

# Grundlagen der Informatik

Prof. Dr. Stefan Enderle

NTA Isny

# Einführung

# Was ist Informatik ?

- Wort kommt wahrscheinlich von Information + Mathematik
- Informatik ist die Wissenschaft der Informationsverarbeitung.

# Inhalt von „Informatik“

- Bemerkung: „Computer“ war nicht in der Definition!
- Inhalte:
  - Programmiersprachen
  - Programmiertechnik (Software-Engineering)
  - Software-Projektmanagement
  - Abstrakte Maschinen
  - Berechenbarkeit und Komplexität
  - Unterbereiche: Datenbanken, Compilerbau, Künstliche Intelligenz, Computergrafik, ...

# Einteilung

- Meist wird die Informatik in folgende Bereiche eingeteilt:
  - Technische Informatik
  - Praktische Informatik
  - Theoretische Informatik
- „Angewandte Informatik“:  
Untersuchung in unterschiedlichen Bereichen, ob Abläufe automatisierbar sind, z.B. Medizin, Naturwissenschaften, Recht, Industrie, Unternehmensabläufe,...

# Technische Informatik

- Konzeption und Aufbau von Rechnern, auf denen Programme ablaufen können.
- (Von Elektronik bis Betriebssystem)

# Praktische Informatik

- Entwicklung von Programmiersprache und Systemen zur Unterstützung der Programmierung
- (Editor, Compiler, Debugger, Windows,...)

# Theoretische Informatik

- Entwicklung von formalen Modellen und mathematischen Methoden zur Formulierung und Untersuchung von Algorithmen.

# 1. Grundbegriffe

# 1.1 Algorithmus

# Vom Problem zum Programm

- Aufgabe:

Eine rechteckige Fläche mit den Seitenlängen 100m und 24m soll mit möglichst großen quadratischen Kacheln ausgelegt werden.

Wie groß können die Kacheln sein?

# 1. Abstraktion

- Abstraktion:
  - Formalisierung: Beschreibung soll auch für andere Anwendungen anwendbar sein
  - Verallgemeinerung: Auch für beliebige andere Zahlen nutzbar
- Formal:
  - Seitenlängen seien natürliche Zahlen  $a$  und  $b$
  - Gesucht ist der größte gemeinsame Teiler von  $a$  und  $b$

# 2. Algorithmus

- „Euklidischer Algorithmus“:

ggT(a,b):

solange  $a \neq b$ , tue folgendes

wenn  $a > b$ , dann setze  $a := a - b$

wenn  $b > a$ , dann setze  $b := b - a$

gib  $a$  als Ergebnis zurück

# 3. Implementierung

- Beispiel: Umsetzung nach C/C++:

```
int ggT(int a, int b)
{
    while (a!=b)
    {
        if (a>b) a=a-b;
        if (b>a) b=b-a;
    }
    return a;
}
```

# 4. Realisierung

- Übersetzen des C/C++ Programmes mit einem Compiler
- Erzeugen von Maschinensprache
- Ausführen auf dem Rechner

- Die intellektuell schwierigen Schritte sind:
  - Abstraktion
  - Algorithmus
- Implementierung ist Handwerk
- Realisierung ist automatisiert

# Algorithmus

- Definition:

Ein Algorithmus ist eine Vorschrift zur Lösung einer Aufgabe, die so präzise formuliert ist, dass sie maschinell ausgeführt werden kann.

- Im Alltag: Kochrezept, Bedienungsanleitung, Multiplikation von Hand

- Satz:

Ein Algorithmus ist ein abstraktes Objekt,  
unabhängig von

- der Programmiersprache
- dem Computertyp
- der aktuellen Technologie

- Eigenschaften eines Algorithmus:
  - Beschreibung durch endlichen Text  
(*Finitheit*)
  - Ablauf in einzelnen definierten Schritten  
(*Effektivität*)
  - Eindeutigkeit des nächsten Schrittes  
(*Determiniertheit*)
  - Ende nach endlich vielen Schritten  
(*Terminiertheit*)

## 1.2 Flußdiagramm

- Ein Flußdiagramm ist eine grafische Methode zur Beschreibung von Algorithmen

# Grundelemente

- Start
- Stop
- Anweisung
- Fallunterscheidung
- Ein- / Ausgabe
  
- Pfeile zeigen den Programmablauf an

# Beispiel: ggT

- (Tafel)