

Schulcurriculum Informatik Jahrgang 12/13

Lehrbuch: Schönigh, Informatik 2, ISBN 978-3-14-037127-8; Programmiersprache: Java mit BlueJ-Entwicklungsumgebung
 Verpflichtung zu insgesamt 8 Wochen Projektarbeit, wobei 4 Wochen innerhalb der ersten drei Kurssemester stattfinden müssen

Q1: Datenstrukturen und Algorithmen

Zeit	Unterrichtsreihe	Verpflichtende Inhalte/Lernziele laut KC	Kompetenzen	Hinweise/ schulinterne Konkretisierungen
6-8 Wochen bis max. Herbstferien	Einführung in die objektorientierte Programmierung Schwerpunkt Datenstrukturen Schönigh Kapitel 1 und 2	<ul style="list-style-type: none"> • unterscheiden zwischen primitiven Datentypen u. Objektreferenzen • eA: entwerfen Klassen und deren Beziehungen (Assoziation, Vererbung) und stellen diese durch Klassendiagramme dar • erläutern das Prinzip, mehrere Daten des gleichen Typs in Reihungen zu verwalten • erläutern das Prinzip der Datenstrukturen Stapel, Schlange und dynamische Reihung • analysieren die Funktionsweise eines gegebenen Algorithmus • unterscheiden zwischen lokalen und globalen Variablen • verwenden Übergabeparameter und Rückgabewerte in Operation 		Darstellung von Objekten/Klassen wie im Buch auch im gA-Kurs Das Grundprinzip Zählschleife zur linearen Datenverarbeitung aus 11 wieder aufgreifen für Buchaufgaben und kleine Übungs-Algorithmen Details zu ADT-Operationen/linearer Verkettung nur knapp/exemplarisch Optionen: Swing-GUI-Programmierung oder Implementation ADTs
4 Wochen bis Ende November	Kleinprojekt ggf. als Langzeit-Hausaufgabe	<ul style="list-style-type: none"> • verwenden geeignete Variablentypen zur Speicherung von Werten • entwerfen und implementieren Algorithmen unter Verwendung von gegebenen und eigenen Klassen/Objekten • entwerfen und implementieren Algorithmen unter Verwendung von ein- und zweidimensionalen Reihungen, Stapeln, Schlangen und dynamische Reihungen 		geeignet sind hierzu Kleinprojekte wie bspw. Implementation der komplexeren Buchaufgaben Die Projektarbeit sollte zur Dokumentation UML Diagramme und Quelltextkommentare umfassen
4-6 Wochen Dezember/ Januar	Algorithmen Schönigh Kapitel 3 (inkl. Seite 82: Datenschutz)	<ul style="list-style-type: none"> • analysieren die Funktionsweise eines gegebenen Algorithmus • stellen Algorithmen in schriftlich verbalisierter Form dar • diskutieren die Chancen und Risiken der automatisierten Datenanalyse. • eA: beurteilen die Effizienz von Algorithmen unter Abschätzung des Speicherbedarfs und der Zahl der Operationen • eA: erläutern das Konzept der Rekursion an gegebenen Beispielen • eA: entwerfen und implementieren rekursive Algorithmen und erläutern dabei die Strategie „Teile und herrsche“ 		Die Entwicklung eigener Suchalgorithmen führt automatisch zur Frage nach Auswahlkriterien wie Effizienz alle Inhalte eignen sich auch für GK Ausreichend und empfehlenswert ist die Auswahl folgender Algorithmen: Sortieren durch Einfügen/Auswahl, Quicksort und lineare Suche Zu Algorithmen gibt es zahlreiche kurzweilige Materialien im Netz.

Gedanken: GK-Inhalte sind wegen der Verpflichtung zur Implementation von Algorithmen mit ADTs und Projektarbeit für 3 Stunden umfangreich, daher ist der Algorithmenblock ganz ans Ende gerückt und kann dann in der Tiefe nach Belieben/Zeit bearbeitet werden.

Q2: Automaten und Datenübertragung

Zeit	Unterrichtsreihe	Verpflichtende Inhalte/Lernziele laut KC	Kompetenzen	Hinweise/ schulinterne Konkretisierungen
6 Wochen	Endliche Automaten Schöningh Kapitel 4 (nur Abschnitt 1)	<ul style="list-style-type: none"> • beschreiben den Aufbau und die Funktionsweise eines deterministischen endlichen Automaten (DEA). • beschreiben den Aufbau und die Funktionsweise eines endlichen Automaten mit Ausgabe (Mealy-Automat). • entwickeln und implementieren Automatenmodelle in Form von Zustandsgraphen. • analysieren die Funktion eines durch einen Zustandsgraphen vorgegebenen Automaten. • erläutern die Grenzen endlicher Automaten bei der Problemlösung. • eA: beschreiben den Aufbau und die Funktionsweise eines Kellerautomaten als erweitertes Automatenmodell. 		Neben dem Einsatz von JFlap bieten sich als Übungsmöglichkeit fürs Programmieren auch Java-Implementationen an (vgl. zwei mdl. Abivorschläge von Soltek) In jahrgangsübergreifenden Kursen kann hier im Q4 Scanner und Parser (als Übung für Algorithmen, s.a. SuM) oder Schaltwerke Thema sein Weiterer Übergang könnte die Fehlerkorrektur/Fehlerprüfung sein
max. 4 Wochen beides bis Ostern	Formale Sprachen (nur eA) Schöningh 4.2	<ul style="list-style-type: none"> • nennen Eigenschaften formaler Sprachen im Vergleich zu natürlichen Sprachen. • beschreiben die von einer Grammatik erzeugte Sprache. • entwerfen reguläre und kontextfreie Grammatiken. 		wegen der Beschränkung auf reguläre und kontextfreie Sprachen auch für gA geeignet.
6-8 Wochen	Aspekte der Datenübertragung Schöningh Kapitel 6 (nur teilweise) Schöningh Kapitel 5 (nur teilweise)	<ul style="list-style-type: none"> • entwerfen und implementieren ein Protokoll zur Übertragung von Daten über einen Kommunikationskanal • beschreiben Möglichkeiten, Daten zu komprimieren, u.a. Lauflängencodierung, Huffman-Codierung. • eA: erläutern die Vor- und Nachteile verlustfreier und verlustbehafteter Kompression von Daten. • eA: erläutern Möglichkeiten der Fehlererkennung und der Fehlerkorrektur bei der Datenübertragung, u. a. Paritätsbit und (7,4)-Hamming-Code. • eA: entwerfen und implementieren ein Kompressionsverfahren zu einem gegebenen Sachverhalt. • eA: erläutern das Prinzip der Datenstruktur Binärbaum • eA: entwerfen und implementieren Algorithmen zur Ausgabe der Daten eines Binärbaums in pre-, post- und inorder- Reihenfolge • eA: entwerfen und implementieren Algorithmen zur Suche und zum Einfügen in binäre Suchbäume 		Einstieg könnte neben Buchideen auch Arduino Glasfasermodell sein Scratch und Filius-Einsatz ist denkbar Die Aufgabe einen verpixelten Smiley zu übertragen führt zu Kompression Implementation kann hier jeweils auf verbaler Ebene (Rollenspiel) bleiben oder in Netzwerkspielprojekt enden. In SuM sind geeignete Java-Gerüste enthalten für Huffman-Implementation Die Themen Binärbaum, Baumtraversierung und Suchen/Einfügen sind bei Verzicht auf Implementation aller Operationen auch gA geeignet

Gedanken: Bietet im Facharbeitssemester und auch für jahrgangs- oder niveauübergreifende Kurse so hoffentlich möglichst viel Spielraum und natürlich die Möglichkeit die Themenblöcke Automaten und Datenübertragung zu tauschen oder bspw. noch weitere Vertiefungen (siehe KC) zu machen.

Q3: Datenbanken und Kryptologie

Zeit	Unterrichtsreihe	Verpflichtende Inhalte/Lernziele laut KC	Kompe- tenzen	Hinweise/ schulinterne Konkretisierungen
max. 6 Wochen	Datenbanken Schöningh Kapitel 7	<ul style="list-style-type: none"> • erläutern den Aufbau relationaler Datenbanken unter Verwendung der Begriffe Datensatz, Attribut, Primär-/Fremdschlüssel, Tabelle • nennen Beispiele für Einfüge-, Änderungs- und Löschanomalien. • untersuchen ein Schema auf Anomalien und Redundanzen. • formulieren Abfragen und Verbundabfragen über mehrere Tabellen auch unter Verwendung von Aggregatfunktionen. • eA: interpretieren ein gegebenes ER-Diagramm. • eA: modellieren Datenbanken unter Verwendung des ER-Modells. • eA: setzen ein ER-Modell in ein relationales Schema um. • eA: beurteilen und verändern eine gegebene Datenbankmodellierung. 		SQL-Tutorial im Internet zur Übung von SQL-Abfragen nutzen ER-Diagramme auch im gA-Kurs (vgl. Datenstrukturen und Algorithmen) ggf. auch Normalisierung um Methoden zur Verbesserung schlechter Schemata zu geben Eine Java-Implementation von DB-Zugriffen kann einen Übergang nahezu aller vorigen Themen bis zur Kryptologie herstellen, da nur eine verschlüsselte Speicherung und Übertragung der DB-Daten sicher ist.
4-6 Wochen bis max. Ende November	Projektphase oder Vertiefung	<ul style="list-style-type: none"> • je nach Thema 		siehe Liste am Ende die Dokumentation und Projektentwicklung muss bei der Bewertung berücksichtigt werden.
4-6 Wochen	Kryptologie	<ul style="list-style-type: none"> • beschreiben das Prinzip der polyalphabetischen Substitution, u. a. am Beispiel des Vigenère-Verfahrens. • beurteilen die Sicherheit eines gegebenen symmetrischen Verschlüsselungsverfahrens. • beschreiben und unterscheiden die Prinzipien der symmetrischen und asymmetrischen Verschlüsselung. • beschreiben Anwendungsbereiche für symmetrische bzw. asymmetrische Verschlüsselungsverfahren. • erläutern das Prinzip von digitalen Signaturen und Zertifikaten. • eA: entwerfen und implementieren ein symmetrisches Verschlüsselungsverfahren. • eA: erläutern die prinzipielle Funktionsweise eines modernen symmetrischen Blockchiffreverfahrens. 		Anknüpfung an 11 und gute Übungsmöglichkeit fürs Programmieren obwohl im gA keine Implementation vorgesehen ist. ggf. Kryptoanalyse für besseres Sicherheitsverständnis alte Zentralabiklausuren enthalten diverse kryptographische Verfahren http://it-lehren.de/asym/Integritaet-und-Authentizitaet-mit-digitaler-Unterschrift-sicherstellen.html

Gedanken: Im Q3 soll das Vorabi geschrieben werden – ggf. kann dies im Rahmen der Projektphase vorbereitet werden; ein Verschieben der Projektphase (ans Ende oder ins Q4) wäre denkbar dürfte aber im Q4 i.d.R. zu Terminengpässen führen.

Q4: Ausgewählte Kapitel der Informatik

Zeit	Unterrichtsreihe	Verpflichtende Inhalte/Lernziele laut KC	Kompetenzen	Hinweise/ schulinterne Konkretisierungen
4 Wochen	Vertiefung oder Projektphase (notfalls)	• je nach Thema		
4 Wochen	Wiederholung und Vertiefung	• je nach Thema		

Gedanken: Die Q4 Länge beträgt eventuell nur knappe 9 Wochen wobei eine Klausur erforderlich ist und fürs Abitur möglichst viel wiederholt und geübt werden muss; die Projektphase müsste hier wegen der terminlichen Klausurvorbereitungs- und Bewertungsrahmenbedingungen eventuell auch in der Mitte liegen; falls jahrgangsübergreifende Kurse bestehen würde die zusätzliche Projektphase hier den Unterricht endgültig atomisieren.

Projektideen für Q3/Q4

- App zur Termberechnung gemäß SuM-Konzept (gute Übung der ADTs)
- App zur Verwaltung von Beständen, Buchungen, Messwerten, ... in einer Datenbank (Übung zu Datenbanken und Datenstrukturen)
- App zum Austausch von Textnachrichten (Übung zu Datenübertragung)
- ...

Erfahrungen liegen zur Termberechnungs-App vor; für die anderen Vorschläge ließen sich den Schülern aber leicht verständliche Gerüste aus Informatik-Unterricht-Kontexten oder Computerzeitungen bereitstellen; gemäß KC-Fortbildung wäre auch eine Themenfindung durch Schüler selbst und die Nutzung von Projektmanagementmethoden wünschenswert – auch wenn dies am Ende ein Scheitern durch Überforderung bedeuten kann.

Möglichkeiten zur Vertiefung (bspw. als Klausurgrundlage in Q4)

- Scanner/Parser (gemäß Schönigh Kapitel 4 und SuM Band III Kapitel 9)
- Algorithmen auf Graphen (gemäß Schönigh Kapitel 5)
- Schaltwerke (eventuell bis zu von-Neumann-Modell)