

**Internes Curriculum Rosa-Luxemburg-Gymnasium  
 Informatik Qualifikationsphase**

**Stand: 07.07.2017**

*Unterrichtsdauer: 135 min pro Woche (3 Unterrichtsstunden)*

**Vorbemerkungen:**

Informatik kann erstmals in Klasse 9 mit 55 min pro Woche belegt werden. Es findet keine Fortsetzung in Klasse 10 statt. In der Oberstufe kann deshalb Informatik erneut als Anfangsunterricht (Grundkurs ohne Möglichkeit der Wahl als 3. oder 4. Prüfungsfach) belegt werden. Schüler, die 4 Semester Informatik belegen, können Informatik im Rahmen der 5. Prüfungskomponente als Prüfungsfach wählen. Die Inhalte sind für 60 % des Schuljahres ausgelegt. Die übrigen 40 % werden nach Leistungsfähigkeit und Interessenlage der Schüler gestaltet.

**Erstes und zweites Semester:**

Themen/Inhalte	Kompetenzen	Rahmenplanbezug	Zeitplan
<b>Datenbanken</b> <ul style="list-style-type: none"> <li>• Datenmodellierung</li> <li>• relationales Datenbankschema</li> <li>• praktische Umsetzung in ein Datenbank-Managementsystem</li> <li>• Abfragen (Projektion,</li> </ul>	<b>Modelle erstellen und nutzen</b> <ul style="list-style-type: none"> <li>• beschreiben Objekte und Beziehungen in einer graphischen Repräsentation,</li> <li>• überführen das Modell in ein Datenbankschema,</li> <li>• normalisieren gegebene Datenbestände nach den ersten drei Normalformen</li> </ul>	<b>4.1 Datenbanken</b>	Mindestens 8 Wochen

<p>Selektion, Join)</p> <ul style="list-style-type: none"> <li>• Normalisierung</li> </ul> <p><b>Mögliche Schwerpunkte und Kontexte:</b></p> <ul style="list-style-type: none"> <li>• Analyse, Modellierung und Implementation einer Datenbank auf der Grundlage eines realen Anwendungsfalles</li> <li>• Analyse, Wartung und Implementation einer bestehenden Datenbank</li> <li>• Umsetzung einer bisher mit einer Programmiersprache gelösten Dateiverwaltung mit einer Datenbank</li> </ul>	<p><b>Anwendungen erfassen und Auswirkungen abschätzen</b></p> <ul style="list-style-type: none"> <li>• nehmen das Recht auf informationelle Selbstbestimmung wahr und halten die Gesetze zum Datenschutz ein</li> </ul>		
<p><b>Objektorientierte Programmierung</b></p> <ul style="list-style-type: none"> <li>• Grundlagen der Programmentwicklung</li> <li>• objektorientierte Modellierung (UML-Klassendiagramme) Algorithmen und Datenstrukturen</li> <li>• objektorientierte Programmierung</li> <li>• Grundlagen systematischer Softwareentwicklung (Software-Life-Cycle)</li> </ul> <p><b>Mögliche Schwerpunkte und Kontexte:</b></p> <ul style="list-style-type: none"> <li>• Strukturelle Aspekte bei der Konstruktion von</li> </ul>	<p><b>Modelle erstellen und nutzen</b></p> <ul style="list-style-type: none"> <li>• analysieren Realitätsausschnitte und wählen ein geeignetes Modellierungsverfahren aus,</li> <li>• variieren und erweitern vorgegebene Modelle,</li> <li>• entwickeln, implementieren, testen und validieren einfache Modelle,</li> <li>• reflektieren und beurteilen die eigene Modellierung,</li> <li>• beschreiben Basiskonzepte der objektorientierten Modellierung (Klasse, Objekt, Attribut, Methode),</li> <li>• entwerfen Methoden für die Manipulation von Objekten,</li> <li>• bilden Beziehungen zwischen Klassen ab,</li> <li>• wenden die Konzepte von Vererbung, Polymorphie und Kapselung an.</li> </ul> <p><b>Information in Form von Daten darstellen und verarbeiten</b></p> <ul style="list-style-type: none"> <li>• konstruieren Daten- bzw. Objektstrukturen und wenden auf</li> </ul>	<p><b>4.3 Softwareentwicklung</b></p>	<p>Mindestens 12 Wochen</p> <p>Je nach Vorgaben ist ein Tausch mit dem Themenbereich Datenbanken möglich.</p>

<p>Programmsystemen stehen im Vordergrund.</p> <ul style="list-style-type: none"> <li>• Analyse und Modellierung folgen dem objektorientierten Ansatz.</li> <li>• Projektorientierte Vorgehensweise mit einem oder mehreren Kurzprojekten</li> <li>• Dokumentation der Entwurfsschritte mit Hilfe des Rechners</li> <li>• Funktionale Spezifikation bzw. Klassenbeschreibungen (z. B. mit Hilfe von UML), Anforderungsdefinition, Testumgebungen, Quelltexte</li> <li>• Einsatz eines einfachen CASE-Werkzeuges.</li> </ul>	<p>diese geeignete Algorithmen bzw. Methoden an,</p> <ul style="list-style-type: none"> <li>• speichern und übertragen wieder verwendbare Ergebnisse in geeigneter Form,</li> <li>• unterscheiden zwischen Syntax und Semantik und erläutern dies in natürlichen, halb- formalen und formalen Sprachen,</li> <li>• implementieren zusammengesetzte und dynamische Daten- bzw. Objektstrukturen (Listen, Bäume) und wenden diese an.</li> </ul> <p><b>Wirkprinzipien kennen und anwenden</b></p> <ul style="list-style-type: none"> <li>• konstruieren Software unter Beachtung des Prinzips der Modularisierung (Schnittstellen).</li> </ul> <p><b>Problemlösen</b></p> <ul style="list-style-type: none"> <li>• wenden die Phasen des Problemlöseprozesses (informelle Problembeschreibung, formale Modellierung, Implementierung und Realisierung, Bewertung und Modellkritik) an,</li> <li>• setzen im Problemlöseprozess einfache Entwicklungswerkzeuge ein.</li> </ul> <p><b>Kommunizieren und Kooperieren</b></p> <ul style="list-style-type: none"> <li>• verfügen über eine angemessene Fachsprache und verwenden sie sachgerecht,</li> <li>• verwenden selbstständig Fachtexte, Dokumentationen und Hilfesysteme,</li> <li>• dokumentieren, visualisieren, präsentieren und verteidigen Ergebnisse der Teamarbeit ,</li> <li>• organisieren selbstständig Projektarbeit.</li> </ul>		
---	--	--	--

### Drittes Semester

Themen/Inhalte	Kompetenzen	Rahmenplanbezug	Zeitplan
<p><b>Netzwerke</b></p> <ul style="list-style-type: none"> <li>• Schichtenmodelle</li> <li>• Client-Server-Struktur</li> <li>• Protokolle</li> <li>• Kommunikations- und</li> </ul>	<p><b>Kommunizieren und Kooperieren</b></p> <ul style="list-style-type: none"> <li>• setzen netzbasierte Kommunikations- und Kooperationssysteme in der Gruppenarbeit ein,</li> <li>• dokumentieren, visualisieren, präsentieren und verteidigen Ergebnisse der Teamarbeit ,</li> <li>• organisieren selbstständig Projektarbeit.</li> </ul>	<p><b>4.3</b> <b>Softwareentwicklung</b></p>	<p>Mindestens 2 Wochen</p>

<p>Kooperationssysteme</p> <p><b>Mögliche Kontexte:</b></p> <ul style="list-style-type: none"> <li>• Simulation von Rechnernetzen</li> <li>• Physical Computing</li> </ul>	<p><b>Mit Information umgehen</b></p> <ul style="list-style-type: none"> <li>• analysieren und strukturieren Informationen.</li> </ul> <p><b>Informatiksysteme verstehen</b></p> <ul style="list-style-type: none"> <li>• erläutern einfache Schichtenmodelle von Netzwerken und Informatiksystemen,</li> <li>• wenden das Adressierungsprinzip (IP-Adresse, Subnetze) in Netzwerken auf der Basis des Internetprotokolls an.</li> </ul>		
<p><b>Algorithmen und Datenstrukturen</b></p> <ul style="list-style-type: none"> <li>• Suchen, Sortieren</li> <li>• Rekursion</li> <li>• Rekursive Datenstrukturen (Listen, Bäume, ...)</li> <li>• Syntaxdiagramme</li> </ul> <p><b>Mögliche Schwerpunkte und Kontexte:</b></p> <ul style="list-style-type: none"> <li>• Einfache Sortierverfahren</li> <li>• Quicksort</li> <li>• Sequentielle und Binäre Suche</li> <li>• einfache Komplexitätsbetrachtungen</li> </ul>	<p><b>Informatische Modellieren</b></p> <ul style="list-style-type: none"> <li>• entwickeln, implementieren, testen und validieren einfache Modelle,</li> <li>• wenden rekursive Verfahren an.</li> </ul> <p><b>Problemlösen</b></p> <ul style="list-style-type: none"> <li>• wenden die Phasen des Problemlöseprozesses (informelle Problembeschreibung, formale Modellierung, Implementierung und Realisierung, Bewertung und Modellkritik) an,</li> <li>• nutzen informatische Werkzeuge zur Problemlösung,</li> <li>• beachten Grenzen des Problemlösens mit Informatiksystemen.</li> </ul> <p><b>Mit Information umgehen</b></p> <ul style="list-style-type: none"> <li>• implementieren zusammengesetzte und dynamische Daten- bzw. Objektstrukturen (Listen, Bäume) und wenden diese an.</li> </ul> <p><b>Informatiksysteme verstehen</b></p> <ul style="list-style-type: none"> <li>• beurteilen Algorithmen hinsichtlich ihrer Effizienz,</li> <li>• ordnen Algorithmen gegebenen Komplexitätsklassen zu.</li> </ul>	<p><b>4.3 Softwareentwicklung</b></p>	<p>Mindestens 8 Wochen</p>

#### Viertes Semester:

Das vierte Semester steht je nach Leistungstand und Interesse der Schüler zur freien Verfügung.

#### Mögliche Inhalte:

- Technische Informatik
- Komplexes Softwareprojekt
- Computergrafik
- Künstliche Intelligenz
- Kryptographie und Datensicherheit
- Theoretische Informatik
- Maschinennahe Programmierung
- Computernetze