

Exercises of *The Mathematica GuideBook for Programming*

This document was downloaded from <http://www.mathematicaguidebooks.org>. It contains extracted material from the notebooks accompanying *The Mathematica GuideBook for Programming*. Copyright Springer 2004. Redistributed with permission.

CHAPTER 1

Exercises

1.^{L2} What You Always Wanted to Compute

Find a problem or lengthy calculation (or a few of them) that you have always wanted to solve or carry out. Make a list of such problems, and as you read this book, try to find ways to solve your problems with *Mathematica*.

2.^{L2} *Mathematica* or Axiom or Maple or MuPAD or REDUCE or Form?

Compare the mathematical capabilities, clarity and uniformity of the syntax, required computational times, and directness with which mathematical ideas can be converted to programs using *Mathematica* and other (general-purpose) computer algebra systems. Use your computer if you have the technical (and financial) means. If not, page through, look at, read, and carefully study the corresponding handbooks and documentation. Here is some information on some of the various systems (for a more detailed listing and additional references, see [17★]).

axiom: <http://www.nongnu.org/axiom/>

References: [23★], [27★]

Maple: <http://www.maplesoft.com>

References: [4★], [5★], [37★], [34★], [6★], [16★], [19★], [26★], [11★], [12★], [21★], [14★], [36★], [28★], [24★], [10★], [25★], [8★], [9★], [2★], [22★], [3★], [18★], [29★], [30★], and [1★], as well as the Maple newsletters Maple Tech published by Birkhäuser and the newsgroup `comp.soft-sys.math.maple`.

MuPAD: <http://www.mupad.de>

References: [13★],[15★], [33★], [38★], [32★]

Reduce: <http://www.uni-koeln.de/REDUCE/>

References: [42★], [40★], [41★], [20★], [31★], and [35★]

Form: <http://www.nikhef.nl/~form/>

References: [43★], [44★]

For an overview of all general and special-purpose computer algebra systems and how to obtain them, see *Computeralgebra--Report from Germany* [7★], <http://SymbolicNet.mcs.kent.edu/systems/Systems.html>, http://www.can.nl/Systems_and_Packages/Per_Purpose/General/index_table.html.

For a standardized format (OpenMath) concerning the mutual exchange of data between computer algebra systems, see [39★], <http://www.openmath.org/>.

3.^{L1} Improvements?

We refer occasionally to some inconsistencies, restrictions, or bugs in *Mathematica* Version 4.0. If the reader has a newer version, check if these inconsistencies, restrictions, or bugs are still there, or if they have been removed.

CHAPTER 2

Exercises

1.^{L1} What Is the Answer?

Predict the results of the following *Mathematica* inputs, and compare each prediction with the actual *Mathematica* output.

- a) `b + a + a`
- b) `2 + 4 + u + 8 + i + u - i`
- c) `2 + 0I`
- d) `Head[2 + 0I]`
- e) `0.0I - 0.0I`
- f) `FullForm[0.0I - 0.0I]`
- g) `Infinity^Infinity`
- h) `Infinity/Infinity`
- i) `Infinity - Infinity`
- j) `1/Indeterminate`
- k) `FullForm[s + s^s/s - s]`
- l) `Times[Times, Times]`

m) `Times[Times[], Times[]]`

n) `Times[Times[Times], Times[Times]]`

2.^{L1} `FullForm[expression]` with `()`?

Try to find a *Mathematica* expression `expression` so that `FullForm[expression]` contains parentheses.

3.^{L1} `na38bv94iwymmwpu1k5h6jhtye934` and `((1/2 + 1/4 I)^(7))^(1/7)`

a) What could the input be if the output is `na38bv94iwymmwpu1k5h6jhtye934`. Give at least two possible answers.

b) Why is the result of inputting `((1/2 + 1/5 I)^(7))^(1/7)` just `1/2 + 1/5 I`, but the result of `((1/2 + 1/4 I)^(7))^(1/7)` is `(-139/8192 - 29 I/16384)^(1/7)` and not `1/2 + 1/4 I`?

c) Find a built-in function `f`, such that the input `Head @ (Im[f[3]] // N)` returns the output `Complex`.

4.^{L2} `Level`, `Depth`, and `Part`

Analyze the following expression as a *Mathematica* expression:

$$\text{expr} = \sin(\tan(1 + e^{-x}) + x^x - \ln(\ln(r t + a x)) + d(x) + x(x) \arccos(\arcsin(x^2)) + h(h(h(i))))$$

What is its depth? Examine all possible levels. Where does `x` appear? Investigate all sensible values of `Part[expr, nonNegativeNumber]`.

5.^{L2} `Level[expr, {-2, 2}]` versus `Level[expr, {2, -2}]`

What are the results of the following two inputs?

`Level[Sin[3 x + Cos[6/(t + Tan[r])]]/Exp[-x^2]], {-2, 2}]`

`Level[Sin[3 x + Cos[6/(t + Tan[r])]]/Exp[-x^2]], {2, -2}]`

6.^{L2} `Branch Cuts`

a) Discuss the location of the branch cuts of the function $f(z) = 1 / (z^4)^{\frac{1}{4}}$ in *Mathematica* (meaning `1 / (z^4)^(1/4)`). What are the values of the function $f(z)$ on the other sheets of the Riemann surface?

b) Theoretically the location of branch cuts of an analytic function is not fixed. But by using the built-in functions of *Mathematica*, the branch cuts of nested functions built from the built-in functions are determined. Determine the location in *Mathematica* of the branch cuts of the function $f(z) = \sqrt{z+1/z} \sqrt{z-1/z}$.

c) Determine the branch points and the branch cuts of the following function $w(z)$: $w(z) = \arctan(\tan(z/2)/2)$.

d) Characterize the function `Sqrt[z] - 1/Sqrt[1/z]`.

e) Characterize the function `1/(z + Sqrt[z^2])`.

f) Discuss the branch cut and branch point structure of the function $g(z) = \sqrt{z + \sqrt{z-1}} \sqrt{z+1}$ in *Mathematica*.

g) Describe the branch cut location of the function $f(z) = 1/\text{Log}[\text{Exp}[1/z]]$.

h) Discuss the branch point and branch cut structure of the functions $\operatorname{arccoth}$, $\operatorname{arccosh}$, and $\operatorname{arcsech}$. In *Mathematica* they are defined as

$$\begin{aligned}\operatorname{ArcCoth}[z] &= \operatorname{Log}[1 + 1/z]/2 - \operatorname{Log}[1 - 1/z]/2 \\ \operatorname{ArcCosh}[z] &= \operatorname{Log}[z + \operatorname{Sqrt}[z + 1] \operatorname{Sqrt}[z - 1]] \\ \operatorname{ArcSech}[z] &= \operatorname{Log}[\operatorname{Sqrt}[1/z + 1] \operatorname{Sqrt}[1/z - 1] + 1/z].\end{aligned}$$

How many different sheets does one reach by encircling the origin and ± 1 (on the corresponding Riemann surface) at various radii? What happens if one moves around the eight-shaped contour $\{2 \cos(\varphi), \sin(\varphi)\}$? Is infinity a branch point? What are the differences of the function values across the branch cuts?

7.^{L2} “Strange” Analytic Functions

For all parts of this exercise, use only analytic functions like `Exp`, `Log`, `Power`, `Sqrt`, ... as building blocks, do not use functions like `Abs`, `Re`,

- Construct a function f that is 1 on the unit circle $|z| = 1$ and 0 everywhere else (with the possible exception of a finite number of other points).
- Construct a function f that is 1 inside the unit circle and 0 everywhere else.
- Construct a function f that evaluates to 1 at $x = 0$ and to 0 at every other real x .
- Construct a function f that evaluates 1 in the open interval $(0, 1)$ and to 0 at every other real x .
- Construct a function f that is equal to the staircase function $\lfloor x \rfloor$ for real values x (with the possible exception of the points of discontinuity of $\lfloor x \rfloor$). (Here, $\lfloor x \rfloor$ is equal to the smallest integer less than or equal to x .)
- Construct a function f that is equal to the “castle rim function” $x \bmod 2$ (with the possible exception of the points of discontinuity of $x \bmod 2$).
- Construct a function f that is equal to the sawtooth function $1 - 2 \lfloor [x] - x/2 \rfloor$, where $\lfloor x \rfloor$ denotes the rounding to nearest integer to x .

8.^{L2} `ArcTan[(x + 1)/y] - ArcTan[(x - 1)/y]` Picture

Predict how a picture of $\tan^{-1}((x + 1)/y) - \tan^{-1}((x - 1)/y)$ over the real x, y -plane will look. We get such a picture in *Mathematica* by using the following code.

```
f[x_,y_] := ArcTan[(x + 1)/y] - ArcTan[(x - 1)/y]
ε = 10^-14;
Plot3D[Evaluate[f[x,y]], {x, -Pi, Pi}, {y, ε, Pi},
  PlotPoints -> 50, PlotRange -> All];
```

9.^{L2} `ArcSin[ArcSin[z]]` Picture

Predict how a 3D picture of $\operatorname{Im}(\sin^{-1}(\sin^{-1}(x + iy)))$ over the real x, y -plane will look. We get such a picture in *Mathematica* by using the following code.

```
Plot3D[Im[ArcSin[ArcSin[x + I y]]], {x, -3, 3}, {y, -3, 3}, PlotPoints -> 40];
```

10.^{L2} Singularities of $\tanh(\sinh(\cot(z)))$, $\exp(\ln^{i\pi}(z))$ Properties

- At which points z does the function $w(z) = \tanh(\sinh(\cot(z)))$ have singularities? What kind of singularities?

b) Describe the branch cuts of the function $f(z) = \arg(\exp(\ln^{i\pi}(z)))$ over the complex z -plane. Express $\arg(f(z))$ in an explicit real way as a function of $|z|$ and $\arg(z)$ and give a qualitative description of $\arg(f(z))$ over the complex z -plane.

11.^{L1} `Exp[-1/Im[1/(-Log[Infinity] + 2)^2]]`

Predict the result of evaluating `Exp[-1/Im[1/(-Log[Infinity] + 2)^2]]`.

12.^{L1} Predict the Result

Predict the results of

```
N[(1 - 10^-21) Exp[I 2], 22]^Infinity
```

and

```
N[(1 - 10^-23) Exp[I 2], 22]^Infinity.
```

13.^{L1} `tan(k/α) + tan(α k)` Picture

The following input defines a function `tanPicture` that displays the set of points $\{k, \tan(k\alpha) + \tan(k/\alpha)\}$ for $k = 1, \dots, 20000$. Find different real values of α such that `tanPicture[α]` looks “qualitatively different”.

```
tanPicture[α_] :=
ListPlot[Table[Tan[α k] + Tan[1/α k], {k, 20000}],
  PlotStyle -> {PointSize[0.001]}, PlotRange -> {-2, 2},
  Frame -> True, Axes -> False, FrameTicks -> None]
```

CHAPTER 3

Exercises

1.^{L1} Predictions

What will be the results of the following *Mathematica* inputs?

a) `#^2&[1/#^3&[2]]`

```
Function[#, #^2][x]
```

```
Function[Slot, Slot^2][x]
```

```
Function[Slot, #^2&[Slot]][5]
```

b) `sin[##]&[1, 2]`

```
#[[1]][#[[0]]]&[C[Print]]
```

```
fun[SlotSequence[1 + 1]]&[1, 2]
```

```
(Slot[Slot[1]])&[1]
```

```

c) fg[x_Integer] := {x, Integer}
   fg[x_Times] := {x, Times}
   fg[x_Rational] := {x, Rational}
   fg[x_Divide] := {x, Divide}
   fg[x_] := {x, arbitrary};
   {fg[3],
    fg[Unevaluated[3/1]],
    fg[Unevaluated[3]],
    fg[Divide[3, 1]],
    fg[Unevaluated[Divide[3, 1]]],
    fg[Unevaluated[Rational[3, 1]]]}

d) f[x_x] := x[[1]]
   f[x[3]]
   f[x[x]]
   f[x]
   Clear[f];
   f[Head_] := Head[Head]
   f[Sin[Cos]]

e) f[Symbol_Symbol] := Symbol
   f[Sin]

f) f[Integer_] := Integer
   f[3]
   f[x]

g) clock[Print[4]; #]&[Print[3]; #]&[Print[2]; #]&[Print[1]; #]&[hand]

h) #1&[]
   #2&[0]
   #0&[]
   ##&[]
   ##0&[]
   ###&[2, 3, 4]
   xa&b&c&[d]
   (#&)[2][3]
   ((#&)&)[1][2][3]

i) Evaluate[D[#, x]]&

j) f[x_Pattern] := x^2
   f[x_Integer] = f[x_Integer]
   f[3]

k) f[x_] := Evaluate[Expand[x]]
   f[(x + y)^3]

l) 2 // ((1 ~ #1 + #2& ~ #1)& @ #1)&

m) a = x_y; f[a] := x^2; f[a]

n) N[1[1]] N[1[1]]

```

```

o) f[x_Blank] := x^2;
   f[x_y[x]] := x^-2;
   f[x_(y)] := x
   f[Pattern[x, Blank[Blank[y]]]] := x^-1
   f[_] + f[y[z]] + f[y[z][x]] + f[_head] +
   f[Blank[y]] + f[Blank[y][1]] + f[2 y[5]]

p) Flat[Flat[Flat, Flat]]

q) v := (Remove[a]; 1); a = 2;
   v + a
   Remove[a, v];
   v := (Remove[a]; 1); a = 2;
   a + v

r) Function[x, x] - Function[y, y]
   Function[Function, #^2&][Depth]

s) f[x_] := (f[Evaluate[Pattern[y, Blank[Head[x]]]]) := y + Head[x]; f[x^2])
   f[2]

t) f[x_][y_] := f[y][x]
   f[1][2]

u) Function[functionBody, Function[s, functionBody][3]][11 s + 111 s^11]

   Function[{functionArg, functionBody}, Function[functionArg,
   functionBody][3]][s, 11 s + 111 s^11]

v) Function[f, (# f)&[3]][#]

w) # & & & & & [1][2][3][4][5][6]

x) noGo[x_] := (x = 11)
   myNewVar = 1; noGo[Unevaluated[myNewVar]]

   Remove[noGo]
   SetAttributes[noGo, HoldFirst]
   noGo[x_] := (x = 11)
   myNewVar = 1; noGo[myNewVar]

   myNewVar = 1; noGo[Unevaluated[myNewVar]]

   myNewVar = 1; noGo[Evaluate[myNewVar]]

y) Function[c, Slot[c] SlotSequence[c]&[1, 2, 3], Listable][{1, 2, 3}]

   Function[Slot, Slot[1]]&[C][1][1] - Function[Slot, Slot[1]][C]

z) Slot[1/2 + 1/2]&[1, 2]

   k = 1; k = 2; (#1 - #k) + (#2 - #k)

```

2.^{L2} $(a + b)^{2n+1}$, Laguerre Polynomials

a) Expressions of the form $(a + b)^{2n+1}$ can be written in the following way, at least for the odd powers given here:

$$(a + b)^3 = a^3 + b^3 + 3ab(a + b)(a^2 + ab + b^2)^0$$

$$(a + b)^5 = a^5 + b^5 + 5ab(a + b)(a^2 + ab + b^2)^1$$

$$(a + b)^7 = a^7 + b^7 + 7ab(a + b)(a^2 + ab + b^2)^2,$$

Program a function that tries to write expressions of the form $(var_1 + var_2)^{exp}$ in this way, and determine whether this can be done for $(a + b)^9$, $(a + b)^{11}$, ..., etc.

b) Use the symbolic formula [46★], [45★]

$$L_n^{(a)}(z) = \frac{(-1)^n}{n!} \exp\left(-\frac{\partial}{\partial z} z \frac{\partial}{\partial z} + a \frac{\partial}{\partial z}\right) z^n$$

to derive explicit forms of the first few ($n = 0, 1, \dots, 5$) Laguerre polynomials $L_n^{(a)}(z)$.

3.^{L1} $\frac{d}{dax} \int^{ax} f(y) dy$

Given a function of the form $f[x_] := \text{notAnalyticallyIntegrable}$, examine the results of $D[\text{Integrate}[f[x], x], x]$ and $D[\text{Integrate}[f[a x], a x], a x]$. How must D and/or Integrate be modified to get the desired results in the second case?

4.^{L1} `Pattern[name, _]`

a) `⊕[Pattern[2, _]] = 2^2;`

??⊕

`{⊕[2], ⊕[3], ⊕[Pattern[2, _]]}`

b) `⊕[Pattern[I, _]] = I^2;`

??⊕

`{⊕[2], ⊕[I]}`

c) `⊕[Pattern[I, _]] := I^2;`

??⊕

`{⊕[2], ⊕[I]}`

d) `⊕[Pattern[a[2], _]] := a[2]^2;`

??⊕

`{⊕[2], ⊕[Pattern[a[2], _]]}`

d) `FullForm[_[_]]`

5.^{L1} Puzzles

What could the input `In[1]` be to get the following two sets of inputs and outputs?

a)

`In[2] := b[c]`

`Out[2] = b[c]`


```
In[3] := Head[b[c]]
Out[3] = d
```

b)

```
In[2] := Remove[f, x]
In[3] := f[x_] := x^2
In[4] := f[2]
Out[4] = 16
```

c) Find a *Mathematica* expression *expression* for which *expression*[[1 , 1]] and *expression*[[1]][[1]] are different.

d) Given the definition `f[x_Real] := x^2`, can one give any argument *arg* that is not a real number, such that `f[arg]` evaluates to its square?

e) What will the result of this input be?

```
FixedPoint[Head, arbitraryExpression]
```

f) What will the results of the following three inputs be?

```
Function[{s}, OIOI[s]][Unevaluated[κ]]
```

```
Function[{s}, OIOI[s]][Unevaluated[Unevaluated[κ]]]
```

```
Function[{s}, OIOI[s]][Unevaluated[Unevaluated[Unevaluated[κ]]]]
```

g) What will be the result of the following input?

```
DirectedInfinity[Infinity[[1]]]
```

h) What could have been the input `In[1]` to get the following inputs and outputs?

```
In[2] := a^2
Out[2] = b
In[3] := Clear[a]
Out[3] = a
In[4] := Remove[a]
Out[4] = a
```

i) Predict the result of the following input.

```
f[_] := (f[_] := #0[# + 1]; # + 1)&[1]
```

```
f[f[f[f[f[2]]]]]
```

6.^{L2} Different Patterns

For the following function definitions, find realizations for which the corresponding pattern matches.

a) `f[x_, a[b_, c_]_] := f[x, a, b, c]`

b) `f[x_, a_[b_, c_]] := f[x, a, b, c]`

c) `f[a_ b_ c_] := f[a, b, c]`

d) `f[_ _] := f`

7.^{L1} Plot [numberFunction]

Define a function $f(x)$ that gives 3 for integer arguments, 2 for rational arguments, and 1 for real arguments. Try to plot this function using `Plot[f[x], {x, -3, 3}]`. What can one conclude from this attempt to make a `Plot`?

8.^{L1} Tower of Powers

What is the limit value of the following power tower?

$$(\sqrt{2})^{(\sqrt{2})^{(\sqrt{2})^{(\sqrt{2})^{\dots}}}}$$

Calculate numerical values for the first few iterations.

9.^{L1} Cayley Multiplication

Implement the (associative) Cayley multiplication \mathcal{CT} (short for `CaleyTimes`). This operation is binary with the following multiplication table.

<i>a</i>	<i>b</i>	<i>c</i>	<i>e</i>
<i>a</i>	<i>e</i>	<i>c</i>	<i>b</i>
<i>b</i>	<i>c</i>	<i>e</i>	<i>a</i>
<i>c</i>	<i>b</i>	<i>a</i>	<i>e</i>
<i>e</i>	<i>a</i>	<i>b</i>	<i>c</i>

Find the following result.

$\mathcal{CT}[a, b, c, a, c, e, a, c, b, b, c, a, e, a, c, c, a, b, a, c,$
 $a, c, a, e, b, b, a, a, e, c, b, b, a, a, c, e, e, e, a, a,$
 $b, b, b, a, b, c, b, c, a, a, c, c, c, b, a, a, e, e, c]$

How often are the various multiplication rules applied?

CHAPTER 4

Exercises

1.^{L1} Explain the Errors

Why do the following inputs generate messages?

a) $a + b = 5$

```

b) a = 3; a[x_] = x
c) (3 + 5)[[1]]
d) f[x_] = x_
e) expression = TreeForm[6 + u^(Sin[r + 78 z^z])]; expression[[2]]
f) x = With[{x = x}, x^12]
g) Set[##]&[1, 2]
h) f[1]([1], [1])
i) Remove[f]; f[x_] := (f[y_] = f[y]); f[1]
j) Remove[f]; f[x_] := x[f[x[f]]]; Short[f[1], 12]
k) Remove[f]; f[x_] := (f[y_] = f[x][y]); f[1]
l) Length[Sin[1, 2, 3, 4]]
m) headOnly = a1b2c3[1][2][3]; headOnly[[2]]
n) (#2. + #1.)&[1, 2]
o) Remove[f]; f[x_] = Function[x, x^2]; f[1]
p) Remove[x, f1, f2]; x /: f1[_ , f2_[x], _] := f1 f2 x
q) Remove[p]; p = 1; p /: Hold[p] = 0; 1/Hold[p]
r) mySet = Set; myVar = 1; #1[#2, #3]&[mySet, myVar, 2]
s) Module[{Slot}, (#1^2&[3])[[1, 1]][[2]]
t) (f1[x_] = Block[{x = x}, x^2];
    f2[x_] := Block[{x = x}, x^2];
    {f1[2], f2[2]})
u) Function[a, Block[{a}, a], {HoldAll}] @
    (Function[a, Function[a, a + a]][x][[1]])
v) Function[Slot[Slot[1]]][2]
    Block[{v = 1}, Slot[v]&[Pi][[1]] - (Evaluate[Slot[v]]&[Pi])]
w) Module[Evaluate[{a = 1}], a^2]
    Module[Unevaluated[Unevaluated[{a = 1}], a^2]

```

2.^{L1} Unevaluated and Evaluate

a) The standard procedure for the computation of a *Mathematica* expression is altered for expressions containing an Unevaluated. Examine the following, and draw some conclusions.

```
Plus[Unevaluated[1], Unevaluated[2]]
```

```
plus[Unevaluated[1], Unevaluated[2]].
```

b) Explain the result of `Nest[Set[Evaluate[Unique[x]], #]&, 1, 4]`. What happens in this construction without the `Evaluate`?

3.^{L1} `Alias[]`

Using `Information`, `?`, or `??` we can get some information on *Mathematica* commands. `Alias[]` provides an overview of those *Mathematica* commands for which an abbreviation exists. Examine them.

4.^{L1} Built-in `builtinCommand[]`

Examine how built-in commands react to the wrong number of arguments, for example, to none at all.

5.^{L1} Explain the Problem, Puzzle

a) The following simple implementation of an alternative to the function `plus` for adding two integers has problems with `plus[m[3], m[4]]`. Use `Trace` to see what happens.

```
Remove[plus];
```

```
SetAttributes[plus, {Flat, Orderless}]
```

```
plus[m[i_], m[j_]] := plus[m[i + 1], m[j - 1]]
```

```
plus[m[i_], m[0]] = m[i];
```

b) Find an expression `expr` that has zero length (meaning `Length[expr]` gives 0), small depth (meaning `Depth[expr]` is less or equal to 2) and is big (meaning `ByteCount[expr]` is $\geq 10^6$). (Don't use any tricks like unprotecting `Length` and/or `Depth` and/or `ByteCount`.)

6.^{L1} Predictions

a) Predict the result of the following inputs.

```
globalVar = True;
```

```
f[x_Symbol, n_Integer] :=
Module[{sum = 0}, globalVar = False;
  CheckAbort[Do[sum = sum + If[globalVar, 0, x[i]],
    {i, n}]; globalVar = True; sum,
  Print[Length[sum]];
  globalVar = True; Abort[]]]
```

b) Does the following input evaluate to 0?

```
Module[{x =  $\xi$ }, Function[x, x] - Function[x + 0, x] +
  Function[x, x + 0] - Function[Evaluate[x], x]]
```

c) Will the following input issue messages? If yes, what kind of messages are to be expected?

```
Block[{Message, C, Do},
  C[Sin[1, 1], 0/0, 0^0, Do[k, {k, 1, 2I}]]]
```

d) Predict the results of the following two inputs.

```
Table[ξ[1][1], {ξ[1][1], 3, 4}, {ξ[1], 1, 2}]
```

```
Table[ξ[1][1], {ξ[1], 1, 2}, {ξ[1][1], 3, 4}]
```

e) Predict the result of the following inputs.

```
f[SetAttributes[f, HoldAll], 1 + 1]
```

```
CompoundExpression[SetAttributes[g, HoldAll], g][1 + 1]
```

f) Predict the result of the following input.

```
Exp[2 I Pi] - (Exp := 2)/(I := Pi)
```

g) Will the following two inputs give the same result?

```
Sum[1/((k + 1/2)^2 + 1), {k, -Infinity, Infinity}]
```

```
Sum[1/(k^2 + 1), {k, -Infinity + 1/2, Infinity + 1/2}]
```

7.^{L2} Contexts

Predict the result of the following inputs.

a)

```
BeginPackage["question1`"]
f1::usage = " ... is the question here ..."
Begin["`Private`"]
f1[x_String] := (ToExpression[x]; xAx1 + xAx2)
End[]
EndPackage[]
```

```
f1["xAx1 = 1; xAx2 = 2; "]
```

```
f1["question1`Private`xAx1 = 1;
  question1`Private`xAx2 = 2; "]
```

b)

```
BeginPackage["question2`"]
f2::usage = " ... is also the question here ..."
Begin["`Private`"]
f2[x_String] := Module[{x1 = x, x2}, ToExpression[x]; x1 + x2]
End[]
EndPackage[]
```

```
f2["x1 = 1; x2 = 2; "]
```

c)

```
BeginPackage["question3`"]
f3::usage = " ... is still the question here ..."
Begin["`Private`"]
f3[x_String] := Module[{x = x}, ToExpression[x]; x]
End[]
EndPackage[]
```

```
f3["x"]
```

d)

```
xa = 5; xb = 6;
f4[x_String] := (Begin["context4`"]; ToExpression[x];
                 Print[ToExpression["xa + xb"]]; End[]; )
```

```
f4["xa = 1; xb = 2"];
```

```
f4["context4`xa = 1; context4`xb = 2"];
```

```
f4["xa = 11; xb = 22"];
```

e)

```
A`f[x_Real] := x
```

```
B`f[x_Integer] := x^2
```

```
$ContextPath = {"Global`", "System`", "A`", "B`"};
```

```
f[2] // N
```

8.^{L1} **2 + I versus Complex[2, 1]**

What happens to the input of $2 + I$ as compared with the input `Complex[2, 1]`?

9.^{L1} **Local Values in Block**

`Block` allows local values of variables. Which values (downvalues, ownvalues, ...) are local? When attributes are set inside a `Block` for a local variable, are they local too? What will be the result of evaluating `(a = 1; Block[{a}, Remove[a]]); a`?

10.^{L2} **Remove[f]**

What will be the result of the following inputs?

a) `(Remove[f]; f[x_] := x + 1; f[1] + f[1, 1])`

b) `Remove[f]`

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

```
f[1] + f[1, 1]
```

CHAPTER 5

Exercises

1.^{L1} myExpand

Write a function `myExpand` using `Rule`, which multiplies out polynomials and products.

2.^{L1} ReplaceAll versus ReplaceRepeated

Discuss the following replacements:

```
replacement = {x + 1 -> px};
```

```
1 + x + 1/(1 + x) + (1 + x)^(1 + x) + f[1 + x] /. replacement
```

```
Plus[1 + x, 1/(1 + x), (1 + x)^(1 + x), f[1 + x]] /. replacement
```

```
1 + x + 1/(1 + x) + (1 + x)^(1 + x) + f[1 + x] //. replacement
```

```
plus[1 + x, 1/(1 + x), (1 + x)^(1 + x), f[1 + x]] /. replacement
```

Here, the function `plus` should have the same attributes as the function `Plus`.

3.^{L1} All Other Patterns with s, t, _, _, :

Examine all of the ways of creating a syntactically correct *Mathematica* expression from `s`, `t`, `_`, `_`, or `s`, `t`, `_`, `_`, `:` using at most two blanks. From the 1440 possible combinations, about two-thirds as many syntactically correct expressions exist, which reduce to about 8% different ones. An implementation of a program producing them is given in the solution (its operation will become clear after the discussion in Chapter 6).

4.^{L1} $\cos(x)^n \rightarrow f(\sin(x))$

Consider the following sum:

$$\cos(x)^2 + \cos(x)^4 + \cos(x)^6 + \cos(x)^8 + \cos(x)^{10} + \cos(x)^{12} + \cos(x)^{14} + \cos(x)^{16}$$

Express this sum using only $\sin(x)^i$. Use a rule-based approach.

5.^{L1} a[a]

Examine the results of the following *Mathematica* inputs, and explain what happens.

a) `Clear[a]; a = a`

```

b) Clear[a]; a := a; a
c) Clear[a]; a[a_] = a; a[a]
d) Clear[a]; a[a_] := a; a[a]
e) Clear[a]; a = a == a; a
f) Clear[a]; a := a == a; a
g) Clear[a]; a := a == a; Unevaluated[a]
h) Clear[a]; a := a == a; Hold[a]
i) Clear[a]; a := Unevaluated[a] == Unevaluated[a]; a
j) Clear[a]; a := Unevaluated[a] == a; a
k) Remove[a, x, y, z, Aah]
   a/: b[x_][a] := Null /; (Clear[a]; b[y_][a][z_] = Aah[y, z]; False)
   b[a][a][a]

```

6.^{L1} Extended Equal

Modify the built-in function `Equal` so that equations can be added, multiplied, and raised to given powers. In addition, make it possible to add something to both sides or multiply both sides of an equation by a constant.

7.^{L2} Weights for Finite Differences

Finite difference methods [79★] of higher order provide an important alternative to the finite element method. To use them, we need corresponding weights. For the one-dimensional case, the following recurrence formulas hold for the weights $c_{i,j}^k$ of the nodes x_j ($j = 0, 1, \dots, n$) in the approximation of a k th derivative with a total of $i + 1$ nodes. Here, x_0 is the point at which the derivative is to be approximated:

$$f_{(i)}^{(k)}(x)|_{x=x_0} \approx \sum_{j=0}^{i+1} c_{i,j}^k f(x_j)$$

$$c_{i,j}^k = \frac{1}{x_i - x_j} (x_i c_{i-1,j}^k - k c_{i-1,j}^{k-1}), \quad j = 0, 1, \dots, i-1$$

$$c_{i,i}^k = \frac{\omega_{i-2}(x_{i-1})}{\omega_{i-1}(x_i)} (k c_{i-1,i-1}^{k-1} - x_{i-1} c_{i-1,i-1}^k)$$

$$\omega_i(x) = \prod_{j=0}^i (x - x_j)$$

(The order of the other nodes x_j is arbitrary.)

Find the associated initial conditions for these recurrence formulas, and implement the computation of the $c_{i,j}^k$. (For the derivation of these recurrence formulas, see [70★], [71★], [101★], [72★], [102★], [51★], [98★], [106★], and [62★].)

Use this finite difference approximation to calculate reliable values for the first 20 derivatives of $\hat{\zeta}(1/2)$. Here, $\hat{\zeta}(s)$ is $\hat{\zeta}(s) = s(s-1)\pi^{-s/2}\Gamma(s/2)\zeta(s)/2$ [89★], [56★], [93★] and fulfills the functional equation $\hat{\zeta}(s) = \hat{\zeta}(1-s)$. In the last equation,

$\Gamma(s)$ is the Gamma function (in *Mathematica* `Gamma[s]`) and $\zeta(s)$ is the Riemann Zeta function (in *Mathematica* `Zeta[s]`). What is remarkable about these derivatives?

8.^{L3} Operator Product, q -, h -Binomial Theorem, Ordered Derivative

a) Define a function `operatorProduct` describing the noncommutative, associative multiplication of operators. Suppose the operators are given in the form `o[operatorIndex]`. All quantities that do not depend on the operators (numbers, constants, variables) should be factored out (before computing the operator product). Implement the additivity and associativity and a way to multiply out positive integer powers of sums of operators. If the reader has an appropriate application of such operator products, implement it also.

b) The famous binomial theorem $(x + y)^n = \sum_{k=0}^n \binom{n}{k} y^k x^{n-k}$ has two very interesting generalizations for noncommuting x and y . In the case of $xy = qyx$ ($q \in \mathbb{C}$) [77★], [55★], the binomial theorem becomes the q -binomial theorem [100★], [68★], [50★], [99★], [80★], [82★], [54★], [96★], [88★], [48★], [103★], [60★]

$$(x + y)^n = \sum_{k=0}^n \begin{bmatrix} n \\ k \end{bmatrix}_q y^k x^{n-k}$$

$$\begin{bmatrix} n \\ k \end{bmatrix}_q = \frac{(q; q)_n}{(q; q)_k (q; q)_{n-k}}$$

$$(a; q)_n = \prod_{k=0}^{n-1} (1 - a q^k), \quad a \in \mathbb{C}, n \in \mathbb{N}.$$

How often do the transformation rules in the transformation of $(x + y)^{10}$ to expanded form get applied?

In the case of $xy = yx + h y^2$ ($h \in \mathbb{C}$), the generalization of the binomial theorem is the h -binomial theorem [52★], [75★]

$$(x + y)^n = \sum_{k=0}^n \begin{bmatrix} n \\ k \end{bmatrix}_h h^k \left(\frac{1}{h}\right)_k y^k x^{n-k}$$

$$(a)_n = \prod_{k=0}^{n-1} (a + k), \quad a \in \mathbb{C}, n \in \mathbb{N}.$$

For $1 \leq n \leq 8$, verify explicitly the q -binomial theorem and the h -binomial theorem by straightforward calculation. (For the q - h -binomial theorem, see [53★], and for other generalizations, see [91★].)

The q -binomial coefficients $\begin{bmatrix} n \\ k \end{bmatrix}_q$ appear, for instance, when q -differentiating q -functions. If $\frac{d_q f(x)}{d_q x}$ is the q -derivative of a function $f(x)$ defined by [81★], [67★], [78★], [66★], [57★], [83★]

$$\frac{d_q f(x)}{d_q x} = \frac{f(x) - f(qx)}{(1 - q)x}$$

(this derivative can be interpreted as a discrete derivative approximation after a change of variables [65★]) and $\frac{d_q^n f(x)}{d_q x^n}$ the n th q -derivative ($\frac{d_q^1 f(x)}{d_q x^1} = \frac{d_q f(x)}{d_q x}$)

$$\frac{d_q^n f(x)}{d_q x^n} = \frac{d_q \frac{d_q^{n-1} f(x)}{d_q x^{n-1}}}{d_q x}$$

then the following two identities hold:

$$\frac{d_q^n f(x) g(x)}{d_q x^n} = \sum_{k=0}^n \begin{bmatrix} n \\ k \end{bmatrix}_q \frac{d_q^{n-k} f(q^k x)}{d_q x^{n-k}} \frac{d_q^k g(x)}{d_q x^k}$$

$$\frac{d_q^n f(x)}{d_q x^n} = \frac{1}{(1-q)^n x^n} \sum_{k=0}^n (-1)^k \begin{bmatrix} n \\ k \end{bmatrix}_q q^{-(k-1)k/2 - (n-k)k} f(x q^k)$$

Check these two identities for $0 \leq n \leq 10$.

c) In [94★] an “ordered derivative” of an operator product $\hat{x}_{\alpha_1} \hat{x}_{\alpha_2} \dots \hat{x}_{\alpha_n}$ with respect to a sequence of corresponding classical symbol $x_{\beta_1} \dots x_{\beta_m}$ has been defined. The \hat{x}_α are assumed to be noncommuting and the x_α to be commuting. The “ordered derivative” δ operatorProduct / δ classicalSymbols is defined in the following way:

$$\frac{\delta \hat{x}_\alpha}{\delta x_\beta} = \begin{cases} 1 & \text{if } \alpha = \beta \\ 0 & \text{else} \end{cases}$$

$$\frac{\delta \hat{x}_\alpha}{\delta (x_{\beta_1} \dots x_{\beta_m})} = \prod_{k=1}^m \frac{\delta \hat{x}_\alpha}{\delta x_{\beta_k}}$$

$$\frac{\delta (\hat{x}_{\alpha_1} \dots \hat{x}_{\alpha_n})}{\delta (x_{\beta_1} \dots x_{\beta_m})} = \sum_{k=0}^m \frac{\delta (\hat{x}_{\alpha_1} \dots \hat{x}_{\alpha_l})}{\delta (x_{\beta_1} \dots x_{\beta_k})} \frac{\delta (\hat{x}_{\alpha_{l+1}} \dots \hat{x}_{\alpha_n})}{\delta (x_{\beta_{k+1}} \dots x_{\beta_m})}$$

where l in the last definition is an arbitrary integer between 1 and $n-1$ (the results of the “ordered derivative” does not depend on l). Implement a function that carries out the “ordered derivative”.

The \mathfrak{f} in the “ordered derivative” of

$$\frac{\delta (\hat{x}_{\alpha_1} \dots \hat{x}_{\alpha_n})}{\delta (x_{\beta_1} \dots x_{\beta_m})} = \mathfrak{f} \text{ productOfThe } x_\alpha s$$

(*productOfThe* is proportional to the “ordinary derivative” $\partial(x_{\alpha_1} \dots x_{\alpha_n}) / \partial(x_{\beta_1} \dots x_{\beta_m})$ with all products of the same symbol collapsed) counts how many possibilities exist to delete the string of $x_{\beta_1} \dots x_{\beta_m}$ from the string $x_{\alpha_1} \dots x_{\alpha_n}$. (The x_{β_1} must appear in order, but not contiguously in $x_{\alpha_1} \dots x_{\alpha_n}$.) Check this statement for

$$\frac{\delta (\hat{x}_1 \hat{x}_2^2 \hat{x}_3^3 \hat{x}_4^4 \hat{x}_5^5 \hat{x}_6^4 \hat{x}_7^3 \hat{x}_8^2 \hat{x}_9)}{\delta (x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9)}$$

d) Let $\mathbf{A}(t)$ be a parametrized, nonsingular $n \times n$ matrix. Using $\partial(\mathbf{A}(t) \cdot \mathbf{A}(t)^{(-1)} - \mathbf{1}) / \partial t = \mathbf{0}$ ($\mathbf{0}$ being the n -dimensional matrix with all elements being 0 and $\mathbf{1}$ being the n -dimensional identity matrix) can derive the expression $\partial \mathbf{A}(t)^{(-1)} / \partial t = -\mathbf{A}(t)^{(-1)} \cdot (\partial \mathbf{A}(t) / \partial t) \cdot \mathbf{A}(t)^{(-1)}$ for the derivative of the inverse matrix $\mathbf{A}(t)^{(-1)}$ (here differentiation with respect to t is understood componentwise). Calculate the explicit form of $\partial^5 (\mathbf{A}(t)^{(-1)})^5 / \partial t^5$. Simplify the result when all occurring matrices commute. Count how often the needed definitions are applied during the calculation.

9.L2 Patterns and Replacements

Program solutions to the following problems; base the programs on pattern matching and the use of replacement rules.

a) Given a list of elements (some of which may appear more than once), construct a list containing all (different) permutations of the elements.

- b) Given a list of the form $\{integer, nZeros\}$, for example, $\{23, 0, 0, 0, 0, 0\}$, construct all lists of integers a_i ($i = 1, \dots, n + 1$) (i.e., of the same length as the original list) with $\sum_{i=1}^{n+1} a_i = integer$ (i.e., which have the same sum as the original list). Put the a_i in increasing order: $a_i \leq a_{i+1}$, ($i = 1, \dots, n - 1$).
- c) Given lists of the form $\{0, \dots, 0, number_1, 0, \dots, 0, number_2, \dots, number_n, 0, \dots, 0\}$ and $\{newNumber_1, \dots, newNumber_n\}$, construct a new list from the first list by replacing $number_i$ by $newNumber_i$ ($i = 1, n$) $\{a_1, a_2, \dots, a_{n-1}, a_n\}$. (This problem was proposed by R. Gaylord.)
- d) Given a list of positive integers, construct a list containing all pairs of numbers with no common factor.
- e) Given a list of positive integers in decreasing order, construct the Ferrer conjugate of this list. The Ferrer conjugate is defined in the following way [104★], [61★], [73★], and [49★]: Associate with the list $\{n_1, n_2, \dots, n_k\}$ an array of dots: n_1 in the first row, n_2 in the second, and so on. Then, the list of the lengths of the columns, starting from the left, is the Ferrer conjugate. An example: the Ferrer conjugate of $\{5, 3, 2, 1\}$ is $\{4, 3, 2, 1, 1\}$ as can be seen by

```

. . . . .
. . .
. .
.

```

10.^{L1} Hermite Polynomials, Peakons

- a) The Hermite polynomials $H_n(x)$ satisfy the following identity: $x H_n(x) = H_{n+1}(x)/2 + n H_{n-1}(x)$, $n \in \mathbb{N}$.

Program the repeated use of this identity in terms of the form $x^m H_n(x)$ to write them as linear combinations of the Hermite polynomials without x -dependent prefactors. Make the program work for user-defined objects $H[n, z]$. (Do not modify the built-in function `HermiteH`.)

- b) Show that $\psi(x, t) = c \exp(-|x - ct|)$ is a solution of the nonlinear Camasso–Holm partial differential equation [58★], [59★], [85★], [86★], [76★], [69★], [84★], [74★], [63★], [47★], [64★], [87★]

$$\frac{\partial \psi(x, t)}{\partial t} - \frac{\partial^3 \psi(x, t)}{\partial x^3} + 3 \psi(x, t) \frac{\partial \psi(x, t)}{\partial x} = 2 \frac{\partial \psi(x, t)}{\partial x} \frac{\partial^2 \psi(x, t)}{\partial x^2} + \frac{\partial^3 \psi(x, t)}{\partial x^2 \partial t} + \psi(x, t) \frac{\partial^3 \psi(x, t)}{\partial x^3}.$$

11.^{L1} `f[x___] := ...`

What outputs correspond to the following inputs?

- a) `f[x___] := x + 1`

```

f[]
f[1, 2, 3]

```

- b) `f[x___] := x - 1`

```

f[]
f[1, 2, 3]

```

- c) `f[x___] := Subtract[x, 1]`

```

f[]
f[1, 2, 3]

```

- d) `f /: HoldPattern[HoldPattern[Verbatim[HoldPattern[f]]]] := 4`

```

HoldPattern[f]

```

12.^{L1} Result and Error Messages

Predict the output and warning/ messages generated when evaluating the following:

```
{1, 2} //. {{x___, y___} :> ({x, Unique[c], y} /;
              (Head[{x}][[-1]] != Symbol &&
               Head[{y}][[ 1]] != Symbol))}
```

13.^{L1} Patterns

a) Construct at least five different patterns that match integers greater than 1 and less than 9.

b) To obtain the result $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11\}$, identify which arguments must be given to `f`, defined by

```
f[Condition[Condition_, Condition; True],
  Optional[Blank_, Optional],
  Pattern[Pattern, Blank[Integer]],
  Four:(4 | 4.),
  PatternTest[Pattern[PatternTest, Blank[]], PatternTest; True&],
  Alternatives:Alternatives[Alternatives, 6],
  Flat_Flat,
  Stub:Blank[Orderless[OneIdentity]],
  HoldPattern_HoldPattern?(# == #&),
  HoldPattern[Set[3, 4]] :=
  {Condition, Blank, Pattern, Four, PatternTest,
   Alternatives, Flat[[1]], Stub[[1]], 9, HoldPattern[[1]]}
```

c) What are the results of the following inputs?

```
Remove[a]
```

```
SetAttributes[a, HoldAll]
```

```
f:a[a_] := Function[#, Hold[#], {HoldAll}][f]&[Unique[a]]
```

```
{a[a], a[b], a[2a], a[a + a]}
```

```
Remove[a]
```

```
SetAttributes[a, HoldAll]
```

```
f:a[a_] := Function[#, Hold[#], {HoldAll}][f]&[Unique["a"]]
```

```
{a[a], a[b], a[2a], a[a + a]}
```

d) What are the results of the following inputs?

```
SetAttributes[AtomQ, HoldAll]
```

```
{AtomQ[1/2], AtomQ[1 + I]}
```

e) What is the result of the following input?

```
blank[Pattern[Blank, Blank[Blank]]] = Blank
```

```
blank[Blank[Blank]]
```

f) Predict the results and side effects of the following three inputs.

```
f1[x0_] := Block[{x = x0}, Print[C1]; x = x + 1; Print[C2] /; Positive[x]]
```

```
f1[-2]
```

```
f2[x0_] := Block[{x = x0}, ToExpression[
    "Print[C1]; x = x + 1; Print[C2] /; Positive[x]"]]
```

```
f2[-2]
```

```
f3[x0_] := Block[{x = x0}, (Print[C1]; x = x + 1;
    condition[Print[C2], Positive[x]]) /.
    condition -> Condition]
```

```
f3[-2]
```

14.^{L1} Replacements

Explain what happens when evaluating the following expressions.:

a) {1, 2, 3, 4, 5} //. {a___, b_, c_, d___} :>
If[b > 2, {b, c, d}, {a, b, c, d}]

b) {1, 2, 3, 4, 5} //. {a___, b_, c_, d___} :>
If[b > 2, {b, c, d}, {a, b, c, d}]

c) {1, 2, 3, 4, 5} //. {a___, b_, c_, d___} :> {b, c, d} /; b > 2

d) {1, 2, 3, 4, 5} //. {a___, b_, c_, d___} :> {b, c, d} /; b > 2

e) {1, 2, 3, 4, 5} //. (({a___, b_, c_, d___} /; b > 2) :> {b, c, d} /; b > 2)

f) {1, 2, 3, 4, 5} //. (({a___, b_, c_, d___} /; b > 2) :> {b, c, d} /; b > 2)

15.^{L1} Puzzles

a) What might have been the `In[1]` in the following example?

```
In[2]= a
Out[2]= True
```

```
In[3]= And[a, a]
Out[3]= False
```

Give at least three different possibilities for `In[1]`. Find some solutions that do not involve unprotecting built-in functions.

b) Predict the result of the following input:

```
(Im[3 I] == 0) // Function[{x}, Block[{I}, x], {HoldAll}]
```

c) Predict the result of the following input:

```
Hold[With[{z = Abort[]}, z^2]] /. z_?Quit :> Quit[]
```

d) Have a look at:

```
On[]; 2/3 === Unevaluated[2/3]
```

We see there the line:

```
      2      2
```

```
SameQ::trace: - === - --> False.
```

```
      3      3
```

Explain this “surprising” printout!

e) For which built-in symbols *builtInSymbol* does *builtInSymbol* == *builtInSymbol* not yield True? Why?

f) What will be the result of the following input?

```
(X[_?(# === _?#0&), C_ /; MatchQ[C, _ /; MatchQ[C, _]]] := Y;
 X[_?(# === _?#0&), C_ /; MatchQ[C, _ /; MatchQ[C, _]])
```

g) For many cases `IntegerQ[x]` will return True or False. Find three different values for *x* such that something else is returned.

h) Evaluating `f[a, b]` after making the function definition

```
SetAttributes[f, {Flat, OneIdentity}]
```

```
f[ξ_] := ξ
```

leads to iteration errors. How can one change the definition `f[ξ_] := ξ` to prevent this problem?

i) Let *g*, *h*, and *i* in the following be all possible combinations of `HoldPattern` and `Verbatim`. For which of the eight possible combinations of *g*, *h*, and *i* does the input

```
f[g[h][x_]] := x; f[i[1]]
```

return 1?

j) After defining *f* by

```
With[{a = x}, HoldPattern[f[y_, g[y_] = y^2]] := a]
```

find arguments, such that the definition for *f* will be used.

k) Predict the result of the following input.

```
Block[{Function}, (#&[2]) /. Function -> Print]
```

l) Predict the side effects of evaluating the following:

```
SetAttributes[{M, TagUnset, ToString}, HoldAllComplete]
```

```
M[e_] := (e /: HoldPattern[e:h_[___], e, ___]) :=
  (Print["Found: ", h, " ", HoldForm[e]]);
```

```
ToExpression[# <> " /: HoldPattern[e:h_[___], " <>
# <> ", ___]] =."]&[ToString[e]];
M[h, e]; e))
```

```
M[h_, e_] := (h /: HoldPattern[e:l_[___], e, ___]] :=
(Print["Found: ", HoldForm[e]];
TagUnset @@ {h, UpValues[h][[1, 1, 1]]}; M[l, e]; e))
```

```
M[x];
α[1, β[y], a[b[c[2, d[f[x], 1]]]]]
```

m) A bivariate function $f(x, y)$ can be written in separated form $f(x, y) = \varphi(x)\phi(y)$ in a neighborhood of a point $\{x_0, y_0\}$ if $f(x_0, y_0) \neq 0$ and [97★], [90★], [105★], [95★], [92★]

$$f(x, y) \frac{\partial^2 f(x, y)}{\partial x \partial y} = \frac{\partial f(x, y)}{\partial x} \frac{\partial f(x, y)}{\partial y}.$$

What is “wrong” with the following function `separableVariablesQ` that checks if a function f of the two variables x and y can be written in separated form?

```
separableVariablesQ[f_, {x_, y_}, {x0_, y0_}] :=
(Simplify[f /. {x -> x0, y -> y0}] != 0) &&
Simplify[f D[f, x, y] - D[f, x] D[f, y]] === 0
```

n) Predict the result of the following input.

```
SetAttributes[PrimeQ, HoldAll]
PrimeQ[2 + 3 I, {GaussianIntegers -> True}]
```

o) What might have been the `In[1]` in the following example? (No unprotecting of built-in symbols was involved.)

```
In[2]:= {NumericQ[%], NumberQ[%], MemberQ[%, _?InexactNumberQ],
StringLength[StringDrop[ToString[
DownValues[In][[$Line - 1]]], 22]],
Context /@ Cases[%, _Symbol, {-1}, Heads -> True]}
Out[2]= {True, False, True, 9, {System`}}
```

p) Construct an example of expressions a and b such that `FreeQ[a, b]` yields `False` and `Position[a, b]` yields `{}`.

16.^{L1} Evaluation Sequence

Discuss the evaluation sequence in the following four examples:

```
(f[x_] := g) /; c
```

```
(f[x_] /; c) := g
```

```
(f[x_] := g /; c)
```

```
(f[x_ /; c] := g)
```

17.^{L1} Nested Scoping

Predict the results of the following inputs.

- a) `Clear[f]; f[x_] := Function[x, x]; f[y]`
- b) `With[{x = z}, Function[x, x]]`
- c) `Function[x, x] /. x -> z`
- d) `Clear[f]; Function[x, f[x_] := x^2][y]; DownValues[f]`
- e) `Clear[f]; With[{x = y}, f[x_] := x^2]; DownValues[f]`
- f) `Function[y, Function[x, x + y]][x]`
- g) `Clear[f]; f[y_] := Function[x, x + y]; f[x]`
- h) `Clear[f];
Module[{x, y, z = a}, f[x, y_, z] := Function[x, x + y + z];
DownValues[f]`
- i) `Clear[f];
With[{z = a},
Module[{x, y}, f[x, y_, z] := Function[x, x + y + z]]];
DownValues[f]`

18.^{L1} Why {b,b}?

Explain why it might be possible to get the following behavior. (For reproducing this behavior, the reader might have to redo the `Table[a, {10000}] // Union` line a few times until one get the result shown here.)

```
In[1]:= a := b /; EvenQ[Last[Date[]]]
```

```
In[2]:= Table[a, {10000}] // Union
```

```
Out[2]= {b, b}
```


CHAPTER 6

Exercises

1.^{L2} Benford's Rule

Given a long list of empirical data (e.g., lengths of rivers, areas of deserts and seas, addresses, bank account balances, physical data, chemical data), check whether this data satisfies Benford's rule: The probability distribution of the appearance of the digit i ($1 \leq i \leq 9$) in the first place in a data entry is $\log_{10}(1 + 1/i)$, where all zeros to the left of the first significant digit are ignored. For details on Benford's rule, see [142★], [140★], [192★], [111★], [169★], [132★], [176★], [168★], [170★], [171★], [121★], [125★], [156★], [120★], [119★], [123★], [141★], [146★], [151★], [172★], and [163★] and the references therein.

2.^{L1} Map, Outer, Inner, and Thread versus Table and Part, Iteratorless Generated Tables, Sum-free Sets

a) Compare the computational times for `Map`, `Outer`, `Inner`, and `Thread` on reasonably-sized vectors to those for `Table`, `Do`, and `Part` using analogous constructions.

b) Write a function that generates the same output as

```
Table[f[i1, i2, ..., in], {i1, 1, ..., m}, {i2, 1, ..., m}, ..., {in, 1, ..., m}]
```

but which does not contain any explicit iterator variable.

c) Given a square matrix \mathbf{A} of dimension d with elements a_{ij} and a vector of operators \mathbf{f} of length d with elements f_j , form a new matrix \mathbf{B} with elements $b_{ij} = f_j(a_{ij})$. Write several different programs that form the matrix \mathbf{B} , and compare their timings for some matrices \mathbf{A} and vectors \mathbf{f} of different dimensions.

d) Given a set \mathcal{S}_o of positive integers $\{n_1, \dots, n_o\}$, recursively enlarge this set by adding the smallest integer that cannot be expressed as a sum $n_i + n_j$, $1 \leq i, j \leq o$ [131★].

Implement the recursive enlargement first using a procedural (list-based) approach and then using a caching approach. Compare the timing of the two approaches for the starting set of the first ten primes for 2000 recursive enlargements.

3.^{L1} Index

Create an index for the *Mathematica* commands that are introduced in this book. It should consist of a list of the form $\{\dots, \{"command_i", "chapterSectionSubsection_i"\}, \dots\}$. The function `whereIntroduced[command]` should give the number of the section where the command is introduced. Check that no command was misspelled when it was introduced. Which commands were introduced twice? The list of commands can be found in the package `ChapterOverview``.

4.^{L3} Functions Used Too Early?, Check of References, Closing]], Line Lengths, Distribution of Initials, Check of Spacings

- a) In the preface, we stated our aim that every time a command is used in this book, it should already have been discussed. Create a *Mathematica* program to check for specific examples to see how close we came to our goal. Collect all commands that have been introduced in gray boxes in a list `alreadyIntroducedCommands`. The command `$Pre` might be useful here.
- b) This *GuideBooks* have many references. Check if each mentioned reference is really present and if each reference is at least mentioned once. Which journals are the most cited? What is the number of electronic papers referenced and how did the fraction of electronic papers change over the last years?
- c) What are the most common first letters of the initials and last names of all authors of the quoted papers and books?
- d) What is the distribution of the line lengths used in the inputs of this book? How much white space (in the form of raw space characters) is on average present in the inputs? What is the average density of code comments?
- e) Brackets are very prominent in *Mathematica* code. Not more than two opening brackets can occur in a row, but arbitrarily many closing brackets can occur in a row. Analyze the literal inputs of the *Mathematica GuideBooks* to determine how often n closing brackets occur. If the inputs had been expressed in `FullForm`, how often would n closing brackets occur?
- f) As discussed in the Introduction, the inputs of the *GuideBooks* are in `InputForm`. To make the inputs as easy as possible to read, care has been taken to format them properly. This includes white spaces after all commas, white spaces around operators with relatively low binding power (such as `->` or `/.`). Write a program that checks for violations of such spacing rules and check all inputs from this volume of the *GuideBooks*.
- g) If one considers language as a network and the words as vertices, one can analyze the distribution of neighbors in this network. The natural interpretation of neighbors of a word are the preceding and postceding words. (Viewing sentences as natural units of a language, the first word of a sentence does not have a predecessor and the last word does not have a postceder [165★].) Analyzing large amounts of sentences and graphing the resulting (binned) frequency of the number of neighbors versus the number of neighbors in a double logarithmic plot shows two clearly different linear regimes [127★]. Analyze the texts of the *GuideBooks* and see if, despite this relatively small amount of data and the nonnativeness of the author, the two different power laws are nevertheless present.

5.^{L1} Tube Points

Write two different programs to solve the following problem. Suppose we are given lists of the form

```
points = {p1, p2, p3, ..., pn}
radii  = {r1, r2, r3, ..., rn}
vecv   = {v1, v2, p3, ..., vn}
vecu   = {u1, u2, u3, ..., un}
cos     = {c1, c2, c3, ..., cm}
sin     = {s1, s2, s3, ..., sm}
```

Here, p_i, v_i, u_i are vectors of the form $\{px_i, py_i, pz_i\}, \{vx_i, vy_i, vz_i\}, \{ux_i, uy_i, uz_i\}$. The px_i, \dots are atomic objects (in a typical application, real numbers); the c_i, s_i, r_i are assumed to be atomic objects.

Create a list of the following form:

```
{{p1 + r1 c1 v1 + r1 s1 u1,
  p1 + r1 c2 v1 + r1 s2 u1,
  p1 + r1 c3 v1 + r1 s3 u1, ...}
```

```

p1 + r1 c_m v1 + r1 s_m u1 } ,
{p2 + r2 c1 v2 + r2 s1 u2 } ,
p2 + r2 c2 v2 + r2 s2 u2 } ,
p2 + r2 c3 v2 + r2 s3 u2 } , ... ,
p2 + r2 c_m v2 + r2 s_m u2 } ,
:
{pn + rn c1 vn + rn s1 un } ,
pn + rn c2 vn + rn s2 un } ,
pn + rn c3 vn + rn s3 un } , ... ,
pn + rn c_m vn + rn s_m un } } .

```

($p_i + r_i c_j v_i + r_i s_j u_i$ is a list (head List) with three elements.)

6.^{L1} All Subsets

Explain the operation of the following command `allSubsets[list]`, which produces all subsets of a given set *list*, including the empty set and the set *list* itself. Here is the implementation (coming from [187★]):

```

allSubsets[l_List] :=
Sort[Distribute[{{}, {#}} & /@ Union[l], List, List, List, Union]]

```

Use this function definition to implement a one-liner for the sums

$$\mathcal{A}(k_1, k_2, \dots, k_n) = \frac{1}{K} \sum_{j=0}^{K-1} \prod_{m=1}^n \left\lfloor \frac{k_m j}{K} \right\rfloor$$

where $K = \prod_{j=1}^n k_j$. This sum can be expressed as [181★]

$$\mathcal{A}(k_1, k_2, \dots, k_n) = \prod_{j=1}^n (k_j - 1) + \sum_{\{k_{i_1}, \dots, k_{i_m}\}} (-1)^m \sum_{j=0}^{(k_{i_1}, \dots, k_{i_m})-1} \prod_{h=i_m+1}^n \left\lfloor \frac{k_h j}{(k_{i_1}, \dots, k_{i_m})} \right\rfloor.$$

Here the outer sum runs over all nonempty subsets of the set $\{k_1, k_2, \dots, k_n\}$, and the inner product over all k_j not in a given subset. $(k_{i_1}, \dots, k_{i_m})$ denotes the greatest common divisor of the numbers $k_{i_1}, k_{i_2}, \dots, k_{i_m}$. Calculate $\mathcal{A}(p_1, p_2, \dots, p_{10})$ where p_k is the k th prime.

7.^{L1} Moessner's Process, Ducci's Iterations

a) Write all integers in natural order. Then delete every second one, and form the following sequence of partial sums:

start sequence	1	2	3	4	5	6	7	8	...
delete every second element	1		3		5		7		...
form new partial sums	1		4		9		16		...

They are all squares. Now, delete every third number of the initial sequence, and form the partial sums. If we then strike every second number and again form the partial sums, we get a sequence of cubes.

start sequence	1	2	3	4	5	6	7	8	9	...
delete every third element	1	2		4	5		7	8		...
form new partial sums	1	3		7	12		19	27		...
delete every second element	1		7		19					...
form new partial sums	1		8		27					...

Find a functional program for this process that first deletes every i th number and then produces j numbers. Conjecture if $i = 4$ and $i = 5$ lead to fourth and fifth powers?

b) Take four positive integers $m, n, p,$ and $q,$ and form four new integers $|m - n|, |n - p|, |p - q|,$ and $|q - m|.$ Iterate this process until it converges [157★], [143★], [161★], [147★], [118★].

8.^{L1} Triangles, Group Elements, Partitions, Stieltjes Iterations

Describe what the following pieces of code do.

a)

```
NestedTriangles[n_Integer?Positive] :=
(Function[{x, y}, x.#& /@ y] @@ #)& /@
  Distribute[{Table[{{ Cos[i Pi/2], Sin[i Pi/2]},
                    {-Sin[i Pi/2], Cos[i Pi/2]}}, {i, 0, 3}],
            Flatten[NestList[#/2&,
                            {{{1, 1}, {3, +1}, {1, 3}},
                             {{1, 0}, {2, -1}, {2, 1}}}, n], 1]], List]
```

Look at the output graphically using the following input.

```
Show[Graphics[Polygon /@ Triangles[6]],
      AspectRatio -> Automatic, PlotRange -> All]
```

b)

```
FixedPoint[Union[Flatten[Outer[Function[C, #]& @
                             Simplify[#1[#2[C]]]&, #, #]]&,
                {Function[C, -C], Function[C, (C + I)/(C - I)]}]
```

c)

```
PartitionsLists[n_Integer?Positive] := Drop[FixedPointList[
  Complement[Union[Flatten[ReplaceList[#,
    {{a___, b_, c_, d___} :> {a, b - 1, c + 1, d} /; b - c >= 2,
    {a___, b_, c:(d_ ...), e_, f___} :> {a, b - 1, c, e + 1, f} /;
    b - 1 == d == e + 1}]& /@ #, 1]], #]&,
    {{n, ##}& @@ Table[0, {n - 1}]]], -2]
```

d)

```
Unprotect[Table];
```

```
Table[body_, iters__, Heads -> l_List] :=
With[{d = Length[{iters}]},
Fold[Apply[First[#2], #1, {Last[#2]}]&, Table[body, iters],
      Reverse[MapIndexed[{#1, #2[[1]] - 1}&,
                        Take[Flatten[Table[1, {d}]], d]]]]]
```

```
Table[body_, iters__, Heads -> l_] := Table[body, iters, Heads -> {l}]
```

e)

```

S $\mathcal{A}$ [l_List] := With[{ $\lambda$  = Length[l]},
Module[{ $\mathbb{P}$  = NestList[Flatten[
Outer[Join, {#}, List /@ Range[Last[#] + 1,  $\lambda$ ], 1]& /@ #, 2]&,
List /@ Range[ $\lambda$ ,  $\lambda - 1$ ]},
FixedPointList[Function[ $\ell$ ,
Divide @@@ Partition[Append[Reverse[Apply[(Plus[##])/Length[{##}]&,
Apply[Times, Map[ $\ell$ [[##]]&,  $\mathbb{P}$ , {-2}], {2}], {1}]], 1], 2, 1]],  $\ell$ ]] /;
(Or @@ (InexactNumberQ /@  $\ell$ )) && (And @@ (NumericQ /@  $\ell$ ))

```

f)

```

pseudoRandomTree[kStart_] :=
Module[{ $r$ ,  $k$ ,  $y$ ,  $t$ },
 $r$  := If[IntegerPart[Abs[Sqrt[2] Sin[Pi  $k$  Sin[ $k = k + 1$ ]]]] === 0,
0, 2];
 $k$  = kStart;  $y$ [_] := -1;
 $t$  /: Line[{ $x$ _,  $y$ _},  $t$ []} :=
Table[{Line[{ $x$ ,  $y$ }, { $x + 1$ ,  $y[x + 1] = y[x + 1] + 1$ }},
Line[{ $x + 1$ ,  $y[x + 1]$ },  $t$ []}], { $i$ ,  $r$ };
tree = Line[{0, 0},  $t$ {}];
symmetrizeRules = Dispatch[Flatten [Function[ $l$ ,
(# -> (# - {0, 1[[-1, 2]]/2})& /@ 1] /@
Split[Union[DeleteCases[Level[tree, {-2}], {}]],
#1[[1]] === #2[[1]]&]]];
Graphics[tree /. symmetrizeRules /. Line[ $l$ _] :=> Line[{ $l$ }],
Frame -> True]]

```

9.^{L3} $\varepsilon\varepsilon \rightarrow \Sigma\delta\cdots\delta$, $\text{Tr}(\gamma_{\mu_1}\gamma_{\mu_2}\cdots\gamma_{\mu_{2n}})$, tanh Identity, Multidimensional Determinant

a) Implement the following identity (meaning the calculation of its right-hand side) for Levi-Civita tensors [177★] and [115★] $\varepsilon_{\nu\dots\pi}$:

$$\varepsilon_{\tau_1 \tau_2 \dots \tau_{r-1} \tau_r \nu_{r+1} \dots \nu_n} \varepsilon_{\tau_1 \tau_2 \dots \tau_{r-1} \tau_r \mu_{r+1} \dots \mu_n} = r! \{\delta_{\nu_{r+1} \mu_{r+1}} \delta_{\nu_{r+2} \mu_{r+2}} \cdots \delta_{\nu_n \mu_n}\}_{[\mu_{r+1} \dots \mu_n]}$$

The expression $\{expression\}_{[\mu_{r+1} \dots \mu_n]}$ denotes the Bach bracket and means a complete antisymmetrization in the variables $\mu_{r+1} \dots \mu_n$.

Here, $\delta_{\nu\mu}$ is the Kronecker symbol, and for all variables with double subscripts, we assume an implicit summation over 1 to *dimension*.

b) Given n matrices ${}^{(k)}A_i^j$ ($i, j, k = 1, \dots, n$) of dimensions $n \times n$, the expression [130★]

$$\{ {}^{(1)}A_{k_1}^{k_1} \cdots {}^{(n)}A_{k_n}^{k_n} \delta_a^b \}_{[k_1 k_2 \dots k_n, a]}$$

vanishes identically. Here the expression $\{expression\}_{[\mu_1 \dots \mu_n]}$ denotes again the Bach bracket and means a complete antisymmetrization in the variables $\mu_1 \dots \mu_n$, and δ_a^b is the Kronecker symbol. Summation from 1 to n is assumed for all doubly occurring indices. For $n = 2, 3, 4$ verify this identity by explicit calculation. Is $n = 5$ feasible for explicit verification?

c) In many quantum electrodynamics calculations, the trace of the product of Dirac matrices γ_μ , $\mu = 0, 1, 2, 3$ [116★] has to be calculated. A compact formula for this trace is [188★], [189★]

$$\text{Tr}(\gamma_{\mu_1} \cdot \gamma_{\mu_2} \cdot \dots \cdot \gamma_{\mu_{2n}}) = 4 \sum_{\text{all pairings}} \delta_{\text{pairing}} \prod_{\text{all pairs}} \eta_{\mu_i, \mu_j}.$$

Here, the summation extends over all permutations $\{\mu_{i_1}, \mu_{i_2}, \dots, \mu_{i_{2n}}\}$ of $\{\mu_1, \mu_2, \dots, \mu_{2n}\}$ such that $\mu_{i_1} < \mu_{i_3} < \dots < \mu_{i_{2n-1}}$ and $\mu_{i_1} < \mu_{i_2}, \mu_{i_3} < \mu_{i_4}, \dots, \mu_{i_{2n-1}} < \mu_{i_{2n}}$. The symbol δ_{pairing} is the signature of the permutation $\{\mu_{i_1}, \mu_{i_2}, \dots, \mu_{i_{2n}}\}$. The inner product extends over all pairs $\{\mu_i, \mu_j\}$. All of the indices μ_i run conventionally from 0 to 3. η_{μ_i, μ_j} is the metric tensor $\eta = \text{diag}\{-1, 1, 1, 1\}$.

Calculate the trace for the product of 2, 4, 6, and 8 Dirac matrices. Use the following representation of the Dirac γ matrices to check the results:

$$\gamma_0 = \begin{pmatrix} 0 & 0 & -i & 0 \\ 0 & 0 & 0 & -i \\ -i & 0 & 0 & 0 \\ 0 & -i & 0 & 0 \end{pmatrix}, \gamma_1 = \begin{pmatrix} 0 & 0 & 0 & -i \\ 0 & 0 & -i & 0 \\ 0 & +i & 0 & 0 \\ +i & 0 & 0 & 0 \end{pmatrix}, \gamma_2 = \begin{pmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & +1 & 0 \\ 0 & +1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{pmatrix}, \gamma_3 = \begin{pmatrix} 0 & 0 & -i & 0 \\ 0 & 0 & 0 & +i \\ +i & 0 & 0 & 0 \\ 0 & -i & 0 & 0 \end{pmatrix}.$$

d) The following identity holds for almost all complex z_k [179★]:

$$\prod_{\substack{k,l=1 \\ k < l}}^n \tanh(z_k - z_l) = \frac{1}{2^{\lfloor n/2 \rfloor} \lfloor n/2 \rfloor!} \sum_{\sigma} \text{signature}(\sigma) \prod_{k=1}^{\lfloor n/2 \rfloor} \tanh(z_{\sigma(2k-1)} - z_{\sigma(2k)})$$

The summation runs over all elements of the permutations σ of $\{1, 2, \dots, n\}$. Prove this identity for $n = 6$. (Do not use functions like `Simplify`, `Together`, `TrigToExp`, etc., but only functions discussed so far.)

e) The determinant of a (2D) $n \times n$ matrix \mathbb{A} with elements $a_{i,j}$ can be written in the form

$$\det(\mathbb{A}) = \varepsilon_{1,2,\dots,n} \varepsilon_{k_1,k_2,\dots,k_n} a_{1,k_1} a_{2,k_2} \dots a_{n,k_n}$$

where summation from 1 to n is understood for the doubly occurring variables k_1, \dots, k_n and $\varepsilon_{1,2,\dots,n}$ is fully antisymmetric. This suggests the generalization of the determinant for an d -dimensional (dD) $n \times n \dots \times n$ “matrix” \mathbb{A}_d with elements a_{i_1,i_2,\dots,i_d} of the form [148★], [164★], [150★], [166★], [139★], [183★]

$$\det(\mathbb{A}_d) = \prod_{j=1}^d \varepsilon_{k_1^{(j)}, k_2^{(j)}, \dots, k_n^{(j)}} \prod_{i=1}^n a_{k_i^{(1)}, k_i^{(2)}, \dots, k_i^{(d)}}$$

where $k_m^{(1)} = m$ and again summation from 1 to n is understood for the doubly occurring variables. Implement a function `MultidimensionalDet` that for a given dD matrix \mathbb{A}_d of size $n \times n \dots \times n$ calculates this multidimensional determinant.

10.^{L1} Digits in π , Mediant Insertion, Matrix Product

a) Let $l_{ij}(\pi)$ be the sequence of number pairs $\{\{i_1, j_1\}, \{i_2, j_2\}, \dots\}$ of the positions of the first appearance of the digit i after the digit j in the decimal expansion of π [167★], where $i_1 < j_1 < i_2 < j_2 < \dots$. Program the computation of the $l_{ij}(\pi)$.

b) Given a list l of (reduced) rational numbers, write a function that inserts the mediant between each two numbers of the list l . The mediant of two rational numbers p_1/q_1 and p_2/q_2 (where $\text{gcd}(p_1, q_1) = \text{gcd}(p_2, q_2) = 1$) is defined as the number $(p_1 + p_2)/(q_1 + q_2)$.

c) The constant e can be calculated through the following limit of matrix product [129★], [138★]

$$\begin{pmatrix} a_n & c_n \\ b_n & d_n \end{pmatrix} = \prod_{k=1}^n \begin{pmatrix} 2k & 2k-1 \\ 2k-1 & 2k-2 \end{pmatrix}$$

$$e = \lim_{n \rightarrow \infty} \frac{a_n + c_n}{b_n + d_n}.$$

(This is basically an unfolded continued fraction expansion). How large a n is needed to obtain 1000 correct digits of e ?

11.^{L1} d'Hondt Voting

Implement the d'Hondt's voting method. If possible, try not to use any temporary variables. The d'Hondt's voting method is as follows: Suppose a parliament with a given number of seats is to be filled with representatives from several parties on the basis of an election. Divide the number of votes received by each party by 1, 2, 3, etc. Put the resulting numbers in decreasing order (where each number is included according to its multiplicity). Now, assign one seat to the party with the largest number, one seat to the party with the second largest number, etc., until all seats have been assigned. If, at the end, more equal numbers than seats are available, choose the parties to get the seats randomly. (We discuss the generation of random numbers in detail in the next chapter, so either do not treat this possibility at the moment or come back to this later.)

Here is an example. Suppose six seats are to be assigned and that the results of the election are: A received 8 votes, B received 5, and C received 9 votes. We get the following table of numbers after dividing by 1, 2, 3 ...:

$$\begin{array}{l} A: 8 \quad 4 \quad \frac{8}{3} \quad \frac{8}{4} \quad \frac{8}{5} \quad \frac{4}{3} \quad \dots \\ B: 5 \quad \frac{5}{2} \quad \frac{5}{3} \quad \frac{5}{4} \quad \frac{5}{5} \quad \frac{5}{6} \quad \dots \\ C: 9 \quad \frac{9}{2} \quad \frac{9}{3} \quad \frac{9}{4} \quad \frac{9}{5} \quad \frac{3}{2} \quad \dots \end{array}$$

Then, the decreasing list (with corresponding parties) is

$$\begin{array}{cccccc|c} 9 & 8 & 5 & \frac{9}{2} & 4 & 3 & \frac{8}{3} \\ C & A & B & C & A & C & A \end{array}$$

Thus, A gets two seats, C gets three, and B gets one seat. For more on the interesting mathematical aspects of elections, see [174★], [112★], [185★], [173★], [110★], [133★], [190★], [175★], [180★], [149★], [135★], [153★], and [184★]; for an interesting nonpolitical application, see [186★]. For a nice *Mathematica*-based proof of the related Arrow's theorem, see [182★].

12.^{L2} Grouping, Unsorted Complements

- Suppose we want to divide a given a list of real (complex) numbers into groups of numbers that are "close together". Program a corresponding function.
- Given a list of real positive integers $\{z_1, z_2, \dots, z_n\}$ and a positive number ϵ , extract all possible maximal length chains of numbers $\{z_{i_1}, z_{i_2}, \dots, z_{i_j}\}$, $j \geq 2$, such that $|z_{i_{k+1}} - z_{i_k}| \leq \epsilon$. Do not make use of the built-in function `Split`.
- Given a list of lists with vector-valued elements. (The vectors are lists (of equal length) of real numbers.) Assume some vectors occur possibly more than once, but the components of the vectors are slightly different (`Equal` would return `True`, but the last digits might be different). Write an efficient `VectorUnion` function that unions the list of vectors. Why is it possible to implement a function more efficient than the built-in `Union`?
- Write a function `UnSortedComplement[l1, l2]`, that forms the complement of l_1 with respect to the list l_2 and does not reorder the list l_1 and takes the multiplicity in the list l_2 into account when removing elements from l_1 .

13.^{L1} All Arithmetic Expressions

Given a list of numbers (atoms) and a list of binary operations, use the numbers and the operations to form all syntactically correct nested expressions. The order of the numbers should not be changed, and only the binary operations and parentheses () should be inserted between the numbers [126★].

14.^{L1} Symbols with Values, SetDelayed Assignments, Counting Integers

a) Identify which values will be collected in the following list `li`:

```
name = DeleteCases[Names["*"], "names"];
```

```
li = {};
```

```
Do[Clear[f];
```

```
  f[Evaluate[ToExpression[names[[i]] <> "_"]] =
                                     ToExpression[names[[i]]]^2;
  If[f[3] != 9, AppendTo[li, {names[[i]], ToExpression[names[[i]]}],
    {i, 1, Length[names]}];
```

```
li
```

b) Identify which built-in functions will be returned from the following code:

```
Cases[{#, ToExpression[StringJoin["f[x_] := " <> # <> "[x]"]];
       StringPosition[ToString[FullForm[DownValues[f]]], #]} & /@
      Names["System`*"], {_, {}}]
```

c) Given the following list of numbers

```
Do[data[n] = Table[IntegerPart[k Sin[k]], {k, 10^n}], {n, 4}].
```

Use various implementations to count how often each integer appears in `data[n]`.

15.^{L1} Sort[list, strangeFunction]

Examine whether `Sort` generates error messages for nontransitive, asymmetric order relations.

16.^{L3} Bracket-Aligned Formatting, Fortran Real*8, Method Option, Level functions, Conversion to StandardForm inputs

a) Write a function that formats a *Mathematica* expression in such a way that pairs of corresponding brackets [and] of the `FullForm` are aligned.

b) *Mathematica* includes the command `FortranForm`. Unfortunately, no type declarations are allowed, so the output is not always in an appropriate form to be given directly to a Fortran program. Write a function that rewrites arbitrary integers in the form of Fortran `Real*8` numbers. Let the result be a string.

c) Which *Mathematica* functions have a `Method` option? Can one use the *Mathematica* kernel to find the possible option settings?

d) Which *Mathematica* functions take level specifications? Can one use the *Mathematica* kernel to find these functions?

e) Write a function that converts the `InputForm` input cells of the *GuideBooks* into `StandardForm` cells. Preserve all comments, indentation (as much as possible) and use typical `StandardForm` symbols such as π , i , e , and \rightarrow .

17.^{L2} ReplaceAll Order, Pattern Realization, Pure Functions

a) The function `orderedTriedExpressions` returns a list of all (sub)expressions of `expr` in the order tried by `ReplaceAll`.

```
orderedTriedExpressions[expr_] :=
Module[{bag = {}}, expr /. x_ :=> Null /; (AppendTo[bag, x]; False); bag]
```

Implement a version of `orderedTriedExpressions` that uses only built-in functions. Implement another version of `orderedTriedExpressions` that does not make any assignments (no `Set` or `SetDelayed`).

b) Write a function `patternRealization` that, analogous to `MatchQ`, takes two arguments `expression` and `expressionWithPatterns` and gives a list of the actual realizations of the pattern variables. Write a version of `patternRealization` that does not contain any auxiliary variables. Test both versions on a few examples.

c) Given an expression that contains one-argument pure functions using `Slots` (like `#^2&[(#1 + #2)^3&[#1, 2#1]&[(#1 + #2 + (#^2&[#]))&[#1, #4]]&]`), write a function that replaces these pure functions with ones that have two arguments, and use explicit variables (like `Function[x, bodyContainingx]`).

18.^{L3} Matrix Identities, Frobenius Formula, Iterative Matrix Square Root

a) For an arbitrary 3×3 matrix \mathbf{A} ,

$$\mathbf{A}^3 - \text{tr}(\mathbf{A})\mathbf{A}^2 + \frac{1}{2}(\text{tr}(\mathbf{A})^2 - \text{tr}(\mathbf{A}^2))\mathbf{A} - \det(\mathbf{A})\mathbf{1} = 0,$$

where tr is the trace, \det is the determinant, and $\mathbf{1}$ is the 3D identity matrix. (This identity follows from the Theorem of Cayley–Hamilton together with the Newton relations.) Prove this identity.

b) For arbitrary 2×2 matrices \mathbf{A} and \mathbf{B} the following identity hold [109★]:

$$\mathbf{B}\mathbf{A} = (\text{tr}(\mathbf{A}\mathbf{B}) - \text{tr}(\mathbf{A})\text{tr}(\mathbf{B}))\mathbf{1} + \text{tr}(\mathbf{A})\mathbf{B} + \text{tr}(\mathbf{B})\mathbf{A} - \mathbf{A}\mathbf{B}$$

Here again tr stands for the trace, and $\mathbf{1}$ is the 2D identity matrix. Prove this identity. Does it also hold for 3×3 matrices? If not, does there exist a generalization of the form

$$\mathbf{B}\mathbf{A} = \left(\sum_{i,j,k,l=0}^1 c_{i,j,k,l}^{(\mathbf{1})} \text{tr}(\mathbf{A}^i \mathbf{B}^j) \text{tr}(\mathbf{A})^k \text{tr}(\mathbf{B})^l \mathbf{1} \right) + \left(\sum_{\alpha=1}^2 \sum_{i,j,k,l=0}^1 c_{i,j,k,l}^{(\mathbf{A},\alpha)} \text{tr}(\mathbf{A}^i \mathbf{B}^j) \text{tr}(\mathbf{A})^k \text{tr}(\mathbf{B})^l \mathbf{A}^\alpha \right) + \left(\sum_{\alpha=1}^2 \sum_{i,j,k,l=0}^1 c_{i,j,k,l}^{(\mathbf{B},\alpha)} \text{tr}(\mathbf{A}^i \mathbf{B}^j) \text{tr}(\mathbf{A})^k \text{tr}(\mathbf{B})^l \mathbf{B}^\alpha \right) - \mathbf{A}\mathbf{B}$$

for 3×3 matrices?

c) Prove the Amitsur–Levitzky identity [113★] for $n = 3$. The Amitsur–Levitzky identity states that for the $2n$ matrices of dimension $n \times n$, denoted by $\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_{2n}$, the following sum over all permutations σ of the numbers $(1, 2, \dots, 2n)$ yields the $n \times n$ zero matrix $\mathbf{0}_n$: $\sum_{\sigma} \text{signature}(\sigma) \mathbf{A}_{\sigma(1)} \mathbf{A}_{\sigma(2)} \cdots \mathbf{A}_{\sigma(2n)} = \mathbf{0}_n$.

d) Fix a univariate polynomial $q(x)$ of degree n and consider the eigenvalue problem [158★]

$$\frac{\partial^k (q(x) \psi_j^{(n,k)}(x))}{\partial x^k} = \lambda_j^{(n,k)} \psi_j^{(n,k)}(x).$$

Assume that the $\psi_j^{(n,k)}(x)$ are polynomials too and conjecture a closed form for the eigenvalues $\lambda_j^{(n,k)}$.

e) The well-known Frobenius formula [134★] expresses the inverse of a 2×2 block matrix $\begin{pmatrix} \mathbb{A} & \mathbb{B} \\ \mathbb{C} & \mathbb{D} \end{pmatrix}$ (\mathbb{A} , \mathbb{B} , \mathbb{C} , and \mathbb{D} being $n \times n$ matrices) in the form

$$\begin{pmatrix} \mathbb{A}^{-1} - \mathbb{A}^{-1} \cdot \mathbb{B} \cdot (\mathbb{B} - \mathbb{A} \cdot \mathbb{C}^{-1} \cdot \mathbb{D})^{-1} & -\mathbb{A}^{-1} \cdot \mathbb{B} \cdot (\mathbb{D} - \mathbb{C} \cdot \mathbb{A}^{-1} \cdot \mathbb{B})^{-1} \\ (\mathbb{B} - \mathbb{A} \cdot \mathbb{C}^{-1} \cdot \mathbb{D})^{-1} & (\mathbb{D} - \mathbb{C} \cdot \mathbb{A}^{-1} \cdot \mathbb{B})^{-1} \end{pmatrix}.$$

(The last expression can be rewritten in various equivalent forms.) Here we assume that the inverses of all four block matrices \mathbb{A} , \mathbb{B} , \mathbb{C} , and \mathbb{D} exist. Implement a function that derives this type of representation for an $n \times n$ block matrix. Test the function for $n = 2, 3$, and 4 .

f) Let \mathbf{A} be an $n \times n$ matrix. Its square root can be calculated by iterating the map $\mathbf{B} \rightarrow \mathbf{B} \cdot (\mathbf{B} \cdot \mathbf{B} + 3 \mathbf{A}) (3 \mathbf{B} \cdot \mathbf{B} + \mathbf{A})^{-1}$ starting with an $n \mathbf{D}$ identity matrix [154★]. Use this iteration to calculate a numerical approximation to the square root of a 10×10 Hilbert matrix.

g) Consider the following three $n \times n$ matrices $\mathbf{G}(a, b)$, $\mathbf{W}(x)$, and $\mathbf{M}(x_1, \dots, x_n)$ with elements

$$g_{i,j}(a, b) = \int_a^b f_i(x) f_j(x) dx$$

$$w_{i,j}(x) = \frac{\partial^{i-1} f_j(x)}{\partial x_j^{i-1}}$$

$$m_{i,j}(x_1, \dots, x_n) = f_j(x_j).$$

Here the f_j are unspecified real-valued functions. Verify the following relations for small n by explicit calculation [122★]:

$$\frac{\partial^{n^2} \det(\mathbf{G}(a, b))}{\partial b^{n^2}} \Big|_{b=a} = \frac{\prod_{k=1}^{n-1} k^{n-n-k}}{n^2!} \det(\mathbf{W}(a))$$

$$\det(\mathbf{G}(a, b)) = \int_a^b \dots \int_a^b \det(\mathbf{M}(x_1, \dots, x_n))^2 dx_1 \dots dx_n$$

$$\det(\mathbf{W}(x)) = \frac{\partial^{n(n-1)/2} \det(\mathbf{M}(x_1, \dots, x_n))}{\partial x_2 \partial x_3^2 \dots \partial x_n^{n-1}}$$

where $1 \leq i, j \leq n$. How far can one go with n ?

19.^{L2} Autoloading and Package Test

a) Many *Mathematica* functions are programmed in the *Mathematica* language and autoloaded when needed. Find these functions.

b) Analyze all packages from the standard packages directory according to the following criteria:

- How many commands are exported?
- Are undocumented commands exported?
- How many variables are used inside the packages?
- Which packages export no commands?
- Which packages change the attributes of built-in commands?

- Which packages alter the options for the built-in commands?
- Which packages give error messages when loaded?

Do not load each individual package “manually”.

20.^{L2} PrecedenceForm

Examine the meaning of the built-in command `PrecedenceForm`, and determine the precedence of all built-in commands (when possible). Knowing preferences is important, for instance, for determining if `2 + 4 // #; &` means `2 + 4 // (#;)&` or `(2 + 4 // #); &` and so on. Do the same with all named characters (like `CircleTimes`, `DoubleLong`, `RightArrow`) that can be used as operators.

21.^{L2} One-Liners

a) Write a “one-liner” that makes the following: Given a positive integer sum s and a list *summands* of positive integers a_i , the function `AllPossibleFactors[s, summands]` should return a list of all possibilities of lists of factors $\{f_1, f_2, \dots, f_n\}$, such as

$$s \leq \sum_{i=0}^n f_i a_i, \text{ with } f_i \geq 0 \quad \forall i$$

(A one-liner is, “by definition”, a *Mathematica* program that consists only of one piece of code, uses no named or temporary variables or functions, and is often a nice example in functional programming. A one-liner does not necessarily fit into one line.) How many different arrangements of 1¢, 5¢, 10¢, and 25¢ coins can one make, so that the net total is less than \$1?

b) Write a one-liner for the Ferrer conjugate from Exercise 9.d) in Chapter 5.

c) Model the function `AppendTo` by using the function `Append`.

d) Write a one-liner that recursively implements Meissel’s formula [191★], [162★], [117★], [159★], [160★] for the calculation of $\pi(n)$, the number of primes less than or equal to n .

$$\pi(n) = n - 1 + \pi(\sqrt{n}) + \sum_{j=1}^k (-1)^j \sum_{p_{i_1} < p_{i_2} < \dots < p_{i_j}} \left\lfloor \frac{n}{p_{i_1} \dots p_{i_j}} \right\rfloor$$

Here p_1, p_2, \dots, p_k are all primes less than or equal to \sqrt{n} . (The n th prime can be obtained using `Prime[n]`.) Use only built-in symbols.

e) Write a one-liner for generating the following polynomials $p_n(x_1, \dots, x_n)$ [155★] in factored form:

$$p_n(x_1, \dots, x_n) = \sum_{\sigma} \prod_{k=1}^n x_k^{\mu_k(\sigma)}$$

The summation runs over all elements of the permutations σ of $\{1, 2, \dots, n\}$. For a permutation $\sigma = \{j_1, j_2, \dots, j_n\}$ the function $\mu_k(\sigma)$ counts the number of j_l , $l = k + 1, \dots, n$ that are smaller than j_k . Calculate p_1 to p_8 explicitly.

f) Write a one-liner that, for given positive integers k and p ($0 < p < k$) proves the following identity [114★]:

$$\sum_{n_1=1}^{k-1} \sum_{n_2=1}^{k-n_1-1} \sum_{n_3=1}^{k-n_1-n_2-1} \cdots \sum_{n_p=1}^{k-n_1-n_2-\dots-n_{p-1}} \frac{1}{n_1!} \frac{\partial f(x)^{n_1}}{\partial x^{n_1}} \frac{1}{n_2!} \frac{\partial f(x)^{n_2}}{\partial x^{n_2}} \cdots \frac{1}{n_p!} \frac{\partial f(x)^{n_p}}{\partial x^{n_p}}$$

$$\frac{1}{(k-n_1-n_2-\dots-n_p)!} \frac{\partial f(x)^{k-n_1-n_2-\dots-n_p}}{\partial x^{k-n_1-n_2-\dots-n_p-1}} = (p+1) \frac{(k-1)!}{(k-1-p)!} \frac{1}{k!} \frac{\partial^{k-p-1} f(x)^k}{\partial x^{k-p-1}}.$$

g) For any $n \times n$ matrix \mathbb{A} , the following identity holds [107★]:

$$\det(\mathbb{A}) = \frac{1}{n!} \det \begin{pmatrix} \text{tr}(\mathbb{A}) & 1 & 0 & \cdots & 0 & 0 \\ \text{tr}(\mathbb{A}^2) & \text{tr}(\mathbb{A}) & 2 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \text{tr}(\mathbb{A}^{n-1}) & \text{tr}(\mathbb{A}^{n-2}) & \text{tr}(\mathbb{A}^{n-2}) & \cdots & \text{tr}(\mathbb{A}) & n-1 \\ \text{tr}(\mathbb{A}^n) & \text{tr}(\mathbb{A}^{n-1}) & \text{tr}(\mathbb{A}^{n-2}) & \cdots & \text{tr}(\mathbb{A}^2) & \text{tr}(\mathbb{A}) \end{pmatrix}.$$

Implement a one-liner that checks this identity for a given matrix \mathbb{A} . ($n!$ is the factorial function, in *Mathematica* $n!$.)

h) Let $p_{\mathbb{A}}(z) = \sum_{j=0}^n c_j z^j$ be the characteristic polynomial of the $n \times n$ matrix \mathbb{A} . Then, the inverse \mathbb{A}^{-1} can be expressed as $\mathbb{A}^{-1} = -c_0^{-1} \sum_{j=1}^n c_j \mathbb{A}^{j-1}$ [107★]. Implement a one-liner that uses this identity to calculate the inverse.

i) Write a one-liner that implements the expansion of an arbitrary function $f(z)$ in the product [108★]

$$\Pi_o(f(z), z_0) = \prod_{k=0}^o (\mathcal{E}_k(f(z_0)))^{\frac{\ln(z/z_0)}{-k!}}$$

around the point $f(z_0) \neq 0$ and where $\mathcal{E}_0(f(z)) = f(z)$ and $\mathcal{E}_k(f(z)) = \exp(\zeta \partial \ln(f(z)) / \partial \zeta)$. For a sufficiently smooth function $f(z)$, we have $\lim_{o \rightarrow \infty} \Pi_o(f(z), z_0) = f(z)$. Calculate $\Pi_{12}(\cos(\pi/2), 1)$.

j) Write a one-liner that generates all different expressions resulting from the repeated application of a binary (two-argument) function to n sorted arguments. For example, for the four arguments a, b, c, d and the function f , the five compositions $f(a, f(b, f(c, d)))$, $f(a, f(f(b, c), d))$, $f(f(a, b), f(c, d))$, $f(f(a, f(b, c)), d)$, and $f(f(f(a, b), c), d)$ should be formed. How frequently do k consecutive closing ‘)’ occur for ten arguments? For six equal arguments $a = b = \dots = \sqrt{-1}$, and $f = \text{Power}$, how many numerically different expressions result [136★], [137★]?

k) Write a one-liner `KolakoskiSequence[n]` that calculates the first n terms of the Kolakoski sequence [152★], [124★], [178★]. With the exception of n , the function `KolakoskiSequence` should not use any not built-in commands. The Kolakoski sequence $\{2, 2, 1, 1, 2, 1, 2, 2, 1, 2, 2, 1, 1, \dots\}$ is the (unique) sequence of its own run lengths (meaning 2 twos, then 2 ones, then 1 two, 1 one, then 2 twos, ...).

l) Given the three differential identities

$$\begin{aligned} x'(t) &= y(t) z(t) \\ y'(t) &= x(t) z(t) \\ z'(t) &= x(t) y(t) \end{aligned}$$

define a sequence of functions recursively through $\sigma_k(t) = \partial \sigma_{k-1}(t) / \partial t$, starting with $\sigma_0(t) = x(t)$. The resulting $\sigma_n(t)$ have the form $\sigma_n(t) = \sum_{i,j,k=0}^{n+1} c_{i,j,k} x(t)^i y(t)^j z(t)^k$. The coefficients fulfill the following sum rule: $\sum_{i,j,k=0}^{n+1} c_{i,j,k} = n!$ [128★]. Implement a one-liner `factorialSumTest` that, by explicit calculation, checks this property for a given n (the factorial of n is just $n!$ in *Mathematica*). The implementation should not use any built-in symbol. Check the sum property for $0 \leq n \leq 100$.

m) Write a one-liner that, for a given n , calculates the number of permutations having k ($k = 0, 1, \dots, n$) increasing two-sequences in all permutations of $\{1, 2, \dots, n\}$. (An increasing two sequence in a permutation $\{j_1, j_2, \dots, j_n\}$ is a pair $\{j_i, j_{i+1}\}$ such that $j_{i+1} = j_i + 1$ [144★], [145★].)

22.^{L2} Precedences

a) What are the results of the following expressions?

```
Function[x, Hold[x], {Listable}] @
  Hold[{1 + 1, 2 + 2, 3 + 3}]
```

```
Function[x, Hold[x], {Listable}] @@
  Hold[{1 + 1, 2 + 2, 3 + 3}]
```

```
Function[x, Hold[x], {Listable, HoldAll}] @
  Hold[{1 + 1, 2 + 2, 3 + 3}]
```

```
Function[x, Hold[x], {Listable, HoldAll}] @@
  Hold[{1 + 1, 2 + 2, 3 + 3}]
```

```
Function[x, Hold[x], {Listable, HoldAll}] @
  (#& @@ Hold[{1 + 1, 2 + 2, 3 + 3}])
```

```
Function[x, Hold[x], {Listable}] @@
  (#& @@ Hold[{1 + 1, 2 + 2, 3 + 3}])
```

```
Function[x, Hold[x], {Listable, HoldAll}] @ #& @@
  Hold[{1 + 1, 2 + 2, 3 + 3}]
```

```
Function[x, Hold[x], {Listable, HoldAll}] @
  Function[x, Hold[x], {Listable, HoldAll}] @@
  Hold[{1 + 1, 2 + 2, 3 + 3}]
```

```
Function[x, Hold[x], {Listable, HoldAll}] @@
  Function[x, Hold[x], {Listable, HoldAll}] @
  Hold[{1 + 1, 2 + 2, 3 + 3}]
```

```
Function[x, Hold[x], {Listable, HoldAll}] @@
  Function[x, Hold[x], {Listable, HoldAll}] @@
  Hold[{1 + 1, 2 + 2, 3 + 3}]
```

b) If

```
localVar = 11;
```

```
Block[{localVar = 1}, Print[localVar]; WhatIsHere]
```

prints out 11, what might have been coded in *WhatIsHere*? Find a *WhatIsHere* that also works if `Block` is replaced by `With`.

23.^{L2} Puzzles

a) What is the result of the following input? (Here the spaces in the input matter; do not introduce or remove blanks.)

```
1 @ 2 @@ 3 / 4 /@ 6 //@ 7 || 8 | 9 /.10 /.11
```

b) Find a value for `factor`, such that the following two definitions for give different results.

```
scaledReversedShiftedListV1[factor_, list_List] :=
  Function[Join[factor #, Reverse[factor/2 #]]][list]
```

```
scaledReversedShiftedListV2[factor_, list_List] :=
  Function[x, Join[factor x, Reverse[factor/2 x]]][list]
```

c) Predict the result of the following input.

```
{#, InputForm[ToExpression @ #],
  FullForm[ToExpression @ #]}& /@
  Table["1"<>Table[".", {i}], {i, 1, 11}] // TableForm
```

d) Predict the result of the following input.

```
Power @@ Unevaluated[Times[2, 2, 2]].
```

e) Predict the result of the following input.

```
Power[Delete @@ Cos[Sin[2], 0]].
```

f) Predict the result of the following input.

```
{Dimensions[#], Length[Flatten[#]]}& /@
  NestList[Outer[List, #, #]&, {1., 2}, 3]
```

g) Given a held expression, write a function that replaces all occurrences of `p_Plus` by the evaluated result of `Length[p]`.

h) Predict the result of the following input.

```
Block[{Infinity}, Apply[Subtract, {Infinity, Infinity}]]
```

i) Predict the result of the following input.

```
inherit[fNew_, fOld_] :=
CompoundExpression[
  SetAttributes[fNew, Attributes[fOld]];
  Options[fNew] = Options[fOld];
  (#[fNew] = (#[fOld] /. fOld -> fNew))& /@
  {NValues, SubValues, DownValues,
   OwnValues, UpValues, FormatValues}]
```

```
SetAttributes[f, {Listable}];
f[x_Plus] := Length[Unevaluated[x]];
```

```
Module[{f},
  inherit[f, ToExpression["f"]];
  SetAttributes[f, HoldAll];
```

```
f[i_Integer] = i^2;
f @@ f[{1 + 1, 2 + 2}]
```

j) Predict which messages will be issued when evaluating the following:

```
Evaluate //@ Block[{I = 1}, I^2]
```

What will be the result?

k) Find a *Mathematica* expression *expr* such that `First[expr]` and `expr[[1]]` give different results.

l) Predict the result of the following inputs:

```
f[x_] := Block[{α = Not[TrueQ[α]]}, f[x + 1] /; α]
f @@ f[0]
```

m) Predict the result of the following input:

```
Module[{x = D, f}, C @@ f[x_] ~ Set ~ x // f[C]&] -
Module[{x = D, f}, Set @@ f[x_] ~ C ~ x // f[C]&]
```

n) Predict the result of the following input:

```
c = 0;
Union[Array[1&, {100}], SameTest -> ((c = c + 1; False)&)];
c
```

o) Implement a function `virtualMatrix[dim]` that generates a “virtual” matrix of size $dim \times dim$ that behaves like a “real” matrix as in the following:

```
In[2]:= M = virtualMatrix[10^6];

In[3]:= {MatrixQ[M], Dimensions[M], Length[M[[1]]],
        {M[[1, 1]], M[[-1, -1]]},
        M[[1000, 1000]] = 1000; M[[1000, 1000]]}

Out[3]= {True, {1000000, 1000000}, 1000000, {1, 1}, 1000}
```

Do not unprotect any built-in function or use upvalues.

p) Given an expression *expr* (fully evaluated and not containing any held parts) and two integers *k* and *l*, what is the result of `MapIndexed[(Part[expr, ##]& @@ #2)&, expr, {k, l}, Heads -> True]`?

24.^{L2} Hash Value Collisions, Permutation Digit Sets

a) The function `Hash[expr]` returns the hash value of *expr*. Find two integers that are hashed to the same hash value.

b) Let $\mathcal{S}_o^{(b)}$ be the set of all *o*-digit integers in base *b* with every digit from the range $[1, o]$ appearing exactly once in the base *b* representation. (For instance $\mathcal{S}_3^{(10)} = \{123, 132, 213, 231, 312, 321\}$.) Let $\mathcal{M}_o^{(b)}$ be the set of pairs $\{s_1, s_2\}$, $s_1, s_2 \in \mathcal{S}_o^{(b)}$ such that $s_2 = m s_1$, $j \in \mathbb{N}$, $m \geq 2$. Find the sets $\mathcal{M}_o^{(b)}$ for $2 \leq b \leq 10$, $1 \leq o \leq b - 1$. How many elements does $\mathcal{M}_{11}^{(12)}$ have? Format the results in the form $s_2 = m s_1$.

25.^{L1} Function Calls in GluedPolygons

In the construction of the glued polygons in Section 6.0, the function `Trace` was used to show the relative frequency of the use of various list manipulating functions. This was overestimating the number of function calls. Determine the actual number of calls to the functions `Reverse`, `Join`, `Dot`, `Map`, `Partition`, `Apply`, `Take`, `MapThread`, `Drop`, `Table`, `Part`, and `Flatten` when evaluating `GluedPolygons[5, 3Pi/4, 1, Polygon, DisplayFunction -> Identity]`.

References Chapter 1

- ★1 E. Baylis. *Theoretical Methods in the Physical Sciences*, Birkhäuser, Boston, 1994.
- ★2 N. Blachman, M. Mossinghoff. *Maple V Quick Reference*, Brooks/Cole, New York, 1994.
- ★3 J. Borgert, H. Schwarze. *Maple in der Physik*, Addison-Wesley, Bonn, 1995.
- ★4 B. W. Char, K. O. Geddes, G. H. Gonnet, B. L. Leong, M. B. Monagan, S. M. Watt. *Maple V Language Reference Manual*, Springer-Verlag, New York, 1991.
- ★5 B. W. Char, K. O. Geddes, G. H. Gonnet, B. L. Leong, M. B. Monagan, S. M. Watt. *Maple V Library Reference Manual*, Springer-Verlag, New York, 1991.
- ★6 B. W. Char, K. O. Geddes, G. H. Gonnet, B. L. Leong, M. B. Monagan, S. M. Watt. *First Leaves, A Tutorial Introduction to Maple*, Springer-Verlag, New York, 1991.
- ★7 *Computeralgebra in Deutschland Bestandsaufnahme, Möglichkeiten, Perspektiven*, published by Fachgruppe of the GI, DMV, GAMM, Passau and Heidelberg, <http://www.uni-karlsruhe.de/~CAIS/mitteilungen/ca-report-info.ps>, 1993.
- ★8 R. M. Corless. *Essential Maple*, Springer-Verlag, New York, 1994.
- ★9 R. M. Corless. *Symbolic Recipes*, Springer-Verlag, New York, 1994.
- ★10 J. S. Devitt. *Calculus with Maple V*, Brooks/Cole, Pacific Grove, 1993.
- ★11 R.H. Enns, G. McGuire. *Nonlinear Physics with Maple for Scientists*, Birkhäuser, Basel, 1997.
- ★12 R. Enns, G. McGuire. *Computer Algebra Recipes*, Birkhäuser, Boston, 2002.
- ★13 B. Fuchssteiner, W. Wiwianka, K. Gottheil, A. Kemper, O. Kluge, K. Morisse, H. Naundorf, G. Oevel, T. Schulze. *MuPAD Multi-Processing Algebra Data Tool Benutzerhandbuch*, Birkhäuser, Basel, 1993.
- ★14 W. Gander, J. Hrebicek. *Solving Problems in Scientific Computing Using Maple and Matlab*, Springer-Verlag, Berlin, 1993.
- ★15 J. Gerhard, W. Oevel, F. Postel, S. Wehmeier. *MuPAD Tutorial*, Springer-Verlag, Berlin, 2000.
- ★16 H. Gloggenieser. *Maple V*, Markt und Technik, Haar, 1993.
- ★17 J. Grabmeier, E. Kaltofen, V. Weisspfenning (eds.). *Computer Algebra Handbook*, Springer-Verlag, Berlin, 2002.
- ★18 R. L. Greene. *Classical Mechanics with Maple*, Springer-Verlag, New York, 1995.
- ★19 A. Heck. *Introduction to Maple*, Springer-Verlag, New York, 1993.

- ★20 F. W. Hehl, V. Winkelmann, H. Meyer. *REDUCE: Ein Kompaktkurs über die Anwendung von Computer-Algebra*, Springer-Verlag, Berlin, 1993.
- ★21 M. H. Holmes, J. G. Ecker, W. Boyce, W. Siegmann. *Exploring Calculus with Maple*, Addison-Wesley, Reading, 1993.
- ★22 M. Horbatsch. *Quantum Mechanics Using Maple*, Springer-Verlag, New York, 1995.
- ★23 R. D. Jenks, R. S. Sutor. *AXIOM: The Scientific Computation System*, Springer-Verlag, New York, 1992.
- ★24 E. Johnson. *Linear Algebra Using Maple*, Brooks/Cole, Pacific Grove, 1993.
- ★25 E. Kamerich. *A Guide to Maple*, Springer-Verlag, New York, 1994.
- ★26 M. Kofler. *Maple V, Release 2 Einführung und Leitfaden für den Praktiker*, Addison-Wesley, Bonn, 1993.
- ★27 L. Lambe. *Notices Am. Math. Soc.* 41, 14 (1994).
- ★28 T. Lee (ed.). *Mathematical Computations with Maple V*, Birkhäuser, Basel, 1993.
- ★29 R. J. Lopez (ed.). *Maple V: Mathematics and Its Applications*, Birkhäuser, Boston, 1994.
- ★30 R. J. Lopez. *Maple via Calculus*, Birkhäuser, Basel, 1994.
- ★31 N. MacDonald. *REDUCE for Physicists*, World Scientific, Singapore, 1994.
- ★32 M. Majewski. *MuPAD Pro Computing Essentials*, Springer-Verlag, Berlin, 2002.
- ★33 *mathPAD 3*, n1 (1993).
- ★34 R. A. Nicolaides, N. J. Walkington. *Maple—A Comprehensive Introduction*, Cambridge University Press, Cambridge, 1996.
- ★35 G. Rayna. *Reduce: Software for Algebraic Computation*, Springer-Verlag, Berlin, 1987.
- ★36 D. Redfern. *The Maple Handbook*, Springer-Verlag, New York, 1993.
- ★37 D. Richards. *Advanced Mathematical Methods with Maple*, Cambridge University Press, Cambridge, 2002.
- ★38 S. Sorgatz, S. Wehmeier. *Math. Comput. Simul.* 49, 235 (1999).
- ★39 Special Issue on OpenMath. *SIGSAM Bull.* 34 (2000).
- ★40 W.-H. Steeb, D. Lewien. *Algorithms and Computation with REDUCE*, BI-Verlag, Mannheim, 1992.
- ★41 W. Steeb. *Quantum Mechanics Using Computer Algebra*, World Scientific, Singapore, 1994.
- ★42 J. Ueberberg. *Einführung in die Computeralgebra mit REDUCE*, BI-Verlag, Mannheim, 1992.
- ★43 J. A. M. Vermaseren. *arXiv:math-ph/0010025* (2000). *Get Preprint*
- ★44 J. A. M. Vermaseren. *arXiv:hep-ph/0211297* (2002). *Get Preprint*

References Chapter 3

- ★45 G. Dattoli, H. M. Srivastava, C. Cesarano. *Appl. Math. Comput.* 124, 117 (2001).
- ★46 A. Wünsche. *J. Comput. Appl. Math.* 133, 665 (2001).

References Chapter 5

- ★47 M. S. Alber, C. Miller. *arXiv:nlin.PS/0001004* (2000). *Get Preprint*
- ★48 G. E. Andrews. *SIAM Rev.* 16, 441 (1974).
- ★49 G. E. Andrews. *The Theory of Partitions*, Addison-Wesley, Reading, 1976.
- ★50 R. Askey. *CRM Proc. Lecture Notes* 9, 13 (1996).
- ★51 K. J. Baumeister in S. Sengupta, J. Häuser, P. R. Eiseman, J. F. Thompson (eds.). *Numerical Grid Generation in Computational Fluid Mechanics*, Pineridge Press, Swansea, 1988.
- ★52 H. B. Benaoum. *J. Phys. A* 31, L 751 (1998).
- ★53 H. B. Benaoum. *arXiv:math-ph/9812028* (1998). *Get Preprint*
- ★54 H. B. Benaoum. *J. Phys. A* 32, 2037 (1999).
- ★55 D. Bonatsos, C. Daskaloyannis. *arXiv:nucl-th/9999003* (1999). *Get Preprint*
- ★56 J. M. Borwein, D. M. Bradley, R. E. Crandall. *J. Comput. Appl. Math.* 121, 247 (2000).
- ★57 D. Bowman. *q-Difference Operators, Orthogonal Polynomials, and Symmetric Expansions*, American Mathematical Society, Providence, 2002.
- ★58 R. Camassa, D. D. Holm. *Phys. Rev. Lett.* 71, 1661 (1993).
- ★59 R. Camassa, D. L. Holm, J. M. Hyman. *Adv. Appl. Mech.* 31, 1 (1994).
- ★60 H. Cohn. *Am. Math. Monthly* 111, 487 (2004).
- ★61 L. Comtet. *Advanced Combinatorics*, Reidel, Dordrecht, 1974.
- ★62 B. Costa, W. S. Don. *Appl. Num. Math.* 33, 151 (2000).
- ★63 A. Degasperis, D. D. Holm, A. N. W. Hone. *arXiv:nlin.SI/0205023* (2002). *Get Preprint*
- ★64 A. Degasperis, D. D. Holm, A. N. W. Hone. *arXiv:nlin.SI/0209008* (2002). *Get Preprint*
- ★65 A. Dimakis, F. Müller–Hoissen. *Phys. Lett. B* 295, 242 (1992).
- ★66 L. Di Vizio. *arXiv:math.NT/0211217* (2002). *Get Preprint*
- ★67 V. K. Dobrev. *arXiv:quant-ph/0207077* (2002). *Get Preprint*
- ★68 H. Exton. *q-Hypergeometric Functions and Applications*, Ellis Horwood, Chichester, 1983.
- ★69 A. S. Fokas, P. J. Olver, P. Rosenau in A. S. Fokas and I. M. Gel'fand (eds.). *Progress in Nonlinear Differential Equations*, Birkhäuser, Boston, 1996.
- ★70 B. Fornberg in G. D. Byrne, W. E. Schiesser (eds.). *Recent Developments in Numerical Methods and Software for ODEs/DAEs/PDEs*, World Scientific, Singapore, 1992.
- ★71 B. Fornberg. *A Practical Guide to Pseudospectral Methods*, Cambridge University Press, Cambridge, 1996.
- ★72 B. Fornberg. *SIAM Rev.* 40, 685 (1998).
- ★73 I. P. Goulden, D. M. Jackson. *Combinatorial Enumeration*, Wiley, New York, 1983.

- ★74 R. Grimshaw, B. A. Malomed, G. A. Gottwald. *arXiv:nlin.PS/0203056* (2002). *Get Preprint*
- ★75 A. S. Hegazi, M. Mansour. *Int. J. Theor. Phys.* 41, 1815 (2002).
- ★76 D. D. Holm, M. F. Staley. *arXiv:nlin.CD/0203007* (2002). *Get Preprint*
- ★77 R. Jaganathan. *arXiv:math-ph/0003018* (2000). *Get Preprint*
- ★78 W. P. Johnson. *Discr. Math.* 157, 207 (1996).
- ★79 I. R. Khan, R. Ohba. *J. Comput. Appl. Math.* 107, 179 (1999).
- ★80 T. H. Koornwinder. *arXiv:math.CA/9403216* (1994). *Get Preprint*
- ★81 T. Koornwinder. *Informal Paper*(1999).
<http://www.wins.uva.nl/pub/mathematics/reports/Analysis/koornwinder/qbinomial.ps>
- ★82 B. A. Kupershmidt. *J. Nonlinear Math. Phys.* 7, 244 (2000).
- ★83 D. Larsson, S. D. Silvestrov. *J. Nonlin. Math. Phys.* 10, 95 (2003).
- ★84 Y. A. Li, P. J. Olver, P. Rosenau in M. Grosser, G. Hormann, M. Kunzinger, and M. Oberguggenberger (eds.). *Nonlinear Theory of Generalized Functions*, Chapman and Hall, New York, 1999.
- ★85 Z. Liu, T. Qian. *Int. J. Bifurc. Chaos.* 11, 781 (2001).
- ★86 Z. Liu, T. Qian. *Appl. Math. Model.* 26, 473 (2002).
- ★87 Z. Liu, R. Wang, Z. Jing. *Chaos, Solitons, Fractals* 19, 77 (2003).
- ★88 M. Lothaire. *Algebraic Combinatorics on Words*, Cambridge University Press, Cambridge, 2002.
- ★89 K. Maurin. *The Riemann Legacy*, Kluwer, Dordrecht, 1997.
- ★90 S. A. Messaoudi. *Int. J. Math. Edu. Sci. Technol.* 33, 425 (2002).
- ★91 J. Morales, A. Flores–Riveros. *J. Math. Phys.* 30, 393 (1989).
- ★92 F. Neuman. *Adv. Difference Eq.* 2, 111 (2004).
- ★93 L. D. Pustyl'nikov. *Izvest. Math.* 65, 85 (2001).
- ★94 A. V. Ryzhov, L. G. Yaffe. *arXiv:hep-ph/0006333* (2000). *Get Preprint*
- ★95 H. Scheffé. *Technometrics* 12, 388 (1970).
- ★96 A. Schilling, S. O. Warnaar. *Ramanujan J.* 2, 459 (1998).
- ★97 D. Scott. *Am. Math. Monthly* 92, 422 (1985).
- ★98 C. Shu. *Differential Quadrature and its Applications in Engineering Sciences*, Springer-Verlag, Berlin, 2000.
- ★99 R. P. Stanley. *Enumerative Combinatorics* v.1, Cambridge University Press, Cambridge, 1997.
- ★100 D. Stanton in E. Koelink, W. Van Assche (eds.). *Orthogonal Polynomials and Special Functions*, Springer-Verlag, Berlin, 2003.
- ★101 S. Steinberg, P. J. Roache in D. V. Shirkov, V. A. Rostovtsev, V. P. Gerdt (eds.). *IV. International Conference on Computer Algebra in Physical Research*, World Scientific, Singapore, 1991.
- ★102 B. Strand. *J. Comput. Phys.* 110, 47 (1994).
- ★103 H. Suyari. *arXiv:cond-mat/0401546* (2004). *Get Preprint*

- ★104 J. H. van Lint, R. M. Wilson. *A Course in Combinatorics*, Cambridge University Press, Cambridge, 1992.
- ★105 C. P. Viazminsky. *arXiv:math.NA/0210167* (2002). *Get Preprint*
- ★106 B. D. Welfert. *SIAM J. Numer. Anal.* 34, 1640 (1997).

References Chapter 6

- ★107 R. Aldrovandi. *Special Matrices of Mathematical Physics*, World Scientific, Singapore, 2001.
- ★108 A. Arache. *Am. Math. Monthly* 72, 861 (1965).
- ★109 Y. Avishai, D. Berend. *J. Phys. A* 26, 2437 (1993).
- ★110 J. Baylis. *Math. Gazette* 69, 95 (1985).
- ★111 F. Benford. *Proc. Am. Philos. Soc.* 78, 551 (1938).
- ★112 H. Bergold. *Didaktik Math.*, 4, 266 (1979).
- ★113 D. Birmajer. *arXiv:math.RA/0305430* (2003). *Get Preprint*
- ★114 L. Blanchet, G. Faye. *J. Math. Phys.* 42, 4391 (2001).
- ★115 A. L. Bondarev. *Teor. Mat. Fiz.* 101, 315 (1994).
- ★116 V. I. Borodulin, R. N. Rogalyov, S. R. Slabospitsky. *arXiv:hep-ph/9507456* (1995). *Get Preprint*
- ★117 A. Brauer. *Am. Math. Monthly* 53, 521 (1946).
- ★118 R. Brown, J. L. Merzel. *Period. Math. Hung.* 47, 45 (2003).
- ★119 B. Buck, A. C. Merchant, S. M. Perez. *Eur. J. Phys.* 14, 59 (1993).
- ★120 J. Burke, E. Kincaid. *Am. J. Phys.* 59, 952 (1991).
- ★121 D. I. A. Cohen. *J. Comb. Th. A* 20, 367 (1976).
- ★122 D. R. Curtiss. *Bull. Am. Math. Soc.* 17, 463 (1911).
- ★123 M. A. B. Deakin. *Austral. Math. Soc. Gazette* 20, 149 (1993).
- ★124 F. M. Dekking in R. V. Moody (ed.). *The Mathematics of Long-Range Aperiodic Order*, Kluwer, Dordrecht, 1997.
- ★125 P. Diaconis. *Ann. Prob.* 5, 72 (1977).
- ★126 L. E. Dickson. *History of the Theory of Numbers*, v. I, Chelsea, New York, 1952.
- ★127 S. N. Dorogovtsev, J. F. F. Mendes. *arXiv:cond-mat/0105093* (2001). *Get Preprint*
- ★128 D. Dumont. *Math. Comput.* 33, 1293 (1979).
- ★129 A. Edalat, P. J. Potts. *Electr. Notes Theor. Comput. Sc.* 6, (1997).
<http://www.elsevier.com/gej-ng/31/29/23/31/23/61/tcs6007.ps>
- ★130 S. B. Edgar, A. Höglund. *arXiv:gr-qc/0105066* (2001). *Get Preprint*
- ★131 P. Erdős, V. Lev, G. Rauzy, C. Sandor, A. Sárközy. *Discr. Math.* 200, 119 (1999).
- ★132 L. V. Furlan. *Das Harmoniegesetz der Statistik*, Verlag für Recht und Gesellschaft AG, Basel, 1946.

- ★133 S. Galam. *Physica A* 274, 132 (1999).
- ★134 F. R. Gantmacher. *The Theory of Matrices*, Chelsea, New York, 1959.
- ★135 S. Garfunkel, C. A. Steen. *Mathematik in der Praxis*, Spektrum der Wissenschaft-Verlag, Heidelberg, 1989.
- ★136 F. Gobel, R. P. Nederpelt. *Am. Math. Monthly* 78, 1097 (1971).
- ★137 R. K. Guy, J. F. Selfridge. *Am. Math. Monthly* 80, 868 (1973).
- ★138 R. Heckmann. *Theor. Comput. Sc.* 279, 65, (2002).
- ★139 E. R. Hedrick. *Ann. Math.* 1, 49 (1899).
- ★140 T. P. Hill. *Am. Math. Monthly* 102, 322 (1995).
- ★141 T. P. Hill. *Proc. Am. Math. Soc.* 123, 887 (1995).
- ★142 T. P. Hill. *Stat. Sc.* 10, 354 (1995).
- ★143 R. Honsberger. *Ingenuity in Mathematics*, Random House, New York, 1970.
- ★144 D. M. Jackson, R. Aleliunas. *Can. J. Math.* 29, 971 (1977).
- ★145 B. C. Johnson. *Methol. Comput. Appl. Prob.* 3, 35 (2001).
- ★146 B. K. Jones in D. Abbott, L. B. Kish (eds.). *Unsolved Problems of Noise and Fluctuations*, American Institute of Physics, Melville, 2000.
- ★147 J. H. Jordan. *Am. Math. Monthly* 71, 61 (1964).
- ★148 S. A. Kamal. *Matrix Tensors Quart.* 31, 64 (1981).
- ★149 J. S. Kelly. *Arrow Impossibility Theorems*, Academic Press, New York, 1978.
- ★150 J. B. Kim, J. E. Dowdy. *J. Korean Math. Soc.* 17, 141 (1980).
- ★151 D. E. Knuth. *The Art of Computer Programming*, v.2, Addison-Wesley, Reading, 1969.
- ★152 W. Kolakoski. *Am. Math. Monthly* 72, 674 (1965).
- ★153 K. Kopferman. *Mathematische Aspekte der Wahlverfahren*, BI, Mannheim, 1991.
- ★154 S. Lakić, M. S. Petković. *ZAMM* 78, 173 (1998).
- ★155 A. Lascoux. *Ann. Combinatorics* 1, 91 (1997).
- ★156 D. S. Lemons. *Am. J. Phys.* 54, 816 (1986).
- ★157 M. Lotan. *Am. Math. Monthly* 56, 535 (1948).
- ★158 G. Másson, B. Shapiro. *Experimental Math.* 10, 609 (2001).
- ★159 E. Meissel. *Math. Ann.* 2, 636 (1870).
- ★160 E. Meissel. *Math. Ann.* 3, 523 (1870).
- ★161 R. Miller. *Am. Math. Monthly* 85, 183 (1978).
- ★162 W. Narkiewicz. *The Development of Prime Number Theory*, Springer-Verlag, Berlin, 2000.
- ★163 M. J. Nigrini. *J. Am. Tax Ass.* 18, 72 (1996).
- ★164 R. Oldenburger. *Am. Math. Monthly* 47, 25 (1940).

- ★165 F. Palmer (ed.). *Selected Papers by J. R. Firth*, Longman, London, 1968.
- ★166 E. Pascal. *Die Determinanten*, Teubner, Leipzig, 1900.
- ★167 A. R. Penner. *Am. J. Phys.* 69, 332, (2001).
- ★168 L. Pietronero, E. Tosatti, V. Tosatti, A. Vespignani. *arXiv:cond-mat* 9808305 (1998). *Get Preprint*
- ★169 L. Pietronero, E. Tosatti, V. Tosatti, A. Vespignani. *Physica A* 293, 297 (2001).
- ★170 R. S. Pinkham. *Ann. Math. Stat.* 32, 1223 (1962).
- ★171 R. A. Raimi. *Am. Math. Monthly* 83, 521 (1976).
- ★172 P. Renauld. *New Zealand J. Math.* 31, 73 (2002).
- ★173 D. G. Saari. *The Geometry of Voting*, Springer-Verlag, New York, 1994.
- ★174 D. G. Saari. *Chaotic Elections! A Mathematicians Looks at Voting*, American Mathematical Society, Providence, 2001.
- ★175 D. G. Saari. *Math. Mag.* 70, 83 (1997).
- ★176 P. Schatte. *ZAMM* 53, 553 (1973).
- ★177 E. Schmutzer. *Relativistische Physik*, Geest and Portig, Leipzig, 1968.
- ★178 B. Sing. *arXiv:math-ph/0207037* (2002). *Get Preprint*
- ★179 E. Stade. *Rocky Mountain J. Math.* 29, 691 (1999).
- ★180 E. Stensholt. *SIAM Rev.* 38, 96 (1996).
- ★181 Z.-W. Sun. *Discr. Math.* 257, 143 (2002).
- ★182 S.-I. Takekuma. *Hitotsubashi J. Econom.* 38, 139 (1997).
- ★183 V. Tapia. *arXiv:gr-qc/0408007* (2004). *Get Preprint*
- ★184 A. D. Taylor. *Mathematics and Politics*, Springer-Verlag, New York, 1995.
- ★185 A. D. Taylor. *Am. Math. Monthly* 109, 321 (2002).
- ★186 C. Van den Broeck, J. M. R. Parrondo. *Phys. Rev. Lett.* 71, 2355 (1993).
- ★187 I. Vardi. *The Mathematica Journal* 1, n3, 53 (1991).
- ★188 S. Weinberg. *The Quantum Theory of Fields* v.1, Cambridge University Press, Cambridge, 1996.
- ★189 T. West. *Comput. Phys. Commun.* 77, 286 (1993).
- ★190 D. R. Woodall. *Math. Intell.* 8, n4, 36 (1986).
- ★191 S. Y. Yan. *Number Theory for Computing*, Springer-Verlag, Berlin, 2000.
- ★192 L. Zhipeng, C. Lin, W. Huajia. *arXiv:math.ST/0408057* (2004). *Get Preprint*