

Hans Walser, [20080204a]

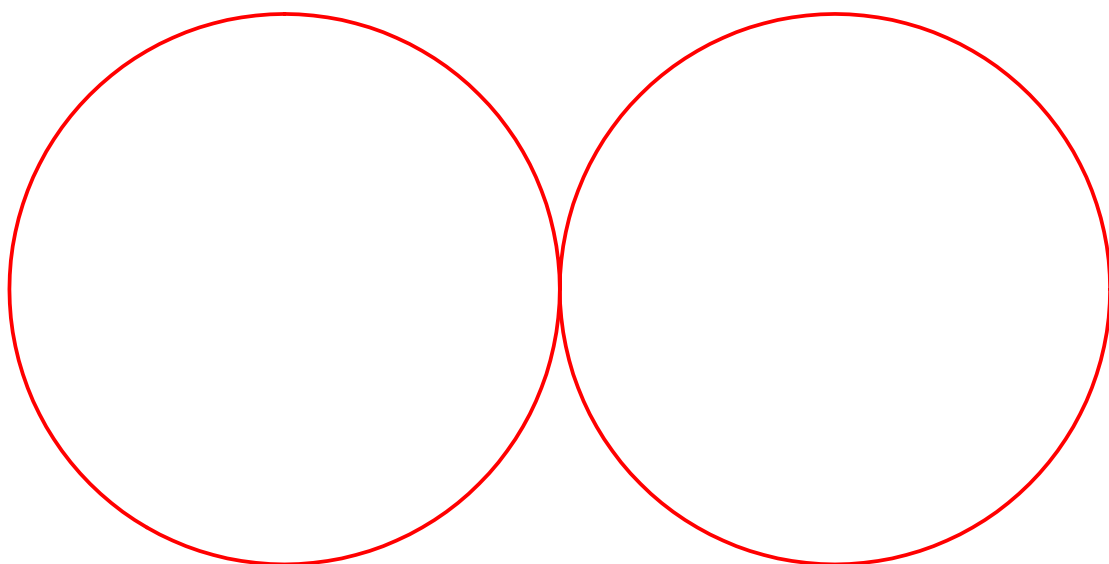
Kreisfiguren mit berührenden Kreisen

1 Worum es geht

Zu $n \in \mathbb{N}$ werden 2^n gleich große Kreise gezeichnet, von denen jeder n andere Kreise berührt.

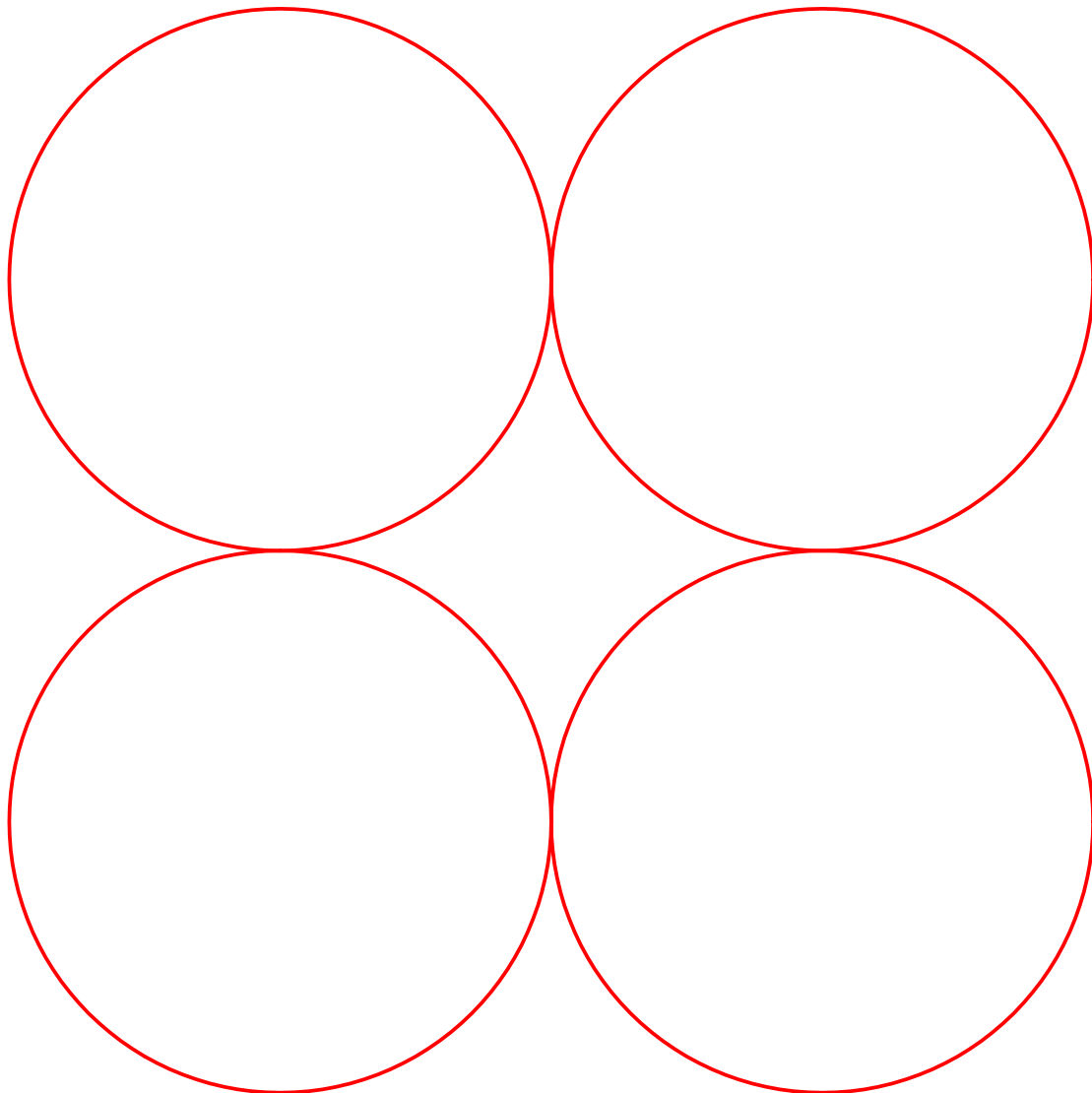
2 Beispiele

2.1 $n = 1$

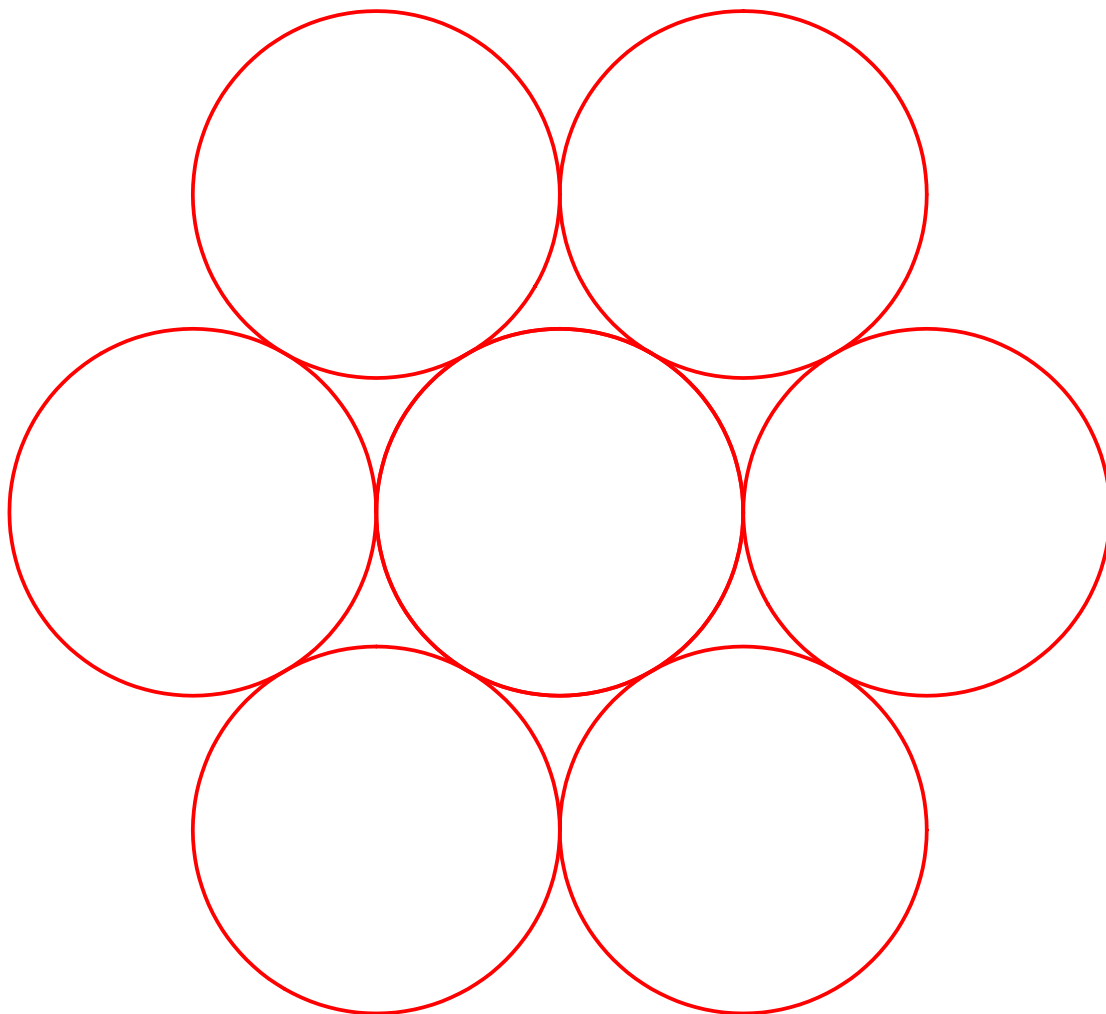


Zwei Kreise

2.2 $n = 2$

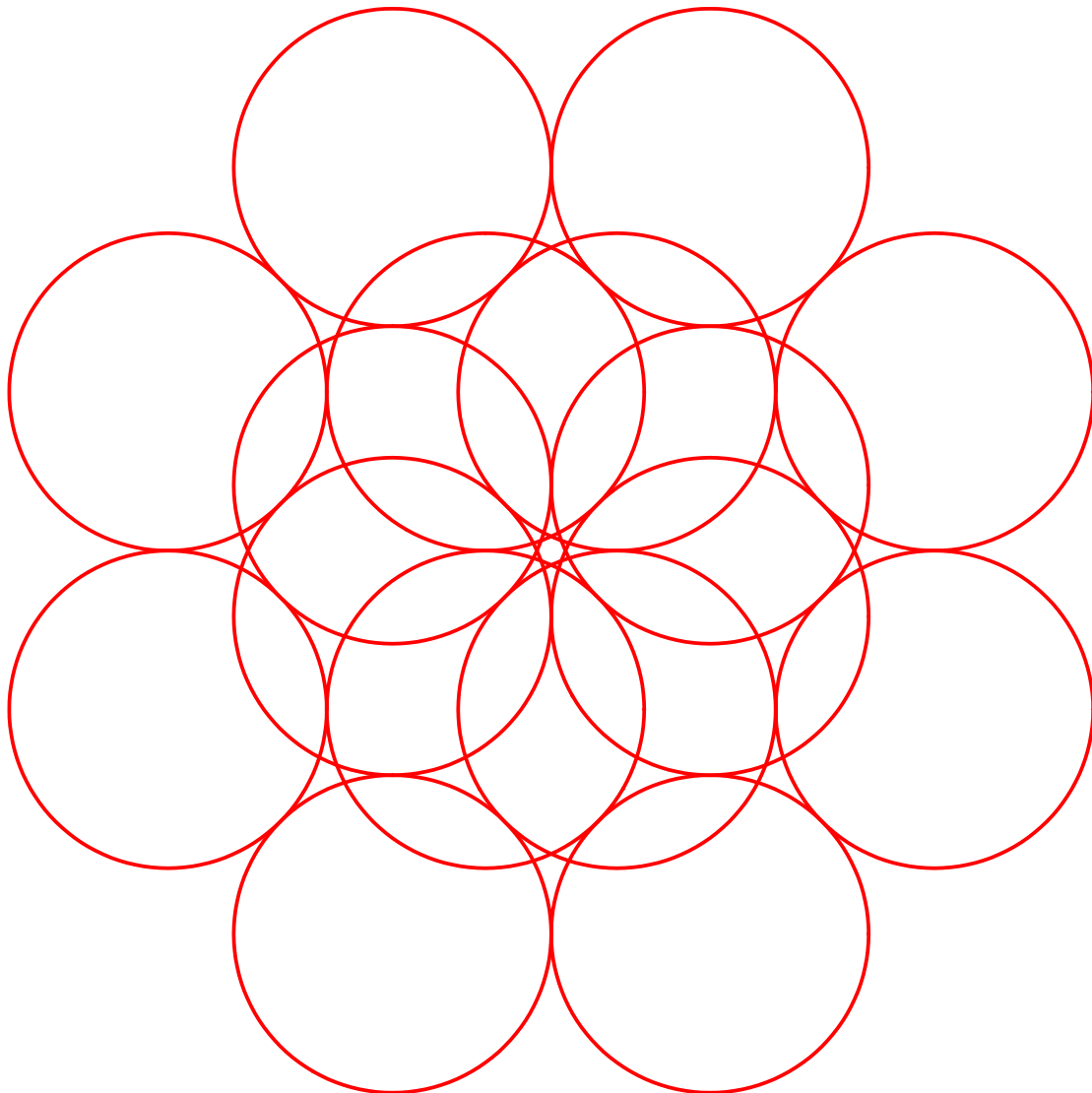


Vier Kreise

2.3 $n = 3$ **Acht Kreise**

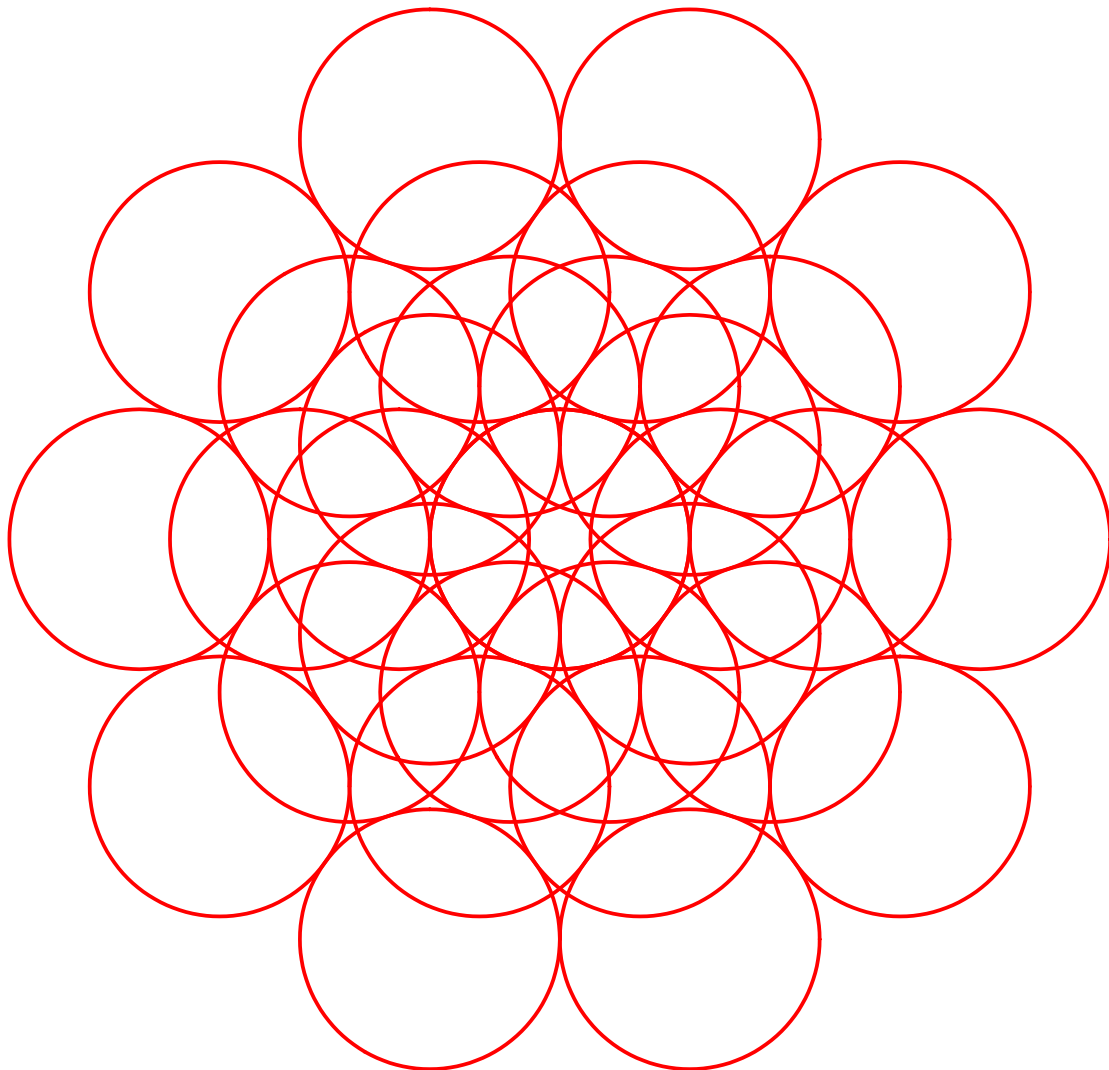
Eigentlich sehen wir nur sieben Kreise, von denen einer, der zentrale Kreis, von den sechs anderen Kreisen berührt wird. Wir müssen diesen Kreis doppelt zählen und jeden von den beiden alternierend durch je drei Kreise berühren lassen. Wie es zu dieser eigenartigen Zählweise kommt, wird später erklärt.

2.4 $n = 4$



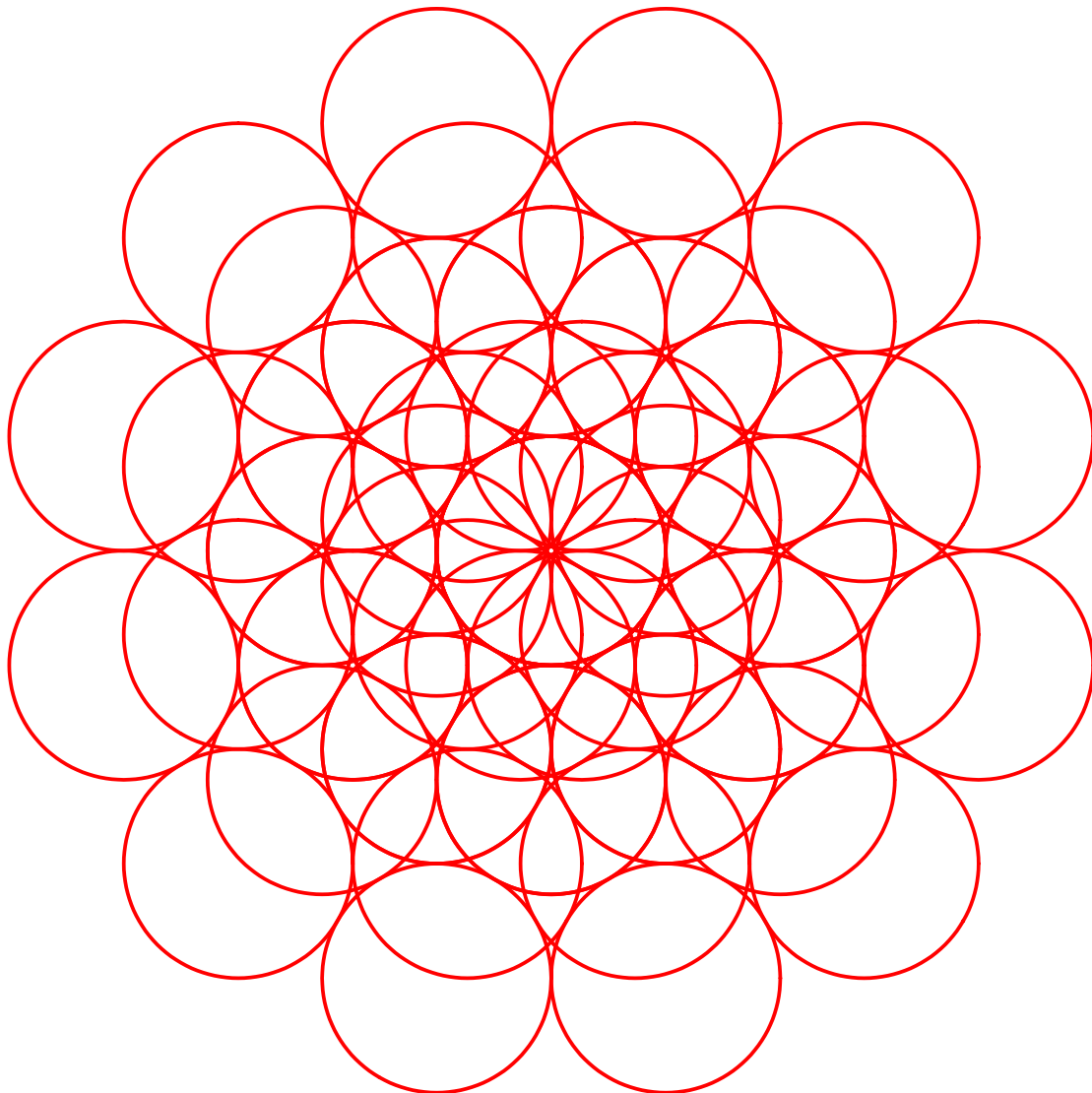
16 Kreise

2.5 $n = 5$



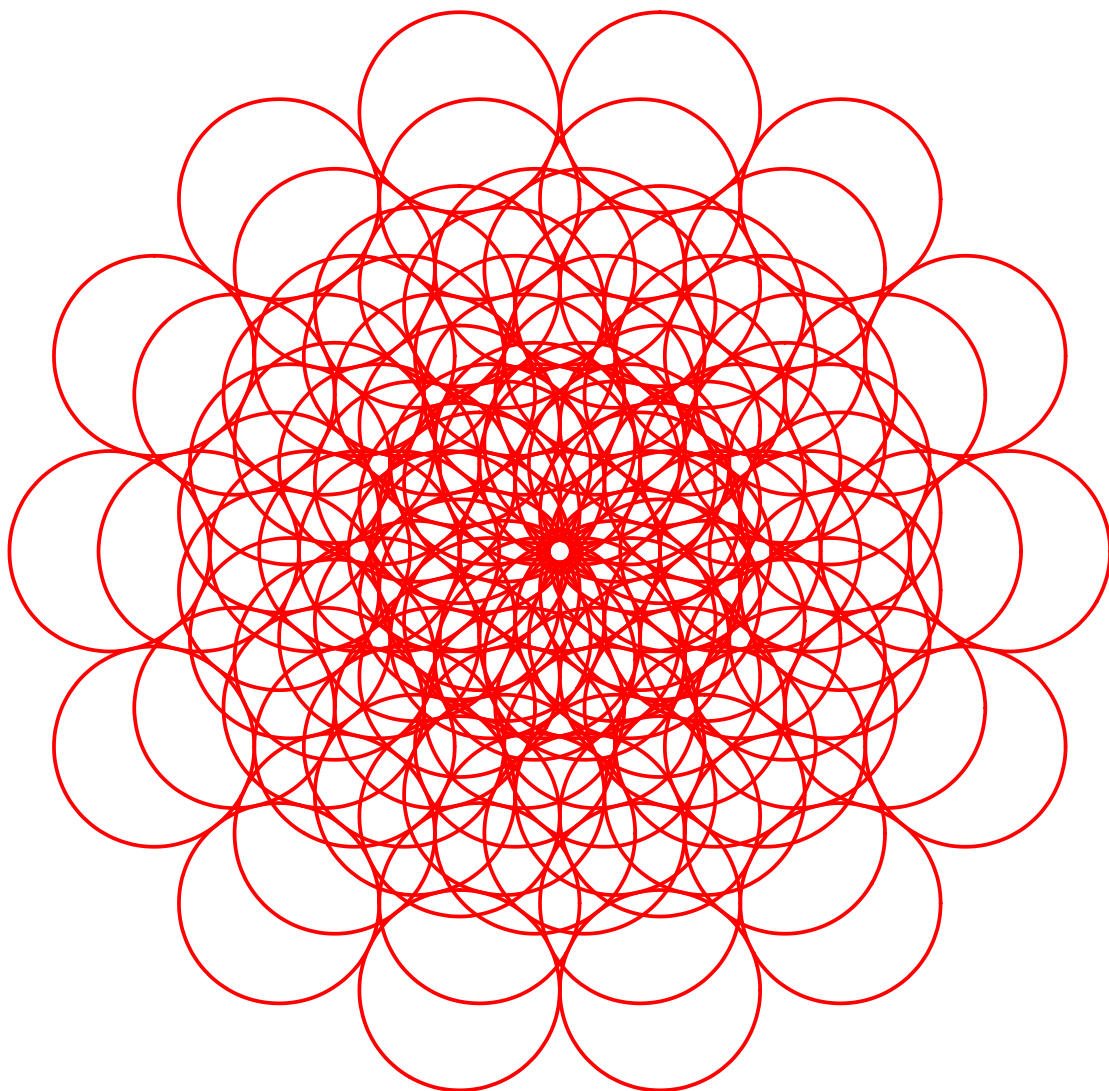
$32 = 2^5$ Kreise

2.6 $n = 6$



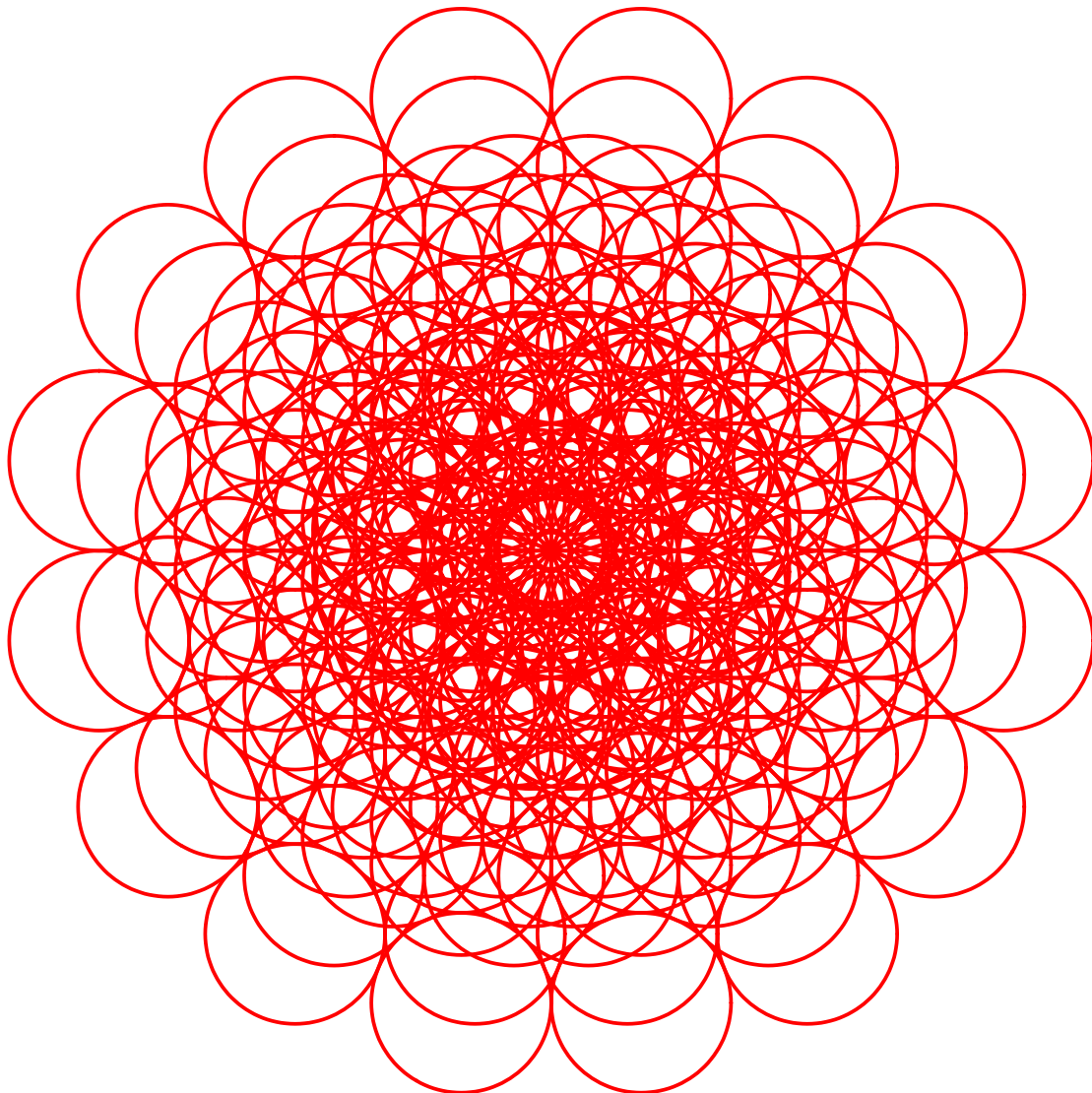
$64 = 2^6$ Kreise

2.7 $n = 7$



128 = 2⁷ Kreise

2.8 $n = 8$

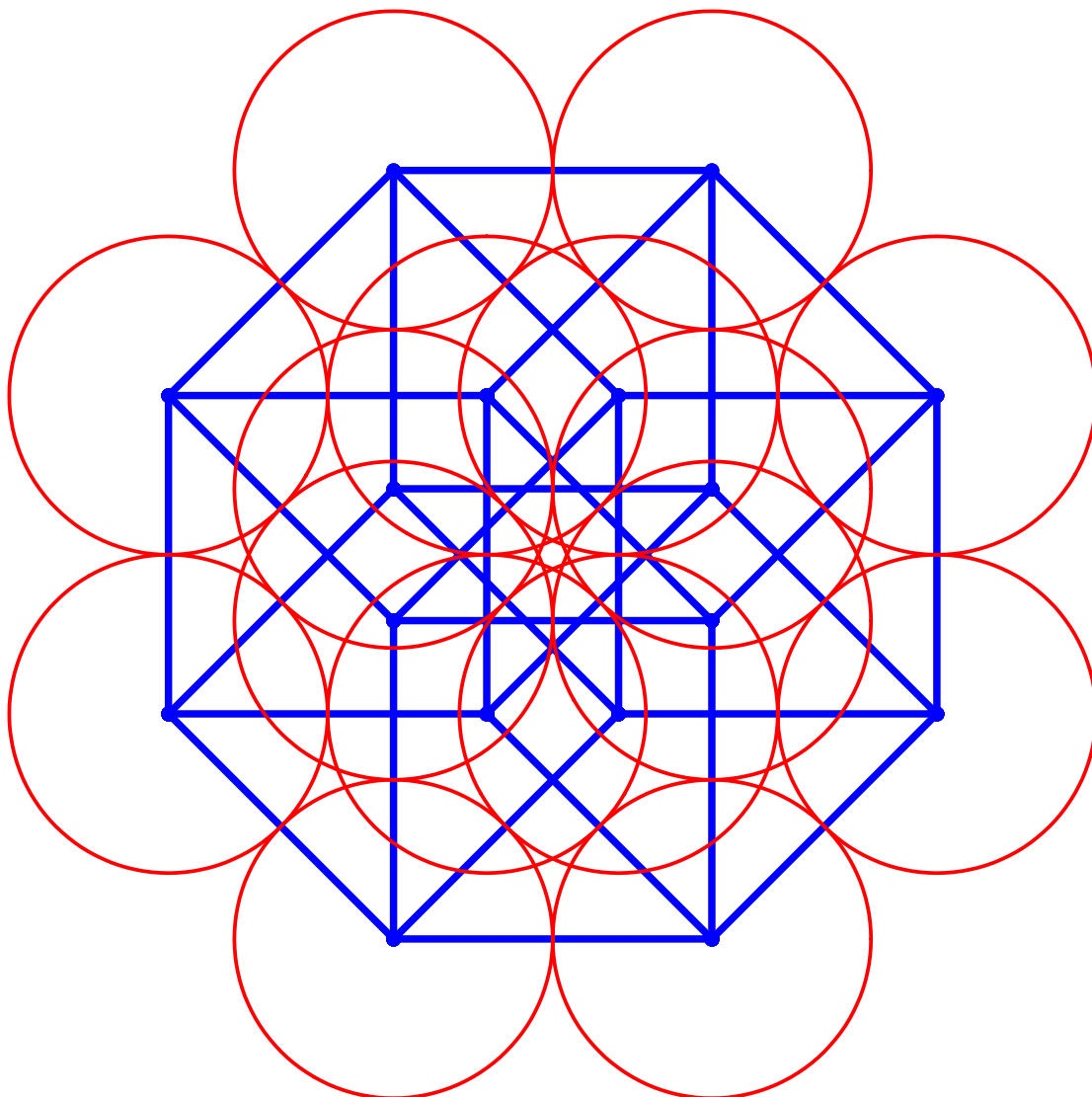


$256 = 2^8$ Kreise

3 Hintergrund

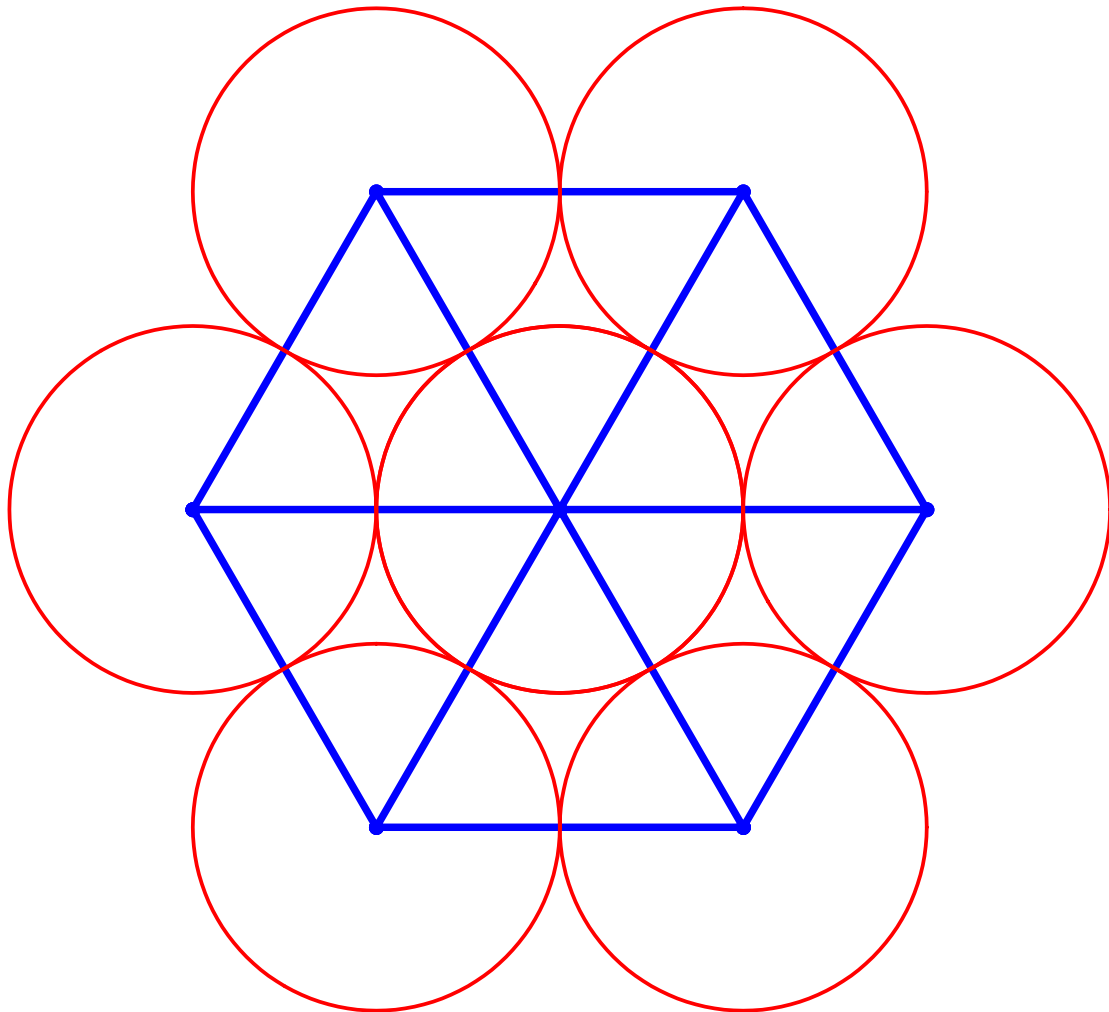
Wir zeichnen einen n -dimensionalen Hyperwürfel in isometrischer Darstellung. Von jeder Ecke aus verlaufen kann n gleich stark verkürzte Kanten. Daher kann jeder Ecke als Zentrum eines Kreises verstanden werden, dessen Radius die Hälfte der Länge der verkürzten Kante ist und der daher n Nachbarkreise berührt.

Dies im ebenen Bild. Im n -dimensionalen Raum entsprechen den Kreisen Hyperkugeln. Das folgende Bild illustriert den Fall $n = 4$.



Vierdimensionaler Hyperwürfel mit 16 Kreisen

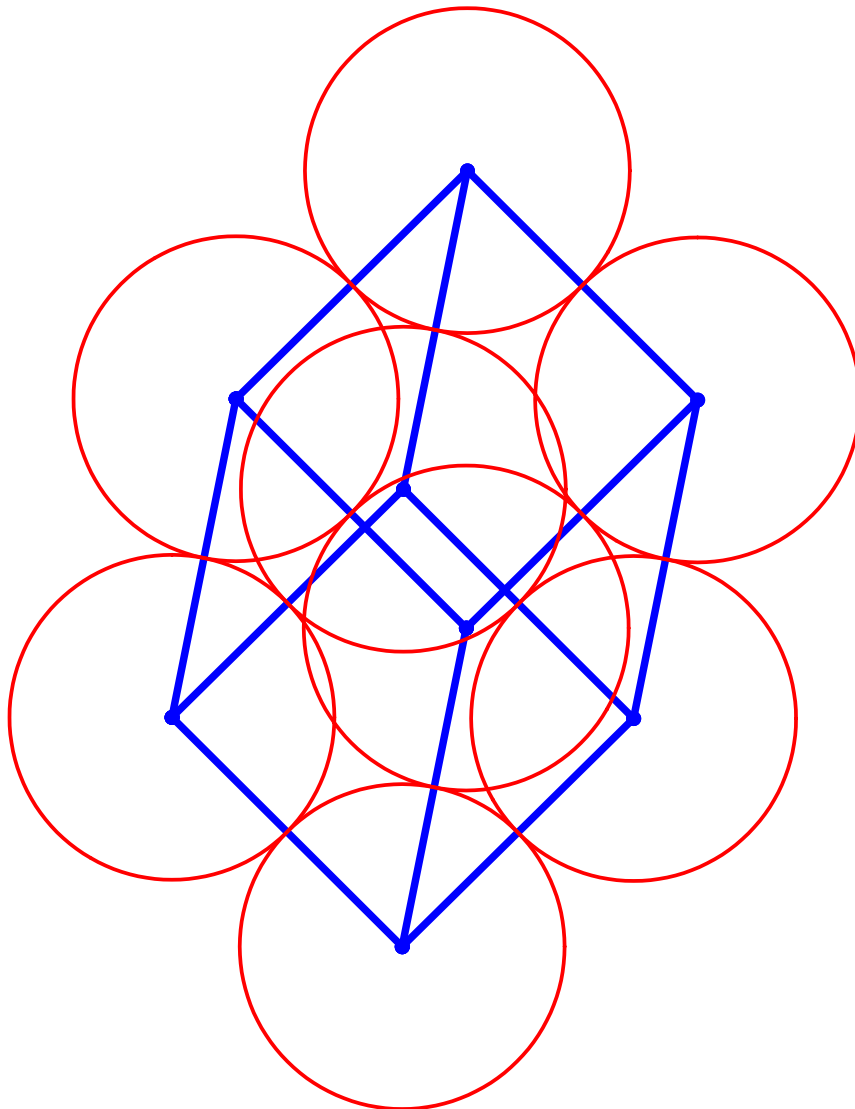
Bei $n = 3$ fallen in der isometrischen Darstellung zwei Würfecken aufeinander. Daher die Doppelzählung des „zentralen“ Kreises.



Zwei Ecken fallen aufeinander

Dieser Effekt tritt auch bei anderen Dimensionen auf.

Wenn wir die Würfecken weniger symmetrisch zeichnen, werden alle acht Kreise sichtbar. In dieser Situation sind die Winkel zwischen den projizierten Kanten nicht mehr regelmäßig.



Alle acht Kreise sichtbar

4 MuPAD-Programm

Exemplarisch das Programm für $n = 4$:

```
n:=4:
N:=2^n-1:

for k from 0 to N do
  x[k]:=0:
  y[k]:=0:
  for j from 1 to n do
    x[k]:=x[k]+round(frac(k*(1/2)^(n-j+1)))*cos(j*PI/n):
    y[k]:=y[k]+round(frac(k*(1/2)^(n-j+1)))*sin(j*PI/n):
  end_for:
end_for:

Kreis:=k->plot::Curve2d([x[k]+1/2*cos(t), y[k]+1/2*sin(t)],
t=0..2*PI, LineWidth=1/2, LineColor=[1,0,0]):

plot(Kreis(k)$k=0..N, Scaling=Constrained, Axes=None,
Width=150, Height=150):
```