

Schulinternes Curriculum für den Grundkurs Informatik

- **Stand: 4/2024**
- **Grundlage: Rahmenlehrplan für die gymnasiale Oberstufe, Teil C, Informatik, 2006/2007**

Im Grundkursfach werden die Schülerinnen und Schüler in grundlegende Sachverhalte, Probleme und Zusammenhänge des Faches eingeführt. Das Grundkursfach zielt auf die wesentlichen Arbeitsmethoden der Informatik und die exemplarische Erkenntnis fachübergreifender Zusammenhänge ab.

Zeitliche Grundlage des Plans ist ein wöchentlicher Unterricht von 3 Unterrichtsstunden mit je 45 Minuten. Die Themen erweitern die Kompetenzen aus dem Wahlpflichtkurs der 10. Klasse.

Kurs-halb-jahr	Thema	Inhalte	Bezug zum Rahmenlehrplan Kompetenzbezug	Zeitleiste	zu verwendende Software/ Tools
Q1/Q2	Datenbanken und Softwareentwicklung	<ul style="list-style-type: none"> • Objektorientierte Modellierung und Programmierung <ul style="list-style-type: none"> – Klasse, Objekt – Attribut, Methode, Konstruktor – Geheimnisprinzip (Datenkapselung) – Dokumentation – UML-Klassendiagramm – Vererbung – Polymorphie – Fakultativ: <ul style="list-style-type: none"> ▪ Abstrakte Klassen ▪ Interfaces ▪ MVC ▪ GUI-Programmierung ▪ Exceptions 	4.3 Softwareentwicklung [RLP S. 21] Informatisches Modellieren Problemlösen	45h	JAVA Java-Editor
Q1/Q2	Datenbanken und Softwareentwicklung	<ul style="list-style-type: none"> • Datenmodellierung <ul style="list-style-type: none"> – Datenbanksystem, Datenbankmanagementsystem – Tabellen, Beziehungen zwischen Tabellen, Redundanzen, Anomalien, Inkonsistenzen, Primärschlüssel, Fremdschlüssel 	4.1 Datenbanken Informatisches Modellieren Problemlösen	21h	

	<ul style="list-style-type: none"> – ER-Modell (Entitätsmenge, Relationenmenge, Kardinalitäten), Chen-Notation – Datenbankentwurf – Normalformen – Integritätsbedingungen 			
	<ul style="list-style-type: none"> • Relationales Datenbankschema <ul style="list-style-type: none"> – ER-Modell <-> Relationenmodell (Überführungsregeln) 		3h	
	<ul style="list-style-type: none"> • Abfragen (Operationen auf Tabellen): <ul style="list-style-type: none"> – Relationenalgebra – SQL: Selektion, Projektion, Join – SQL: Gruppierung, Aggregation, Sortierung, Logische Operationen, geschachtelte Select-Ausdrücke – SQL: Create, Update, Delete, Alter 		18h	Videocenter-DB Fitnesscenter-DB
	<ul style="list-style-type: none"> • Praktische Umsetzung in ein Datenbank-Management-system <ul style="list-style-type: none"> – Mini-Projekt (DB-Entwurf, Schnittstelle JAVA), z.B. Buchhandlung 		15h	USB-Webserver (Apache, MySQL, PHP und phpMyAdmin) Java-Editor
	<ul style="list-style-type: none"> • Algorithmen und Datenstrukturen <ul style="list-style-type: none"> – Abstrakte Datentypen – Datenfeld (Array) – Liste (list) – Fakultativ: <ul style="list-style-type: none"> ▪ Stapel (Stack) ▪ Schlange (Queue) ▪ Suchalgorithmen: linear, binär • Sortieralgorithmen: Quicksort, Bubblesort, Selectionsort 		15h	JAVA Java-Editor
	<ul style="list-style-type: none"> • Datenschutz und Datensicherheit <ul style="list-style-type: none"> – Personenbezogene Daten, Recht auf informationelle Selbstbestimmung 		6h	

Kurs- halb- jahr	Thema	Inhalte	Bezug zum Rahmenlehrplan Kompetenzbezug	Zeitleiste	zu verwendende Software/ Tools
Q3	Formale Sprachen und Automaten		4.4 Sprachen und Automaten [RLP S.22]		
		<ul style="list-style-type: none"> • Grundlagen formaler Sprachen und deren Vergleich mit natürlichen Sprachen <ul style="list-style-type: none"> – Formale Sprache L über einem Alphabet Σ als Teilmenge der Kleeneschen Hülle des Σ^* – Operationen auf formalen Sprachen (Vereinigung, Schnitt, Differenz, Konkatenation, Potenz) – Vergleich formaler und natürlicher Sprachen 	Informatiksysteme verstehen	6h	
		<ul style="list-style-type: none"> • endliche Automaten (Akzeptor) <ul style="list-style-type: none"> – DEA und NEA, formal als 5-Tupel (Übergangsfunktion als Übergangstabelle) – Bezug zu regulären Sprachen • zustandsorientierte Modellierung <ul style="list-style-type: none"> – DEA/NEA als Graph • fakultativ: <ul style="list-style-type: none"> – Potenzmengenkonstruktion – Minimierung eines DEAs – Kellerautomat 	Informatiksysteme verstehen Informatisches Modellieren Problemlösen	15h	JFlap
<ul style="list-style-type: none"> • einfache Grammatiken (Typ-3) <ul style="list-style-type: none"> – formal als 4-Tupel – Bezug zu regulären Sprachen und DEA/NEA – Überführung von DEA/NEA zu reg. Grammatik und umgekehrt • Syntax und Semantik (Syntaxdiagramme) <ul style="list-style-type: none"> – Definition von Syntax und Semantik – Syntaxdiagramme • fakultativ: 	Informatiksysteme verstehen Mit Information umgehen	12h	JFlap		

		<ul style="list-style-type: none"> – Grammatiken und deren Einteilung in Typen (Chomsky-Hierarchie) – reguläre Ausdrücke 			
	Deklarative Programmierung		4.3 Softwareentwicklung [RLP S. 21]		
		<ul style="list-style-type: none"> • Funktionale Programmierung mit Haskell <ul style="list-style-type: none"> – Vergleich von imperativer und deklarativer Programmierung – Funktionen, Typdefinitionen – Logische und arithmetische vordefinierte Operationen – Pattern Matching, if-then, Guards, lokale Funktionsdefinition – Tupel, Listen, Algorithmen auf Listen • fakultativ: <ul style="list-style-type: none"> – Entwurf und Umsetzung eines DEAs in Haskell 	Problemlösen Informatisches Modellieren Informatiksysteme verstehen Mit Information umgehen	10h	WinHugs
Q4	Softwareprojekt		4.1 Datenbanken [RLP S. 19] 4.3 Softwareentwicklung [RLP S. 21] 4.5 Informatik, Mensch und Gesellschaft [RPL S. 23]		
		<ul style="list-style-type: none"> • Neues oder fortführendes Projekt unter Einbeziehung von Inhalten verschiedener Themenfelder <ul style="list-style-type: none"> – Softwareentwicklung in Java unter Einbindung einer relationalen Datenbank – Thema des Projekts soll aus dem Erfahrungskreis der SuS kommen – Das Projekt ist so zu wählen, dass es von den SuS in der geforderten Zeit fertig gestellt werden kann – Das Projekt erstreckt sich mindestens über alle Phasen des Software-Life-Cycle • Grundlagen systematischer Softwareentwicklung (Software-Life-Cycle) 	Kommunizieren und Kooperieren Wechselwirkungen zwischen Informatiksystemen, Mensch und Gesellschaft beurteilen Problemlösen Informatiksysteme verstehen Informatisches Modellieren Mit Informationen umgehen	30h	JAVA (JavaEditor, BlueJ) USB-Webserver (Apache, MySQL, PHPmyAdmin)

		<ul style="list-style-type: none">– Wasserfallmodell mit verschiedenen Phasen: Problem- analyse, Entwurf, Implementationsphase, Tes- ten/Wartungsphase, Betriebsphase– Software-Ergonomie (Problemanalyse)– UML-Klassendiagramm; ER-Modell (Entwurfsphase)– Implementation, Anlegen von Relationen (Imple- mentationsphase, Testen/Wartungsphase)			
--	--	--	--	--	--