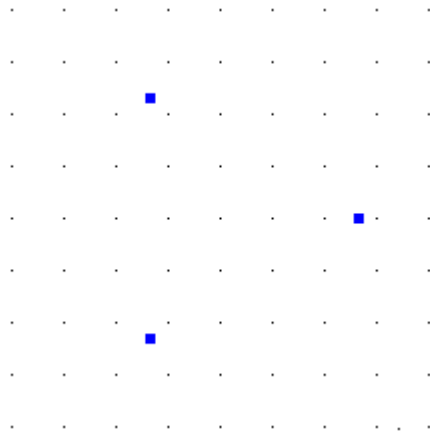


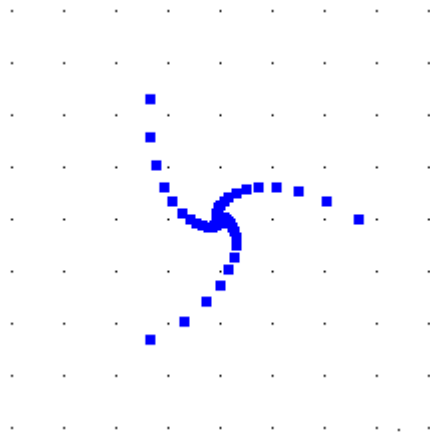
Und laufen, und laufen, und laufen ...

Das Problem: In den Eckpunkten eines reelmäßigen Dreiecks sitzen gleich schnelle Käfer. Nach dem Startsignal läuft jeder in die Richtung, in der er seinen Vordermann sieht.
Was wird passieren?

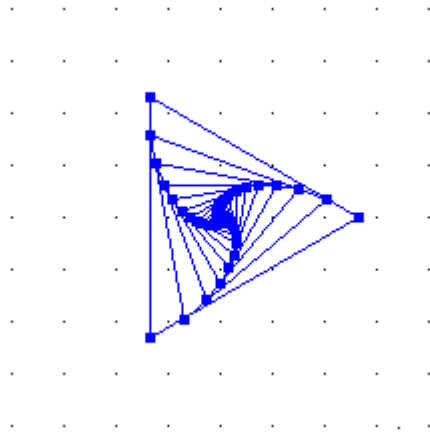


Auf die Plätze - fertig - los! Zeichne #1 in einem 2-D-Graphik-Fenster!
(Einstellungen: Einstellen - Verzerrungsverhältnis ... - 1 : 1
Extras - Approximieren vor dem Zeichnen
Extras - Anzeige - Punkte ... - Verbinden: *Nein*, Größe: *Groß*)

#1: `Rennen_slow(3, 10, 15, 3)`



Wenn die Punkte verbunden werden, so werden auch die Sehstrahlen der Käfer dargestellt:



Wir vereinbaren:

- Der Ursprung des Koordinatensystems liegt dort, wo die Käfer zusammenstoßen.
- Die Momentaufnahmen werden im Abstand von z.B. 10° gezeichnet.

Drei-Käfer-Rennen

- Stelle eine Koordinatenmatrix zum Zeichnen des Ausgangsdreiecks auf.
- Wende darauf eine Drehmatrix für eine Drehung um 10° an:
- Zeige mit dem Sinussatz, dass $\sin(30^\circ) / \sin(140^\circ)$ der benötigte Streckfaktor ist.
- Multipliziere das vorherige Ergebnis mit dem Streckfaktor.
- Zeichne weitere Dreiecke.
- Verwende den vector-Befehl zur Automatisierung.

Vier-Käfer-Rennen

- Übertrage die Ergebnisse auf das Rennen mit vier Käfern:

n-Käfer-Rennen

- Definiere Abbildungen für Drehung, Streckung, Hintereinanderausführung von Drehung und Streckung in Abhängigkeit von der Anzahl der Ecken n und vom Drehwinkel φ .
- Definiere eine Funktion $\text{Poly}(n)$ zur Erzeugt der Koordinatenmatrix der Eckpunkte eines regelmäßigen n -Ecks.
- Nun können n Käfer rennen.

Abbildungen für die Eingangsbeispiele:

#2: Kreis(M_-, r_-) := $(x - M_{-1})^2 + (y - M_{-2})^2 = r_-^2$

#3: Delay(Laufvar, Dauer) := VECTOR(Kreis([Laufvar, -100], $k_- \cdot 0.01$), k_- ,
1, Dauer)

#4: Rennen_slow($n, \varphi, stop, Dt$) := VECTOR($\left[\left(\text{DrehStreck}(n, \varphi) \cdot \text{Poly}(n) \right) \right]$, $k, 0, stop$)