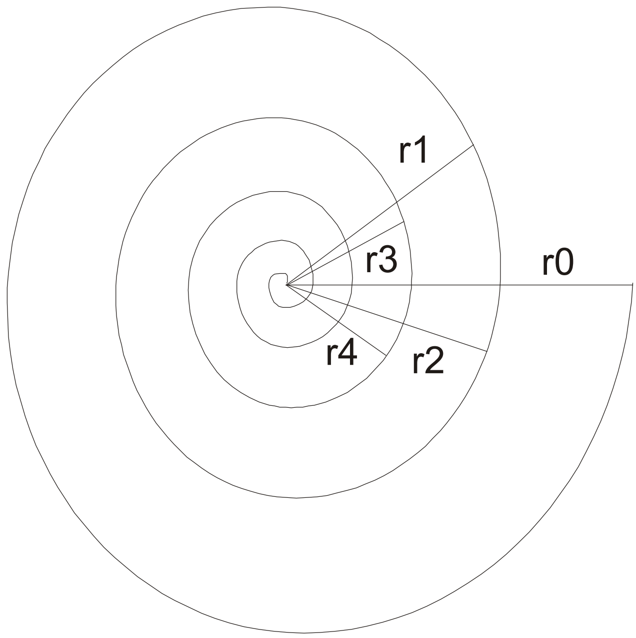
Umfang einer Spirale

von Stephan Senn

Gegeben sei eine gleichmässige Spirale:



Für die Radien gilt:

$r_n = q^n \cdot r_0$ mit $q = \frac{1}{2}$ und $r_0 = 1$

Es handelt sich also um eine geometrische Folge (GF).

Der Umfang eines Kreises berechnet sich wie folgt:

$U = 2 \cdot r \cdot \pi$

Da eine Bogenlänge B_i der Spirale gerade einen Halbkreis darstellt gilt:

$$B_i = \frac{1}{2} \cdot 2 r_i \pi = r_i \pi$$

Für den gesamten Umfang der Spirale gilt dann:

$$U(n) = B_0 + B_1 + \ldots + B_n = \sum_{i=0}^{n} B_i = \sum_{i=0}^{n} \pi \cdot r_i = \pi \cdot \sum_{i=0}^{n} r_i = \pi \cdot \sum_{i=0}^{n} q^i \cdot r_0 \quad (1)$$

Die Summenformel für die geometrische Reihe lautet:

$$S(n) = a_0 \cdot \frac{1 - q^{n+1}}{1 - q}$$

n soll gegen unendlich streben (Grenzprozess!). Es gilt also: $n \to \infty$

Weiter gilt: $q = \frac{1}{2} \Rightarrow |q| < 1$

Somit gilt für die Summenformel:

$$\lim_{n\to\infty}\left(S(n)\right)=\lim_{n\to\infty}\left(a_0\cdot\frac{1-q^{n+1}}{1-q}\right)=a_0\cdot\frac{1}{1-q}\qquad(2)$$

Deshalb gilt für den Umfang der Spirale, wenn n gegen unendlich strebt:

$$\lim_{n\to\infty}\left(U(n)\right)=\lim_{n\to\infty}(\pi\cdot\sum_{i=0}^{n}q^i\cdot r_0)=\pi\cdot\left(r_0\cdot\frac{1}{1-q}\right)=\pi\cdot\left(r_0\cdot\frac{1}{1-\frac{1}{2}}\right)=\pi\cdot\left(r_0\cdot 2\right)=2\pi$$

Die Summenformel (2) wurde dabei in die Gleichung (1) für das ‚Summenzeichen' eingesetzt. Das ist erlaubt, da die Summenformel ja nichts anderes als die Summe der einzelnen Glieder der Geometrischen Folge (also die Reihe) berechnet.

Wir sehen also, dass hier ein unendlicher Grenzprozess zu einem endlichen Resultat führt. Wir bezeichnen dieses Phänomen als Konvergenz, da sich offenbar die Länge der Spirale 2π ergibt, wenn n gegen unendlich strebt.

Wir sind nun sehr skeptisch, ob dieses Resultat auch wirklich stimmt. Deshalb verwenden wir den Computer und probieren für sehr grosse n, den Wert von 2π zu verifizieren. Unser C-Programm lautet:

```
// Umfang einer Spirale
// von Stephan Senn

// Bibliothek für Ausgabe einbinden
#include <iostream>

// Standardbibliotheken verwenden
using namespace std;

// Hauptprogramm
int main(void)
{
      long n; // Endwert, Abbruchbedingung
      long double r0=1; // erster Bogenradius
      long double k=0.5; // Laufvariable für Streckungsfaktoren der Radien
      long double q=0.5; // Streckungsfaktor der Radien
      const double pi=3.14159265359; // Zahl Pi
      const double l=2*pi; // exakte Lösung des Grenzwertproblems
      long double r=pi; // Wert bei i=0

      // Eingabe
      cout<<"Umfang einer Spirale"<<endl;
      cout<<"von Stephan Senn"<<endl;
      cout<<"------------------------------------"<<endl;
      cout<<"Wert fuer n eingeben:"<<endl;
      cin>>n;

      cout<<"r(i=0)="<<r0<<endl; // erster Wert ausgeben

      // Iteration definieren...
      for(int i=1;i<n+1;i++)
      {
            r=r+r0*k*pi;
            k=k*q;
            cout<<"r("<<"i="<<i<<")="<<r<<endl;
      }
```

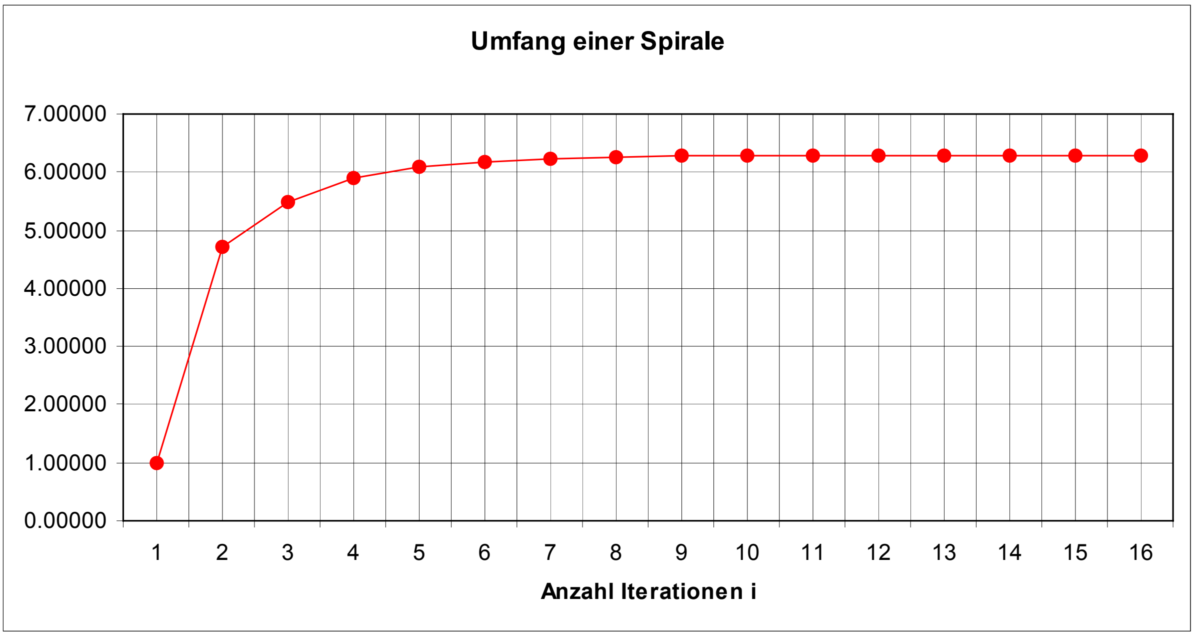
```
        // Ausgabe
        cout<<"Numerischer Wert fuer r bei i="<<n<<": "<<r<<endl;
        cout<<"Exakter  Wert  nach  anayltischer  Berechnung  mit  Grenzprozess:
"<<l<<endl;
        cout<<"Absoluter Fehler: "<<l-r<<endl;

        return 0;

}
```

Für n=15 erhalten wir:

```
Umfang einer Spirale
von Stephan Senn
------------------------------------
Wert fuer n eingeben:
15
r(i=0)=1
r(i=1)=4.71239
r(i=2)=5.49779
r(i=3)=5.89049
r(i=4)=6.08684
r(i=5)=6.18501
r(i=6)=6.2341
r(i=7)=6.25864
r(i=8)=6.27091
r(i=9)=6.27705
r(i=10)=6.28012
r(i=11)=6.28165
r(i=12)=6.28242
r(i=13)=6.2828
r(i=14)=6.28299
r(i=15)=6.28309
Numerischer Wert fuer r bei i=15: 6.28309
Exakter Wert nach anayltischer Berechnung mit Grenzprozess: 6.28319
Absoluter Fehler: 9.58738e-005
Press any key to continue
```



Tragen wir die berechneten Werte für i in einer Grafik zusammen, so erkennen wir, dass der Umfang der Spirale tatsächlich gegen einen Wert konvergiert. Dieser Wert beträgt exakt 2π. Wenn wir n genügend gross wählen, so wird der absolute Fehler immer kleiner. Unsere Grenzwertberechnung scheint also zu stimmen.

Das bedeutet, dass wir den exakten Wert dieser Aufgabe ohne numerische Annäherung berechnen können (mit laufenden Iterationen[1]). Wir werden noch sehen, dass der Begriff der Konvergenz in der Mathematik von fundamentaler Bedeutung ist; vor allem für die Differential- und Integralrechnung.

[1] Iterationen sind Wiederholungen, sogenannte Zyklen. Iteration darf nicht mit Rekursion verwechselt werden! Bei Rekursion ruft sich eine Funktion solange selbst auf, bis eine Abbruchbedingung erfüllt ist. Bei Iterationen erfolgt dies mit den Schleifen while() und for().