

## Ergänzungsfach

# Informatik

### 1. Stundendotation

1. Klasse	2. Klasse	3. Klasse	4. Klasse
-	-	-	4

### 2. Bedeutung des Faches

Die Informatik durchdringt zunehmend alle Bereiche des Lebens. Sie betrifft in der Anwendung alle wissenschaftlichen Fachrichtungen. Das Ergänzungsfach vermittelt die Kompetenz, Wesen und Stellenwert der Informatik zu erkennen und einzuordnen, sowie die Einsatzmöglichkeiten der Informatik zu beurteilen. Informatik verbindet mathematisches, naturwissenschaftliches und ingenieurwissenschaftliches Denken in *einem* Fach. Für die Lernenden stehen team- und projektorientiertes Arbeiten, das konstruktive Auffinden unterschiedlicher Lösungen sowie deren Vergleich und kritische Beurteilung im Vordergrund.

Das Ergänzungsfach Informatik befähigt die Lernenden zur Analyse und Modellierung von Problemstellungen sowie zum Entwurf von algorithmischen Lösungen. Deren Realisierung durch selbst geschriebene Programme ermöglicht eine direkte Überprüfung der Lösungsqualität. Die Lernenden erfahren, welche Lösungen technisch machbar sind, sinnvoll eingesetzt werden können und welche Ressourcen dazu nötig sind. Das Ergänzungsfach Informatik soll Grundlagen vermitteln in den Bereichen Algorithmik, Programmieren, theoretische Informatik sowie Information und Kommunikation. In einem

oder mehreren dieser Bereiche findet eine Vertiefung statt, die sich besonders für ein projektorientiertes und vernetztes Vorgehen eignet.

### 3. Richtziele

#### Grundkenntnisse

- Grundbegriffe und Grundkonzepte zur Problemmodellierung, Problemanalyse und Entwurfsmethodik von Informatiklösungen verstehen
- Verfahren zur Bewertung und Überprüfung der Korrektheit von Lösungen kennen
- Grundlagen einer Programmiersprache kennen
- verschiedene Darstellungen von Informationen kennen
- Grundlagen der digitalen Kommunikation verstehen
- die Grenzen der Berechenbarkeit kennen

#### Grundfertigkeiten

- Probleme aus verschiedenen Bereichen analysieren und strukturieren
- Algorithmen entwerfen, beurteilen und in einer Programmiersprache umsetzen
- der Problemstellung angepasste Datenmodelle entwerfen
- Informatiklösungen bezüglich Korrektheit, Effizienz und Benutzerfreundlichkeit beurteilen und dokumentieren

### **Grundhaltungen**

- Informatiklösungen kritisch beurteilen und hinterfragen
- zu Team- und Projektarbeit und interdisziplinärem Austausch bereit sein
- strukturiert planen und handeln
- bei der Suche nach Informatiklösungen und deren Umsetzung Ausdauer zeigen
- sich mit den Auswirkungen der Informatik im Alltag auseinandersetzen

## **4. Lerninhalte**

Zusätzlich empfohlener Lerninhalt, welcher je nach verfügbarer Zeit, Interesse oder schulinternen Gegebenheiten vermittelt werden kann, ist mit (+) bezeichnet.

### **Themenkreise**

#### **Programmierung I**

Einführung in die strukturierte Programmierung im Kleinen. Umsetzung in einer Programmiersprache. Grundbegriffe und Konzepte:

- Programm, Sequenz, Anweisung (Rechnerbefehle)
- Wiederholungsanweisung (Iteration, Schleife)
- Auswahlanweisung (Selektion, bedingte Verzweigung)
- Abstraktion: Unterprogramm, Prozedur, Funktion, Parameter und Variable
- Rekursion

#### **Theoretische Informatik**

Grundlagen der Informatik und ihre fundamentalen Konzepte:

- Bildung der Grundbegriffe: Algorithmus und Programm (Turing, Church),
- Modell des Computers (von Neumann)
- Formale Sprachen
- Berechenbarkeit und Komplexität

#### **Datenstrukturen und Algorithmen**

Einführung in die grundlegenden Datenstrukturen zur effizienten Implementierung von Algorithmen.

- Datentypen und ihre Darstellung im Computer, Arten der Datenspeicherung.
- Fundamentale Datenstrukturen (Felder, Mengen, Listen, Bäume, Graphen)
- Wahl geeigneter Datenstrukturen zur Implementierung von Algorithmen für einfache Aufgaben der Datenverarbeitung im Hinblick auf die Effizienz des resultierenden Programms.
- Grundlegende Algorithmen wie zum Beispiel Such und Sortieralgorithmen.

#### **Programmierung II**

- Einführung in objektorientiertes Programmieren (Klassen, Methoden und Eigenschaften, Vererbung)

#### **Programmierung III (+)**

- Einführung in funktionales Programmieren
- Programmierung von mathematischen Methoden (z.B. Lösen von linearen Gleichungssystemen)
- Programmieren von Robotern

#### **Informationssysteme und Netzwerke (+)**

- Datenstrukturen zur dynamischen Datenverwaltung

- Relationale Datenbanken
- Suchalgorithmen im Internet
- Übertragungsprotokolle
- Darstellung und Verarbeitung von Daten in Netzwerken (z.B. HTML, XML, PHP, JAVA, JAVASCRIPT)

#### **Algorithmik (+)**

- Entwurf von Algorithmen und Analyse der Effizienz
- Geometrische Algorithmen
- Zufallsgesteuerte Algorithmen
- Simulationen

#### **Aufbau und Funktionsweise eines Computers (+)**

- Organisation des Rechners (Prozessor, Speicher, Busse, Eingabe, Ausgabe, interne Darstellung von Daten)
- Rechnerarithmetik
- Struktur eines Prozessors
- Darstellungen von booleschen Funktionen
- Entwurf von elektronischen Schaltkreisen (Multiplexer, Halbaddierer, Addierer)

#### **Kryptologie (+)**

Einführung in die sichere Kommunikation und den Aufbau von Kryptosystemen.

- Vermittlung von Grundbegriffen: Sender, Empfänger, Klartext, Kryptotext, Verschlüsselung, Entschlüsselung, Schlüssel.
- Kurze Geschichte der Geheimschriften (Caesar, Richelieu) und Kerkhoffsches Prinzip.
- Symmetrische Kryptosysteme, ihre Stärken und Schwächen, Attacken gegen monoalphabetische Systeme, One-Time-Pad.
- Der Begriff eines sicheren Kryptosystems anhand der Algorithmik und des Konzeptes der Berechnungskomplexität.
- Schlüsselaustausch über unsichere Kanäle (Diffie-Hellman).
- Public-Key-Kryptographie und der Begriff der Einweg-Funktionen. Digitale Unterschriften und andere Anwendungen im E-Commerce.

## **5. Pädagogisch-didaktische Hinweise**

Neben den herkömmlichen Unterrichtsformen soll der Teamarbeit bei der Realisierung von kleineren Projekten genügend Zeit eingeräumt werden. Die einzelnen Themenkreise bieten Gelegenheit für die Einbettung der Lerninhalte in den gesellschaftlichen und wissenschaftshistorischen Kontext.

## **6. Bemerkungen zum fachübergreifenden Unterricht**

Sprachen	Syntax und Semantik
Mathematik	Diskrete Mathematik, Numerik, Beweise, Kryptographie
Naturwissenschaften	Modellierung, Simulationen, Mess- und Regeltechnik
Bildnerisches Gestalten	Digitale Bildverarbeitung, Computerkunst, Raytracing

Wirtschaft und Recht

Finanzmodelle, Datenanalysen, Datenschutz und Sicherheit, Urheberrechte