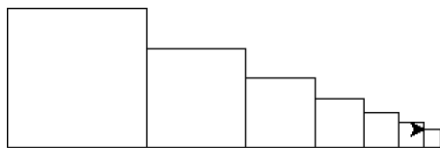


Einfache Aufgaben

1. Schreiben Sie ein Unterprogramm, das Quadrate zeichnet. Der Nutzer gibt eine Startseitenlänge ein, daraufhin zeichnet das Programm ein Quadrat und ein Folgequadrat, dessen Diagonale so lang ist, wie die Seitenlänge des vorherigen Quadrates. Überlegen Sie sich selbstständig ein Abbruchkriterium und recherchieren Sie, wie sie die Seitenlängen berechnen (Mathematik!)

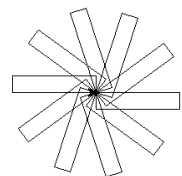


2. Schreiben Sie ein Programm, das die empirische Varianz einer Liste von Zahlen berechnet.
3. Schreiben Sie ein Programm, das die Koordinaten zweier Punkte einliest, diese Punkte zeichnet und ihren Abstand berechnet.

Zusatz: Erweitern Sie ihr Programm, so dass ein dritter Punkt eingelesen wird und überprüft wird, ob dieser auf der Geraden der ersten beiden Punkte liegt (Punktprobe).

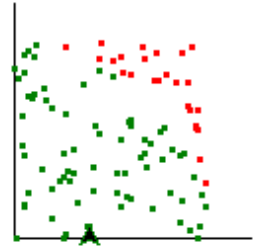
4. Herr D. aus D. ist mit Abstand der faulste Mathematiklehrer im Lande. Er soll für einen Schüler folgenden Term berechnen: $100^2 - 99^2 + 98^2 - 97^2 + \dots - 3^2 + 2^2 - 1^2$. Nun möchte er das ungern in den Taschenrechner eintippen, das Tippen wäre ja tatsächlich mit Arbeit verbunden.
 - a) Schreiben Sie ein Programm, das die dargestellte Summe berechnet (Die Eingabe als Gesamtterm ist keine akzeptable Lösung!).
 - b) Die Summe startet mit der Zahl 100. Erweitern Sie ihr Programm, so dass ein Nutzer den Startwert eingeben kann. Gibt ein Nutzer die Zahl 200 ein, so soll das Programm folgenden Term berechnen: $200^2 - 199^2 + \dots - 3^2 + 2^2 - 1^2$. Schreiben Sie dafür ein geeignetes Unterprogramm.

5. Schreiben Sie ein Programm, das ein Windrad grafisch darstellt. Das Windrad bestehe aus n Rechtecken der Länge x und Höhe $\frac{1}{5}x$, wobei n und x vom Nutzer eingegeben werden können. Nutzen Sie mindestens ein Unterprogramm. Speichern Sie ihr Programm unter dem Dateinamen:



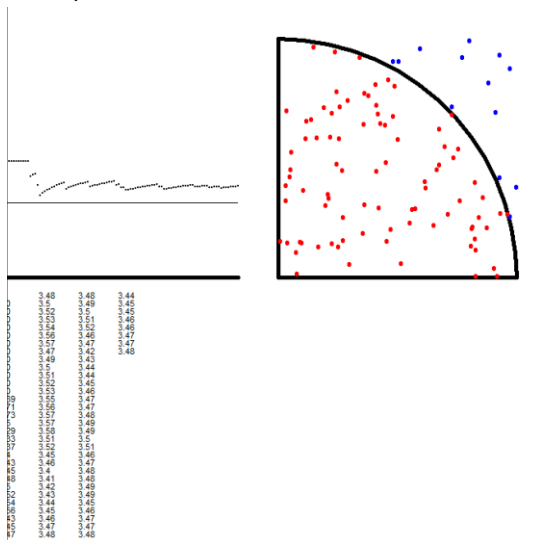
Approximation von Pi mithilfe einer Montecarlo Methode

Es werden in einem Quadrat Zufallszahlen für die x- und y-Koordinate eines Punktes zwischen 0 und 1 erzeugt. Dann wird gezählt, wie viele Punkte davon in einen Viertelkreis fallen. Das Verhältnis Anzahl der Punkte im Viertelkreis zu Anzahl der Punkte insgesamt multipliziert mit der Zahl 4 liefert etwa die Kreiszahl π . Die Näherung für π ist um so besser, je mehr Zufallspunkte erzeugt worden sind.



Aufgabe:

- 1) Schreiben Sie ein Programm, das mithilfe dieser Simulation eine Näherung für π berechnet.
- 2) Begründen Sie, wie das Verfahren funktioniert (das ist eine Mathematik Aufgabe).
- 3) Visualisieren Sie den Verlauf der Berechnung mithilfe einer geeigneten Turtle Grafik (Beispiel anbei).



Tipp:

Für die Entscheidung, ob der Punkt im Zufallskreis liegt, benötigt man einen wichtigen Satz der Mathematik, den Satz des

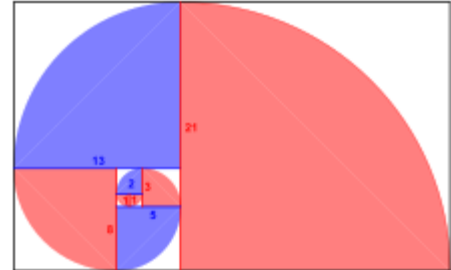
Fibonacci

Die Fibonacci-Folge ist lautet wie folgt:

0, 1, 1, 2, 3, 5, 8, 13, ...
(optional) $0+1$ $1+1$ $1+2$ $2+3$ $3+5$ $5+8$

Das Bildungsgesetz ist dabei sehr einfach. Die „Startwerte“ 0 und 1 sind gegeben. Jedes Folgeglied berechnet sich dann dadurch, dass die Summe der beiden vorhergehenden Folgeglieder bestimmt wird.

Die Fibonacci Folge hat trotz ihres einfachen Bildungsgesetzes viele interessante Eigenschaften¹



Aufgabe

- 1) Schreiben Sie ein Programm, das die ersten n Glieder der Fibonacci Folge ausgibt. Diese sollen in einer Liste gespeichert werden.
- 2) Erweitern Sie ihr Programm, so dass fortwährend die Quotienten aufeinanderfolgender Fibonacci Glieder bestimmt werden. Was stellen Sie fest?

¹ Ein Blick in den Wikipedia Eintrag ist interessant.

Random Walk

Problem

Ein Betrunkener verlässt die Kneipe. Wir nennen ihn Dr. Drunkenstein. Seine kognitiven Fähigkeiten sind vollends zum Erliegen gekommen und seine motorischen Fähigkeiten sind nur noch dem Zufall unterworfen.

Er kann sich nur noch nach Norden, Süden, Westen oder Osten bewegen. Die Wahl seiner Richtung in jedem Schritt ist dabei – wie angemerkt – ausschließlich zufällig. Er entscheidet sich bei jedem Schritt zufällig für genau eine Richtung.

Sollten Sie in einem Schritt nicht weiterkommen, dann können Sie fragen. Sie erhalten dann gegebenenfalls keine Punkte für diese Teilaufgabe. Die Aufgabe ist so konzipiert, dass Sie nacheinander ein komplexes Programm schreiben. Sie benötigen nicht für jede Teilaufgabe ein eigenes Programm.

Geben Sie am Ende Ihre Programmlösung ab - das Ergebnis wird bewertet.

- 1) Beschreiben Sie, wie Sie die Wahl der zufälligen Richtung mit Python simulieren können. Nutzen Sie hierfür die Kommentarfunktion von Python (ein # am Anfang der Zeile). (3)
- 2) Simulieren Sie zunächst einen zufälligen Schritt von Dr. Drunkenstein. Geben Sie die Richtung, für die er sich entschieden hat aus. (4)
- 3) Beschreiben Sie, wie Sie die zurückgelegte Entfernung von einem Start zu einem Endpunkt berechnen können. Nutzen Sie wieder die Kommentarfunktion. (4)
- 4) Simulieren Sie nun 1000 Schritte des Betrunkenen. Berechnen Sie die Entfernung zum Startpunkt nach den 1000 Schritten. (4)

Tipp: Eine Position kann man durch eine x und eine y Koordinate beschreiben. Nutzen Sie diese Idee zur Speicherung der Position oder Positionsänderung und zur Berechnung der Entfernung.

- 5) Visualisieren Sie den Weg mithilfe der Turtle-Grafik. Sie können davon ausgehen, dass es keine Hindernisse gibt. Eine Möglichkeit ist rechts dargestellt (Ausschnitt). (4)



- 6) Zu Dr. Drunkenstein gehört eine Mrs. Drunkenstein. Diese möchte nun gerne wissen, wo sich ihr Mann befindet (oder zumindest wie weit weg von der Kneipe er ist).

Simulieren Sie 1000 Versuche mit je 1000 Schritten und berechnen Sie die durchschnittliche Entfernung vom Startpunkt. Die 1000-mal 1000 Versuche sollen nicht gezeichnet werden. (5)

Tipp: Jede der 1000 Durchführungen besteht aus 1000 Schritten. Berechnen Sie nach jeder Durchführung die durchschnittliche Entfernung (siehe Aufgabe 3!) und speichern Sie diese in einer geeigneten Datenstruktur. Berechnen Sie danach den „Durchschnitt aller Durchschnitte“.