

Endklausuren EidP an der BTU, 01.08.2016 und 20.09.2016

Relevanter Stoff (pdf umfasst 3 Druckseiten)

Grundlagenwissen Java und zur Programmierung

- Eigenschaften der Programmiersprache Java
- Schlüsselwörter von Java
- wissen, was ein Compiler ist
- Unterscheidung Quellcode und Bytecode, Begriff Java Virtual Machine
- main-Methode mit Parametrisierung und Klasseneinbettung, Quelltextkommentare

~~Programmierung auf Maschinenebene, Arbeitsweise Prozessor, Geschichte Java, Funktionsweise Compiler, nichtimperative Programmierparadigmen~~

Elementare Datentypen, Variablen, Arithmetik, Typecast

- elementare Datentypen boolean, char, byte, short, int, long, float, double: jeweils Speicherplatzbedarf, Operatoren, Priorisierung der Operatoren, Notation von Literalen, Wertebereiche, Ganzzahlen im Binär- und Hexadezimalsystem
- Bitrepräsentation von float-Werten gemäß IEEE754 (Kodierung in Bitmuster)
- Begriffsdefinitionen: Wert, Typ, Variable, Konstante, Literal
- Variablen und Konstanten anlegen und Operatoren darauf anwenden
- Inkrementieren, Dekrementieren
- Bibliotheksfunktionen aus Math soweit in den Übungen verwendet
- Tastatureingaben mittels Scanner programmieren, Bildschirmausgaben mittels print, println und printf programmieren und verstehen können
- Typecast (implizit, explizit)

~~Zweierkomplement~~

Imperative Kontrollstrukturen und logische Ausdrücke

- Begriff „imperativ“, welche Kontrollstrukturen gibt es in Java
- Block und lokale Deklarationen, Hintereinanderausführung von Anweisungen
- Fallunterscheidungen mit if-elseif-else
- Schleifenarten while, do-while und for kennen und wissen, wie sie abgearbeitet werden einschl. Verlassen von Schleifen mit break
- Schleifen ineinander schachteln
- Begriff Aussage (im logischen Sinne), Wahrheitswerte
- Handling von Wahrheitswerten in Java
- aussagenlogische Operatoren und ihre Priorisierung
- Anwendung aussagenlogischer Ausdrücke in Schaltbelegungstabellen und in Bedingungen
- aussagenlogische Funktion (Boolesche Funktion) aus Schaltbelegungstabelle auslesen, Schaltbelegungstabellen programmieren

Abstimmung: Entweder ~~Programmoptimierung~~ oder Programmtransformation?

~~Wann if-else und wann switch-case, continue, wann welche Schleifenart bevorzugt
Rechengesetze der Aussagenlogik anwenden~~

Methoden selbst programmieren und daraus Klassen bauen

- Wofür sind Funktionen sinnvoll? Begriff „Methode“ in der Programmierung
- Definieren von Methoden in Java einschl. Rückgabe und Parameterübergabe
- Methoden aufrufen, Werte übergeben, Funktionswerte entgegennehmen
- Prinzip call by value
- Wissen, was in Java eine Methodenbibliothek ist und einfache Beispiele programmieren

~~tiefe Verschachtelungen von Methodenaufrufen, Parameterübernahme von der Kommandozeile, Methoden überladen~~

Rekursion

- Begriff Rekursion in der Programmierung
- Rekursionen programmieren (numerische Funktionen)
- rekursiver Abstieg und rekursiver Aufstieg
- Abarbeitung rekursiver Funktionen bzw. Methoden nachvollziehen
- (einfache) Rekursion auflösen in Iteration und/oder in explizite Formel
- Begriffe Baum, binärer Baum

~~Funktionsaufrufstack, Transformation beliebiger while-Schleifen in Rekursion, Ackermannfunktion, Baumtraversierung~~

Objektorientiert programmieren

- Begriffe „Klasse“, „Objekt“, „Attribut“, „Konstruktor“
- eigene Klassen anlegen mit Attributen, Konstruktoren und weiteren Methoden
- Zugriff auf Attributwerte innerhalb der Klasse mittels this
- einfache Getter- und Setter-Methoden programmieren und verstehen können
- Objekte einer Klasse anlegen, initialisieren und damit arbeiten können
- Begriff Kapselung, Sichtbarkeitsvermerke „public“, ohne Vermerk und „private“
- Begriff Vererbung, programmtechnische Umsetzung, „super“, überschreiben

~~Objekte liquidieren, formale Klassenbeziehungen („hat“, „ist“, „nutzt“)
Sichtbarkeitsvermerk „protected“, Package
vererbte Methoden vor Überschreiben schützen
Polymorphie~~

Felder und Graphen

- Begriff Feld in der Programmierung
- eindimensionales Feld in Java anlegen, initialisieren, mit Feldelementen arbeiten
- einfache Berechnungen auf Feldern
- Felder für Methoden zugänglich machen (Zugriff auf Feld per Parameterübergabe)
- Eigenschaften von Feldern
- Begriff „Graph“

~~mehrdimensionale Felder, Graphen programmieren,
Floyd-Warshall-Algorithmus (für kürzeste Wege in Graphen)~~

Sortieren

- Namen von mindestens vier verschiedenen Sortierverfahren kennen
- Begriffe Schlüssel, stabil, in-place, best case, worst case, aufsteigend, absteigend
- Algorithmische Ideen Selectionsort und Insertionsort, jeweils best und worst case

Abstimmung: Selectionsort oder ~~Insertionsort~~ anwenden?

~~mathematische Definition des Sortierens über totale Ordnung, average case, Wachstumsklassen, O-Notation, Mergesort, Bucketsort, Vergleich von Sortierverfahren~~

Zeichenketten, Ausnahmebehandlung und Arbeit mit Textdateien in Java

- Begriff Zeichenkette, Typ bzw. Klasse String
- Zeichenketten in Java anlegen, initialisieren, in Programmen nutzen
- Zeichenketten von Tastatur eingeben mittels Scanner
- Zeichenketten am Bildschirm ausgeben mit print, println und formatiert mit printf
- vordefinierte Methoden auf Zeichenketten kennen und anwenden können: length(), Konkatenation, charAt()
- Textdateien mittels Java-Programm anlegen, lesen, schreiben, schließen unter Nutzung vordefinierter Klassen File, BufferedReader und BufferedWriter

Abstimmung: Programmierung entweder Textdatei lesen oder ~~Textdatei schreiben~~?

~~Zeichenketteneingabe per Console, Zeichenketten vergleichen
Zeichenketten alphabetisch sortieren, Umwandlung Groß-/Kleinbuchstaben
Ausnahmebehandlung, soweit nicht zwingend für Dateiarbeit notwendig
Größe einer Datei in Byte bestimmen mittels Java Programm,
Textverschlüsselung mittels One-Time-Pad, Erzeugung (pseudo)zufälliger Zeichen~~

Dynamische Datenstruktur Lineare Liste

- Begriffe Lineare Liste, einfach verkettete Lineare Liste, (Listen)Anker
- Eigenschaften Linearer Listen
- Datensatzstruktur als Klasse beschreiben einschl. Verkettung
- Anker anlegen
- Datensätze als Objekte, die über einen Zugangspunkt (Referenz, Verweis, Adresse) angesprochen werden
- Zugriff auf Datensätze mittels Zugangspunkten kennen, programmieren und interpretieren
- Einfügen neuer Datensätze am Listenanfang programmieren
- Methode zur Bestimmung der Anzahl Datensätze in der Liste programmieren
- Methoden zum Suchen und Auslesen von Datensätzen programmieren
- Methoden programmieren, die durch die Liste laufen und dabei Datensatzinhalte auswerten
- Methoden zum Anzeigen einer Liste programmieren

~~Methoden zum Löschen von Listenelementen,
Methoden zum Einfügen von Listenelementen an beliebiger Indexposition,
Listenelemente in der Liste umordnen,
Begriff abstrakter Datentyp, Stack, Warteschlange, Ringpuffer,
dynamische Suchbäume~~