

ERGÄNZUNGSFACH INFORMATIK

1. STUNDENDOTATION

	1. Klasse	2. Klasse	3. Klasse	4. Klasse	5. Klasse	6. Klasse
1. Semester					2	2
2. Semester					2	2

2. BILDUNGSZIELE

Die Informatik durchdringt zunehmend alle Bereiche des Lebens. Sie betrifft in der Anwendung alle wissenschaftlichen Fachrichtungen. Das Ergänzungsfach vermittelt die Kompetenz, die Einsatzmöglichkeiten der Informatik zu beurteilen, Lösungen zu vergleichen und bei deren Ausgestaltung mitzusprechen.

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 kritische Beurteilung im Vordergrund.

Das Ergänzungsfach Informatik befähigt die Lernenden zur Analyse und Modellierung von Problemstellungen sowie zum Entwurf von algorithmischen Lösungen. Ihre Realisierung durch selbst geschriebene Programme ermöglicht eine direkte Überprüfung der Lösungsqualität. Die Lernenden erfahren, was automatisch machbar bzw. durch Maschinen realisierbar ist und welche Ressourcen dazu nötig sind.

Das Ergänzungsfach Informatik soll Grundlagen in den Bereichen Algorithmik, Programmieren, Theoretische Informatik, Information und Kommunikation vermitteln. In einem oder mehreren dieser Bereiche findet eine Vertiefung statt, die sich besonders für ein projektorientiertes und vernetztes Vorgehen eignet.

Der Unterricht in Informatik

- baut auf die im Informatikunterricht erworbenen Kenntnisse auf und erweitert diese
- entwickelt die Fähigkeit zu erkennen, dass vielfältige Probleme aus dem Alltag, der Technik und der Wissenschaften einer informatischen Bearbeitung zugänglich sind. Dazu werden Programme entworfen, geprüft, weiter entwickelt und bezüglich der abgebildeten Wirklichkeit beurteilt
- befähigt, eine Programmiersprache beim Lösen von Problemen einzusetzen
- fördert genaues analytisches Denken gepaart mit pragmatischem, zielgerichtetem Vorgehen sowie ausdauerndes exaktes Arbeiten
- hilft wesentlich mit, sich in unserer komplexen hoch technisierten Welt zurechtzufinden

3. RICHTZIELE

GRUNDKENNTNISSE

Maturandinnen und Maturanden kennen

- Algorithmen und ihre Darstellungsarten
- eine objektorientierte Programmiersprache
- verschiedene Datenstrukturen und ihre Einsatzmöglichkeiten
- diverse Such- und Sortieralgorithmen sowie ihre Implementierung
- die Anwendungsgebiete der Informatik
- die Funktionsweise der Datendarstellung, der Datenstrukturierung und der Datenkommunikation

GRUNDFERTIGKEITEN

Maturandinnen und Maturanden

- erfassen und analysieren Probleme aus verschiedenen Sachgebieten und setzen sie in Algorithmen um
- implementieren Algorithmen mit einer strukturierten und objektorientierten Programmiersprache und testen sie
- wenden die einfachen und strukturierten Datentypen an und können sie programmieren
- setzen die dynamischen Datenstrukturen und ihre Anwendungen bei Listen und Bäumen ein
- verstehen die Grundlage der Datenkommunikation (LAN- und WLAN-Technologie)
- führen ein Informatik-Projekt mit Hilfe des Software Engineerings (Planung, Analyse, Implementation, Tests, Dokumentation) durch

GRUNDHALTUNGEN

Maturandinnen und Maturanden sind bereit,

- mit einer Programmiersprache verantwortungsbewusst zu arbeiten
- sich den Schwierigkeiten und Anforderungen angewandter Probleme zu stellen und für Kritik offen zu sein
- sich mit den Auswirkungen der Informatik im Alltag auseinanderzusetzen
- andere Fachbereiche der Informatik zu unterstützen und auch deren fachliche Beiträge und Anregungen anzunehmen
- mit Freude zu experimentieren und Informatiklösungen zu realisieren

ERGÄNZUNGSFACH	5. und 6. KLASSE	INFORMATIK
Grobziele	Lerninhalte	Querverweise
ICT-Grundkenntnisse ausgleichen	Praktische Hardware-Kunde; Interne Funktionsweise des Computers; Codes; OS; Seitenbeschreibungssprachen; Anwenderprogramme	IN 2. Klasse LZG
<i>Grundlagen: Logik</i>	<i>Aussagenlogik Formalisierungen von Alltagssprache, Wahrheitstafeln Einführung in Prolog mit Anwendungen</i>	MA
Grundlagen: Natürliche / formale Sprachen	Grammatik, Syntax, Semantik	DE, Fremdsprachen
Grundlagen: Laufzeitverhalten	Modelle, Bedingungen, <i>Klasse P, Grenzen von P</i> ; Naiver Ansatz und Reflexion über Effizienz, Berechenbarkeit	MA PH: reflektierter Umgang mit Texten und Modellen, Schriftlichkeit
Techniken und Theorie: Programmieren 1	Algorithmus und Darstellungen EBNF, FSM, Pseudocode, Struktogramm, <i>Datenflussdiagramm, UML</i>	MA: euklidischer Algorithmus
Techniken: Programmieren 2	Einführung in grundlegende Typen von Algorithmen und Datenstrukturen: Rekursion, <i>Backtracking, Divide & Conquer, dynamic, greedy</i>	
Techniken: Programmieren 3	Imperatives Paradigma und/oder <i>OO-Paradigma vertiefen</i>	INT (IN integriert in Mathematik)
<i>Techniken: Programmieren 4</i>	<i>Qualitätssicherung: Verifikation, Reviews, Testverfahren</i>	
Techniken und Theorie: Datenbanken	Relationale Modellierung und Normalisierung <i>SQL, Transaktionsbegriff Aufbau und Betrieb eines RDBMS</i>	
<i>Techniken: Netzwerke und Kommunikation</i>	<i>Architekturen und Strategien des Internets (inkl. Chancen und Gefahren) Aufbau und Betrieb eines LANs</i>	
Projekte: Theorie	Projektorganisation und -methoden: Wasserfallmodell und <i>Alternativen Erhebungs-, Beschreibungs- und Dokumentationsmethoden</i>	
Projekte: Praxis 1	Teambildung, Gruppendynamik, <i>Führung, Umgang mit Widerstand</i>	PP (Pädagogik und Psychologie)
Projekte: Praxis 2	Konkrete Projekterfahrung - selbständige Arbeit - Arbeit im Team	
<i>Reflexion und Kommunikation</i>	<i>Der Sinn der Informatik über die Zeit Die Rolle der InformatikerIn und deren sozialkompetente Kommunikation</i>	PH Medienkunde