Saarland

Ministerium für Bildung,
Kultur und Wissenschaft

Achtjähriges Gymnasium

Lehrplan Informatik

für die Einführungsphase
der gymnasialen Oberstufe

Februar 2006

Vorbemerkungen

Zu Beginn des Informatikunterrichtes in der Einführungsphase der gymnasialen Oberstufe soll eine **Einführung in die Informatik** wesentliche fachliche Grundkenntnisse vermitteln.

Der Themenbereich **Modellieren und Entwerfen** beschäftigt sich mit der Abbildung konkreter Problemstellungen auf Datenstrukturen und zugehörige Lösungsalgorithmen. In diesem Kapitel stehen die Modellbildung (als Abstraktion von der Wirklichkeit) und die Beschreibung der Modellkomponenten und Lösungsstrategien im Vordergrund. Die thematische Abgrenzung von der **Einführung in die Programmentwicklung**, welche sich mit der Implementierung der Datenstrukturen und Algorithmen in einer Programmiersprache beschäftigt, ist gewünscht. Die Phasen der Problemlösung sollen an Beispielen erläutert und veranschaulicht werden. Diese Beispiele können später bei der Einführung in die Programmentwicklung wieder aufgegriffen und durch ihre Implementierung ergänzt werden.

Die Abschnitte **Modellieren und Entwerfen** und **Einführung in die Programmentwicklung** können verzahnt behandelt werden: Als Beispiele bieten sich Algorithmen aus der klassischen Kryptographie (à Thema Klassische kryptographische Verfahren) an.

Im Kapitel **Klassische kryptographische Verfahren** werden historisch bedeutsame Verfahren besprochen. Diese symmetrischen Verfahren erlauben die Vermittlung wichtiger kryptographischer Grundbegriffe. Das Thema wird im 2. Jahr der Hauptphase mit der Behandlung asymmetrischer Verfahren fortgesetzt.

Im fakultativen Teil können die in den vorangegangenen Kapiteln erworbenen Kenntnisse und Fertigkeiten im Modellieren und Programmieren aufgegriffen und vertieft werden.

Informatik, Einführungsphase	
Grundbegriffe	**5 Stunden**

VERBINDLICHE INHALTE	Vorschläge und Hinweise
Aufbau und Funktionsweise von Informatiksystemen • **Hardware:** Zentraleinheit, Mikroprozessor, Speicher, Eingabegeräte, Ausgabegeräte • **Software:** Betriebssysteme, Entwicklungswerkzeuge, Anwendersoftware	
Rechnerarchitektur, Von-Neumann-Prinzipien	
Zahlensysteme und Codierung • Binär-, Hexadezimalsystem, • Bit und Byte, • ASCII-Code	2 (Mathematik Klasse 5)
• Codierung von Zeichen und Zahlen, • Zeichenketten, Zeichensätze	**Literatur/Medien:** Rechnerarchitektur und Von-Neumann-Prinzip: • Gasper, Leiß, Spengler, Stimm: Technische und theoretische Informatik, BSV-Verlag 1992 • Duden Informatik

Informatik, Einführungsphase	
Modellieren und Entwerfen	**10 Stunden**

VERBINDLICHE INHALTE	Vorschläge und Hinweise
Strategien und Konzepte bei der Entwicklung von Computerprogrammen	
Phasen des Problemlösungsprozesses:	An Beispielen unterschiedlicher Komplexität soll die Notwendigkeit eines geplanten, zielorientierten Vorgehens bei der Konzeption und Erstellung von Computerprogrammen verdeutlicht werden.
1. Beschreiben und Analysieren eines Problems aus der realen Welt	Die Beschreibung einer realen Situation führt zur Formulierung der Problemstellung. Nach Erfassen und Beschreiben des Problems werden Lösungsstrategien erkannt und als Text formuliert.
2. Erarbeitung eines Modells	Der Ausschnitt aus der realen Welt (das Original) wird auf ein Modell abgebildet. Beim Modellbildungsprozess wird abstrahiert: nur relevante Aspekte des Originals werden berücksichtigt und gegebenenfalls durch Idealisierung die Komplexität verringert. Ergebnis ist ein Modell, das umgangssprachlich oder in anschaulich-grafischer Form beschrieben wird.
3. Entwurf einer detaillierten Problemlösungsstrategie, Beschreibung von Modellkomponenten und Lösungsstrategien in einer auf den Computer übertragbaren Form	Aus der sprachlichen oder grafischen Beschreibung des Modells werden die Datenstrukturen zur Darstellung der Modellobjekte und die den Ablaufstrukturen und Wechselwirkungen entsprechenden Algorithmen abgeleitet. Datenstrukturen und Algorithmen werden durch geeignete graphisch–deskriptive Medien (z. B. Struktogramme, Flussdiagramme, UML–Modellierung) veranschaulicht.
4. Implementierung	è Programmentwicklung Realisierung des Modells mit den Mitteln eines geeigneten Entwicklungssystems als lauffähiges Programm.

Informatik, Einführungsphase	
Einführung in die Programmentwicklung	**20 Stunden**

VERBINDLICHE INHALTE	Vorschläge und Hinweise
Grundlegende Sprachelemente • Bezeichner, Literale und Namenskonventionen • Variablen und Konstanten	Unterscheidung der Begriffe anhand geeigneter Problemstellungen
Einfache Datentypen mit ihren wesentlichen Operationen und Relationen • Datentyp • Ganzzahl, Gleitkommazahl • Wahrheitswert, Zeichen	Name, Wertebereich, Operationen Ganzzahl- und Gleitkommazahloperationen Logische Verknüpfungen, Ordnungsrelation
Aufbau eines Programms • Grundstruktur eines Programms • reservierte Wörter • Kommentare	Die Einführung in die Programmiersprache soll auf die unbedingt notwendigen Sprachkonstrukte beschränkt werden. Eine vollständige Darstellung der Syntax der Sprache wird nicht angestrebt.
Einfache Anweisungen • Ausdruck • Wertzuweisung • Ein-, Ausgabeanweisung	
Steuerung des Kontrollflusses • Sequenz • Auswahl • Iteration	Darstellung der Sequenz, einseitiger und zweiseitiger Auswahl und der Iterationen in der verwendeten Programmiersprache
Strukturierte Datentypen • Feld • Zeichenkette • Zugriff auf die Komponenten	Ausgehend von geeigneten Problemstellungen können die Vorteile der Verwendung strukturierter Datentypen begründet werden.

Informatik, Einführungsphase	
Einführung in die Programmentwicklung	**20 Stunden**

VERBINDLICHE INHALTE	Vorschläge und Hinweise
Programmentwicklung, Test und Dokumentation **Entwurf** • Algorithmenentwicklung, • Prüfung anhand von Werteverlaufstabellen, • Implementierung in der Programmiersprache, • Testläufe mit ausgewählten Eingabedaten, • Fehlersuche **Programmdokumentation** • Verbesserung der Lesbarkeit des Quelltextes durch Gestaltung des Quelltextes Erläuterung durch Kommentare Auswahl aussagekräftiger Bezeichner **Eigenschaften** eines Algorithmus • Endlichkeit der Beschreibung, Eindeutigkeit und Ausführbarkeit der Abfolge, Allgemeinheit	Für umfangreichere Arbeitsaufträge an Schüler(gruppen) bieten sich an: Verfahren der Klassischen Kryptographie è Klassische kryptographische Verfahren Modellierung von Abläufen mit Hilfe von Automaten (z.B. Ampelsteuerung, Verkaufsautomat, Paritätsbitprüfer) è Fakultatives Thema

Informatik, Einführungsphase	
Klassische kryptographische Verfahren	**5 Stunden**

VERBINDLICHE INHALTE	Vorschläge und Hinweise
Grundbegriffe der Chiffrierung	Motivation: Notwendigkeit der Verwendung von Kryptosystemen bei der Datenübertragung in Rechnernetzen Klärung der grundlegenden Begriffe Klartext, Geheimtext, Schlüssel, Chiffrieralgorithmus Beispiele: Historische Transpositions- und (monoalphabetische) Substitutionsverfahren
Sicherheit	Prinzip von Kerckhoffs One-Time-Pad
Angriffsarten	Passive und aktive Angriffe: Brute-force, Ciphertext-only, Known-plaintext, Chosen-plaintext
	Literatur/Medien Albrecht Beutelspacher: Kryptologie, Vieweg 1993 Beutelspacher, Schwenk, Wolfenstetter: Moderne Verfahren der Kryptographie, Vieweg 2001 Schmeh: Kryptografie, dpunkt.verlag; 2001 Simon Singh: Geheime Botschaften, Carl Hanser, 2000

Informatik, Einführungsphase	
FAKULTATIVE INHALTE	**Vorschläge und Hinweise**
Vorschlag 1 Vertiefung des Themas Klassische Kryptographie	Homophone, polyalphabetische Verschlüsselungsverfahren Einfache kryptoanalytische Methoden
Vorschlag 2 Modellieren mit Automaten **Endlicher (deterministischer) Automat** **Eingabealphabet, Zustände, Anfangszustand, Endzustände, Übergangsgraph**	Modellieren von Abläufen durch endliche Automaten mit und ohne Ausgabe Beispiele: Ampelsteuerung, Verkaufsautomat, Paritätsbitprüfer è Modellieren und Entwerfen
	Literatur Albrecht Beutelspacher: Kryptologie, 5. Auflage, Vieweg 1996 Gasper, Leiß, Spengler, Stimm: Technische und theoretische Informatik, BSV-Verlag 1992