Problem K - Daily Dividend

Seminar – Algorithmen für Programmierwettbewerbe

Andre Johnson, Hoang Hai Nguyen, Ha Giang Hoang Tran

Agenda

1 **Problem**

2 Ansätze

3 Laufzeiten

Problem

David ist ein kleiner Junge und liebt es, mit Zahlen zu spielen. Erst neulich hat er gelernt, wie man die Division von zwei Zahlen durchführt. David macht nun den ganzen Tag lang Divisionsrechnungen. Er ist dann auch sehr zufrieden, wenn das Ergebnis der Division eine ganze Zahl ist. Allerdings ist er nicht sonderlich erfreut, wenn das nicht der Fall sein sollte. Nach einer Weile hat David sich dazu entschieden, nur noch einen Dividend pro Tag zu benutzen.

$$\underbrace{a}_{\text{Dividend}} : \underbrace{b}_{\text{Divisor}} = c$$

Davids Eltern sind sehr fürsorglich und möchten daher sicherstellen, dass David möglichst glücklich ist. Aus diesem Grund entscheiden sie jeden Tag, welche Zahl David als Dividend benutzen wird.

Allerdings gibt es da ein Problem: Die Eltern sind nicht besonders gut in Mathe und wissen daher nicht, wie man die Anzahl an positiven ganzzahligen Divisoren für einen gegebenen Dividend N berechnet, welche einen ganzzahligen Quotienten liefern.

Nun ist es Zeit, dass ihr Davids Eltern bei dieser Aufgabe helft.

Problem

David ist ein kleiner Junge und liebt es, mit Zahlen zu spielen. Erst neulich hat er gelernt, wie man die Division von zwei Zahlen durchführt. David macht nun den ganzen Tag lang Divisionsrechnungen. Er ist dann auch sehr zufrieden, wenn das Ergebnis der Division eine ganze Zahl ist. Allerdings ist er nicht sonderlich erfreut, wenn das nicht der Fall sein sollte. Nach einer Weile hat David sich dazu entschieden, nur noch einen Dividend pro Tag zu benutzen.

$$\underbrace{a}_{\text{Dividend}} : \underbrace{b}_{\text{Divisor}} = c$$

Davids Eltern sind sehr fürsorglich und möchten daher sicherstellen, dass David möglichst glücklich ist. Aus diesem Grund entscheiden sie jeden Tag, welche Zahl David als Dividend benutzen wird.

Allerdings gibt es da ein Problem: Die Eltern sind nicht besonders gut in Mathe und wissen daher nicht, wie man die Anzahl an positiven ganzzahligen Divisoren für einen ***gegebenen Dividend N*** berechnen, welche einen ganzzahligen Quotienten liefern.

Nun ist es Zeit, dass ihr Davids Eltern bei dieser Aufgabe helft.

Beispiele

Eingabe:	Teiler:	Ausgabe:
12	1, 2, 3, 4, 6, 12	6
31	1, 31	2
100	1, 2, 4, 5, 10, 20, 25, 50, 100	9
999999999999999989	1, 999999999999999989	2
1000000077000000049	1, 100000007,1000000007, 1000000077000000049	4

Die **Eingabe** besteht aus **einer Zeile**. Sie enthält eine Ganzzahl $\boldsymbol{N}$, wobei $\mathbf{1} \leq \boldsymbol{N} \leq \mathbf{10^{18}}$

Die **Ausgabe** besteht aus **einer Zeile** und gibt die Anzahl der **Divisoren von $\boldsymbol{N}$** an.

Agenda

1 Problem

2 **Ansätze**

3 Laufzeiten

Ansatz 1 – Der Intuitive

Zähler zum Festhalten der Anzahl der Divisoren von n.

Gehe alle Zahlen von 1 bis n durch und prüfe, ob sie ein Divisor von n ist.

Ist die aktuelle Zahl ein Divisor von n, dann erhöhe den Zähler um 1.

Laufzeit?

Ansatz 1 – Der Intuitive

Beispiel n = 12

$12 : 1 = 12$	✓	$12 : 7 = 1{,}7142 \ldots$	✗
$12 : 2 = 6$	✓	$12 : 8 = 1{,}5$	✗
$12 : 3 = 4$	✓	$12 : 9 = 1{,}3333 \ldots$	✗
$12 : 4 = 3$	✓	$12 : 10 = 1{,}2$	✗
$12 : 5 = 2{,}4$	✗	$12 : 11 = 1{,}0909 \ldots$	✗
$12 : 6 = 2$	✓	$12 : 12 = 1$	✓

6 × ✓ ➔ Ergebnis **6**

Ansatz 1 – Der Intuitive

$100000007700000049 : 1$	$= 100000007700000049$	✓
$100000007700000049 : 2$	$= 5,0 \ldots 385 \cdot 10^{16}$	✗
$100000007700000049 : 3$	$= 3,33 \ldots \cdot 10^{16}$	✗
…	…	✗✗✗
$100000007700000049 : 100000007$	$= 1000000007$	✓
…	…	✗✗✗
$100000007700000049 : 1000000007$	$= 100000007$	✓
…	…	✗✗✗
$100000007700000049 : 100000007700000049$	$= 1$	✓

4 × ✓

➔ Ergebnis **4**

Ansatz 1 – Der Intuitive

Zähler zum Festhalten der Anzahl der Divisoren von n.

Gehe alle Zahlen von 1 bis n durch und prüfe, ob sie ein Divisor von n ist.

Ist die aktuelle Zahl ein Divisor von n, dann erhöhe den Zähler um 1.

Laufzeit? – $\boldsymbol{O(n)}$

Für sehr große Zahlen dauert das lange …

Geht doch besser?!

Ansatz 2 – Der Weg zur Erkenntnis

Beispiel n = 12

$12 : 1 = 12$
$12 : 2 = 6$
$12 : 3 = 4$
$12 : 4 = 3$
$12 : 5 = 2{,}4$
$12 : 6 = 2$

$12 : 7 = 1{,}7142 \ldots$
$12 : 8 = 1{,}5$
$12 : 9 = 1{,}3333 \ldots$
$12 : 10 = 1{,}2$
$12 : 11 = 1{,}0909 \ldots$
$12 : 12 = 1$

Ansatz 2 – Die Erkenntnis

Für zusammengesetzte Zahlen gilt: $n = p \cdot q$

$\sqrt{12} = 3{,}46 \ldots$

$p \leq \sqrt{n}$ und $q \geq \sqrt{n}$

$12 : 1 = 12$
$12 : 2 = 6$
$12 : 3 = 4$
$12 : 4 = 3$
$12 : 6 = 2$
$12 : 12 = 1$

Ansatz 2 – Die Erkenntnis

Für zusammengesetzte Zahlen gilt: $n = p \cdot q$

$p \leq \sqrt{n}$ und $q \geq \sqrt{n}$

Beispiel?

$$100 = 10 \cdot 10 = 5 \cdot 20$$

Erkenntnis:

Wir müssen nur noch bis $\boldsymbol{\sqrt{n}}$ iterieren!

Ansatz 2 – Die Erkenntnis

Beispiel n = 100, $\sqrt{100} = 10$

$100 : 1 =$	100	✓	✓	$100 : 100 = 1$
$100 : 2 =$	50	✓	✓	$100 : 50 = 2$
$100 : 3 =$	$33{,}3 \ldots$	✗		
$100 : 4 =$	25	✓	✓	$100 : 25 = 4$
$100 : 5 =$	20	✓	✓	$100 : 20 = 5$
$100 : 6 =$	$16{,}6 \ldots$	✗		
$100 : 7 =$	$14{,}2 \ldots$	✗		
$100 : 8 =$	$12{,}5$	✗		
$100 : 9 =$	$11{,}1 \ldots$	✗		
$100 : 10 =$	10	✓	~~✓~~	~~$100 : 10 = 10$~~

Hier sind es nur noch **10 Schritte statt 100!**

9 × ✓

➔ Ergebnis **9**

Ansatz 2 – Die Erkenntnis

Initialisiere den Zähler für die Anzahl der Divisoren.

Prüfe für jede Zahl von 1 bis $\sqrt{\boldsymbol{n}}$, ob sie ein Divisor von n ist

Ist das der Fall, dann erhöhe den Zähler (um wie viel?)

Laufzeit? - $\boldsymbol{O}\left(\sqrt{\boldsymbol{n}}\right)$

Besser … aber noch „zu langsam“ für sehr große Zahlen

Ansatz 3 – Die Lösung

$$12 = 1 \cdot \boxed{2 \cdot 2} \cdot 3$$

Teiler:

$$1 = 1 = \text{leeres Produkt}$$

$$2 = 2$$

$$3 = 3$$

$$4 = \boxed{2 \cdot 2}$$

$$6 = 2 \cdot 3$$

$$12 = 2 \cdot 2 \cdot 3$$

Ansatz 3 – Die Lösung

Primfaktorzerlegung von n

Vorgehensweise Primfaktorzerlegung:

1) Teile n durch 2 solange es geht

2) Teile den Rest r durch alle ungeraden Zahlen bis $\sqrt{r}$

3) Wenn der Rest größer als 2 ist, dann nimm den Rest auch mit auf als Primfaktor

Ansatz 3: Beispiel Primfaktorzerlegung

2	3	5
100		
↓ : 2		
50		
↓ : 2		
25	25	25
↓ : 2	↓ : 3	↓ : 5
~~12,5~~	~~8,33...~~	5
		↓ : 5
		1

Ergebnis: $2 \cdot 2 \cdot 5 \cdot 5$

Ansatz 3 – Die Lösung

Primfaktorzerlegung von n

Beispiel: $PFZ(12) = 2 \cdot 2 \cdot 3$

Von der Primfaktorzerlegung zur Anzahl der Divisoren – **eine Idee**:

Alle **Kombinationen** ermitteln:

$$|\{(2 \cdot 2 \cdot 3), (2 \cdot 2), (2 \cdot 3), (2), (3)\} \cup \{1\}| = 6$$

Das dauert lange für sehr große Zerlegungen!!

Es gibt eine Formel!

Ansatz 3 – Die Lösungsformel

$$\boldsymbol{Divs(N) = (\#PF_1 + 1) \cdot (\#PF_2 + 1) \cdot \ldots \cdot (\#PF_n + 1)}$$

Beispiel: $PFZ(12) = 2 \cdot 2 \cdot 3 = 2^2 \cdot 3^1$

Häufigkeit von Primfaktor 2: 2x

Häufigkeit von Primfaktor 3: 1x

$$Divs(12) = (2 + 1) \cdot (1 + 1) = 3 \cdot 2 = 6$$

Ansatz 3 – Veranschaulichung der Lösungsformel

$$\boldsymbol{Divs(N) = (\#PF_1 + 1) \cdot (\#PF_2 + 1) \cdot \ldots \cdot (\#PF_n + 1)}$$

$$2^0 \cdot 3^0 = 1$$

$$2^0 \cdot 3^1 = 3$$

$$2^1 \cdot 3^0 = 2$$

$$2^2 \cdot 3^0 = 4$$

$$2^1 \cdot 3^1 = 6$$

$$2^2 \cdot 3^1 = 12$$

$$\boldsymbol{PFZ(12) = 2^2 \cdot 3^1}$$

$$\boldsymbol{Divs(12) = (2 + 1) \cdot (1 + 1) = 6}$$

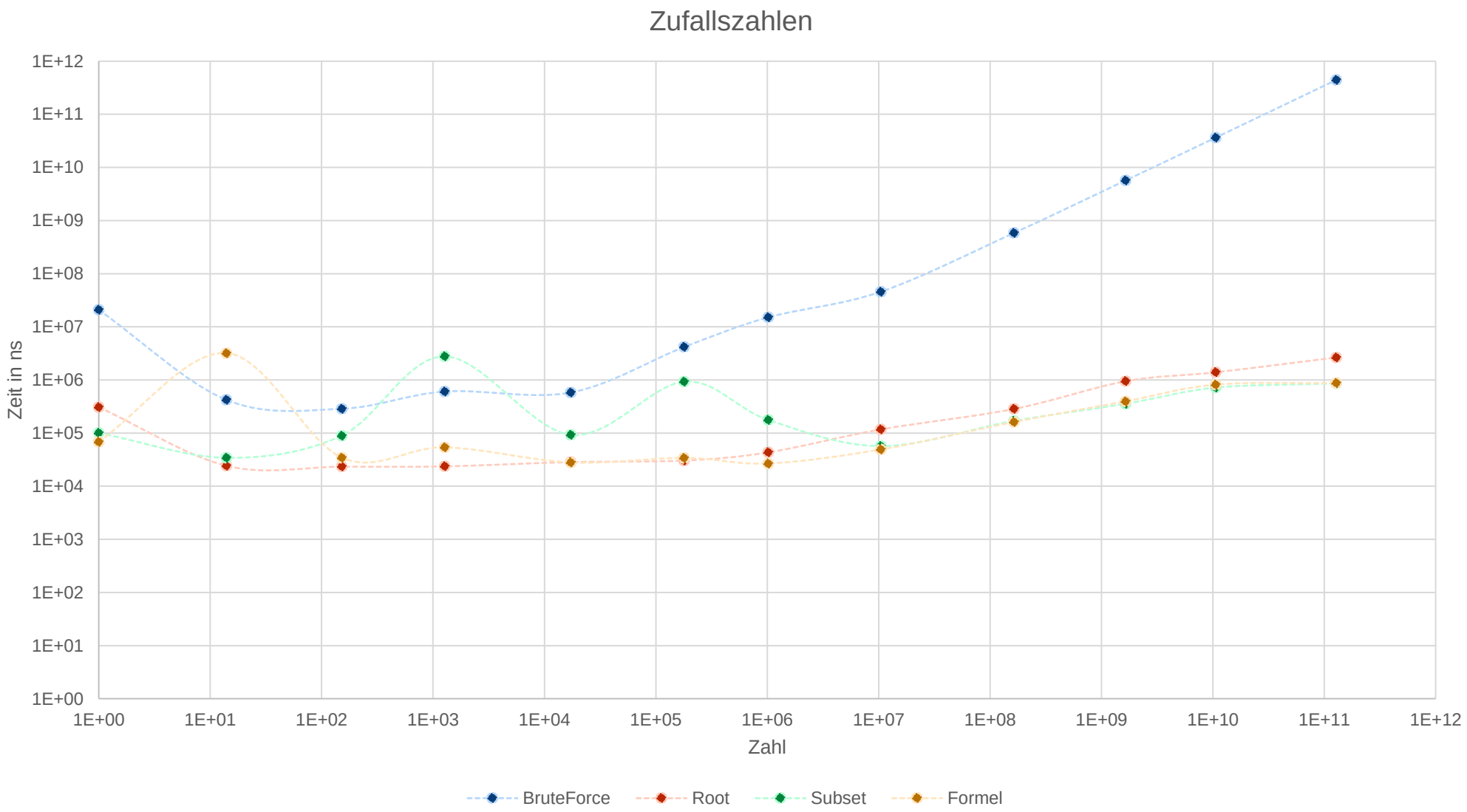
Agenda

1 Problem

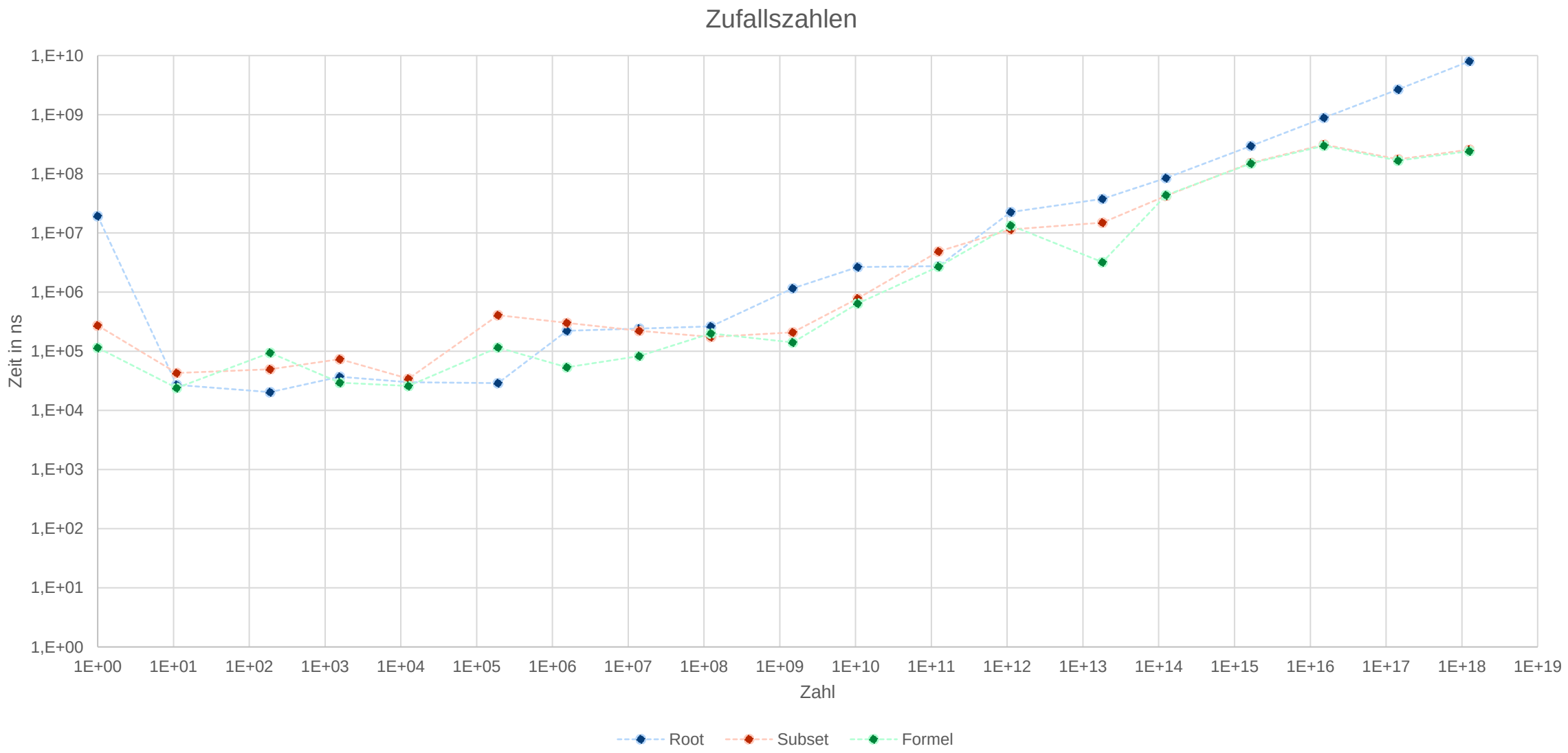
2 Ansätze

3 **Laufzeiten**

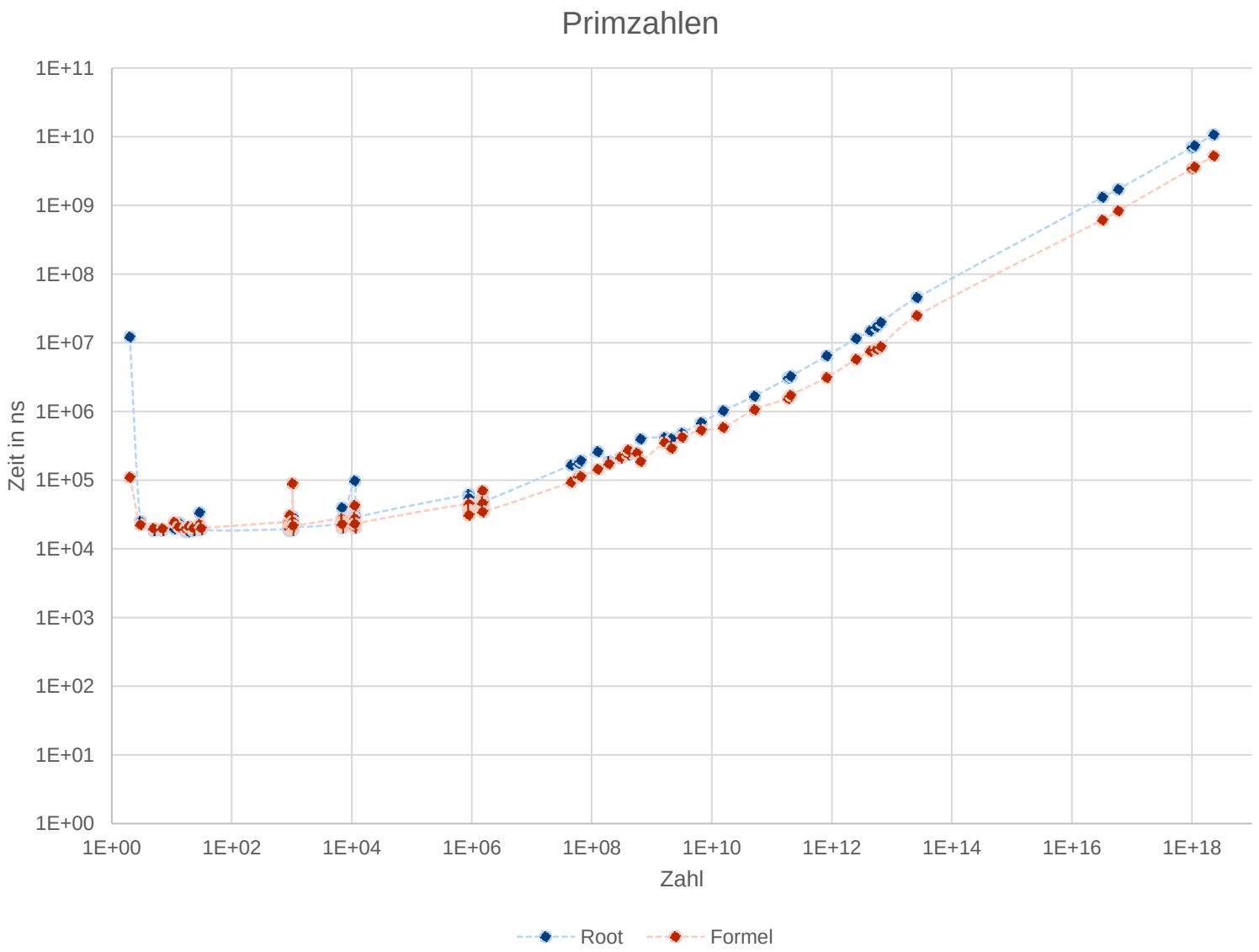
Benchmark


Zufallszahlen
Zeit in ns
1E+12
1E+11
1E+10
1E+09
1E+08
1E+07
1E+06
1E+05
1E+04
1E+03
1E+02
1E+01
1E+00
1E+00
1E+01
1E+02
1E+03
1E+04
1E+05
1E+06
1E+07
1E+08
1E+09
1E+10
1E+11
1E+12
Zahl
BruteForce
Root
Subset
Formel

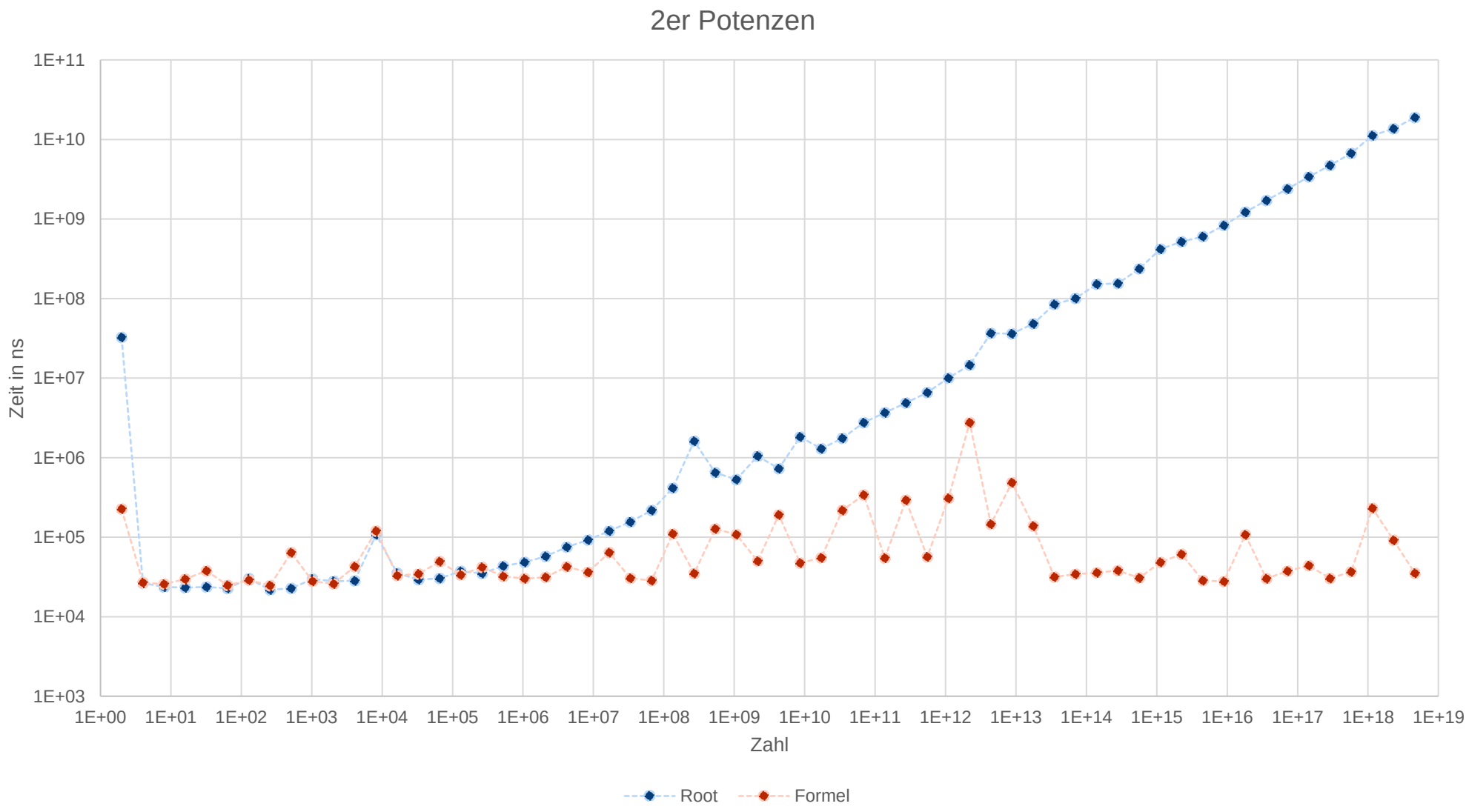
Benchmark


Zufallszahlen
1,E+10
1,E+09
1,E+08
1,E+07
1,E+06
1,E+05
1,E+04
1,E+03
1,E+02
1,E+01
1,E+00
Zeit in ns
1E+00
1E+01
1E+02
1E+03
1E+04
1E+05
1E+06
1E+07
1E+08
1E+09
1E+10
1E+11
1E+12
1E+13
1E+14
1E+15
1E+16
1E+17
1E+18
1E+19
Zahl
Root
Subset
Formel

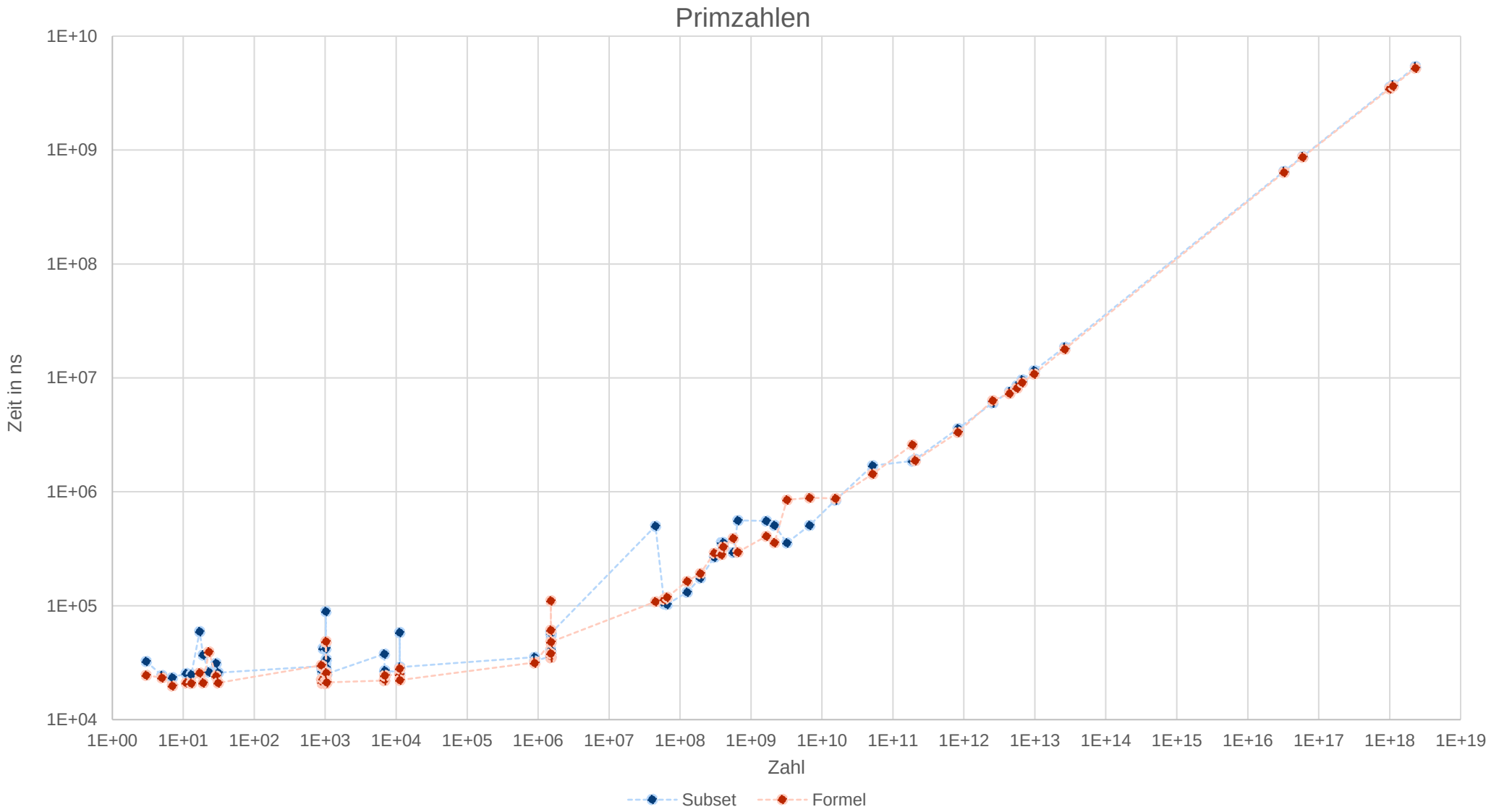
Benchmark


Primzahlen
1E+11
1E+10
1E+09
1E+08
1E+07
1E+06
1E+05
1E+04
1E+03
1E+02
1E+01
1E+00
Zeit in ns
1E+00
1E+02
1E+04
1E+06
1E+08
1E+10
1E+12
1E+14
1E+16
1E+18
Zahl
Root
Formel

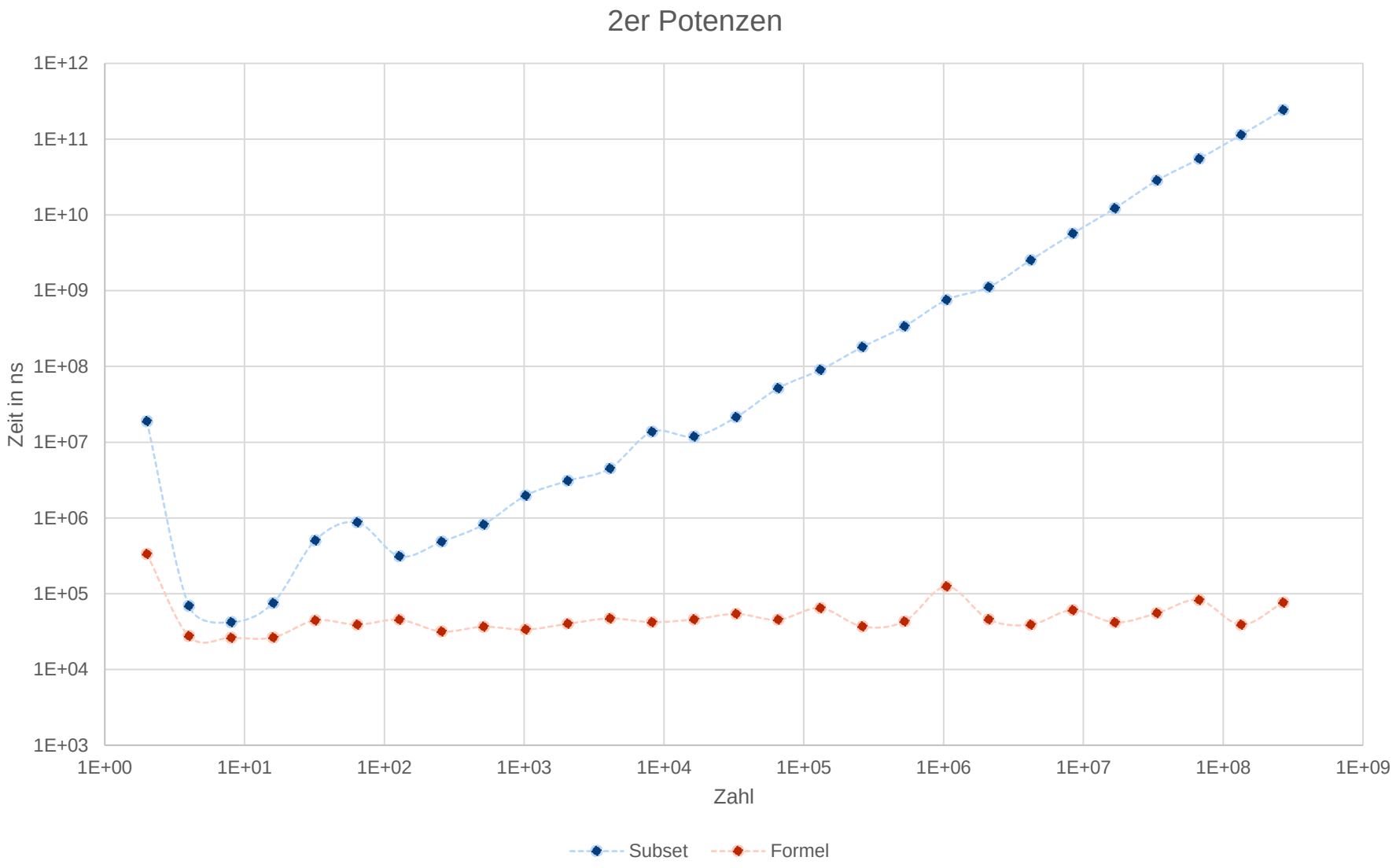
Benchmark


2er Potenzen
Zeit in ns
1E+11
1E+10
1E+09
1E+08
1E+07
1E+06
1E+05
1E+04
1E+03
1E+00
1E+01
1E+02
1E+03
1E+04
1E+05
1E+06
1E+07
1E+08
1E+09
1E+10
1E+11
1E+12
1E+13
1E+14
1E+15
1E+16
1E+17
1E+18
1E+19
Zahl
Root
Formel

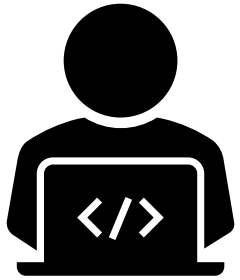
Benchmark


Primzahlen
Zeit in ns
1E+10
1E+09
1E+08
1E+07
1E+06
1E+05
1E+04
1E+00
1E+01
1E+02
1E+03
1E+04
1E+05
1E+06
1E+07
1E+08
1E+09
1E+10
1E+11
1E+12
1E+13
1E+14
1E+15
1E+16
1E+17
1E+18
1E+19
Zahl
Subset
Formel

Benchmark


2er Potenzen
1E+12
1E+11
1E+10
1E+09
1E+08
1E+07
1E+06
1E+05
1E+04
1E+03
Zeit in ns
1E+00
1E+01
1E+02
1E+03
1E+04
1E+05
1E+06
1E+07
1E+08
1E+09
Zahl
Subset
Formel

DANKESCHÖN

Was ihr wissen müsst:

- Iterieren der Zahlen von 1 bis $\sqrt{n}$ reicht
- Was besonders wichtig ist:
  - Effizienz der Primfaktorzerlegung von n
  - Effizienz der Berechnung der Anzahl der Divisoren aus der PFZ