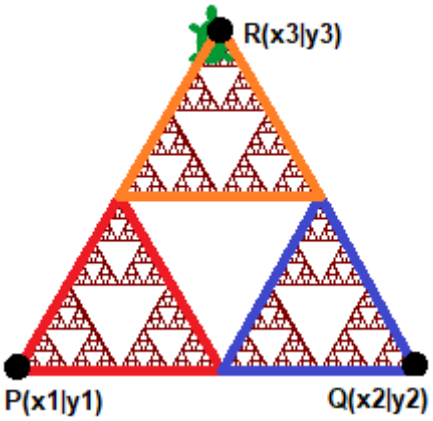
Sierpinski-Dreieck

Gleichseitiges Sierpinski-Dreieck

Ein einfaches Sierpinski-Dreieck setzt sich rekursiv aus drei **gleichseitigen** Dreiecken halber Seitenlängen zusammen solange die Seitenlängen größer als eine minimale Länge `m` sind:

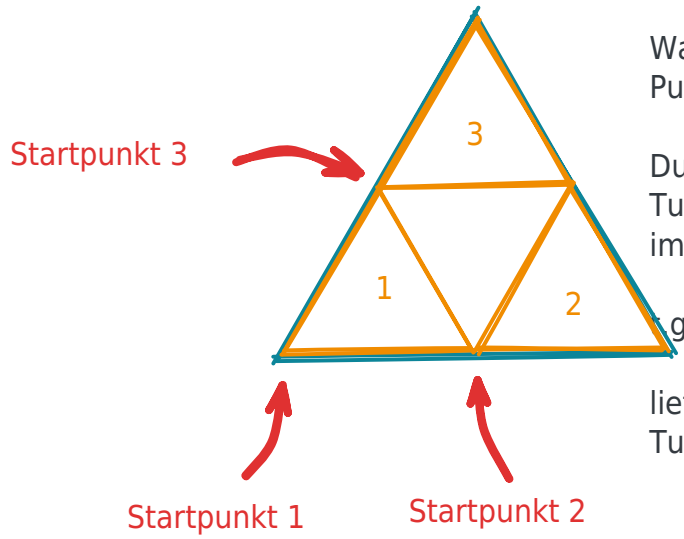


Im Folgenden soll zunächst die Methode `zeichneSierpinskiGleichseitig(int laenge, int m)` implementiert werden. Die Grundseite der Dreiecke soll dabei parallel zur x-Achse ausgerichtet sein.

Verwende weiter die Vorlage aus der Einführung in die Turtle Grafik.

(A1)

- Welche Bedingung ist maßgeblich, dass der Basisfall eintritt. Welche Operation muss der Algorithmus im Basisfall ausführen?
- Wenn der Basisfall nicht zutrifft, müssen drei Dreiecke gezeichnet werden, indem sich die Methode rekursiv selbst aufruft. Vor jedem Aufruf muss der Startpunkt neu berechnet werden und die Turtle mit `t.setPos(X,Y)` an den berechneten Startpunkt gesetzt werden. Mit welchen Parametern muss sich die Methode selbst aufrufen?



Was sind die Koordinaten der bezeichneten Punkte?

Du musst von der aktuellen Position der Turtle aus rechnen, falls deine Reise nicht im Ursprung beginnt!

t.getX() und t.getY()

liefern dir die aktuellen Koordinaten der Turtle t

Hilfestellung (Codegerüst mit Lücken)

```
public void zeichneSierpinskiGleichseitig(int laenge, int m) {
    if(laenge < m) {
        [                    ]
    } else {
        double x=t.getX();
        double y=t.getY();
        zeichneSierpinskiGleichseitig([      ], m);
        t.setPos([      ],[ ]);
        zeichneSierpinskiGleichseitig([      ], m);
        t.setPos([      ],[                ]);
        zeichneSierpinskiGleichseitig([      ], m);
        t.setPos([ ],[ ]);
    }
}
```

Was muss im Basisfall passieren?

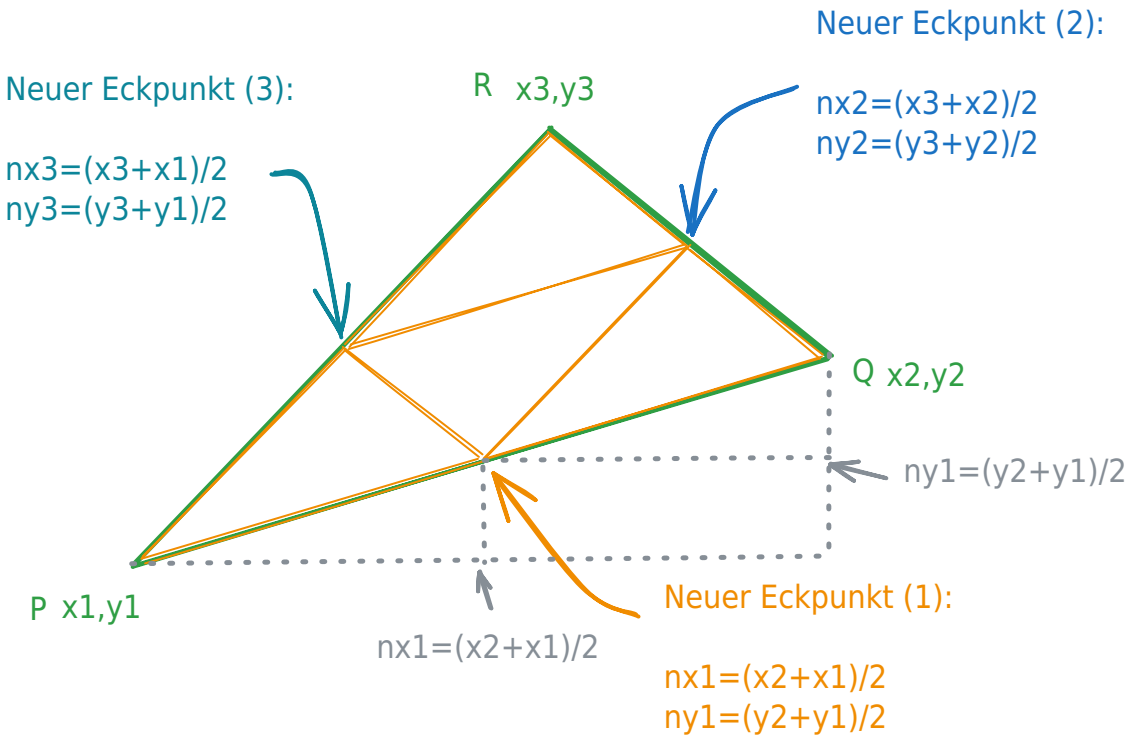
Was muss an den unkenntlich gemachten Stellen eingefügt werden?

Warum benötigt man diese Anweisung? Lass sie weg und teste.

Beliebige Sierpinski-Dreiecke

Wenn man beliebige Sierpinski-Dreiecke zeichnen möchte, ändert sich am Prinzip der Rekursion nichts, allerdings muss man das Dreieck anders beschreiben, beispielsweise durch die Koordinaten seiner Eckpunkte. Es genügt jetzt auch nicht mehr, lediglich die Startpunkte der rekursiv

gezeichneten Dreiecke zu bestimmen, sondern man muss die Eckpunkte dieser Dreiecke bestimmen, das ist etwas aufwändiger. Die Methodensignatur sieh in diesem Fall also so aus: `zeichneSierpinskiBeliebig(x1,y1,x2,y2,x3,y3,m)`



(A2)

- Überlege dir auch hier welche Bedingung den Basisfall definiert.
- Implementiere in deiner Methode, dass das Dreieck mit den Eckpunkten `P(x1|y1)`, `Q(x2|y2)` und `R(x3|y3)` gezeichnet wird.
- Überlege dir, wie man die fehlenden Eckpunkte der orangenen Dreiecke mithilfe der Koordinaten x1, y1, x2, y2, x3 und y3 in der Abbildung bestimmen kann.
- Ergänze deine Methode auf Basis dieser Überlegungen um geeignete Selbstaufrufe und implementiere die Methode. Geeignete Eckpunkte sind z.B. `(0|0) (200|0) (100|174)`, eine geeignete minimale Seitenlänge für diese Koordinaten ist zwischen 5 und 10.

Lösungsvorschlag Codegerüst mit Lücken

```
public void zeichneSierpinskiBeliebig(int x1,int y1,int x2,int y2,int x3, int y3,int m) {

    if(x2-x1 <= m ) {
        t.setPos(  ,  );
        t.moveTo(  ,  );
        t.moveTo(  ,  );
        t.moveTo(  ,  );
    } else {
        // Koordinaten der neuen Ecken auf den
        // Mittelpunkten der Dreiecksseiten
        int nx1=
        int ny1=
        int nx2=
        int ny2=
        int nx3=
        int ny3=
        // Jetzt die drei innen[illegible]cke zeichnen
        zeichneSierpinskiBeliebig(                    );
        zeichneSierpinskiBeliebig(                    );
        zeichneSierpinskiBeliebig(                    );
    }
}
```

Wie zeichnest du jetzt das "elementare" Dreieck?

Wie berechnest du die Koordinaten der Seitenmitten?

Welche sind jetzt die jeweiligen Ecken der drei "rekursiven" Dreiecke?

From:
https://info-bw.de/ -

Permanent link:
https://info-bw.de/faecher:informatik:oberstufe:algorithmen:rekursion:uebungen02:sierpinski_dreieck:start?rev=1738848950

Last update: **06.02.2025 13:35**