Konzepte des prozeduralen Programmierens

Stand Dezember 2022

- Prof. Dr.-Ing. M.B. Endejan / Christian Frank

3a Pseudo-Code

Übersicht

- Kontrollstrukturen
- Pseudocode
  - Schlüsselwörter
  - Konventionen
- Literatur & Links

Kontrollstrukturen

Kontrollstrukturen
steuern den Ablauf eines Algorithmus und geben an, ob bzw. wie oft einzelne Anweisungen ausgeführt werden sollen.

Es werden vier Kontrollstrukturen unterschieden:

- Sequenz
eine Aneinanderreihung von Anweisungen, die nacheinander (von oben nach unten bzw. von rechts nach links) ausgeführt werden

- Auswahl (Verzweigung)
Ausführung in Abhängigkeit bestimmter Bedingungen; es werden drei Auswahlkonzepte unterschieden: 1) einseitige Auswahl, 2) zweiseitige Auswahl, 3) Mehrfachauswahl

Quelle: Balzert (1996): Lehrbuch der Software-Technik: Software-Entwicklung. Spektrum Akademischer Verlag, Heidelberg.

Kontrollstrukturen

- Wiederholung (Schleife)
  Eine oder mehrere Anweisungen werden in Abhängigkeit von einer Bedingung oder für eine gegebene Anzahl von Wieder-holungen durchlaufen.
  Es werden drei Wiederholungskonstrukte unterschieden:
  Wiederholung mit
  - Abfrage vor jedem Wiederholungsdurchlauf
  - Abfrage nach dem Wiederholungsdurchlauf
  - fester Wiederholungszahl (Zählschleife)
- Aufruf anderer Algorithmen
  Ein Aufruf wird verwendet, um in einem Algorithmus einen anderen Algorithmus anzugwenden. **Kontrollstrukturen**
  steuern den Ablauf eines Algorithmus und geben an, ob bzw. wie oft einzelne Anweisungen ausgeführt werden sollen.

Quelle: Balzert (1996): Lehrbuch der Software-Technik: Software-Entwicklung. Spektrum Akademischer Verlag, Heidelberg.

Kontrollstrukturen

- Die vier Kontrollstrukturen Sequenz, Auswahl, Wiederholung und Aufruf haben jeweils genau einen Eingang und einen Ausgang.
- Der Kontrollfluss verlässt den zwischen Eingang und Ausgang definierten Kontrollbereich nicht (Lokalitätsprinzip).
- Der Kontrollfluss verläuft linear (sequenziell vom Eingang zum Ausgang) durch den Algorithmus.
- Werden in einem Algorithmus bzw. Programm nur lineare Kontrollstrukturen verwendet, spricht man von **strukturiertem Programmieren i.e.S.**
- Die Einhaltung der Lokalität und der Linearität erleichtert auch den Korrektheitsbeweis für einen Algorithmus.

- *Achtung*: Der Sprung (goto) ist keine lineare Kontrollstruktur.

Darstellung

- Für die Darstellung von Kontrollstrukturen können unterschiedliche grafische oder textuelle Beschreibungen genutzt werden. Hierzu zählen Struktogramme (Nassi-Shneiderman-Diagramm), Programm-Ablaufpläne und Pseudocode.
- Pseudocode ist eine textuelle, semiformale Darstellung von Kontrollstrukturen in Anlehnung an problemorientierte Programmiersprachen (wie C, C++, Pascal, Java).
  - Pseudocode kann normalsprachliche Beschreibungen enthalten
  - Pseudocode kann nicht durch Rechner ausgeführt werden
  - Pseudocode ist nicht standardisiert
  - Die folgenden Erklärungen beziehen sich auf die Schlüsselwörter und Pseudocode-Konventionen, wie sie von *Cormen et al. (2010)* verwendet werden.

Pseudocode: Schlüsselwörter

- Auswahl
  - if
  - else
  - elseif
- Wiederholung
  - for
  - to
  - by
  - downto
  - while
  - repeat
  - until
- Sprünge
  - return
  - error
  - goto
- Multithreading
  - spawn
  - sync
  - parallel for
- Kommentar
  - //

Quelle: Cormen et al. (2010): Algorithmen – Eine Einführung. 3. Auflage. Oldenbourg Verlag, München

Pseudocode: Konventionen

- Einrückung kennzeichnet Blockstruktur (übersichtlicher als Verwendung von *begin* und *end*)
- Bei *if* wird das Schlüsselwort *then* weggelassen
- Schleifenkonstrukte while, *for*, *repeat-until* und Auswahl *if-else* in Anlehnung an C/C++/Java/Python/Pascal.
- Bei *for-Schleifen* behält die Laufvariable ihren (Abbruch-) Wert auch nach Verlassen der Schleife.
- Variablen gelten als lokal innerhalb einer Prozedur.
- A[i] entspricht dem Zugriff auf das i-te Element des Feldes A.
- A[1..j] entspricht dem Teilfeld A[1], A[2], …, A[j] von A.
- Für die Darstellung von Kontrollstrukturen können unterschiedliche grafische oder textuelle Beschreibungen genutzt werden. Hierzu zählen Struktogramme (Nassi-Shneiderman-Diagramm), Programm-Ablaufpläne und Pseudocode.
- Pseudocode ist eine textuelle, semiformale Darstellung von Kontrollstrukturen in Anlehnung an problemorientierte Programmiersprachen (wie C, C++, Pascal, Java).
  - Pseudocode kann normalsprachliche Beschreibungen enthalten
  - Pseudocode kann nicht durch Rechner ausgeführt werden
  - Pseudocode ist nicht standardisiert
  - Die folgenden Erklärungen beziehen sich auf die Schlüsselwörter und Pseudocode-Konventionen, wie sie von *Cormen et al. (2010)* verwendet werden.

Quelle: Cormen et al. (2010): Algorithmen – Eine Einführung. 3. Auflage. Oldenbourg Verlag, München

Pseudocode: Konventionen

- Die Angabe *Objekt.Attribut* entspricht dem Zugriff auf ein Objektattribut (Bsp. A.länge)
- Parameter werden einer Prozedur als Werte übergeben (call by value), Objekte werden über Zeiger auf das Objekt übergeben (call by reference)
- Die *return*-Anweisung gibt die Kontrolle direkt zurück an die aufrufende Prozedur. Eine return-Anweisung kann mehrere Werte auf einmal zurückgeben.
- Die boolschen Operatoren „und“ und „oder“ sind träge Operatoren. Bei „A und B“ wird B nur ausgewertet, wenn A wahr ist, bei „A oder B“ wird B nur ausgewertet, wenn A falsch ist.
- *error* gibt an, dass ein Fehler aufgetreten ist, weil Bedingungen einer Prozedur verletzt sind.

Quelle: Cormen et al. (2010): Algorithmen – Eine Einführung. 3. Auflage. Oldenbourg Verlag, München

Literatur & Links

Literatur

•Cormen et al. (2010)
Cormen, Thomas H.; Leiserson, Charles E.; Rivest, R.; Stein, C. (2010): Algorithmen – Eine Einführung. 3. Auflage. Oldenbourg Verlag, München

•Balzert (1996)
Lehrbuch der Software-Technik: Software-Entwicklung. Spektrum Akademischer Verlag, Heidelberg.

Links

•http://www.cs.dartmouth.edu/~thc/clrscode/
Cormen (2009): clrscode3e Paket für LATEX zum Setzen von Pseudocode im Stile von *Cormen et al. (2010)*

•http://www.tex.ac.uk/cgi-bin/texfaq2html?label=algorithms
Nützliche Informationen zum Thema Latex und Pseudocode.

Literatur & Links

Literatur

-Cormen et al. (2010)
Cormen, Thomas H.; Leiserson, Charles E.; Rivest, R.; Stein, C. (2010): Algorithmen – Eine Einführung. 3. Auflage. Oldenbourg Verlag, München

-Balzert (1996)
Lehrbuch der Software-Technik: Software-Entwicklung. Spektrum Akademischer Verlag, Heidelberg.

Links

-http://www.cs.dartmouth.edu/~thc/clrscode/
Cormen (2009): clrscode3e Paket für LATEX zum Setzen von Pseudocode im Stile von *Cormen et al. (2010)*

-http://www.tex.ac.uk/cgi-bin/texfaq2html?label=algorithms
Nützliche Informationen zum Thema Latex und Pseudocode.