



## Informatik im Sekundarbereich II

### Informatikunterricht in der Qualifikationsphase der gymnasialen Oberstufe

Mithilfe zunehmend textbasierter Entwicklungsumgebungen werden die Kenntnisse zum algorithmischen Problemlösen vertieft und die Schülerinnen und Schüler (ggfs. in Kooperationskursen) in die Lage versetzt, selbstständig Algorithmen für Problemstellungen zu entwickeln und zu testen. Sie machen sich in unterschiedlichen Kontexten mit den Prinzipien der algorithmischen Problemlösung und der Implementierung ihrer Lösungsideen vertraut. Dazu entwerfen sie eigenen Operationen, Klassen und Objekte, um zu einer strukturierten Realisierung ihrer Lösungsideen zu gelangen.

Weiterhin vertiefen sie ihre Kenntnisse in der Codierung, Verschlüsselung und Übertragung von Daten und begreifen dabei, dass Informatik in der Lebenswirklichkeit ein unverzichtbarer Bestandteil ist. Strukturen von Datenbanken und deren Auswertung sind ein weiterer Schwerpunkt. Die automatisierte Datenanalyse hat eine hohe gesellschaftliche Relevanz. So ist es notwendig, sowohl die rechtlichen Aspekte als auch die technischen Möglichkeiten zum Schutz persönlicher oder sensibler Daten zu kennen.

Neu in der Qualifikationsphase ist die zustandsbasierte Modellierung zur Entwicklung und Realisierung von technischen Systemen und Programmen. Hier soll das Verhalten technischer Systeme in Form von endlichen Automaten modelliert werden, um ihren Aufbau und ihre Funktionsweise zu erschließen. Auf erhöhtem Niveau schließt dies die Beschreibung in Form von formalen Sprachen sowie deren Synthese und Analyse mit Hilfe von Grammatiken ein.

#### Berufsorientierung:

- Die Schülerinnen und Schüler erkennen konkrete berufliche Situationen, in denen informatische Kompetenzen unverzichtbar sind.
- Die Schülerinnen und Schüler entwickeln ein Bewusstsein, dass informatische Bildung für die Berufswegplanung ständig an Bedeutung gewinnt.
- Im IT-Bereich handelt es sich oft um Berufsfelder, in denen Frauen derzeit noch unterrepräsentiert sind. Diesem Missverhältnis entgegenzuwirken, ist ein weiteres Ziel der Mädchenbildung an der Liebfrauenschule Vechta.

Hinweise zum Layout:

*Kursiv gedruckt: nur relevant für Kurse auf erhöhtem Anforderungsniveau*

**Grün geschrieben: Fächerübergreif**



Lernfeld / Modul	Die Schülerinnen und Schüler ...	Methodische Hinweise / Materialien
<p>Algorithmen und Datenstrukturen</p> <p><b>Grundlagen der Algorithmik</b></p>	<ul style="list-style-type: none"><li>- analysieren die Funktionsweise eines gegebenen Algorithmus.</li><li>- stellen Algorithmen in schriftlich verbalisierter Form dar.</li><li>- <i>beurteilen die Effizienz von Algorithmen unter Abschätzung des Speicherbedarfs und der Zahl der Operationen.</i></li></ul> <p>[Kompetenzen laut KC: PK1.2, PK2 IK1.2, IK2.2, <i>PK1.1, PK1.2, PK2.2, PK2.3 IK2.2</i>]</p> <ul style="list-style-type: none"><li>- verwenden geeignete Variablentypen zur Speicherung von Werten.</li><li>- unterscheiden zwischen lokalen und globalen Variablen.</li><li>- unterscheiden zwischen primitiven Datentypen und Objektreferenzen.</li></ul> <p>[Kompetenzen laut KC: PK1.2, PK2 IK1.2, IK2.2]</p> <ul style="list-style-type: none"><li>- verwenden Übergabeparameter und Rückgabewerte in Operationen.</li><li>- <i>erläutern das Konzept der Rekursion an gegebenen Beispielen.</i></li><li>- <i>entwerfen und implementieren rekursive Algorithmen.</i></li><li>- <i>erläutern die Strategie „Teile und herrsche“ beim Entwurf rekursiver Algorithmen.</i></li></ul> <p>[Kompetenzen laut KC: PK1.2, PK2 IK1.2, IK2.2, <i>PK1.1, PK1.2, PK2.2, PK2.3 IK2.2</i>]</p>	<p>ca. 12 Wochen in 12.1 (inklusive Projektarbeit) Programmiersprache: Java/Processing Grundkurs: Java-Editor oder Processing Leistungskurs: Java-Editor oder Eclipse oder NetBeans Einstieg mit einfachen klassischen Algorithmen wie <b>ggT und kgV (→ Mathematik)</b></p> <p>Sortierverfahren, z.B.:</p> <ul style="list-style-type: none"><li>- BubbleSort</li><li>- MergeSort</li></ul> <p><i>Rekursive Algorithmen, z.B.</i></p> <ul style="list-style-type: none"><li>- <i>Fakultät (→ Mathematik)</i></li><li>- <i>Binomialkoeffizienten (→ Mathematik)</i></li><li>- QuickSort</li><li>- <i>hinsichtlich Laufzeitanalyse: Laufzeit in Abhängigkeit der Anzahl der zu sortierenden Elemente</i></li></ul> <p><i>etwa nach den Herbstferien:</i> <b>Projektarbeit:</b> Erklärvideos erstellen zum Themenkomplex „Sortieren und Suchen“</p> <ul style="list-style-type: none"><li>- 2 Wochenstunden Unterricht</li><li>- 3 Wochenstunden Projektarbeit</li></ul>
<p>Informationen und Daten</p> <p><b>Codierung und Übertragung von Daten</b></p>	<ul style="list-style-type: none"><li>- beschreiben Möglichkeiten, Daten zu komprimieren, u. a. Lauflängencodierung, Huffman-Codierung.</li><li>- <i>entwerfen und implementieren ein Kompressionsverfahren zu einem gegebenen Sachverhalt.</i></li><li>- <i>erläutern die Vor- und Nachteile verlustfreier und verlustbehafteter Kompression von Daten.</i></li><li>- <i>erläutern Möglichkeiten der Fehlererkennung und der Fehlerkorrektur bei der Datenübertragung, u. a. Paritätsbit, (7,4)-Hamming-Code.</i></li></ul> <p>[Kompetenzen laut KC: PK1.1, PK2.1, PK2.2, PK3.3 IK1.1, IK2.3, IK3.4, <i>PK1.1, PK2.1, PK2.2, PK3.3 IK1.1, IK2.3</i>]</p> <ul style="list-style-type: none"><li>- entwerfen und implementieren ein Protokoll zur Übertragung von Daten über einen Kommunikationskanal</li></ul> <p>[Kompetenzen laut KC: PK1.1, PK2.1, PK2.2, PK3.3 IK1.1, IK2.3, IK3.4]</p>	<p>ca. 6 Wochen in 12.1 (gA: 4 Wochen) nur wenig Implementierungen – nur z.B. Lauflängencodierung implementieren</p>



Informationen und Daten <b>Kryptologie</b>	<ul style="list-style-type: none"><li>- beschreiben das Prinzip der polyalphabetischen Substitution, u.a. am Beispiel des Vigenère-Verfahrens.</li><li>- beurteilen die Sicherheit eines gegebenen symmetrischen Verschlüsselungsverfahrens.</li><li>- beschreiben und unterscheiden die Prinzipien der symmetrischen und asymmetrischen Verschlüsselung.</li><li>- beschreiben Anwendungsbereiche für symmetrische bzw. asymmetrische Verschlüsselungsverfahren.</li><li>- erläutern das Prinzip von digitalen Signaturen und Zertifikaten.</li><li>- <i>entwerfen und implementieren ein symmetrisches Verschlüsselungsverfahren.</i></li><li>- <i>erläutern die prinzipielle Funktionsweise eines modernen symmetrischen Blockchiffreverfahrens.</i></li></ul> <p>[Kompetenzen laut KC: PK3.3 IK4.3; PK2.1, PK2.2, PK3.3 IK1.1, IK2.3]</p>	ca. 8 Wochen in 12.1/12.2  Sinnvolle Ergänzungen: <ul style="list-style-type: none"><li>- Abgrenzung zu Codierung</li><li>- AES</li><li>- RSA-Verschlüsselung</li><li>- Diffie-Hellman-Schlüsseltausch</li></ul> Implementierung: <ul style="list-style-type: none"><li>- 2-dimensionale Reihungen nutzen (abiturrelevant)</li><li>- Zeichenkettenoperationen</li></ul>
Informationen und Daten <b>Datenschutz</b>	<ul style="list-style-type: none"><li>- diskutieren die Chancen und Risiken der automatisierten Datenanalyse.</li></ul> <p>[Kompetenzen laut KC: PK3.3; IK4.1]</p>	ca. 2 Wochen in 12.2
Informationen und Daten <b>Datenbanken</b>	<ul style="list-style-type: none"><li>- erläutern den Aufbau relationaler Datenbanken unter Verwendung der Begriffe Datensatz, Attribut, Primärschlüssel, Fremdschlüssel und Tabelle.</li><li>- nennen Beispiele für Einfüge-, Änderungs- und Löschanomalien.</li><li>- untersuchen ein gegebenes Datenbankschema auf Anomalien und Redundanzen.</li><li>- formulieren einfache Abfragen und Verbundabfragen über mehrere Tabellen.</li><li>- formulieren Abfragen an Datenbanken unter Verwendung von Aggregatfunktionen.</li><li>- <i>interpretieren ein gegebenes ER-Diagramm.</i></li><li>- <i>modellieren Datenbanken unter Verwendung des ER-Modells.</i></li><li>- <i>setzen ein ER-Modell in ein relationales Schema um.</i></li><li>- <i>beurteilen und verändern eine gegebene Datenbankmodellierung.</i></li></ul> <p>[Kompetenzen laut KC: PK1.2, PK1.3, PK2.2, PK3.1, IK2.4, PK1.4, PK1.5, PK3.1, PK3.2 IK1.5]</p>	ca. 8 Wochen in 12.2  Tutorial mit Lösungen: <a href="https://www.luo-darmstadt.de/sqltutorial/index.html">https://www.luo-darmstadt.de/sqltutorial/index.html</a>  Mögliche Alternative bzw. Ergänzung: <a href="https://inf-schule.de">https://inf-schule.de</a>



Lernfeld / Modul	Die Schülerinnen und Schüler ...	Methodische Hinweise / Materialien
Algorithmen und Datenstrukturen  <b>Klassen und Objekte</b>	<ul style="list-style-type: none"><li>- entwerfen und implementieren Algorithmen unter Verwendung von gegebenen und eigenen Klassen/Objekten.</li><li>- <i>entwerfen Klassen und deren Beziehungen (Assoziation, Vererbung) und stellen diese durch Klassendiagramme dar.</i></li></ul> <p>[Kompetenzen laut KC: PK1.2, PK1.3, PK2.1 IK1.4, <i>PK1.4, PK1.5, PK3.1, PK3.2 IK1.4</i>]</p>	ca. 6-7 Wochen in 13.1 (gA: ca. 8-10 Wochen)  <b>Projekt:</b> Spieleentwicklung - 2 Wochenstunden Unterricht - 3 Wochenstunden Projektarbeit
Algorithmen und Datenstrukturen  <b>Statische und dynamische Datenstrukturen</b>	<ul style="list-style-type: none"><li>- erläutern das Prinzip, mehrere Daten des gleichen Typs in Reihungen zu verwalten, zu suchen und zu sortieren.</li><li>- entwerfen und implementieren Algorithmen unter Verwendung von ein- und zweidimensionalen Reihungen.</li><li>- erläutern das Prinzip der Datenstrukturen Stapel, Schlange und dynamische Reihung.</li><li>- entwerfen und implementieren Algorithmen unter Verwendung der Datenstrukturen Stapel, Schlange und dynamische Reihung.</li></ul> <p>[Kompetenzen laut KC: PK1.3, PK2.1, PK2.2 IK1.3, IK2.2]</p>	ca. 5-6 Wochen in 13.1 (gA: ca. 8 Wochen)  Mögliche Ergänzungen: - Algorithmen auf Graphen - genetische Algorithmen - Komplexitätstheorie



<p>Automaten und Sprachen</p> <p><b>Automatenmodelle</b></p>	<ul style="list-style-type: none"><li>- beschreiben den Aufbau und die Funktionsweise eines deterministischen endlichen Automaten (DEA).</li><li>- beschreiben den Aufbau und die Funktionsweise eines endlichen Automaten mit Ausgabe (Mealy-Automat).</li><li>- entwickeln und implementieren Automatenmodelle in Form von Zustandsgraphen.</li><li>- analysieren die Funktion eines durch einen Zustandsgraphen vorgegebenen Automaten.</li><li>- erläutern die Grenzen endlicher Automaten bei der Problemlösung.</li><li>- <i>beschreiben den Aufbau und die Funktionsweise eines Kellerautomaten als Erweiterung des Modells des endlichen Automaten.</i></li></ul> <p>[ Kompetenzen laut KC: PK1.1, PK1.5, PK2.2, PK2.3, PK3.1, PK3.2 IK3.1, IK3.2, IK4.3, IK3.1]</p>	<p>ca. 5-6 Wochen in 13.1 (gA: ca. 6 Wochen in 13.1/13.2)</p> <p>Mögliche Ergänzungen:</p> <ul style="list-style-type: none"><li>- Turingmaschine</li><li>- zelluläre Automaten</li><li>- Petri-Netze</li><li>- Schaltnetze</li><li>- Schaltwerke</li></ul>
<p>Automaten und Sprachen</p> <p><b>Formale Sprachen</b></p>	<ul style="list-style-type: none"><li>- <i>nennen Eigenschaften formaler Sprachen im Vergleich zu natürlichen Sprachen.</i></li><li>- <i>beschreiben die von einer Grammatik erzeugte Sprache.</i></li><li>- <i>entwerfen reguläre und kontextfreie Grammatiken für formale Sprachen.</i></li><li>- <i>erläutern den Zusammenhang zwischen regulären Grammatiken und endlichen Automaten.</i></li></ul> <p>[ Kompetenzen laut KC: PK3.1 IK3.4]</p>	<p>ca. 2-3 Wochen in 13.1/13.2 (entfällt auf gA)</p> <p>Mögliche Ergänzungen:</p> <ul style="list-style-type: none"><li>- Chomsky Hierarchie</li><li>- L-Systeme</li></ul>
<p>Algorithmen und Datenstrukturen</p> <p><b>Statische und dynamische Datenstrukturen</b></p>	<ul style="list-style-type: none"><li>- <i>erläutern das Prinzip der Datenstruktur Binärbaum.</i></li><li>- <i>entwerfen und implementieren Algorithmen zur Ausgabe der Daten eines Binärbaums in Pre-, Post- und Inorder-Reihenfolge.</i></li><li>- <i>entwerfen und implementieren Algorithmen zur Suche und zum Einfügen in binäre Suchbäume.</i></li></ul> <p>[ Kompetenzen laut KC: PK1.3, PK2.2 IK1.3, IK2.2]</p>	<p>ca. 4 Wochen in 13.2 (entfällt auf gA)</p> <p>Mögliche Ergänzungen:</p> <ul style="list-style-type: none"><li>- Algorithmen auf Graphen</li><li>- genetische Algorithmen</li><li>- Komplexitätstheorie</li></ul>