

Näherungsverfahren

- Wiederhole den Algorithmusbegriff.
- Erläutere die Begriffe:
 - Klasse der P-Probleme
 - Klasse der NP-Probleme
 - Approximative Algorithmen
 - Stochastische Algorithmen

ALGORITHMEN

Def.: Eine Bearbeitungsvorschrift heißt **Algorithmus**, wenn sie folgende Eigenschaften erfüllt:

1. Die Vorschrift ist mit endlichen Mitteln beschreibbar.
2. Sie liefert auf eine eindeutig festgelegte Weise zu einer vorgegebenen Eingabe in endlich vielen Schritten genau eine Ausgabe.

Aus der Definition folgt:

- Die Verfahrensbeschreibung muss an jeder Stelle eindeutig festlegen, welcher Bearbeitungsschritt als nächstfolgender auszuführen ist.
- Jeder Bearbeitungsschritt in der Folge muss auch ausführbar sein.
- Die Verfahrensbeschreibung muss eine endliche Länge besitzen.
- Jeder Bearbeitungsschritt muss festlegen, was zu tun ist und womit (mit welchen Daten) der Schritt durchzuführen ist.

Da man nicht alle Bearbeitungsvorschriften durch eindeutige Folgen von Anweisungen als Algorithmus darstellen kann, wurde im Laufe der Zeit der Begriff ausgedehnt. Man unterscheidet heute zwischen **deterministischen** und **nicht-deterministischen** Algorithmen.

- Enthält ein Algorithmus elementare Anweisungen, deren Ergebnis durch einen Zufallsmechanismus beeinflusst wird, so heißt dieser Algorithmus **nicht-deterministisch**.
- Liefert er bei der gleichen Eingabe immer dieselbe Ausgabe, so heißt er **deterministisch**.

NÄHERUNGSVERFAHREN

Zu den Aufgaben der **theoretischen Informatik** gehört

- ▶ das Finden von Algorithmen, die ein gegebenes Problem **deterministisch** in (mindestens) **polynomialer Zeit** lösen bzw.
- ▶ der Nachweis, dass es einen solchen Algorithmus für ein Problem **nicht** gibt.

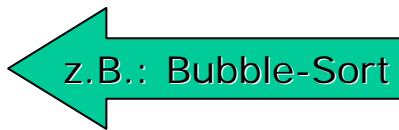
Die Probleme, für die es einen solchen Algorithmus gibt, werden in der Klasse der **P-Probleme** zusammengefasst. Die anderen gehören zur Klasse der **NP-Probleme**.

In **polynomialer Zeit** bedeutet (**unexakt**):

Es sei n das Maß für die Größe eines Problems (z.B. Sortieren $\rightarrow n$ =Anzahl der zu sortierenden Elemente).

Die Anzahl der Arbeitstakte A zur Lösung des Problems lässt sich mit einem Polynom in n berechnen:

$$A(n) = \sum_{i=1}^{n-1} i$$



$$A(n) = \sum_{i=0}^k c_i n^i \text{ mit } i, k \in \mathbb{N}; c_i \in \mathbb{R}$$

s.a. „Informatik bis zum Abitur“ S. 461 ff.

Gibt es nun für die Lösung eines Problems keinen effizienten Algorithmus, dann kommen die Näherungsverfahren in's Spiel.

Approximative Algorithmen

Problemlösung von „Grund auf“ finden

Numerische Mathematik hat zu zeigen, dass Ergebnis Lösung ist →
z.B. BANACHscher Fixpunktsatz

- ✓ Newton-Verfahren
- ✓ Regula-falsi
- ✓ Trapezmethode
- ✓ ...

Eine mit einem anderen Algorithmus gefundene Lösung verbessern

- Neuronale Netze
- Genetische Algorithmen
- ...

stochastische Algorithmen

Arbeiten mit Pseudozufallszahlen

numerische

Newton-Verfahren mit zufälligem Startwert

Las-Vegas-Algorithmen

Liefern ein richtiges Ergebnis oder finden keines

Monte-Carlo-Algorithmen

Die gefundene Lösung ist mit einer vorgegebenen Wahrscheinlichkeit falsch

Monte-Carlo-Methode zur Bestimmung eines Näherungswertes für π

Monte-Carlo-Methode :

Simulation zur näherungsweise Bestimmung von Größen unter Verwendung stochastischer (zufälliger) Komponenten

Praxis:

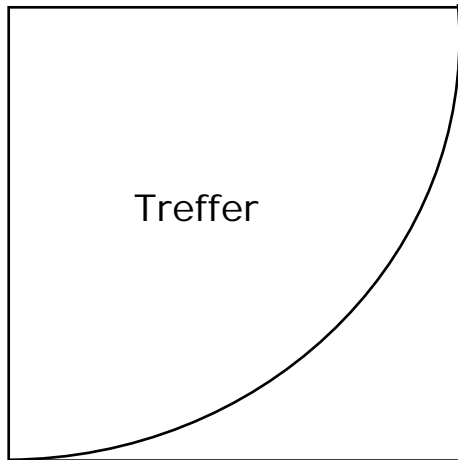
Physik → Verhalten von Vielteilchensystemen

Wirtschaft → Produktionsabläufe – Ersatzteilbereithaltung

Mathematik → Berechnung von Integralen

Börse → Voraussagen über Entwicklung des Aktienkurses

Informatik → Softwaretests



- (1) Es werden zufällig n Punkte erzeugt, die im Quadrat liegen.
- (2) Die Punkte, die im Viertelkreis liegen, werden als Treffer t gezählt.
- (3) Für großes n „füllen“ die Treffer die Fläche des Viertelkreises und alle Punkte die des Quadrates.

Wie lässt sich ein Näherungswert $\tilde{\pi}$ aus n und t berechnen ?

$$\frac{A_{\text{VK}}}{A_{\text{Q}}} = \frac{\frac{\pi}{4} r^2}{r^2} = \frac{\pi}{4} \quad \Rightarrow \quad \frac{t}{n} = \frac{\pi}{4}$$

$\tilde{\pi} = \frac{4t}{n}$