

Informatik

1. Allgemeines

Stundendotation	3. Klasse	4. Klasse	5. Klasse	6. Klasse
Grundlagenfach	2/2	2/2		
Schwerpunktfach				
Ergänzungsfach			3/3	3/3

2. Leitideen und Richtziele

Die Informatik durchdringt zunehmend alle Bereiche des Lebens und betrifft alle wissenschaftlichen Fachrichtungen. Das Fach Informatik leistet einen Beitrag zur Vermittlung und Stärkung von Grundkenntnissen und Grundfähigkeiten, welche für die wachsenden Anforderungen im Umfeld der Informationstechnologien und der Digitalisierung unentbehrlich sind.

Die Informatik versteht sich als eigenständige Wissenschaft nicht auf die Vermittlung von Kenntnissen von Anwendersoftware, sondern als systematisch aufgebautes Schulfach, welches das tiefere Verständnis von automatisierten Prozessen beinhaltet.

Der Lehrplan ist altersgerecht aus den drei Hauptpfeilern **Datendarstellung**, **Automatisierung** und **Algorithmik** sowie Informationstechnologie als Spiralcurriculum aufgebaut. Er beinhaltet die grundlegenden Konzepte der Informatik und die zu erreichenden Hauptziele:

- Die von den Menschen erschaffene Welt zu verstehen, zu steuern und mitgestalten zu können
- Die Grundkompetenzen in der Mathematik und in der Sprache zu fördern
- Die konstruktive Denkweise der technischen Disziplin zu fördern

Das Fach leistet eine Brückenfunktion und baut im Sinne eines Spiralcurriculums auf die Kenntnisse der Volksschule auf und bietet die Grundlage für das Ergänzungsfach Informatik. Es hat einen hohen interdisziplinären Charakter und soll sowohl mathematisch-naturwissenschaftliches wie auch ökonomisches und gesellschaftswissenschaftliches Denken gleichermaßen fördern. Deshalb soll der Austausch und die Zusammenarbeit über die Fachgrenzen hinaus angestrebt und gepflegt werden. Die Lernenden werden zur Analyse und Modellierung von Problemstellungen sowie zum Entwurf von algorithmischen Lösungen befähigt. Die Realisierung in einer Programmiersprache ermöglicht eine direkte Überprüfung der Lösungsqualität und dient als Entscheidungsgrundlage, welche Lösungen technisch machbar sind, sinnvoll eingesetzt werden können und welche Ressourcen dazu notwendig sind. Das kritische Hinterfragen von Lösungen und des eigenen Arbeitsprozesses soll den Lernenden ermöglichen, sich eigene Arbeits- und Denkprozesse anzueignen oder bestehende zu verbessern.

Grundkenntnisse

Die Lernenden kennen

- die Grundzüge der historischen Entwicklung der Informatik und die Motivation zur Entwicklung grundlegender Informatikkonzepte.
- wichtige Grundkonzepte und Begriffe der Informatik wie Algorithmus, Programm, Graph, Modell, Verifikation, Berechnungskomplexität, Sicherheit.
- Methoden zur digitalen Darstellung und Codierung von Informationen.
- die Bezüge und Unterschiede zwischen Zeichen, Daten und Informationen.
- grundlegende Programmierkonzepte, die sie zur Steuerung des Computers mit einer Programmiersprache einsetzen können.
- die Bedeutung des Datenschutzes und der Sicherheit in der vernetzten Welt sowie die Grundkonzepte der Kryptografie.
- den modularen Aufbau von Programmen aus Komponenten und Schnittstellen.
- die wichtigsten technischen Grundkonzepte von Computernetzwerken.

Grundfertigkeiten

Die Lernenden können

- Algorithmen in einer Programmiersprache implementieren und testen.
- eigene und fremde Lösungswege formal beschreiben und kritisch analysieren.
- Ursachen von Problemen und Fehlern systematisch und zielgerichtet eruieren.
- Informatiklösungen bezüglich Korrektheit, Effizienz und Sicherheit beurteilen.
- in einer Gruppenarbeit die Planung, die Analyse, die Implementierung und das Erproben eines Informatikprojekts durchführen und das Vorgehen transparent und genau dokumentieren.
- elementare Methoden der Verschlüsselung verstehen, implementieren, analysieren und ihren Sicherheitsgrad bewerten.
- die Bedeutung der Informatik im wissenschaftlichen, gesellschaftlichen und historischen Kontext erkennen und einordnen.

Grundhaltungen

Die Lernenden

- sind sich bewusst, dass die algorithmische Denkweise die Grundlage für die erfolgreiche Automatisierung von Prozessen aller Art ist.
- systematisches und modulares Vorgehen massgebend für den Entwurf von Softwaresystemen und Algorithmen ist.
- zeigen Ausdauer, Sorgfalt und Kreativität bei der Erarbeitung von Lösungen und planen strukturiert.
- sind sich bewusst, dass Datenschutz und sichere Kommunikation unbedingt angestrebt werden müssen.
- sind bereit, sich mit den Schwierigkeiten und Anforderungen angewandter Probleme auseinanderzusetzen und die Lösungen schrittweise zu verbessern.
- möchten sich aktiv in Team- und Projektarbeiten einbringen.
- sind bereit, die Ergebnisse der eigenen Arbeit kritisch zu beurteilen und die Möglichkeiten der Weiterentwicklung und Verbesserung zu ergreifen.
- entwickeln eine persönliche Einstellung zu den Problemen der Informatik, auch aus Sicht ethischer Grundformen.
- sind offen und zeigen Interesse für Neuerungen in der Informatik.

3. Grobziele, Lerninhalte, Querverweise

Vorbemerkung zum detaillierten Lehrplan:

Das Fach hat einen hohen interdisziplinären Charakter. Deshalb soll der Austausch und die Zusammenarbeit über die Fachgrenzen hinaus angestrebt und gepflegt werden.

Grundlagenfach: 3. Klasse		
Grobziele	Lerninhalte	Querverweise
<ul style="list-style-type: none"> ■ Aufbau und Funktionsweise eines Rechners 	<ul style="list-style-type: none"> • Computerarchitektur und Funktionsweise der wichtigsten Komponenten beschreiben (EVA-Prinzip) • Speicher, Prozessor, Register, Schnittstelle, In-/Output, Betriebssystem • Zusammenspiel von Hard- und Software erklären 	<ul style="list-style-type: none"> → Physik → NPU
<ul style="list-style-type: none"> ■ Datenschutz und Datensicherheit 2 	<ul style="list-style-type: none"> • Polyalphab. Kryptosysteme (Vigenère, One-Time-Pad) • Chancen und Risiken der Informatik • Computermalware und Schutzmassnahmen (Viren, Trojaner, Würmer) 	<ul style="list-style-type: none"> → Mathematik → Geschichte → Wirtschaft und Recht → Alle Fächer → Physik
<ul style="list-style-type: none"> ■ Graphen 	<ul style="list-style-type: none"> • auflisten von Lösungen mit Baumdiagrammen • Knoten, Kanten • Weg, Kreis • Netze als Graphen • binäre Darstellung • Hamiltonweg • Nachbarschaftsmatrix 	<ul style="list-style-type: none"> → Physik
<ul style="list-style-type: none"> ■ Computernetzwerke 	<ul style="list-style-type: none"> • einfache Netzwerkarchitektur skizzieren • verschiedene Netzwerkschichten unterscheiden • Client-Server, Cloud • Kommunikation (Protokolle), Übermittlung und Adressierung von Daten in Computernetzwerken beschreiben • Dienste 	<ul style="list-style-type: none"> → Alle Fächer
<ul style="list-style-type: none"> ■ Programmieren 2 	<ul style="list-style-type: none"> • bedingte Verzweigungen, • lokale, globale Variablen • logische Operatoren • Listen (Arrays) • Animationen 	
<ul style="list-style-type: none"> ■ Robotik 1 	<ul style="list-style-type: none"> • Grundlagen zu Sensoren, Aktoren, Regelung/Steuerung • Programmierung • künstliche Intelligenz, lernende Software 	<ul style="list-style-type: none"> → Physik

■ Projekt	• Mindstorms	
■ Kurs (z.B. Blockunterricht)	• Tabellenkalkulation (Befehle, Diagramme, Einfache Datenanalyse)	→ Mathematik (Funktionen, Statistik)

Grundlagenfach: 4. Klasse

Grobziele	Lerninhalte	Querverweise
■ Fehlererkennung	<ul style="list-style-type: none"> • selbstverifizierende Codes • Prüfziffern, Redundanz • Teilbarkeit, modulares Rechnen, Hammingcode, • Strichcode, QR-Code 	<ul style="list-style-type: none"> → Mathematik (GLS) → Wirtschaft und Recht → Bildnerisches Gestalten → Physik
■ Suchen und Sortieren 2	<ul style="list-style-type: none"> • Datenorganisation, Hashing • Einführung in die Konzepte „Teile und Herrsche“ und „Rekursion“ • Suchmethoden (Merge-, Quicksort) • Suchmaschinen 	→ Wirtschaft und Recht
■ Optimieren	<ul style="list-style-type: none"> • kürzeste Wege • Greedy Algorithmus • minimaler Spannbaum 	<ul style="list-style-type: none"> → Wirtschaft und Recht → Geschichte
■ Programmieren 3	<ul style="list-style-type: none"> • Logische Operatoren (AND, OR, XOR, NOT) • Datenstrukturen: Matrix, Stack, Queue • integrierte Anwendungen in den obigen Gebieten (Suchen, Sortieren etc.) 	→ Mathematik (Aussagenlogik)
■ Robotik 2	<ul style="list-style-type: none"> • Sensorik, Mechanik • Antriebssteuerung • Programmierung 	→ Physik
■ Projekt	<ul style="list-style-type: none"> • z.B. Robotikprojekt (Teil 2) • z.B. Raspberry Pi-Projekt 	

Methoden: Präsentationstechnik (3. Klasse)
Internetrecherche (3. Klasse)
Umfrage (4. Klasse)

Ergänzungsfach: 5. Klasse

Grobziele	Lerninhalte	Querverweise
■ Datenbanktechnik	<ul style="list-style-type: none"> • Konzept der relationalen DB • Datenanalyse • Attribut, Schlüssel, Entitäten, Beziehungen • Modellierung (ER-Modell) • Umsetzung mit Tabellen • Datenbankabfragen 	→ Wirtschaft und Recht
■ Kryptologie	<ul style="list-style-type: none"> • historischer Kontext • (Turing, Enigma, Public-Key) • moderne Kryptosysteme • (Anwendungen Public-Key-Kryptografie: Digitale Unterschrift, Zero-Knowledge) 	→ Geschichte → Mathematik
■ Programmieren	<ul style="list-style-type: none"> • Rekursion vs. Iteration • Anwendungen in Spieltheorie, Fraktale, etc.) • numerische Methoden • (Regula Falsi, Newton, Euler) 	→ Mathematik (Folgen)
■ Mögliche Vertiefungsgebiete	<p><i>Robotik/Mikrocontroller</i></p> <ul style="list-style-type: none"> • Odometrie • Steuer-/Regelungstechnik • Machine Learning • konkrete Programmierung <p><i>Webtechnologien</i></p> <ul style="list-style-type: none"> • Funktionsweise von Webapplikationen • Planung/Realisierung von statischen und dynamischen Webseiten • Datenbankanbindung <p><i>Spieltheorie</i></p> <ul style="list-style-type: none"> • Spielbaum • MinMax-/AlphaBeta-Algor. • Breiten-/Tiefensuche, A* • Heuristik, Bewertungsfunktion 	→ Physik (Elektronik) → Mathematik (Stochastik) → Mathematik (Rekursion)

Ergänzungsfach: 6. Klasse

Grobziele	Lerninhalte	Querverweise
<ul style="list-style-type: none"> ■ Berechenbarkeit 	<ul style="list-style-type: none"> • Grenzen der Automatisierbarkeit • Methode der Reduktion • Konzept der Unendlichkeit (Hilbert, Cantor) • Entscheidungsprobleme 	<ul style="list-style-type: none"> → Mathematik (Logik) → Mathematik (Analysis) → Philosophie
<ul style="list-style-type: none"> ■ Algorithmmik 	<ul style="list-style-type: none"> • Augenmerk auf Entwurf und Analyse von Algorithmen • Optimierung (Greedy) • Iteration (Dynamische Suche) • Rekursion (Teile u. Herrsche) • Backtracking • Berechnungskomplexität • O-Notation, Rekurrenzgleichungen 	<ul style="list-style-type: none"> → Mathematik (Kombinatorik, Folgen / Reihen) → Mathematik (Wachstumsf.)
<ul style="list-style-type: none"> ■ Programmieren 	<ul style="list-style-type: none"> • Randomisierte Algorithmen • Programmierung von Simulationen 	<ul style="list-style-type: none"> → Mathematik (Stochastik) → Mathematik, NW
<ul style="list-style-type: none"> ■ Mögliche Vertiefungsgebiete 	<p><i>Automaten</i></p> <ul style="list-style-type: none"> • modularer Entwurf von endlichen Automaten • Alphabet, Wörter, Syntax, Semantik, Entscheidungsproblem <p><i>Computergrafik</i></p> <ul style="list-style-type: none"> • geometrische Modellierung • affine Abbildungen • Rendering, Shading • Animationen <p><i>Robotik / Mikrocontroller</i></p> <ul style="list-style-type: none"> • Odometrie • Steuer-/Regelungstechnik • Machine Learning • Programmierung <p><i>Objektorientiertes Programmieren</i></p> <ul style="list-style-type: none"> • Prozess der Softwareentwicklung (Software Engineering) 	<ul style="list-style-type: none"> → Sprachfächer → Mathematik (VG) → Bildnerisches Gestalten → Mathematik (Trigonometrie) → Physik (Elektrotechnik)