

Thema

„Programmiersprachen“

Klassifizierung nach Generationen

1. Generation:
Maschinensprachen
2. Generation:
Assembler-Sprachen
3. Generation:
Prozedurale Sprachen
4. Generation:
4GL (Generation Language)
5. Generation:
Künstliche Intelligenz

Maschinensprachen

Anweisungen in binärer Form

Bitfolge 1 oder 0

Hardware abhängig

schwer zu lesen

unübersichtlich

hoher Programmieraufwand

$3 + 4 \rightarrow 00011010 \ 0011 \ 0100$

Assembler-Sprachen

einfache, aussagekräftige Operationskürzel

Hardware abhängig

Übersetzung in Maschinencode mit dem Programm Assembler

verbrauchen weniger Speicherplatz

schneller als Programme in anderen Programmiersprachen

3 + 4 -> ADD 3, 4

Prozedurale Sprachen

unabhängig von Computersystem

höhere Programmiersprache

auf bestimmte Anwendungsbereiche zugeschnitten

leichter zu erlernen

verbrauchen mehr Speicherplatz

langsamer als Maschinenprogramme

3 + 4 ⌚ Summe = 3 + 4

Cobol, Basic, C, C++, Java ...

4GL Generation Language

beschreiben was das Programm leisten soll, ohne genaue Schritte zu formulieren

eine Anweisung löst eine Folge von Schritten aus

auf spezielle Anwendungsgebiete ausgerichtet (Datenbank, Tabellenkalkulation)

langsamer als prozedurale Sprachen

SQL, Natural ...

```
SELECT Kunde FROM Tabelle
```

Künstliche Intelligenz

Probleme werden durch Rand- und Zwangsbedingungen beschrieben und nicht mehr explizit gelöst

Untersuchung ob Computer menschliche Verhaltensweisen nachvollziehen können

Robotik, Wissensverarbeitung, Spracherkennung

Prolog, Lisp ...

Klassifizierung nach Sprachtypen

Prozedurale Programmiersprachen

Objektorientierte Programmiersprachen

Funktionale Programmiersprachen

Logische Programmiersprachen

Prozedurale Programmiersprachen

höhere Programmiersprachen

eine Folge von Anweisungen, die nacheinander abgearbeitet werden

Gesamtaufgabe ist in kleinere Teilaufgaben aufgelöst

Programmcode kann wieder verwendet werden

C, Cobol, Basic, Pascal ...

Objektorientierte Programmiersprachen

Weiterentwicklung von prozeduralen Programmiersprachen

Programme werden in Einheiten (Objekte) unterteilt

Funktionen (Methode) und Daten (Attribute) werden im Objekt zusammengefasst

nur eigene Funktionen können Daten im Objekt ändern
(Kapselung)

Wiederverwendung von Softwareteilen

Smalltalk, C++, C#, Java

Funktionale Programmiersprachen

Reihe von Funktionsdefinitionen

Hauptprogramm ist eine Funktion, die weitere Funktionen verwendet

Funktion hat Definitions- und Wertebereich

werden im Bereich der künstlichen Intelligenz, Compilerbau, Computeralgebra eingesetzt

LISP, XSLT ...

Logische Programmiersprachen

Aufbau einer Datenbasis, die aus Fakten und Regeln besteht

Problemformulierung im Vordergrund

Bedingungen für korrekte Lösung werden bestimmt

Programm berechnet eine Beziehung zwischen den einzelnen Daten

Forschungsgebiet Künstliche Intelligenz

PROLOG

Kurzvortrag

Eine Programmiersprache Ihrer Wahl
vorstellen!

Welche Generation?

Welcher Sprachtyp?

Entwicklung?

Einsatz?

Vor-und Nachteile?

(ca. 5 min)

Vielen Dank für Ihre
Aufmerksamkeit!