

Primzahlen – Altbekanntes und Neues

Internet: <https://yapps-arrgh.de>
Datei: [Primzahlen Ergaenzungen.pdf](#)

Ergänzungen zum Buch ab Version 46

Stand: Dezember 2020

Buch: ISBN 978-3-939247-92-0

- 5.7 Das Riemann-Spektrum
- 8.9 [Gleichungssysteme mit Integer-Sequenzen als Lösungen \(Neu\)](#)
- 11.2 Die Funktionen NotPrimeQ und SignedPrimeQ
- 18.4 Körper und erweiterte Rechenoperatoren (Ko-Autor: Adil Bulus) (Neu)
- 20.8.2 [Werterhaltende Übergänge von 2-er-Potenzen im OCRON-Raum](#)
- 20.8.3 [Virtuelle OCRONs und Matula-Goebel Zahlen](#)
- 20.10.5 Nullstellen der Ramanujanschen Tau-L-Funktion $L(s)$
- 20.10.6 'Repunit' Zahlen und 'repunit' Primzahlen (in Vorbereitung)

5.7_DAS RIEMANN-SPEKTRUM

Wir verwenden den Begriff „Riemann-Spektrum“ oder „Riemann-Transformation“ wenn wir die Transformation der Primzahlen (und ihrer Potenzen) in die Menge der Nullstellen der Zeta-Funktion meinen.

Auf der anderen Seite verwenden wir den Begriff "Inverses Riemann-Spektrum" oder "Inverse Riemann-Transformation", wenn wir die Transformation der Riemannschen Nullstellen in die Menge der Primzahlen (und ihrer Potenzen) meinen.

Die folgende Formel für die inverse Riemann-Transformation zeigt sehr beeindruckende Peaks an den Positionen der Primzahlen und ihrer Potenzen. Die Formel zeigt das "Inverse Riemann-Spektrum" der Nullstellen der Zeta-Funktion, wenn wir von der "Zeta-Domäne" zur "Primzahlen-Domäne" gehen:

$$f(x) = nFact * \sum_{n=1}^{nMax} \cos(\Im(\rho_n) \cdot \ln(x))$$

$\Im(\rho_n)$: Imaginärteile der n-ten Nullstelle der Zeta-Funktion.

nFact: Normalisierungsfaktor (z. B. $\frac{\ln(nMax)}{nMax}$).

Die Summe geht über die Imaginärteile der nicht-trivialen Nullstellen der Riemannschen Zeta-Funktion ($nMax$ Stück). Je mehr Nullstellen für die Summation verwendet werden, desto ausgeprägter erscheinen Peaks an den Positionen der

Primzahlen und deren Potenzen. Die Länge des zu einer Primzahlpotenz p^n gehörenden Peaks scheint ein $\frac{1}{n}$ des Wertes für die ursprüngliche Primzahl zu haben.

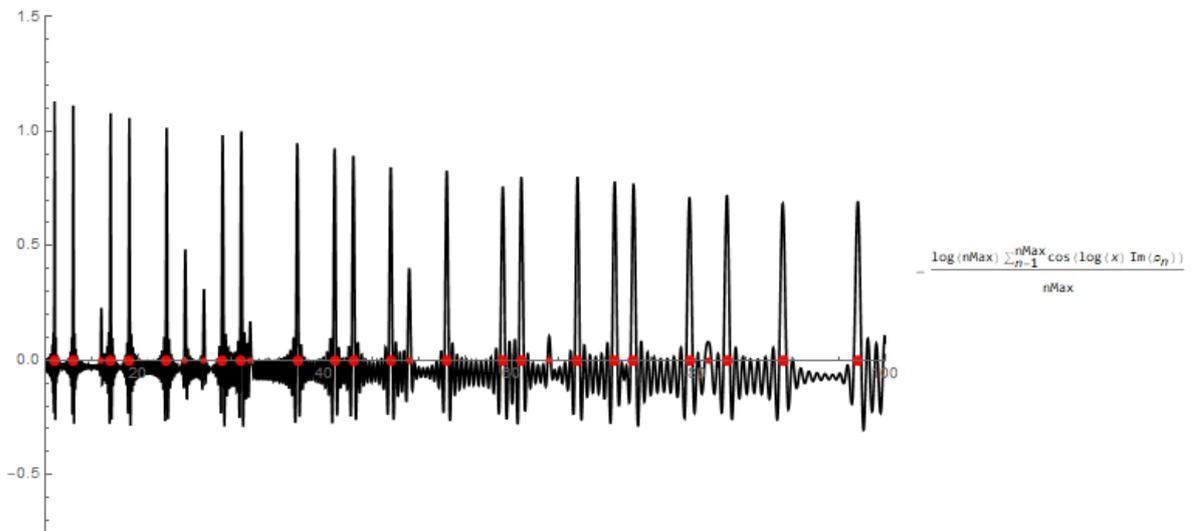
Wenn n gegen ∞ geht, ähnelt die Funktion einer Art Delta-Funktion (bekannt aus der Physik). Wenn es einem begabten Mathematiker gelingen sollte, das Integral über $f(x)$ zu berechnen, könnte man eine Formel für $\pi(x)$ erhalten (streng genommen für die Funktion $J(x)$, $\pi(x)$ kann jedoch leicht von $J(x)$ mit Hilfe der Moebius-Inversionsformel abgeleitet werden).

Hier ist ein Beispiel, das mit den ersten 400 Nullstellen der Zeta-Funktion berechnet wurde, wobei $f(x)$ im Bereich von 10 bis 100 liegt. Primzahlen und Primzahlpotenzen sind durch große und kleine rote Kreise gekennzeichnet:

Mathematica:

```
ClearAll[myFunc];
nMax=400;
o[k_]:=Im[ZetaZero[k]];
myFunc[p_]:=-1*Log[nMax]/nMax*Sum[Cos[o[n]*Log[p]],{n,1,nMax}]
xmin=10;xmax=100;ymin=-0.75;ymax=1.5;
grPlot=Plot[myFunc[x],{x,xmin,xmax},PlotRange-
>{{xmin,xmax},{ymin,ymax}},PlotStyle->Black,ImageSize-
>Large];//AbsoluteTiming
grPrimeMarkers=ListPlot[Table[{Prime[i],0},{i,PrimePi[xmin]+1,Pr
imePi[xmax]}],PlotStyle->{Red,PointSize[Large]},ImageSize-
>Large];
primePowers=Select[Range[xmin,xmax],PrimePowerQ[#]==True&&PrimeQ
[#]==False&];
grPrimePowerMarkers=ListPlot[Table[{primePowers[[i]],0},{i,1,Len
gth[primePowers]}],PlotStyle->{Red,PointSize[Medium]},ImageSize-
>Large,PlotLegends->Placed["Inverse Riemann spectrum shows prime
resonances. Range: 10 - 100\nno. of zeros of Zeta-function used:
nMax= "<>ToString[nMax],Above]];
ClearAll[nMax];
grFormula=ListPlot[{}],ImageSize->Large,PlotLegends-
>TraditionalForm[myFunc[x]],ImageSize->Large];
Show[grPlot,grPrimeMarkers,grPrimePowerMarkers,grFormula]
```

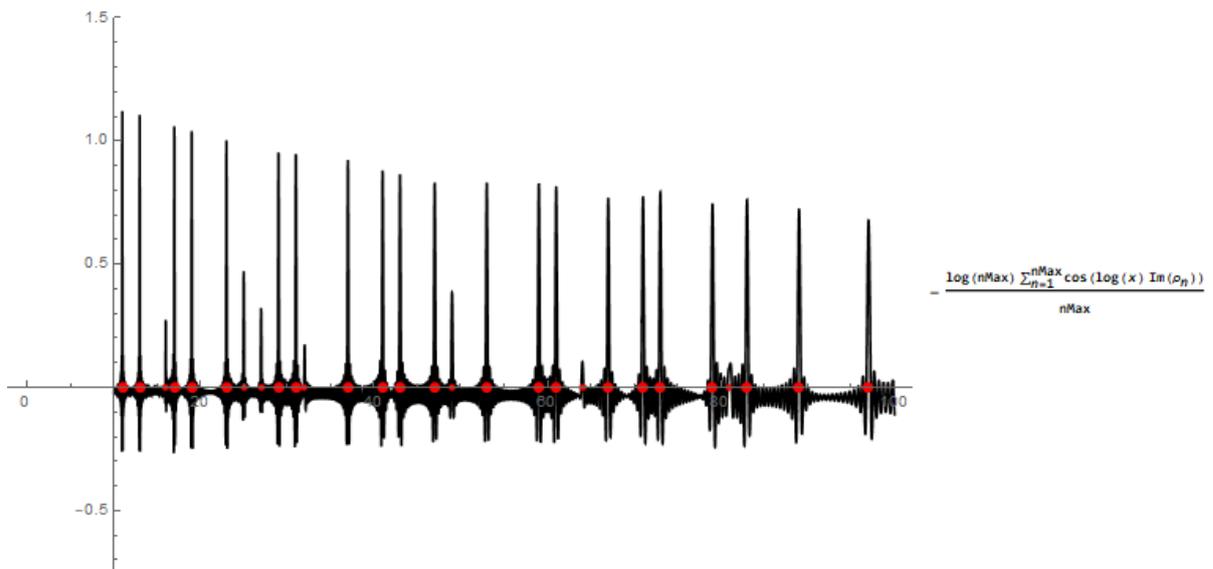
Inverse Riemann spectrum shows prime resonances. Range: 10 - 100
 no. of zeros of Zeta-function used: nMax= 400



Noch ein paar weitere Beispiele:

Wir verwenden die ersten 800 Nullstellen der Zeta-Funktion. $f(x)$ liegt im Bereich von 10 bis 100. Primzahlen und Primzahlpotenzen sind durch große und kleine rote Kreise gekennzeichnet:

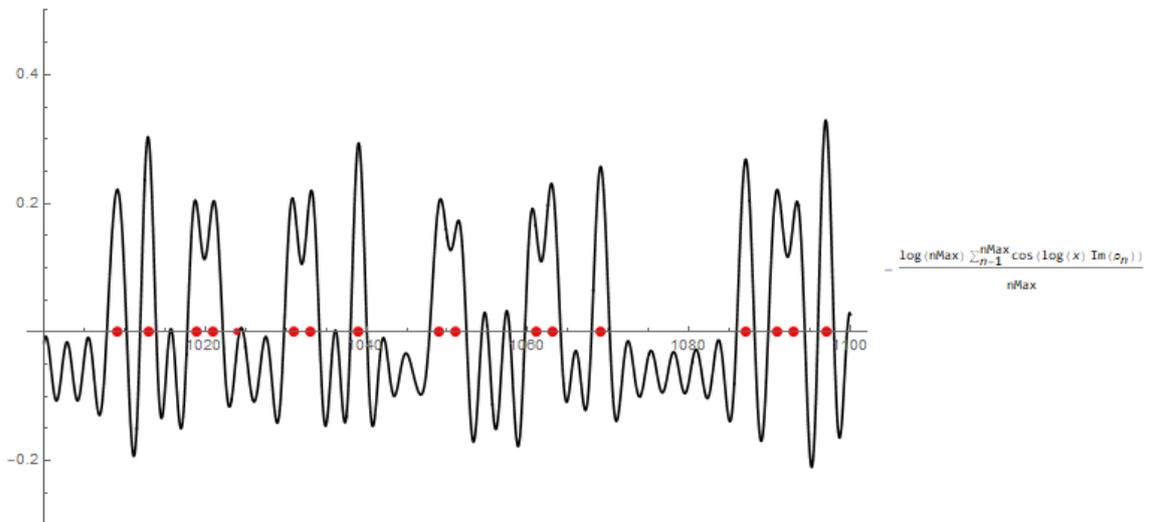
Inverse Riemann spectrum shows prime resonances. Range : 10 - 100
 no. of zeros of Zeta - function used : nMax = 800



In höheren Zahlenbereichen benötigen wir mehr Nullstellen.

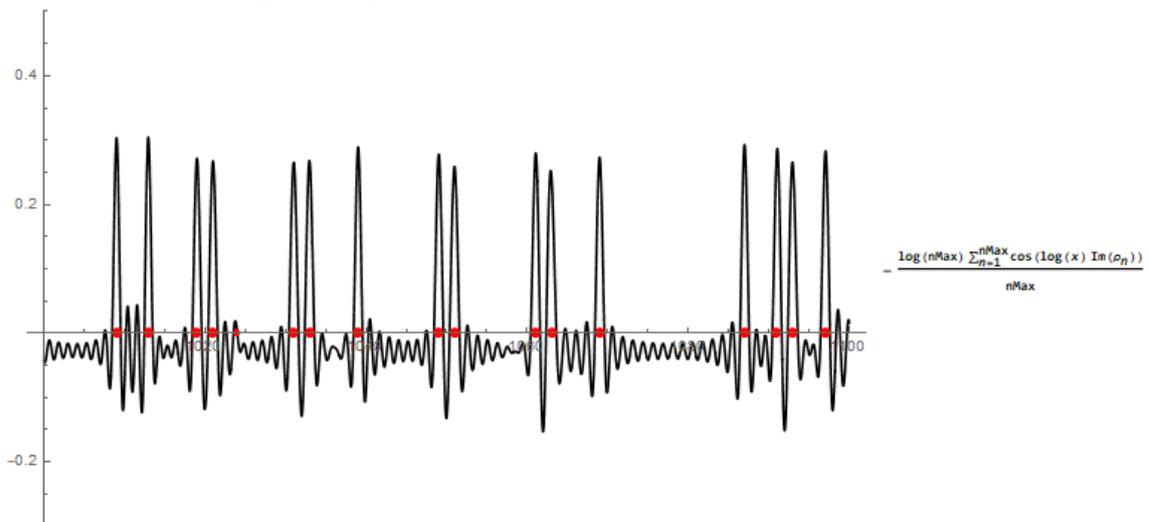
Wir verwenden hier 2000 Nullstellen der Zeta-Funktion. $f(x)$ liegt im Bereich von 1000 bis 1100:

Inverse Riemann spectrum shows prime resonances. Range: 1000 - 1100
 no. of zeros of Zeta-function used: nMax= 2000



Und hier ein Beispiel mit 5000 Nullstellen der Zeta-Funktion. $f(x)$ liegt im Bereich von 1000 bis 1100:

Inverse Riemann spectrum shows prime resonances. Range : 1000 - 1100
 no. of zeros of Zeta - function used : nMax = 5000



Die originale, in der Literatur zufindende Formel für die Riemann-Transformation lautet:

$$f(x) = -\frac{1}{\pi} \sum_{p^n} \frac{\ln(p)}{p^2} \cos(x \cdot \ln(p^n))$$

Hinweis: p^n bedeutet: Die Summation läuft über alle Primzahlen und deren Potenzen mit Werten bis zu einer vorgegebenen Grenze, p läuft über alle Primzahlen (ohne Potenzen) bis zu dieser Grenze.

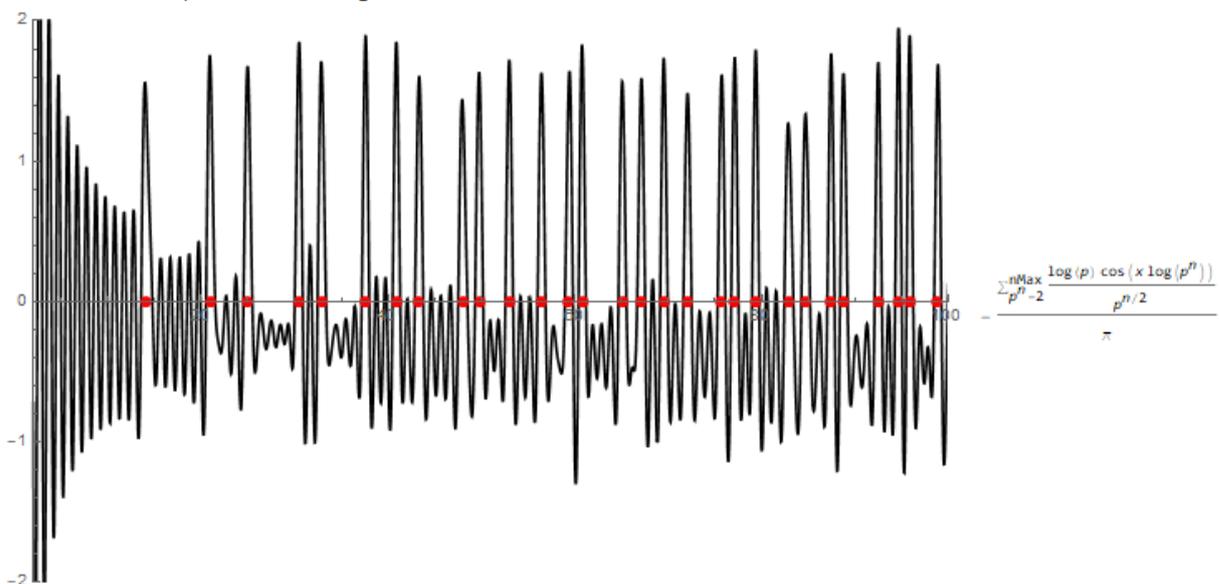
Wiederum zeigen die Funktionsgraphen für $f(x)$ ausgeprägte Peaks. Die Peaks befinden sich an den Positionen der Imaginärteile der Nullstellen der Zeta-Funktion. Je mehr Primzahlen und Primzahlpotenzen für die Summation verwendet werden, desto stärker erscheinen die Peaks bei den Nullstellen. Es gibt ein schnelles oszillierendes "Grundrauschen", das tendenziell negative Werte hat.

Hier ist ein Beispiel, das die ersten 100 Primzahlen auswertet. Der Plotbereich geht von 2 bis 100. Die Positionen der Nullstellen sind mit roten Kreisen markiert:

Mathematica:

```
ClearAll[myFunc]; Clear[nMax];
Clear[grPlot]; Clear[grZeroMarkers]; Clear[p]; Clear[n];
pmax=100; primeRange=Prime[pmax];
pP=Select[Range[primeRange], PrimePowerQ]; nmax=Length[pP];
myFunc[s_]:= -1/Pi*Sum[Log[FactorInteger[pP[[n]]][[1,1]]]*pP[[n]]^(-
1/2)*Cos[s*Log[pP[[n]]]], {n,1,nmax}];
myFuncLegend[s_]:= -1/Pi*Sum[Log[p]/p^(n/2)*Cos[s*Log[p^n]], {p^n,2,nMax}];
xmin=2; xmax=100; ymin=-2; ymax=2;
firstZetaIndexInst=FindInstance[N[Im[ZetaZero[n]]]>=xmin,n]; firstZetaIndex=n/.
firstZetaIndexInst[[1]];
lastZetaIndexInst=FindInstance[N[Im[ZetaZero[n]]]>=xmax,n]; lastZetaIndex=n/.las
tZetaIndexInst[[1]];
lastZetaIndex--; numberOfZetaZeros=lastZetaIndex-firstZetaIndex+1;
grPlot=Plot[myFunc[x], {x,xmin,xmax}, PlotRange-
>{{xmin,xmax},{ymin,ymax}}, PlotStyle->Black, ImageSize->Large]; //AbsoluteTiming
grZeroMarkers=ListPlot[Table[{Im[ZetaZero[i]],0}, {i,firstZetaIndex,lastZetaInd
ex}], PlotStyle->{Red, PointSize[Large]}, ImageSize->Large,
PlotLegends->Placed["Riemann spectrum shows resonances in the Zeta
domain\nClear peaks at "<>ToString[numberOfZetaZeros]<>" zero-positions\nno.
of primes and prime powers used: nmax= "<>ToString[nmax]<>"\n nmax:
"<>ToString[pmax]<>", xRange: "<>ToString[xmin]<>"-"<>ToString[xmax], Above]];
grFormula =ListPlot[{}, ImageSize->Large, PlotLegends-
>TraditionalForm[myFuncLegend[x]], ImageSize->Large];
Show[grPlot,grZeroMarkers,grFormula]
```

```
Riemann spectrum shows resonances in the Zeta domain
Clear peaks at 29 zero-positions
no. of primes and prime powers used: nmax= 121
pmax: 100, xRange: 2-100
```



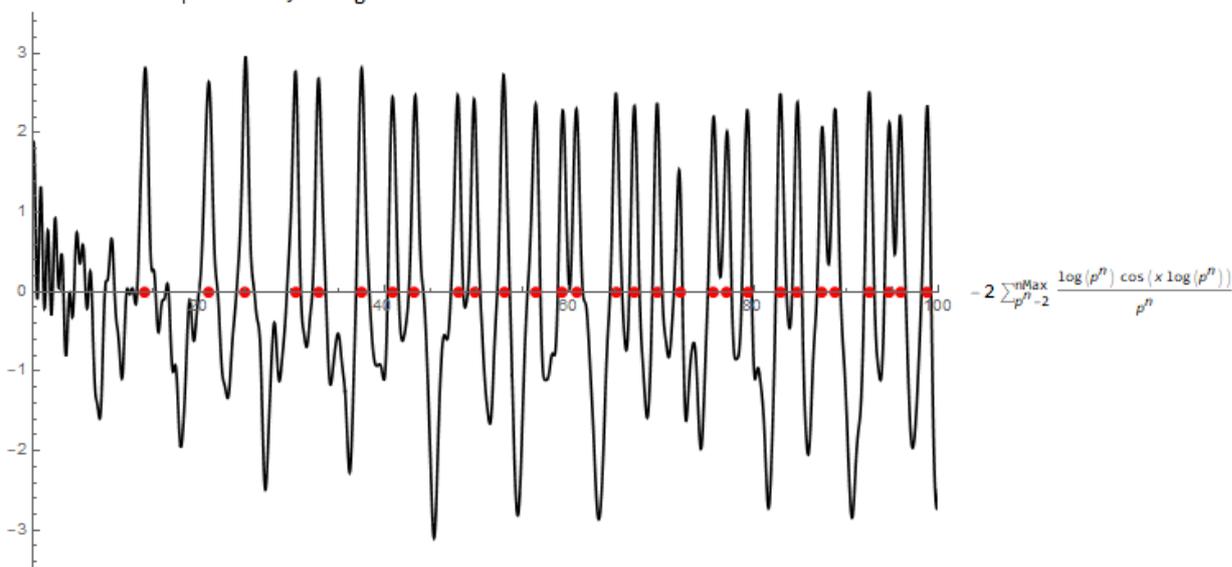
Das oszillierende ‘Grundrauschen’ kann unterdrückt werden, indem die Formel für die Riemann-Transformation leicht modifiziert wird:

Wir ersetzen den Term $\frac{\ln(p)}{p^{\frac{n}{2}}}$ durch $\frac{\ln(p^n)}{p^n}$ und den ‘Normalisierungsfaktor’ $\frac{1}{\pi}$ durch den Faktor 2 und bekommen so die leicht veränderte Formel:

$$f(x) = -2 \sum_{p^n} \frac{\ln(p^n)}{p^n} \cos(x \cdot \ln(p^n))$$

Die Summe geht jetzt über alle Primzahlen und deren Potenzen. Das Ergebnis ist ein glatterer Funktionsgraph. Der Nachteil ist, dass Nullstellen mit kleinen Abständen nicht deutlich genug ‘erfasst’ werden. Hier ein Beispiel für den resultierenden Funktionsgraphen, der sehr scharfe Peaks für die ersten 29 Nullstellen zeigt, wenn die ersten 500 Primzahlen verwendet werden (die Positionen der Nullstellen sind durch rote Kreise gekennzeichnet):

Riemann spectrum shows resonances in the Zeta domain
 Clear peaks at 29 zero-positions
 no. of primes and prime powers used: nMax= 538
 pmax: 500, xRange: 2-100



8.9_GLEICHUNGSSYSTEME MIT INTEGER-SEQUENZEN ALS LÖSUNGEN (VANDERMONDE-MATRIZEN)

8.9.1_LÖSUNGSVEKTOR: FOLGE DER NATÜRLICHE ZAHLEN

Wir betrachten Gleichungssysteme der Art:

$$1: x_1 + x_2 + x_3 + \dots + x_n = a_1 = 1$$

$$2: x_1^2 + x_2^2 + x_3^2 + \dots + x_n^2 = a_2$$

$$3: x_1^3 + x_2^3 + x_3^3 + \dots + x_n^3 = a_3$$

⋮

$$n: x_1^n + x_2^n + x_3^n + \dots + x_n^n = a_n$$

Diese Matrix wird in der Literatur als Vandermonde-Matrix bezeichnet. Die Lösungen dieses Gleichungssystems besitzen die Form:

$$a_n = r_1^n + r_2^n + r_3^n + \dots + r_n^n, \text{ wobei } \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} \text{ der Lösungsvektor und } r_n \text{ die } n \text{ komplexen}$$

Wurzeln eines Polynoms p_n der Ordnung n ist:

$$p_n(y) = c_0 + c_1 y + c_2 y^2 + \dots + c_n y^n$$

Das Polynom $p_n(y)$ hängt in typischer Weise von der Folge $\mathbf{intF}_n = \mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3, \dots, \mathbf{a}_n$ ab.

Denken wir uns die Konstanten $c_{0,n}, c_{1,n}, c_{2,n}, \dots, c_{n,n}$ als die unterste Zeile einer Dreiecksmatrix. Die Spitze dieser Dreiecksmatrix (oberste Linie) besteht aus dem einzigen Wert den Wert 1. Dann lassen sich aus der Kenntnis der ersten Diagonalreihe c_{0k} (von oben nach links unten) der Dreiecksmatrix **alle** Werte dieser Dreiecksmatrix (und somit auch der untersten Zeile) berechnen. Konkret berechnet sich z. B. die Zeile beim Index i aus der Vorgängerzeile mit dem Index $i - 1$ wie folgt:

$$\text{Sei } c_{0,i} \text{ vorgegeben, dann ist } \mathbf{c}_{k,i} = \mathbf{c}_{k-1,i-1} \cdot (\mathbf{i} - \mathbf{1}), \text{ mit } k = 1, 2, \dots, i - 1$$

Die Werte der ersten Diagonalreihe (in absteigender Reihenfolge) vom obersten Wert 1 aus nennen wir $\mathbf{Coeff}_{0,k} = \mathbf{c}_{0,1}, \mathbf{c}_{0,2}, \mathbf{c}_{0,3}, \dots, \mathbf{c}_{0,n}$ (mit $k \leq n$), da sie jeweils den konstanten Term eines Polynoms $p_k(y) = c_{0k} + c_{1k}y + c_{2k}y^2 + \dots + c_{nk}y^n$ darstellen. Uns soll hier nicht die Berechnung der Polynome $p_n(y)$ und deren Nullstellen interessieren, sondern der Zusammenhang **zwischen $\mathbf{c}_{0,k}$ und \mathbf{a}_k** . Interessant ist auch, dass die Folge \mathbf{a}_k über \mathbf{a}_n hinaus erweitert werden kann (obwohl zunächst nur \mathbf{a}_1 bis \mathbf{a}_n vorgegeben war). Die Werte der Folgenglieder über \mathbf{a}_n hinaus ($\mathbf{a}_{n+1}, \mathbf{a}_{n+2}, \dots$) stellen sich als rationale Zahlen heraus.

Die Verbindung zwischen \mathbf{intF}_n und $\mathbf{Coeff}_{0,k}$ stellt eine Transformation dar, mit der Integer- (und auch rationale) Zahlenfolgen hin- und her transformiert werden können. Im Folgenden sei mit der Abkürzung ‚PEQS‘ der Begriff ‚Polynomial EQUation System‘ gemeint. Unter der PEQS-Transformation verstehen wir somit die Transformation, mit der die Folgen \mathbf{intF}_n und $\mathbf{Coeff}_{0,k}$ ineinander transformiert werden können. Hier ein Beispiel für ein PEQS-Gleichungssystem der Ordnung n , dessen Lösungen die Menge der natürlichen Zahlen von 1 bis n ist:

$$1: x_1 + x_2 + x_3 + \dots + x_n = 1$$

$$2: x_1^2 + x_2^2 + x_3^2 + \dots + x_n^2 = 2$$

$$3: x_1^3 + x_2^3 + x_3^3 + \dots + x_n^3 = 3$$

⋮

$$n: x_1^n + x_2^n + x_3^n + \dots + x_n^n = n$$

Ein konkretes Beispiel: $n = 5$. $\mathit{intF}_k = a_k = \{1, 2, 3, 4, 5\}$. Dann ergibt sich folgende Dreiecksmatrix, deren Zeilenwerte jeweils die Koeffizienten von Polynomen der Ordnung 0 bis 5 sind:

{ PEQS triangle based on natural numbers up to 5, }

$$\begin{array}{cccccc}
 & & & & & 1 \\
 & & & & -1 & 1 \\
 & & -1 & -2 & 2 & \\
 & -1 & -3 & -6 & 6 & \\
 1 & -4 & -12 & -24 & 24 & \\
 19 & 5 & -20 & -60 & -120 & 120
 \end{array}$$

Die erste Diagonalreihe lautet: $\mathit{Coeff}_{0,k} = \{1, -1, -1, -1, 1, 19\}$

$p_5(y) = 19 + 5y - 20y^2 - 60y^3 - 120y^4 + 120y^5$ mit den 5 komplexen Nullstellen:

roots
 $= \{-0.577053079542094, 0.5579610065070687, 1.3884072929663431, -0.18465760996565891 - 0.5657660375057798i, -0.18465760996565891 + 0.5657660375057798i\}$

Die 5 Lösungen des Gleichungssystems lauten nun:

$$\mathit{seqFunc}[n] = (-0.577\dots)^n + (0.558\dots)^n + (1.39\dots)^n + (-0.185\dots - 0.566\dots i)^n + (-0.185\dots + 0.566\dots i)^n$$

Wobei wir für n die Werte von 1 bis 5 einsetzen. Die ersten 8 Werte von $\mathit{seqFunc}[n]$ lauten:

$\{1, 2, 3, 4, 5, \frac{871}{120}, \frac{599}{60}, \frac{3313}{240}, \dots\}$ Man sieht, dass $\mathit{seqFunc}[n]$ für die ersten 5 Werte identisch mit intF_n ist (bzw. a_n , mit den Bezeichnungen von oben), jedoch für höhere Indizes weitergeführt werden kann.

In diesem Fall (a_n ist die Folge der natürlichen Zahlen) ist das Bildungsgesetz für $\mathit{Coeff}_{0,n}$ bekannt: $\mathit{Coeff}_{0,n} = -n! {}_1F_1(1 - n; 2; 1)$, wobei ${}_1F_1$ die Kummersche konfluente Hypergeometrische Funktion ist (A293226 in <https://oeis.org>).

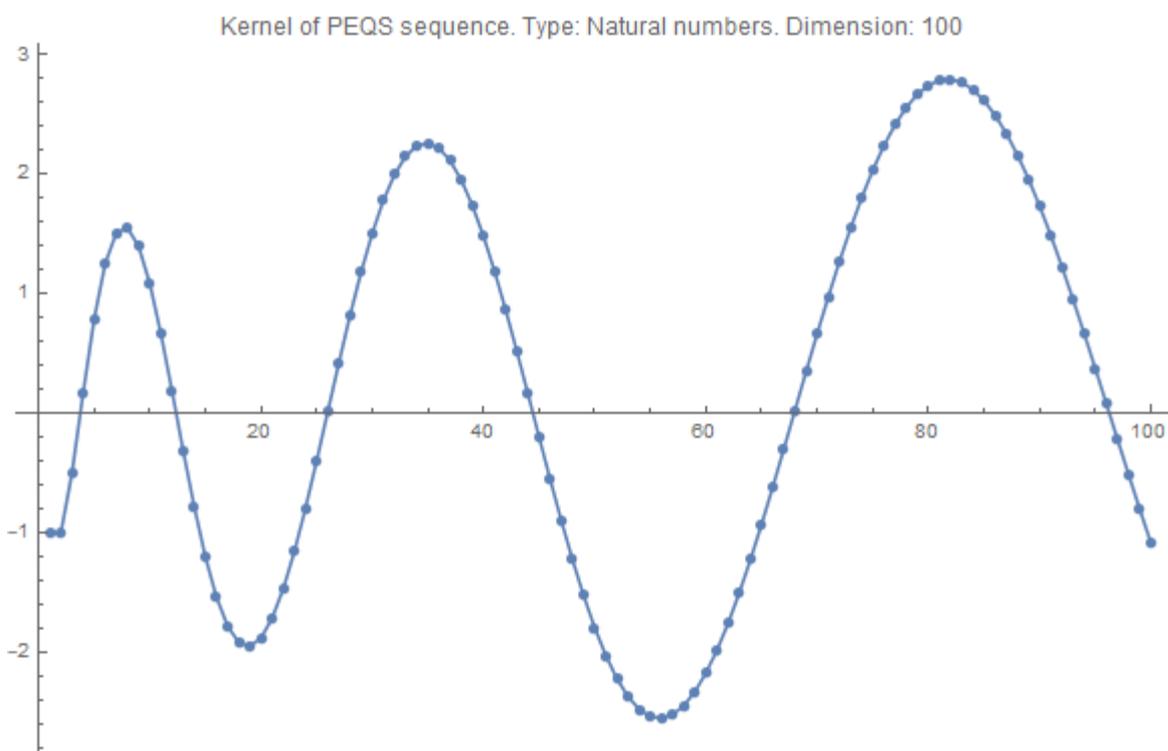
Die Folge $\mathit{seqFunc}[n]$ gehört auch zur Gruppe der linear rekursiven Funktionen mit dem Kernel:

$\{1, \frac{1}{2}, \frac{1}{6}, -\frac{1}{24}, -\frac{19}{120}\}$ und lässt sich auf diese Weise leicht berechnen.

Hinweis: Kehren wir das Konstruktionsprinzip der Dreiecksmatrix aus der ersten Diagonalreihe $\mathit{Coeff}_{0,n}$ um, dann lässt sich die Dreiecksmatrix (,in Richtung nach links oben‘) zu einer quadratischen Matrix erweitern. In dieser steht dann der negative Kernel der rekursiven Beziehung in umgekehrter Reihenfolge ($\{\frac{19}{120}, \frac{1}{24}, -\frac{1}{6}, -\frac{1}{2}, -1\}$ in der 1. bzw. 2. Zeile:

$$\begin{pmatrix} \frac{19}{120} & \frac{1}{24} & -\frac{1}{6} & -\frac{1}{2} & -1 & 1 \\ \frac{19}{120} & \frac{1}{24} & -\frac{1}{6} & -\frac{1}{2} & -1 & 1 \\ \frac{19}{60} & \frac{1}{12} & -\frac{1}{3} & -1 & -2 & 2 \\ \frac{19}{20} & \frac{1}{4} & -1 & -3 & -6 & 6 \\ \frac{19}{5} & 1 & -4 & -12 & -24 & 24 \\ 19 & 5 & -20 & -60 & -120 & 120 \end{pmatrix}$$

Die rationalen Werte des Kerns ergeben sich außerdem aus den Quotienten des ersten und letzten Koeffizienten c_0 und c_n des n -ten Polynoms (jeweils in roter und blauer Farbe markiert). Der Graph der ersten 100 Kernel-Werte sieht wie folgt aus:



Das wirkliche Problem bei diesen Berechnungen ist die Bestimmung von $\mathbf{Coeff}_{0,n}$ aus einem vorgegebenen Lösungsvektor \mathbf{intF}_k . Für den Fall der natürlichen Zahlen wurde oben eine direkte Berechnungsmethode mit Hilfe der Hypergeometrischen Funktion angegeben. Es geht für diesen Fall jedoch sogar noch leichter:

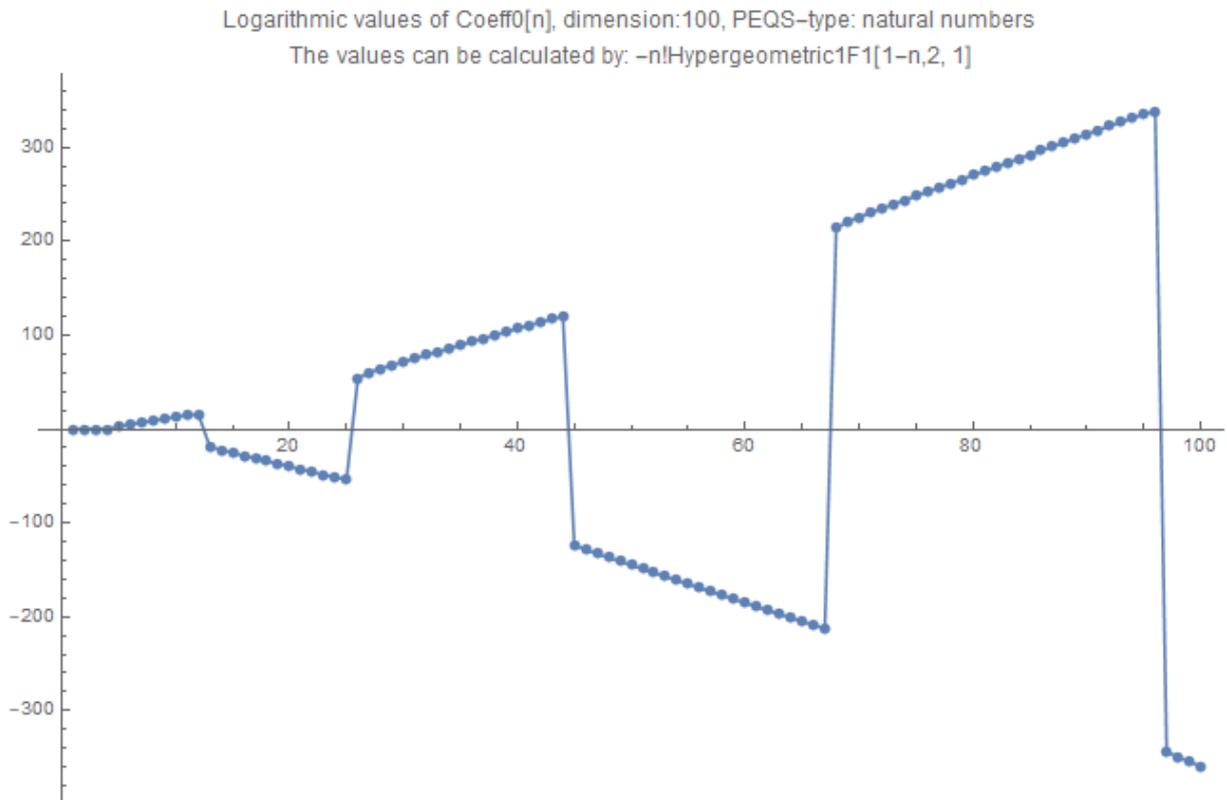
Sei \mathbf{mat}_n die Folge:

$$(1), \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}, \begin{pmatrix} 1 & 1 & 2 \\ 1 & 2 & 1 \\ 1 & 1 & 3 \end{pmatrix}, \begin{pmatrix} 1 & 1 & 2 & 3 \\ 1 & 2 & 1 & 2 \\ 1 & 1 & 3 & 1 \\ 1 & 1 & 1 & 4 \end{pmatrix}, \begin{pmatrix} 1 & 1 & 2 & 3 & 4 \\ 1 & 2 & 1 & 2 & 3 \\ 1 & 1 & 3 & 1 & 2 \\ 1 & 1 & 1 & 4 & 1 \\ 1 & 1 & 1 & 1 & 5 \end{pmatrix}, \begin{pmatrix} 1 & 1 & 2 & 3 & 4 & 5 \\ 1 & 2 & 1 & 2 & 3 & 4 \\ 1 & 1 & 3 & 1 & 2 & 3 \\ 1 & 1 & 1 & 4 & 1 & 2 \\ 1 & 1 & 1 & 1 & 5 & 1 \\ 1 & 1 & 1 & 1 & 1 & 6 \end{pmatrix}, \dots$$

mit dem Bildungsgesetz $mat[i,j]=\begin{cases} i, & \text{wenn } i == j \\ 1, & \text{wenn } i > j \\ j - i, & \text{sonst} \end{cases}$

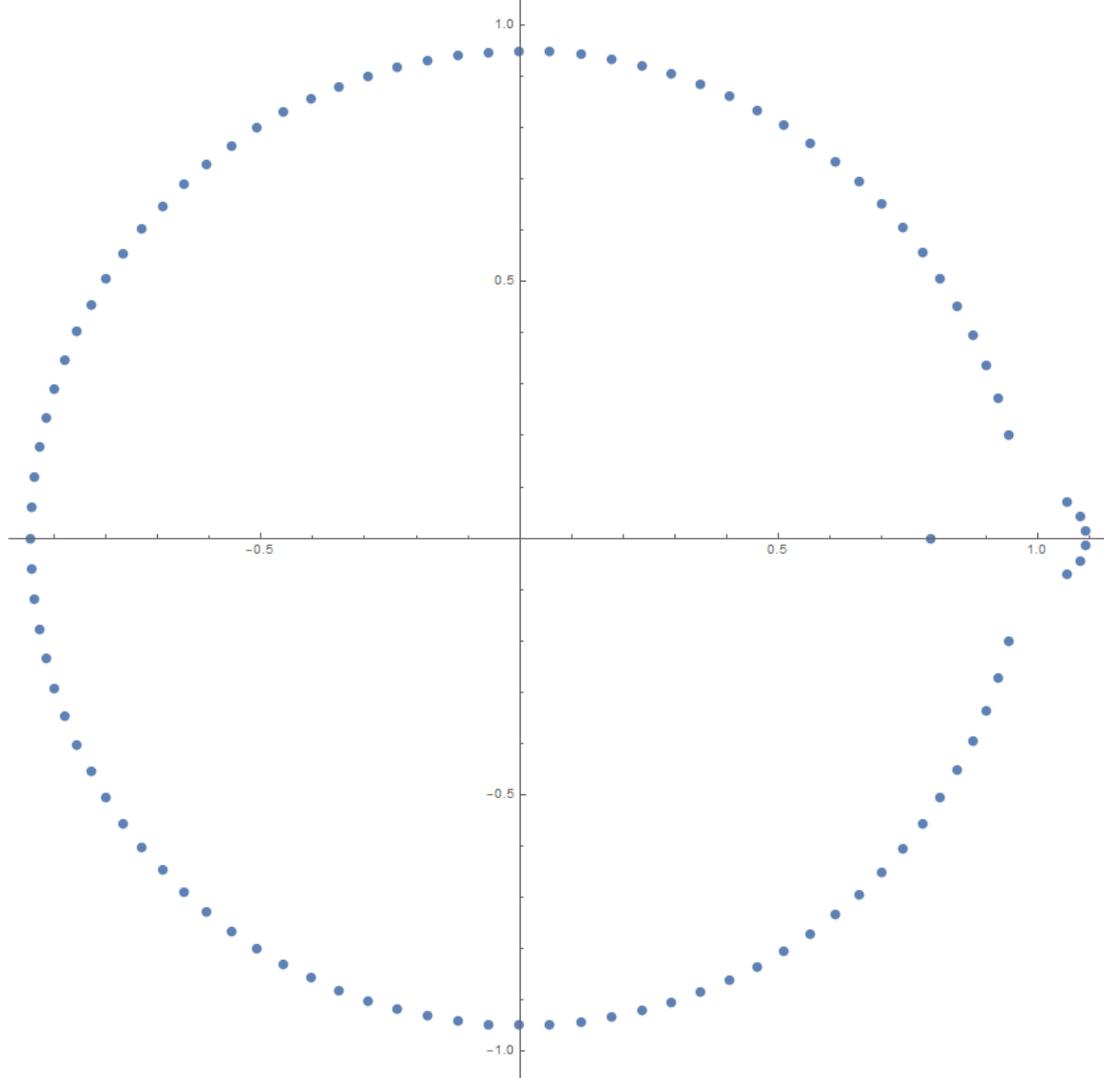
dann gilt: $\mathbf{Det(mat_n) = Coeff_{0,n}}$

Der Graph der ersten 100 Coeff0-Werte sieht wie folgt aus:

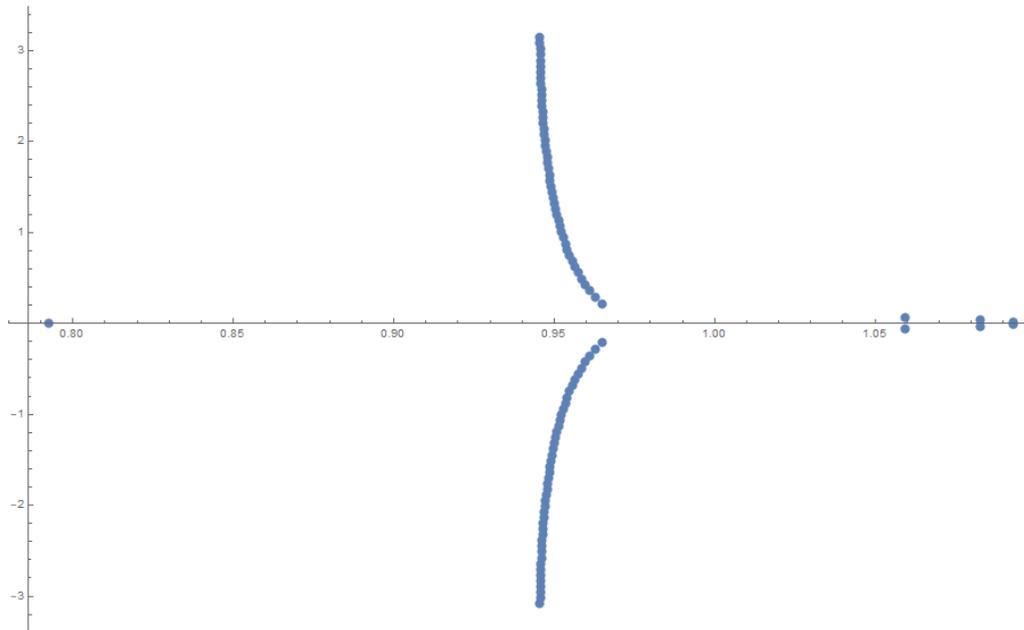


Was an diesem Beispiel für den konkreten Fall der Dimension 5 erläutert wurde lässt sich für beliebig hohe Dimensionen erweitern. Hier noch ein Beispiel für $n = 100$ mit einer graphischen Darstellung der komplexen Nullstellen (in kartesischen und Polarkoordinaten):

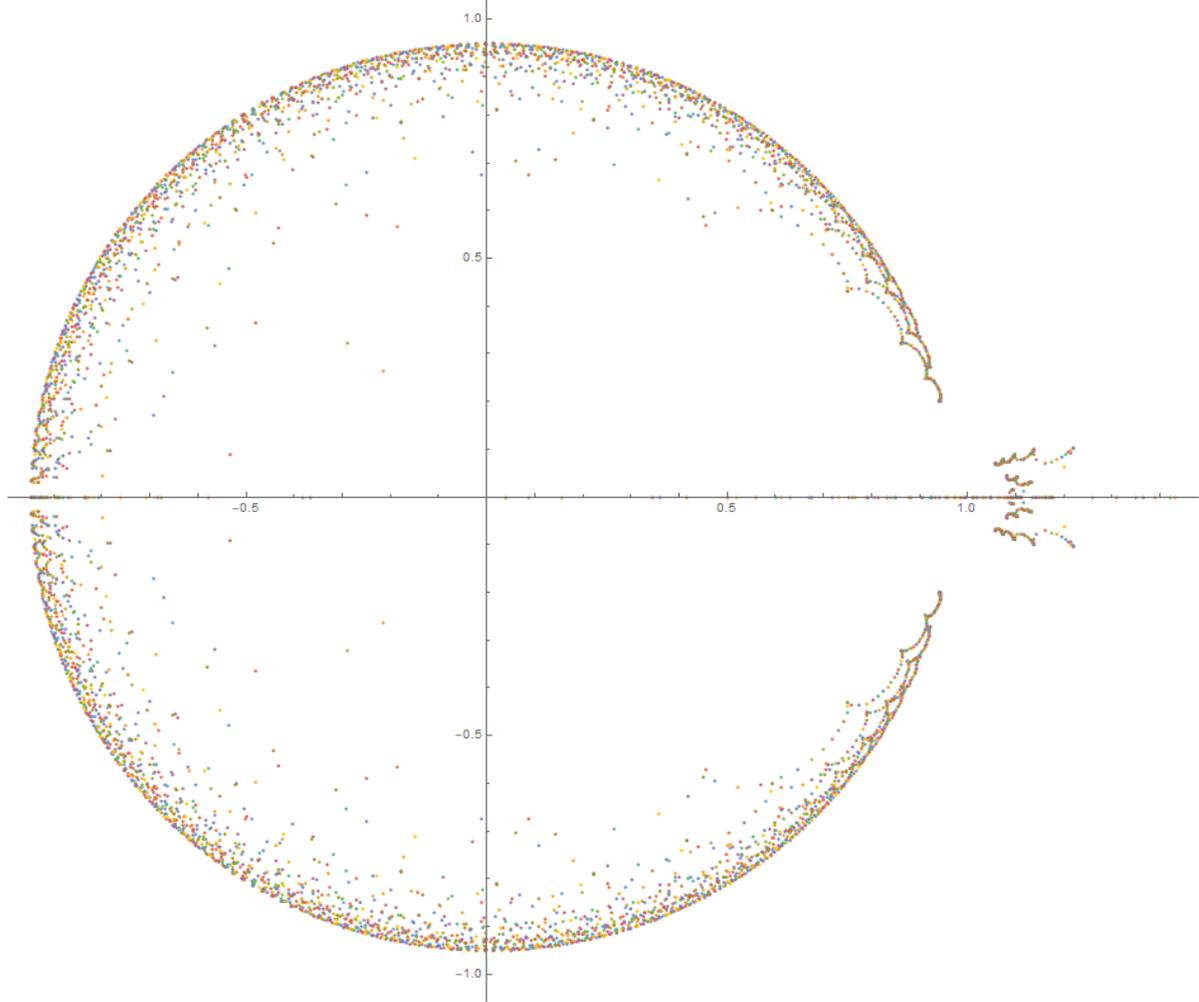
location (cart. coordinates) of the complex roots for qSequence type 1
Dimension: 100



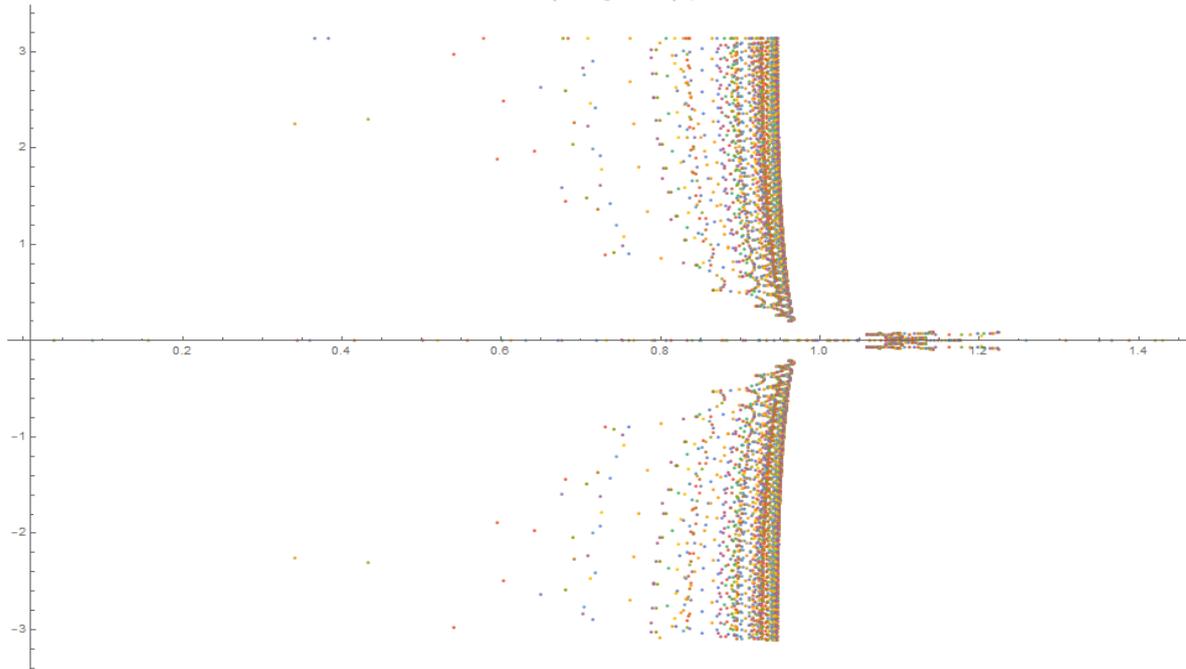
location (polar coordinates) of the complex roots for qSequence type 1
Dimensions: 100



location (cart. coordinates) of the complex roots for qSequence type 1
Dimensions: 2 (=Plotlegend no 1) up to 100



location (polar coordinates) of the complex roots for qSequence type 1
Dimensions: 2 (=Plotlegend no 1) up to 100



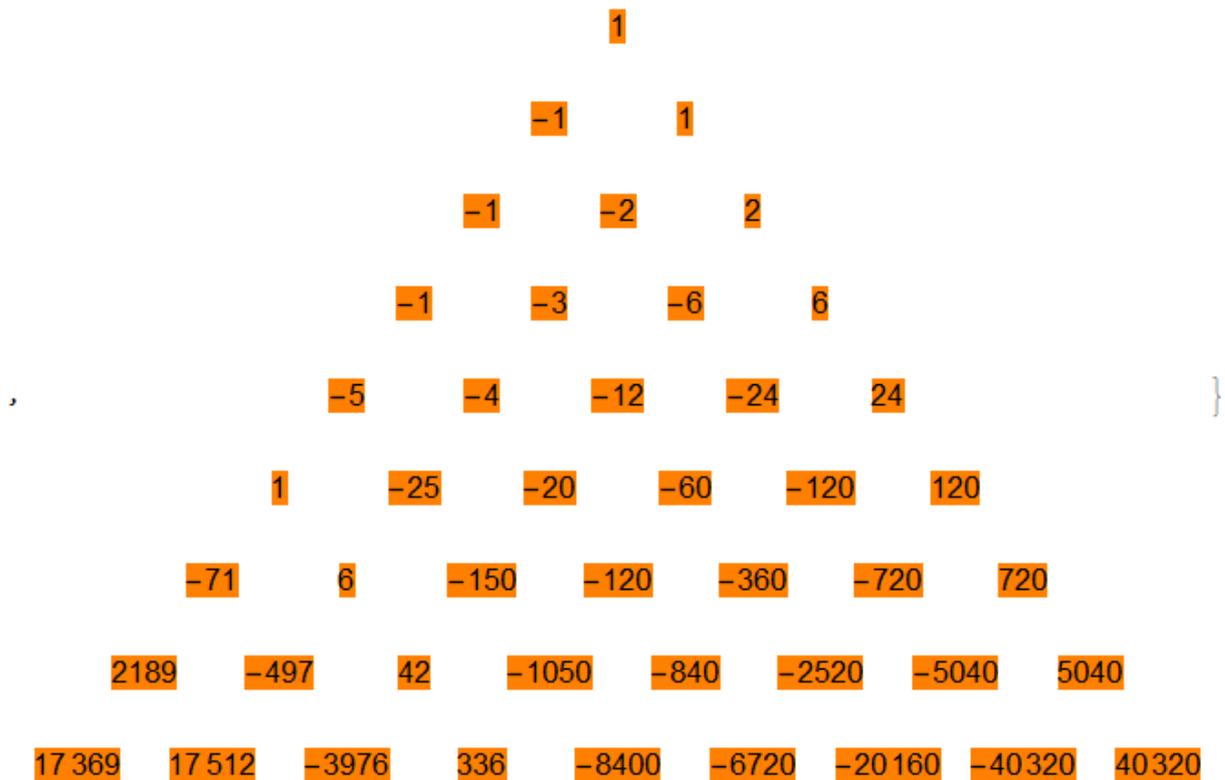
Und noch ein Blick auf die Primfaktorzerlegung der ersten 50 Werte von $\mathit{Coeff}_{0,n}$, die keinerlei Auffälligkeiten zeigt:

1	{{1,1}}
2	{{1,1}}
3	{{1,1}}
4	{{1,1}}
5	{{19,1}}
6	{{151,1}}
7	{{1091,1}}
8	{{7841,1}}
9	{{56519,1}}
10	{{223,1}, {1777,1}}
11	{{23,1}, {103,1}, {1031,1}}
12	{{7701409,1}}
13	{{23,1}, {6316067,1}}
14	{{5867,1}, {823787,1}}
15	{{59,1}, {1763664719,1}}
16	{{2002667085119,1}}
17	{{3362633,1}, {11035753,1}}
18	{{4289,1}, {158516607841,1}}
19	{{1801,1}, {6907445473321,1}}
20	{{103,1}, {49393,1}, {44771341561,1}}
21	{{2731,1}, {4993,1}, {54583,1}, {5583649,1}}
22	{{548079347,1}, {136339117843,1}}
23	{{67891,1}, {19047624174793631,1}}
24	{{19,1}, {70195849,1}, {15281774905949,1}}
25	{{2273,1}, {132632251,1}, {809726832587,1}}
26	{{59,1}, {1031,1}, {152791,1}, {28786290689309,1}}
27	{{29,1}, {137,1}, {17837,1}, {2431572221454948151,1}}
28	{{1091,1}, {13687,1}, {1702690571,1}, {351811724743,1}}
29	{{361702062324149751903132843499,1}}
30	{{2857,1}, {4674028378481509129971784783,1}}
31	{{61,1}, {16067,1}, {984103174279,1}, {490600467281563,1}}
32	{{457,1}, {941,1}, {7829,1}, {4884665371786076435244233,1}}
33	{{79,1}, {7349,1}, {976046331680316442389698582029,1}}
34	{{23,1}, {67,1}, {7331,1}, {1723166171858963399165217371761,1}}
35	{{195593,1}, {52855733933,1}, {242222367593,1}, {266946775787,1}}
36	{{23,1}, {998229455402129726075516494398995399207,1}}
37	{{10397987,1}, {394418522203,1}, {192099812273276678223331,1}}
38	{{239,1}, {43403,1}, {2596080359706103738851141696073679723,1}}
39	{{149,1}, {670844597,1}, {9124587114735743732707412704918043,1}}
40	{{228828199674511,1}, {132492806214159332040949011205711,1}}
41	{{6566801183,1}, {148066316705471672725042558703660116793,1}}
42	{{67,1}, {267193810720792588447,1}, {1621966420250627774099864251,1}}
43	{{19,1}, {38720220357651297635515977184282535937987689769821,1}}
44	{{997,1}, {30632132437,1}, {5724052011253,1}, {57737875048226218431440213,1}}
45	{{10712407481724409,1}, {47962155441999128904276109046869785709,1}}
46	{{65712222112317165183589964642284924435961435874518608889,1}}
47	{{38501,1}, {51461,1}, {39710001171251,1}, {62486369789949521475437558024641039,1}}
48	{{148279,1}, {36911387,1}, {910785131,1}, {63219590349635839362267416075137730033,1}}
49	{{199,1}, {146375891,1}, {10795417408492229,1}, {59936493382835752308549010189780921,1}}
50	{{53,1}, {137,1}, {3467,1}, {5903,1}, {91997,1}, {6439003,1}, {7639455163,1}, {2004907113617,1}, {806219781329822269,1}}

8.9.2_LÖSUNGSVEKTOR: FOLGE DER PRIMZAHLEN (MIT DER ZAHL 1)

Ein konkretes Beispiel: $n = 8$. $\mathit{int}F_k = a_k = \{1, 2, 3, 5, 7, 11, 13, 17\}$. Dann ergibt sich folgende Dreiecksmatrix, deren Zeilenwerte jeweils die Koeffizienten von Polynomen der Ordnung 0 bis 8 sind:

{ PEQS triangle based on Primenumbers up to dimension 8



Die erste Diagonalreihe lautet: $\mathit{Coeff}_{0,k} = \{1, -1, -1, -1, -5, 1, -71, 2189, 17369\}$

$p_8(y) = 17369 + 17512y - 3976y^2 + 336y^3 - 8400y^4 - 6720y^5 - 20160y^6 - 40320y^7 + 40320y^8$ mit den 8 komplexen Nullstellen

$$\{0.970..., 1.43..., -0.714... - 0.172...i, -0.714... + 0.172...i, -0.363... - 0.761...i, -0.363... + 0.761...i, 0.377... - 0.818...i, 0.377... + 0.818...i\}$$

Die 8 Lösungen des Gleichungsystems lauten nun:

$$\{0.970...^n + 1.43...^n + (-0.714... - 0.172...i)^n + (-0.714... + 0.172...i)^n + (-0.363... - 0.761...i)^n + (-0.363... + 0.761...i)^n + (0.377... - 0.818...i)^n + (0.377... + 0.818...i)^n\}$$

Wobei wir für n die Werte von 1 bis 8 einsetzen. Die ersten 12 Werte von $\mathit{seqFunc}[n]$ lauten:

$\{1, 2, 3, 5, 7, 11, 13, 17, \frac{207619}{8064}, \frac{496943}{13440}, \frac{470619}{8960}, \frac{18053729}{241920}, \dots\}$ Man sieht, dass $\mathit{seqFunc}[n]$ für die ersten 8 Werte identisch mit intF_n ist (bzw. a_n , mit den Bezeichnungen von oben), jedoch für höhere Indizes weitegeführt werden kann.

In diesem Fall (a_n ist die Folge der Primzahlen mit der 1) ist das Bildungsgesetz für $\mathit{Coeff}_{0,n}$ (noch) nicht bekannt. Die Folge $\mathit{seqFunc}[n]$ gehört auch zur Gruppe der linear rekursiven Funktionen mit dem Kernel:

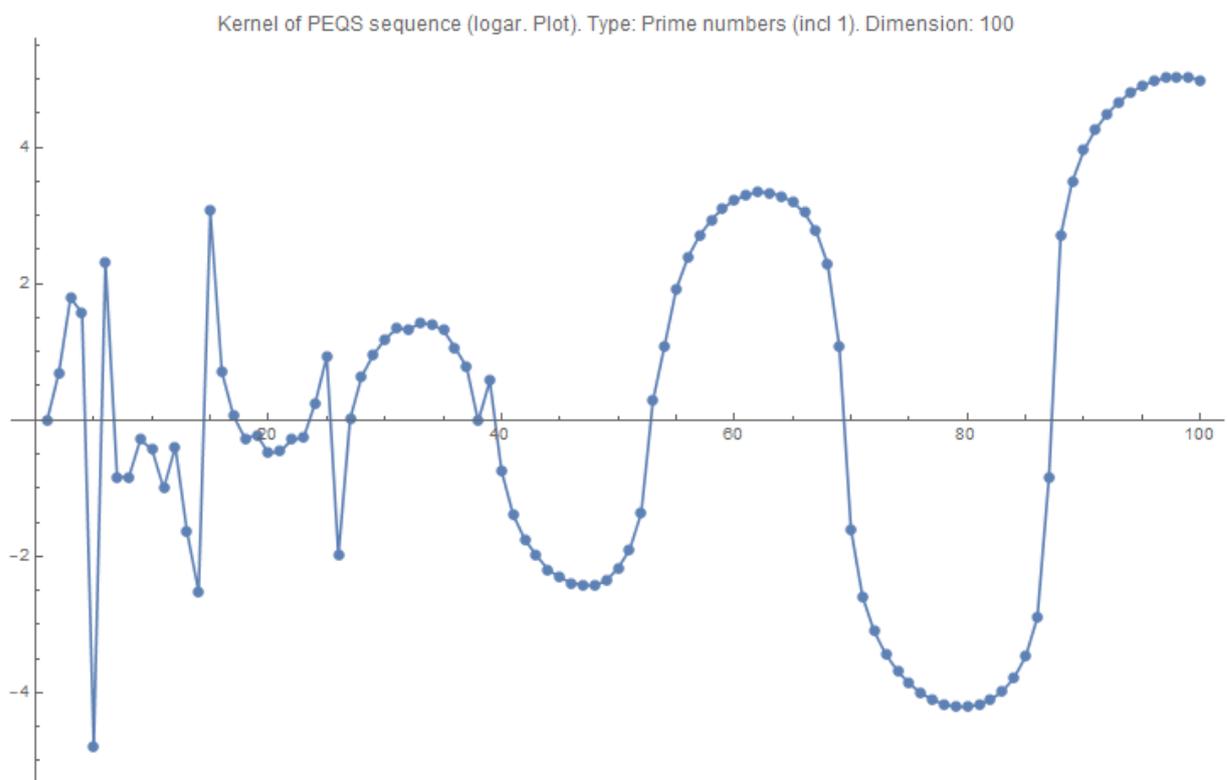
$$\{1, \frac{1}{2}, \frac{1}{6}, \frac{5}{24}, -\frac{1}{120}, \frac{71}{720}, -\frac{2189}{5040}, -\frac{17369}{40320}\}$$

Hinweis: Kehren wir das Konstruktionsprinzip der Dreiecksmatrix aus der ersten Diagonalreihe $\mathit{Coeff}_{0,n}$ um, dann lässt sich die Dreiecksmatrix (,in Richtung nach links oben‘) zu einer quadratischen Matrix erweitern:

$\frac{17369}{5040}$	$\frac{2189}{720}$	$-\frac{71}{120}$	$\frac{1}{24}$	$-\frac{5}{6}$	$-\frac{1}{2}$	-1	-1
$\frac{17369}{5040}$	$\frac{2189}{720}$	$-\frac{71}{120}$	$\frac{1}{24}$	$-\frac{5}{6}$	$-\frac{1}{2}$	-1	-1
$\frac{17369}{2520}$	$\frac{2189}{360}$	$-\frac{71}{60}$	$\frac{1}{12}$	$-\frac{5}{3}$	-1	-2	-2
$\frac{17369}{840}$	$\frac{2189}{120}$	$-\frac{71}{20}$	$\frac{1}{4}$	-5	-3	-6	-6
$\frac{17369}{210}$	$\frac{2189}{30}$	$-\frac{71}{5}$	1	-20	-12	-24	-24
$\frac{17369}{42}$	$\frac{2189}{6}$	-71	5	-100	-60	-120	-120
$\frac{17369}{7}$	2189	-426	30	-600	-360	-720	-720
17369	15323	-2982	210	-4200	-2520	-5040	-5040

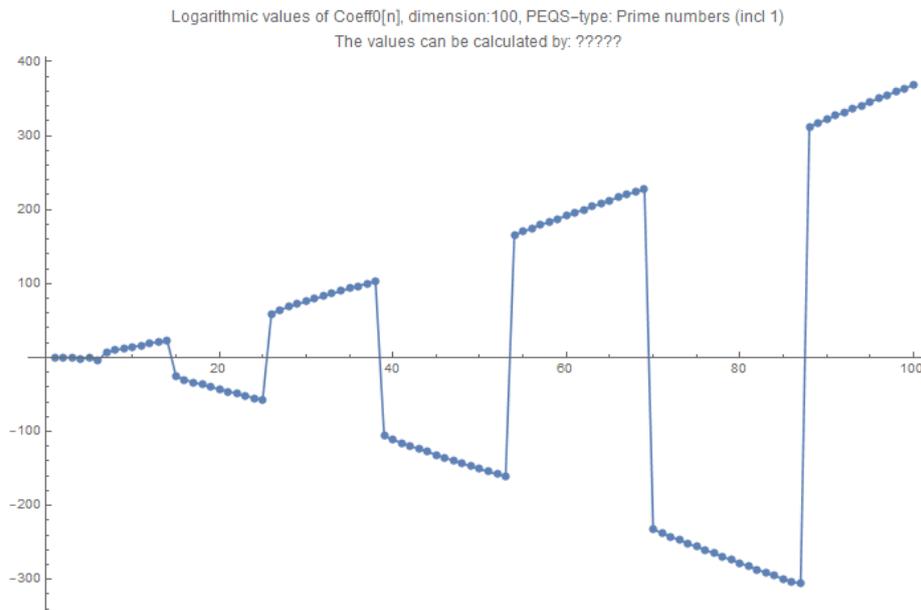
Die rationalen Werte des Kerns ergeben sich außerdem aus den Quotienten des ersten Koeffizienten c_0 (der n-ten Zeile) und c_n (der (n+1)-ten Zeile des (jeweils in roter und blauer Farbe markiert)). Die Kernel-Sequenz lässt sich auch einfach so schreiben: $\frac{Coeff0[n]}{n!}$

Der Graph der ersten 100 Kernel-Werte (logarithmiert) sieht wie folgt aus:



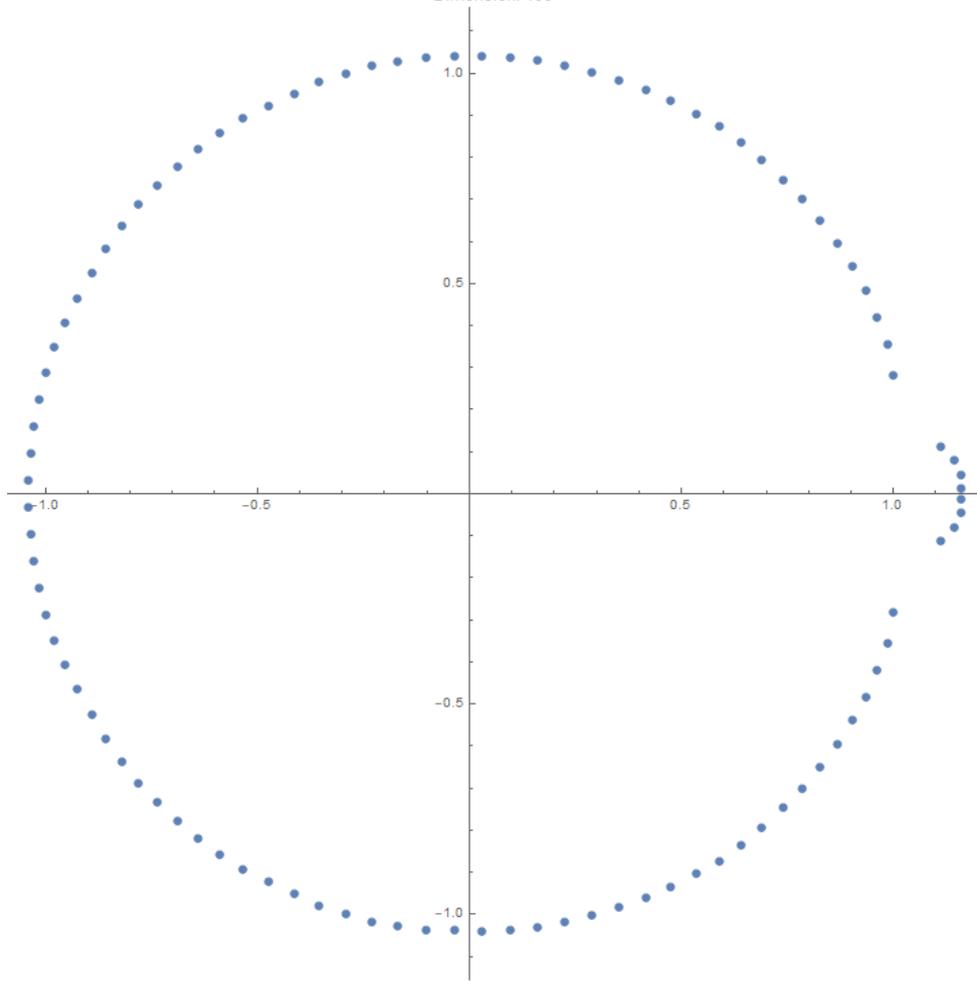
Nach anfänglichen Irregularitäten bis zu 40. Primzahl scheint der Graph einer glatten Funktion zu folgen.

Der Graph der ersten 100 Coeff0-Werte sieht wie folgt aus:

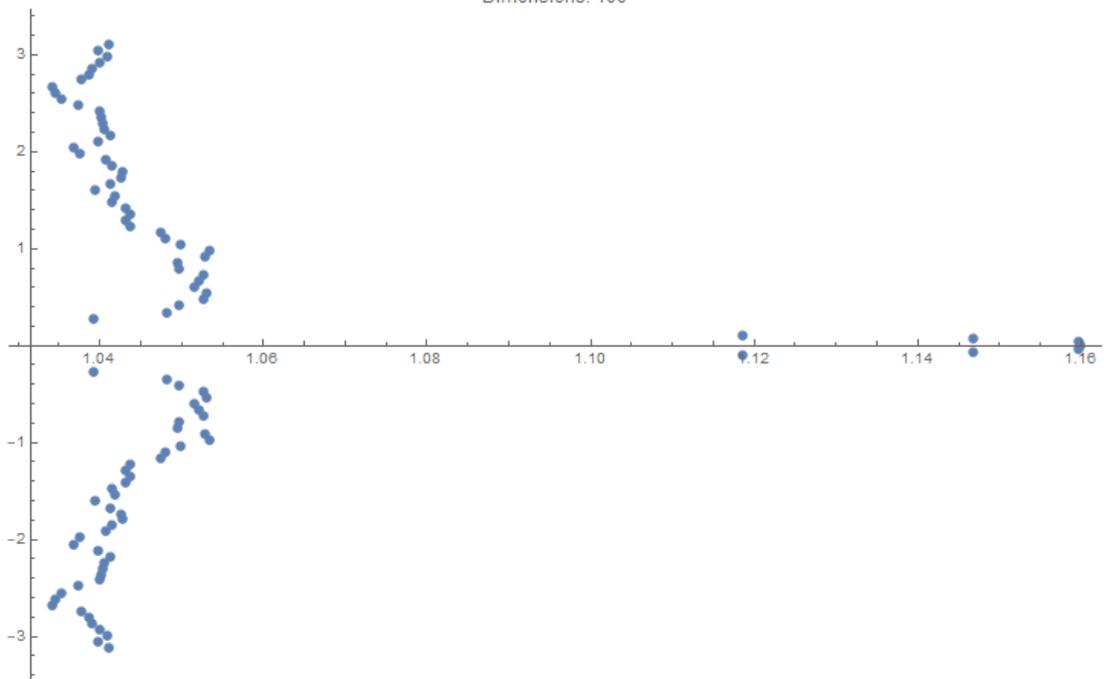


Hier noch ein Beispiel für $n = 100$ mit einer graphischen Darstellung der komplexen Nullstellen auf Basis der Folge der Primzahlen (mit ,1'), in kartesischen und Polarkoordinaten:

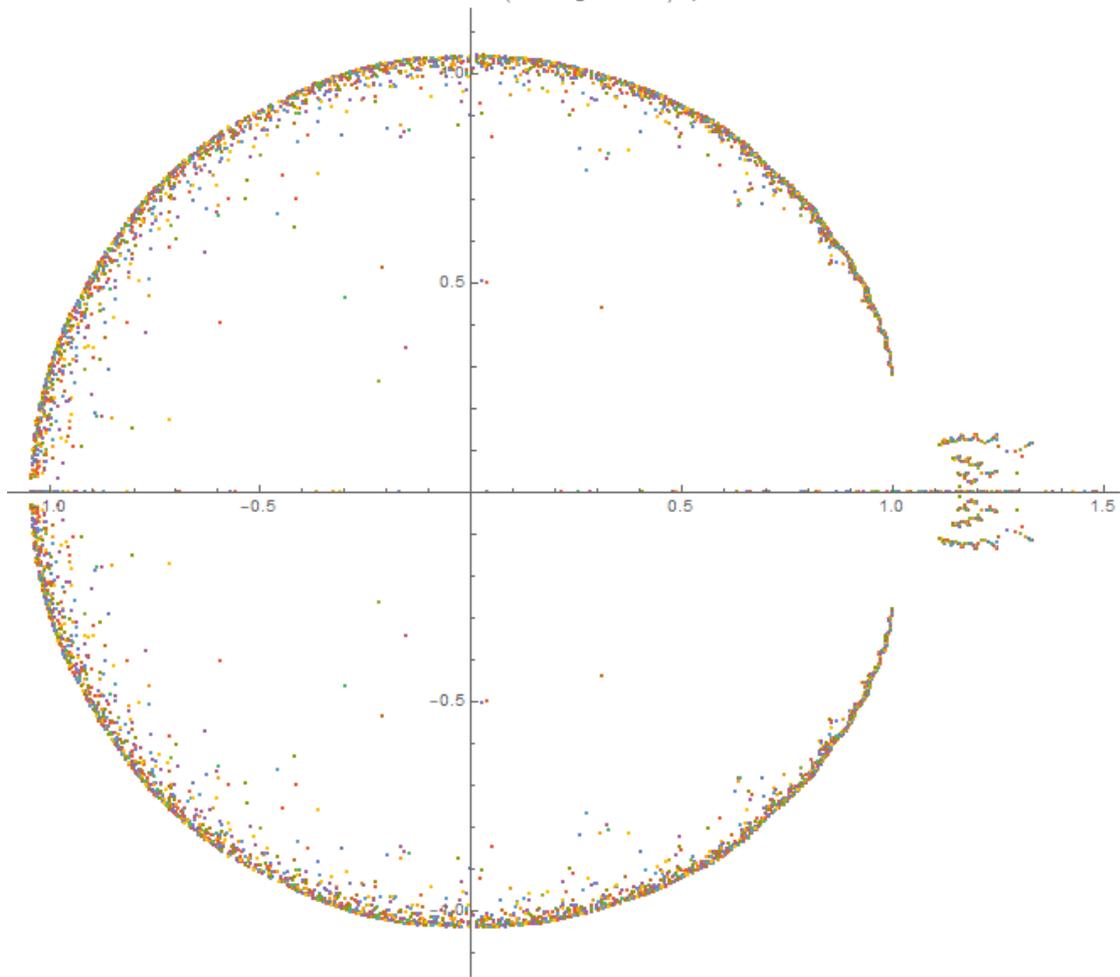
location (cart. coordinates) of the complex roots for qSequence type P1
Dimension: 100



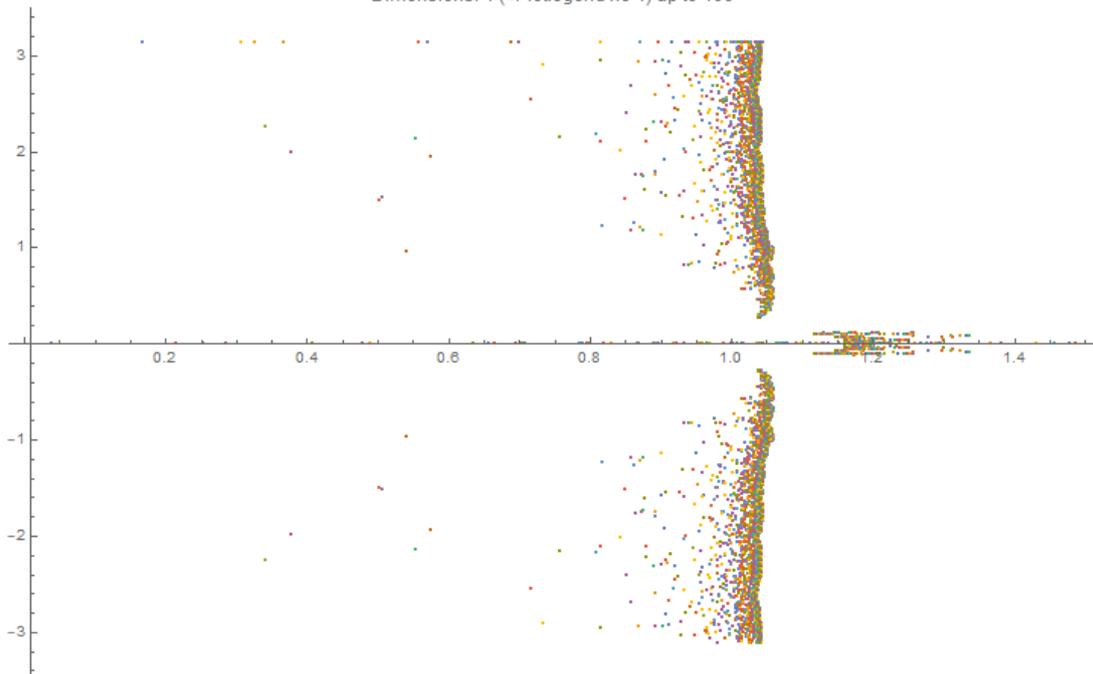
location (polar coordinates) of the complex roots for qSequence type P1
Dimensions: 100



location (cart. coordinates) of the complex roots for qSequence type P1
Dimensions: 1 (=Plotlegend no 1) up to 100



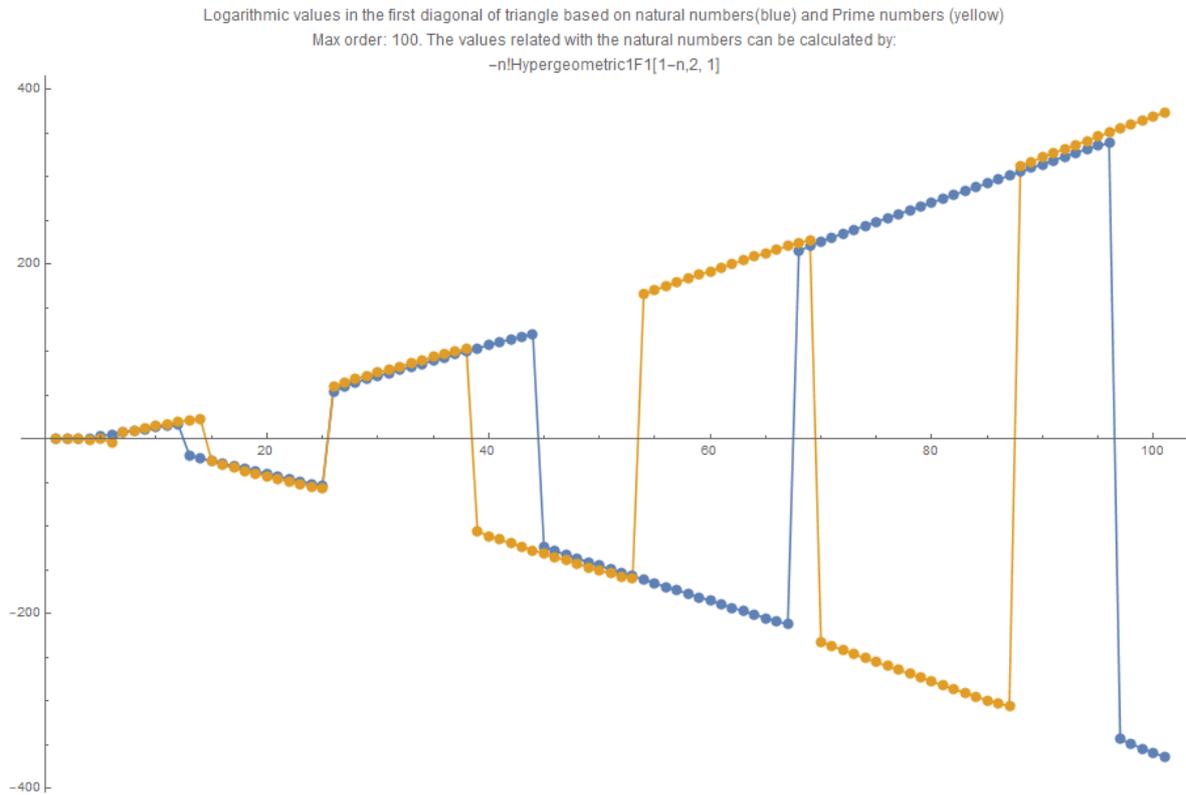
location (polar coordinates) of the complex roots for qSequence type P1
Dimensions: 1 (=Plotlegend no 1) up to 100



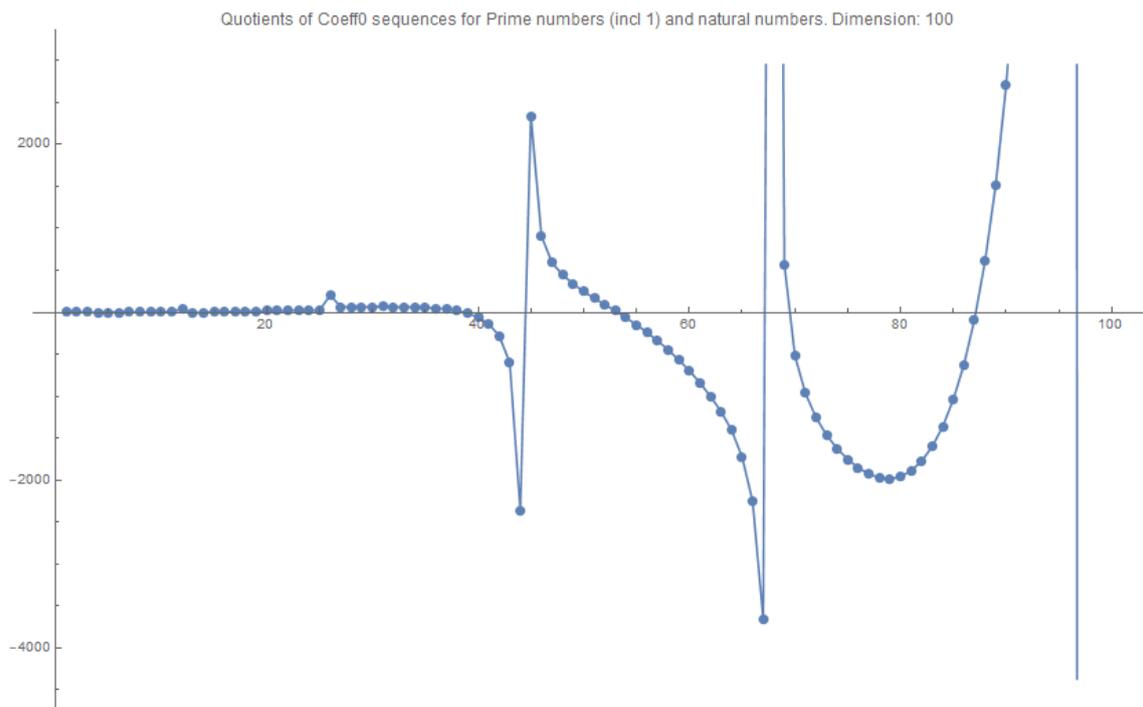
Und noch ein Blick auf die Primfaktorzerlegung der ersten 50 Werte von $Coeff_{0,n}$:

1	{1,1}
2	{1,1}
3	{1,1}
4	{5,1}
5	{1,1}
6	{71,1}
7	{11,1}, {199,1}
8	{11,1}, {1579,1}
9	{5,1}, {54403,1}
10	{31,1}, {149,1}, {509,1}
11	{14621239,1}
12	{281,1}, {1147739,1}
13	{1217307569,1}
14	{5,1}, {10529,1}, {133733,1}
15	{4951,1}, {12278989,1}
16	{1120907,1}, {9248653,1}
17	{31,1}, {167,1}, {64949345833,1}
18	{11,1}, {953,1}, {21961,1}, {36566627,1}
19	{5,1}, {11,1}, {17679667,1}, {155914477,1}
20	{31,1}, {30588403,1}, {4178880553,1}
21	{29,1}, {420803,1}, {565463,1}, {11764271,1}
22	{13537,1}, {110848891214900903,1}
23	{32609,1}, {25427321,1}, {40639606619,1}
24	{5,3}, {5550251869,1}, {698493033631,1}
25	{643,1}, {38261,1}, {248881304175496463,1}
26	{1973,1}, {27904679166759164551363,1}
27	{197,1}, {55897240950785955716993467,1}
28	{8863,1}, {77263,1}, {844586701020730907971,1}
29	{5,1}, {11,1}, {17579,1}, {3407271193,1}, {6979935756062957,1}
30	{11,1}, {137,1}, {9043,1}, {11777,1}, {28364647,1}, {190533140945879,1}
31	{67,1}, {1245191,1}, {1373495499781,1}, {274557304919197,1}
32	{657469,1}, {1506125486212660074232764836981,1}
33	{88681,1}, {303781,1}, {2357807,1}, {9520206263,1}, {59202318419,1}
34	{5,1}, {281,1}, {96849439788919813,1}, {8841427248979156591,1}
35	{199,1}, {911,1}, {1355731193,1}, {193793756917,1}, {812267729749219,1}
36	{61,1}, {1498843,1}, {11802277715851056385913568043367893,1}
37	{11827,1}, {98993,1}, {25598033811869283428758498123282219,1}
38	{61,1}, {13297,1}, {97665349,1}, {9301825277573,1}, {696736579259231941,1}
39	{5,1}, {1483,1}, {1272989,1}, {12514917277,1}, {96886196973294584331021277,1}
40	{11,2}, {41,1}, {2417,1}, {202291,1}, {707907639285813015424527889937058317,1}
41	{11,1}, {31,1}, {196357319,1}, {9563926637,1}, {21193516919,1}, {9879495844107365633,1}
42	{2861,1}, {8905199908952780767189,1}, {317931904100938067261450929,1}
43	{71,1}, {7540793471249,1}, {2220057913322783,1}, {370405541768408417718593,1}
44	{5,1}, {1297,1}, {15299,1}, {55987,1}, {1270445153450608341769,1}, {3383979802038295048681,1}
45	{83,1}, {1009,1}, {8939407,1}, {38074871,1}, {42192534451089560702168725307868912781,1}
46	{34537,1}, {926707,1}, {3696593,1}, {30537990645505529089,1}, {16599250962806839019557,1}
47	{383,1}, {96858953530575753108751,1}, {78406389023851848767152440248791867,1}
48	{31,1}, {277,1}, {2173307,1}, {346102703,1}, {21842292327057143973217122168009604136019833,1}
49	{5,2}, {257,1}, {2339,1}, {123863,1}, {7633829256046840371293,1}, {451394577456034889716827336793,1}
50	{29,1}, {179,1}, {761185113629063949410171680543,1}, {67669210599599788288233929157673,1}

Ein Vergleich der ersten 100 Werte von $\text{Coeff}_{0,n}$ für Primzahlen (mit ,1‘) und natürlichen Zahlen (für natürliche Zahlen in blauer Farbe, für Primzahlen in gelber Farbe):



Interessant ist auch ein (linearer) Plot der Quotienten beider $\text{Coeff}_{0,n}$ Sequenzen:



In diesem Fall (a_n ist die Folge der Mersenne Primzahlexponenten mit der ‚1‘) ist das Bildungsgesetz für **Coeff_{0,n}** (noch) nicht bekannt. Die Folge *seqFunc10[n]* gehört auch zur Gruppe der linear rekursiven Funktionen mit dem Kernel:

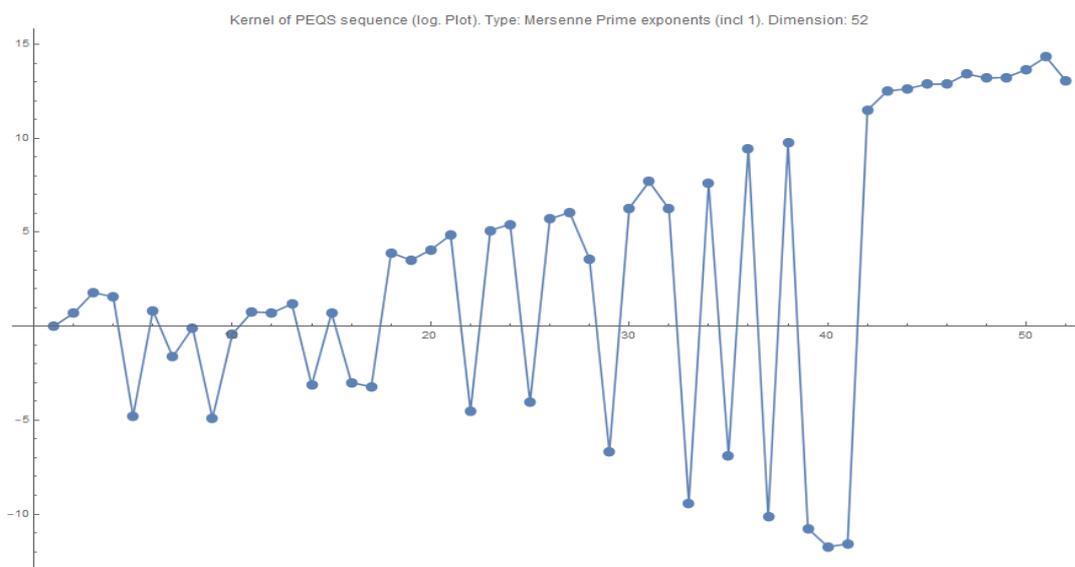
$\{1, \frac{1}{2}, \frac{1}{6}, \frac{5}{24}, -\frac{1}{120}, \frac{311}{720}, -\frac{989}{5040}, -\frac{37049}{40320}, -\frac{547}{72576}, \frac{5548769}{3628800}\}$ und lässt sich auf diese Weise leicht berechnen.

Hinweis: Kehren wir das Konstruktionsprinzip der Dreiecksmatrix aus der ersten Diagonalreihe **Coeff_{0,n}** um, dann lässt sich die Dreiecksmatrix (‚in Richtung nach links oben‘) zu einer quadratischen Matrix erweitern:

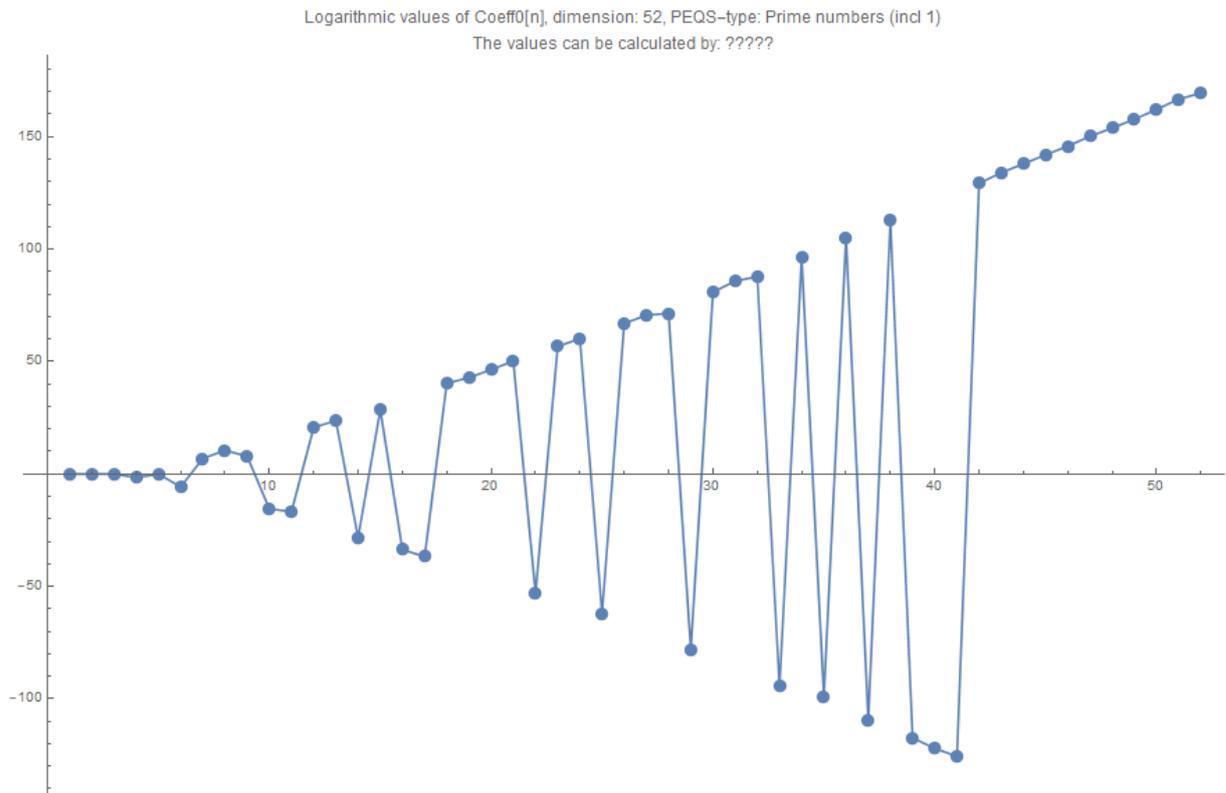
5548769	547	37049	989	311	1	5	1	-1	-1
362880	8064	5040	720	120	24	6	2		
5548769	547	37049	989	311	1	5	1	-1	-1
362880	8064	5040	720	120	24	6	2		
5548769	547	37049	989	311	1	5	1	-1	-1
181440	4032	2520	360	60	12	3		-2	-2
5548769	547	37049	989	311	1	5	1	-1	-1
60480	1344	840	120	20	4		-3	-6	-6
5548769	547	37049	989	311	1	5	1	-1	-1
15120	336	210	30	5		-20	-12	-24	-24
5548769	2735	37049	989	-311	5	-100	-60	-120	-120
3024	336	42	6						
5548769	2735	37049	989	-1866	30	-600	-360	-720	-720
504	56	7							
5548769	2735	37049	989	-13062	210	-4200	-2520	-5040	-5040
72	8		6923	-104496	1680	-33600	-20160	-40320	-40320
5548769	2735	296392	55384	-940464	15120	-302400	-181440	-362880	-362880
9									
-5548769	24615	2667528	498456						

Die rationalen Werte des Kernels ergeben sich außerdem aus den Quotienten des ersten Koeffizienten c_0 (der n-ten Zeile) und c_n (der (n+1)-ten Zeile des (jeweils in roter und blauer Farbe markiert)). Die Kernel-Sequenz lässt sich auch einfach so schreiben: $\frac{Coeff_0[n]}{n!}$

Der Graph der ersten 52 Kernel-Werte (logarithmischer Plot) sieht wie folgt aus:

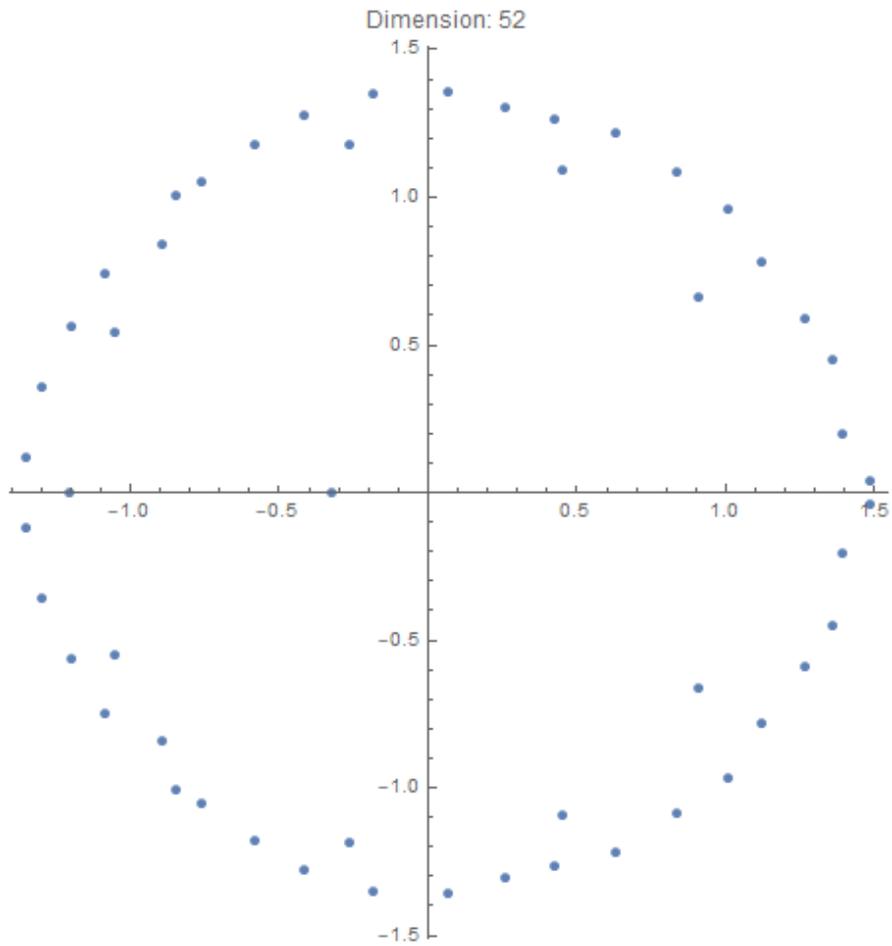


Der Graph der ersten 52 Coeff0-Werte sieht wie folgt aus:

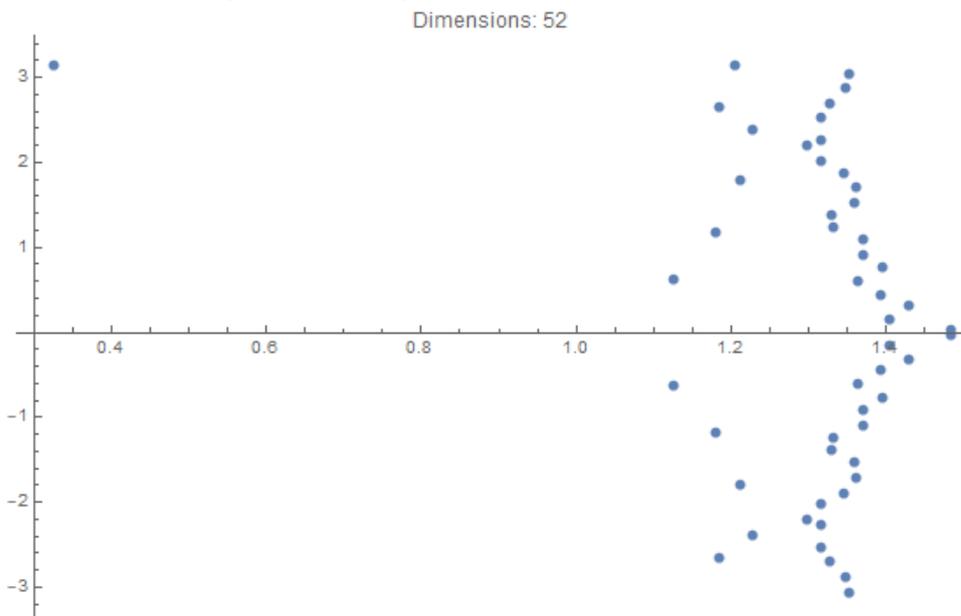


Hier noch ein Beispiel für $n = 52$ mit einer graphischen Darstellung der komplexen Nullstellen auf Basis der Folge der Mersenne-Primzahlexponenten (mit ,1'), in kartesischen und Polarkoordinaten:

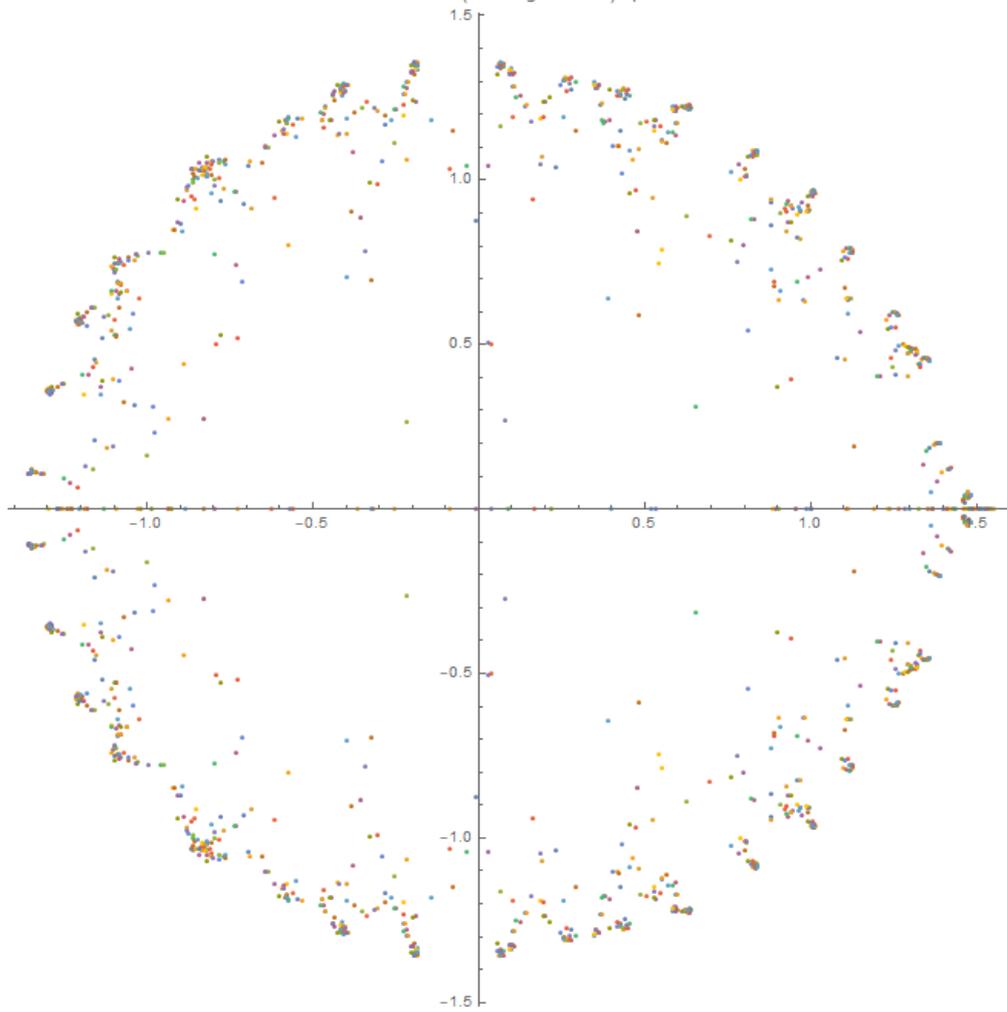
location (cart. coordinates) of the complex roots for qSquence type MP1



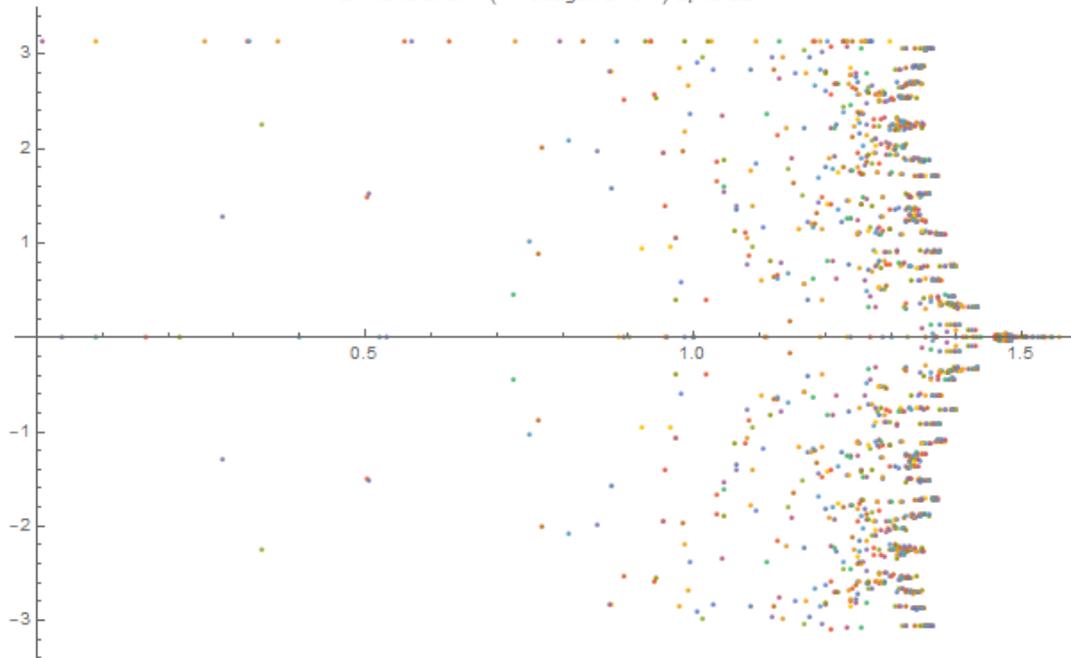
location (polar coordinates) of the complex roots for qSquence type MP1



location (cart. coordinates) of the complex roots for qSequence type MP1
Dimensions: 1 (=Plotlegend no 1) up to 52



location (polar coordinates) of the complex roots for qSequence type MP1
Dimensions: 1 (=Plotlegend no 1) up to 52



Und noch ein Blick auf die Primfaktorzerlegung der ersten 52 Werte von $Coeff_{0,n}$:

1	{{1,1}}
2	{{1,1}}
3	{{1,1}}
4	{{5,1}}
5	{{1,1}}
6	{{311,1}}
7	{{23,1}, {43,1}}
8	{{37049,1}}
9	{{5,1}, {547,1}}
10	{{5548769,1}}
11	{{11,1}, {523,1}, {3217,1}}
12	{{11,1}, {89376809,1}}
13	{{11,1}, {1861618579,1}}
14	{{5,2}, {11,1}, {66643,1}, {108287,1}}
15	{{11,1}, {5347,1}, {45001573,1}}
16	{{11,1}, {40039,1}, {948934699,1}}
17	{{11,1}, {67,1}, {12060363522433,1}}
18	{{11,2}, {6269,1}, {423205940291,1}}
19	{{5,1}, {11,1}, {74496368234700233,1}}
20	{{11,1}, {43,1}, {157,1}, {443353,1}, {4321340407,1}}
21	{{11,1}, {37363,1}, {3679219,1}, {4339045387,1}}
22	{{11,1}, {97,1}, {409,1}, {457,1}, {1174919,1}, {431608979,1}}
23	{{11,1}, {387162452943284411786279,1}}
24	{{5,1}, {11,1}, {602029,1}, {1469856667,1}, {2903134177,1}}
25	{{11,1}, {31,1}, {277025893213,1}, {9325459350353,1}}
26	{{11,1}, {41,1}, {392259521,1}, {696675806810700469,1}}
27	{{11,1}, {71,1}, {366599,1}, {917753,1}, {17610852381065357,1}}
28	{{11,1}, {200680884080753,1}, {4889008376390953,1}}
29	{{5,1}, {11,1}, {89,1}, {109,1}, {130982653,1}, {262839761,1}, {389046428297,1}}
30	{{11,1}, {23,1}, {43,2}, {73,1}, {2113,1}, {18260491,1}, {107859559756304887,1}}
31	{{11,1}, {5233,1}, {15018917,1}, {8563419253,1}, {2451139061220503,1}}
32	{{11,1}, {26648501026127,1}, {461119407414864344463437,1}}
33	{{11,2}, {16231,1}, {5369537564926535646233398224727951,1}}
34	{{5,1}, {11,2}, {199,1}, {433,1}, {17581,1}, {644702356685001291176388405761,1}}
35	{{11,2}, {345679,1}, {241194142549084191766614912829384391,1}}
36	{{11,2}, {307,1}, {1117,1}, {2399933,1}, {48533212210967532327610861507097,1}}
37	{{11,2}, {83572067,1}, {331405349937752477,1}, {105112666469906764609,1}}
38	{{11,2}, {89,1}, {829,1}, {934909,1}, {963559,1}, {467550868191383,1}, {2382098523560753,1}}
39	{{5,2}, {11,2}, {11747086553,1}, {2906957265248501,1}, {9297370524320474620351,1}}
40	{{11,2}, {838056689924826426132378086946264203294163903830879,1}}
41	{{11,2}, {4481,1}, {207227,1}, {3632309,1}, {857742281633,1}, {10290855309903647200026959,1}}
42	{{11,2}, {85339543,1}, {13458836424230417313469439588806651411272508993,1}}
43	{{11,2}, {89,1}, {15427,1}, {360289,1}, {1515763756014093389,1}, {181664976621562017499899233,1}}
44	{{5,1}, {11,2}, {9145825441,1}, {200244966637,1}, {734604976985185949004121092212457979,1}}
45	{{11,2}, {71,1}, {127,1}, {43268260897205836543348060236496366873948076233700243233,1}}
46	{{11,2}, {47,1}, {5189,1}, {7367421007385055446691421574871633912153208585155290003,1}}
47	{{11,2}, {454969,1}, {31359674865749,1}, {32773811900279,1}, {3100477740189438153232836937351,1}}
48	{{11,2}, {16073,1}, {7393777,1}, {461995567073472739,1}, {1020857869580026511759046442059013931,1}}
49	{{5,1}, {11,2}, {991,1}, {3767,1}, {532733,1}, {284994249944021490150615533026913522129266415946753391,1}}
50	{{11,2}, {43,1}, {10574967671,1}, {475433450508228428424421698941393092825827319072330028227,1}}
51	{{11,2}, {59,1}, {773,1}, {148579,1}, {13146241,1}, {47775051807017,1}, {5075219927620547247218093997004981331059,1}}
52	{{11,2}, {107,1}, {127,1}, {354657409,1}, {687691133,1}, {3854014861,1}, {24588314176737086300893947288566779589263,1}}

Auffällig bei der Primfaktorzerlegung von $Coeff_{0,n}$ (basierend auf den Mersenne-Primzahl-Exponenten) ist, dass ab dem Index 11 jeder Wert durch 11 teilbar ist und ab dem Index 33 jeder Wert durch 11^2 teilbar ist. Weder bei der Primfaktorenzerlegung der $Coeff_{0,n}$ bei den natürlichen Zahlen noch bei der Primfaktorenzerlegung der $Coeff_{0,n}$ bei den Primzahlen gibt es eine ähnliche Auffälligkeit in der Primfaktorzerlegung.

Tabelle: Beispiele von Sequenzen $intF_n$ und deren Transformierte $Coeff_{0k}$
 Explizite Formeln (falls bekannt) sind in roter Farbe angegeben.

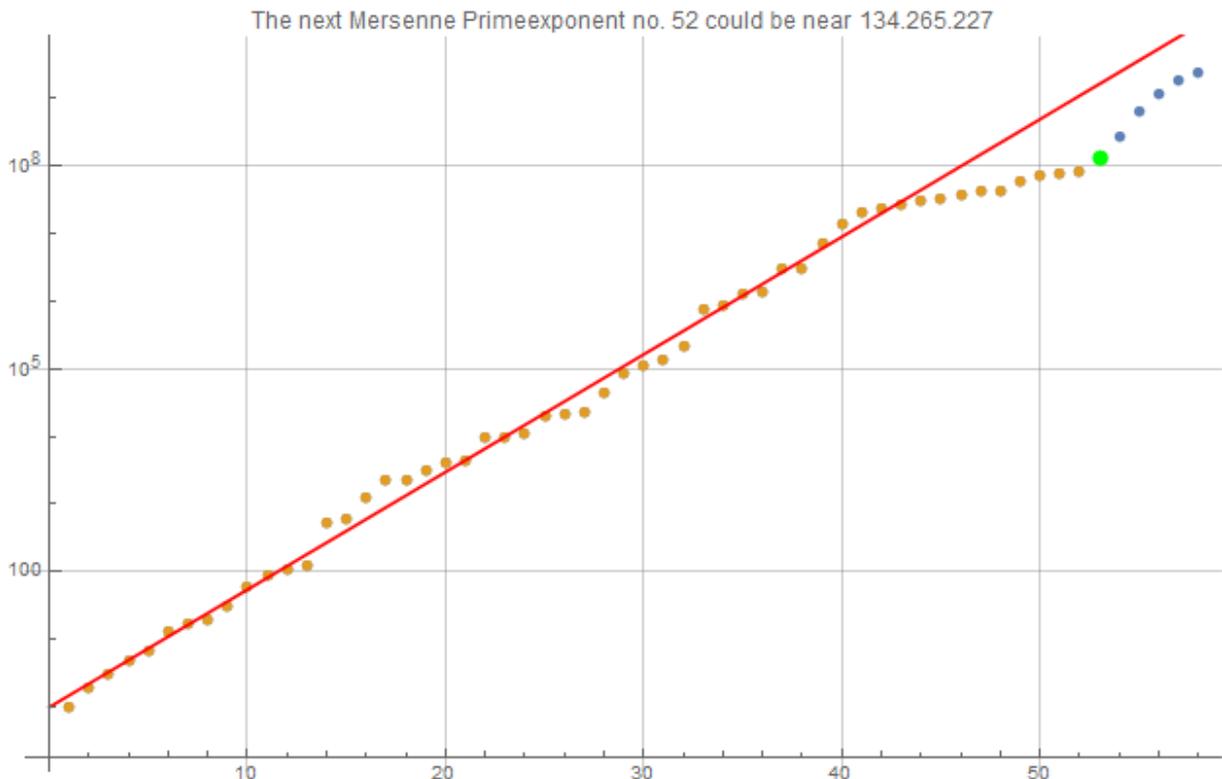
Nr.	$Coeff_{0k}$	$intF_n$
1	$\{-1, -1, -1, 1, 19, 151, 1091, 7841, 56519, 396271, 2442439, 7701409\}$ $-n! \cdot F_1(1-n; 2; 1)$	$\{1, 2, 3, 4, 5, 6, 7, \dots, n\}$ Natürliche Zahlen \mathbb{N}_+ , $a_n = n$
2	$\{-1, 1, 3, 3, -21, -207, -1233, -4869, 5751, 436833, 6908571, 83211219\}$ $-n! \cdot F_1(1-n; 2; 3)$	$\{1, 4, 7, 10, 13, 16, 19, 22, 25, 28, 31, \dots\}$ $a_n = 3n - 2, n = 1, 2, 3, \dots$
3	$\{1, -1, -3, -7, 1, 219, 2581, 22973, 162177, 554039, -10506419, -343049631, -68464\}$ $(-1)^{n+1} n! \cdot {}_2F_2(1-n, n+1; \frac{3}{2}, \frac{1}{4})$	$\{1, 4, 9, 16, 25, 36, 49, 64, 81, 100, 121, 144, \dots\}$ $a_n = n^2, n = 1, 2, 3, \dots$
4	$\{1, 2, 3, 4, 5, 6, 7, \dots, n\}$ Natürliche Zahlen \mathbb{N}_+	$\{-2, 1, -1, 1, -1, 1, -1, 1, -1, 1, -1, 1, \dots\}$ $a_1 = -2, a_n = (-1)^n, n > 1$
5	$\{1, 4, 7, 10, 13, 16, 19, 22, 25, 28, 31\}$ $3n - 2$	$\{-4, 9, -27, 81, -243, 729, -2187, 6561, -19683, 59049, \dots\}$ $a_1 = -4, a_n = (-3)^n, n > 1$
6	$\{1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, \dots\}$ $Fibonacci(n)$	$\{-1, -1, \frac{1}{2}, \frac{1}{6}, -\frac{7}{24}, \frac{1}{24}, \frac{17}{144}, -\frac{67}{1008}, \dots\} =$ $\{-1, -1, 1, 1, -7, 5, 85, -335, -1135, \dots\} / n!$ a_n berechenbar aus Reihenentwicklung von $Ln \left(\sum_{n=1}^{\infty} Fibonacci(n) \frac{x^{n-1}}{(n-1)!} \right)$
7	$\{1, 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, \dots\}$ $a_1 = 1, a_n = Prime(n-1), n > 1$	$\{-2, 1, -\frac{3}{2}, 2, -\frac{59}{24}, \frac{59}{20}, -\frac{169}{48}, \frac{2119}{504}, -\frac{67577}{13440}, \dots\} =$ $\{-2, 1, -3, 12, -59, 354, -2535, 21190, -202731, \dots\} / n!$ a_n berechenbar aus Reihenentwicklung von $Ln \left(\sum_{n=1}^{\infty} Prime1(n) \frac{x^{n-1}}{(n-1)!} \right)$
8	$\{1, 0, -1, -2, -9, -52, -395, -3618, -39151, -486872, -6848361, \dots\}$	$a_n = Fibonacci(n); n = 0, 1, 2, 3, \dots$
9	$\{1, -1, 0, -2, -4, -36, -224, -2200, -22608, -281456, -3879424, \dots\}$	$a_n = Fibonacci(n); n = 1, 2, 3, 4, \dots$
10	$\{-1, -1, -1, -5, 1, -71, 2189, 17369, 272015, 2351071, 14621239, 322514659, \dots\}$	$\{1, 2, 3, 5, 7, 11, 13, \dots, n\}$, Primzahlen (mit '1') $\mathbb{P}_1, a_1 = 1, a_n = Prime(n-1), n > 1$
11	$\{-1, -1, -1, -5, 1, -311, 989, 37049, 2735, -5548769, -18507401, 983144899, \dots\}$	$\{1, 2, 3, 5, 7, 11, 13, \dots, n\}$, Mersenne Primzahlexponenten (mit '1'), $a_1 = 1, a_n =$ $MersennePrimeExponent(n-1), n > 1$

TABLE 1: BEISPIELE VON PEQS-TRANSFORMATIONS-PAAREN

8.9.4_WO LIEGT DIE NÄCHSTE MERSENNE-PRIMZAHL?

Es wird vermutet, dass die Werte $MPExp[n]$ der Mersenne-Primzahlexponenten im statistischen Mittel der Funktion $2^{\gamma n}$ folgen und somit in einer logarithmischen Darstellung durch eine Gerade approximiert werden kann (im Plot mit roter Farbe).

Dabei ist γ die Euler-Mascheroni Konstante. Benutzen wir die oben beschriebene Methode zur Bestimmung einer Sequenzfunktion, die mit den ersten 52 Werten der Mersenne-Primzahl-Exponenten (inclusive der ,1‘) übereinstimmt (gelb markiert) und extrapolieren wir mit dieser Sequenzfunktion noch ein paar Indizes weiter (blau markiert), dann bekommen wir folgende Darstellung:



Natürlich gibt es überhaupt keinen (mathematischen) Grund anzunehmen, dass die nächste Mersenne-Primzahl wirklich bei $2^{134265227} - 1$ liegt (mit grüner Farbe markiert). Der (nach der Methode im Kapitel 8.9.3) extrapolierte Funktionsverlauf war jedoch zu schön und der Autor konnte nicht widerstehen, die ‚extrapolierten‘ Mersenne-Primzahlexponenten 52 bis 57 zu plotten!

Wenn man den Graph betrachtet wird man das Gefühl nicht los, dass die nächsten Mersenne-Primzahl-Exponenten wieder zu höheren Werten hin drängen, da die letzten 8 Werte deutlich unter dem vorhergesagten Mittelwert liegen!

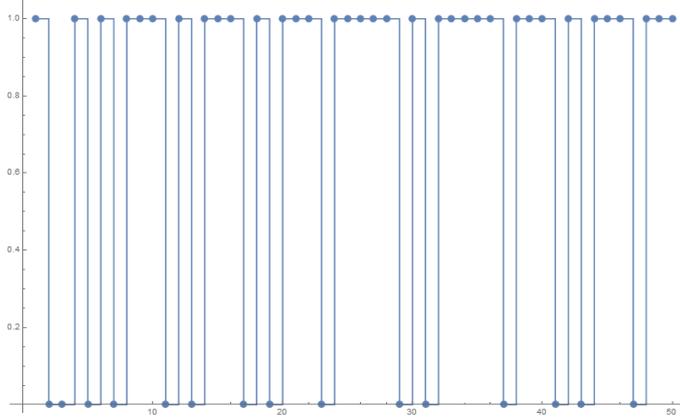
11.2_DIE FUNKTIONEN NOTPRIMEQ UND SIGNEDPRIMEQ

Die Funktion `notPrimeQ(n)` ergibt das logische Gegenteil der Funktion `PrimeQ(n)`.

Sie ergibt 1 wenn n keine Primzahl ist und 0 wenn n eine Primzahl ist.

Mathematica:

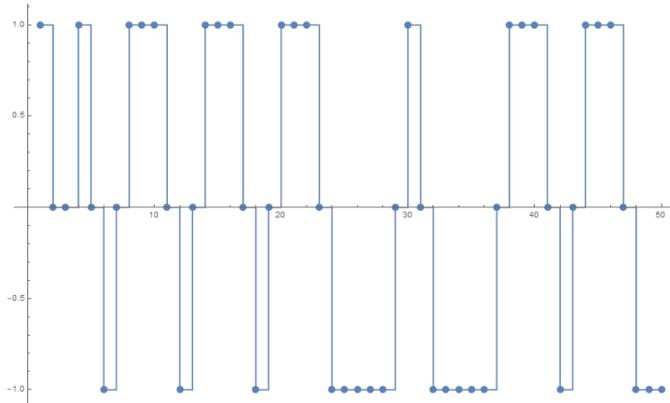
```
notPrimeQ[n_]:=1-Boole[PrimeQ[n]]
range=50;tabNotPrimeQ=Table[notPrimeQ[n],{n,1,range}]
Show[ListLinePlot[tabNotPrimeQ,InterpolationOrder\{Rule\}0],ListPlot[tabNotPrimeQ]]
(* zeros at prime positions:
{1,0,0,1,0,1,0,1,1,1,0,1,0,1,1,1,0,1,0,1,1,1,0,1,1,1,1,0,1,0,1,
1,1,1,1,0,1,1,1,0,1,0,1,1,1,0,1,1,1}
*)
```



Die Funktion `signedPrimeQ(n)` ergibt 0 wenn n eine Primzahl ist, sowie alternierende Werte von 1 und -1 wenn n keine Primzahl ist.

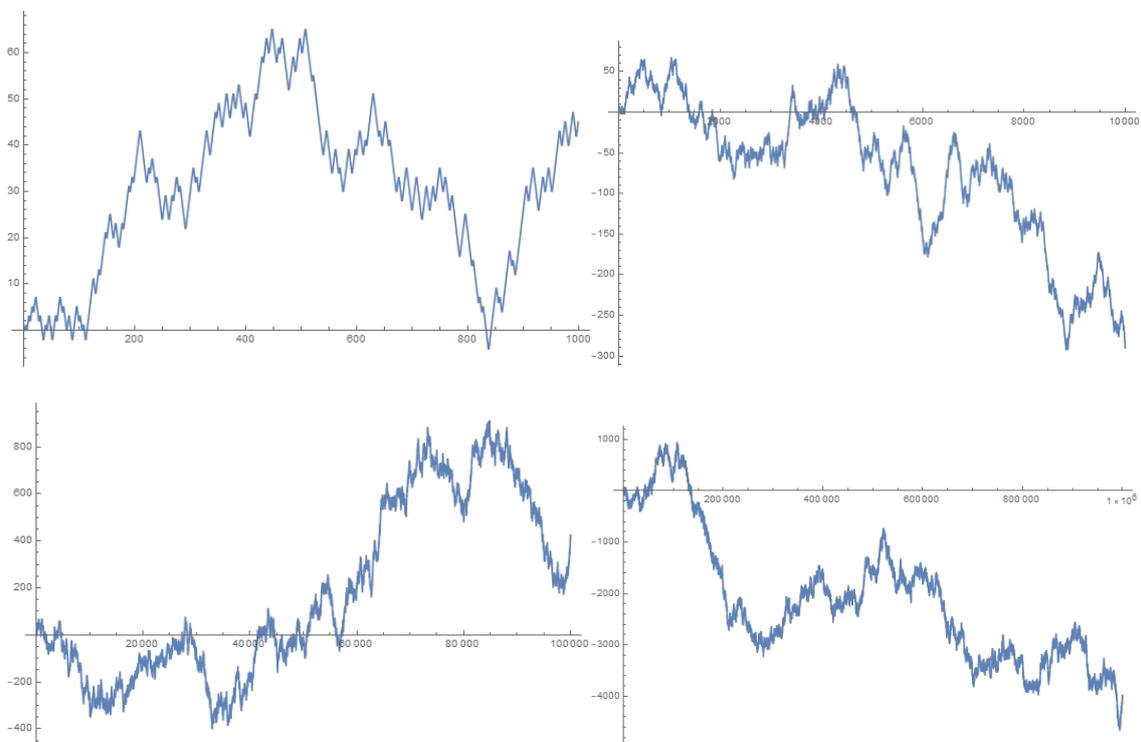
Mathematica:

```
signedPrimeQ[n_]:=If[PrimeQ[n]==True,0,If[EvenQ[PrimePi[n]]==True,1,-1]]
(*range=50;tabSignedPrimeQ=Table[signedPrimeQ[n],{n,2,range}]*)
(*Show[ListLinePlot[tabSignedPrimeQ,InterpolationOrder\{Rule\}0],ListPlot[tabSignedPrimeQ]]*)
(* Zeros at prime positions:*)
{1,0,0,1,0,-1,0,1,1,1,0,-1,0,1,1,1,0,-1,0,1,1,1,0,-1,-1,-1,-1,-1,0,1,0,-1,-1,-1,-1,-1,0,1,1,1,0,-1,0,1,1,1,0,-1,-1,-1}
```



Schauen wir uns nun die Summenfunktion von `signedPrimeQ[]` an.

Die Summenfunktion von `signedPrimeQ[]` ist ein selbstähnliches Fraktal auf einer logarithmischen Skala. Sie hat unabhängig von dem betrachteten Zahlenbereich (z. B. : 1000, 10000, 100000 or 1 Mill.) immer das gleiche statistische Verhalten. Sie scheint auch in höheren Zahlenbereichen ein ausgeglichenes Verhältnis von positiven zu negativen Werten zu haben.

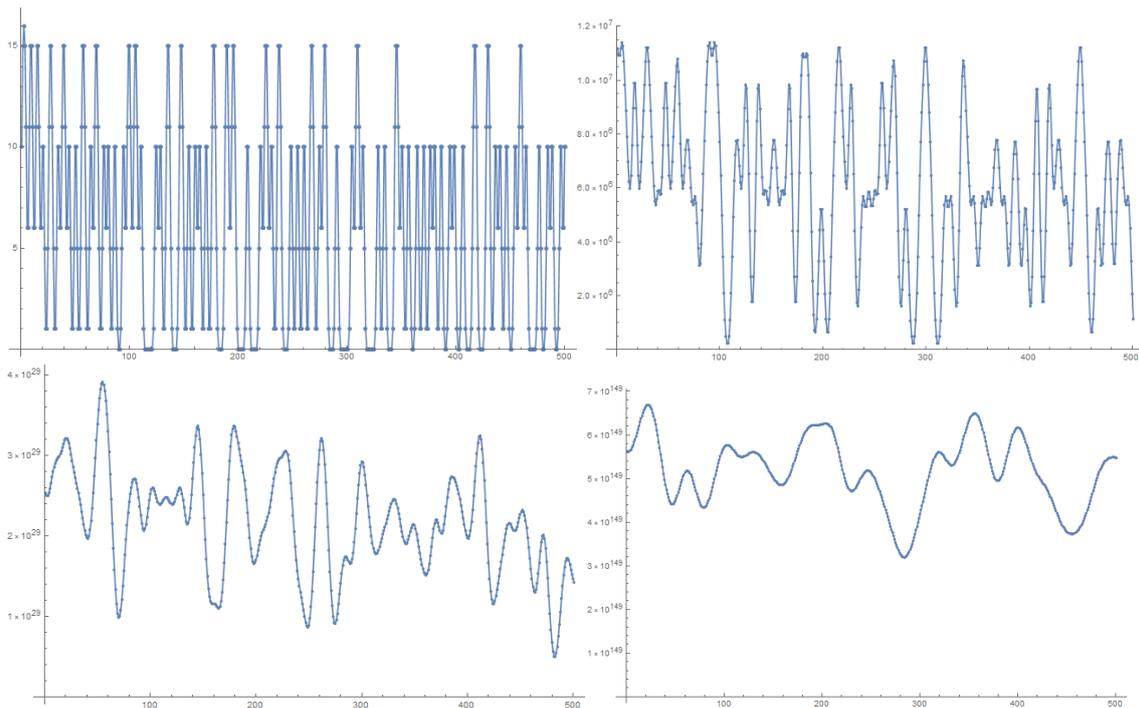


Mathematica:

```
range=1000;  
tabSignedPrimeQ=Table[signedPrimeQ[n], {n, 1, range}];  
sumtabSignedPrimeQ=Accumulate[tabSignedPrimeQ];  
ListLinePlot[sumtabSignedPrimeQ, ImageSize->Large]
```

Die Funktion `notPrimeQ[]` ist nicht weniger interessant, wenn wir anstelle der Summationsfunktion ihre Differenz-Funktionen betrachten. Die Differenzfunktionen von `notPrimeQ[]` scheinen jedoch zunächst für kleine Ordnungen nicht sehr interessant zu sein (bis zur 10. Ordnung). Höhere Differenzenordnungen zeigen jedoch ein unerwartetes, glattes Erscheinungsbild:

Die folgenden Diagramme zeigen das Verhalten für die Differenzenordnungen 5, 25, 100 und 500.



Der Autor hat bisher keine Idee, was der Grund für diesen Effekt ist.
 Es muss geklärt werden, ob dieses Verhalten auf Rundungsfehler oder unzureichende Rechengenauigkeit zurückzuführen ist...

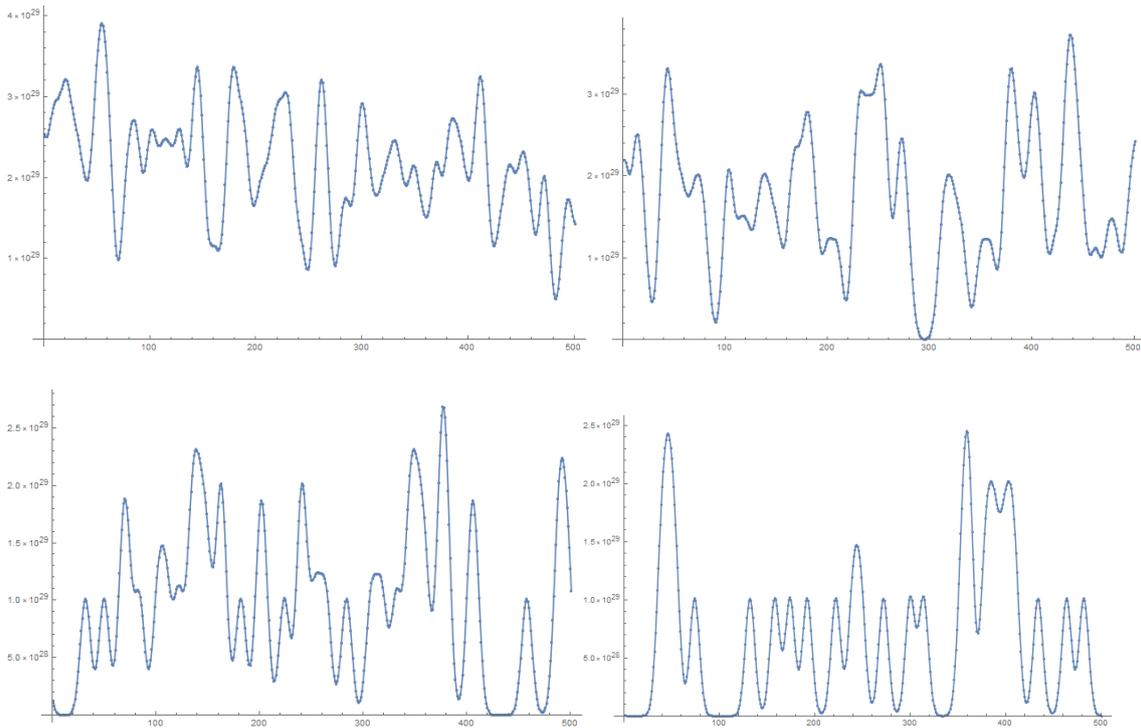
Mathematica:

```
range=1; order=5;
tabNotPrimeQ=Table[notPrimeQ[n], {n, range, range+500+order}];
diftabNotPrimeQ=Abs[Differences[tabNotPrimeQ, order]];
Show[ListLinePlot[diftabNotPrimeQ, ImageSize->Large], ListPlot[diftabNotPrimeQ, ImageSize->Large]]
```

Wenn Sie in einen höheren Zahlenbereich gehen, verändern sich die spektralen Eigenschaften nicht, mit der Ausnahme, dass der gesamte Graph tendenziell häufiger relativ kleinere Werte aufweist.

Die nachfolgenden Darstellungen gehören zu Differenzen der Ordnungen 100 und den Bereichen 1, 1000, 1 Mill., 1000 Mill.

Im Fall von 1000 Mill liegen die minimalen Werte um 10^{21} , die maximalen Werte um $2.5 \cdot 10^{29}$. Dies umfasst einen Bereich von 10^8 . Das ist überraschend...



18.4_KÖRPER UND ERWEITERTE RECHENOPERATOREN (KO-AUTOR: ADIL BULUS)

18.4.1_VERSCHACHTELUNG VON EXPONENTIAL- UND LOGARITHMUSFUNKTIONEN

Unter Rechenoperatoren verstehen wir die Operatoren „+“ (Addition), „*“ (Multiplikation), „^“ (Potenzierung) und deren mögliche Verallgemeinerungen bzw. Fortsetzungen.

Die folgende Methode erlaubt eine sehr allgemeine Definition von Rechenoperatoren. Das Prinzip ist eine Verschachtelung und Kombination von Exponential- und Logarithmus-Funktionen unter Verwendung eines ‚Verschachtelungsoperators‘, der ebenfalls „+“, „*“ oder „^“ sein kann. Formuliert mit der Mathematica-Programmiersprache lässt sich jeweils eine ganze Matrix von Rechenoperatoren nur mit einer einzigen Programmzeile erzeugen. Die sich ergebenden Operatoren werden eventuell erst ‚sichtbar‘, nachdem sie mit den für Exponential- und Logarithmusfunktionen geltenden Rechengesetzen vereinfacht wurden. Beispiele für solche verschachtelte Konstruktionen sind zum Beispiel:

Ohne Vereinfachung:

$$\left(\begin{array}{cccc} a + b & e^{a+\log(b)} & e^{e^{a+\log(\log(b))}} & e^{e^{e^{a+\log(\log(\log(b)))}}} \\ e^{\log(a)+b} & e^{\log(a)+\log(b)} & e^{e^{\log(a)+\log(\log(b))}} & e^{e^{e^{\log(a)+\log(\log(\log(b)))}}} \\ e^{e^{\log(\log(a))+b}} & e^{e^{\log(\log(a))+\log(b)}} & e^{e^{\log(\log(a))+\log(\log(b))}} & e^{e^{e^{\log(\log(a))+\log(\log(\log(b)))}}} \\ e^{e^{e^{\log(\log(\log(a))+b}}} & e^{e^{e^{\log(\log(\log(a))+\log(b)}}} & e^{e^{e^{\log(\log(\log(a))+\log(\log(b))}}} & e^{e^{e^{e^{\log(\log(\log(a))+\log(\log(\log(b)))}}} \end{array} \right)$$

Nach Vereinfachung:

$$\begin{pmatrix} a + b & e^a b & b^{e^a} & e^{\log^{e^a}(b)} \\ ae^b & ab & b^a & e^{\log^a(b)} \\ a^{e^b} & a^b & a^{\log(b)} & e^{a \log(\log(b))} \\ e^{\log^{e^b}(a)} & e^{\log^b(a)} & e^{b \log(\log(a))} & e^{\log \log(\log(b))(a)} \end{pmatrix}$$

Ohne Vereinfachung:

$$\begin{pmatrix} ab & \log(ae^b) & \log(\log(ae^{e^b})) & \log(\log(\log(ae^{e^{e^b}}))) \\ \log(e^a b) & \log(e^{a+b}) & \log(\log(e^{a+e^b})) & \log(\log(\log(e^{a+e^{e^b}}))) \\ \log(\log(e^{e^a} b)) & \log(\log(e^{e^a+b})) & \log(\log(e^{e^a+e^b})) & \log(\log(\log(e^{e^a+e^{e^b}}))) \\ \log(\log(\log(e^{e^{e^a}} b))) & \log(\log(\log(e^{e^{e^a+b}}))) & \log(\log(\log(e^{e^{e^a+e^b}}))) & \log(\log(\log(e^{e^{e^a+e^{e^b}}})) \end{pmatrix}$$

Nach Vereinfachung:

$$\begin{pmatrix} ab & \log(a) + b & \log(\log(a) + e^b) & \log(\log(\log(a) + e^{e^b})) \\ a + \log(b) & a + b & \log(a + e^b) & \log(\log(a + e^{e^b})) \\ \log(e^a + \log(b)) & \log(e^a + b) & \log(e^a + e^b) & \log(\log(e^a + e^{e^b})) \\ \log(\log(e^{e^a} + \log(b))) & \log(\log(e^{e^a} + b)) & \log(\log(e^{e^a} + e^b)) & \log(\log(e^{e^a} + e^{e^b})) \end{pmatrix}$$

Ohne Vereinfachung:

$$\begin{pmatrix} \log((e^a)^b) & \log((e^a)^{e^b}) & \log(\log((e^a)^{e^{e^b}})) & \log(\log(\log((e^a)^{e^{e^{e^b}}})) \\ \log(\log((e^{e^a})^b)) & \log(\log((e^{e^a})^{e^b})) & \log(\log((e^{e^a})^{e^{e^b}})) & \log(\log(\log((e^{e^a})^{e^{e^{e^b}}})) \\ \log(\log(\log((e^{e^{e^a}})^b))) & \log(\log(\log((e^{e^{e^a}})^{e^b}))) & \log(\log(\log((e^{e^{e^a}})^{e^{e^b}}))) & \log(\log(\log((e^{e^{e^a}})^{e^{e^{e^b}}})) \\ \log(\log(\log(\log((e^{e^{e^{e^a}}})^b)))) & \log(\log(\log(\log((e^{e^{e^{e^a}}})^{e^b})))) & \log(\log(\log(\log((e^{e^{e^{e^a}}})^{e^{e^b}})))) & \log(\log(\log(\log((e^{e^{e^{e^a}}})^{e^{e^{e^b}}})) \end{pmatrix}$$

Nach Vereinfachung:

$$\begin{pmatrix} ab & ae^b & \log(a) + e^b & \log(\log(a) + e^{e^b}) \\ a + \log(b) & a + b & a + e^b & \log(a + e^{e^b}) \\ \log(e^a + \log(b)) & \log(e^a + b) & \log(e^a + e^b) & \log(e^a + e^{e^b}) \\ \log(\log(e^{e^a} + \log(b))) & \log(\log(e^{e^a} + b)) & \log(\log(e^{e^a} + e^b)) & \log(\log(e^{e^a} + e^{e^b})) \end{pmatrix}$$

Hier noch ein paar mit Mathematica erzeugte Beispiele mit $dim = 2$ (der Wert von dim läßt sich beliebig erhöhen).

Beispiele vom Typ ‚Exp-Log-Log‘ (mit dem entsprechenden Mathematica-Code):

OpMatrix1 (kommutative Operatoren auf der Diagonale, Verschachtelungsoperator:

„+“):

`Table[Nest[Exp,Nest[Log,a,i]+Nest[Log,b,j],Max[i,j]],{i,0,dim},{j,0,dim}]/MatrixForm`

$$\begin{pmatrix} a + b & e^a b & b^{e^a} \\ ae^b & ab & b^a \\ a^{e^b} & a^b & a^{\log(b)} \end{pmatrix}$$

OpMatrix2 (kommutative Operatoren auf der Diagonale, Verschachtelungsoperator:

„*“):

`Table[Nest[Exp,Nest[Log,a,i]*Nest[Log,b,j],Max[i,j]],{i,0,dim},{j,0,dim}]/MatrixForm`

$$\begin{pmatrix} ab & b^a & e^{\log^a(b)} \\ a^b & a^{\log(b)} & e^{a^{\log(\log(b))}} \\ e^{\log^b(a)} & e^{b^{\log(\log(a))}} & e^{\log^{\log(\log(b))}(a)} \end{pmatrix}$$

OpMatrix3 (kommutative Operatoren auf der Diagonale, Verschachtelungsoperator:

„^“):

Table[Nest[Exp,Nest[Log,a,i]^Nest[Log,b,j+1],Max[i,j]],{i,0,dim},{j,0,dim}]/MatrixForm

$$\begin{pmatrix} a^{\log(b)} & e^{a^{\log(\log(b))}} & e^{e^{a^{\log(\log(\log(b)))}}} \\ e^{\log^{\log(b)}(a)} & e^{\log^{\log(\log(b))}(a)} & e^{e^{\log^{\log(\log(\log(b)))}(a)}} \\ e^{e^{\log^{\log(b)}(\log(a))}} & e^{e^{\log^{\log(\log(b))}(\log(a))}} & e^{e^{\log^{\log(\log(\log(b)))}(\log(a))}} \end{pmatrix}$$

OpMatrix4 (kein kommutativer Operator auf der Diagonale, Verschachtelungsoperator:

„^“):

Table[Nest[Exp,Nest[Log,a,i]^Nest[Log,b,j],Max[i,j]],{i,0,dim},{j,0,dim}]/MatrixForm

$$\begin{pmatrix} a^b & e^{a^{\log(b)}} & e^{e^{a^{\log(\log(b))}}} \\ e^{\log^b(a)} & e^{\log^{\log(b)}(a)} & e^{e^{\log^{\log(\log(b))}(a)}} \\ e^{e^{\log^b(\log(a))}} & e^{e^{\log^{\log(b)}(\log(a))}} & e^{e^{\log^{\log(\log(b))}(\log(a))}} \end{pmatrix}$$

Beispiele vom Typ ‚Log-Exp-Exp‘:

OpMatrix5 (kommutative Operatoren auf der Diagonale, Verschachtelungsoperator:

„+“):

Table[Nest[Log,Nest[Exp,a,i]+Nest[Exp,b,j],Max[i,j]],{i,0,dim},{j,0,dim}]/MatrixForm

$$\begin{pmatrix} a+b & \log(a+e^b) & \log(\log(a+e^{e^b})) \\ \log(e^a+b) & \log(e^a+e^b) & \log(\log(e^a+e^{e^b})) \\ \log(\log(e^{e^a}+b)) & \log(\log(e^{e^a}+e^b)) & \log(\log(e^{e^a}+e^{e^b})) \end{pmatrix}$$

OpMatrix6 (kommutative Operatoren auf der Diagonale, Verschachtelungsoperator:

„*“):

Table[Nest[Log,Nest[Exp,a,i]*Nest[Exp,b,j],Max[i,j]],{i,0,dim},{j,0,dim}]/MatrixForm

$$\begin{pmatrix} ab & \log(a)+b & \log(\log(a)+e^b) \\ a+\log(b) & a+b & \log(a+e^b) \\ \log(e^a+\log(b)) & \log(e^a+b) & \log(e^a+e^b) \end{pmatrix}$$

OpMatrix7 (kommutative Operatoren auf der Diagonale, Verschachtelungsoperator:

„^“):

Table[Nest[Log,Nest[Exp,a,i+1]^Nest[Exp,b,j],Max[i+1,j]],{i,0,dim},{j,0,dim}]/MatrixForm

$$\begin{pmatrix} ab & ae^b & \log(a)+e^b \\ a+\log(b) & a+b & a+e^b \\ \log(e^a+\log(b)) & \log(e^a+b) & \log(e^a+e^b) \end{pmatrix}$$

OpMatrix8 (kein kommutativer Operator auf der Diagonale, Verschachtelungsoperator: „^“):

```
Table[Nest[Exp,Nest[Log,a,i]^Nest[Log,b,j],Max[i,j]],{i,0,dim},{j,0,dim}]/MatrixForm
```

$$\begin{pmatrix} a^b & e^b \log(a) & \log(\log(a)) + e^b \\ ab & ae^b & \log(a) + e^b \\ a + \log(b) & a + b & a + e^b \end{pmatrix}$$

Von den auf diese Weise generierten Operatoren sind einige gut bekannt: „+“, „*“ und „^“ (Potenzierung).

Weniger bekannt sind die Jacobi-Addition ($\log(e^a + e^b)$) sowie die logarithmische Potenzierung ($a^{\log(b)}$), die beide die Eigenschaften der Kommutativität und Assoziativität besitzen. Neutrale Elemente und inverse Elemente existieren ebenfalls.

Folgende Eigenschaften sind direkt abzulesen bzw. leicht zu überprüfen:

- 1) Bei allen Operator-Matrizen (mit Ausnahme der Beispiele 4 und 8) liegen kommutative Operatoren auf der Diagonale (markiert in roter Farbe)
- 2) Die Operatoren abseits der Diagonalen sind nicht kommutativ.
- 3) Die Operatoren auf der Diagonale gehen beim Typ ‚Exp-Log-Log‘ in Richtung ‚höhere Rechenarten‘, beim Typ ‚Log-Exp-Exp‘ in Richtung ‚niedrigere Rechenarten‘
- 4) ‚benachbarte‘ Operatoren auf den Diagonalen bilden einen Körper bezüglich der Menge der komplexen Zahlen (Ausnahme: Beispiele 4 und 8)
- 5) Jeweils zwei Operatoren, die an der gleichen Diagonalposition zweier Operatormatrizen sitzen, bilden ebenfalls einen Körper bezüglich der Menge der komplexen Zahlen für folgende Operator-Matrizen-Paare:
 (OpMatrix1,OpMatrix2), (Verschachtelungsoperatoren „+“ und „*“)
 (OpMatrix2,OpMatrix3), (Verschachtelungsoperatoren „*“ und „^“)
 (OpMatrix5,OpMatrix6), (Verschachtelungsoperatoren „+“ und „*“)

Beispiele für Körper bezüglich der Menge der komplexen Zahlen sind die Operator-Paare $\{a + b \text{ und } a * b\}$, $\{\log(e^a + e^b) \text{ und } (a + b)\}$ sowie $\{a * b \text{ und } a^{\log(b)}\}$.

Wir definieren wir den ‚Jacobi-Additionsoperator‘ \oplus mit Hilfe von Mathematica in leicht veränderter Form wie folgt (wir ersetzen e^x durch 2^x und den natürlichen Logarithmus durch den ‚Logarithmus dualis‘):

```
ld[x_]:=Log[2,x];exp[x_]:=2^x;
CirclePlus[a_,b_]:=Evaluate[Nest[ld,Nest[exp,a,1]+Nest[exp,b,1],1]//PowerExpand
//Simplify
```

Dieser Operator (in expliziter Form: $\text{ld}(2^a + 2^b)$) besitzt dann außer den oben schon erwähnten Körpereigenschaften wie z. B. Assoziativität und Kommutativität folgende Eigenschaften:

- 1) $a \oplus a = a + 1$ (a mit sich selbst ‚Jacobi-addiert‘ erzeugt den Nachfolger von a)
- 2) Wendet man den Operator rekursiv n mal auf a an, so ergibt sich $a + n$:
 $a1 = a \oplus a$ ergibt $a + 1$

$a1 = a1 \oplus a1$ ergibt $a + 2$

$a1 = a1 \oplus a1$ ergibt $a + 3$

...usw.

Folglich ist der ‚Jacobi-Additions-Operator‘ tatsächlich der ‚Vorgänger-Operator‘ der normalen Addition in der ‚Hierarchie‘ der Rechenarten!

Hinweis: In der Literatur wird diese Hierarchie der Rechenarten durch sogenannte Hyperoperatoren beschrieben. Diese Methode benutzt einen völlig anderen Ansatz. Ein Resultat dieses Ansatzes sind die Hyperoperatoren Tetration und Pentation, die durch den hier beschriebenen Ansatz nicht konstruiert werden können.

Man kann noch weiter gehen und als Verschachtelungsoperator die Jacobi-Addition $\log(e^a + e^b)$ bzw. die logarithmische Potenzierung $a^{\log(b)}$ verwenden. Dies bringt jedoch keine neuen Erkenntnisse.

18.4.2_ERWEITERUNG DER OPERATOREN: UNENDLICH VIELE RECHENARTEN

Gibt es noch andere Rechenarten außer Addition, Multiplikation und Potenzierung? Diese Frage wurde zumindest teilweise im letzten Kapitel beantwortet.

Eine noch weiter gehende Antwort liefern die folgenden Ideen des Physikers **Adil Bulus** und beschreiben einige Körper-Konstruktionen, die einen ‚nahtlosen‘ Übergang zwischen verschiedenen Rechenarten ermöglichen.

Die Originalarbeiten von Adil Bulus können auf der Internetseite <https://www.mathematische-notizen.de> nachgelesen werden.

Die hier vorgestellten Methoden ermöglichen unter anderem eine Parametrisierung zwischen:

- Jacobi-Addition und Addition
- Addition und Multiplikation
- Multiplikation und logarithmischer Potenzierung

Durch die Einführung eines Parameters n ergibt sich eine unendliche Anzahl von Körpern, die sozusagen ‚nahtlos‘ in einander übergehen.

Hinweis: Viele der nachfolgenden Aussagen gelten nur, wenn n positiv ganzzahlig und ungerade ist, wenn auch viele Aussagen weiterhin für rationale oder sogar reelle Werte von n gelten. Wir verzichten deshalb auf eine Fallunterscheidung und nehmen im Folgenden der Einfachheit halber an, dass n positiv ganzzahlig und ungerade ist. Die Werte von a und b können beliebige komplexe Werte annehmen.

Folgende Grenzwerte sind bekannt:

$$\lim_{n \rightarrow \infty} \left(1 + \frac{x}{n}\right)^n = e^x \quad \text{und} \quad \lim_{n \rightarrow \infty} n(x^{1/n} - 1) = \log(x)$$

Definieren wir $\text{expn}(x, n) = (1 + \frac{x}{n})^n$ und $\text{logn}(x, n) = n(x^{1/n} - 1)$, dann haben wir ‚parametrisierte‘ Exponential- und Logarithmusfunktionen, die für $n \rightarrow \infty$ asymptotisch gegen die originalen Funktionen e^x und $\log(x)$ gehen.

Ersetzen wir in den Operator-Matrizen ‚OpMatrix1‘ und ‚OpMatrix2‘ e^x durch expn und \log durch logn , dann erhalten wir folgende Operator-Matrizen (mit $\text{dim} = 2$):

OpMatrix1n (kommutative Operatoren auf der Diagonale, Verschachtelungsoperator: „+“):

Table[Nest[expn,Nest[logn,a,i]+Nest[logn,b,j],Max[i,j]],{i,0,dim},{j,0,dim}]/MatrixForm

$$\begin{pmatrix} a+b & (\frac{a}{n} + \frac{b}{n})^n & (\frac{a}{n} + \frac{1}{n^n}(b^{1/n} - 1)^{1/n})^n + n \\ (\frac{a}{n} + \frac{b}{n})^n & (a^{1/n} + b^{1/n} - 1)^n & (\frac{a^{1/n} + b^{1/n}}{n} - 1)^n + n \\ (\frac{1}{n^n}(a^{1/n} - 1)^{1/n} + \frac{b}{n})^n + n & (\frac{1}{n^n}(a^{1/n} - 1)^{1/n} + \frac{1}{n^n}(b^{1/n} - 1)^{1/n})^n + n & (\frac{1}{n^n}((a^{1/n} - 1)^{1/n} + (b^{1/n} - 1)^{1/n}) - 1)^n + n \end{pmatrix}$$

Wobei gilt: $\lim_{n \rightarrow \infty} (a^{1/n} + b^{1/n} - 1)^n = a b$ (siehe OpMatrix1)

OpMatrix2n (kommutative Operatoren auf der Diagonale, Verschachtelungsoperator: „*“):

Table[Nest[expn,Nest[logn,a,i]*Nest[logn,b,j],Max[i,j]],{i,0,dim},{j,0,dim}]/MatrixForm

$$\begin{pmatrix} ab & (a(b^{1/n} - 1) + 1)^n & (\frac{a^{1/n}(b^{1/n} - 1)^{1/n} - a + 1}{n})^n + n \\ (b(a^{1/n} - 1) + 1)^n & (n(a^{1/n} - 1)(b^{1/n} - 1) + 1)^n & (\frac{(n(a^{1/n} - 1)(b^{1/n} - 1) - 1) + 1}{n})^n + n \\ (\frac{b^{1/n}(a^{1/n} - 1)^{1/n} - b + 1}{n})^n + n & (\frac{(n^{1/n}(a^{1/n} - 1)^{1/n} - 1)(b^{1/n} - 1) + 1}{n})^n + n & (\frac{(n^{1/n}(a^{1/n} - 1)^{1/n} - 1)(n^{1/n}(b^{1/n} - 1)^{1/n} - 1) + 1}{n})^n + n \end{pmatrix}$$

Wobei gilt: $\lim_{n \rightarrow \infty} (n(a^{1/n} - 1)(b^{1/n} - 1) + 1)^n = a^{\log(b)}$ (siehe OpMatrix2)

OpMatrix5n (kommutative Operatoren auf der Diagonale, Verschachtelungsoperator: „+“):

Table[Nest[logn,Nest[expn,a,i]+Nest[expn,b,j],Max[i,j]],{i,0,dim},{j,0,dim}]/MatrixForm

$$\begin{pmatrix} a+b & n((a + (\frac{b+n}{n})^n)^{1/n} - 1) & n((n((a + (\frac{b+n}{n})^n + n)^{1/n} - 1))^{1/n} - 1) \\ n(((\frac{a+n}{n})^n + b)^{1/n} - 1) & n(((\frac{a+n}{n})^n + (\frac{b+n}{n})^n)^{1/n} - 1) & n((n(((\frac{a+n}{n})^n + (\frac{b+n}{n})^n + n)^{1/n} - 1))^{1/n} - 1) \\ n((n(((\frac{a+n}{n})^n + n)^{1/n} + b)^{1/n} - 1))^{1/n} - 1 & n((n(((\frac{a+n}{n})^n + n)^{1/n} + (\frac{b+n}{n})^n)^{1/n} - 1))^{1/n} - 1 & n((n(((\frac{a+n}{n})^n + n)^{1/n} + (\frac{b+n}{n})^n + n)^{1/n} - 1))^{1/n} - 1 \end{pmatrix}$$

Wobei gilt: $\lim_{n \rightarrow \infty} n(((\frac{a+n}{n})^n + (\frac{b+n}{n})^n)^{1/n} - 1) = \text{Log}[e^a + e^b]$ (siehe OpMatrix5)

OpMatrix6n (kommutative Operatoren auf der Diagonale, Verschachtelungsoperator:

„*“):

Table[Nest[logn,Nest[expn,a,i]*Nest[expn,b,j],Max[i,j]],{i,0,dim},{j,0,dim}]/MatrixForm

$$\begin{pmatrix} ab & a^{\frac{1}{n}}(b+n) - n & n(n^{\frac{1}{n}}(\frac{a^{\frac{1}{n}}((\frac{b+n}{n})^n + n) - 1)^{\frac{1}{n}} - 1) \\ (a+n)b^{\frac{1}{n}} - n & \frac{ab}{n} + a + b & n(n^{\frac{1}{n}}(\frac{(a+n)((\frac{b+n}{n})^n + n)}{n^2} - 1)^{\frac{1}{n}} - 1) \\ n(n^{\frac{1}{n}}(\frac{((\frac{a+n}{n})^n + n)b^{\frac{1}{n}}}{n} - 1)^{\frac{1}{n}} - 1) & n(n^{\frac{1}{n}}(\frac{((\frac{a+n}{n})^n + n)(b+n)}{n^2} - 1)^{\frac{1}{n}} - 1) & n(n^{\frac{1}{n}}(\frac{((\frac{a+n}{n})^n + n)((\frac{b+n}{n})^n + n)}{n^2} - 1)^{\frac{1}{n}} - 1) \end{pmatrix}$$

Wobei gilt: $\lim_{n \rightarrow \infty} \frac{ab}{n} + a + b = a + b$ (siehe OpMatrix6)

Ähnlich wie bei den ursprünglichen Operator-Matrizen stellt sich natürlich auch hier die Frage nach den Körpereigenschaften. Die Körpereigenschaften sind hier nicht ganz so einfach zu untersuchen, mit etwas Fleiß und Ausdauer ist die Aufgabe jedoch zu lösen.

Von den Eigenschaften der originalen Operator-Matrizen sind folgende Eigenschaften erhalten geblieben bzw. gelten nicht mehr:

- 1) Auch hier gilt: Bei allen Operator-Matrizen (mit Ausnahme der Beispiele 4 und 8) liegen kommutative Operatoren auf der Diagonale (markiert in roter Farbe)
- 2) Auch hier gilt: Die Operatoren abseits der Diagonalen sind nicht kommutativ.
- 3) Gilt ebenfalls (ist klar wegen der Definitionen von logn und expn)
- 4) **Diese Eigenschaft gilt hier nicht mehr: ‚benachbarte‘ Operatoren auf den Diagonalen bilden keinen Körper bezüglich der Menge der komplexen Zahlen**
- 5) Diese Eigenschaft bleibt erhalten: Jeweils zwei Operatoren, die an der gleichen Diagonalposition zweier Operatormatrizen sitzen, bilden ebenfalls einen Körper bezüglich der Menge der komplexen Zahlen für folgende Operator-Matrizen-Paare: (OpMatrix1n,OpMatrix2n), entsprechend den Verschachtelungsoperatoren „+“ und „*“

Im Einzelnen bilden folgende auf gleichen Diagonalpositionen liegenden Operator-Paare einen Körper:

(OpMatrix1n(1,1) und OpMatrix2n(1,1) :
 $a + b$ und $a \cdot b$ (ist trivial, da n nicht vorkommt)

(OpMatrix1n(2,2) und OpMatrix2n(2,2):
 $(a^{\frac{1}{n}} + b^{\frac{1}{n}} - 1)^n$ und $(n(a^{\frac{1}{n}} - 1)(b^{\frac{1}{n}} - 1) + 1)^n$ (relativ einfach zu überprüfen)

(OpMatrix1n(3,3) und OpMatrix2n(3,3):
 $(\frac{n^{\frac{1}{n}}((a^{\frac{1}{n}} - 1)^{\frac{1}{n}} + (b^{\frac{1}{n}} - 1)^{\frac{1}{n}}) - 1)^n + n}{n})^n$ und $(\frac{(n(n^{\frac{1}{n}}(a^{\frac{1}{n}} - 1)^{\frac{1}{n}} - 1)(n^{\frac{1}{n}}(b^{\frac{1}{n}} - 1)^{\frac{1}{n}} - 1) + 1)^n + n}{n})^n$
 (schwierig zu verifizieren, daher bisher nur Vermutung)

(OpMatrix5n(1,1) und OpMatrix6n(1,1):
 $a + b$ und $a \cdot b$ (ist trivial, da n nicht vorkommt)

(OpMatrix5n(2,2) und OpMatrix6n(2,2):

$$n\left(\left(\left(\frac{a+n}{n}\right)^n + \left(\frac{b+n}{n}\right)^n\right)^{\frac{1}{n}} - 1\right) \text{ und } \frac{ab}{n} + a + b \text{ (relativ einfach zu überprüfen)}$$

(OpMatrix5n(3,3) und OpMatrix6n(3,3):

$$n\left(\left(n\left(\left(\left(\frac{a+n}{n}\right)^{n+n} + \left(\frac{b+n}{n}\right)^{n+n}\right)^{\frac{1}{n}} - 1\right)\right)^{\frac{1}{n}} - 1\right) \text{ und } n\left(n^{\frac{1}{n}}\left(\frac{\left(\left(\frac{a+n}{n}\right)^{n+n}\right)\left(\left(\frac{b+n}{n}\right)^{n+n}\right)}{n^2} - 1\right)^{\frac{1}{n}} - 1\right)$$

(schwierig zu verifizieren, daher bisher nur Vermutung)

Schauen wir uns nun die interessanten Operator-Paare (OpMatrix1n(2,2) und OpMatrix2n(2,2) sowie (OpMatrix5n(2,2) und (OpMatrix6n(2,2) etwas genauer an (wir verwenden für die Operatoren die Symbole \oplus und \otimes) dann finden wir:

(OpMatrix1n(2,2) und OpMatrix2n(2,2):

„Additionsoperator“:	$a \oplus b := (-1 + a^{\frac{1}{n}} + b^{\frac{1}{n}})^n$
„Multiplikationsoperator“:	$a \otimes b := (1 + (-1 + a^{\frac{1}{n}})(-1 + b^{\frac{1}{n}})n)^n$
Assoziativität:	ist erfüllt für beide Operatoren
Kommutativität:	ist erfüllt für beide Operatoren
Neutrales Element der Addition:	1
Inverses Element der Addition:	$(2 - a^{\frac{1}{n}})^n$
Neutrales Element der Multiplikation:	$(1 + \frac{1}{n})^n$
Inverses Element der Multiplikation:	$(1 + \frac{1}{(-1 + a^{\frac{1}{n}})n^2})^n$
Distributivgesetz:	ist erfüllt
Körpereigenschaft:	ist erfüllt

(OpMatrix1(2,2) und OpMatrix2(2,2) (Grenzfall für $n \rightarrow \infty$):

„Additionsoperator“:	$a \oplus b := a + b$
„Multiplikationsoperator“:	$a \otimes b := a^{\log(b)}$
Assoziativität:	ist erfüllt für beide Operatoren
Kommutativität:	ist erfüllt für beide Operatoren
Neutrales Element der Addition:	1
Inverses Element der Addition:	$\frac{1}{a}$
Neutrales Element der Multiplikation:	e
Inverses Element der Multiplikation:	$e^{\frac{1}{\log[a]}}$
Distributivgesetz:	ist erfüllt
Körpereigenschaft:	ist erfüllt

(OpMatrix1n(2,2) und OpMatrix2n(2,2) (Grenzfall für $n = 1$):

„Additionsoperator“:	$a \oplus b := a + b - 1$
„Multiplikationsoperator“:	$a \otimes b := (a - 1)(b - 1) + 1$

Assoziativität:	ist erfüllt für beide Operatoren
Kommutativität:	ist erfüllt für beide Operatoren
Neutrales Element der Addition:	1
Inverses Element der Addition:	$2 - a$
Neutrales Element der Multiplikation:	2
Inverses Element der Multiplikation:	$1 + \frac{1}{a-1}$
Distributivgesetz:	ist erfüllt
Körpereigenschaft:	ist erfüllt

(OpMatrix5n(2,2) und OpMatrix6n(2,2):

„Additionsoperator“:	$a \oplus b := n \left(\left(\frac{a+n}{n} \right)^n + \left(\frac{b+n}{n} \right)^n \right)^{\frac{1}{n}} - 1$
„Multiplikationsoperator“:	$a \otimes b := \frac{ab}{n} + a + b$
Assoziativität:	ist erfüllt für beide Operatoren
Kommutativität:	ist erfüllt für beide Operatoren
Neutrales Element der Addition:	$-n$
Inverses Element der Addition:	$n \left(-1 + \left(- \left(\frac{a+n}{n} \right)^n \right)^{\frac{1}{n}} \right) = -2n - a$
	(n ungerade)
Neutrales Element der Multiplikation:	0
Inverses Element der Multiplikation:	$-\frac{an}{a+n}$
Distributivgesetz:	ist erfüllt
Körpereigenschaft:	ist erfüllt

(OpMatrix5(2,2) und OpMatrix6(2,2) (Grenzfall für $n \rightarrow \infty$):

„Additionsoperator“:	$a \oplus b := \log(e^a + e^b)$
„Multiplikationsoperator“:	$a \otimes b := a + b$
Assoziativität:	ist erfüllt für beide Operatoren
Kommutativität:	ist erfüllt für beide Operatoren
Neutrales Element der Addition:	$-\infty$
Inverses Element der Addition:	$a + i\pi$ (genauer: $a + i(2k+1)\pi, k \in \mathbb{N}$)
Neutrales Element der Multiplikation:	0
Inverses Element der Multiplikation:	$-a$
Distributivgesetz:	ist erfüllt
Körpereigenschaft:	ist erfüllt

(OpMatrix5n(2,2) und OpMatrix6n(2,2) (Grenzfall für $n = 1$):

„Additionsoperator“:	$a \oplus b := a + b + 1$
„Multiplikationsoperator“:	$a \otimes b := a + b + ab$
Assoziativität:	ist erfüllt für beide Operatoren
Kommutativität:	ist erfüllt für beide Operatoren

Neutrales Element der Addition:	-1
Inverses Element der Addition:	$-2 - a$
Neutrales Element der Multiplikation:	0
Inverses Element der Multiplikation:	$\frac{-a}{a+1}$
Distributivgesetz:	ist erfüllt
Körpereigenschaft:	ist erfüllt

Betrachten wir nun das Beispiel **OpMatrix1n(2,2)** und die Grenzfälle für $n \rightarrow \infty$ und $n = 1$:

$n = 1$: $a \oplus b$ wird zu $a + b - 1$ (bis auf -1 identisch mit der normalen Addition)

$n \rightarrow \infty$: $a \oplus b$ wird zu $a \cdot b$ (identisch mit normaler Multiplikation)

Durch Variieren des Parameters n bekommen wir unendlich viele Operatoren, die einen kontinuierlichen Übergang von der normalen Addition zur normalen Multiplikation darstellen.

Beim Beispiel **OpMatrix5n(2,2)** und den Grenzfällen für $n \rightarrow \infty$ und $n = 1$ erhalten wir:

$n = 1$: $a \oplus b$ wird zu $a + b + 1$ (bis auf 1 identisch mit der normalen Addition)

$n \rightarrow \infty$: $a \oplus b$ wird zu $\log(e^a + e^b)$ (identisch mit dem Jacobi-Additions-Operator)

Durch Variieren des Parameters n bekommen wir unendlich viele Operatoren, die einen kontinuierlichen Übergang von der normalen Addition zur Jacobi-Addition darstellen.

18.4.3_WERTEBEREICH DER ERWEITERTEN OPERATOREN

Man mag sich die Frage stellen, welchen Wertebereich (insbesondere ganzzahlige Werte) derartige Operatoren besitzen.

Wir nehmen an, die ursprüngliche Definitionsmenge für a und b sei \mathbb{N} . Weiterhin wollen wir iterativ vorgehen, in dem wir beliebig viele Werte von $a \oplus b$ bzw. $a \otimes b$ berechnen und iterativ die sich ergebende Wertemenge zur neuen Definitionsmenge des nächsten Iterationsschrittes machen. Interessant ist nun die Frage, welche (positiven) ganzzahligen Werte auf diese Weise entstehen. Folgende Tabellen könnten darüber Auskunft geben:

n	Positiver Ganzzahliger Wertebereich für $a \oplus b := (-1 + a^{\frac{1}{n}} + b^{\frac{1}{n}})^n$
1	{1,2,3,4,5,6,7,8,9,10,...}: alle natürlichen Zahlen
2	{4,8,9,12,16,18,20,24,25,27,...}: Zahlen, die nicht quadratfrei sind
3	{8,16,24,27,32,40,48,54,56,...}: Zahlen, die nicht kubikfrei sind
4	(keine)
5	(keine)
6	(keine)

n	Positiver Ganzzahliger Wertebereich für $a \otimes b := (1 + (-1 + a^{\frac{1}{n}})(-1 + b^{\frac{1}{n}})n)^n$
1	{1,2,3,4,5,6,7,8,9,10,...}: alle natürlichen Zahlen
2	{9,25,49,81,121,169,289,...}: $(2n + 1)^2$ n=1,2,3,4,5,...
3	(???)
4	(???)

Hinweis: Der Autor gibt keine Gewähr, dass diese Daten richtig sind, da sie rein empirisch ermittelt wurden.

20.8.2_WERTERHALTENDE ÜBERGÄNGE VON 2-ER-POTENZEN IM OCRON-RAUM

Wir wollen nun sogenannte werterhaltende Übergänge von OCRONs (bzw. GOCRONs) betrachten. Von besonderem Interesse sind diese Übergänge (man könnte sie auch Umformungen nennen) für Zweier-Potenzen, da sie eine Möglichkeit bieten, die Addition ins Spiel zu bringen, gemäß der Formel

$$(2^a)^b = 2^{a*b} \text{ sowie } (2^{2^a})^{2^b} = 2^{2^{a+b}}$$

Zuerst gehen wir eine noch einfachere Variante an: Die Inkrementation um den Wert 1.

Sei im Folgenden $\text{ld}(x) = \frac{\ln(x)}{\ln(2)}$ der Zweierlogarithmus (Logarithmus Dualis).

Dann gilt offensichtlich: $x + 1 = \text{ld}(2^{x+1}) = \text{ld}(2^x \cdot 2)$, oder in OCRON-Schreibweise:

$0+1=\text{ld}(2\langle\text{ocron}\rangle^{2*})$ (wobei $\langle\text{ocron}\rangle$ für die Zeichenkette eines beliebigen OCRONs steht. Den Zweierlogarithmus eines OCRONs können wir formal ausführen, indem wir eine vorangestellte 2 und ein angehängtes ^ einfach weglassen. Dies setzt voraus, dass wir ein OCRON betrachten, das die Form $2\langle\text{ocron}\rangle^{\wedge}$ besitzt. Daher stehen wir nun vor dem Problem der Umwandlung von beliebigen OCRONs (von denen wir wissen, dass sie als Zweierpotenz dargestellt werden könne, da sie den Wert einer Zweierpotenz besitzen) in OCRONs der Form $2\langle\text{ocron}\rangle^{\wedge}$, die den Vorzug besitzen, ganz einfach ‚zweier-logarithmiert‘ werden zu können. Diese Umwandlung, die im Allgemeinen in mehreren, möglichst kleinen Schritten vollzogen wird, nennen wir werterhaltende Übergänge oder Spur eines OCRONs.

Hier eine Tabelle der ersten 50 Zweierpotenzen in OCRON-Darstellung, jeweils in $(2\langle\text{ocron}\rangle^{2*})$ -Darstellung als auch in $(2\langle\text{ocron}\rangle^{\wedge})^*$ Darstellung:

n	(G)OCRON($2^{n-1} * 2$)	(G)OCRON(2^n)
1	2: 2	2: 2
2	22*: 40	22^: 43
3	22^2*: 696	22P^: 167
4	22P^2*: 2680	222^^: 687
5	222^^2*: 11000	22PP^: 663
6	22PP^2*: 10616	222P^^: 2707
7	222P^^2*: 43320	222^P^: 2743
8	222^P^2*: 43896	222P^^: 2719
9	222P^^2*: 43512	22P2^^: 2671
10	22P2^^2*: 42744	222PP^^: 10835
11	222PP^^2*: 173368	22PPP^: 2647
12	222PPP^^2*: 42360	222^2P^^: 43923
13	222^2P^^2*: 702776	222P*P^: 10823
14	222P*P^2*: 173176	2222^P^^: 43731
15	2222^P^^2*: 699704	22P2PP^^: 42579
16	22P2PP^^2*: 681272	2222^^^: 10943
17	2222^^^2*: 175096	222^PP^: 10967
18	222^PP^2*: 175480	222P2^^^: 43443
19	222P2^^^2*: 695096	222P^P^: 10871
20	222P^P^2*: 173944	222^2PP^^: 175699
21	222^2PP^^2*: 2811192	22P22^P^^: 170707

22	22P22^P^2*:	2731320	222PPP^:	43347
23	222PPP^2*:	693560	22P2^P^:	10679
24	22P2^P^2*:	170872	222P^2P^:	173971
25	222P^2P^2*:	2783544	22PP2^:	10607
26	22PP2^2*:	169720	2222P^P^:	174355
27	2222P^P^2*:	2789688	22P2P^:	10655
28	22P2P^2*:	170488	222^22^P^:	703187
29	222^22^P^2*:	11251000	222PP^P^:	43335
30	222PP^P^2*:	693368	222P^2PP^:	692819
31	222P^2PP^2*:	11085112	22PPPP^:	10583
32	22PPPP^2*:	169336	222PP^:	10847
33	222PP^2*:	173560	22P2PPP^:	170323
34	22P2PPP^2*:	2725176	2222^PP^:	174931
35	2222^PP^2*:	2798904	22PP22^P^:	678611
36	22PP22^P^2*:	10857784	222^2P2^:	702899
37	222^2P2^2*:	11246392	222^2P^P^:	175687
38	222^2P^P^2*:	2811000	2222P^P^:	174547
39	2222P^P^2*:	2792760	22P22P^P^:	682259
40	22P22P^P^2*:	10916152	222P^2PP^:	695891
41	222P^2PP^2*:	11134264	222P^PP^:	43287
42	222P^PP^2*:	692600	222P^22^P^:	2771667
43	222P^22^P^2*:	44346680	2222^P^P^:	174919
44	2222^P^P^2*:	2798712	222^2PPP^:	702803
45	222^2PPP^2*:	11244856	22P2^2PP^:	683603
46	22P2^2PP^2*:	10937656	222P2^P^:	173779
47	222P2^P^2*:	2780472	22P2PP^P^:	170311
48	22P2PP^P^2*:	2724984	2222^2P^:	700307
49	2222^2P^2*:	11204920	222^P2^:	43887
50	222^P2^2*:	702200	222PP2^:	173491

Bei den in roter Schrift dargestellten Werten hat die $2^{\langle \text{ocron} \rangle}$ -Darstellung einen größeren Wert als die $(2^{\langle \text{ocron} \rangle})^2$ -Darstellung.

Betrachten wir nun z.B. die Zeile 20, die den Übergang von 19 zum Nachfolger 20 beschreibt.

Offensichtlich sind $222P^P^2^*$ (173944) sowie 222^2PP^* (175699) gleichwertige (G)OCRONs. Wie finden wir eine Umformungs- oder Transformationsregel zwischen diesen beiden OCRONs?

Hier ein Beispiel mit EOCRON-Daten (Typ ‚Enhanced OCRONs‘), die den Übergang von 41 nach 42 beschreiben: Die ‚Spur‘ besteht aus 467 Werten und beschreibt den Übergang zwischen den gleichwertigen EOCRONs $22P^*PP^2$ und $22P^*22^P^*$.

Übergang von 41 nach 42, CodeValues: {0,1,2,3}

Ausgabe des Programmes:

Berechnet mit EOCRONs.

eocronValue: $22P^*PP^2$: 42078

eocronvaluel: $22P^*22^P^*$: 674515

myNumber=42

Anzahl Werte: 468

Die 467 Differenzen:

spur48EOCRON={441, 768, 2940, 84, 348, 4296, 4488, 84, 60, 24, 72, 192, 1056, 24, 24, 84, 11584, 4608, 5180, 24, 24, 84, 3553, 24, 2, 6, 3, 325, 2, 8, 337, 37, 2, 8, 1142, 2, 8, 145, 1536, 4453, 2, 8, 145, 4540, 48, 5232, 748, 192, 948, 48, 4716, 24, 72, 1728, 768, 480, 24, 72, 9597, 2, 8, 32, 128, 2273, 37, 2, 8, 337, 1829, 2, 8, 809, 24, 408, 192, 144

, 24, 780, 48, 2745, 2, 8, 3473, 4488, 84, 60, 24, 72, 192, 1056, 24, 24, 84, 10304, 748, 192, 948, 48, 377, 2, 8, 145, 4269, 1739, 768, 3016, 3464, 84, 348, 2869, 1764, 3072, 4 151, 84, 348, 4296, 4488, 84, 60, 24, 72, 192, 1056, 24, 24, 84, 13337, 2, 8, 32, 128, 22 73, 37, 2, 8, 337, 1829, 2, 8, 809, 24, 408, 192, 144, 24, 780, 48, 2745, 2, 8, 19857, 336 5, 2, 8, 145, 1352, 192, 948, 48, 732, 84, 2988, 84, 5280, 48, 8980, 13877, 2, 8, 32, 128 , 493, 19, 4989, 1536, 11417, 2, 8, 32, 5334, 2, 8, 32, 4669, 22076, 24, 24, 84, 2368, 59 85, 24, 12, 1308, 24, 2, 6, 3, 1501, 24, 2, 6, 3, 4573, 24, 2, 6, 3, 325, 2, 8, 337, 37, 2, 8, 1142, 2, 8, 758, 2, 8, 3830, 2, 8, 182, 2, 8, 145, 37, 2, 8, 13421, 4257, 24, 2, 6, 3, 325, 2 , 8, 337, 37, 2, 8, 1142, 2, 8, 145, 1536, 4453, 2, 8, 145, 1637, 2, 8, 1270, 2, 8, 145, 552 8, 24, 24, 84, 8096, 48, 1556, 293, 2, 8, 465, 768, 7516, 4608, 5180, 24, 24, 84, 39204, 9920, 4028, 48, 8980, 9789, 24, 2, 6, 3, 85, 2, 8, 24, 5, 1, 3, 3, 338, 2, 8, 32, 96, 1, 19, 4 , 1, 4, 1, 3, 3, 1, 11, 1, 5189, 2, 8, 32, 128, 3870, 24, 2, 6, 3, 1440, 37, 2, 8, 337, 192, 53 , 2, 8, 32, 128, 6430, 24, 2, 6, 3, 7544, 24, 24, 84, 4068, 8957, 24, 2, 6, 3, 11541, 2, 8, 3 2, 128, 2273, 37, 2, 8, 337, 1829, 2, 8, 809, 24, 408, 192, 144, 24, 780, 48, 2745, 2, 8, 1 9857, 3365, 2, 8, 145, 1352, 192, 948, 48, 732, 84, 2988, 84, 5280, 48, 8980, 9820, 460 8, 5180, 24, 24, 84, 3553, 24, 2, 6, 3, 325, 2, 8, 337, 37, 2, 8, 1142, 2, 8, 145, 1536, 445 3, 2, 8, 145, 4540, 48, 5232, 748, 192, 948, 48, 4716, 24, 72, 1728, 768, 480, 24, 72, 95 97, 2, 8, 32, 128, 2273, 37, 2, 8, 337, 1829, 2, 8, 809, 24, 408, 192, 144, 24, 780, 48, 27 45, 2, 8, 3473, 4488, 84, 60, 24, 72, 192, 1056, 24, 24, 84, 10304, 748, 192};

Tabelle 1: „Spur“ aller gleichwertigen EOCRONs für 2⁴²

EGOCRON: {EOCRON} number: 2 ⁴²	EGOCRON: {EOCRON} number: 2 ⁴²
42078: {2,2,P,*P,P,^,2}	126599: {P,^,2,^,2,2,*P,^}
42519: {2,2,P,2,*P,P,^}	126647: {P,^,2,^,2,2,^,P,^}
43287: {2,2,2,P,*P,P,^}	129392: {P,^,^,2,P,P,^,*P}
46227: {2,^,P,*2,P,*^}	129394: {P,^,^,2,P,P,^,*2}
46311: {2,^,P,*^,2,P,^}	129402: {P,^,^,2,P,P,^,2,2}
46659: {2,^,P,2,P,*,*^}	132875: {2,*,*P,^,*,*2,^}
50955: {^,*P,^,*,*2,^}	137363: {2,*P,2,*2,P,*^}
55443: {^,P,2,*2,P,*^}	137447: {2,*P,2,*^,2,P,^}
55527: {^,P,2,*^,2,P,^}	137507: {2,*P,2,P,*2,*^}
55587: {^,P,2,P,*2,*^}	137531: {2,*P,2,P,*^,2,^}
55611: {^,P,2,P,*^,2,^}	137603: {2,*P,2,P,2,*,*^}
55683: {^,P,2,P,2,*,*^}	137795: {2,*P,2,2,P,*,*^}
55875: {^,P,2,2,P,*,*^}	138851: {2,*P,^,2,P,2,*^}
56931: {^,P,^,2,P,2,*^}	138875: {2,*P,^,2,P,^,2,^}
56955: {^,P,^,2,P,^,2,^}	138899: {2,*P,^,2,2,P,*^}
56979: {^,P,^,2,2,P,*^}	138983: {2,*P,^,2,^,2,P,^}
57063: {^,P,^,2,^,2,P,^}	149287: {2,P,*P,^,*2,P,^}
68647: {P,*,*^,*,*2,2,^}	150035: {2,P,*2,2,2,*,*^}
73255: {P,*P,^,2,*2,P,^}	150227: {2,P,*2,2,^,P,*^}
78435: {P,*,*^,2,P,2,*^}	151175: {2,P,*^,2,2,*P,^}
78459: {P,*^,*2,2,P,^,2,^}	151223: {2,P,*^,2,2,^,P,^}
78483: {P,*^,*2,2,2,P,*^}	151600: {2,P,P,*,*^,*^}
78567: {P,*^,*2,^,2,P,^}	151602: {2,P,P,*,*^,*2,^}
82120: {P,P,*,*^,*^,2,*}	151610: {2,P,P,*,*^,2,2,^}
82144: {P,P,*,*^,2,2,*}	151755: {2,P,P,*,*^,*2,^}
82146: {P,P,*,*^,2,2,*}	156024: {2,P,2,*P,P,^,2,*}
82152: {P,P,*,*^,2,2,*}	157763: {2,P,2,2,*P,*,*^}
82155: {P,P,*,*^,2,2,^}	158531: {2,P,2,2,^,P,*,*^}
82480: {P,P,*,*2,*^,*}	161547: {2,P,^,P,^,*,*2,^}
82482: {P,P,*,*2,*^,*2,^}	165011: {2,2,*P,*2,P,*^}
82490: {P,P,*,*2,*^,2,2}	165095: {2,2,*P,*^,2,2,^}
82827: {P,P,*,*^,2,*2,^}	165443: {2,2,*P,2,P,*,*^}
82864: {P,P,*,*^,2,^,*}	168312: {2,2,P,*P,P,^,2,*}
82866: {P,P,*,*^,2,^,*2}	170076: {2,2,P,2,*P,P,^,*}
82874: {P,P,*,*^,2,^,2,2}	173148: {2,2,2,P,*P,P,^,*}
84016: {P,P,*2,*^,*,*}	177299: {2,2,^,P,*2,P,*^}
84018: {P,P,*2,*^,*^,2,2}	177383: {2,2,^,P,*^,2,P,^}
84026: {P,P,*2,*^,*^,2,2}	177731: {2,2,^,P,2,P,*,*^}
84171: {P,P,*2,*^,*^,2,^}	182027: {2,^,*P,^,*,*2,^}
85707: {P,P,*^,2,^,*2,^}	186515: {2,^,P,2,*2,P,*^}
90160: {P,P,2,*,*^,*,*}	186599: {2,^,P,2,*^,2,P,^}
90162: {P,P,2,*,*^,*^,2,2}	186659: {2,^,P,2,P,*2,*^}
90170: {P,P,2,*,*^,*^,2,2}	186683: {2,^,P,2,P,*^,2,^}
90315: {P,P,2,*,*^,*2,^}	186755: {2,^,P,2,P,2,*,*^}
94855: {P,P,^,*2,2,*P,^}	186947: {2,^,P,2,2,P,*,*^}
94903: {P,P,^,*2,2,^,P,^}	188003: {2,^,P,^,2,P,2,*^}
100135: {P,2,*P,P,^,*2,P,^}	188027: {2,^,P,^,2,P,^,2,^}
100883: {P,2,*2,2,*P,*^}	188051: {2,^,P,^,2,2,P,*^}
101075: {P,2,*2,2,^,P,*^}	188135: {2,^,P,^,2,^,2,P,^}
102023: {P,2,*^,2,2,*P,^}	201472: {^,*P,*^,*,*,*}
102071: {P,2,*^,2,2,^,P,^}	201474: {^,*P,*^,*,*2,2}
106787: {P,2,2,*P,*2,*^}	201482: {^,*P,*^,*,*2,2}
106811: {P,2,2,*P,*^,2,^}	201514: {^,*P,*^,*2,2,2}
106883: {P,2,2,*P,2,*,*^}	201642: {^,*P,*^,2,2,2,2}
108611: {P,2,2,2,*P,*,*^}	203915: {^,*P,^,*2,*2,^}
109379: {P,2,2,2,^,P,*,*^}	203952: {^,*P,^,*2,^,*^}
109859: {P,2,2,^,P,*2,*^}	203954: {^,*P,^,*2,^,*2}
109883: {P,2,2,^,P,*^,2,^}	203962: {^,*P,^,*2,^,2,2}
109955: {P,2,2,^,P,2,*,*^}	204299: {^,*P,^,2,*,*2,^}
119552: {P,^,P,*^,*,*,*}	206128: {^,*2,P,P,*,*^}
119554: {P,^,P,*^,*,*,*2}	206130: {^,*2,P,P,*,*^,2}
119562: {P,^,P,*^,*,*2,2}	206138: {^,*2,P,P,*,*2,2}
119594: {P,^,P,*^,*2,2,2}	206947: {^,*2,2,*P,2,*^}
119722: {P,^,P,*^,2,2,2,2}	206971: {^,*2,2,*P,^,2,^}
121995: {P,^,P,^,*2,*2,^}	207379: {^,*2,2,2,*P,*^}
122032: {P,^,P,^,*2,^,*}	207571: {^,*2,2,2,^,P,*^}
122034: {P,^,P,^,*2,^,*2}	207715: {^,*2,2,^,P,2,*^}
122042: {P,^,P,^,*2,^,2,2}	207739: {^,*2,2,^,P,^,2,^}
122379: {P,^,P,^,2,*,*2,^}	208519: {^,*2,^,2,2,*P,^}
124208: {P,^,2,P,P,*,*,*}	208567: {^,*2,^,2,2,^,P,^}
124210: {P,^,2,P,P,*,*^,2}	211312: {^,*^,2,P,P,^,*,*}
124218: {P,^,2,P,P,*^,2,2}	211314: {^,*^,2,P,P,^,*2}
125027: {P,^,2,2,*P,2,*^}	211322: {^,*^,2,P,P,^,2,2}
125051: {P,^,2,2,*P,^,2,^}	231179: {^,2,*P,^,*,*2,^}
125459: {P,^,2,2,2,*P,*^}	234544: {^,2,P,P,*,*^,*}
125651: {P,^,2,2,2,^,P,*^}	234546: {^,2,P,P,*,*^,*2}
125795: {P,^,2,2,^,P,2,*^}	234554: {^,2,P,P,*,*^,2,2}
125819: {P,^,2,2,^,P,^,2,^}	234699: {^,2,P,P,*,*^,2,^}

EGOCRON: {EOCRON} number: 2^42	EGOCRON: {EOCRON} number: 2^42
236051: {^,2,P,2,2,*P,*^}	360674: {P,P,2,*,*^,2,*2}
236243: {^,2,P,2,2,^P,*^}	360680: {P,P,2,*,*^,2,2,*}
237191: {^,2,P,^,2,2,*P,^}	360683: {P,P,2,*,*^,2,2,^}
237239: {^,2,P,^,2,2,^P,^}	361008: {P,P,2,*,*2,*^,*}
237971: {^,2,2,*P,2,P,*^}	361010: {P,P,2,*,*2,*^,*2}
238055: {^,2,2,*P,^,2,P,^}	361018: {P,P,2,*,*2,*^,2,2}
241043: {^,2,2,^P,2,P,*^}	361355: {P,P,2,*,*^,2,*2,^}
241127: {^,2,2,^P,^,2,P,^}	361392: {P,P,2,*,*^,2,^,*}
246407: {^,^,*,*2,2,*P,^}	361394: {P,P,2,*,*^,2,*2}
246455: {^,^,*,*2,2,^P,^}	361402: {P,P,2,*,*^,2,^,2}
255435: {^,^,2,P,P,^,*2,^}	362544: {P,P,2,*2,*,*^,*}
269312: {P,*,*P,^,*,*,*}	362546: {P,P,2,*2,*,*^,2}
269314: {P,*,*P,^,*,*,*2}	362554: {P,P,2,*2,*,*^,2,2}
269322: {P,*,*P,^,*,*2,2}	362699: {P,P,2,*2,*^,*2,^}
269354: {P,*,*P,^,*,*2,2,2}	364235: {P,P,2,*^,2,^,*2,^}
269482: {P,*,*P,^,*2,2,2,2}	368688: {P,P,2,2,*,*^,*}
269975: {P,*,*P,^,2,2,P,P,^}	368690: {P,P,2,2,*,*^,*2}
269994: {P,*,*P,^,2,2,2,2,2}	368698: {P,P,2,2,*,*^,2,2}
274983: {P,*,*^,*2,*2,P,^}	368843: {P,P,2,2,*,*^,*2,^}
276519: {P,*,*^,2,*,*2,P,^}	370480: {P,P,2,2,P,^,*^,*}
287936: {P,*P,2,P,*^,*,*}	370482: {P,P,2,2,P,^,*^,*2}
287938: {P,*P,2,P,*^,*,*2}	370490: {P,P,2,2,P,^,*^,2,2}
287946: {P,*P,2,P,*^,*2,2}	371760: {P,P,2,2,^,*^,*,*}
287978: {P,*P,2,P,*^,2,2,2}	371762: {P,P,2,2,^,*^,*^,2}
293312: {P,*P,^,2,P,^,*,*}	371770: {P,P,2,2,^,*^,*^,2,2}
293314: {P,*P,^,2,P,^,*,*2}	371915: {P,P,2,2,^,*^,*^,2,^}
293322: {P,*P,^,2,P,^,*2,2}	377443: {P,P,^,*,*2,P,2,*^}
293354: {P,*P,^,2,P,^,2,2,2}	377467: {P,P,^,*,*2,P,^,2,^}
298023: {P,*2,*^,*2,P,^}	377491: {P,P,^,*,*2,2,P,^}
320099: {P,*^,2,*2,P,2,*^}	377575: {P,P,^,*,*2,^,2,P,^}
320123: {P,*^,2,*2,P,^,2,^}	385671: {P,P,^,2,*2,2,*P,^}
320147: {P,*^,2,*2,2,P,*^}	385719: {P,P,^,2,*2,2,^P,^}
320231: {P,*^,2,*2,^,2,P,^}	387275: {P,P,^,2,2,*^,*2,^}
322599: {P,*^,2,^,*2,P,^}	387568: {P,P,^,2,2,P,^,*^,*}
328584: {P,P,*,*^,2,*2,*}	387570: {P,P,^,2,2,P,^,*^,2}
328608: {P,P,*,*^,2,2,*,*}	387578: {P,P,^,2,2,P,^,*^,2,2}
328620: {P,P,*,*^,2,2,^,*}	388043: {P,P,^,2,2,^,*^,2,^}
329928: {P,P,*,*2,*^,*2,*}	388811: {P,P,^,2,^,2,^,*2,^}
329952: {P,P,*,*2,*^,2,*,*}	396327: {P,2,*,*^,*2,P,^}
329954: {P,P,*,*2,*^,2,*2}	400935: {P,2,*P,^,2,*2,P,^}
329960: {P,P,*,*2,*^,2,2,*}	406115: {P,2,*^,*2,P,2,*^}
329963: {P,P,*,*2,*^,2,2,^}	406139: {P,2,*^,*2,P,^,2,^}
331464: {P,P,*,*^,2,^,*2,*}	406163: {P,2,*^,*2,2,P,*^}
331488: {P,P,*,*^,2,^,2,*,*}	406247: {P,2,*^,*2,^,2,P,^}
331490: {P,P,*,*^,2,^,2,*2}	445451: {P,2,^,*^,*,*2,^}
331496: {P,P,*,*^,2,^,2,2,*}	455371: {P,2,^,^,*2,^,*2,^}
331498: {P,P,*,*^,2,^,2,2,^}	459399: {P,^,*,*2,2,*P,^}
336072: {P,P,*2,*,*^,*2,*}	459447: {P,^,*,*2,2,^P,^}
336096: {P,P,*2,*,*^,2,*,*}	468427: {P,^,*2,P,P,^,*2,^}
336098: {P,P,*2,*,*^,2,*2}	478216: {P,^P,*^,*,*2,*}
336104: {P,P,*2,*,*^,2,2,*}	478240: {P,^P,*^,*,*2,*,*}
336107: {P,P,*2,*,*^,2,2,^}	478242: {P,^P,*^,*,*2,*2}
336432: {P,P,*2,*2,*^,*,*}	478248: {P,^P,*^,*,*2,2,*}
336434: {P,P,*2,*2,*^,*2}	478251: {P,^P,*^,*,*2,2,^}
336442: {P,P,*2,*2,*^,2,2}	478336: {P,^P,*^,*2,*,*}
336779: {P,P,*2,*^,2,*2,^}	478338: {P,^P,*^,*2,*,*2}
336816: {P,P,*2,*^,2,^,*,*}	478346: {P,^P,*^,*2,*2,2}
336818: {P,P,*2,*^,2,^,*2}	478370: {P,^P,*^,*2,2,*2}
336826: {P,P,*2,*^,2,^,2,2}	478375: {P,^P,*^,*2,2,P,^}
337968: {P,P,*2,2,*^,*,*}	478376: {P,^P,*^,*2,2,2,*}
337970: {P,P,*2,2,*^,*^,*2}	478379: {P,^P,*^,*2,2,2,^}
337978: {P,P,*2,2,*^,*2,2}	478382: {P,^P,*^,*2,2,^,2}
338736: {P,P,*2,2,^,*^,*}	478720: {P,^P,*^,2,*,*,*}
338738: {P,P,*2,2,^,*^,*2}	478722: {P,^P,*^,2,*,*,*2}
338746: {P,P,*2,2,^,*^,2,2}	478730: {P,^P,*^,2,*,*2,2}
342576: {P,P,*^,2,2,*^,*}	478762: {P,^P,*^,2,*2,2,2}
342578: {P,P,*^,2,2,*^,*2}	478858: {P,^P,*^,2,2,*2,2}
342586: {P,P,*^,2,2,*^,*2}	478859: {P,^P,*^,2,2,*2,^}
342768: {P,P,*^,2,2,^,*^,*}	478878: {P,^P,*^,2,2,P,^,2}
342770: {P,P,*^,2,2,^,*^,*2}	478882: {P,^P,*^,2,2,2,*2}
342778: {P,P,*^,2,2,^,*^,2,2}	478883: {P,^P,*^,2,2,2,*^}
342923: {P,P,*^,2,^,2,*2,^}	478887: {P,^P,*^,2,2,2,P,^}
342960: {P,P,*^,2,^,2,^,*}	478888: {P,^P,*^,2,2,2,2,*}
342962: {P,P,*^,2,^,2,^,*2}	478891: {P,^P,*^,2,2,2,2,^}
342970: {P,P,*^,2,^,2,^,2,2}	478894: {P,^P,*^,2,2,2,^,2}
356391: {P,P,P,^,*,*2,P,^}	478895: {P,^P,*^,2,2,2,^,2}
360648: {P,P,2,*,*^,*2,*}	478906: {P,^P,*^,2,2,^,2,2}
360672: {P,P,2,*,*^,2,*,*}	478907: {P,^P,*^,2,2,^,2,^}

EGOCRON: {EOCRON} number: 2^42	EGOCRON: {EOCRON} number: 2^42
484096: {P,^,P,2,*^,*^,*^,*^,*^,*^}	602723: {2,P,*^,*^,2,P,2,*^,*^}
484098: {P,^,P,2,*^,*^,*^,*^,*^,*^}	602747: {2,P,*^,*^,2,P,^,2,*^,*^}
484106: {P,^,P,2,*^,*^,*^,*^,*^,*^}	602771: {2,P,*^,*^,2,2,P,*^,*^}
484138: {P,^,P,2,*^,*^,2,2,2,2}	602855: {2,P,*^,*^,2,^,2,P,*^,*^}
484266: {P,^,P,2,*^,*^,2,2,2,2}	606408: {2,P,P,*^,*^,*^,2,*^,*^}
488136: {P,^,P,^,*^,2,*^,2,*^,*^,*^}	606432: {2,P,P,*^,*^,*^,2,*^,*^}
488160: {P,^,P,^,*^,2,*^,2,*^,*^,*^}	606434: {2,P,P,*^,*^,*^,2,*^,*^}
488162: {P,^,P,^,*^,2,*^,2,*^,*^,*^}	606440: {2,P,P,*^,*^,*^,2,2,*^,*^}
488168: {P,^,P,^,*^,2,*^,2,2,2,*^,*^}	606443: {2,P,P,*^,*^,*^,2,2,2,*^,*^}
488171: {P,^,P,^,*^,2,*^,2,2,2,*^,*^}	606768: {2,P,P,*^,*^,2,*^,*^,*^,*^}
489611: {P,^,P,^,2,*^,2,*^,2,*^,*^,*^}	606770: {2,P,P,*^,*^,2,*^,*^,*^,*^}
489648: {P,^,P,^,2,*^,2,*^,2,*^,*^,*^}	606778: {2,P,P,*^,*^,2,*^,*^,2,2,*^,*^}
489650: {P,^,P,^,2,*^,2,*^,2,*^,*^,*^}	607115: {2,P,P,*^,*^,*^,2,*^,2,*^,*^}
489658: {P,^,P,^,2,*^,2,*^,2,2,2,*^,*^}	607152: {2,P,P,*^,*^,*^,2,*^,*^,*^,*^}
489995: {P,^,P,^,2,2,*^,*^,2,*^,*^,*^}	607154: {2,P,P,*^,*^,*^,2,*^,*^,*^,*^}
490187: {P,^,P,^,2,2,*^,*^,2,*^,*^,*^}	607162: {2,P,P,*^,*^,*^,2,*^,2,2,*^,*^}
490240: {P,^,P,^,2,*^,*^,*^,*^,*^,*^}	608304: {2,P,P,*^,2,*^,*^,*^,*^,*^,*^}
490242: {P,^,P,^,2,*^,*^,*^,*^,*^,*^}	608306: {2,P,P,*^,2,*^,*^,*^,*^,*^,*^}
490250: {P,^,P,^,2,*^,*^,*^,*^,2,2,*^,*^}	608314: {2,P,P,*^,2,*^,*^,*^,*^,2,2,*^,*^}
490282: {P,^,P,^,2,*^,*^,*^,*^,2,2,2,*^,*^}	608459: {2,P,P,*^,2,*^,*^,*^,*^,2,*^,*^}
490410: {P,^,P,^,2,*^,2,2,2,2,2,*^,*^}	609995: {2,P,P,*^,*^,2,*^,*^,*^,*^,*^}
496840: {P,^,2,P,P,*^,*^,*^,2,*^,*^,*^}	614448: {2,P,P,2,*^,*^,*^,*^,*^,*^,*^}
496864: {P,^,2,P,P,*^,*^,*^,2,*^,*^,*^}	614450: {2,P,P,2,*^,*^,*^,*^,*^,*^,*^}
496866: {P,^,2,P,P,*^,*^,*^,2,*^,*^,*^}	614458: {2,P,P,2,*^,*^,*^,*^,*^,2,2,*^,*^}
496872: {P,^,2,P,P,*^,*^,*^,2,2,*^,*^,*^}	614603: {2,P,P,2,*^,*^,*^,*^,*^,2,*^,*^}
496875: {P,^,2,P,P,*^,*^,*^,2,2,2,*^,*^,*^}	619143: {2,P,P,*^,*^,2,2,2,*^,*^,*^,*^}
504419: {P,^,2,*^,*^,2,P,2,*^,*^,*^,*^,*^}	619191: {2,P,P,*^,*^,2,2,2,*^,*^,*^,*^}
504443: {P,^,2,*^,*^,2,P,^,2,*^,*^,*^,*^,*^}	624423: {2,P,2,*^,*^,*^,*^,2,P,*^,*^,*^}
504467: {P,^,2,*^,*^,2,2,P,*^,*^,*^,*^,*^}	625171: {2,P,2,*^,2,2,*^,*^,*^,*^,*^,*^}
504551: {P,^,2,*^,*^,2,*^,2,*^,2,P,*^,*^,*^}	625363: {2,P,2,*^,2,2,*^,2,*^,*^,*^,*^,*^}
508619: {P,^,*^,*^,*^,2,*^,*^,2,*^,*^,*^,*^}	626311: {2,P,2,*^,*^,2,2,*^,*^,*^,*^,*^,*^}
517576: {P,^,*^,*^,2,P,P,*^,*^,*^,2,*^,*^,*^}	626359: {2,P,2,*^,*^,2,2,*^,*^,*^,*^,*^,*^}
517600: {P,^,*^,*^,2,P,P,*^,*^,*^,2,*^,*^,*^}	631075: {2,P,2,2,*^,*^,*^,*^,2,*^,*^,*^,*^}
517602: {P,^,*^,*^,2,P,P,*^,*^,*^,2,*^,*^,*^}	631099: {2,P,2,2,*^,*^,*^,*^,2,*^,*^,*^,*^}
517608: {P,^,*^,*^,2,P,P,*^,*^,*^,2,2,*^,*^,*^}	631171: {2,P,2,2,*^,*^,*^,*^,2,*^,*^,*^,*^}
517611: {P,^,*^,*^,2,P,P,*^,*^,*^,2,2,*^,*^,*^}	632899: {2,P,2,2,2,*^,*^,*^,*^,*^,*^,*^,*^}
529152: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	633667: {2,P,2,2,2,*^,*^,*^,*^,*^,*^,*^,*^}
529154: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	634147: {2,P,2,2,*^,*^,*^,*^,*^,*^,*^,*^,*^}
529162: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	634171: {2,P,2,2,*^,*^,*^,*^,*^,*^,*^,*^,*^}
529194: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	634243: {2,P,2,2,*^,*^,*^,*^,*^,*^,*^,*^,*^}
529322: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	643840: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
531595: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	643842: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
531632: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	643850: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
531634: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	643882: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
531642: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	644010: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
531979: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	646283: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
533808: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	646320: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
533810: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	646322: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
533818: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	646330: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
534627: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	646667: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
534651: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	648496: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
535059: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	648498: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
535251: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	648506: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
535395: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	649315: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
535419: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	649339: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
536199: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	649747: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
536247: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	649939: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
538992: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	650083: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
538994: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	650107: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
539002: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	650887: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
558859: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	650935: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
562224: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	653680: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
562226: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	653682: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
562234: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	653690: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
562379: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	657163: {2,2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
563731: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	661651: {2,2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
563923: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	661735: {2,2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
564871: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	661795: {2,2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
564919: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	661819: {2,2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
565651: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	661891: {2,2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
565735: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	662083: {2,2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
568723: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	663139: {2,2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
568807: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	663163: {2,2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
574087: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	663187: {2,2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
574135: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	663271: {2,2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
583115: {2,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	673575: {2,2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
592935: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	674323: {2,2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}
597543: {2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}	674515: {2,2,P,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^,*^}

20.8.3_VIRTUELLE OCRONS UND MATULA-GOEBEL ZAHLEN

Es gibt, wie wir im Kapitel 20.8.1 sahen, virtuelle OCRONS der Ordnung 0, 1 oder 2. Wir beobachteten, dass jede natürliche Zahl sich als virtuelles OCRON der Ordnung 0, 1 oder 2 darstellen läßt. Wir fassen noch einmal zusammen:

Virtuelle OCRONS der Ordnung 0:

Dies sind eigentlich keine virtuellen OCRONS sondern ganz normale OCRONS, die eine „*“-freie OCRON-Darstellung besitzen. Im Zahlenbereich von 1 bis 10000 gibt es 95 virtuelle OCRONS der Ordnung 0:

```
Mathematica (benötigt OCRON-Library):
numbersOrder0=Select[Range[10000],kVirt0Q]
{2,3,4,5,7,8,9,11,16,17,19,23,25,27,31,32,49,53,59,64,67,81,83,97,103,
121,125,127,128,131,227,241,243,256,277,289,311,331,343,361,419,431,50
9,512,529,563,625,661,691,709,719,729,739,961,1024,1331,1433,1523,1543
,1619,1787,1879,2048,2063,2187,2221,2309,2401,2437,2809,2897,3001,3125
,3481,3637,3671,3803,4091,4096,4489,4637,4913,4943,5189,5381,5441,5519
,5623,6561,6859,6889,7573,8161,8192,9409}
```

Dies sind alle Primzahlen und deren Potenzen, die sich als „*“-freie OCRONS darstellen lassen. Hier die Darstellungen der ersten 54 virtuellen OCRONS der Ordnung 0:

```
Mathematica (benötigt OCRON-Library):
ocronsOrder0={};

For[i=1,i<= 54,i++,
AppendTo[ocronsOrder0,nToVirtualOCRONLevel0[numbersOrder0[[i]]]];
Print["i= ",i," number= ",numbersOrder0[[i]],": ",ocronsOrder0[[i]]]];

(*virtual OCRONS order 0: Matula-Goebel numbers of generalized Bethe
trees *)
i= 1, number= 2: {2}
i= 2, number= 3: {2,P}
i= 3, number= 4: {2,2,^}
i= 4, number= 5: {2,P,P}
i= 5, number= 7: {2,2,^,P}
i= 6, number= 8: {2,2,P,^}
i= 7, number= 9: {2,P,2,^}
i= 8, number= 11: {2,P,P,P}
i= 9, number= 16: {2,2,2,^,^}
i= 10, number= 17: {2,2,^,P,P}
i= 11, number= 19: {2,2,P,^,P}
i= 12, number= 23: {2,P,2,^,P}
i= 13, number= 25: {2,P,P,2,^}
i= 14, number= 27: {2,P,2,P,^}
i= 15, number= 31: {2,P,P,P,P}
i= 16, number= 32: {2,2,P,P,^}
i= 17, number= 49: {2,2,^,P,2,^}
i= 18, number= 53: {2,2,2,^,^,P}
i= 19, number= 59: {2,2,^,P,P,P}
i= 20, number= 64: {2,2,^,2,P,^}
i= 21, number= 67: {2,2,P,^,P,P}
i= 22, number= 81: {2,P,2,2,^,^}
i= 23, number= 83: {2,P,2,^,P,P}
i= 24, number= 97: {2,P,P,2,^,P}
```

```

i= 25, number= 103: {2, P, 2, P, ^, P}
i= 26, number= 121: {2, P, P, P, 2, ^}
i= 27, number= 125: {2, P, P, 2, P, ^}
i= 28, number= 127: {2, P, P, P, P, P}
i= 29, number= 128: {2, 2, 2, ^, P, ^}
i= 30, number= 131: {2, 2, P, P, ^, P}
i= 31, number= 227: {2, 2, ^, P, 2, ^, P}
i= 32, number= 241: {2, 2, 2, ^, ^, P, P}
i= 33, number= 243: {2, P, 2, P, P, ^}
i= 34, number= 256: {2, 2, 2, P, ^, ^}
i= 35, number= 277: {2, 2, ^, P, P, P, P}
i= 36, number= 289: {2, 2, ^, P, P, 2, ^}
i= 37, number= 311: {2, 2, ^, 2, P, ^, P}
i= 38, number= 331: {2, 2, P, ^, P, P, P}
i= 39, number= 343: {2, 2, ^, P, 2, P, ^}
i= 40, number= 361: {2, 2, P, ^, P, 2, ^}
i= 41, number= 419: {2, P, 2, 2, ^, ^, P}
i= 42, number= 431: {2, P, 2, ^, P, P, P}
i= 43, number= 509: {2, P, P, 2, ^, P, P}
i= 44, number= 512: {2, 2, P, 2, ^, ^}
i= 45, number= 529: {2, P, 2, ^, P, 2, ^}
i= 46, number= 563: {2, P, 2, P, ^, P, P}
i= 47, number= 625: {2, P, P, 2, 2, ^, ^}
i= 48, number= 661: {2, P, P, P, 2, ^, P}
i= 49, number= 691: {2, P, P, 2, P, ^, P}
i= 50, number= 709: {2, P, P, P, P, P, P}
i= 51, number= 719: {2, 2, 2, ^, P, ^, P}
i= 52, number= 729: {2, P, 2, ^, 2, P, ^}
i= 53, number= 739: {2, 2, P, P, ^, P, P}
i= 54, number= 961: {2, P, P, P, P, 2, ^}

```

Diese Zahlen sind in der mathematischen Literatur nicht unbekannt und stammen aus der Graphentheorie:

Es sind die sogenannten Matula-Goebel Zahlen für verallgemeinerte „Bethe-Bäume“ (in Englisch: ‚Bethe trees‘).

Wieder einmal ergeben sich Zusammenhänge aus vollkommen unterschiedlichen mathematischen Disziplinen, in unserem Fall aus der Graphentheorie und der Primzahltheorie! Somit haben wir die Folge der virtuellen OCRONs vom Typ 0 identifiziert als die sogenannten Matula-Goebel-Zahlen, die bei der Untersuchung von verallgemeinerten Bethe-Bäumen (manchmal auch Bethe-Gitter genannt) auftauchen.

In oeis.org sind sie gelistet unter [A214577](https://oeis.org/A214577).

Virtuelle OCRONS der Ordnung 1:

Dies betrifft alle OCRONS, die sich als Produkt von unterschiedlichen „*“freien Faktoren schreiben lassen (somit alle zusammengesetzten Zahlen, die sich als Produkt von Matula-Goebel Zahlen (Zahlen die eine virtuelle OCRON-Ordnung 0 besitzen) schreiben lassen. Viele zusammengesetzte Zahlen lassen sich als virtuelle OCRONS der Ordnung 1 darstellen (im Bereich von 1 bis 100 sind dies 62 Zahlen):

```

Mathematica (benötigt OCRON-Library):
numbersOrder1=Select[Range[100],kVirt1Q]
{4, 6, 8, 9, 10, 12, 14, 15, 16, 18, 20, 21, 22, 24, 25, 27, 28, 30, 32, 33, 34, 35, 36, 38, 40, 42, 44, 45, 46, 48, 49, 50, 51, 54, 55, 56, 57, 60, 62, 63, 64, 66, 68, 69, 70, 72, 75, 76, 77, 80, 81, 84, 85, 88, 90, 92, 93, 95, 96, 98, 99, 100}

```

```
ocronsOrder1={};
For[i=1,i<= 62,i++,
AppendTo[ocronsOrder1,nToVirtualOCRONLevel1[numbersOrder1[[i]]]];
Print["i= ",i," number= ",numbersOrder1[[i]],": ",ocronsOrder1[[i]]];
```

```
i= 1, number= 4: {2,^,2}
i= 2, number= 6: {2,^,2,P}
i= 3, number= 8: {2,^,2,2,^}
i= 4, number= 9: {2,P,^,2,P}
i= 5, number= 10: {2,^,2,P,P}
i= 6, number= 12: {2,2,^,^,2,P}
i= 7, number= 14: {2,^,2,2,^,P}
i= 8, number= 15: {2,P,^,2,P,P}
i= 9, number= 16: {2,2,^,^,2,2,^}
i= 10, number= 18: {2,^,2,P,2,^}
i= 11, number= 20: {2,2,^,^,2,P,P}
i= 12, number= 21: {2,P,^,2,2,^,P}
i= 13, number= 22: {2,^,2,P,P,P}
i= 14, number= 24: {2,2,P,^,^,2,P}
i= 15, number= 25: {2,P,P,^,2,P,P}
i= 16, number= 27: {2,P,^,2,P,2,^}
i= 17, number= 28: {2,2,^,^,2,2,^,P}
i= 18, number= 30: {2,^,2,P,^,2,P,P}
i= 19, number= 32: {2,2,^,^,2,2,P,^}
i= 20, number= 33: {2,P,^,2,P,P,P}
i= 21, number= 34: {2,^,2,2,^,P,P}
i= 22, number= 35: {2,P,P,^,2,2,^,P}
i= 23, number= 36: {2,2,^,^,2,P,2,^}
i= 24, number= 38: {2,^,2,2,P,^,P}
i= 25, number= 40: {2,2,P,^,^,2,P,P}
i= 26, number= 42: {2,^,2,P,^,2,2,^,P}
i= 27, number= 44: {2,2,^,^,2,P,P,P}
i= 28, number= 45: {2,P,2,^,^,2,P,P}
i= 29, number= 46: {2,^,2,P,2,^,P}
i= 30, number= 48: {2,2,2,^,^,^,2,P}
i= 31, number= 49: {2,2,^,^,2,2,^,P}
i= 32, number= 50: {2,^,2,P,P,2,^}
i= 33, number= 51: {2,P,^,2,2,^,P,P}
i= 34, number= 54: {2,^,2,P,2,P,^}
i= 35, number= 55: {2,P,P,^,2,P,P,P}
i= 36, number= 56: {2,2,P,^,^,2,2,^,P}
i= 37, number= 57: {2,P,^,2,2,P,^,P}
i= 38, number= 60: {2,2,^,^,2,P,^,2,P,P}
i= 39, number= 62: {2,^,2,P,P,P,P}
i= 40, number= 63: {2,P,2,^,^,2,2,^,P}
i= 41, number= 64: {2,2,P,^,^,2,2,P,^}
i= 42, number= 66: {2,^,2,P,^,2,P,P,P}
i= 43, number= 68: {2,2,^,^,2,2,^,P,P}
i= 44, number= 69: {2,P,^,2,P,2,^,P}
i= 45, number= 70: {2,^,2,P,P,^,2,2,^,P}
i= 46, number= 72: {2,2,P,^,^,2,P,2,^}
i= 47, number= 75: {2,P,^,2,P,P,2,^}
i= 48, number= 76: {2,2,^,^,2,2,P,^,P}
i= 49, number= 77: {2,2,^,^,2,P,P,P}
i= 50, number= 80: {2,2,2,^,^,^,2,P,P}
i= 51, number= 81: {2,P,2,^,^,2,P,2,^}
i= 52, number= 84: {2,2,^,^,2,P,^,2,2,^,P}
i= 53, number= 85: {2,P,P,^,2,2,^,P,P}
i= 54, number= 88: {2,2,P,^,^,2,P,P,P}
i= 55, number= 90: {2,^,2,P,2,^,^,2,P,P}
```

```

i= 56, number= 92: {2,2,^,^,2,P,2,^,P}
i= 57, number= 93: {2,P,^,2,P,P,P,P}
i= 58, number= 95: {2,P,P,^,2,2,P,^,P}
i= 59, number= 96: {2,2,P,P,^,^,2,P}
i= 60, number= 98: {2,^,2,2,^,P,2,^}
i= 61, number= 99: {2,P,2,^,^,2,P,P,P}
i= 62, number= 100: {2,2,^,^,2,P,P,2,^}

```

Virtuelle OCRONs der Ordnung 1 können leicht konstruiert werden, in dem wir den Produktoperator “ \wedge ” definieren (der genauso aussieht wie der OCRON Potenzoperator „ \wedge “ und das Produkt mit Hilfe von Faktoren, die aus Matula-Goebel Zahlen bestehen, hinschreiben. Hier ein Beispiel: $6 = 2 * 3$:

2 (Produktoperator) $2P \rightarrow „2\wedge 2P“$.

Ein weiteres Beispiel: $90 = 2 * 3^2 * 5$:

$90 = 2\wedge 2P2\wedge 2PP$

Ein paar Zahlen lassen sich weder als virtuelle OCRONs der Ordnung 0 noch der Ordnung 1 darstellen. Im Bereich von 1 bis 100 sind dies 23 Zahlen:

```

Mathematica (benötigt OCRON-Library):
strangeNumbers=Select[Range[100],kVirt0Q[#]==False && kVirt1Q[#]
==False&]
{13,26,29,37,39,41,43,47,52,58,61,65,71,73,74,78,79,82,86,87,89,91,94}

```

Dies sind alle Zahlen, die in ihrer Primfaktorzerlegung eine der „*-enthaltenden“ Primzahlen (13,29,37,41,43, ... usw.) enthalten.

Die OCRON-Bibliothek enthält die Funktion `findMatulaProduct[]`, mit der Zahlen in Matula-Faktoren zerlegt werden können.

Manche Zahlen lassen sich sowohl als virtuelle OCRONs der Ordnung 0 als auch der Ordnung 1 darstellen (dies sind Potenzen von Matula-Goebel-Zahlen).

Virtuelle OCRONs der Ordnung 2:

Dies betrifft alle OCRONs, die sich als Summe von unterschiedlichen „*“freien Summanden schreiben lassen (und somit alle natürlichen Zahlen, da sich alle natürlichen Zahlen (genau genommen ≥ 4) als Summe von Matula-Goebel Zahlen schreiben lassen). Jede natürliche Zahl läßt sich als virtuelles OCRON der Ordnung 2 darstellen. Viele Zahlen lassen sich sogar als Summe nur zweier Matula-Summanden schreiben. Drei Matula-Summanden sind jedoch in jedem Fall ausreichend (der Leser wird inzwischen bemerkt haben, dass dieses Buch fast keine Beweise enthält, deshalb steht der Beweis noch aus...). Hier ein paar Beispiele für Zahlen, die in drei Matula-Summanden zerlegt werden müssen, um als virtuelles OCRON der Ordnung 2 dargestellt zu werden (unter 100 gibt es genau 4 solche Zahlen):

```

Mathematica (benötigt OCRON-Library):

For[i=1,i<=100,i++,mgSum=findMatulaGoebelSum[i];
If[Length[mgSum]> 2,Print["i= ",i," : ",mgSum]]];
i= 45: {32,11,2}
i= 77: {67,8,2}
i= 79: {67,9,3}
i= 93: {83,8,2}

```


Es scheint gleichzeitig auch der letzte Mersenne Primzahlexponent zu sein, der sich als virtuelles OCRON der Ordnung 0 darstellen lässt. Alle danach kommenden Werte besitzen nur Darstellungen der Ordnung 2!

Hinweis:

Das additive Prinzip bei der Konstruktion virtueller OCRONs der Ordnung 2 erlaubt es vermutlich auch, eine Zahl in Matula-Summanden zu zerlegen, die nur aus Zweierpotenzen bestehen, wobei die Potenzen wiederum rekursiv auch nur aus Summen von Zweier-Potenzen bestehen sollen. Dies würde zu einer Representation virtueller OCRONs der Ordnung 2 führen bei denen das P-Symbol nicht mehr vorkommt.

Eine reine Zahlendarstellung mit Hilfe der Symbole 2 und ^!

Der Autor hat diesen Gedanken nicht weiter verfolgt, da die ganze Sache vermutlich nur auf eine ‚verklausulierte‘ Binärdarstellung hinausläuft.

Virtuelle OCRONs der Ordnung 3:

Eigentlich könnten uns virtuelle OCRONs der Ordnung 0,1 und 2 genügen, da wir mit ihnen alle natürlichen Zahlen darstellen können. Man kann die ganze Sache jedoch noch weiter treiben und virtuelle OCRONs dritter Ordnung untersuchen. Solche virtuellen OCRONs sind praktisch nicht mehr numerisch auswertbar, da sehr schnell utopisch hohe Zahlen entstehen. Wir wollen dennoch einen Blick darauf werfen. Analog zu den Ordnungen 1 und 2 definieren wir einen virtuellen Operator: „ 22 “, den wir ‚Jacobi-Additionsoperator‘ nennen wollen, da sich herausstellt, dass er im äquivalenten Zahlenbereich der sogenannten ‚Jacobi-Addition‘ entspricht. Unternehmen wir einen kurzen Ausflug in die Welt der Jacobi-Arithmetik:

Mathematica:

```
CirclePlus[a_,b_]:=
Quiet[FullSimplify[Distribute[PowerExpand[Log[2^a+2^b]/Log[2]]]]]
CircleMinus[a_,b_]:=CirclePlus[a,(b+(I \[Pi])/Log[2])]
```

Beispiele:

Mathematica:

$3 \oplus 3$ ergibt 4, $4 \oplus 4$ ergibt 5, $x \oplus x$ ergibt $x + 1$

Eine Zahl, mit sich selbst ‚Jacobi-addiert‘, inkrementiert diese Zahl um 1.

$x \oplus x == x + 1$ ergibt True

$((x \oplus x) \oplus x) \oplus x == x + 2$ ergibt True

$(((((x \oplus x) \oplus x) \oplus x) \oplus x) \oplus x) \oplus x == x + 3$ ergibt True

Eine Zahl x , $2^n - 1$ mal mit sich selbst ‚Jacobi-addiert‘ berechnet sich zu $x + n$.

Man könnte die Jacobi-Addition in der Hierarchie der Rechenarten eine Ebene niedriger als die Addition ansiedeln, da die Addition aus der Jacobi-Addition durch mehrmaliges sukzessives Anwenden der Jacobi-Addition hervorgeht. Betrachten wir $x \oplus x$ rekursiv, dann ergibt das rekursive n -malige Anwenden von $x = x \oplus x$ den Wert $x + n$.

Folglich ist die Jacobi-Addition tatsächlich die ‚Vorgänger‘-Rechenart der Addition, ähnlich wie die normale Addition der ‚Vorgänger‘ der Multiplikation ist, da die Multiplikation aus der Addition durch wiederholtes Anwenden hervorgeht.

Die Jacobi-Addition ist assoziativ, kommutativ, und sie hat ein neutrales Element. Es gibt auch zu jedem Element ein Inverses so dass die Jacobi-Addition das neutrale Element

ergibt. Auch das Distributivgesetz ist gültig. Die Jacobi-Addition bildet somit zusammen mit der Addition einen Körper bezüglich der Menge der ganzen Zahlen. Hier ein Beweis der Körpereigenschaft, sowie ein paar Rechengesetze, mit Hilfe von Mathematica:

```
(* Properties of the Jacobi-Addition *)
ClearSystemCache[];
ClearAll[a];ClearAll[b];ClearAll[c];Clear[CirclePlus];
Clear[CirclePlusN];
ld[x_]:=Log[2,x]
f0[a_,b_]:=FullSimplify[PowerExpand[ld[ld[ld[(2^2^2^a)^2^2^b]]]]] (*Jacobi-Addition*)
CirclePlus[a_,b_]:=Quiet[FullSimplify[Distribute[PowerExpand[Log[2^a+2^b]/Log[2]]]]] (*Jacobi-Addition*)
CircleMinus[a_,b_]:=CirclePlus[a,(b+(I*Pi)/Log[2])]
simplify[term_]:=Distribute[PowerExpand[term]]

(*Assoziativität: (a ⊕ b) ⊕ c == a ⊕ (b ⊕ c) *)
(a ⊕ b) ⊕ c == a ⊕ (b ⊕ c)
(*Kommutativitaet: (a ⊕ b) == (b ⊕ a) *)
(a ⊕ b) == (b ⊕ a)
(*Distributivgesetz : (a ⊕ b) + c == (a + c) ⊕ (b + c)*)
(a ⊕ b) + c == simplify[(a + c) ⊕ (b + c)]
(*Neutrales Element *)
Print["neutrales Element: ",-Infinity];
(*Verknuepfung von a mit dem neutralen Element a ⊕ (-∞) *)
a ⊕ (-∞)
(*Inverses Element von a: *)
Print["Inverses Element von a: "];
simplify[Log[2,-2^a]]
(*Verknüpfung von a mit dem inversen Element von a gibt das neutrale Element: *)
Print["Verknüpfung von a mit dem inversen Element von a: "];
(a +  $\frac{i\pi}{\text{Log}[2]}$ ) ⊕ a
a ⊕ (a +  $\frac{i\pi}{\text{Log}[2]}$ )
(* n-maliges Anwenden des Operators ⊕ a ergibt das gleiche wie a+ld[n+1] *)
simplify[a ⊕ a]
simplify(a ⊕ a) ⊕ a]
simplify((a ⊕ a) ⊕ a) ⊕ a]
simplify(((a ⊕ a) ⊕ a) ⊕ a) ⊕ a]
simplify((((a ⊕ a) ⊕ a) ⊕ a) ⊕ a) ⊕ a]
(*Potenzrechengesetze: 2^(a ⊕ b) ergibt:*)
2a⊕b
(*Potenzrechengesetze: 2^(a ⊖ b) ergibt:*)
2a⊖b
(*Weitere Formeln:*)
(*a ⊖ a ergibt:*)
a ⊖ a
(*Neues Logarithmisches Additionsgesetz: ld[a] ⊕ ld[b]: *)
(*ergibt ld[a] ⊕ ld[b] *)
Log[2,a] ⊕ Log[2,b]
```

Ausgabe des Mathematica-Programmes :

Assoziativität : $(a \oplus b) \oplus c == a \oplus (b \oplus c)$

True

Kommutativität: $(a \oplus b) == (b \oplus a)$

True

Distributivität: $(a \oplus b) + c == (a + c) \oplus (b + c)$

True

Neutrales Element: $-\infty$

Verknüpfung von a mit dem neutralen Element $a \oplus (-\infty)$:

a

Inverses Element von a:

$$a + \frac{i\pi}{\text{Log}[2]}$$

Verknüpfung von a mit dem inversen Element von a:

$-\infty$

n-maliges Anwenden des Operators $\oplus a$ ergibt das gleiche wie $a + \text{ld}(n+1)$:

$$1 + a, a + \frac{\text{Log}[3]}{\text{Log}[2]}, 2 + a, a + \frac{\text{Log}[5]}{\text{Log}[2]}, 3 + a$$

Potenzrechengesetze: $2^{(a \oplus b)}$ ergibt:

$$2^a + 2^b$$

Potenzrechengesetze: $2^{(a \ominus b)}$ ergibt:

$$2^a - 2^b$$

$a \ominus a$ ergibt:

$-\infty$

Zurück zu unseren virtuellen OCRONs dritter Ordnung.

Virtuelle OCRONs der Ordnung 3 können leicht konstruiert werden, in dem wir den Jacobi-Summenoperator „**^^^22**“ definieren (der genauso aussieht wie die OCRON-Kette „**^^^22**“) und das OCRON mit Hilfe von Jacobi-Summanden, die aus Matula-Goebel Zahlen bestehen, hinschreiben. Hier ein Beispiel:

$3 = 2 \oplus 2$. Nun ergibt **2** (Jacobi-Summenoperator) **2**: „**2^^^222**“.

Somit haben wir für die Zahl 3 ein virtuelles OCRON der Ordnung 3 durch Zerlegung der Zahl 3 in die zwei Jacobi-Matula-Summanden 2 und 2 gefunden: „**2^^^222**“.

Ein weiteres Beispiel:

$$7 = ((5 \oplus 5) \oplus 5) \oplus 5$$

Unser virtuelles Ocron der Ordnung 3 für die Zahl 7 lautet dann:

2PP^^^222PP^^^222PP^^^222PP

Die traditionelle Form Zahl 7 in virtueller Ordnung-2-OCRON-Darstellung lautet:

Log[2, Log[2, (2^(2^2))^(2^Prime[Prime[2]])]] (Mathematica Input-Form)

log₂(log₂((2^{2²})^{2^{p²}})) (Traditional Form)

Der Leser möge die entsprechende traditionelle Darstellung der Zahl 7 für das virtuelle Ordnung-3-OCRON herausfinden!

Das Ganze von der anderen Seite her betrachtet:

Gibt es zu jeder Zeichenkette, die aus den Symbolen „2“, „^“ und „P“ besteht (die auf den ersten Blick ein virtuelles OCRON zu sein scheint), eine entsprechende Zahl?

Wir finden, daß nicht jede derartige Zeichenkette ein ‚wohlgeformtes‘ virtuelles OCRON darstellt.

Die Menge der ‚wohlgeformten‘, interpretierbaren virtuellen OCRONs läßt sich mit unserer OCRON-Bibliothek leicht herausfinden.

Hier ein Beispiel der ersten 50 ‚wohlgeformten‘ interpretierbaren virtuellen OCRONs:

```
Mathematica (benötigt OCRON-Library):
list=createAscendingVirtualOcron4List[50]
Output:
{{2}, {P, 2}, {2, P}, {P, P, 2}, {P, 2, P}, {2, ^, 2}, {2, P, P}, {2, 2, ^}, {P, ^, 2, 2}, {P,
P, P, 2}, {P, P, 2, P}, {P, 2, ^, 2}, {P, 2, P, P}, {P, 2, 2, ^}, {2, ^, P, 2}, {2, ^, 2, P}, {2,
P, ^, 2}, {2, P, P, P}, {2, P, 2, ^}, {2, 2, ^, P}, {2, 2, P, ^}, {P, ^, P, 2, 2}, {P, ^, 2, P, 2},
{P, ^, 2, 2, P}, {P, P, ^, 2, 2}, {P, P, P, P, 2}, {P, P, P, 2, P}, {P, P, 2, ^, 2}, {P, P, 2, P,
P}, {P, P, 2, 2, ^}, {P, 2, ^, P, 2}, {P, 2, ^, 2, P}, {P, 2, P, ^, 2}, {P, 2, P, P, P}, {P, 2, P,
2, ^}, {P, 2, 2, ^, P}, {P, 2, 2, P, ^}, {2, ^, ^, 2, 2}, {2, ^, P, P, 2}, {2, ^, P, 2, P}, {2, ^,
2, ^, 2}, {2, ^, 2, P, P}, {2, ^, 2, 2, ^}, {2, P, ^, P, 2}, {2, P, ^, 2, P}, {2, P, P, ^, 2}, {2,
P, P, P, P}, {2, P, P, 2, ^}, {2, P, 2, ^, P}, {2, P, 2, P, ^}}
```

Diese Liste enthält, bunt gemischt, die ersten 50 interpretierbaren virtuellen OCRONs der Ordnung 0, 1 und 2.

Schauen wir uns die resultierenden Zahlenwerte dieser virtuellen OCRONs an, so finden wir, daß die sich ergebenden Zahlen nicht immer Natürliche Zahlen sind, sondern auch Terme mit Zweierlogarithmen enthalten können.

Hinweis: Die erste Spalte enthält den (im Dreier-System) ‚Gödelisierten‘ Zahlenwert des virtuellen OCRONs; die zweite Spalte das virtuelle OCRON, die dritte Spalte den interpretierten Wert zusammen mit seiner virtuellen Ordnung (0, 1 oder 2):

```
(* the first 50 virtual OCRONs, (needs OCRON-Library) *)
For[i=1,i<=50,i++,
val=virtualOCRON4ToN[list[[i]]];
Print[goedelSymbolListToNForVirtualOCRONS[list[[i]],": ",list[[i]],":
",val];]
```

```
2: {2}: {2,0}
5: {P,2}: {Log[9]/Log[2],1}
7: {2,P}: {3,0}
14: {P,P,2}: {Log[25]/Log[2],1}
16: {P,2,P}: {Log[27]/Log[2],1}
20: {2,^,2}: {4,1}
22: {2,P,P}: {5,0}
24: {2,2,^}: {4,0}
35: {P,^,2,2}: {Log[12]/Log[2],2}
41: {P,P,P,2}: {Log[121]/Log[2],1}
43: {P,P,2,P}: {Log[125]/Log[2],1}
```

47: {P, 2, ^, 2}: {Log[81]/Log[2], 1}
 49: {P, 2, P, P}: {Log[243]/Log[2], 1}
 51: {P, 2, 2, ^}: {Log[81]/Log[2], 1}
 59: {2, ^, P, 2}: {Log[49]/Log[2], 1}
 61: {2, ^, 2, P}: {6, 1}
 65: {2, P, ^, 2}: {6, 1}
 67: {2, P, P, P}: {11, 0}
 69: {2, P, 2, ^}: {9, 0}
 73: {2, 2, ^, P}: {7, 0}
 75: {2, 2, P, ^}: {8, 0}
 98: {P, ^, P, 2, 2}: {Log[Log[130321]/Log[2]]/Log[2], 2}
 104: {P, ^, 2, P, 2}: {Log[27]/Log[2], 2}
 106: {P, ^, 2, 2, P}: {Log[24]/Log[2], 2}
 116: {P, P, ^, 2, 2}: {Log[20]/Log[2], 2}
 122: {P, P, P, P, 2}: {Log[961]/Log[2], 1}
 124: {P, P, P, 2, P}: {Log[1331]/Log[2], 1}
 128: {P, P, 2, ^, 2}: {Log[625]/Log[2], 1}
 130: {P, P, 2, P, P}: {Log[3125]/Log[2], 1}
 132: {P, P, 2, 2, ^}: {Log[625]/Log[2], 1}
 140: {P, 2, ^, P, 2}: {Log[529]/Log[2], 1}
 142: {P, 2, ^, 2, P}: {Log[729]/Log[2], 1}
 146: {P, 2, P, ^, 2}: {Log[729]/Log[2], 1}
 148: {P, 2, P, P, P}: {Log[177147]/Log[2], 1}
 150: {P, 2, P, 2, ^}: {Log[19683]/Log[2], 1}
 154: {P, 2, 2, ^, P}: {Log[2187]/Log[2], 1}
 156: {P, 2, 2, P, ^}: {Log[6561]/Log[2], 1}
 170: {2, ^, ^, 2, 2}: {4, 2}
 176: {2, ^, P, P, 2}: {Log[289]/Log[2], 1}
 178: {2, ^, P, 2, P}: {Log[343]/Log[2], 1}
 182: {2, ^, 2, ^, 2}: {8, 1}
 184: {2, ^, 2, P, P}: {10, 1}
 186: {2, ^, 2, 2, ^}: {8, 1}
 194: {2, P, ^, P, 2}: {Log[361]/Log[2], 1}
 196: {2, P, ^, 2, P}: {9, 1}
 200: {2, P, P, ^, 2}: {10, 1}
 202: {2, P, P, P, P}: {31, 0}
 204: {2, P, P, 2, ^}: {25, 0}
 208: {2, P, 2, ^, P}: {23, 0}
 210: {2, P, 2, P, ^}: {27, 0}

Die Zahlenwerte der ersten Spalte (unserer Terminologie entsprechend: virtuelle GOCRONs) können wir auch einfacher bekommen:

```
(* the first 50 virtual 'wellformed' GOCRONs, (needs OCRON-Library) *)
Select[Range[2, 210], checkVirtualOCRON4[nToGoedelSymbolListForVirtualOCRONS[#]]==True&]
```

```
{2, 5, 7, 14, 16, 20, 22, 24, 35, 41, 43, 47, 49, 51, 59, 61, 65, 67, 69, 73, 75, 98, 104, 106, 116, 122, 124, 128, 130, 132, 140, 142, 146, 148, 150, 154, 156, 170, 176, 178, 182, 184, 186, 194, 196, 200, 202, 204, 208, 210}
```

Diese Folge ist bekannt (*oeis.org: A031457*) und kann auch wie folgt berechnet werden:

```
Select[Range[210], DigitCount[#, 3, 0]==DigitCount[#, 3, 2]-1&]
```

Das ist nicht überraschend, da die Folge einfach alle Zahlen enthält, deren Anzahl von '0'-en (im Dreiersystem dargestellt) genau um 1 kleiner ist als die Anzahl von '2'-en.

Genau dies trifft auch bei unseren virtuellen OCRONs zu, da wir folgende ‚Gödelisierung‘ gewählt haben:
 $2 = 2$, $P = 1$, $\wedge = 0$. Da \wedge ein binärer Operator ist (also 2 Operanden benötigt) ist die Gesamtanzahl von \wedge -Symbolen in einer ‚wohlgeformten‘ virtuellen OCORN immer um 1 kleiner als die Anzahl von 2 -en.

Beschränken wir uns auf solche virtuellen OCRONs, die bei Interpretation nur natürliche Zahlen ergeben, dann erhalten wir im Bereich zwischen 1 und 100 genau 11 Werte:

```
(* the first virtual 'wellformed' (G)OCRONS in the Range up to 100,
which yield natural numbers (needs OCRON-Library) *)
createIntValuesListFromAscendingVirtualOcron4s[100,2,100]
```

Hinweis: Das OCRON nach \rightarrow ist die (in das normale OCRON-Format) expandierte Version des entsprechenden virtuellen OCRONs.

```
0 i= 2: {2}->{2}
1 i= 7: {2,P}->{2,P}
2 i= 20: {2,^,2}->{2,2,^,2,^}
3 i= 22: {2,P,P}->{2,P,P}
4 i= 24: {2,2,^}->{2,2,^}
5 i= 61: {2,^,2,P}->{2,2,^,2,P,^}
6 i= 65: {2,P,^,2}->{2,2,P,^,2,^}
7 i= 67: {2,P,P,P}->{2,P,P,P}
8 i= 69: {2,P,2,^}->{2,P,2,^}
9 i= 73: {2,2,^,P}->{2,2,^,P}
10 i= 75: {2,2,P,^}->{2,2,P,^}
```

Wenn wir uns nur die Gödel-Werte der virtuellen OCRONs ansehen möchten, geht es auch schneller (z. B. für den Bereich bis 210, s. oben):

```
Select[Range[2,210],IntegerQ[(virtualGOCRONTon[#])[[1]]]==True&&(virtualGOCRONTon[#])[[1]]>0&]

{2,7,20,22,24,61,65,67,69,73,75,170,182,184,186,196,200,202,204,208,210}
```

20.10.5_NULLSTELLEN DER RAMANUJANSCHEN TAU-L-FUNKTION $L(s)$

Tabelle: Die ersten 128 Nullstellen der Ramanujanschen Tau-L-Funktion entlang der kritischen Geraden $\text{Re}(s) = 6$ mit einer Genauigkeit von 40 Ziffern

n	n-te Nullstelle (Imaginärteil)
1	9.222379399921084797142611932940781116486
2	13.907549861392134005200205137953162193298
3	17.44277697823447326186396821867674589157
4	19.65651314195496013326192041859030723572
5	22.33610363720986669022749993018805980682
6	25.27463654811236537511831556912511587143
7	26.80439115835040198021488322410732507706
8	28.83168262418687532999683753587305545807
9	31.17820949836026045431935926899313926697
10	32.77487538223120822067357948981225490570
11	35.19699584121007518433543737046420574188
12	36.74146297671030936271563405171036720276
13	37.75391597562427392631434486247599124908
14	40.21903437422132299161603441461920738220
15	41.73049228930784693147870711982250213623
16	43.59174123557517077642842195928096771240
17	45.04007921377559853226557606831192970276
18	46.19731875314330693527153925970196723938
19	48.35905247802367057374794967472553253174
20	49.27605353655818021252343896776437759399
21	51.15656028143634870275491266511380672455
22	53.06671423542580612320307409390807151794
23	54.09995263156227451872837264090776443481
24	55.21778745348462535957878571934998035431
25	56.71529404472536839421081822365522384644
26	58.58016100791407154702028492465615272522
27	59.78593800331714191997889429330825805664
28	61.13672295792679989290263620205223560333
29	62.66499232630715710001823026686906814575
30	64.08664571892624906013224972411990165710
31	64.84864127982825721119297668337821960449
32	66.49476926718958225137612316757440567017
33	67.93860977475046070139796938747167587280
34	69.04339787488993351871613413095474243164
35	71.11465341424647590429231058806180953979
36	71.74750419616562169267126591876149177551
37	72.81406066758940198724303627386689186096
38	74.09582544001794701671315124258399009705
39	75.77216168976411836410989053547382354736
40	77.10183189348964560849708504974842071533
41	77.68461125026033187168650329113006591797
42	79.79293909123566663765814155340194702148
43	80.56019206809750698994321282953023910522
44	82.00757620451852858423080760985612869263
45	82.84252583957207605180883547291159629822
46	83.97564035576498042701132362708449363708
47	85.46221814858006382564781233668327331543
48	86.75597218825528500474320026114583015442
49	88.07513099425673885889409575611352920532

50	89.02289034074360074555443134158849716187
51	90.45103289616260155980853596702218055725
52	91.11271853147249544235819485038518905640
53	92.44292549472127973331225803121924400330
54	93.76912394743676770758611382916569709778
55	95.13807853977348827356763649731874465942
56	95.62492107704515831301250727847218513489
57	97.34104088984686597996187629178166389465
58	98.70980408818076057286816649138927459717
59	99.74664890030413744170800782740116119385
60	100.22461499968198950227815657854080200195
61	101.34359353371037570923363091424107551575
62	103.16663591563629154279624344781041145325
63	103.81733899744642712903441861271858215332
64	105.22181333799052538324758643284440040588
65	106.29382213420061020769935566931962966919
66	107.42670755392653347826126264408230781555
67	108.47543790163686594496539328247308731079
68	109.39169607602677558588766260072588920593
69	110.70966268400202636712492676451802253723
70	111.53473540163911081890546483919024467468
71	112.75715359897023404300853144377470016479
72	113.84343404772059216156776528805494308472
73	115.06276556053481385788472834974527359009
74	116.46348398369597987311863107606768608093
75	117.11654084727238966934237396344542503357
76	118.14687073684822848917974624782800674438
77	119.08216779664660123216890497133135795593
78	119.99454209523629799605259904637932777405
79	121.78633067852094029603904346004128456116
80	122.55731782502655846656125504523515701294
81	123.21241716312161429414118174463510513306
82	124.60624049116798062186717288568615913391
83	125.94289344930038510028680320829153060913
84	126.75939204586923381157248513773083686829
85	127.55580316015350206271250499412417411804
86	128.62383894451065202702011447399854660034
87	129.60342208412549780405242927372455596924
88	130.94859240739617689541773870587348937988
89	131.70819904811898481966636609286069869995
90	132.96854278614409849978983402252197265625
91	134.34729668877156427697627805173397064209
92	135.07869588873938937467755749821662902832
93	135.55289998752846258867066353559494018555
94	137.09033471100445922274957410991191864014
95	137.70022292031720212435175199061632156372
96	139.28400855168445104936836287379264831543
97	139.93658439005704963165044318884611129761
98	140.89653322681010649830568581819534301758
99	142.1411519890185388703685021027922630310
100	143.0835552634784448855498339980840682983
101	144.3547263694031244085635989904403686523
102	145.1653120064068502870213706046342849731
103	146.1487705718024301404511788859963417053

104	146.4097883646259958823065971955657005310
105	148.1177541226128084872470935806632041931
106	149.0412678815713718449842417612671852112
107	150.2750742969780901603371603414416313171
108	150.9064237539794532949599670246243476868
109	152.1344343784803641028702259063720703125
110	153.1151471940314081621181685477495193481
111	154.0518290966241181649820646271109580994
112	154.7953122295758987547742435708642005920
113	155.7320793911374607887410093098878860474
114	157.0957831922944762936822371557354927063
115	157.9127528865146530279162107035517692566
116	158.6608139225808713490550871938467025757
117	159.6686139103367452207749010995030403137
118	161.3063702811864743580372305586934089661
119	161.8503586051299976134032476693391799927
120	162.8714549225416021727141924202442169189
121	163.5474941087671822970150969922542572021
122	164.3389052284337310538830934092402458191
123	165.6101228957916760009538847953081130981
124	166.5807970056847295836632838472723960876
125	167.6436347091075731441378593444824218750
126	168.6591247847260888192977290600538253784
127	169.2457741065447009987110504880547523499
128	170.5979320487521135873976163566112518311

Mathematica-Programm: Bitte den Autor kontaktieren.

```
startValues={9,14,17.5,20,22.5,25,27,29,31,33,35,37,38,40,42,44,45,46,48,49,51,53,54,55,57,58.5,60,61,62.5,64,65,66.5,68,69,71,72,73,74,76,77,78,80,80.5,82,83,84,85.5,87,88,89,90.5,91,92.5,94,95,95.5,97.5,98.5,99.5,100.5,101,103,104,105,106,107.5,108.5,109.5,110.6,111.5,112.8,114,115,116.5,117,118,119,120,122,122.5,123.2,124.6,126,126.8,127.6,128.6,129.6,131,131.7,133,134.4,135,135.6,137,137.6,139.2,140,141,142.1,143.1,144.4,145.1,146.1,146.5,148,149,150.2,151,152,153,154,154.9,155.8,157.1,158,158.7,159.6,161.3,161.9,162.9,163.5,164.3,165.6,166.6,167.6,168.7,169.3,170.5};
```

```
tabZeros=SetPrecision[Table[{n,FindRoot[RamanujanTauL[6+I*t]},{t,startValues[[n]]}],AccuracyGoal->40,PrecisionGoal->40}],{n,1,Length[startValues]},40];
ramanujanTauLZerosTab128=Table[Re[(tabRZeros[[n,2]])[[1,2]]],{n,1,Length[tabRZeros]}]
ramanujanTauLZerosTab128=Table[Re[(tabZeros[[n,2]])[[1,2]]],{n,1,Length[tabZeros]}]
```

Die Anwendung der Eulerschen Produktdarstellung für Ramanujans Tau-L-Funktion $L(s)$ entlang der kritischen Geraden mit Realteil 6 liefert ausgeprägte Minima an den Positionen der ersten 59 Nullstellen. Die Ähnlichkeit der Situation zu den Berechnungen für die Riemannsche Zetafunktion im Kapitel 5.3 ist offensichtlich!

Produktdarstellung für Ramanujans Tau-L-Funktion $L(s)$ entlang der kritischen Geraden:

$$L(6 + t \cdot i) = \prod_{p \in \mathbb{P}} \frac{1}{1 - \tau(p)p^{-6-t \cdot i} + p^{11-2(6+t \cdot i)}}, \quad t > 0$$

Ergibt folgende Darstellung (unter Verwendung der ersten 128 Primzahlen):

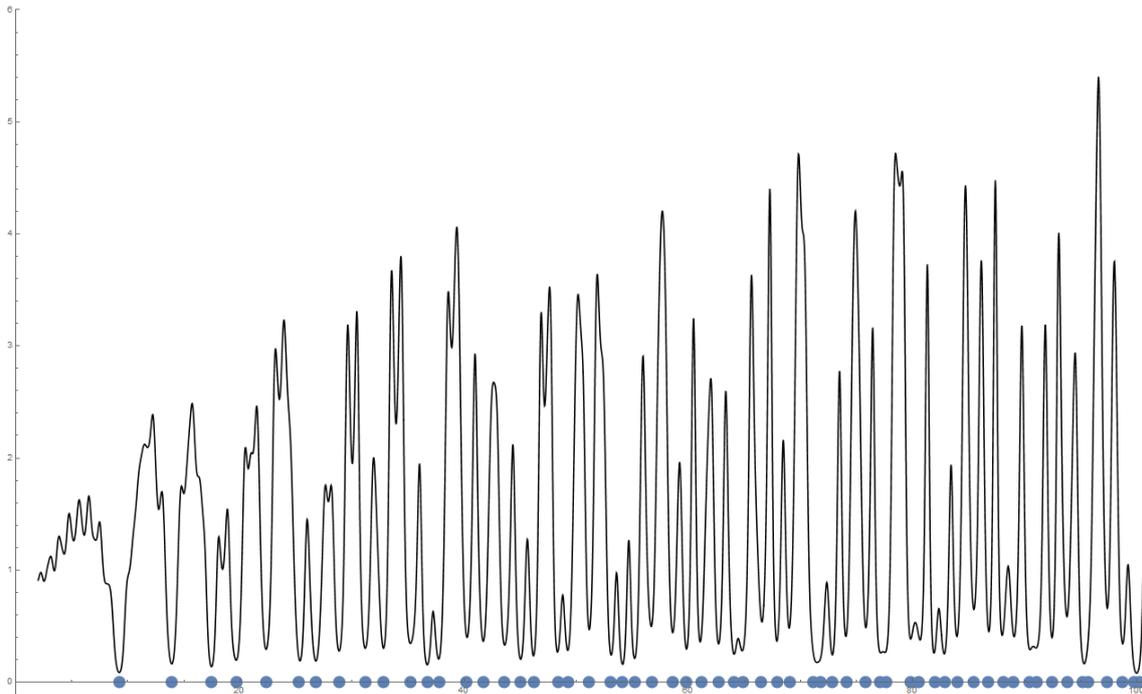


Abbildung: Ramanujan-Tau-L-Funktion entlang der krit. Geraden, berechnet mit der Produktformel im Bereich 0 bis 100, Minima an den Nullstellen

Mathematica-Programm (bitte den Autor kontaktieren):

```
crterm[n_,x_]:=1/(1-RamanujanTau[Prime[n]]*(Prime[n])^(-
(6+x*I))+Prime[n]^(11.0-2*(6+x*I)));
myRFunc[x_]:=Product[crterm[n,x],{n,1,128}]
xmax=101;
Show[ListPlot[Table[{ramanujanTauLZerosTab128[[i]],0},{i,1,60}],
PlotRange->{{0,xmax},{-0.2,6}},
Plot[Abs[myRFunc[x]],{x,2,xmax},PlotStyle->Black],PlotRange->{{0,xmax},{-
0.2,6}}]
```

20.10.6 REPUNIT ZAHLEN UND REPUNIT PRIMZAHLEN

(In Vorbereitung!)