

Auswahl von Maple-Kommandos, Funktionen und vordefinierten Variablen

Stichworte alphabetisch; Befehle aus packages sind in einem geringen Ausmaß berücksichtigt.

abs	Absolutbetrag
add	numerische (endliche) Summation [vgl. <code>sum</code>]
algsubs	Variante von <code>subs</code> , unterstützt ersetzen von Teilausdrücken
alias	Festlegung alternativer Bezeichnungen
and	Boole'scher Operator: logisches und [vgl. <code>or</code> , <code>xor</code>]
arcsin	Arcus Sinus (auch: <code>arccos</code> , <code>arctan</code> , <code>arccot</code>)
arcsinh	(<i>nicht</i> : <code>arsinh</code>) Area Sinus hyperbolicus (auch: <code>arccosh</code> , <code>arctanh</code> , <code>arccoth</code>)
Array	Datentyp: Array (ein- oder mehrdimensionales Feld); auch: Konstruktor
[un]assign	Wertzuweisung an Variablen (Analogon zu <code>:=</code>); bzw. rückgängig machen
assume	Eigenschaft[en] von Variablen festlegen; auch: <code>assuming</code> , <code>additionally</code>
binomial	Binomialkoeffizient
break	Abbruch einer Schleife [vgl. <code>next</code>]
cat	Zusammenfügung von Strings oder Namen
[f]close	Schließen einer externen Datei [vgl. <code>[f]open</code>]
coeff[s]	Extrahieren von Koeffizienten bei Ausdrücken, z.B. Polynomen
collect	Ausdruck nach einer Variablen ordnen
conjugate	konjugiert komplexe Zahl [vgl. <code>Im</code> , <code>Re</code>]
convert	Typ- bzw. Darstellungsumwandlung (diverse Varianten)
copy	Kopieren von [r]tables (Duplikat in separatem Speicher)
cos	Cosinus
cosh	Cosinus hyperbolicus
cot	Cotangens
coth	Cotangens hyperbolicus
currentdir	Anzeige / Wechseln aktuelles Verzeichnis
[r]curry	Parameter einer Prozedur fixieren; definiert reduzierte Prozedur [vgl. <code>overload</code>]
D	Ableitungsoperator [vgl. <code>diff</code>]
[l]degree	[unterer] Grad eines Polynoms
denom	Nenner eines Bruches [vgl. <code>numer</code>]
diff	Ableitung eines Ausdruckes nach einer Variablen [vgl. <code>D</code>]
Diff	Ableitung, unausgewertet
Digits	Umgebungsvariable: Genauigkeit in Dezimalstellen (Gleitpunktarithmetik)
divide	Test auf Teilbarkeit bei Polynomen [vgl. <code>quo</code> , <code>rem</code>]
dsolve	Lösung von Differentialgleichung[ssystem]en
plots[display]	gemeinsame Anzeige zuvor generierter plot-Strukturen [vgl. <code>plot</code>]
error	Abbruch einer Prozedur, mit Fehlermeldung [vgl. <code>return</code>]
eval	Auswertung eines Ausdruckes (mit Einsetzen von Werten)
evalb	Boole'sche (logische) Auswertung
evalc	Auswertung komplexer Ausdrücke in der Form $x + iy$ [vgl. <code>polar</code>]
evalf	Auswertung in Gleitpunktarithmetik
evalhf	Auswertung in Hardware-Gleitpunktarithmetik (double precision)
exp	Exponentialfunktion
expand	Expandieren (Ausmultiplizieren) eines Ausdruckes [vgl. <code>factor</code>]
exprseq	Datentyp: expression sequence [vgl. <code>seq</code>]
factor	Faktorisieren eines Ausdruckes [vgl. <code>expand</code>]
FAIL	Konstante: logisch unentscheidbar
false	Konstante: logisch falsch
for ... do ... end do	Schleife (diverse syntaktische Varianten) [vgl. <code>break</code> , <code>next</code> , <code>while</code> ; <code>in</code>]
fsolve	Numerische Lösung von algebraischen Gleichung[ssystem]en [vgl. <code>solve</code>]

gamma GAMMA gcd global	Konstante: Euler-Mascheroni-Konstante γ Gamma-Funktion; beachte $n! = \Gamma(n + 1)$ für $n \in \mathbb{N}_0$ größter gemeinsamer Teiler [vgl. <code>lcm</code>] Deklaration globaler Variablen in Prozeduren [vgl. <code>local</code>]
has heap[...]	Test des Auftretens einer Variablen in einem Ausdruck Deklaration und Verwendung eines heap-Objektes (Haldenspeicher)
I if ... else ... elif ... end if ifactor Im in infinity int, Int interface intersect iquo irem is isprime; ithprime, nextprime	Konstante: imaginäre Einheit $i = \sqrt{-1}$ if-Konstrukt; kurz auch: <code>ifelse(...)</code> Primzahlfaktorisation Imaginärteil einer komplexen Zahl [vgl. <code>conjugate</code> , <code>Re</code>] Operator \in (für Mengen, Listen) [vgl. <code>subset</code>]; auch: <code>for ... in ... do ...</code> Konstante: ∞ [un]bestimmtes Integral, ausgewertet bzw. unausgewertet Einstellungen für user interface Operator: Durchschnitt von Mengen [vgl. <code>minus</code> , <code>union</code>] ganzahlige Division Rest bei ganzzahliger Division Test auf Eigenschaft Primzahltest; Ermittlung von Primzahlen
kernelopts	Einstellungen für kernel
lcm lhs, rhs limit, Limit list ln local log lprint	kleinstes gemeinsames Vielfaches [vgl. <code>gcd</code>] linke und rechte Seite einer Gleichung Limes (Grenzwert), ausgewertet bzw. unausgewertet Datentyp: Liste (lineare Anordnung); [<code>.</code> , <code>.</code> , <code>...</code>] natürlicher Logarithmus Deklaration lokaler Variablen in Prozeduren [vgl. <code>global</code>] allgemeiner Logarithmus Ausgabe in 'Schreibmaschinschrift' (nicht-proportional) [vgl. <code>print</code>]
max map, map2 Matrix member min minus mod mtaylor mul	Maximum Anwendung einer Prozedur auf mehrere Operanden Datentyp: Matrix (2-dimensionales Array); auch: Konstruktor [vgl. <code>Vector</code>] Test auf Enthaltensein in einer Datenstruktur [vgl. <code>in</code>] Minimum Operator: Mengendifferenz [vgl. <code>intersect</code> , <code>union</code>] Operator: modulo Multivariate Taylor-Entwicklung [vgl. <code>coeftayl</code> , <code>Order</code> , <code>series</code> , <code>taylor</code>] numerisches (endliches) Produkt [vgl. <code>product</code>]
next nops normal not NULL numelems numer	nächsten Schleifendurchlauf beginnen [vgl. <code>break</code>] Anzahl der Operanden eines Ausdruckes [vgl. <code>op</code>] Normalisierung eines rationalen Ausdruckes [vgl. <code>denom</code> , <code>numer</code> , <code>parfrac</code>] Boole'scher Operator: logische Verneinung Konstante: 'nichts' (leeres Objekt) Anzahl der Elemente einer Datenstruktur Zähler eines Bruches [vgl. <code>denom</code>]
op [f]open options or Order overload	Extraktion eines Operanden aus einem Ausdruck [vgl. <code>nops</code>] Öffnen einer externen Datei [vgl. <code>[f]close</code>] Deklaration von Optionen für Prozedur Boole'scher Operator: logisches oder [vgl. <code>and</code> , <code>xor</code>] Umgebungsvariable: Länge von Entwicklungen (<code>mtaylor</code> , <code>series</code> , <code>taylor</code>) Kapseln von Prozeduren in eine einzige (gleiche Argumentliste, verschiedene Typen) [vgl. <code>[r]curry</code>]

<code>parfrac</code>	Schlüsselwort für <code>convert</code> (Partialbruchzerlegung) [vgl. <code>normal</code>]
<code>parse</code>	Einlesen eines Maple-Ausdruckes aus einem String
<code>Pi</code>	Konstante: $\pi = 3.14159\dots$
<code>piecewise</code>	stückweise Funktionsdefinition
<code>plot</code>	Zeichnen von Funktionsgraphen [vgl. <code>display</code>] (viele weitere Funktionen, z.B. <code>plot3d</code> , <code>[plots]pointplot</code> , ...)
<code>plotsetup</code>	Initialisierung diverser Eigenschaften für <code>plots</code>
<code>polar</code>	Konstruktor für komplexe Zahl in Polarform $z = r e^{i\varphi}$
<code>proc ... end proc</code>	Deklaration einer Prozedur [vgl. <code>options</code>]
<code>print</code>	Ausgabe (unformatiert) [vgl. <code>lprint</code>]
<code>[f,s]printf</code>	Formatierte Ausgabe [vgl. <code>scanf</code>]
<code>printlevel</code>	Umgebungsvariable: spezifiziert Ausmaß der ausgegebenen Information bei Ausführung von Prozeduren
<code>product</code>	Produktbildung (symbolisch, auch unendliche Produkte) [vgl. <code>mul</code>]
<code>Product</code>	Produkt, unausgewertet
<code>queue[...]</code>	Deklaration und Verwendung eines <code>queue</code> -Objektes (Warteschlange)
<code>quo</code>	Polynomdivision [vgl. <code>rem</code>]
<code>rand; randomize</code>	Zufallszahlengenerator
<code>Re</code>	Realteil einer komplexen Zahl [vgl. <code>conjugate</code> , <code>Im</code>]
<code>read</code>	Einlesen Maple-Kommandos aus externer Datei [vgl. <code>save</code>]
<code>readdata</code>	Einlesen numerischer Daten aus externer Textdatei
<code>rem</code>	Rest bei Polynomdivision [vgl. <code>quo</code>]
<code>remove</code>	Entferne Objekte mit spezifizierter Eigenschaft aus Datenstruktur [vgl. <code>select</code>]
<code>restart</code>	Löschen des internen Speichers (Neustart)
<code>return</code>	Rückgabe von Werten innerhalb einer Prozedur [vgl. <code>error</code>]
<code>save</code>	Speicherung von Variablenzuweisungen in externer Datei [vgl. <code>read</code>]
<code>[f,s]scanf</code>	Formatiertes Lesen [vgl. <code>printf</code>]
<code>select</code>	Selektiere Objekte mit spezifizierter Eigenschaft aus Datenstruktur [vgl. <code>remove</code>]
<code>seq</code>	Konstruktor für <code>exprseq</code>
<code>series</code>	Allgemeine Reihenentwicklung [vgl. <code>Order</code> , <code>taylor</code>]
<code>set</code>	Datentyp: (endliche) Menge; { ... }
<code>sign; signum</code>	Signum-Funktion (reelles bzw. komplexes Vorzeichen)
<code>sin</code>	Sinus
<code>sinh</code>	Sinus hyperbolicus
<code>simplify</code>	Vereinfachen eines Ausdruckes [vgl. <code>collect</code> , <code>expand</code> , <code>factor</code>]
<code>solve</code>	exakte Lösung von algebraischen Gleichung[ssystem]en [vgl. <code>fsolve</code>]
<code>sort</code>	Sortierung
<code>sqrt</code>	Quadratwurzel
<code>stack[...]</code>	Deklaration und Verwendung eines <code>stack</code> -Objektes (Stapelspeicher)
<code>subset</code>	Operator: \subset (Teilmenge) [vgl. <code>in</code>]
<code>subs</code>	Substituieren eines Wertes für eine Variable in einem Ausdruck
<code>subsop</code>	Substituieren eines Operanden durch einen Ausdruck [vgl. <code>op</code>]
<code>sum</code>	[un]bestimmte Summe (symbolisch, auch unendliche Reihen) [vgl. <code>add</code>]
<code>Sum</code>	[un]bestimmte Summe, unausgewertet
<code>surd</code>	allgemeine Wurzel
<code>table</code>	Datentyp: <code>table</code> (assoziatives array); auch: Konstruktor
<code>tan</code>	Tangens
<code>tanh</code>	Tangens hyperbolicus
<code>taylor</code>	Taylor-Entwicklung [vgl. <code>coeftayl</code> , <code>Order</code> , <code>mtaylor</code> , <code>series</code>]
<code>time</code>	CPU Stoppuhr
<code>try</code>	Konstrukt zur kontrollierten Fehlerbehandlung
<code>true</code>	Konstante: logisch wahr
<code>type</code>	Typ-Überprüfung [vgl. <code>whattype</code>]

unapply	Umwandlung eines Ausdruckes in eine Funktion (inverse Operation zu apply)
union	Operator: Vereinigung von Mengen [vgl. intersect , minus]
uses	(in Prozeduren:) Spezifikation verwendeter packages [vgl. with]
Vector	Datentyp: Vector (1-dimensionales Array); auch: Konstruktor [vgl. Matrix]
whattype	Typ-Abfrage [vgl. type]
[un]with	Laden bzw. Deaktivieren eines packages [vgl. uses]
while	in Schleifen (do): Test auf Abbruch
xor	Boole'scher Operator: ausschließendes oder [vgl. and , or]
zip	Rekombination zweier Listen etc. zu einer einzigen, unter Anwendung einer elementweise wirkenden Operation

Sonderzeichen:

? ...	Hilfe aktivieren
# ...	Kommentar
'...'	'Maskieren' eines Ausdruckes (Auswertung verzögern)
'...'	Konstruktion eines Namens
"..."	Zeichenkette (string)
%, %, %%	die letzten drei berechneten Ergebnisse ('ditto')
%	wird auch in Formatstrings verwendet [vgl. printf , scanf]
\$	Operator (Konstruktor) für konstante expression sequence
->	Konstruktor für Funktionen
:=	Operator: Zuweisung
..	Operator: von - bis
_	(underscore:) Trennzeichen ('sichtbares Leerzeichen') in Namen
.	(Dezimal)-Punkt
	auch: Operator für nichtkommutative Multiplikation (z.B. Matrix mal Matrix)
\	'escape' - bzw. Fortsetzungszeichen
~	Postfix-Notation für elementweise Operation (z.B. +~)
	auch: Suffix bei Anzeige von Variablen mit zugewiesenen Eigenschaften (assume)
,	Begrenzer bei Aufzählungen (exprseq)
;	Trennzeichen für Kommandos
	auch: Trennzeichen für neue Zeile in Matrix << ... >>
:	Trennzeichen für Kommandos (unterdrückt Ausgabe)
	Trennzeichen für neue Spalte in Matrix << ... >>
()	Klammerung von Ausdrücken
[]	Begrenzer für Listen
	auch: Indizierung von Variablen, Listenelementen etc.
{ }	Begrenzer für Mengen
+, -, *, /	arithmetische Operatoren
!	Fakultät
^	Operator: Potenz
@	Operator: funktionale Komposition
@@	Operator: funktionale Potenz
&	Prefix für 'neutralen Operator'
=	Relation: gleich
<>	Relation: ungleich
<, <=	Relation: kleiner[gleich]
>, >=	Relation: größer[gleich]

In Verbindung mit der Online-Hilfe (? *command*) soll diese Liste dazu dienen, sich rasch über die Verwendung der wichtigsten Elemente der Kommandosprache von Maple zu orientieren. *W. Auzinger, K. Sturm (2019)*