

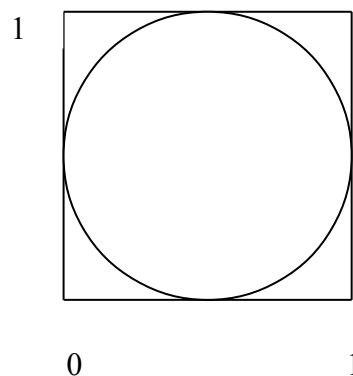
Dienstag, 6. Mai 2014

Programmierung 2

SS 2014

Aufgabe 5: **Gruppe 1:** 27.05.2014 **Gruppe 2:** 20.05.2014

Folgende einfache Überlegung führt zu einem probabilistischen Verfahren zur Berechnung der Zahl π :



Für die Fläche F_Q des Einheitsquadrates, die Fläche F_K des Kreises und die Restfläche $F_R = F_Q - F_K$ gilt offenbar:

$$F_Q = 1$$

$$F_K = \pi * (1/2)^2 = \pi/4$$

$$F_R = 1 - \pi/4 = (4 - \pi) / 4$$

Man wähle nun mit einem Zufallsmechanismus Punkte aus dem Einheitsquadrat und zähle, wie viele der gewählten Punkte im Kreis (Anzahl A_K) bzw. in der Restfläche liegen (Anzahl A_R). Nach dem Bernoullischen Gesetz der großen Zahlen werden sich die Anzahlen A_K und A_R zueinander verhalten, wie die Größe der Flächen F_K zu F_R , d.h. es wird

$$A_K / A_R$$

stochastisch gegen

$$F_K / F_R = (\pi/4) / ((4 - \pi) / 4) = \pi / (4 - \pi)$$

konvergieren. Durch Auflösen dieser Beziehung nach π folgt, dass

$$(4 * A_K) / (A_R + A_K)$$

stochastisch gegen π konvergiert, d.h. dass bei einer großen Anzahl zufällig gewählter Punkte aus dem Einheitsquadrat der leicht zu ermittelnde Wert von

$$A = (4 * A_K) / (A_R + A_K)$$

in etwa π sein muss.

Schreiben und dokumentieren Sie ein objektorientiertes Programm, das dieses Verfahren realisiert. Es soll Folgendes leisten:

- Die Benutzer oder Benutzerinnen sollen über die Tastatur eingeben können, wie genau π durch **A** approximiert werden soll, d.h. wie groß der Abstand **d** von π zu **A** maximal sein darf ($0.00000000001 \leq d \leq 0.1$).
- Das Programm soll so lange zufällig Punkte aus dem Einheitsquadrat wählen, bis der Abstand von π zu **A** kleiner als **d** ist.
- Als Ergebnis sollen auf dem Bildschirm folgende Informationen erscheinen:

Wie gross darf die Abweichung zu Pi sein? [0.00000000001 - 0.1] 0.00000000001

Bitte warten, ich arbeite ...

PI probabilistisch mit 133957642 Punkten:

Punkte im Kreis:	105210086
Punkte nicht im Kreis:	28747556

Näherungswert:	3,141592653594186
Pi:	3,141592653589793

erlaubte Abweichung:	0,000000000010000
tatsächliche Abweichung:	0,000000000004393

Dauer:	7810279000	Nanosekunden
entspricht:	7,810279000000000	Sekunden

- Das Programm soll so lange zur Verfügung stehen, bis dies vom Benutzer oder von der Benutzerin explizit nicht mehr gewünscht wird. Es soll benutzerfreundlich sein.

[Hinweise:

- Die Eingaben sollen auf Plausibilität überprüft werden (Wertebereich). Das Programm soll weitgehend tolerant sein gegenüber Fehleingaben.
- Achten Sie darauf, dass Ihr Programm objektorientiert ist. Sehen Sie mehrere (sehr einfache) Klassen vor (z.B. **PiSimulator**, **Punkt**, **Stoppuhr**, etc.).
- Benutzen Sie die Konstante **PI** der Klasse **Math** zum Vergleich. Nutzen Sie die Klasse **System**.
- Auf den Rechnern des Labors sind (in dieser Reihenfolge) zu präsentieren:

- die mit Hilfe von **javadoc** erzeugte (HTML-)Dokumentation,
- die Java-Dateien,
- die Übersetzung des Programms,
- die Ausführung des Programms.

]

