



Sek. I und Sek. II

Das Fach Informatik wird am Ernst-Mach-Gymnasium im Rahmen der Differenzierung in Form eines 3-stündigen Kurses angeboten. Die Auswahl der Unterrichtsinhalte, Methoden und die Leistungsbewertung im Fach Informatik für die Sekundarstufe I orientiert sich unter anderem an den von der Gesellschaft für Informatik (GI) herausgegebenen Bildungsstandards und den Richtlinien und Lehrplänen des Landes NRW.

Die Auswahl der inhaltlichen Kompetenzen legt ihren Schwerpunkt auf folgende Säulen:

- Reflektion über und Analyse von bereits fertigen Produkten und Programmen: Medienkompetenz
- Einblick in den Computer und seine Arbeitsweise → Technische Informatik
- Anwendung von Software und Handhabung des PCs als Arbeitswerkzeug
- Gestaltung und Entwicklung eigener Produkte und Programme

Weitere grundlegende Kompetenzen, die das Fach Informatik vermitteln möchte, sind sprachliche Ausdrucksfähigkeit, Umgang mit mathematischen Systemen, Verfahren und Modellen. Ebenso sollen erste Einblicke in das Denken und Arbeiten mit komplexen Systemen gegeben werden.

Die Reihenfolge der Themen stellt nur eine mögliche Verteilung über die einzelnen Schulhalbjahre dar und die Themenbereiche können in ihrer Reihenfolge zeitlich variiert werden.

Je nach Interessenlage der Schüler und vorhandener Zeit kann hier (9.2) ein weiteres Thema gewählt werden. Dieses Thema kann eine weitere Programmiersprache z.B. PHP oder Python, aber auch praktischer orientierte Themen wie Bild- oder Videobearbeitung .

Der Informatik-Unterricht in der Sekundarstufe II folgt dem objektorientierten Ansatz (vgl. Sequenz „objektorientiert allgemein“, Richtlinien und Lehrpläne Sekundarstufe II) in Form von 3-stündigen Grundkursen. Die Auswahl der Schwerpunkte orientiert sich an den Vorgaben des Zentralabiturs NRW.

Programmiersprache / Entwicklungswerkzeuge in der Sekundarstufe II

Als objektorientierte Programmiersprache wird Java eingesetzt. In der Jahrgangsstufe in der Einführungsphase wird die Entwicklungsumgebung BlueJ verwendet. Spätestens in der Qualifikationsphase I erfolgt der Umstieg auf eine fortgeschrittene Entwicklungsumgebung (Netbeans).

Die genauen Inhalte der einzelnen Schulhalbjahre begründet und verstärkt durch die Beschreibung von Zielen, Kompetenzen und Methoden lauten:



Jahrgang:	Inhalt	Ziele, Kompetenzen
Stufe 8.1	Hard- und Software <ul style="list-style-type: none">• Historische Entwicklung der Computertechnologie• Betriebssysteme und ihre Funktionsweise	<ul style="list-style-type: none">• Kennenlernen der Hardwarekomponenten• Kennenlernen der Architektur eines Betriebssystems• Die Struktur von Daten verstehen• Umgang mit Standardsoftware
	Technische Informatik <ul style="list-style-type: none">• Aufbau eines Computers• Das von-Neumann-Prinzip• Binäre Zahlencodierung• die Darstellung der Zahlen im Computer• Bit und Byte• einfache Schaltnetze (MOPS)• Halb- und Volladdierer, Paralleladdierer (MOPS)• Ausblick in die Aussagenlogik	<ul style="list-style-type: none">• Erfassen der Wirklichkeit durch die Darstellung und die Veranschaulichung des Prozessors• Die Arbeitsweise eines PCs verstehen und begreifen• Anwenden von Simulationssoftware• Modellieren der Wirklichkeit, abstrakte Zusammenhänge verstehen und darstellen können
Stufe 8.2	Textverarbeitung <ul style="list-style-type: none">• Analyse von Texten hinsichtlich ihrer Formatierung• Umsetzung von verschiedenen Gestaltungsmöglichkeiten mit Hilfe eines Textverarbeitungsprogramms• Formatieren• Tabellen und Objekte einbinden• Aufbau von Briefen• Serienbriefe• Erstellung von Formularen	<ul style="list-style-type: none">• Bewertung von und Reflektion über Textformatierungen• Anwenden von Software• Anwenden der eigens erstellten „Regeln“ zur Formatierung• Argumentieren und Modellieren



Stufe 8.2	Tabellenkalkulation <ul style="list-style-type: none">• Rechnen mit Rechenblättern• Arbeiten mit Formeln (relativen und absoluten Bezüge)• Datentypen• Zuordnungen• grafische Darstellung von Funktionen• Funktionen mit mehreren Parametern• Verkettung von Funktionen• Bedingte Terme• Logische Funktionen• S-Verweis	<ul style="list-style-type: none">• Anwenden von Software• Probleme erfassen, erkunden und lösen• Mit Daten arbeiten und deren Beziehungen und Veränderungen beschreiben und mit Hilfe des PCs Ergebnisse bestimmen• mathematische Probleme mit dem Computer als• Werkzeug lösen• Abstrakte Zusammenhänge erkennen,• modellieren und mit Hilfe des PCs darstellen
Stufe 9.1	Internet, HTML und CSS <p>Analyse und Bewertung der HP-Gestaltung von unterschiedlichen Internetseiten unter Gesichtspunkten:</p> <ul style="list-style-type: none">• Übersichtlichkeit• Zielgruppe · kommerzielles Bestreben etc.• Internetseiten gestalten (Editor)• Formatierungssprache HTML	<ul style="list-style-type: none">• Argumentieren und Bewerten• Anwenden der gewonnenen technischen und medialen Erkenntnisse durch Gestalten eigener Webseiten
	Datenbanken <ul style="list-style-type: none">• Benutzung eines einfachen relationalen Datenbanksystems aus der Lebenswelt der Schüler• Daten suchen, sortieren, eintragen, ändern, löschen in einem System mit grafischer Benutzeroberfläche	Informatisches Modellieren <p>Hinterfragen und bewerten Ergebnisse einer Modellbildung kritisch.</p> Infomatiksysteme verstehen <p>Beschreiben Grundlagen des Aufbaus und der Arbeitsweise eines Informatiksystems.</p>



Stufe 9.2	Einführung in die Software-Entwicklung mit einer einfachen imperativen Programmiersprache (z.B. FMS-Logo, Python usw.) Basiskonzepte der imperativen Programmierung <ul style="list-style-type: none">• Prozeduren · Algorithmen mit Verzweigung• Algorithmen mit Wiederholungen• Variablen• Einführung in die algorithmische Denkweise	<ul style="list-style-type: none">• Softwareentwicklung• Vom Problem zum Algorithmus• Algorithmen und Problemlösestrategien entwickeln und anwenden• Modellieren und programmieren
	Thema nach Wahl z.B. <ul style="list-style-type: none">• Bildbearbeitung mit Gimp,• Modellbildungssoftware,• Logische Schaltungen, etc.	Je nach Thema unterschiedlich <ul style="list-style-type: none">• Weitere Werkzeuge der Informatik kennenlernen und anwenden können• Einblick in vernetzte Zusammenhänge• Präsentation der Ergebnisse
	Datenschutz	Medienkompetenz, <ul style="list-style-type: none">• Welche Informationen darf ich weitergeben und welche nicht.• Wie werden persönliche Daten verarbeitet.• Welche Rechte habe ich an meinen Daten.• Wie sollten und wie können Daten geschützt werden!



Stufe EF		
	<p>Konzepte des objektorientierten Modellierens und Programmierens</p> <ul style="list-style-type: none">▪ Elementare Datentypen▪ Kontrollstrukturen (Bedingte Anweisung, Schleifen)▪ Klasse, Objekt, Attribut, Methode, Geheimnisprinzip▪ Klassendiagramme▪ Beziehungen zwischen Klassen: (gerichtete) Assoziation mit Multiplizität, Vererbung▪ Abstrakte Klassen, Polymorphie▪ Elementare Such- und Sortieralgorithmen für Arrays<ul style="list-style-type: none">– Lineare Suche / ggf. Binäre Suche– Selection-Sort und Insertion-Sort▪ Standard-Klassen der Java-API (insbesondere Klasse String)▪ Model-View-Controller-Prinzip	<p>Nutzung der Prinzipien der Objektorientierung zur Modellierung (Abstraktion und Systematisierung) einer Anwendungssituation</p> <p>Einsatz der Modellierungssprache UML</p> <p>Implementieren und Testen Dokumentieren der erstellten Software</p> <p>Analyse und kritische Betrachtung der eigenen Lösung</p> <p>Analyse und Vergleich verschiedener Algorithmen hinsichtlich ihrer Komplexität (Effizienz)</p> <p>Werkzeugeinsatz BlueJ (Entwicklungswerkzeug) Javadoc (Dokumentation)</p>
	<p>Aufbau und Funktionsweise eines Computers</p> <ul style="list-style-type: none">▪ Von-Neumann-Rechner▪ Informationsrepräsentation, Codierung	



Stufe Q1		
	Rekursive Programmierstrategien <ul style="list-style-type: none">▪ Beispiele für rekursive Algorithmen: Fakultät, Fibonacci-Zahlen<ul style="list-style-type: none">– Dynamic-Programming-Paradigma▪ Binäre Suche<ul style="list-style-type: none">– Aufwandsbetrachtungen / Vergleich mit linearer Suche	Einführung in rekursive Programmierstrategien – Vertiefung erfolgt im Bereich der dynamischen Datenstrukturen (Binärbaum)
	Dynamische Datenstrukturen <ul style="list-style-type: none">▪ Stack und Queue:<ul style="list-style-type: none">– Implementierung und Anwendung der Standardoperationen▪ Liste:<ul style="list-style-type: none">– Anwendung der Standardoperationen– Implementierung der Standardoperationen (nur LK)▪ Binärbaum:<ul style="list-style-type: none">– Anwendung der Standardoperationen– Traversierungsalgorithmen▪ Binärer Suchbaum:<ul style="list-style-type: none">– Anwendung der Standardoperationen– Implementierung von insert() und search() (nur LK)▪ Ungerichteter gewichteter Graph (nur LK):<ul style="list-style-type: none">– Anwendung der Standardoperationen– Breiten- und Tiefensuche▪ Anwendungen: z.B. Warteschlangensimulation, Huffman-Codec	Visualisieren der Datenstrukturen und der Funktionsweise der Standardoperationen mithilfe geeigneter Diagramme Verwendung der Klassen aus den Vorgaben des Zentralabiturs Weiterführende Prinzipien des objektorientierten Software-Entwurfs: Einsatz von Design-Patterns wie „Iterator“, „Template Method“, „Factory Method“



Stufe Q1		
	<p>Rechnernetze</p> <ul style="list-style-type: none">▪ Protokolle, Schichtenmodelle, Grundlagen des Internet▪ Client-Server-Modell▪ Zentralabitur-Klassen: Connection, Client, Server▪ Kryptographie:<ul style="list-style-type: none">– Symmetrische Verschlüsselungsverfahren (Caesar, Vigenère)– Asymmetrische Verschlüsselungsverfahren (RSA)– Verfahren zum Schlüsselaustausch (Diffie-Hellman)▪ Anwendungen: z.B. Chat-Software, netzwerkfähiges Spiel	<p>Visualisieren der Kommunikation zwischen Client und Server (z.B. Einsatz von Time-Sequence-Diagrammen)</p> <p>Typische Einsatzbereiche, Möglichkeiten, Grenzen, Chancen und Risiken der Informations- und Kommunikationssysteme untersuchen und einschätzen</p> <p>Behandlung von Datenschutzaspekten</p>
	<p>Endliche Automaten und formale Sprachen</p> <ul style="list-style-type: none">▪ Modellieren kontextbezogener Problemstellungen als deterministische endliche Automaten▪ Darstellung von deterministischen endlichen Automaten als Graph und als Tabelle▪ Formale Sprachen: Reguläre Sprachen und ihre Grammatiken	<p>Algorithmen, Sprachkonzepte und Automatenmodelle beurteilen</p>



Stufe Q2		
	Projektarbeit <ul style="list-style-type: none">▪ Vertiefung der in den Jahrgangsstufen 11 und 12 erarbeiteten Grundlagen▪ Themenwahl der Projektarbeit orientiert sich an den Interessen der Schülerinnen und Schüler	Projektorientierter Unterricht Grundsätzliche Prinzipien der Projektarbeit Planung von Projekten Arbeiten im Team Dokumentation der Projektarbeit unter Verwendung von Projektplanungssoftware (z.B. „Ganttproject“)
	Datenbanken <ul style="list-style-type: none">▪ Modellieren kontextbezogener Problemstellungen mit Entity-Relationship-Modell▪ Normalisierung: Überführung in die erste, zweite und dritte Normalform▪ Relationenalgebra (Selektion, Projektion, Vereinigung, Differenz, kartesisches Produkt, Umbenennung, Join)▪ SQL-Abfragen über mehrere verknüpfte Tabellen	Problemstellungen eingrenzen und Probleme strukturieren Allgemeine Strategien und Standardlösungen kennen lernen und verstehen