

## Grundlagen der Programmierung

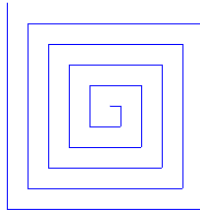
### Übung 2 24.10/07.11.2017

#### Aufgabe 1 (Spinnennetz)

Lade das Snap-Programm aus der letzten Übung (Dreieck, Hexagon). Definiere einen neuen Befehl *Web*, der 5 Hexagons zeichnet, dabei die Seitenlänge der Hexagons jeweils um 15 erhöht.

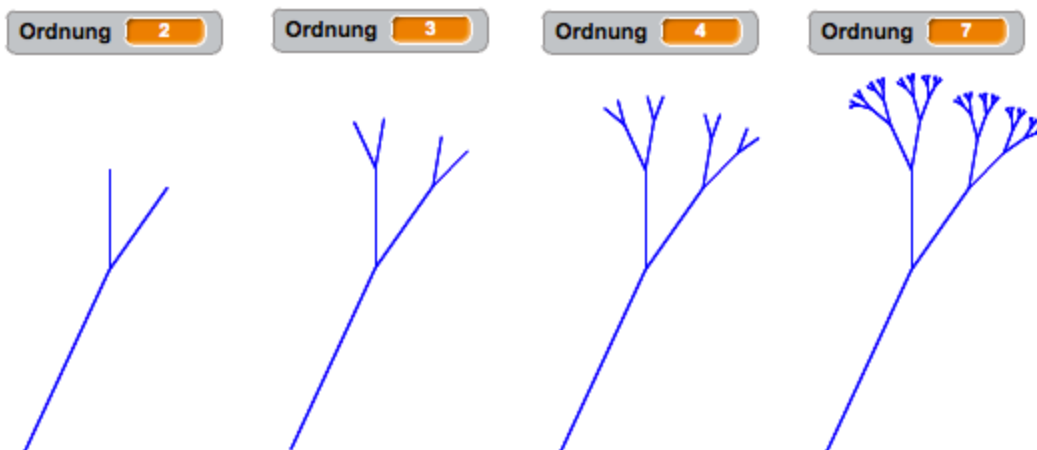
#### Aufgabe 2 (Spirale)

Definiere einen Befehl *Spirale* mit dem Parameter *sides* für die Anzahl der Spiralarms, der eine rechtwinklige Spirale wie in der folgenden Abbildung zeichnet.



#### Aufgabe 3 (Rekursive Graphik, Knobelaufgabe für Fortgeschrittene!)

Versuche, das Schema der Rekursion in folgender Abbildung zu erkennen und programmiere die Rekursion in Snap.

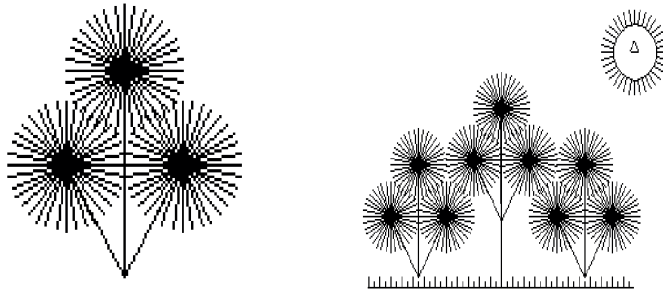


*bitte wenden*

#### Aufgabe 4 (Freiwillige Hausaufgabe)

Experimentiere mit Snap! Hier ein paar Anregungen:

- a) Zeichne das „Haus vom Nikolaus“.
- b) Ein Garten:
  - (a) Beginne mit einem Block *Bluete*, der 36 Strahlen der Länge 25 zeichnet.
  - (b) Definiere mit Hilfe von *Bluete* einen Block *Blume*, der einen Stil der Länge 50 und eine Blüte zeichnet (und zum Anfangspunkt zurückkehrt!).
  - (c) Definiere mit Hilfe von *Blume* einen Block *Strauss*, der 3 Blumen wie in der Abbildung links gezeigt zeichnet.
  - (d) Die Sonne im rechten Bild entsteht, indem zunächst ein Kreis mit dem Radius 20 gezeichnet wird. Danach werden 36 Strahlen gezeichnet, die jeweils um 10 Grad nach rechts gedreht sind. Die einzelnen Strahlen bestehen jeweils aus einem unsichtbaren Strich der Länge 20 und einem sichtbaren der Länge 10. Ein Block zum zeichnen eines Kreises findest du auf der Homepage zur Vorlesung.
  - (e) Entwirf weitere Blöcke, z.B. einen Rasen und erzeuge daraus ein Bild wie in der linken Abbildung.



- c) Eine Uhr:
  - (a) Definiere zwei Blöcke *Kleiner Zeiger* und *Grosser Zeiger*, die die Zeiger einer Uhr z.B. als Trapeze zeichnen. (Oder: Definiere einen Block *Zeiger* mit einem Parameter für die Größe des Zeigers).
  - (b) Definiere damit einen Block *Uhr* mit zwei Parametern, *stunde* und *minute*, die die Zeiger entsprechend dreht und die Uhrzeit anzeigt.
  - (c) Programmiere eine bewegende Uhr. Tip: Stiftfarbe auf schwarz setzen, Zeiger zeichnen, warten, Stiftfarbe auf weiss setzen, Zeiger erneut zeichnen, die nächsten Zeigerpositionen berechnen
- d) Schreibe ein Programm, dass zwei Widerstände in Parallelschaltung zeichnet.
- e) Auf den Webseiten zu Scratch und Snap finden sich auch viele Beispiele, z.B. das Spiel *Pong* mit Scratch programmiert ([scratch.mit.edu/projects/18990417/](http://scratch.mit.edu/projects/18990417/)).