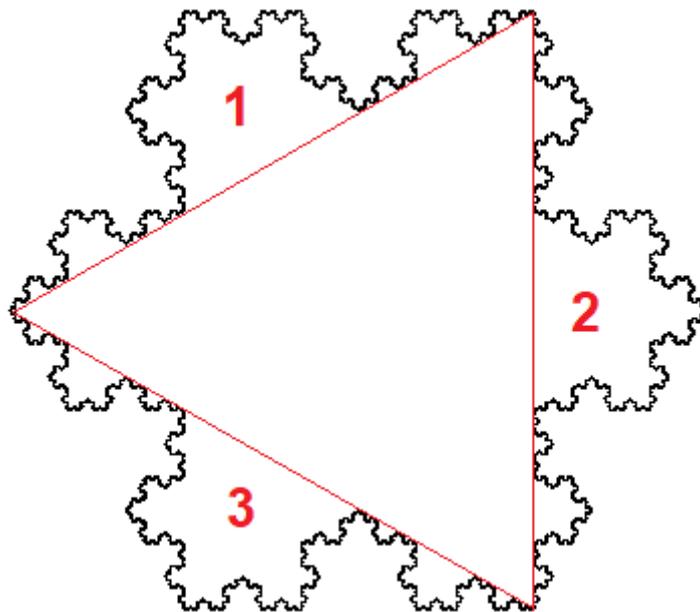


Koch'sche Schneeflocke

Die Kochsche Schneeflocke setzt sich aus drei identischen Teilen zusammen, die mithilfe einer rekursiven Methode gezeichnet werden können:



Jede Seite der Schneeflocke besteht aus einer *Kochkurve*.

0. Iteration



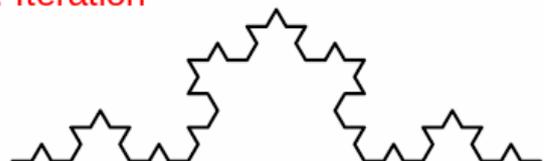
1. Iteration



2. Iteration



3. Iteration



Eine Kochkurve der Stufe n und Länge l kann mithilfe einer rekursiven Methode mit Turtle-Grafik `zeichneKochkurve(int n, int l)` mit Turtle-Grafik gezeichnet werden.



(A1)

- Verwende die Vorlage aus [der Einführung in die Turtle Grafik](#)
- Notiere den Basisfall zum Zeichnen einer Kochkurve der Iteration 0 und Länge l in einer Methode `zeichneKochkurve(int n, int l)` im Quelltext.
- Zeichnen die Kochkurve in der 0. Iteration auf ein Blatt Papier. Zeichne anschließend mit anderer Farbe ein, wo sich die (um $1/3$ verkleinerte) Kochkurve der Iteration 0 in der Kochkurve der Iteration 1 wiederfindet und mit einer weiteren Farbe, wo sich die (wieder um $1/3$ verkleinerte) Kochkurve der Iteration 1 in der Iteration 2 wiederfindet.
- Wie oft muss die Methode `zeichneKochkurve` sich für jede Iteration selbst aufrufen?
- Wie verändern sich die Parameter der Selbstaufrufe.
- Welche Turtle-Befehle müssen jeweils zwischen den Selbstaufrufen ausgeführt werden?
- Vervollständige Quelltext und implementiere die Methode `zeichneKochkurve(int n, int l)`.
- Implementiere die Methode `zeichneKochflocke(int n, int l)`.

Hilfestellungen

Hilfe 1: Basisfall

```
public void zeichneKochkurve(int n, int l) {
    if(n==0) {
        // Zeichne Strich der Länge l
    } else {
        // Anderer Fall
    }
}
```

Hilfe 1: Codegerüst

```
public void zeichneKochkurve(int n, int l) {
    if(n==0) {
        t.forward(l);
    } else {
        // Was muss mit n und l geschehen, wenn man eine Stufe weiter
        hineinzoomt?
        n=FIXME
        l=FIXME
        // Nun müssen drei "kleinere" Kochkurven gezeichnet werden
        zeichneKochkurve(n, l);
        // Was muss zwischendrin passieren?
        FIXME
        zeichneKochkurve(n, l);
        FIXME

        zeichneKochkurve(n, l);
        FIXME
    }
}
```

```
    zeichneKochkurve(n, l);  
}
```

Hilfe 3: Lösungsvorschlag

```
public void zeichneKochkurve(int n, int l) {  
    if(n==0) {  
        t.forward(l);  
    } else {  
        n=n-1;  
        l=l/3;  
        zeichneKochkurve(n, l);  
        t.left(60);  
        zeichneKochkurve(n, l);  
        t.right(120);  
        zeichneKochkurve(n, l);  
        t.left(60);  
        zeichneKochkurve(n, l);  
    }  
}
```

From:
<https://www.info-bw.de/> -

Permanent link:
https://www.info-bw.de/faecher:informatik:oberstufe:algorithmen:rekursion:uebungen02:kochsche_schneeflocke:start?rev=1702307343

Last update: **11.12.2023 15:09**

