

Ergänzungsfach Informatik

1. Stundendotation

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

Die Informatik durchdringt zunehmend alle Bereiche des Lebens. Sie betrifft in der Anwendung alle wissenschaftlichen Fachrichtungen. Das Ergänzungsfach EFI 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 EFI 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.

2. Allgemeine Bildungsziele

Der Unterricht in Informatik

- baut auf die im Informatikunterricht erworbenen Kenntnisse auf und erweitert diese
- entwickelt die Fähigkeit zu erkennen, dass vielfältige Probleme im Alltag, der Technik und der Wissenschaft einer informatischen Bearbeitung zugänglich sind. Dazu werden Programme entworfen, geprüft, weiterentwickelt 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 zurecht zu finden

3. Richtziele Fach

Grundkenntnisse

Die 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
- wie man ein Projekt aus dem Gebiet der Informatik mit Hilfe der Software Engineering durchführt

Grundfertigkeiten

Die Maturandinnen und Maturanden können

- Probleme aus verschiedenen Sachgebieten erfassen, analysieren und in Algorithmen umsetzen
- Algorithmen mit einer strukturierten und objektorientierten Programmiersprache implementieren und testen
- die einfachen und strukturierten Datentypen anwenden und programmieren
- die dynamischen Datenstrukturen und ihre Anwendungen bei Listen und Bäumen einsetzen
- die Grundlage der Datenkommunikation (LAN- und WLAN-Technologie) verstehen
- Einsicht in die Anwendungsgebiete der Informatik nehmen
- An einem Informatik-Projekt die Planung, die Analyse, die Implementation, das Testen und das Dokumentieren durchführen.

Grundhaltungen

Die Maturandinnen und Maturanden sind bereit

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

Grobziele	Inhalte	Querverweise
<p>Algorithmen Algorithmusbegriff, seine Struktur und seine Darstellungsarten verstehen und anwenden</p>	<p>Definition des Begriffs «Algorithmus», Beispiele von Algorithmen Darstellungsarten von Algorithmen: Pseudocodes, Struktogramme, Datenflussdiagramme algorithmisch unlösbare Probleme Strukturen von Algorithmen: Sequenz, Selektion, Iteration, Objekt-konzept, Rekursion</p>	<p>PAM-AM 12: Euklid'scher Algorithmus (GGT), Heronscher Wurzelalgorithmus; Numerische Verfahren: Methode von Newton; Numerische Integration: Methode von Simpson PAM-AM 10: Simulation, Elektronik WR 11: Kapitalverzinsung</p>
<p>Programmierung Eine moderne höhere Programmiersprache kennen lernen</p>	<p>Einfache Algorithmen entwerfen Entwicklung der Programmiersprachen Programmiertechniken: Einfache Datentypen, Objekte, Eingabe, Ausgabe, Zuweisung, Verzweigung, Schleifen, Unterprogramme</p>	
<p>Probleme mit Hilfe von einfachen und strukturierten Datentypen lösen Sich in der Programmierung vertiefen (Wahlthemen)</p>	<p>Strukturierte Datentypen: Felder (Array), Mengen (Set), Verbund (Record) Arbeit mit dynamischen Variablen: Zeiger, Operationen Anwendungen: Listen und Bäume Programmierung einer Datenbank, rekursive Programmierung <i>Anwendung, z.B. Robotersteuerung</i> Qualitätssicherung: Verifikation, Reviews, Testverfahren</p>	<p>MA 11: Vektoren, Matrizen MNS 8: <i>Maschinensteuerung</i></p>

Grobziele	Inhalte	Querverweise
<p>Weblösungen programmieren (Themen nach Wahl)</p> <p>Datenstrukturen Daten verarbeiten Datenkommunikation Datenbank entwerfen</p> <p>Projektthemen Sich im Rahmen von Informatik-Projekten in einem Gebiet auseinandersetzen und sich vertiefen</p>	<p>Dynamische Webseiten mit Datenbank-Anbindung (z.B. PHP/MySQL), Content-Management-Systeme, Design mit CSS-Standards, Accessibility</p> <p>Suchverfahren Sortieren von Feldern: Sortieren durch Auswahl, Einfügen, Austauschen sowie Quicksort Information: Darstellung und Übertragung, Schichtenmodelle, Grundlagen von LAN- und WLAN-Standards, Verschlüsselungsverfahren, Kryptographie, Datenkompression Datenbankkonzepte und -modelle, Normalisierung Entwurf, Abfrage, Datenschutz</p> <p>Projekte nach Wahl aus den Bereichen: Datenbank, Visualisierung, Webdesign, Multimedia, Datenschutz und Datensicherheit, Robotik, Netzwerke Konkrete Projekterfahrung: Arbeit im Team, Projektorganisation und -methoden, Erhebungs-, Beschreibungs- und Dokumentationsmethoden, Testing, Usability</p>	<p>alle Fächer</p>

Fächerübergreifender Unterricht

DE/PP 12: Kommunikationsmodelle