Einführung in *Mathematica*

Grundlegendes

Ausführen von Befehlen

```
4 + 1
```

```
5
```

```
(3 + 6) * 7 ^ 2 / 700
```

$$\frac{63}{100}$$

```
(* Symbolisches Rechnen *)
2 a + 5 a
```

```
{490, 469, 532}
```

```
(a + b) (a - b)
```

```
{-50325, -234632, -6545}
```

```
Expand[(a + b) (a - b)]
```

```
{-50325, -234632, -6545}
```

```
Simplify[a ^ 2 - 2 a b + b ^ 2]
```

```
{27225, 178084, 1225}
```

```
(* Ergebniss unterdrücken *)
4 + 1;
```

```
(* Befehle hintereinander ausführen *)
2 ^ 10
4 ^ 5
5 ^ 23
```

```
1024
```

```
1024
```

```
11920928955078125
```

```
(* Sequenz *)
x = 1 ; y = x + 1; y + x
```

```
3
```

Eine Sequenz ist ein einfacher Ausdruck, und kann Funktionen als Parameter übergeben werden. Wichtig für das imperative Programmieren mit *Mathematica*.

```
Exp[x = 1; y = x + 1; y + x]
```

e^3

Syntax

Kommentare: (* ... *)
Ausdrücke gruppieren: ( )
Funktionsparamter: [ ]
Listen, Mengen, Matrizen: { }
Indizierung: [[ ]]

Zugriff auf berechnete Ergebnisse

```
2 + 3;
```

```
(* Aufrufen des zuletzt berechneten Ergebnisses *)
Sqrt[%]
```

$\sqrt{5}$

```
(* Aufrufen des vorletzten Ergebnisses *)
Sqrt[%%]
```

$\sqrt{5}$

```
(* Numerische Berechnung *)
N[%]
```

```
2.23607
```

Variablen und Zuweisungen

```
a = 2 + 3
```

```
5
```

```
c = Sqrt[a]
```

$\sqrt{5}$

```
a = 10
```

10

```
c
```

$\sqrt{5}$

```
(* Löschen von Variablen *)
Clear[a, c]
```

```
a
```

a

```
b = 5
```

5

```
(* Alternativ mit dem Unset-Operator =. *)
b =.
```

```
b
```

b

Elementare Funktionen und Konstanten

```
Pi
```

π

```
N[Pi, 1000]
```

```
3.141592653589793238462643383279502884197169399375105820974944592307816406286208998628034825342117067982148086513282306647093844609550582231725359408128481117450284102701938521105559644622948954930381964428810975665933446128475648233786783165271201909145648566923460348610454326648213393607260249141273724587006606315588174881520920962829254091715364367892590360011330530548820466521384146951941511609433057270365759591953092186117381932611793105118548074462379962749567351885752724891227938183011949129833673362440656643086021394946395224737190702179860943702770539217176293176752384674818467669405132000568127145263560827785771342757789609173637178721468440901224953430146549585371050792279689258923542019956112129021960864034418159813629774771309960518707211349999998372978049951059731732816096318595024459455346908302642522308253344685035261931188171010003137838752886587533208381420617177669147303598253490428755468731159562863882353787593751957781857780532171226806613001927876611195909216420199
```

```
(* Eulersche Zahl *)
E
```

```
ⅇ
```

```
N[E]
```

```
2.71828
```

```
(* Komplexe Einheit *)
I
```

```
ⅈ
```

```
I^2
```

```
-1
```

```
Cos[Pi]
```

```
-1
```

```
Sin[Pi]
```

```
0
```

```
Tan[Pi]
```

```
0
```

```
Cot[Pi]
```

```
ComplexInfinity
```

```
Exp[Pi] - E^Pi
```

```
0
```

```
Log[E^2]
```

```
2
```

```
Log[10, 100]
```

```
2
```

Hilfe

```
? Sin
```

```
Sin[z] gives the sine of z. Mehr...
```

```
?? Solve
```

```
Solve[eqns, vars] attempts to solve an equation or set of equations for
   the variables vars. Solve[eqns, vars, elims] attempts to solve
   the equations for vars, eliminating the variables elims. Mehr...
```

```
Attributes[Solve] = {Protected}
```

```
Options[Solve] = {InverseFunctions → Automatic,
  MakeRules → False, Method → 3, Mode → Generic, Sort → True,
  VerifySolutions → Automatic, WorkingPrecision → ∞}
```

Ausdrücke und Funktionen

Ausdrücke

```
Clear[x]
f = 1 + Sqrt[x]
```

$1 + \sqrt{x}$

```
f
```

$1 + \sqrt{x}$

```
(* f ist keine Funktion *)
f[1]
```

$\left(1+\sqrt{x}\right)[1]$

```
(* Substitution *)
f /. x → 1
```

```
2
```

Funktionen

```
Clear[f, x]
```

Beim definieren von Funktionen müsssen die Variablen als Pattern (z.B.: *var_*) deklariert werden.

```
f[x_] := 1 + Sqrt[x];
```

```
f[3]
```

$1+\sqrt{3}$

```
Clear[g]
```

ACHTUNG!
Die folgende Gleichung definiert **keine** Funktion.

```
g[x] := 1 + Sqrt[x]
```

```
g[3]
```

```
g[3]
```

```
Mittelwert[x_, y_] := (x + y) / 2;
```

```
Mittelwert[3, 5]
```

```
4
```

"Set" und "SetDelayed"

```
Clear[ff, x]
```

```
x = 4;
```

Bei Funktionsdefinitionen mittels "Set (=)" wird die rechte Seite sofort evaluiert.

```
ff[x_] = 2 x;
```

```
ff[x]
ff[3]
ff[y]
```

```
8
```

```
8
```

```
8
```

```
Clear[ff, x]
```

```
x = 4;
```

Bei Funktionsdefinitionen mittels "SetDelayed (:=)" wird die rechte Seite lazy evaluiert.
Dies ist normalerweise das gewünsche Verhalten.

```
ff[x_] := 2 x;
```

```
ff[x]
ff[3]
ff[y]
```

```
8
```

```
6
```

```
4
```

```
z = 3;
```

Globable Variablen sind trotzdem möglich.

```
gg[x_] := 2 x + z
```

```
gg[x]
gg[3]
gg[y]
```

```
11
```

```
9
```

```
7
```

Listen, Mengen und Matrizen

Listen

```
li = {a, b, c, d}
```

```
{a, b, c, d}
```

```
(* Zugriff auf Elemente *)
First[li]
Last[li]
li[[1]]
li[[3]]
li[[-1]]
li[[{1, 3}]]
```

```
a
```

```
d
```

```
a
```

```
c
```

```
d
```

```
{a, c}
```

```
Append[li, a]
```

```
{a, b, c, d, a}
```

```
Length[li]
Reverse[li]
Delete[li, 2]
```

```
4
```

```
{d, c, b, a}
```

```
{a, c, d}
```

```
(* Aneinanderhängen von Listen *)
Join[li, {1, 2, 3}]
```

```
{a, b, c, d, 1, 2, 3}
```

Erzeugen und Manipulieren von Listen

```
Range[10]
```

```
{1, 2, 3, 4, 5, 6, 7, 8, 9, 10}
```

```
Range[5, 10]
```

```
{5, 6, 7, 8, 9, 10}
```

```
Range[1, 10, 3]
```

```
{1, 4, 7, 10}
```

```
Table[i^2, {i, 1, 10}]
```

```
{1, 4, 9, 16, 25, 36, 49, 64, 81, 100}
```

```
range = Range[10];
MemberQ[range, 5]
MemberQ[range, 11]
```

```
True
```

```
False
```

```
f[x_] := x^2;
Map[f, {1, 2, 3, 4, 5}]
```

```
{1, 4, 9, 16, 25}
```

```
(* Anonyme Funktionen *)
Map[#^2 &, {1, 2, 3, 4, 5}]
```

```
{1, 4, 9, 16, 25}
```

```
(* Elemente auswählen *)
Select[range, # > 5 && # < 8 &]
```

```
{6, 7}
```

```
Nest[# + 1 &, 1, 5]
```

```
6
```

```
Clear[f]
Nest[f, a, 5]
```

```
f[f[f[f[f[a]]]]]
```

```
Sum[i, {i, 1, 100}]
Product[i, {i, 1, 100}]
```

```
5050
```

```
93326215443944152681699238856266700490715968264381621468592963895217599
99322991560894146397615651828625369792082722375825118521091686400000
0000000000000000000
```

```
9332621544394415268169923885626670049071596826438162146859296389521759⋱
 9993229915608941463976156518286253697920827223758251185210916864000000⋱
 0000000000000000000
```

Mengen

```
li2 = {a, c, e, f}
```

```
{a, c, e, f}
```

```
Union[li, li2]
```

```
{a, b, c, d, e, f}
```

```
Intersection[li, li2]
```

```
{a, c}
```

```
Complement[li, li2]
Complement[li2, li]
```

```
{b, d}
```

```
{e, f}
```

```
Subsets[Range[5]]
```

```
{{}, {1}, {2}, {3}, {4}, {5}, {1, 2}, {1, 3}, {1, 4}, {1, 5}, {2, 3}, {2, 4},
 {2, 5}, {3, 4}, {3, 5}, {4, 5}, {1, 2, 3}, {1, 2, 4}, {1, 2, 5}, {1, 3, 4},
 {1, 3, 5}, {1, 4, 5}, {2, 3, 4}, {2, 3, 5}, {2, 4, 5}, {3, 4, 5}, {1, 2, 3, 4},
 {1, 2, 3, 5}, {1, 2, 4, 5}, {1, 3, 4, 5}, {2, 3, 4, 5}, {1, 2, 3, 4, 5}}
```

Matrizen

```
A = {{1, 2, 3}, {6, 3, 4}, {3, 4, 5}}
```

```
{{1, 2, 3}, {6, 3, 4}, {3, 4, 5}}
```

```
A // MatrixForm
```

$$\begin{pmatrix} 1 & 2 & 3 \\ 6 & 3 & 4 \\ 3 & 4 & 5 \end{pmatrix}$$

```
A // TableForm
```

1	2	3
6	3	4
3	4	5

```
Det[A]
```

```
8
```

```
Inverse[A] // MatrixForm
```

$$\begin{pmatrix} -\frac{1}{8} & \frac{1}{4} & -\frac{1}{8} \\ -\frac{9}{4} & -\frac{1}{2} & \frac{7}{4} \\ \frac{15}{8} & \frac{1}{4} & -\frac{9}{8} \end{pmatrix}$$

```
(* Matrix Multiplikation *)
A . Inverse[A] // MatrixForm
```

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

```
(* Multiplication mit Skalar *)
2 A // MatrixForm
```

$$\begin{pmatrix} 2 & 4 & 6 \\ 12 & 6 & 8 \\ 6 & 8 & 10 \end{pmatrix}$$

```
A[[1, 1]] = 0; A // MatrixForm
```

$$\begin{pmatrix} 0 & 2 & 3 \\ 6 & 3 & 4 \\ 3 & 4 & 5 \end{pmatrix}$$

Polynome

```
Clear[f, x, y]
f = (x + y) ^10 - (x + y^2) ^5
```

$$(x+y)^{10} - (x+y^2)^5$$

```
Expand[f]
```

$$-x^5 + x^{10} + 10\,x^9\,y - 5\,x^4\,y^2 + 45\,x^8\,y^2 + 120\,x^7\,y^3 - 10\,x^3\,y^4 + 210\,x^6\,y^4 + 252\,x^5\,y^5 - 10\,x^2\,y^6 + 210\,x^4\,y^6 + 120\,x^3\,y^7 - 5\,x\,y^8 + 45\,x^2\,y^8 + 10\,x\,y^9$$

```
Collect[f, x]
```

$$x^{10} + 10\,x^9\,y + 45\,x^8\,y^2 + 120\,x^7\,y^3 + 210\,x^6\,y^4 + x^5\,(-1 + 252\,y^5) + x^4\,(-5\,y^2 + 210\,y^6) + x^3\,(-10\,y^4 + 120\,y^7) + x^2\,(-10\,y^6 + 45\,y^8) + x\,(-5\,y^8 + 10\,y^9)$$

```
Collect[f, y]
```

$$-x^5 + x^{10} + 10\,x^9\,y + (-5\,x^4 + 45\,x^8)\,y^2 + 120\,x^7\,y^3 + (-10\,x^3 + 210\,x^6)\,y^4 + 252\,x^5\,y^5 + (-10\,x^2 + 210\,x^4)\,y^6 + 120\,x^3\,y^7 + (-5\,x + 45\,x^2)\,y^8 + 10\,x\,y^9$$

```
Factor[f]
```

$$x(-1+x+2y)(x^4+x^5+x^6+x^7+x^8+2x^4y+4x^5y+6x^6y+8x^7y+5x^3y^2+9x^4y^2+17x^5y^2+29x^6y^2+10x^3y^3+28x^4y^3+62x^5y^3+10x^2y^4+30x^3y^4+86x^4y^4+20x^2y^5+80x^3y^5+10xy^6+50x^2y^6+20xy^7+5y^8)$$

```
Collect[f, x, Factor]
```

$$x^{10}+10x^9y+45x^8y^2+120x^7y^3+210x^6y^4+5xy^8(-1+2y)+5x^2y^6(-2+9y^2)+10x^3y^4(-1+12y^3)+5x^4y^2(-1+42y^4)+x^5(-1+252y^5)$$

```
Coefficient[f, x, 5]
```

$$-1+252y^5$$

```
Coefficient[f, x^5]
```

$$-1+252y^5$$

```
Exponent[f, x]
Exponent[f, x, Min]
```

10

1

```
Exponent[f, y]
Exponent[f, y, Min]
```

9

0

Analysis

Differentation

```
f = (Sin[x] + Cos[x]^2) / Exp[x]
```

$$e^{-x}(\text{Cos}[x]^2+\text{Sin}[x])$$

```
D[f, x]
```

$$-e^{-x}(\text{Cos}[x]^2+\text{Sin}[x])+e^{-x}(\text{Cos}[x]-2\,\text{Cos}[x]\,\text{Sin}[x])$$

```
(* 3. Ableitung nach x *)
D[f, {x, 3}]
```

$-e^{-x}(\mathrm{Cos}[x]^2+\mathrm{Sin}[x])+3e^{-x}(\mathrm{Cos}[x]-2\,\mathrm{Cos}[x]\,\mathrm{Sin}[x])+e^{-x}(-\mathrm{Cos}[x]+8\,\mathrm{Cos}[x]\,\mathrm{Sin}[x])-3e^{-x}(-2\,\mathrm{Cos}[x]^2-\mathrm{Sin}[x]+2\,\mathrm{Sin}[x]^2)$

Integration

```
(* Unbestimmtes Integral *)
Integrate[f, x]
```

$-\frac{1}{10}e^{-x}(5+5\,\mathrm{Cos}[x]+\mathrm{Cos}[2x]+5\,\mathrm{Sin}[x]-2\,\mathrm{Sin}[2x])$

```
D[%, x]
```

$-\frac{1}{10}e^{-x}(5\,\mathrm{Cos}[x]-4\,\mathrm{Cos}[2x]-5\,\mathrm{Sin}[x]-2\,\mathrm{Sin}[2x])+\frac{1}{10}e^{-x}(5+5\,\mathrm{Cos}[x]+\mathrm{Cos}[2x]+5\,\mathrm{Sin}[x]-2\,\mathrm{Sin}[2x])$

```
TrigFactor[%]
```

$\frac{1}{2}e^{-x}(1+\mathrm{Cos}[2x]+2\,\mathrm{Sin}[x])$

```
TrigFactor[f]
```

$\frac{1}{2}e^{-x}(1+\mathrm{Cos}[2x]+2\,\mathrm{Sin}[x])$

```
(* Bestimmtes Integral *)
Integrate[f, {x, 0, Pi}]
```

$\frac{1}{10}(11-e^{-\pi})$

Grenzwerte

```
Limit[f, x → Infinity]
```

0

```
(* Grenzwert von links *)
Limit[Tan[x], x → π / 2, Direction → 1]
```

∞

```
(* Grenzwert von rechts *)
Limit[Tan[x], x → π / 2, Direction → -1]
```

$-\infty$

Symbolische Summen, Reihen

```
Sum[Binomial[n, i] 2^i, {i, 0, n}]
```

3^n

```
Sum[1 / i^2, {i, 1, Infinity}]
```

$\frac{\pi^2}{6}$

Symbolische Summen, Reihen

```
Sum[Binomial[n, i] 2^i, {i, 0, n}]
```

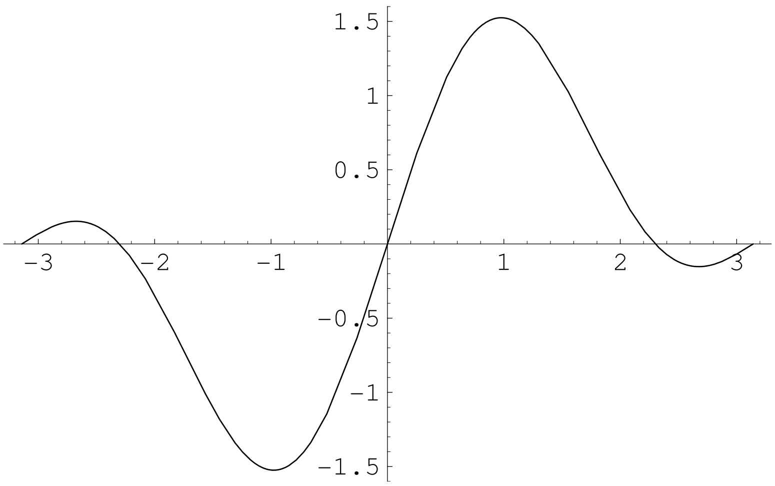
3^n

```
Sum[1 / i^2, {i, 1, Infinity}]
```

$\frac{\pi^2}{6}$

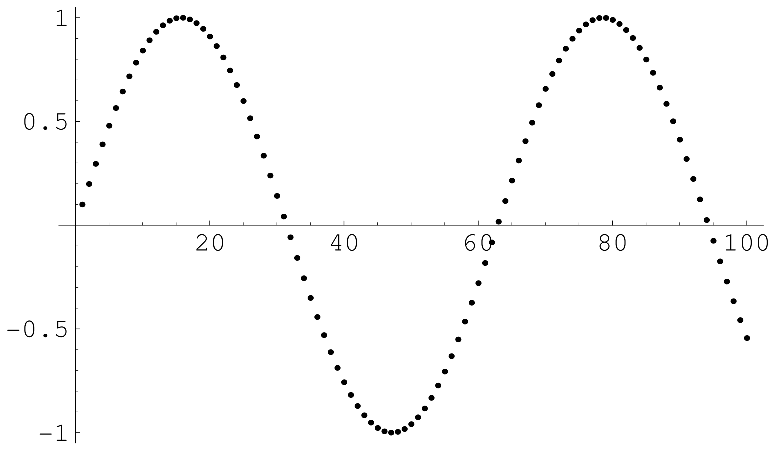
Grafik

```
Plot[Sin[x] + 1 / 2 Sin[2 x] + 1 / 4 Sin[2 x], {x, -Pi, Pi}];
```



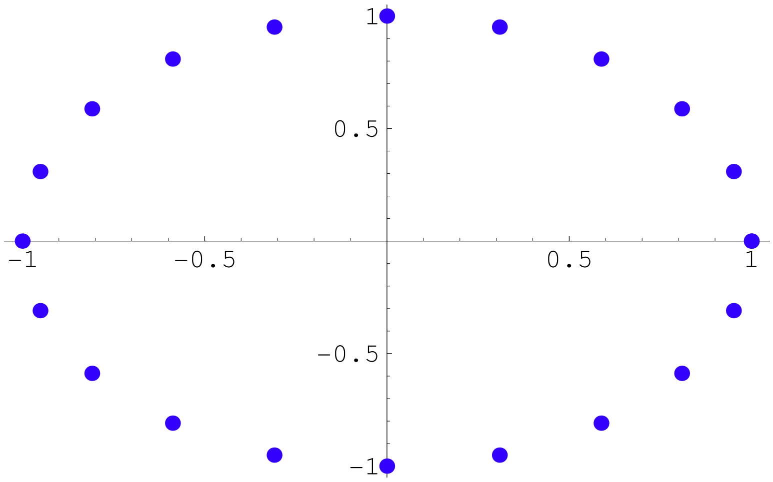

```
sinus = Table[Sin[x / 10], {x, 1, 10^2}];
```

```
ListPlot[sinus];
```

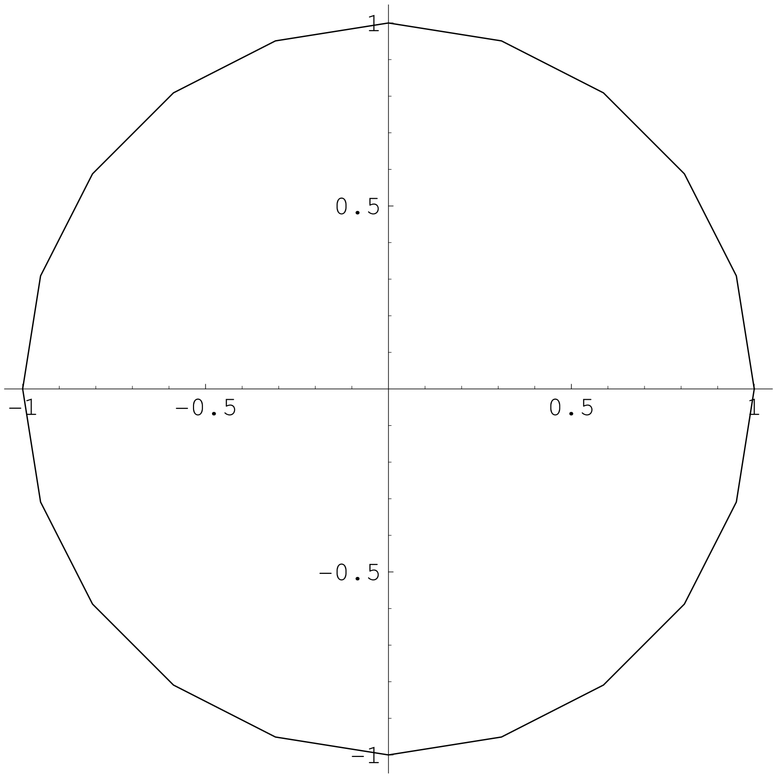



```
data = Table[{Cos[2 Pi x / 20], Sin[2 Pi x / 20]}, {x, 0, 20}];
```

```
ListPlot[data, PlotStyle → {PointSize[0.02], Hue[0.7]}];
```



```
ListPlot[data, PlotJoined → True, AspectRatio → 1];
```



Zahlentheorie

Euklidscher Algorithmus

```
(* Grösster gemeinsamer Teiler *)
GCD[10, 15]
```

```
5
```

```
ExtendedGCD[10, 15]
```

```
{5, {-1, 1}}
```

```
-1 * 10 + 1 * 15
```

```
5
```

```
(* Kleinstes gemeinsame Vielfache *)
LCM[3, 4]
```

```
12
```

```
GCD[2 ^ 2000 - 1, 3 ^ 1000 - 1]
```

```
7145813125
```

```
LCM[2^2000 - 1, 3^1000 - 1]
```

```
212419505327086477198138534957193583143898070498021448428898052667618 6⋱
 540337515520522368836851166261013165008729933741953544430162468464479⋱
 727461751151186344340945575858445926557266109885553627022228068783067⋱
 040585006640638208533320190808275570827485467618046904626080806824528⋱
 840421461862226415218170859165014941661612049604504369676960036923727⋱
 771267696427801070371508919605420590848320560475919989351827463726020⋱
 350764672836561764653224810275232738962409653825074880963423855162968⋱
 865272204060219037456940242749175725036178704374797419964970110566982⋱
 926207887564682912430080876423119914999972920369919548816517800613073⋱
 234429690810649717169538014283725989657747813120374611674289590357451⋱
 713469666802110705158488226888475560168433950863086094880326909910491⋱
 359355342398790980602125753813674562269490545450522533471918821702710⋱
 837319468010696342036223002322664482136921408812124958808978590137757⋱
 522338385567379743015312833980532537456968809973101980377711288035224⋱
 970808722357196893324274068159757092314801932147690662785428141728386⋱
 70708620850862012361534840213400 00
```

Primzahlen und Faktorisierung

```
(* Berechnen der n-ten Primzahl *)
Prime[1]
```

```
2
```

```
Prime[10^9]
```

```
22801763489
```

```
(* Liste der ersten 100 Primzahlen *)
Table[Prime[i], {i, 1, 100}]
```

```
{2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67,
 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 127, 131, 137, 139,
 149, 151, 157, 163, 167, 173, 179, 181, 191, 193, 197, 199, 211, 223,
 227, 229, 233, 239, 241, 251, 257, 263, 269, 271, 277, 281, 283,
 293, 307, 311, 313, 317, 331, 337, 347, 349, 353, 359, 367, 373,
 379, 383, 389, 397, 401, 409, 419, 421, 431, 433, 439, 443, 449,
 457, 461, 463, 467, 479, 487, 491, 499, 503, 509, 521, 523, 541}
```

```
(* Primzahl Test *)
PrimeQ[19]
PrimeQ[20]
```

```
True
```

```
False
```

```
(* 20. Mersienne Primzahl (1961) *)
Timing[PrimeQ[2^4423 - 1]]
```

```
{4.42633 Second, True}
```

```
<< NumberTheory`PrimeQ`
```

```
ProvablePrimeQ[19]
ProvablePrimeQ[20]
```

```
True
```

```
False
```

```
p = NextPrime[2^200]
```

```
1606938044258990275541962092341162602522202993782792835301611
```

```
Timing[ProvablePrimeQ[p]]
Timing[PrimeQ[p]]
```

```
{4.85626 Second, True}
```

```
{0.005999 Second, True}
```

```
(* Primfaktorisierung *)
factors = FactorInteger[2^200 - 1]
```

```
{{3, 1}, {5, 3}, {11, 1}, {17, 1}, {31, 1}, {41, 1}, {101, 1},
 {251, 1}, {401, 1}, {601, 1}, {1801, 1}, {4051, 1}, {8101, 1},
 {61681, 1}, {268501, 1}, {340801, 1}, {2787601, 1}, {3173389601, 1}}
```

```
MultiplyFactors[list_] := Times @@ Map[#[[1]]^#[[2]] &, list]
```

```
MultiplyFactors[factors] == 2^200 - 1
```

```
True
```

Restklassen

```
Mod[67, 20]
Quotient[76, 20]
```

```
7
```

```
3
```

```
Timing[Mod[3^10000000, 77]]
```

```
{3.25651 Second, 67}
```

```
Timing[PowerMod[3, 10000000, 77]]
```

{1.21084 × 10^{-15} Second, 67}

```
Mod[3^(2^100), 77]
```

General::ovfl : Overflow occurred in computation. Mehr…

```
Overflow[]
```

```
PowerMod[3, 2^100, 77]
```

25

```
(* Inverses Element modulo 11 *)
PowerMod[57, -1, 11]
```

6

```
(* Probe *)
Mod[57 * 6, 11]
```

1

Mathematica als Programmiersprache

```
fact1[n_] := n!
```

```
fact2 := #! &
```

```
fact3[0] := 1
fact3[n_] := n fact3[n - 1];
```

```
fact3a[0] := 1
fact3a[n_ /; n > 0] := n fact3a[n - 1];
```

```
fact4[n_] := Product[i, {i, n}]
```

```
fact5[n_] := Module[{t = 1}, For[i = 1, i ≤ n, i++, t *= i]; t]
```

```
fact6[n_] := Module[{i = 1, t = 1}, While[i <= n, t = t * i; i++]; Return[t]]
```

```
fact7[n_Integer] := Fold[#1 #2 &, 1, Range[n]]
```

```
fact7a[n_Integer] := Fold[Function[{x, y}, x * y], 1, Range[n]]
```

```
fact7list[n_Integer] := FoldList[#1 #2 &, 1, Range[n]]
```

```
MyFold[f_, e_, {}] := e;
MyFold[f_, e_, list_] := MyFold[f, f[e, First[list]], Rest[list]]
```

```
fact8[n_Integer] := MyFold[#1 #2 &, 1, Range[n]]
```

```
fact9[n_] := Times @@ Range[n]
```

```
fact10[n_] := Apply[Times, Range[n]]
```

```
fact11[n_ /; n ≥ 0] := If[n == 0, 1, n fact11[n - 1]]
```

```
fact12 = If[#1 == 0, 1, #1 #0[#1 - 1]] &;
```

- **Laufzeitvergleiche**

```
Timing[fact1[50000];]
```

```
{0.286956 Second, Null}
```

```
Timing[fact4[50000];]
```

```
{0.453931 Second, Null}
```

```
Timing[fact5[50000];]
```

```
{5.8851 Second, Null}
```

```
Timing[fact6[50000];]
```

```
{5.8861 Second, Null}
```

```
Timing[fact7[50000];]
```

```
{5.18921 Second, Null}
```

```
Timing[fact10[50000];]
```

```
{0.411937 Second, Null}
```