

Unterrichtsvorhaben I:

Thema: Modellierung und Implementierung von Anwendungen mit dynamischen, nichtlinearen Datenstrukturen

Inhaltsfelder

Inhaltsfeld 1: Daten und ihre Strukturierung

Inhaltsfeld 2: Algorithmen

Inhaltsfeld 3: Formale Sprachen und Automaten

Inhaltliche Schwerpunkte:

- Objekte und Klassen
- Analyse, Entwurf und Implementierung von Algorithmen
- Algorithmen in ausgewählten informatischen Kontexten
- Syntax und Semantik einer Programmiersprache

Zentrale Kompetenzen Argumentieren / Modellieren / Implementieren / Darstellen und Interpretieren / Kommunizieren und Kooperieren	Konkretisierung im Unterricht
Zu entwickelnde Kompetenzen Die Schülerinnen und Schüler • erläutern Operationen dynamischer (linearer oder nicht-linearer) Datenstrukturen (A) • analysieren und erläutern Algorithmen und Programme (A) • beurteilen die syntaktische Korrektheit und die Funktionalität von Programmen (A) • ermitteln bei der Analyse von Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M) • ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen sowie lineare und nichtlineare Datensammlungen zu (M) • modellieren abstrakte und nicht abstrakte Klassen unter Verwendung von Vererbung durch Spezialisieren und Generalisieren (M) • verwenden bei der Modellierung geeigneter Problemstellungen die	Analyse von Baumstrukturen in verschiedenen Kontexten 1. Grundlegende Begriffe (Grad, Tiefe, Höhe, Blatt, Inhalt, Teilbaum, Ebene, Vollständigkeit) 2. Aufbau und Darstellung von binären Bäumen anhand von Baumstrukturen in verschiedenen Kontexten Die Datenstruktur Binärbaum im Anwendungskontext unter Nutzung der Klasse BinaryTree 1. Analyse der Problemstellung, Ermittlung von Objekten, ihren Eigenschaften und Operationen im Anwendungskontext 2. Modellierung eines Entwurfsdiagramms und Entwicklung eines Implementationsdiagramms 3. Erarbeitung der Klasse BinaryTree und beispielhafte Anwendung der Operationen 4. Implementierung der Anwendung oder von Teilen der Anwendung 5. Traversierung eines Binärbaums im Pre-, In- und Postorderdurchlauf

Möglichkeiten der Polymorphie (M) • entwickeln iterative und rekursive Algorithmen unter Nutzung der Konstruktionsstrategien "Modularisierung" und "Teilen und Herrschen" (M) • implementieren iterative und rekursive Algorithmen auch unter Verwendung von dynamischen Datenstrukturen (I) • modifizieren Algorithmen und Programme (I) • nutzen die Syntax und Semantik bei der Implementierung und Analyse von Programmen (I) • interpretieren Fehlermeldungen und korrigieren den Quellcode (I) • testen Programme systematisch anhand von Beispielen (I) • stellen lineare und nichtlineare Strukturen grafisch dar und erläutern ihren Aufbau (D) • stellen iterative und rekursive Algorithmen umgangssprachlich und grafisch dar (D).	Die Datenstruktur <i>binärer Suchbaum</i> im Anwendungskontext unter Verwendung der Klasse <code>BinarySearchTree</code> 1. Analyse der Problemstellung, Ermittlung von Objekten, ihren Eigenschaften und Operationen 2. Modellierung eines Entwurfsdiagramms und Entwicklung eines Implementationsdiagramms 3. grafische Darstellung eines binären Suchbaums und Erarbeitung der Struktureigenschaften 4. Erarbeitung der Klasse <code>BinarySearchTree</code> und Einführung des Interface Item zur Realisierung einer geeigneten Ordnungsrelation 5. Implementierung der Anwendung von Teilen der Anwendung inklusive einer sortierten Ausgabe des Baums Übung und Vertiefungen der Verwendung von Binärbäumen oder binären Suchbäumen anhand weiterer Problemstellungen
--	---

Unterrichtsvorhaben II:

Thema: Endliche Automaten und formale Sprachen

Inhaltsfelder

Inhaltsfeld 3: Formale Sprachen und Automaten

Inhaltliche Schwerpunkte:

- Endliche Automaten
- Grammatiken regulärer Sprachen
- Möglichkeiten und Grenzen von Automaten und formalen Sprachen

Zentrale Kompetenzen

Argumentieren / Modellieren / Darstellen und Interpretieren /
Kommunizieren und Kooperieren

Zu entwickelnde Kompetenzen

Die Schülerinnen und Schüler

- analysieren und erläutern die Eigenschaften endlicher Automaten einschließlich ihres Verhaltens auf bestimmte Eingaben (A)
- analysieren und erläutern Grammatiken regulärer Sprachen (A)
- zeigen die Grenzen endlicher Automaten und regulärer Grammatiken im Anwendungszusammenhang auf (A)
- ermitteln die formale Sprache, die durch eine Grammatik erzeugt wird (A)
- entwickeln und modifizieren zu einer Problemstellung endliche Automaten (M)
- entwickeln zur akzeptierten Sprache eines Automaten die zugehörige Grammatik (M)
- entwickeln zur Grammatik einer regulären Sprache einen zugehörigen endlichen Automaten (M)
- modifizieren Grammatiken regulärer Sprachen (M)
- entwickeln zu einer regulären Sprache eine Grammatik, die die

Konkretisierung im Unterricht

Endliche Automaten

1. vom Automaten in den SuS bekannten Kontexten zur formalen Beschreibung eines endlichen Automaten
2. Untersuchung, Darstellung und Entwicklung endlicher Automaten

Untersuchung und Entwicklung von Grammatiken regulärer Sprachen

1. Erarbeitung der formalen Darstellung regulärer Grammatiken
2. Untersuchung, Modifikation und Entwicklung von Grammatiken
3. Entwicklung von endlichen Automaten zum Erkennen regulärer Sprachen, die durch Grammatiken gegeben werden
4. Entwicklung regulärer Grammatiken zu endlichen Automaten

Grenzen endlicher Automaten

Sprache erzeugt (M) • stellen endliche Automaten in Tabellen oder Graphen dar und überführen sie in die jeweils andere Darstellungsform (D) • ermitteln die Sprache, die ein endlicher Automat akzeptiert (D) • beschreiben an Beispielen den Zusammenhang zwischen Automaten und Grammatiken (D)	
--	--

Unterrichtsvorhaben III:

Thema: Prinzipielle Arbeitsweise eines Computers und Grenzen der Automatisierbarkeit

Inhaltsfelder

Inhaltsfeld 4: Informatiksysteme

Inhaltsfeld 5: Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte:

- Einzelrechner und Rechnernetzwerke
- Grenzen der Automatisierung

Zentrale Kompetenzen Argumentieren / Kommunizieren und Kooperieren	Konkretisierung im Unterricht
Zu entwickelnde Kompetenzen Die Schülerinnen und Schüler • erläutern die Ausführung eines einfachen maschinennahen Programms sowie die Datenspeicherung auf einer "Von-Neumann-Architektur" (A) • untersuchen und beurteilen Grenzen des Problemlösens mit Informatiksystemen (A)	Von-Neumann-Architektur und Ausführung maschinennaher Programme 1. prinzipieller Aufbau einer von-Neumann-Architektur mit CPU, Rechenwerk, Steuerwerk, Register und Hauptspeicher 2. maschinennahe Befehle und ihre Repräsentation in einem Binär-Code, der in einem Register gespeichert werden kann 3. Analyse und Erläuterung der Funktionsweise eines einfachen maschinennahen Programms Grenzen der Automatisierbarkeit 1. Vorstellung des Halteproblems 2. Unlösbarkeit des Halteproblems 3. Beurteilung des Einsatzes von Informatiksystemen hinsichtlich prinzipieller Möglichkeiten und prinzipieller Grenzen