

Algorithmen und Datenstrukturen in C/C++

Prof. Dr. rer. nat. Peter Gerwinski

4. Mai 2023

Algorithmen und Datenstrukturen in C/C++

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

1 Einführung

...

1.8 Templates

1.9 Container-Templates

1.10 Iteratoren

1.11 Exceptions

1.12 Typ-Konversionen

1.13 Intelligente Zeiger

2 Datenorganisation

3 Datenkodierung

3. $(x^2 - 1)$ Mathematische Grundlagen

4 Hardwarenahe Algorithmen

5 Optimierung

6 Numerik



Änderungen
vorbehalten

1 Einführung in C++

1.13 Intelligente Zeiger

Warum?

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

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

1 Einführung in C++

1.13 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>

1.11 Exceptions

```
try
{
    ...
    throw <value>;
    ...
}
catch (<type> <variable>)
{
    ...
}
catch ...
```

- Nach den `catch()`-Statements wird, soweit nicht anders programmiert, das Programm fortgesetzt.
- `throw;` (ohne Wert):
an übergeordneten Exception-Handler weiterreichen
- C-Äquivalent:
`setjmp()`, `longjmp()`
- speziell für `<type>`:
Nachfahren von `class exception`
- veraltet:
dynamic exception specifications

1.12 Typ-Konversionen

- In C:
`char *hello = "Hello, world!";`
`uint64_t address = (uint64_t) hello;`
`printf ("%s" PRlu64 "\n", address);`
- alternative Syntax in C++:
`char *hello = "Hello, world!";`
`uint64_t address = uint64_t (hello);`
`cout << address << endl;`
- zusätzlich in C++:
implizite und explizite Typumwandlung zwischen Zeigern auf Klassen
`dynamic_cast<>()`
`static_cast<>()`
`reinterpret_cast<>()`
`const_cast<>()`

1.12 Typ-Konversionen

<http://www.cplusplus.com/doc/tutorial/typecasting/>

- Zuweisung: Zeiger auf abgeleitete Klasse an Zeiger auf Basisklasse
→ implizite Typumwandlung möglich
- Zuweisung: Zeiger auf Basisklasse an Zeiger auf abgeleitete Klasse
→ nur explizite Typumwandlung möglich:
`dynamic_cast<>()`, `static_cast<>()`
- implizite Typumwandlungen in der Klasse definieren:
 - Initialisierung durch Konstruktor
 - Zuweisungs-Operator
 - Typumwandlungsoperator
- implizite Typumwandlungen ausschalten:
Schlüsselwort `explicit`

1.12 Typ-Konversionen

<http://www.cplusplus.com/doc/tutorial/typecasting/>

- `dynamic_cast<>()`
Zuweisung: Zeiger auf Basisklasse an Zeiger auf abgeleitete Klasse
explizite Typumwandlung mit Prüfung, ggf. Exception
- `static_cast<>()`
Zuweisung: Zeiger auf Basisklasse an Zeiger auf abgeleitete Klasse
explizite Typumwandlung ohne Prüfung
- `reinterpret_cast<>()`
Typumwandlung ohne Prüfung zwischen Zeigern untereinander
und zwischen Zeigern und Integer-Typen
- `const_cast<>()`
„**const**“ ein- bzw. ausschalten

3 Datenkodierung

3 Datenkodierung

3. $(x^2 - 1)$ Der Herr der Ringe: Manchmal ist $1 + 1 = 0$.

3 Datenkodierung

3. $(x^2 - 1)$ Der Herr der Ringe: Manchmal ist $1 + 1 = 0$.

3. $(x^2 - 1).x$ Motivation

Man kann auch mit sehr merkwürdigen Objekten wie mit „ganz normalen“ Zahlen rechnen.

3 Datenkodierung

3. $(x^2 - 1)$ Der Herr der Ringe: Manchmal ist $1 + 1 = 0$.

3. $(x^2 - 1).x$ Motivation

Man kann auch mit sehr merkwürdigen Objekten wie mit „ganz normalen“ Zahlen rechnen.

Anwendungen:

- Funktionsweise von Computern (\longrightarrow Rechnertechnik)
- Fehlererkennung
- Fehlerkorrektur
- Verschlüsselung
- Digitale Signaturen

3 Datenkodierung

3. $(x^2 - 1)$ Der Herr der Ringe: Manchmal ist $1 + 1 = 0$.

3. $(x^2 - 1) \cdot (x + 1)$ Gruppen

Definition: Sei G eine Menge, $*$ eine Verknüpfung auf G . Wenn

- $\forall a, b, c \in G: (a * b) * c = a * (b * c)$ (Assoziativgesetz),
- $\exists e \in G: \forall a \in G: a * e = e * a = a$ (neutrales Element),
- $\forall a \in G: \exists a^{-1} \in G: a * a^{-1} = a^{-1} * a = e$ (inverses Element),

dann heißt $(G, *)$ eine *Gruppe*.

3 Datenkodierung

3. $(x^2 - 1)$ Der Herr der Ringe: Manchmal ist $1 + 1 = 0$.

3. $(x^2 - 1) \cdot (x + 1)$ Gruppen

Definition: Sei G eine Menge, $*$ eine Verknüpfung auf G . Wenn

- $\forall a, b, c \in G: (a * b) * c = a * (b * c)$ (Assoziativgesetz),
- $\exists e \in G: \forall a \in G: a * e = e * a = a$ (neutrales Element),
- $\forall a \in G: \exists a^{-1} \in G: a * a^{-1} = a^{-1} * a = e$ (inverses Element),

dann heißt $(G, *)$ eine *Gruppe*.

Definition: Sei $(G, *)$ eine Gruppe. Wenn zusätzlich

- $\forall a, b \in G: a * b = b * a$ (Kommutativgesetz),

dann heißt $(G, *)$ eine *kommutative Gruppe*.

3. $(x^2 - 1)$ Der Herr der Ringe: Manchmal ist $1 + 1 = 0$.

3. $(x^2 - 1) \cdot (x + 2)$ Ringe

Definition: Sei R eine Menge; seien $+$ und \cdot Verknüpfungen auf R . Wenn

- $(R, +)$ eine kommutative Gruppe ist,
- $\forall a, b, c \in R: (a \cdot b) \cdot c = a \cdot (b \cdot c)$ (Assoziativgesetz),
- $\forall a, b, c \in R: (a + b) \cdot c = a \cdot c + b \cdot c$ und $a \cdot (b + c) = a \cdot b + a \cdot c$ (Distributivgesetze),

dann heit $(R, +, \cdot)$ ein *Ring*.

3. $(x^2 - 1)$ Der Herr der Ringe: Manchmal ist $1 + 1 = 0$.

3. $(x^2 - 1) \cdot (x + 2)$ Ringe

Definition: Sei R eine Menge; seien $+$ und \cdot Verknüpfungen auf R . Wenn

- $(R, +)$ eine kommutative Gruppe ist,
- $\forall a, b, c \in R: (a \cdot b) \cdot c = a \cdot (b \cdot c)$ (Assoziativgesetz),
- $\forall a, b, c \in R: (a + b) \cdot c = a \cdot c + b \cdot c$ und $a \cdot (b + c) = a \cdot b + a \cdot c$ (Distributivgesetze),

dann heißt $(R, +, \cdot)$ ein *Ring*.

Definition: Sei $(R, +, \cdot)$ ein Ring. Wenn zusätzlich

- $\forall a, b \in R: a \cdot b = b \cdot a$ (Kommutativgesetz),

dann heißt $(R, +, \cdot)$ ein *kommutativer Ring*.

3. $(x^2 - 1)$ Der Herr der Ringe: Manchmal ist $1 + 1 = 0$.

3. $(x^2 - 1) \cdot (x + 2)$ Ringe

Definition: Sei R eine Menge; seien $+$ und \cdot Verknüpfungen auf R . Wenn

- $(R, +)$ eine kommutative Gruppe ist,
- $\forall a, b, c \in R: (a \cdot b) \cdot c = a \cdot (b \cdot c)$ (Assoziativgesetz),
- $\forall a, b, c \in R: (a + b) \cdot c = a \cdot c + b \cdot c$ und $a \cdot (b + c) = a \cdot b + a \cdot c$ (Distributivgesetze),

dann heißt $(R, +, \cdot)$ ein *Ring*.

Definition: Sei $(R, +, \cdot)$ ein Ring. Wenn zusätzlich

- $\forall a, b \in R: a \cdot b = b \cdot a$ (Kommutativgesetz),

dann heißt $(R, +, \cdot)$ ein *kommutativer Ring*.

Definition: Sei $(R, +, \cdot)$ ein (kommutativer) Ring. Wenn zusätzlich

- ein $e \in R$ existiert, so daß für alle $a \in R$ gilt: $a \cdot e = e \cdot a = a$ (neutrales Element),

dann heißt $(R, +, \cdot)$ ein *(kommutativer) Ring mit 1*.

3. $(x^2 - 1)$ Der Herr der Ringe: Manchmal ist $1 + 1 = 0$.

3. $(x^2 - 1) \cdot (x + 3)$ Körper

Definition: Sei K eine Menge; seien $+$ und \cdot Verknüpfungen auf K . Wenn

- $(K, +, \cdot)$ ein Ring mit 1 ist und
- $(K \setminus \{0\}, \cdot)$ eine kommutative Gruppe ist,

dann heißt $(K, +, \cdot)$ ein *Körper*.