

Mathematica commands summary (cheat sheet)

Hugo Touchette
 School of Mathematical Sciences, Queen Mary University of London
 Last updated: October 8, 2010

Abbreviations	
approx	Approximation
arg	Argument
char	Character
cnt	Counter
cond	Condition
dist	Distribution
elem	Element
eqn	Equation
eval	Evaluation
expr	Expression
fcn	Function
patt	Pattern
rand	Random
str	String
val	Value
var	Variable

Equalities, assignments [DC]

=	Assign value	
:=	Delayed eval	Also used for repeated eval
==	Equality	
=.	Clear value	Same as Clear
===	Boolean test	Same as SameQ
f[x_]:=f[x]=...	Remember computed vals	

Rules and replacements [DC, Tut]

patt->expr	Transformation rule	Can be assigned to names
patt:->expr	Delayed rule	Same as :>
expr/.rule	Replacement	Same as ReplaceAll
	$1+2x/.x->3$	Output: 7
expr/.{rule1,rule2}	Multiple replacements	
expr//.rule	Replace until no changes	Same as ReplaceRepeated

Applications, functions and maps [DC, Tut]

f@expr	Prefix form f[expr]	
expr//f	Postfix form f[expr]	
x~f~y	Infix form f[x,y]	
f@expr	Replaces the head of expr by f	Same as Apply[f,expr]
	$Plus@@\{1,2\}$	Output: $1+2=3$
	Similar to: $f[x]/.f->g$	Output: $g[x]$
f/@list	Map f on list	Same as Map[f,list]
f/@expr	Map f on parts of expr	
	$Map[f,a+b]$	Output: $f[a]+f[b]$
Function[x,body]	Function with arg x	Same as pure functions
	$Function[x,x^2][n]$	Output: n^2
	$Map[Function[x,x^2],a+b]$	Output: a^2+b^2
Function[{vars},...]	Many args	
body&	Pure (anonymous) function	
	$(\#^2\&)[5]$	Output: 25
	$Map[\#^2\&,a+b]$	Output: a^2+b^2
	$(1+g[\#])\&/@\{1,2\}$	Output: $\{1+g[1],1+g[2]\}$
#n	nth arg in pure function	
##	Seq of all args	

Formatting [DC]

FullForm	Display full head form
MatrixForm	Display in matrix form
Column	Display in column form
TableForm	Display in tabular form
TreeForm	Display in tree structure

Shortcuts [DC]

$\text{Ctrl}-@$	$\sqrt{\quad}$
$\text{Ctrl}-/$	Fraction
$\text{Ctrl}-^$	Superscript
$\text{Ctrl}-_$	Subscript
$\text{Ctrl}-\text{Enter}$	New row
$\text{Ctrl}-,$	New column

Patterns [DC, Tut]

-	Blank: any expr	
--	Double blank: any seq of expr	
---	Triple blank: zero or more args	
x_	Any expr named x	
x_h	Expr named x with head h	
x	Exact matching x	
	$x+2/.x->2$	Output: 4
x:patt	Named expr matching patt	
	$f[a^b]/.f[x:_^_]->p[x]$	Output: $p[a^b]$
x_:v	Optional arg with default val v	
x_h:v	Typed arg with optional val v	
f[n_]	f with any named arg n	
f[n_,n_]	f with identical args	
x^n_	x to any named power n	
x_^n_	Any expr to any power	
a_+b_	Sum of two named exprs	
{a1_,a2_}	List of two named exprs	
patt/;cond	Pattern with cond	Read: patt such that cond
	$(x_/.;NumberQ[x]\&\&Im[x]==0)$	Any real number
rule/;cond	Rule with cond	
patt?test	Pattern matching boolean test	
	$f[x_?NumberQ]:=x+2$	
(patt1 patt2 ...)	patt1 or patt2 or ...	
	$\{1,x,x^2,y^3\}/.(x x^n)->q$	Output: $\{1,q,q,y^3\}$
expr/.rule	Replacement in expr with rule	
	$x+2/.x->2$	Output: 4
	$f[a]+f[b]/.f[x_->x^2$	Output: a^2+b^2
	Position[f[a],g[b],f[b],f[x_]]	Output: $\{\{1\},\{2\}\}$
	$\{1,x,x^2,x^3\}/.x^n->r[n]$	Output: $\{1,x,r[2],r[3]\}$
Except[c]	Anything except c	
Except[c,patt]	Pattern patt except c	
	$Cases[\{1,0,2\},Except[0]]$	Output: $\{1,2\}$
	$Cases[\{1,x,0\},Except[0,_Integer]]$	Output: $\{1\}$

Type specification

Integer	\mathbb{N}	$\text{Esc}-\text{dsN}-\text{Esc}$
Real	\mathbb{R}	$\text{Esc}-\text{dsR}-\text{Esc}$
Complex	\mathbb{C}	$\text{Esc}-\text{dsC}-\text{Esc}$
Rational	\mathbb{Z}	$\text{Esc}-\text{dsZ}-\text{Esc}$
List	List type or head	
Symbol	Any symbol	
String	String type of head	
_Head	Type Head	
	$\text{MatchQ}[x^2,Power]$	

Files [DC]

Import["file"]	Import file or url
Export["file",expr]	Export expr in file
ImageResolution->pts	Image resolution in pts
ReadList["file"]	Returns list from file
<<File'	Load content of file

Useful functions

Sum[f, {i, imin, imax, di}]	Sum
Product[f, {i, imin, imax, di}]	Product
Mod[m, n]	Remainder of m/n
Solve[f[x]==0, x]	Solves polynomial eqn
Expand[p]	Expand polynomial p
Factor[p]	Factorize p
Length[expr]	Number of elems in expr
Numerator[expr]	Numerator of expr
Denominator[expr]	Denominator of expr
Options[cmd]	Display options for cmd
SetOptions[cmd]	Set options for cmd
Evaluate[expr]	Explicit eval
Piecewise[{{val, test}, ...}]	Piecewise definition
InverseFunction[f][x]	Inverse fct
Nest[f, x, n]	n -fold fct composition
NestList[f, x, n]	List of compositions
FixedPoint[f, x]	Iterate until fixed point
FixedPointList[f, x]	Iteration list
NestWhile[f, x, test]	While iteration
NestWhileList	While iteration list
LinearModelFit	Linear model
NonlinearModelFit	Nonlinear model
GraphPlot[m]	Graph (network) for adjacency m
GraphData["name"]	Graph with name
FinancialData["tag", "date"]	Financial data from date
CountryData["tag", "prop"]	Country data
WeatherData["tag", "prop"]	Weather data

Probability and statistics [DC]

RandomReal[{a, b}]	Uniform rand real in $[a, b]$	From $[0, 1]$ if no arg
RandomReal[dist]	Rand real from dist	
RandomInteger[{i, j}]	Uniform integer in $[i, j]$	Uniform 0 or 1 if no arg
RandomInteger[dist]	Rand integer from dist	
RandomSample[list, n]	n samples from list	
RandomSeed[]	Reset seed	
PDF[dist, x]	Density fct of dist	
CDF[dist, x]	Cumulant density fct of dist	
Histogram[list, w]	Histogram (bin width w)	Option: "Probability" "ProbabilityDensity"
BinCounts[list, w]	Counting dist of list (bin width w)	
Mean[list]	Also Mean[dist]	
Variance[list]		
StandardDeviation[list]		
ChiSquareDistribution[ν]		
ExponentialDistribution[λ]		
NormalDistribution[μ, σ]		
BinomialDistribution[n, p]		
BernoulliDistribution[p]		

Linear algebra [DC]

a.b	Dot or matrix product
Cross[a, b]	Cross product
Norm[a]	Euclidean norm
IdentityMatrix[n]	$n \times n$ identity matrix
Diagonal[m]	Diagonal of m
Diagonal[m, k]	k th elem in diagonal of m
Dimensions[m]	Dimensions of m
Inverse[m]	Inverse square matrix m
Det[m]	Determinant square matrix
Tr[m]	Trace matrix or tensor m
Transpose[l]	Transpose first two levels
Eigenvalues[m]	Eigenvals square matrix
Eigenvectors	Eigenvecs square matrix

Parallel computation [DC, Tut]

Parallelize[expr]	Direct parallelization
ParallelEvaluate[expr]	Eval on kernels
DistributeDefinitions	Put def on kernels
ParallelNeeds	Package on kernels
ParallelMap	Parallel Map
ParallelSum	Parallel Sum
ParallelProduct	Parallel Product
ParallelTable	Parallel Table
ParallelDo	Parallel Do
ParallelSubmit	Submit to kernel
	Collect with WaitAll
WaitAll	Results from kernels

Numerical routines [DC, Tut]

N[expr]	Numerical eval of expr	
NSolve[eqn, var]	Approx polynomial eqn	
FindRoot[eqn, {x, x0}]	Approx eqn in var x with seed x_0	
FindRoot[eqn, {x, x0, a, b}]	Approx eqn in $[a, b]$	
NDSolve[eqn, y, {x, a, b}]	Solve differential eqn	Initial cond specified in eqn
NIntegrate[f, {x, a, b}]	Numerical integration. Methods:	
	"SymbolicProcessing" -> 0	Fully numerical eval
	"GlobalAdaptive"	Default
	"LocalAdaptive"	
	"Oscillatory"	ID integrals; automatic detection
	"PrincipalValue"	Cauchy principal value
	"MonteCarlo"	Random sampling
	"AdaptiveMonteCarlo"	
FindMinimum[f, x, x0]	Find local min with seed x_0	
NMinimize[f, x]	Find global min. Methods:	
	"DifferentialEvolution"	Option: "SearchPoints"
	"NelderMead"	
	"RandomSearch"	Option: "SearchPoints"
	"SimulatedAnnealing"	Option: "SearchPoints"
NumericalCalculus package:		
NLimit[expr, x->x0]	Numerical limit	
ND[expr, x, x0]	Numerical D at x_0	
NSeries[f, {x, x0, n}]	Numerical series	

Notes