

Fragestunde SE1 WS16/17

Mathias Weber

Technische Universität Kaiserslautern

06.03.2017

Inhalt I

1. Einführung

- ▶ Was ist ein Softwaresystem?
- ▶ Entwicklung von Softwaresystemen

2. Formale Sprachen

- ▶ Zeichen, Zeichenreihe, Alphabet, formale Sprache
- ▶ Syntaxdiagramme
- ▶ kontextfreie Grammatiken
- ▶ Ableitungen, Syntaxbaum, Mehrdeutigkeit

3. Java: Basisdatentypen

- ▶ Ausführungsvarianten (Übersetzung, Interpretation)
- ▶ elementare Datentypen
- ▶ Variablen, Zuweisung
- ▶ Parameterübergabe
- ▶ Typumwandlung (Cast)

4. Algorithmen

- ▶ Speicherzustand, Steuerzustand, Aktion
- ▶ Anweisungen in Java (z.B. Variablendeklaration/-zuweisung)

Inhalt II

- ▶ Anweisungsblöcke
- ▶ Kontrollfluss: Verzweigung und Schleifen, Terminierung
- ▶ Sprung- und Auswahlanweisungen
- ▶ Unterschied: Anweisung/Ausdruck

5. Arrays

- ▶ Deklaration, lesen, schreiben
- ▶ for-Anweisung
- ▶ Referenzvariablen
- ▶ Mehrdimensionale Arrays
- ▶ Eingabe und Ausgabe

6. Prozeduren

- ▶ formale Parameter, aktuelle Parameter
- ▶ Deklaration in Java, Rückgabeeweisung
- ▶ Prozeduraufruf
- ▶ Überladen von Methoden
- ▶ Prozedurbibliotheken
- ▶ Gültigkeitsbereich von Variablen, Lebensdauer

7. Spezifikation und Testen

Inhalt III

- ▶ Prozedurspezifikation, Vorbedingung, Nachbedingung
- ▶ Unit-Tests für Java, JUnit

8. Einführung in die Objektorientierung

- ▶ Charakterisierung von Objekten; Zustand, Lebensdauer, Ort, Verhalten, Identität
- ▶ Nachrichten und Methoden
- ▶ Referenztypen in Java, Objekterzeugung, **null**-Referenz, Methodenaufruf

9. Klassen in Java

- ▶ Klassendeklarationen, Attribute, Methoden
- ▶ Konstruktoren
- ▶ Einfachverkettete Listen

10. Kapselung und Datenstrukturen

- ▶ Information Hiding, Getter, Setter
- ▶ Listen: Einfachverkettet, Doppeltverkettet, Array-Listen
- ▶ Iteratoren

11. Bäume und Rekursion

Inhalt IV

- ▶ Datenstruktur Baum, Binärbaum, Markierter Baum, Sortiertheit
- ▶ Rekursion; linear, kaskadenartig, repetitiv

12. Objektorientierte Modellierung, Schnittstellenbildung und Klassifizierung

- ▶ Objektorientierte Modellierung
- ▶ Interfaces
- ▶ Klassifizierung von Objekten, Sub-/Supertypen
- ▶ Dynamische Methodenauswahl

13. Collections und Generics

- ▶ Abstrakte Datentypen; Liste, Stapel, Warteschlange
- ▶ Parametrisierte Datentypen
- ▶ Das Java Collections Framework; Listen, Sets, Maps

14. Maps

- ▶ Abstrakter Datentyp: Maps
- ▶ basierend auf Listen, Suchbäumen
- ▶ Hashing
- ▶ Vergleich von Objekten, equals(), compare()

Inhalt V

15. Vererbung

- ▶ Substituierbarkeit
- ▶ Abstraktion und Vererbung
- ▶ überschreiben von Methoden
- ▶ abstrakte Klassen
- ▶ dynamische Methodenauswahl bei Vererbung

16. Typhierarchien, Exceptions und mehr Klassen

- ▶ Quadrat-Recheck Problem
- ▶ Ausnahmebehandlung mit Exceptions
- ▶ Klassenattribute und -methoden
- ▶ geschachtelte Klassen

17. Streams

- ▶ Zeichenströme, Reader, Writer
- ▶ Umgang mit Dateien
- ▶ Ein- und Ausgabe von Objekten, Serialisierung, JSON

18. Pakete

19. Einführung in C

- ▶ Basisdatentypen

Inhalt VI

- ▶ undefiniertes Verhalten

20. Funktionen, Zeiger und Arrays in C

- ▶ Funktionen, Deklaration, Funktionsprototyp
- ▶ Zeiger, Deklaration, Zuweisung, Dereferenzierung
- ▶ dynamische Speicherverwaltung
- ▶ Arrays

21. Strings und Strukturen

22. Sortieren

- ▶ durch Einfügen, durch Auswahl, merge sort, quicksort

23. Algorithmenanalyse

- ▶ Binäre Suche
- ▶ Messen
- ▶ O-Notation

24. Speichermanagement

- ▶ Stack, Heap, Allokation, Deallokation
- ▶ Speicherbereinigung, automatische Speicherbereinigung

25. Programmierparadigmen

Inhalt VII

- ▶ Imperative vs. Deklarative Programmierung
- ▶ Funktionale Programmierung, Funktionen als Werte, Referentielle Transparenz
- ▶ Lambda-Kalkül, Substitution, α -Konversion, β -Konversion, β -Normalform, call-by-value vs. call-by-name
- ▶ Funktionen höherer Ordnung

26. Lambda-Ausdrücke in Java

- ▶ Syntax
- ▶ Streams und Funktionen auf Streams

27. Programmierwerkzeuge