

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

4. GROBZIELE

| GROBZIELE 5. KLASSE | LERNINHALTE | QUERVERWEISE |
|--|--|---|
| <p>ALGORITHMEN</p> <ul style="list-style-type: none"> ● Algorithmusbegriff, seine Struktur und seine Darstellungsarten verstehen und anwenden | <p>Definition des Begriffs „Algorithmus“, Beispiele von Algorithmen Darstellungsarten von Algorithmen: Stilisierte Prosa, Pseudocode, Ablaufdiagramme, Struktogramme, algorithmisch unlösbare Probleme</p> <p>Strukturen von Algorithmen: Sequenz, Selektion, Iteration, Objektkonzept, Rekursion Einfache Algorithmen entwerfen</p> | <p><i>Mathematik:</i> Euklidischer Algorithmus (GGT), Heronscher Wurzelalgorithmus, Numerische Verfahren: Methode von Newton, Numerische Integration: Methode von Simpson</p> <p><i>Physik:</i> Simulation, Elektronik <i>Biologie:</i> Bevölkerungsentwicklung (Jäger-Beute-Simulation) <i>Wirtschaft:</i> Kapitalverzinsung</p> |
| <p>PROGRAMMIERUNG</p> <ul style="list-style-type: none"> ● Eine moderne höhere Programmiersprache kennen lernen ● Probleme mit Hilfe von einfachen und strukturierten Datentypen lösen ○ 2D-Grafik programmieren | <p>Entwicklung der Programmiersprachen Programmiertechniken: Einfache Datentypen, Objekte, Eingabe, Ausgabe, Zuweisung, Verzweigung, Schleifen, Unterprogramme</p> <p>Strukturierte Datentypen: Felder (Array), Mengen (Set), Verbund (Record)</p> <p>Programmieren von Linien, Rechtecken, Ellipsen, Polygonen, Anwendungen, 2D-Animationen</p> | <p><i>Mathematik:</i> Vektoren, Matrizen</p> <p><i>Bildnerisches Gestalten:</i> 2D-Grafik, Präsentation, Webpublikation</p> |
| <p>DATENSTRUKTUREN</p> <ul style="list-style-type: none"> ● Daten verarbeiten ● Datenbank entwerfen | <p>Suchverfahren, Sortieren von Feldern: Sortieren durch Auswahl, durch Einfügen, durch Austauschen, Quicksort</p> <p>Verschiedene Datenbankkonzepte Entwurf, Abfrage, Datenschutz</p> | <p><i>Alle Fächer</i></p> |
| <p>PROJEKTTHEMEN</p> <ul style="list-style-type: none"> ● Sich im Rahmen von Informatik-Projekten in einem Gebiet vertiefen | <p>Projekte nach Wahl aus den Bereichen: Geschichte der Informatik, Theoretische Informatik, Datenbank, Bildverarbeitung, Visualisierung, Künstliche Intelligenz (künstliche neuronale Netze), Expertensysteme, Web-Design, Multimedia, Datenschutz und Datensicherheit</p> | <p><i>Biologie:</i> Das Gehirn, biologische neuronale Netze</p> |

| GROBZIELE 6. KLASSE | LERNINHALTE | QUERVERWEISE |
|--|--|--|
| <p>PROGRAMMIERUNG</p> <ul style="list-style-type: none"> ● Sich in der Programmierung vertiefen ● Datenkommunikation ○ 3D-Grafik programmieren | <p>Arbeit mit dynamischen Variablen: Zeiger, Operationen Anwendungen: Listen und Bäume</p> <p>Programmierung einer Datenbank rekursive Programmierung</p> <p>Information: Darstellung und Übertragung, das OSI-Schichtenmodell Grundlagen von LAN und WLAN Standards, Verschlüsselungsverfahren, Datenkompression</p> <p>3D-Szenen erstellen: Modeling, Rendering, Animation</p> | <p><i>Physik: Signalübertragung</i></p> <p><i>Bildnerisches Gestalten: 3D-Grafik, Darstellung</i></p> |
| <p>PROJEKTTHEMEN</p> <ul style="list-style-type: none"> ● Sich im Rahmen von Informatik-Projekten in einem Gebiet vertiefen | <p>Projekte nach Wahl aus den Bereichen: Numerik, Objekterkennung, Künstliche Intelligenz, Robotik, Datenkommunikation, Computernetzwerke: Grundlage, Topologie, Computerschutz</p> | |

- Pflichtprogramm

5. FACHRICHTLINIEN

LEHRMITTEL

- Bei der Auswahl des Lehrmittels ist die Lehrperson frei.

ORGANISATORISCHES

- In der Regel werden die Klassen zwei Jahre von der gleichen Lehrperson unterrichtet.
- Auf Grund der Komplexität der individuellen Fragestellungen der Lernenden und des damit direkt verbundenen Betreuungsaufwandes ist die Klassengrösse auf maximal 12 Lernende zu beschränken.

UNTERRICHTSSPRACHE

- Hochdeutsch

BEWERTUNG

- Die Bewertungskriterien müssen am Semesteranfang den Klassen bekannt gegeben werden.