

# Einführung in die Programmierertechnik

## Ausnahmebehandlung

# Laufzeitfehler

- Fehler sind Ereignisse, mit denen der Programmierer nicht gerechnet hat
  - Verletzung der Spezifikation, Verletzung von Annahmen an die Umgebung
- Laufzeitfehler: Implementierung der Programmiersprache oder vordefinierte Funktionen/Prozeduren können nicht richtig abgeschlossen werden
  - Division durch 0
  - Versuch, eine nicht vorhandene Datei zu öffnen

# Behandlung von Laufzeitfehlern

- verschiedene Modelle
  - Fehlercodes
  - Ausnahmen
- Fehlercodes: Funktionen, die kein reguläres Ergebnis liefern können, liefern zusätzlichen Fehlerwert
  - z.B. Dateiöffnen in C/POSIX: `open(dateiname, ...)` liefert natürliche Zahl ( $\geq 0$ ); Fehler wird als Ergebnis -1 angezeigt
  - Fehlerbehandlung: Test, ob Ergebnis -1 ist

```
fileno = open("foo.txt", <weitere Parameter>);  
if (fileno == -1)  
    datei_nicht_vorhanden();  
else  
    verarbeite_datei(fileno);
```
- Problem: Fehlerbehandlung wird u.U. vergessen; Programm rechnet mit falschen Daten weiter
  - Fehlerbehandlung mitten im Code verschlechtert Lesbarkeit

# Ausnahmebehandlung

- Modell der Ausnahmebehandlung: Fehler führt zu Abbruch der Aktion und Auslösen einer *Ausnahme*
  - engl. Ausnahme = exception
  - Auslösen der Ausnahme = to throw an exception
  - Behandlung der Ausnahme = exception handling/handler
  - Abfangen der Ausnahme = to catch an exception
- Nach Abbruch der eigentlichen Aktion wird mit einer *Ausnahmebehandlung* fortgesetzt
- Mehrere Modelle für Verhalten am Ende der Ausnahmebehandlung
  - Wiederaufnahme der Aktion nach beendeter Ausnahme (Dylan)
  - Stack-Unwinding: Fortsetzen im Code nach Ausnahmebehandlung (C++, Java, Python)

# Ausnahmebehandlung (2)

- Suche nach „passender“ Ausnahmebehandlung entsprechend rekursiven Funktionsrufen
  - Behandlung zuerst in der inneren Funktion
  - Falls keine Behandlungsroutine gefunden, wird Funktion abgebrochen und im Rufer nach Behandlung gesucht
- „Passende“ Behandlung: Typ der Behandlung entspricht aufgetretener Ausnahme
  - Integration objekt-orientierter Konzepte: Klassenhierarchie wird berücksichtigt (dazu später mehr)

# Ausnahmebehandlung in Python

- Abfangen einer Ausnahme: try-except-Block

try:

    aktion1()

    aktion2()

except Fehler:

    fehlerbehandlung()

- Auslösen einer Ausnahme: raise-Anweisung
  - vordefinierte Ausnahmen und nutzerdefinierte Ausnahmen
  - raise Exception("es gab ein Problem")