

Algorithmen und Datenstrukturen in C/C++

Prof. Dr. rer. nat. Peter Gerwinski

3. Mai 2018

Algorithmen und Datenstrukturen in C/C++

<https://gitlab.cvh-server.de/pgerwinski/ad.git>

- 1 Einführung
- 2 Einführung in C++
- 3 Datenorganisation
 - 3.1 Standard-Container-Templates
 - 3.2 Iteratoren
 - 3.3 Hash-Tabellen
 - 3.4 Balancierte Bäume
 - 3.5 Intelligente Zeiger
- 4 Datenkodierung
- 5 Hardwarenahe Algorithmen
- 6 Optimierung
- 7 Numerik



Änderungen
vorbehalten

3 Datenorganisation

3.3 Hash-Tabellen

Idee: Ein String ist auch nur eine große Zahl.
Wenn man die irgendwie zurechtstutzt,
kann man sie als Array-Index verwenden.

Übungsaufgabe:

Schreiben Sie ein Programm, das zu Strings (z. B. Namen)
Datensätze speichern und diese in $\mathcal{O}(1)$ wieder abrufen kann.

- (a) Verwenden Sie das Hash-Tabellen-Template
aus der C++-Standardbibliothek.
- (b) Programmieren Sie die Hash-Tabelle selbst.

Teamarbeit, Internet-Recherchen
und die Verwendung anderer Container-Standard-Templates
sind ausdrücklich erlaubt.

3 Datenorganisation

3.4 Balancierte Bäume

- **Problem:** Entartung von Bäumen
Lösung: variable Anzahl von Verzweigungen
- **Problem:** Datenstruktur zu groß für Hauptspeicher
Lösung: “intelligentes” Auslagern

B-Baum:

- immer mindestens t , höchstens $2t$ Verzweigungen
- Einfügen: wenn Knoten voll, teilen
- Löschen:
 - unterste Ebene: ggf. verschieben, ggf. verschmelzen
 - in inneren Knoten: Wert von unten ersetzen, ggf. verschmelzen
- einfachster Fall: 2-3-4-Baum

Übungsaufgabe: Implementieren Sie einen 2-3-4-Baum mit Methoden zum Suchen, Einfügen und Löschen

3 Datenorganisation

3.5 Intelligente Zeiger

Warum?

- bereits freigegebene Zeiger werden u. U. weiterhin verwendet
- Speicherlecks
- uninitialisierte Zeiger

- `shared_ptr`
- `weak_ptr`
- `unique_ptr`
- `move()`

3 Datenorganisation

3.5 Intelligente Zeiger

Wie?

- R-Wert-Referenztypen: `&&`
- `move()`-Funktion

Literatur:

- http://thbecker.net/articles/rvalue_references/section_01.html
- <http://www.artima.com/cppsource/rvalue.html>

Algorithmen und Datenstrukturen in C/C++

<https://gitlab.cvh-server.de/pgerwinski/ad.git>

- 1 Einführung
- 2 Einführung in C++
- 3 Datenorganisation
 - 3.1 Standard-Container-Templates
 - 3.2 Iteratoren
 - 3.3 Hash-Tabellen
 - 3.4 Balancierte Bäume
 - 3.5 Intelligente Zeiger
- 4 Datenkodierung
- 5 Hardwarenahe Algorithmen
- 6 Optimierung
- 7 Numerik



Änderungen
vorbehalten