

§2 FEHLERBEHANDLUNG MIT EXCEPTIONS

Leitideen: Fehler in Konstruktoren und Operatorfunktionen können nicht mittels Rückgabewerten behandelt werden. Hier wird ein anderer Mechanismus benötigt, der Fehler wird signalisiert („Werfen“ einer Ausnahme).

Die Ausnahme (exception) kann in einem übergeordneten Block abgefangen werden und das Programm dort fortgesetzt werden.

Nichtabfangen einer Ausnahme führt zum Programmabbruch.

- Exceptions I,II
- Exceptions in der Standardbibliothek

Exceptions I

Problem

- ▶ `cin >> n;`
`vector<double> a(n); // n zu gross`
- ▶ Abfangen eines evtl. Programmabsturzes?

Lösungsansatz

- ▶ Erzeugung eines Signals („Ausnahme“ exception) in der Klasse `vector` im Falle eines Allokationsproblems
- ▶ Mechanismus, mit dem aufrufende Funktionen (z.B. `main`), auf das Signal reagieren können

Erzeugung von Exceptions

- ▶ `throw`-Ausdruck erzeugt („wirft“) Exception
- ▶ Exception ist Objekt eines *beliebigen* Datentyps
- ▶ Vor allem der Datentyp der Exception ist für ihre spätere Behandlung maßgebend

Exceptions II

Abfangen von Exceptions

- ▶ Beim Auftreten einer Exception Suche nach einem umschließenden `try`-Block
Ausschlaggebend: dynamische Blockschachtelung (von innen nach außen)
- ▶ *Falls gefunden:*
Abarbeitung der zugehörigen `catch`-Anweisungen:
Bei passendem Datentyp der Exception Ausführung des entsprechenden Blocks
- ▶ *Falls nicht gefunden:*
Aufruf von `terminate` (Voreinst.: Programmabbruch)
- ▶ Zu einem `try`-Block mehrere `catch`-Anw. möglich.
- ▶ Falls Datentyp der Exception nicht einschlägig, Suche nach einem umschließenden `try`-Block usw.
- ▶ `catch(...)` behandelt *jede* Exception
- ▶ *Kein* Rücksprung in den `try`-Block, statt dessen im Erfolgsfall Fortsetzung mit der auf `try/catch` folgenden Anweisung

Exceptions in der Standardbibliothek

Exceptionklassen (Auszug)

- ▶ Header: `stdexcept (exception, new)`
- ▶ Am wichtigsten im Rahmen der Vorlesung:
 - `out_of_range` Bereichsüberwachung von `vector`:
 - `a.at(i)` - ggf. Exception
 - `a[i]` - keine Exception
 - `bad_alloc` Fehler bei `new`
- ▶ Übergabe von Fehlermeldungen beim „Werfen“ der Exception als String.
- ▶ Erzeugung aussagekräftiger Meldungen, z.B mit Aufrufargumenten, durch Verwendung von Outputstringstreams.
- ▶ Komponentenfunktion `what` liefert Fehlermeldung
- ▶ Exceptions in der C++-Ein/Ausgabe aktivierbar, z.B.
 - `cin.exceptions(ios::failbit|ios::badbit)`
 - Exceptiontyp: `ios::failure`
 - Vorteil: Geringerer Aufwand zur Fehlererkennung
 - Nachteil: Keine direkte Fehlerbehandlung möglich