

# Uebungen ■ Exercices

## 5. Etwas Umgang mit *Mathematica* ■ *Savoir un peu manier Mathematica*

Die Gliederung dieses Kurses folgt in groben Zügen dem Buch von Nancy Blachman: A Practical Approach....

Hinweis: Kapitel 5 lesen!

Run mit WIN+*Mathematica* Version 5.2

■ L'articulation de ce cours correspond à peu près à celle du livre de Nancy Blachman: A Practical Approach....

Indication: Lire le chapitre 5.

Testé avec *Mathematica* version 5.2+WIN

WIR94/98/99/2000/2007 // Copyright Rolf Wirz

---

## Aufgabe 1 ■ Problème 1

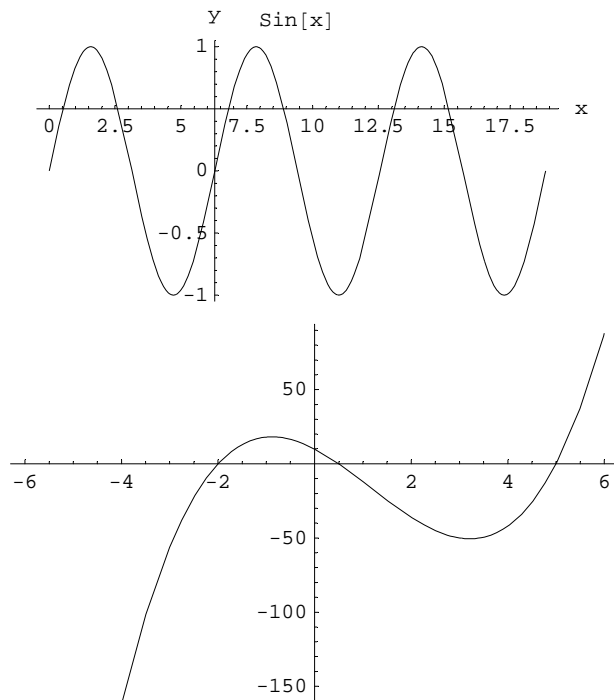
### Ein Vorspann

#### ■ Génériques

Sinus ohne " ; ", damit Input-Lines erzeugt werden.

■ Sinus sans " ; ", pour créer des Input-Lines.

```
In[1]:= Plot[Sin[x],{x,0,6 Pi},
  AspectRatio -> 1/2,
  PlotLabel -> "Sin[x]",
  AxesLabel -> {"x", "y"},
  AxesOrigin -> {2 Pi, 1/2} 1];
Plot[2x^3-7x^2-17x+10,{x,-6,6}];
Solve[2x^3-7x^2-17x+10==0,x]
```



```
Out[3]= {{x -> -2}, {x ->  $\frac{1}{2}$ }, {x -> 5}}
```

## Input-Lines auflisten

### ■ Faire une Liste des Input-Lines

Mit "??": ■ Avec "??":

```
In[4]:= ??In
```

In[n] is a global object that is assigned to have a delayed value of the nth input line.  
Mehr...

```
Attributes[In] = {Listable, Protected}
```

```
In[1] := (Plot[Sin[x], {x, 0, 6 π}, AspectRatio ->  $\frac{1}{2}$ ,  
PlotLabel -> Sin[x], AxesLabel -> {x, y}, AxesOrigin -> {2 π,  $\frac{1}{2}}$ ];)
```

```
In[2] := (Plot[2 x3 - 7 x2 - 17 x + 10, {x, -6, 6}];)
```

```
In[3] := Solve[2 x3 - 7 x2 - 17 x + 10 == 0, x]
```

```
In[4] := Information[In, LongForm -> True]
```

## Aufgabe 2 ■ Problème 2

### Auffinden von Kommandos ■ Trouver des commandements

Finde alle Kommandos mit den Worten "Matrix" und "Power" im Namen:

■ Trouve tous les commandements qui ont les paroles "Matrix" et "Power" dans leur nom:

```
In[5]:= ?*Matrix* *Power*
```

#### System`

```
BezoutMatrix MatrixForm Plot3Matrix
DiagonalMatrix MatrixPower SylvesterMatrix
IdentityMatrix MatrixQ
MatrixExp MatrixRank
```

#### System`

```
MatrixPower PowerExpand PowerModList
Power PowerMod
```

Oder: ■ Ou:

```
In[6]:= ??*Matrix* *Power*
```

#### System`

```
BezoutMatrix MatrixForm Plot3Matrix
DiagonalMatrix MatrixPower SylvesterMatrix
IdentityMatrix MatrixQ
MatrixExp MatrixRank
```

#### System`

```
MatrixPower PowerExpand PowerModList
Power PowerMod
```

Oder: ■ Ou:

```
In[7]:= Print[Names["*Matrix*"]];
Names["*Power*"]

{BezoutMatrix, DiagonalMatrix, IdentityMatrix, MatrixExp,
 MatrixForm, MatrixPower, MatrixQ, MatrixRank, Plot3Matrix, SylvesterMatrix}

Out[8]= {MatrixPower, Power, PowerExpand, PowerMod, PowerModList}
```

Kurze Neugier:

■ Petite curiosité:

`In[9]:= ??Names`

`Names["string"]` gives a list of the names of symbols which match the string. `Names["string", SpellingCorrection->True]` includes names which match after spelling correction. Mehr...

`Attributes[Names] = {Protected}`

`Options[Names] = {IgnoreCase->False, SpellingCorrection->False}`

## Aufgabe 3 ■ Problème 3

### CPU-Zeit messen, 3 x 3-Matrix

#### ■ Mesurer le temps CPU, matrice 3 x 3

Invertierung einer 3 x 3-Matrix. Postfix-Notation verwenden.

Erst Matrix generieren:

■ Invertir une matrice 3 x 3. User la notation postfixe. Générer d'abord la matrice:

`In[10]:= m = Table[Random[ ], {3}, {3}]; MatrixForm[m]`

`Out[10]//MatrixForm=`

$$\begin{pmatrix} 0.792512 & 0.232299 & 0.946558 \\ 0.631949 & 0.677425 & 0.572377 \\ 0.0745678 & 0.852139 & 0.863563 \end{pmatrix}$$

Dann Inverse rechnen und Zeit messen (Postfix-Notation):

■ Ensuite calculer l'inverse et mesurer le temps (notation postfixe):

`In[11]:= m // Inverse // MatrixForm // Timing`

`Out[11]= {0. Second,  $\begin{pmatrix} 0.230388 & 1.43555 & -1.20403 \\ -1.19168 & 1.45405 & 0.342454 \\ 1.15602 & -1.55877 & 0.924037 \end{pmatrix}$ }`

### CPU-Zeit messen, 10 x 10-Matrix

#### ■ Mesurer le temps CPU, matrice 10 x 10

Matrix generieren, Inverse rechnen und Zeit messen (Postfix-Notation):

■ Générer une matrice, calculer l'inverse et mesurer le temps. (Notation postfixe):

`In[12]:= Table[Random[ ], {10}, {10}] // Inverse; // Timing`

`Out[12]= {0. Second, Null}`

---

## Aufgabe 4 ■ Problème 4

### (a) Fakultät berechnen: ■ Calculer des factoriels

1 ! bis 5 ! ■ 1 ! à 5 !

```
In[13]:= Do[Print[i!], {i, 5}]
```

```
1
2
6
24
120
```

### (a) Grössere Fakultät berechnen: ■ Calculer des factoriels plus grands:

1 ! bis 250 ! **Achtung!** Der Rechner kann damit unmöglich fertig werden.

Die Zahlen sind zu gross! Daher musst Du den Vorgang unterbrechen!

Versuche in **Action**, **Interrupt**, **Abort Calculation** oder **Interrupt Calculation**.

■ 1 ! à 250 ! **Attention!** La calculatrice ne peut pas en venir à bout. Les nombres sont trop grands! Tu dois donc interrompre le processus! Essaie

dans (**Action**) **Kernel**, **Interrupt**, **Abort Calculation** ou **Interrupt Calculation**.

```
In[14]:= Do[Print[i!], {i, 42}]
```

```
1
2
6
24
120
720
5040
40320
362880
3628800
39916800
479001600
```

6227020800  
87178291200  
1307674368000  
20922789888000  
355687428096000  
6402373705728000  
121645100408832000  
2432902008176640000  
51090942171709440000  
1124000727777607680000  
25852016738884976640000  
620448401733239439360000  
15511210043330985984000000  
403291461126605635584000000  
10888869450418352160768000000  
304888344611713860501504000000  
8841761993739701954543616000000  
265252859812191058636308480000000  
8222838654177922817725562880000000  
263130836933693530167218012160000000  
8683317618811886495518194401280000000  
295232799039604140847618609643520000000  
10333147966386144929666651337523200000000  
371993326789901217467999448150835200000000  
13763753091226345046315979581580902400000000  
523022617466601111760007224100074291200000000  
20397882081197443358640281739902897356800000000  
815915283247897734345611269596115894272000000000  
33452526613163807108170062053440751665152000000000  
1405006117752879898543142606244511569936384000000000

Bitte nicht ausdrucken:

■ S.v.p. ne pas imprimer:

```
In[15]:= (* Do[Print[i!], {i, 250}] *)
```

```
In[16]:= 500!
```

```
Out[16]= 122013682599111006870123878542304692625357434280319284219241358838584537315388199:
76054964475022032818630136164771482035841633787220781772004807852051593292854779:
07571939330603772960859086270429174547882424912726344305670173270769461062802310:
45264421887878946575477714986349436778103764427403382736539747138647787849543848:
95955375379904232410612713269843277457155463099772027810145610811883737095310163:
56324432987029563896628911658974769572087926928871281780070265174507768410719624:
39039432253642260523494585012991857150124870696156814162535905669342381300885624:
92468915641267756544818865065938479517753608940057452389403357984763639449053130:
62323749066445048824665075946735862074637925184200459369692981022263971952597190:
94521782333175693458150855233282076282002340262690789834245171200620771464097945:
61161276291459512372299133401695523638509428855920187274337951730145863575708283:
55780158735432768888680120399882384702151467605445407663535984174430480128938313:
89688163948746965881750450692636533817505547812864000000000000000000000000000000:
000000000000000000000000000000000000000000000000000000000000000000000000000000:
0000000000000000
```

---

## Aufgabe 5 ■ Problème 5

### Einige Variablen:

#### ■ Quelques variables:

```
In[17]:= ??$Line
```

\$Line is a global variable that specifies the number of the current input line. Mehr...

```
$Line = 17
```

```
In[18]:= ??$Version
```

\$Version is a string that represents the version of Mathematica you are running. Mehr...

```
Attributes[$Version] = {Locked, Protected}
```

```
$Version = 5.2 for Microsoft Windows (June 20, 2005)
```

```
In[19]:= ??$VersionNumber
```

\$VersionNumber is a real number which gives the current Mathematica kernel version number, and increases in successive versions. Mehr...

```
Attributes[$VersionNumber] = {Locked, Protected}
```

```
$VersionNumber = 5.2
```

---

## "Putzmaschine" einsetzen

### ■ Employer la "machine de nettoyage"

```
In[20]:= (* Old Form: Remove["Global`*"] *)
```

```
In[21]:= Remove["Global`*"]
```