte und die Regressionskurve zu plotten: *Regeq*(x) $\rightarrow y1(x)$ und *NewPlot* 1, 1, $L1, L2$. Optional können weitere Listen angegeben werden: L3: Häufigkeit, L4: Klassencodes, L5: Klassenliste, wobei alle Listen ausser L5 die gleiche Dimension besitzen müssen. 'Iterationen' gibt die maximale Anzahl Lösungsversuche an. (standardmäßig: 64)

Lineare Regression	<i>LinReg</i> $L1, L2, [L3], [L4, L5]$
Logarithmische Regression	<i>LnReg</i> $L1, L2, [L3], [L4, L5]$
Logistische Regression	<i>Logistic</i> $L1, L2, [Iterationen], [L3], [L4, L5]$
Potenz-Regression	<i>PowerReg</i> $L1, L2, [L3], [L4, L5]$
Quadratische Polynomische Regression	<i>QuadReg</i> $L1, L2, [L3], [L4, L5]$
Kubische Regression	<i>CubReg</i> $L1, L2, [L3], [L4, L5]$
Polynomische Regression 4-ter Ordnung	<i>QuartReg</i> $L1, L2, [L3], [L4, L5]$

5.3 Zufallszahlen

<i>RandSeed</i> 1147	Setzt die Ausgangsbasis (Seed) für den Zufallszahl-Generator
<i>rand</i> () <i>rand</i> (n)	Gibt eine Zufallszahl zwischen 0 und 1 zurück. Gibt eine Zufallszahl zwischen 0 und n (für n pos.) bzw. zwischen n und 0 (für n neg.) zurück.
<i>randMat</i> (n, m)	Erzeugt eine ganzzahlige Matrix mit n Zeilen und m Spalten mit Werten $-9 < x < +9$.
<i>randNorm</i> (a, sd)	Gibt eine reelle Zufallszahl um den Mittelwert a mit der Standardabweichung sd aus.
<i>randPoly</i> (x, n)	Erzeugt ein Polynom der Variable x der Ordnung n mit Koeffizienten $-9 < x < +9$