



Script generated by TTT

Title: Einf_HF (01.06.2015)

Date: Mon Jun 01 14:14:46 CEST 2015

Duration: 91:02 min

Pages: 34

"Kunst des Programmierens". Grundlagen zu Datenstrukturen, Programmkonstrukte, Strukturierung von Programmen, objekt-orientierte Programmierung.

- Fragestellungen des Abschnitts:
 - Was ist ein Algorithmus?
 - Welche elementaren Datenstrukturen gibt es?
 - Was sind die grundlegenden Konstrukte einer Programmiersprache?
 - Was ist unter Objekt-orientierter Programmierung zu verstehen?
 - Was versteht man unter Modularisierung und Rekursion?

Einführung

Algorithmus

Datentypen und Ausdrücke

Programmkonstrukte

Objektorientierte Programmierung

Modularisierung von Programmen

Rekursion

Generated by Targeteam



Einführung



Programmierung: Vorgang, in dem Anweisungen für den Computer erstellt werden. Aufgeschrieben in für Menschen verständlichen Sprachen, für Computer in Maschinsprache übersetzt (z.B. C) oder mit Interpreterprogramm interpretiert (z.B. Javascript).

Programm = Daten + Algorithmus

Algorithmus legt die Operationen fest, die auf den Daten ausgeführt werden.

Sicht des Programmierers

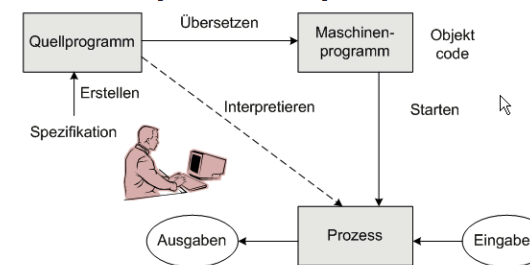
Generated by Targeteam



Sicht des Programmierers



Ablauf der Programmentwicklung



Quellprogramm

Ausgangsform, z.B. in C oder Java.

Übersetzen (Compiler)

Erzeugen eines ausführbaren Objektcodes.

Objektcode - Maschinenprogramm

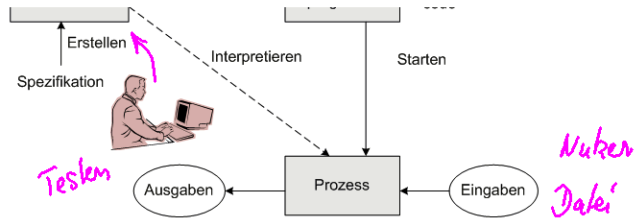
Interdarstellung des Programms, für Computer verständlich (Maschinenprogramm). "Linker": aus mehreren Einzelteilen zusammensetzen, Quellprogramm besteht aus mehreren Dateien.

Starten

zum Ausführen eines Objektcodes.

Prozess





Quellprogramm

Ausgangsform, z.B. in C oder Java.

Übersetzen (Compiler)

Erzeugen eines ausführbaren Objektcodes.

Objektcode - Maschinenprogramm

Interndarstellung des Programms, für Computer verständlich (Maschinenprogramm). "Linker": aus mehreren Einzelteilen zusammensetzen, Quellprogramm besteht aus mehreren Dateien.

Starten

zum Ausführen eines Objektcodes.

Prozess

Ausführung eines Objektcodes; dabei durch Betriebssystem zusätzliche Daten erzeugt. Testläufe, bei Fehlverhalten Quellprogramm korrigieren.

Interpretieren (Interpreter)



Detaillierte, explizite Vorschrift zur schrittweisen Lösung eines Problems, d.h. die

- präzise formuliert,
- in endlicher Form dargestellt und
- effektiv ausführbar ist.

Die Aufgabe kann aus vielen Anwendungsgebieten stammen, z.B. Berechnung numerischer Werte, Text- und Bildverarbeitung, Handhabung von Objekten (Robotik), Zuteilung von Ressourcen, Steuerung von Geräten.



Programmierung: Vorgang, in dem Anweisungen für den Computer erstellt werden. Aufgeschrieben in für Menschen verständlichen Sprachen, für Computer in Maschinsprache übersetzt (z.B. C) oder mit Interpreterprogramm interpretiert (z.B. Javascript).

Programm = Daten + Algorithmus

Algorithmus legt die Operationen fest, die auf den Daten ausgeführt werden.

Sicht des Programmierers



Nach Spezifikation: Lösungsweg entwerfen. Da von Rechner auszuführen, jeden Schritt exakt vorschreiben. Algorithmus.

eine der ältesten Beschreibungstechniken für Abläufe; benannt nach dem Mathematiker Al-Khwarizmi (ca. 780-850), der am Hofe des Kalifen von Bagdad wirkte.

Informelle Charakterisierung eines Algorithmus

Formale Definition (nach H. Kübe):

Formulierung eines Algorithmus



Ein Algorithmus ist eine in der Beschreibung und Ausführung endliche, deterministische und effektive Vorschrift zur Lösung eines Problems, die effizient sein sollte.

Hierin bedeuten:

endlich in Beschreibung : der Algorithmus kann mit endlich viel Text komplett aufgeschrieben werden.

endlich in Ausführung : nach einer endlichen Zeit wird der Algorithmus beendet.

deterministisch : eindeutige Bestimmung des nächsten Schrittes.

effektiv : eindeutige Ausführbarkeit der Einzelschritte.

effizient : möglichst geringer Verbrauch an Ressourcen (z.B. Arbeitsspeicher).

Ein Algorithmus heißt

terminierend , wenn seine Ausführung für jede mögliche Eingabe nach einer endlichen Anzahl von Schritten **endet** .

determiniert , wenn er für eine bestimmte Eingabe bei allen Abläufen immer **dieselbe Ausgabe** liefert

Beispiel Personalführung

Generated by Targteam

Eingabe: Liste aller Mitarbeiter

wiederhole solange die Liste nicht leer ist

wiederhole für alle Mitarbeiter

führe willkürlich einen der folgenden Einzelschritte aus:

belobige den Mitarbeiter

tadel den Mitarbeiter

entlasse den Mitarbeiter (entferne ihn aus der Liste)

Ende Willkür

Ende Wiederhole

Ende Wiederhole

Ausgabe: Ursprüngliche Liste aller (auch ehemaliger) Mitarbeiter mit Lob- und Tadelvorgängen.

Algorithmus ist **nichtdeterministisch** , **nichtterminierend** , **nichtdeterminiert**.

Generated by Targteam



Sprache

In natürlicher oder formaler Sprache

Ausführung

Ausführung durch Menschen oder Maschine. "Effektiv", wenn Folge von Bearbeitungsschritten, die der beabsichtigte Ausführende "beherrscht".

Beispiele für Algorithmen

[Nicht maschinell ausführbares Beispiel](#)

[Maschinell ausführbares Beispiel](#)

Generated by Targteam

Bestimme die größte Zahl aus einer Menge von Zahlen

1. Lies die erste Zahl
2. Initialisiere Platzhalter/Variable z mit der gelesenen Zahl
3. Lies die nächste Zahl
4. Wenn diese Zahl größer z, dann setze z auf diese Zahl
5. Wenn noch Zahlen vorhanden, dann gehe zu Schritt 3
6. Gib den Wert von z aus

Vorausgesetzt, dass Menge so organisiert, dass Maschine Zugriff auf "erste" und "nächste" Zahl ausführen kann.

Generated by Targteam



Was wird verwendet, um Algorithmus zu formulieren?

Objekte und Anweisungen

In allgemeiner Form:

Objekte und

Anweisungen, die Operationen an Objekten realisieren.

Beispiele

Objekte sind z.B. Zahlen, Adressen, Textdokumente. Operationen sind z.B. Addition, Suche, Ausdrucken, Rechtschreibprüfung.

Operationen bringen Objekte aus Anfangszustand ggf. über Zwischenzustände in Endzustand. (EVA-Prinzip: Eingabe -> Verarbeitung -> Ausgabe)

Abstraktion

Bei Beschreibungen ist Abstraktion wichtig: Übergang von konkreten Gegebenheiten in allgemein gültige Strukturierung:

Abbildung der realen Objekte auf Daten, der Operationen auf vorgegebenen Vorrat von Anweisungen und Kontrollstrukturen (Zahlenmenge -> Liste mit Einfügen, Löschen, Sortieren, Suchen).

Abstraktion bedeutet: man konzentriert sich auf das Wesentliche; unwesentliche Anteile werden weggelassen.

Struktur eines Algorithmus

Blöcke

Operationen zu Blöcken zusammenfassen: strukturiertes / systematisches Programmieren. Block als komplexe Anweisung betrachten, Vorstufe zur Modularisierung. Je Block nur ein Eingang und ein Ausgang.



Generelle Grundstrukturen für Operationen (wenn hinreichend verfeinert):

elementarer Einzelschritt : nicht weiter zerlegbare Verarbeitung, z.B. "schalte rotes Licht ein".

Sequenz : Folge von Einzelschritten (Anweisungen); Einzelschritte können zu Blöcken zusammengefasst werden. Beispiel

schalte rotes Licht ein;

warte eine Minute;

schalte gelbes Licht ein;

Alternative : über eine Bedingung gesteuerte Verzweigung,

falls Bedingung

dann ja-Fall

sonst nein-Fall

Beispiel

falls b ungleich 0 dann berechne a/b und gib Ergebnis aus;

sonst melde Fehler

Auswahl (Selektion): Verallgemeinerung der Alternative,

Iteration : Wiederholung einer Menge von Einzelschritten.

wiederhole i = 1, ..., n

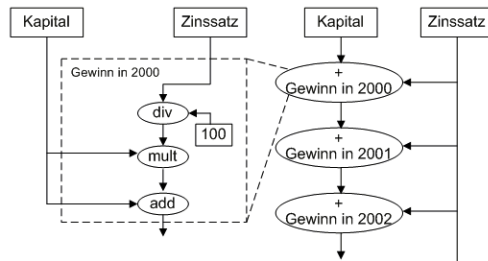
verarbeite aktuelles Konto K_i ;



Beispiel: Berechnung Zinsseszinsen



Zu Beginn des Jahres 2000 werde ein Kapital von 10000 Euro zu einem Zinssatz von 3,75 % angelegt. Wie groß ist das Kapital am Ende des Jahres 2005 ?



Generated by Targeteam

Generated by Targeteam

Berechnung Zinseszinsen - Windows Internet Explorer

C:\www\weif-ss15\whiteboard\weif_course6_2.2.1.1.html

C:\Dokumente und Einstellungen\Schlichter\Startmenü

Zu Beginn des Jahres 2000 ist das Kapital am Ende des Jahres 2006

Start

Excel - uebung3.xls

	A	B	C	D	E	F	G	H
1	Berechnung der Zinseszinsen							
2								
3	Kapital am 01.01.2000	10.000,00 €						
4	Zinssatz	3,75						
5								
6	Kapital am 01.01.	Betrag	Verzinsung jährlich am 31.12.					
7	2000	10.000,00 €						
8	2001	10.375,00 €	$B7 * (1 + B4/100)$					
9	2002	10.764,06 €	$B8 * (1 + B4/100)$					
10	2003	11.167,71 €	$B9 * (1 + B4/100)$					
11	2004	11.586,50 €	$B10 * (1 + B4/100)$					
12	2005	12.021,00 €	$B11 * (1 + B4/100)$					
13	2006	12.471,79 €						
14								
15								
16								
17								
18								
19								
20								
21								
22								
23								
24								
25								
26								
27								
28								
29								
30								
31								

Start

Berechnung Zinseszinsen - Windows Internet Explorer

C:\www\weif-ss15\whiteboard\weif_course6_2.2.1.1.html

C:\www\weif-ss15\uebungen

Zu Beginn des Jahres 2000 ist das Kapital am Ende des Jahres 2006

Start

Komponenten eines Algorithmus

Was wird verwendet, um Algorithmus zu formulieren?

Objekte und Anweisungen

In allgemeiner Form:

- Objekte und
- Anweisungen, die Operationen an Objekten realisieren.

Beispiele

Objekte sind z.B. Zahlen, Adressen, Textdokumente. Operationen sind z.B. Addition, Suche, Ausdrucken, Rechtschreibprüfung.

Operationen bringen Objekte aus Anfangszustand ggf. über Zwischenzustände in Endzustand. (EVA-Prinzip: Eingabe -> Verarbeitung -> Ausgabe)

Abstraktion

Bei Beschreibungen ist Abstraktion wichtig: Übergang von konkreten Gegebenheiten in allgemein gültige Strukturierung:

- Abbildung der realen Objekte auf Daten, der Operationen auf vorgegebenen Vorrat von Anweisungen und Kontrollstrukturen (Zahlenmenge -> Liste mit Einfügen, Löschen, Sortieren, Suchen).

Abstraktion bedeutet: man konzentriert sich auf das Wesentliche; unwesentliche Anteile werden weggelassen.

Struktur eines Algorithmus

Blöcke

Operationen zu Blöcken zusammenfassen: strukturiertes / systematisches Programmieren. Block als komplexe Anweisung betrachten, Vorstufe zur Modularisierung. Je Block nur ein Eingang und ein Ausgang.



Objekte und Anweisungen

In allgemeinsten Form:

Objekte und

Anweisungen, die Operationen an Objekten realisieren.

Beispiele

Objekte sind z.B. Zahlen, Adressen, Textdokumente. Operationen sind z.B. Addition, Suche, Ausdrucken, Rechtschreibprüfung.

Operationen bringen Objekte aus Anfangszustand ggf. über Zwischenzustände in Endzustand. (EVA-Prinzip: Eingabe -> Verarbeitung -> Ausgabe)

Abstraktion

Bei Beschreibungen ist Abstraktion wichtig: Übergang von konkreten Gegebenheiten in allgemein gültige Strukturierung:

Abbildung der realen Objekte auf Daten, der Operationen auf vorgegebenen Vorrat von Anweisungen und Kontrollstrukturen (Zahlenmenge -> Liste mit Einfügen, Löschen, Sortieren, Suchen).

Abstraktion bedeutet: man konzentriert sich auf das Wesentliche; unwesentliche Anteile werden weggelassen.

Struktur eines Algorithmus

Blöcke

Operationen zu Blöcken zusammenfassen: strukturiertes / systematisches Programmieren. Block als komplexe Anweisung betrachten, Vorstufe zur Modularisierung. Je Block nur ein Eingang und ein Ausgang. Klammerung über spezielle Schlüsselwörter oder durch Klammerzeichen, z.B. {} in C oder Java.

Generated by Targeteam

Unterschiedliche Darstellungsmethoden: informell textuell, programmiersprachlich, graphisch (z.B. Flussdiagramme oder Nassi-Shneiderman).

Flussdiagramm

Programmiersprachliche Darstellung

```
x = M;
y = N;
while (x != y)
    if (x > y) x = x - y;
    else y = y - x;
z = x;
```

"x != y" bedeutet hier die Bedingung "x ungleich y".

Generated by Targeteam



Unterschiedliche Darstellungsmethoden: informell textuell, programmiersprachlich, graphisch (z.B. Flussdiagramme oder Nassi-Shneiderman).

Flussdiagramm

Programmiersprachliche Darstellung

```
x = M;
y = N;
while (x != y)
    if (x > y) x = x - y;
    else y = y - x;
z = x;
```

"x != y" bedeutet hier die Bedingung "x ungleich y".

Generated by Targeteam

Vor Programm (Lösungsvorschrift) braucht man: Exakte Beschreibung des Problems.

Spezifikation als Ausgangspunkt für ein Programm

Vollständige, detaillierte, unzweideutige Problembeschreibung.

vollständig: alle relevanten Informationen sind berücksichtigt.

detailliert: alle Hilfsmittel und Grundaktionen sind aufgelistet, die zur Lösung zugelassen sind.

unzweideutig: klare Kriterien sind festgelegt, die bestimmen, wann eine Lösung akzeptabel ist.

Begriff des Algorithmus

Komponenten eines Algorithmus

Darstellung von Algorithmen

Algorithmus und Programm

Generated by Targeteam



Programme sind spezielle Darstellungsformen für Algorithmen.

Zusammenhang

Programm: Formulierung eines Algorithmus in Sprache, die Computer versteht.

Aus Sicht der "Maschine": endliche Folge von ausführbaren Anweisungen.

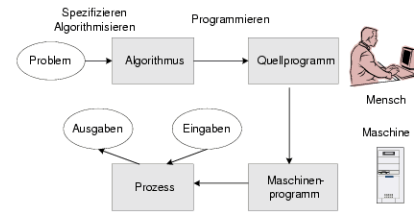
Damit Automatisierung des Algorithmus.

Programmiersprachen sind ein Hilfsmittel zur Abbildung der Daten- und Kontrollstrukturen des Algorithmus in die Sprachkonstrukte des betreffenden Computers (Maschinensprache, Arbeitsspeicher).

Ausführbare Programme: Abbildung der Daten- und Kontrollstrukturen in Interndarstellung des Computers (Objektcode). Verwendet Maschinensprache (abhängig vom jeweiligen Hersteller des Prozessors).

[Zeitlicher Zusammenhang zwischen Algorithmus und Programm](#)

Generated by Targeteam



Generated by Targeteam



Programme sind spezielle Darstellungsformen für Algorithmen.

Zusammenhang

Programm: Formulierung eines Algorithmus in Sprache, die Computer versteht.

Aus Sicht der "Maschine": endliche Folge von ausführbaren Anweisungen.

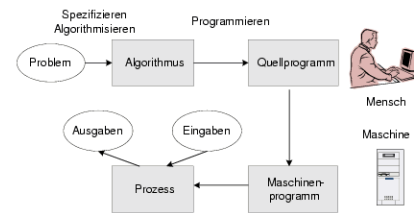
Damit Automatisierung des Algorithmus.

Programmiersprachen sind ein Hilfsmittel zur Abbildung der Daten- und Kontrollstrukturen des Algorithmus in die Sprachkonstrukte des betreffenden Computers (Maschinensprache, Arbeitsspeicher).

Ausführbare Programme: Abbildung der Daten- und Kontrollstrukturen in Interndarstellung des Computers (Objektcode). Verwendet Maschinensprache (abhängig vom jeweiligen Hersteller des Prozessors).

[Zeitlicher Zusammenhang zwischen Algorithmus und Programm](#)

Generated by Targeteam



Generated by Targeteam





Grundkonstrukte in allen Programmiersprachen.

Informationen auf Rechner: Daten. Umsetzung Daten in Information: Interpretation.

Allgemeine Eigenschaften von Daten

1. Basisdaten: Zeichen, Wahrheitswerte (true, false), Zahlen (natürliche Zahlen, ganze Zahlen, Gleitkommazahlen).
2. Daten anordenbar, in Beziehung setzbar (z.B. Bäume, Listen)
3. Algorithmen hängen von gewählter Datenstruktur ab (Übersichtlichkeit, Effizienz).

Elementare Datentypen

"Datenstruktur" (im Gegensatz zu struct-Datentypen): Datentyp mit Menge von zugehörigen Operationen.

Ausdrücke

Generated by Targeteam



Datentyp: Zusammenfassung von Werten gleicher Art (z.B. ganze Zahlen, Gleitkommazahlen, Zeichen).

Basis-Datentypen

Die Verwendung der Schlüsselwörter ("keywords") sind abhängig von der jeweiligen Programmiersprache.

`int` : Ausschnitt der ganzen Zahlen (im Rechner darstellbare ganze Zahlen)

`float` : Menge der Gleitkommazahlen

`double` : Menge der Gleitkommazahlen mit doppelter Genauigkeit

`char` : Menge der Zeichen

`boolean` : Werte `true` und `false`

Datentyp Verbund (struct)

Generated by Targeteam



Zusammenfassung von zusammengehörigen Daten unterschiedlicher Typen.

Beispiel

Vorname	Name	Adresse	Alter
Fritz	Müller	Hauptstr. 7	38
Hans	Albers	Bahnhofstr. 4	63

Programmiersprachliche Darstellung

Beispiel gemäß der Programmiersprache C

```
struct person {
    char vorname[30];
    char name[30];
    char adresse[100];
    int alter;
}
```

Beispiel der Nutzung:

```
struct person p; /* Deklaration der Person p */
p.vorname = "Fritz"; /* Vorname von Person p */
p.name = "Müller"; /* Name von Person p */
```

in Java: Zusammenfassung von zusammengehörigen Datenwerten zu Objekten, d.h. jede Tabellenzeile entspricht einem Objekt.

Generated by Targeteam



Dienen zur Berechnung von Datenwerten. Aufgebaut aus Konstanten, Variablen und Operationszeichen.

Variable

Aus Mathematik übernommen und abgewandelt.

Benannte Speicherzelle für Datenwerte; lesen, schreiben (zuweisen).

Variablennamen (Bezeichner) i.a. deklarieren, dabei Datentyp zuordnen.

Beispiele für Ausdrücke

$2 + 15 * (10 - 6)$

$a + b$

wobei a und b Variable sind.

Generated by Targeteam



"Kunst des Programmierens". Grundlagen zu Datenstrukturen, Programmkonstrukte, Strukturierung von Programmen, objekt-orientierte Programmierung.

- Fragestellungen des Abschnitts:
 - Was ist ein Algorithmus?
 - Welche elementaren Datenstrukturen gibt es?
 - Was sind die grundlegenden Konstrukte einer Programmiersprache?
 - Was ist unter Objekt-orientierter Programmierung zu verstehen?
 - Was versteht man unter Modularisierung und Rekursion?

[Einführung](#)

[Algorithmus](#)

[Datentypen und Ausdrücke](#)

[Programmkonstrukte](#)

[Objektorientierte Programmierung](#)

[Modularisierung von Programmen](#)

[Rekursion](#)

Generated by Targeteam

Anweisungen eines Programms weisen Werte an Variable zu oder steuern Kontrollfluss. Kontrollfluss bestimmt Folge der ausgeführten Anweisungen.

[Zuweisungen](#)

[Alternativanweisungen](#)

[Schleifen](#)

Generated by Targeteam



Anweisungen eines Programms weisen Werte an Variable zu oder steuern Kontrollfluss. Kontrollfluss bestimmt Folge der ausgeführten Anweisungen.

[Zuweisungen](#)

[Alternativanweisungen](#)

[Schleifen](#)

Generated by Targeteam