

STÄDTISCHES
GYMNASIUM AN DER HÖNNE
MENDEN (SAUERLAND)



GHM

Curriculum Informatik
Sekundarstufe II

Städt. Gymnasium an der Hönne, Menden: Curriculum für das Fach Informatik in der Sekundarstufe II

Das Fach Informatik gehört in das 3. Aufgabenfeld und kann als Pflichtbindung die 2. Naturwissenschaft bei einem naturwissenschaftlichen Schwerpunkt ersetzen. Das Fach wird nur als Grundkurs angeboten. D.h., dass Informatik nur als 3. oder 4. Abiturfach gewählt werden kann. Unterrichtet wird das Fach ab der Einführungsphase, für Schüler, die vom Gymnasium kommen gibt es in den Jahrgangsstufen 8 und 9 einen Mathematik-Informatik-Physik – Kurs und viele Schüler von der Realschule hatten im Wahlpflichtbereich einen Kurs Informatik gewählt. Diese nehmen i.a. keine Inhalte der Oberstufe vorweg. Damit ergibt sich allerdings auch das Problem Schüler mit sehr unterschiedlichen Vorkenntnissen und Vorerfahrungen in den Kurs zu integrieren. Darum sind die als „Vorkurs“ gekennzeichneten Unterrichtsvorhaben so wichtig.

Üblicherweise soll das Fach als Grundkurs mit drei Wochenstunden unterrichtet werden, darin ist üblicherweise eine Doppelstunde.

Mit den Kernlehrplänen ab Sommer 2014 wird als Programmiersprache JAVA mit der Programmierumgebung BlueJ vorgegeben. Die neuen Kernlehrpläne ermöglichen, sich auch weiter am Konzept „Stifte und Mäuse“ zu orientieren, das auch Grundlage für die Aufgaben im Zentralabitur ist.

Für dieses Konzept spricht auch, dass alle benötigte Software frei erhältlich ist. Weiterhin spricht für dieses Konzept, dass BlueJ auch als Programmierumgebung der Universitäten benutzt wird. Daneben wird die freie Software Open Office oder Libre Office benutzt.

Unterrichtsmaterial:

Wir benutzen die Bücher: B. Schriek, Informatik mit Java, Bd. I,(EF) Bd. II (Q1); Bd. III (Q2). Die im Curriculum aufgeführten Projekte sind den entsprechenden Kapiteln dieser Bücher entnommen. Für den „Vorkurs“ in der EF und dem Themenbereich „Datenbanken“ wird die Software Libre Office und dazu passende Unterrichtsmaterial aus dem Internet verwandt.

Unterrichtsvorhaben:

Die in der nachfolgenden Auflistung erwähnten Unterrichtsvorhaben sollen alle im Kernlehrplan abdecken. Diese Unterrichtsvorhaben sind demnach für alle Kollegen verbindlich. Mit welchen konkreten Projekten, Materialien und Methoden die Kompetenzen, die den Unterrichtsvorhaben zugeordnet sind erreicht werden, bleibt weiterhin den Kollegen überlassen. Allerdings zeigt die Erfahrung, dass auf die Projekte aus dem eingeführten Lehrbuch zugegriffen werden wird. Die zeitliche Einteilung kann nur eine grobe Orientierung liefern, da es immer wieder Unterbrechungen (Praktika, Fahrten, Kürzungen des Unterrichts) gibt. Somit sollten die verbindlichen Projekte nur ca. 75% der theoretisch zur Verfügung stehenden Unterrichtszeit abdecken. Damit bleibt auch ggf. Zeit für Vertiefungen und Ergänzungen oder Besprechung aktueller technologischer Entwicklungen.

Zentrale Kompetenzen:

Diese orientieren sich auch an denen anderer Fächer im 3. Aufgabenfeld. Dazu zählen:

Argumentieren (A)

Modellieren (M)

Darstellen und Interpretieren (D)

Kommunizieren und Kooperieren (K)

typisch für die Informatik ist die fünfte: Implementieren und Programmieren) (I)

Zur Methode:

Der Informatikunterricht ist grundsätzlich durch Projekte gegliedert. Kleine Ausnahmen davon gibt es nur im Vorkurs EF.1- Grundsätzlich laufen diese Projekte nach dem folgenden Schema ab:

- 1) Ausgabenstellung durch Lehrer und/oder Lehrbuch: Die Erstellung eines Pflichtenheftes (ab EF.2) erfolgt zunehmend selbständig durch die Schüler
- 2) Erste Lösungsansätze → Suche nach entsprechenden Hilfsmitteln, Werkzeugen, Strukturen mit Hilfe des Lehrers, des Lehrbuches, der Hilfeseiten, entsprechender Seiten im Internet (von EF.1 → Q2) in Arbeitsgruppen, zunehmend selbständiger
- 3) Klären und Sichern der benötigten Werkzeuge und Algorithmen (Information: Hier lässt sich nichts erarbeiten, da diese durch die verwandte Programmierumgebung und Programmiersprache vorgegeben sind.)
- 4) Implementieren - Programmieren der erarbeiteten Oberflächen und Algorithmen in Arbeitsgruppen, arbeitsteilig bei umfangreichen Projekten
- 5) Testen, Präsentation, Reflektieren und Erweitern der gefundenen Lösungen in den Projektgruppen und vor der Klasse

Jg	Thema und Unterrichtsvorhaben	Projekte, Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
EF.1 max 3 Mo nate	<p>Die beiden Unterrichtsvorhaben EF I und EF II bilden einen „Vorkurs“. Dieser „Vorkurs“ soll auch Schüler und Schülerinnen, die in der Sekundarstufe I keine Informatik hatten, mit dem Grundwissen über die folgenden Inhalte versorgen:</p> <p><u>Unterrichtsvorhaben (EF I):</u> Einführung in die Nutzung von Informationssystemen und in grundlegende Begrifflichkeiten</p>	<p><u>1. Information, deren Kodierung und Speicherung</u></p> <p>a) Informatik als Wissenschaft der Verarbeitung von Informationen b) Darstellung von Informationen in Schrift, Bild und Ton c) Speichern von Daten mit informatischen Systemen am Beispiel der Schulrechner d) Vereinbarung von Richtlinien zur Datenspeicherung auf den Schulrechnern (z.B. Ordnerstruktur, Dateibezeichner usw.)</p> <p><u>2. Informations- und Datenübermittlung in Netzen</u></p> <p>a) „Sender-Empfänger- Modell und seine Bedeutung für die Eindeutigkeit</p>	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> • beschreiben und erläutern den Aufbau singulärer Rechner am Beispiel der „Von Neumann Architektur“ (A) • nutzen die im Unterricht eingesetzten Informatiksysteme selbstständig, sicher, zielführend und verantwortungsbewusst (D) • nutzen das Internet zur Recherche, zum Datenaustausch und zur Kommunikation. 	<p>Beispiel Textkodierung: Kodierung von Texten mit unbekanntem Zeichensätzen (z. B. Wingsdings)</p> <p>Beispiel Bildkodierung: Kodierung von Bildinformationen in Raster- und Vektorgrafiken</p>

Jg	Thema und Unterrichtsvorhaben	Projekte, Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
	<p><u>Unterrichtsvorhaben EF II):</u> Geschichte der digitalen Datenverarbeitung und die Grundlagen des Datenschutzes</p> <p>Inhaltsfeld : Informatiksysteme: Inhaltsfeld : Informatik, Mensch und Gesellschaft</p> <p>Schwerpunkte: Wirkungen der Automatisierung Geschichte der automatischen Datenverarbeitung Digitalisierung</p> <p>Mögliche Themen zur Erarbeitung in Kleingruppen mit Vorstellung in Form einer Präsentation und Diskussion durch die Schüler:</p>	<p>von Kommunikation b) Informatische Kommunikation in Rechnernetzen am Beispiel des Schulnetzwerkes (z.B. Benutzeranmeldung, Netzwerkordner, Zugriffsrechte, Client-Server) c) Grundlagen der technischen Umsetzung von Rechnerkommunikation am Beispiel des Internets (z.B. Netzwerkadresse, Paketvermittlung, Protokoll) d) Richtlinien zum verantwortungsvollen Umgang mit dem Internet</p> <p><u>3.Aufbau informatischer Systeme</u> a) Identifikation typischer Komponenten informatischer Systeme und anschließende Beschränkung auf das Wesentliche, Herleitung der „Von Neumann Architektur“ b) Identifikation des EVA-Prozesses als Prinzip der Verarbeitung von Daten und Grundlage der „Von Neumann Architektur“</p>	<p>Die Schülerinnen und Schüler bewerten anhand von Fallbeispielen die Auswirkungen von Informatiksystemen (A) erläutern wesentliche Grundlagen der Geschichte der digitalen Datenverarbeitung (A) stellen ganze Zahlen und Zeichen in Binärcodes dar (D)</p>	<p>Beispiel: Rollenspiel zur Paketvermittlung im Internet Schülerinnen und Schüler übernehmen die Rollen von Client und Routern. Sie schicken spielerisch Informationen auf Karten von einem Schüler-Client zum anderen. Jede Schülerin und jeder Schüler hat eine Adresse, jeder Router darüber hinaus eine Routingtabelle. Mit Hilfe der Tabelle und einem Würfel wird entschieden, wie ein Paket weiter vermittelt wird.</p> <p>www.dnstools.ch/visual-traceroute</p> <p>Die Schüler recherchieren selbständig im Internet und in Bibliotheken</p>

Städt. Gymnasium an der Hönne, Menden: Curriculum für das Fach Informatik in der Sekundarstufe II

Jg	Thema und Unterrichtsvorhaben	Projekte, Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
	<p>eine kleine Geschichte der Digitalisierung: Vom Modem zum modernen Digitalcomputer</p> <p>Ein kleine Geschichte der Kryptografie: Von Caesar zur Enigma</p> <p>Von Nullen, Einsen und mehr: Stellenwertsysteme und wie man damit rechnet</p> <p>Kodieren von Texten und Bildern: ASCII, RGB und mehr</p> <p>Auswirkungen, Veränderung der Arbeitswelt und Datenschutzes</p> <p>Vertiefung des Themas Datenschutzes</p> <p>a) Erarbeitung grundlegender Begriffe und Prinzipien</p> <p>b) Problematisierung und Anknüpfung an die Wirklichkeit der Schüler, aktuelle politische Themen.</p> <p>Inhaltsfeld 5: Informatik, Mensch und Gesellschaft: Fahrt zu CeBIT</p>		<p>interpretieren Binärcodes als Zahlen und Zeichen (D)</p> <p>nutzen das Internet zur Recherche, zum Datenaustausch und zur Kommunikation (K)</p>	<p>Ausstellung zu informatischen Themen beim TOT</p> <p>Enigma-Simulationen im www</p> <p>Fallbeispiele aus dem aktuellen Tagesgeschehen</p>
<p>EF. 1</p> <p>EF. 2</p>	<p><u>Unterrichtsvorhaben E III):</u></p> <p>Grundlagen der objektorientierten Analyse, Modellierung und Implementierung anhand von statischen Grafikszenen</p> <p>Installation und Einführung in BlueJ,</p>		<p>Da die Projekte in Partner- und Gruppenarbeit verwirklicht werden sollen, finden sich allen Unterrichtsvorhaben die Kompetenzen Argumentieren, Kommunizieren und Kooperieren</p> <p>Werkzeuge: Umgang mit den Grundfunktionen der BlueJ-Umgebung</p> <p>Modellieren:</p>	<p>Buch Kap. 2 und 3</p>

Jg	Thema und Unterrichtsvorhaben	Projekte, Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
EF. 2	<p>Grundlagen der Programmierung in Java:</p> <p><u>Inhaltsfelder:</u> Formale Sprachen und Automaten, Algorithmen Daten und ihre Strukturierung Aufbau eines Programms, Grundlagen der Java-Syntax und Semantik Identifikation von Objekten Am Beispiel eines lebensweltnahen Beispiels werden Objekte im Sinne der Objektorientierten Programmierung (OOP) eingeführt Objekte werden mit Objektkarten visualisiert und mit sinnvollen Attributen und Methoden versehen Manche Objekte sind prinzipiell typgleich und werden somit zu einer Objektsorte (=Objektklasse) zusammengeführt Analyse von Klassen didaktischer Lernumgebungen OoP als modularisiertes Vorgehen (Entwicklung von Problemlösungen auf Grundlage vorhandener Klassen) Teilanalyse der Klassen der Lernumgebung SUM ((Bunt)Stift, Bildschirm, Maus) Implementierung statischer Szenen Grundaufbau einer JAVA-Klasse Deklaration und Initialisierung von Objekten Methodenaufrufe mit Parameterübergabe (Position, Winkel, Farbe)</p>	<p>Zeichnen von geometrischen Figuren („Haus vom Nikolaus“) Zeichenprojekte Buch Kap. 3</p>	<p>Ermitteln von Objekten, ihren Eigenschaften, Operationen und Beziehungen bei der Analyse einfacher Problemstellungen Modellieren von Klassen mit ihren Attributen, Methoden, Beziehungen Implementieren von einfachen Algorithmen unter der Beachtung von Syntax und Semantik einer Programmiersprache Verwenden einer Klassenbibliothek, Darstellen des Zustandes eines Objekts Implementieren - Programmieren: Darstellung in JAVA</p>	
EF. 2	<p><u>Unterrichtsvorhaben (E IV):</u> Grundlagen der objektorientierten</p>	<p>Freihandzeichnen Gummilinie Dartspiel</p>	<p>Modellieren/Problemlösen: Modellieren: Verwenden einer Klassenbibliothek, Struktogramm</p>	<p><u>Mindestanforderungen:</u> Einfache Algorithmen mit mindestens einer Schleife</p>

Städt. Gymnasium an der Hönne, Menden: Curriculum für das Fach Informatik in der Sekundarstufe II

Jg	Thema und Unterrichtsvorhaben	Projekte, Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
	<p>Objekten: kennt, hat, ist – Beziehungen eigene Klassen, Unterklassen, Abstrakte Klassen,</p> <p><u>Unterrichtsvorhaben (E VI):</u> Such- und Sortieralgorithmen anhand kontextbezogene Beispiele</p> <p>Inhaltsfelder: Algorithmen</p> <p>Sortieren im Alltag: Telefonbuch, Sporttabellen, Karten Tabellenkalkulation, Vergleich und Vertauschen zweier Elemente</p> <p>Problemstellung</p> <p>Iteratives und rekursives Sortieren und Suchen (Bubblesort, Teile und Sortiere, Auswählen)</p> <p>Struktogramme, Pseudocode Effizienzüberlegungen und erste Bewertungen, Binäre Suche</p>		<p>Argumentieren: Beurteilen von Effizienz der vorgestellten Algorithmen bzgl. Zeit und Speicherplatz.</p> <p>Modellieren: Entwurf von Algorithmen in Pseudocode</p> <p>Darstellen und Interpretieren: Analysieren und Anwenden auf Beispiele</p> <p>Kommunizieren und Kooperieren</p>	<p><u>Mindestanforderungen:</u> Unterschiede rekursiv, iterativ Erläutern je eines iterativen Algorithmus</p>
EF. 2	<p>Phasen eines Softwareprojektes</p> <p>Das folgende <u>Unterrichtsvorhaben</u> ist lt. Lernlehrplan nicht verpflichtend, aber sehr sinnvoll, um entsprechende Programmoberflächen und Programme in der Q1 zu generieren. Es ist damit die Schnittstelle zwischen EF und Q1 und kann</p>		<p>Werkzeuge: BlueJ-Programmgenerator</p> <p>Implementieren - Programmieren: Darstellen in JAVA</p> <p>Modellieren/Problemlösen: Phasenmodell, Pflichtenheft, Entwurfsphase, Implementierung, Modultest</p>	<p><u>Mindestanforderungen:</u> Programmgenerator für ein GUI anwenden. 2 Klausuren: 2 Std. in E.2</p>

Jg	Thema und Unterrichtsvorhaben	Projekte, Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
	auch zu Beginn der Q1 unterrichtet werden. Siehe Q1 I): Model-View-Controller Programmgenerator	Bruchrechner		
Jahrgang Q1: Das große Thema der Stufe Q1 sind Datenstrukturen und daraus hervorgehende Algorithmen				
Q1	<p><u>Unterrichtsvorhaben (Q1 I):</u> Wiederholung der objektorientierten Modellierung und Programmierung anhand einer kontextbezogenen Problemstellung: Aufbau eines Programms, Grundlagen der Java-Syntax und Semantik, Klassen und Objekte, Kontrollstrukturen (Schleifen, Verzweigungen)</p> <p><u>Inhaltsfelder:</u> Daten und ihre Strukturierung Algorithmen Formale Sprachen und Automaten Informatiksysteme</p> <p><u>inhaltliche Schwerpunkte:</u> Objekte und Klassen, Dokumentation Analyse der Problemstellung, Entwurf und Implementierung von Algorithmen (Implementationsdiagramm, Vererbung, abstrakte Klassen) Kommunikation zwischen mindestens zwei Objekten Algorithmen in ausgewählten Kontexten Syntax und Semantik einer Programmiersprache Nutzung von Informatiksystemen</p>	Projekt: Bruchrechnen (falls nicht in EF) Rekursive Folgen und Funktionen rekursives Zeichnen, Türme von Hanoi,	<p>Problemlösen: rekursives Verhalten als Lösungsansatz Modellieren: Klassendiagramm mit Attributen und Methoden, rekursive Lösungen im Struktogramm Argumentieren/Kommunizieren: Iterative ↔ rekursive Lösungen Speicherplatzüberlegungen Implementieren - Programmieren: Darstellen in JAVA</p>	<p><u>Mindestanforderungen:</u> Darstellen einfacher iterativer DO-, WHILE-Zählschleifen als rekursive Methode Darstellen einer einfachen rekursiven Methode im Struktogramm</p>
	<u>Unterrichtsvorhaben (Q1 II):</u> Modellierung und Implementierung von Anwendungen mit dynamischen, linearen Datenstrukturen:	Würfel mit zwei Würfeln Demokratur,	<p>Problemlösen/Modellieren: Felder als Lösungsansatz Implementieren - Programmieren:</p>	<p><u>Mindestanforderungen:</u> Verwalten einfacher Felder: Einfügen, Lesen,</p>

Jg	Thema und Unterrichtsvorhaben	Projekte, Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
	<p>Felder</p> <p><u>Inhaltsfelder:</u> Daten und ihre Strukturierung Algorithmen Formale Sprachen und Automaten Lineare Strukturen: Schlangen, Listen, Stapel</p> <p><u>inhaltliche Schwerpunkte:</u> Objekte und Klassen Analyse, Entwurf und Implementierung von Algorithmen Algorithmen in ausgewählten Kontexten Syntax und Semantik einer Programmiersprache</p>	<p>Wartezimmer, Militärkapelle,</p>	<p>Darstellen in JAVA Problemlösen/Modellieren: Lineare Strukturen als Lösungsansatz</p> <p>Werkzeuge: innere Klassen, eigene (lokale) Bibliotheken Implementieren - Programmieren: Darstellen in JAVA</p>	<p>Löschen von Elementen</p> <p><u>Mindestanforderungen:</u> grafische Darstellung der Listenoperationen, Umsetzung graphische Darstellung – Quellcode Benutzen einer vorgegebenen Bibliothek</p>
	<p><u>Unterrichtsvorhaben (Q1 III):</u> Suchen und Sortieren auf linearen Datenstrukturen</p> <p>Sortierverfahren: iterative und rekursive, Laufzeitverhalten</p> <p>Suchverfahren: Binäre Suche</p> <p><u>Inhaltsfelder:</u> Daten und ihre Strukturierung Algorithmen Formale Sprachen und Automaten Informatiksysteme</p> <p><u>inhaltliche Schwerpunkte:</u> Objekte und Klassen</p>	<p>Wettkampfliste</p> <p>Visitenkarten</p>	<p>Argumentieren/Kommunizieren: Bewerten verschiedener Sortieralgorithmen, iterativ – rekursiv Problemlösen/Modellieren: Werkzeuge: innere Klassen, eigene (lokale) Bibliotheken Implementieren - Programmieren: Darstellen in JAVA</p>	<p><u>Mindestanforderungen:</u> Bubblesort auf ein Feld zum Sortieren anwenden</p> <p>ein rekursives Sortierverfahren auf einem Feld deutlich machen</p>

Jg	Thema und Unterrichtsvorhaben	Projekte, Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
	<p>Analyse, Entwurf und Implementierung von Algorithmen Algorithmen in ausgewählten Kontexten Syntax und Semantik einer Programmiersprache Nutzung von Informatiksystemen</p> <p><u>Unterrichtsvorhaben (Q1 IV):(20 Std.)</u> Modellierung und Nutzung von relationalen Datenbanken im Anwendungskontexten 1.Nutzung von relationalen Datenbanken a)Aufbau von Datenbanken und Grundbegriffe Entwicklung von Fragestellungen zur vorhandenen Datenbank. Analyse der Struktur der vorgegebenen Datenbank und Erarbeitung der Begriffe Tabelle, Attribut, Datensatz, Datentyp, Primärschlüssel, Fremdschlüssel, Datenbankschema. b) SQL-Abfragen Analyse vorgegebene SQL-Abfragen und Erarbeitung der Sprachelemente von SQL (SELECT, DISTINCT ...FROM, WHERE,A;D, OR, NOT) auf einer Tabelle Analyse und Erarbeitung von SQL-Abfragen auf einer und mehreren Tabellen zur Beantwortung von Fragestellungen (JOIN, UNION, AS, GROUP BY, ORDER BY, ASC,DESC, COUNT, MAX, MIN, SUM, arithmetische Operatoren:+,-,*,/,... Vergleichsoperatoren, LIKE, BETWEEN, IN IS NULL) c) Vertiefung an einem weiteren Datenbankbeispiel</p>	<p>In der Literatur und im Internet finden sich verschiedene Projekte, von denen eins oder zwei genommen werden sollten:</p> <p>Beispieldatenbank aus dem eingeführten Lehrbuch Beispieldatenbank aus Throll/Bartisch: Einstieg in SQL, http://dokumentation.videocenter.schule.de/old/video/index.html www.brd.nrw.de/lerntreffs/informatik/structure/material/sek2/datenbanken.php (Schulbuchausleihe) http://mrbs.sourceforge.net => online Buchungssystem</p>	<p>Die Schülerinnen und Schüler: Argumentieren/Kommunizieren: erläutern die Eigenschaften und den Aufbau von Datenbanksystemen unter dem Aspekt der sicheren Nutzung analysieren und erläutern die Syntax und Semantik einer Datenbankabfrage analysieren und erläutern eine Datenbankmodellierung erläutern die Eigenschaften normalisierter Datenbankschemata Problemlösen/Modellieren: bestimmen Primär- und Sekundärschlüssel ermitteln für die Projekte Entitäten, zugehörige Attribute, Relationen und Kardinalitäten modellieren zu einem Entity-Relationship-Diagramm ein relationales Datenbankschema modifizieren eine Datenbankmodellierung überführen Datenbankschemata in vorgegebene Normalformen Werkzeuge: können mit einer Datenbanksoftware aus einem Office-Paket umgehen Implementieren - Programmieren: verwenden Syntax und Semantik einer Datenbankabfragesprache(SQL), um Informationen aus einem Datenbanksystem zu extrahieren</p>	<p>siehe links, Throll/Bartsch: Einstieg in SQL Materialien aus dem Internet Implementation einer Datenbank mit dem Datenbankmodul eines Office-Paketes: MS-Access, Libre-Office, Open Office</p>

Jg	Thema und Unterrichtsvorhaben	Projekte, Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
	<p>2. Modellierung von relationalen Datenbanken</p> <p>a) Entity-Relationship-Diagramm Ermittlung von Entitäten, zugehörigen Attributen, Relationen und Kardinalitäten in Anwendungssituationen und Modellierung eines Datenbankentwurfes in Form eines Entity-Relationship-Diagrammes Erläuterung und Modifizierung einer Datenbankmodellierung</p> <p>b) Entwicklung einer Datenbank aus einem Datenbankentwurf. Modellierung eines relationalen Datenbankschemas inklusive der Primär- und Sekundärschlüssel</p> <p>c) Redundanz, Konsistenz und Normalformen Untersuchung einer Datenbank hinsichtlich Konsistenz und Redundanz in einer Anwendungssituation Überprüfung von Datenbankschemata hinsichtlich der 1. bis 3. Normalform (um Redundanzen zu vermeiden und Konsistenz zu gewährleisten)</p> <p><u>Unterrichtsvorhaben Q1-V: Sicherheit und Datenschutz in Netzstrukturen (10 Std.)</u></p> <p>1. Daten in Netzwerken und Sicherheitsaspekte in Netzen sowie beim Zugriff auf Datenbanken.</p> <p>a) Beschreibung eines Datenbankzugriffs im Netz anhand eines Anwendungskontextes und einer Client-Server-Struktur zur Klärung des Funktionsweise eines Datenbankzugriffs</p> <p>b) Netztopologien als Grundlage von Client-Server-Strukturen und TCP/IP-Schichtenmodell</p>		<p>ermitteln Ergebnisse von Datenbankabfragen über mehrere verknüpfte Tabellen</p> <p>Darstellen: Stellen Entitäten mit ihren Attributen und die Beziehungen zwischen Entitäten in einem Entity-Relationship-Diagramm grafisch dar. Überprüfen Datenbankschemata auf vorgegebene Normalisierungseigenschaften</p> <p>Die Schülerinnen und Schüler: Argumentieren/Kommunizieren: beschreiben und erläutern Topologien, die Client-Server-Struktur und Protokolle, sowie ein Schichtenmodell in Netzwerken analysieren und erläutern Eigenschaften und Einsatzbereiche symmetrischer und asymmetrischer Verschlüsselungsverfahren untersuchen und bewerten anhand von Fallbeispielen die Auswirkungen des Einsatzes von Informatiksystemen sowie die Einhaltung der Datenschutzbestimmungen und des Urheberrechtes</p> <p>untersuchen und bewerten Problemlagen, die sich aus dem Einsatz von Informatiksystemen ergeben, hinsichtlich rechtlicher Vorgaben, ethischer Aspekte und gesellschaftlicher Werte unter der Berücksichtigung unterschiedlicher Interessenlagen</p> <p>Darstellen: nutzen bereitgestellte Informatiksysteme und das Internet reflektiert zum Erschließen, zur Aufbereitung und</p>	<p>Beispiele unter:</p> <p>http://bildungsserver.berlin-brandenburg.de/php_mysql.html</p> <p>http://bildungsserver.berlin-brandenburg.de/jahr1_inf.html</p> <p>http://www.brd.nrw.de/lerntreffs/informatik/structure/material/sek2/delphi/clientserver/clientserverDelphi.php</p> <p>http://www.brd.nrw.de/lerntreffs/informatik/structure/material/se</p>

Jg	Thema und Unterrichtsvorhaben	Projekte, Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
	<p>als Beispiel für eine Paketübermittlung in einem Netz.</p> <p>c) Vertraulichkeit, Integrität, Authentizität in Netzwerken sowie symmetrische und asymmetrische kryptografische Verfahren (Cäsar, Vigenere-, RSA- Verfahren) als Methoden Daten im Netz verschlüsselt zu übertragen</p> <p>2. Fallbeispiele zur Datenschutzproblematik und zum Urheberrecht.</p>		Präsentation fachlicher Inhalte	<p>k2/kryptologie/kryptologie.php</p> <p>eingeführtes Lehrbuch</p> <p>http://bildungsserver.berlin-brandenburg.de/iug_recht.html</p> <p>aktuelle Darstellungen und Fallbeispiele</p>
Halbjahr Q2 I: Thema Modellierung und Implementierung von Anwendungen mit dynamischen, nicht linearen Datenstrukturen				
Q2	<p><u>Unterrichtsvorhaben Q2 I:</u></p> <p>Modellierung und Implementierung von Anwendungen mit dynamischen, nichtlinearen Datenstrukturen</p> <p><u>Inhaltsfelder:</u></p> <p>Daten und ihre Strukturierung Algorithmen Formale Sprachen und Automaten</p> <p><u>inhaltliche Schwerpunkte:</u></p> <p>Objekte und Klassen Analyse, Entwurf und Implementierung von Algorithmen</p>	<p>Binäre Bäume:</p> <p>Beispiel: Termbaum</p>	<p><u>Nichtlineare Strukturen - Bäume:</u></p> <p><u>Die SuS</u></p> <p>Argumentieren: erläutern die Operationen dynamischer Datenstrukturen (linear und nichtlinear) analysieren und erläutern Algorithmen und Programme beurteile die syntaktischen Korrektheit und Funktionalität von Programmen</p> <p>Modellieren: ordnen Attribute, Parameter und Rückgaben von Methoden einfache Datentypen, Objekttypen sowie lineare und nichtlineare Datensammlungen zu. modellieren abstrakte und nicht abstrakte Klassen unter der Verwendung der Vererbung durch Spezialisieren und</p>	<p>Q2.1: 2 Klausuren: 3 Std.</p> <p>Q2..2: 1 Klausur unter Abiturbedingungen</p> <p>Ergänzungen im Lehrplannavigator Unterrichtsvorhaben Q2 I. Download Q2-I.1 (Termbaum)</p>

Jg	Thema und Unterrichtsvorhaben	Projekte, Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
	<p>Algorithmen in ausgewählten Kontexten Syntax und Semantik einer Programmiersprache</p> <p><u>1. Analyse von Baumstrukturen in verschiedenen Kontexten</u> (a) Grundlegende Begriffe: Grad, Tiefe, Höhe, Blatt, Inhalt, Teilbaum, Ebene, Vollständigkeit (b) Aufbau und Darstellung von binären Bäumen anhand von Baumstrukturen in verschiedenen Kontexten</p> <p><u>2. Die Klasse Binärbaum unter Nutzung der Klasse BinaryTree</u> (a) Analyse der Problemstellung, Ermittlung von Objekten, ihren Eigenschaften und Operationen</p>	<p>Der Aufbau von Termen wird mit Hilfe von binären Baumstrukturen verdeutlicht.</p> <p>Beispiel: Ahnenbaum Die binäre Baumstruktur ergibt sich daraus, dass jeder genau einen Vater und eine Mutter hat.</p> <p>Weitere Beispiele: Suchbäume zur binären Speicherung von Daten. Codierung durch das Morsealphabet Entscheidungsbäume, die nur ja – nein zulassen</p> <p>Informatikerbaum als binärer Baum: In einem binären Baum werden die Namen und die Geburtsdaten von Informatikern lexikografisch abgespeichert. Alle Namen, die nach dieser Ordnung vor dem Namen im aktuellen Teilbaum stehen, sind in dessen linken Teilbaum, alle die nach dem Namen stehen, im rechten Teilbaum (Dies gilt für alle Teilbäume) Folgende Funktionen werden benötigt: Einfügen der Daten in dem Baum. Suchen nach einem Informatiker über den Schlüssel NAME Ausgabe des kompletten Datenbestands in nach Namen sortierter Reihenfolge</p>	<p>Generalisieren verwenden bei der Modellierung geeigneter Problemstellungen die Möglichkeiten der Polymorphie entwickeln iterative und rekursive Algorithmen unter Nutzung der Konstruktionsstrategien „Modularisierung“ und „Teilen und Herrschen“.</p> <p>Implementieren: modifizieren Algorithmen und Programme nutzen die Syntax und Semantik einer Programmiersprache bei der Implementierung und zur Analyse von Programmen interpretieren Fehlermeldungen und korrigieren den Quellcode testen Programme systematisch anhand von Beispielen</p> <p>Darstellen: stellen lineare und nichtlineare Strukturen grafisch dar und erläutern ihren Aufbau stellen iterative und rekursive Algorithmen umgangssprachlich und grafisch dar.</p>	<p>Materialien: Ergänzungen zum Lehrplannavigator Unterrichtsvorhaben Q2.1 (Informatikerbaum) Download Q2-I.2</p>

Jg	Thema und Unterrichtsvorhaben	Projekte, Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
	<p>im Anwendungskontext</p> <p>(b) Modellierung eines Entwurfsdiagramms und Entwicklung eines Implementationsdiagramms</p> <p>(c) Erarbeitung der Klasse <i>Binary Tree</i> und beispielhafte Anwendung der Operationen</p> <p>(d) Implementierung der Anwendung oder von Teilen der Anwendung.</p> <p>(e) Traversierung eines Binärbaumes im Pre-, In- und Postorderdurchlauf</p> <p><u>3. Die Datenstruktur binärer Suchbaum im Anwendungskontext unter Verwendung der Klasse <i>BinarySearchTree</i></u></p> <p>(a) Analyse der Problemstellung, Ermittlung von Objekten, ihren Eigenschaften und Operationen im Anwendungskontext</p> <p>(b) Modellierung eines Entwurfsdiagramms und Entwicklung eines Implementationsdiagramms, grafische Darstellung eines binären Suchbaumes und Erarbeitung der Struktureigenschaften</p> <p>(c) Erarbeitung der Klasse <i>BinarySearchTree</i> und Einführung des Interface <i>Item</i> zur Realisation einer geeigneten Ordnungsrelation</p> <p>(d) Implementierung der Anwendung oder von Teilen der Anwendung inklusive der sortierten Ausgabe des Baumes</p> <p>4., Übung und Vertiefung der Verwendung von Binärbäumen oder binären Suchbäumen anhand weiterer Problemstellungen</p>	<p>Beispiele: Codierungsbäume oder Huffman-Codierung</p> <p>oder</p> <p>Buchindex: Es soll eine Anwendung entwickelt werden, die anhand von Stichworten und zugehörigen Seitenzahlen ein Stichwortregister erstellt. Da die Stichwörter bei der Analyse des Buches häufig gesucht werden müssen, werden sie in der Klasse <i>Buchindex</i> als <i>BinarySearchTree</i> verwaltet.</p> <p>Oder</p> <p>Entscheidungsbäume</p> <p>oder</p> <p>Termbaum</p> <p>Ahnenbaum</p>		<p>Materialien: Ergänzungen zum Lehrplannavigator Unterrichtsvorhaben Q2.1</p> <p>Download Q2-I.3 (Informatikerbaum als Suchbaum)</p> <p>Materialien: Ergänzungen zum Lehrplannavigator Unterrichtsvorhaben Q2.1</p> <p>Download Q2-I.4</p>

Jg	Thema und Unterrichtsvorhaben	Projekte, Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
	<p>Unterrichtsvorhaben (Q2 II): Endliche Automaten und formale Sprachen</p> <p><u>Inhaltsfelder:</u> Endliche Automaten und formale Sprachen</p> <p><u>inhaltliche Schwerpunkte:</u> Endliche Automaten Grammatiken regulärer Sprachen Möglichkeiten und Grenzen von Automaten und formalen Sprachen</p> <p><u>1. Endliche Automaten</u> (a) Vom Automaten in den Schülerinnen und Schülern bekannten Kontexten zur formalen Beschreibung eines endlichen Automaten (b) Untersuchung, Darstellung und Entwicklung endlicher Automaten</p> <p><u>2. Untersuchung und Entwicklung von Grammatiken regulärer Sprachen</u> (a) Erarbeitung des formalen Darstellung regulärer Grammatiken (b) Untersuchung, Modifikation und Entwicklung von Grammatiken (c.) Entwicklung von endlichen Automaten zum Erkennen regulärer Sprachen, die durch Grammatiken gegeben werden (d) Entwicklung regulärer Grammatiken zu endlichen Automaten</p> <p><u>3. Grenzen endlicher Automaten</u></p>	<p>Beispiele: Cola Automat, Geldspielautomat, Roboter, Zustandsänderungen eines Objektes „Auto“, Akzeptor für bestimmte Zahlen, Akzeptor für bestimmte Zahlen und Zeichen (ketten), Akzeptor für Terme</p> <p>Beispiele: reguläre Grammatik für Wörter mit ungerader Parität Grammatik für Wörter, die bestimmte Zahlen repräsentieren Satzgliedergrammatik</p>	<p>Die Schülerinnen und Schüler: Argumentieren und Kommunizieren: analysieren und erläutern die Eigenschaften endlicher Automaten einschließlich ihres Verhaltens auf bestimmte Eingaben analysieren und erläutern Grammatiken regulärer Sprachen zeigen die Grenzen endlicher Automaten und regulärer Grammatiken im Anwendungszusammenhang auf. Ermitteln die formale Sprache, die durch eine Grammatik erzeugt wird</p> <p>Modellieren und Darstellen: entwickeln und modifizieren zu einer Problemstellung endliche Automaten. Entwickeln zur akzeptierten Sprache eines Automaten, die zugehörige Grammatiken entwickeln, zur Grammatik einer regulären Sprache einen zugehörigen endlichen Automatentheorie modifizieren Grammatiken regulärer Sprachen entwickeln zu einer regulären Sprache eine Grammatik, die die Sprache erzeugt stellen endliche Automaten in Tabellen oder Graphen dar und überführen sie in die jeweils andere Darstellungsform ermitteln die Sprache, die ein endlicher Automat akzeptiert. Beschreiben an Beispielen den Zusammenhang zwischen Automaten und Grammatiken</p>	<p>Materialien: Ergänzungen zum Lehrplannavigator Unterrichtsvorhaben Q2.2 Download Q2-II.1</p> <p>Automatentheorie des Goethe-Gymnasiums www.do.nw.schule.de/.../InfoHomepage/Informatik/Automatentheorie.pdf</p> <p>Materialien: Ergänzungen zum Lehrplannavigator Unterrichtsvorhaben Q2.2 Download Q2-II.2</p> <p>Klammerausdrücke $a^n b^n$ im Vergleich zu $(ab)^n$</p>

Jg	Thema und Unterrichtsvorhaben	Projekte, Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
	<p><u>Unterrichtsvorhaben (Q2 III):</u> Prinzipielle Arbeitsweise eines Computers und Grenzen der Automatisierbarkeit</p> <p><u>Inhaltsfelder:</u> Informatik, Mensch und Gesellschaft Informatiksysteme</p> <p><u>inhaltliche Schwerpunkte:</u> Einzelrechner und Rechnernetzwerke Grenzen der Automatisierung</p> <p><u>1. Von-Neumann-Architektur und die Ausführung maschinennaher Programme</u></p> <p>(a) prinzipieller Aufbau einer von Neumann-Architektur mit CPU, Rechenwerk, Steuerwerk, Register und Hauptspeicher</p> <p>(b) einige maschinennahe Befehle und ihre Repräsentation in einem Binärcode, der in einem Register gespeichert werden kann.</p> <p>(c) Analyse und Erläuterung der Funktionsweise eines einfachen maschinennahen Programms</p> <p><u>2. Grenzen der Automatisierbarkeit</u></p> <p>(a) Vorstellung des Halteproblems</p> <p>(b) Unlösbarkeit des Halteproblems</p> <p>(c.) Beurteilung de Einsatzes von Informatiksystemen hinsichtlich prinzipieller Möglichkeiten und prinzipieller Grenzen</p>	<p>Addition von 4 zu einer eingegeben Zahl mit einem Rechnermodell</p> <p>Halteproblem</p>	<p>Die Schülerinnen und Schüler Argumentieren, Kommunizieren erläutern die Ausführung eines einfachen maschinennahen Programms sowie die Datenspeicherung auf einer „Von-Neumann-Architektur“ untersuchen und beurteilen Grenzen des Problemlösens mit Informatiksystemen</p>	<p>Materialien: Ergänzungen zum Lehrplannavigator Unterrichtsvorhaben Q2.2 Download Q2-III.1</p> <p>Materialien: Ergänzungen zum Lehrplannavigator Unterrichtsvorhaben Q2.2 Download Q2-III.2</p>
	<p>Außerschulische Lernorte: 1) TU-Dortmund</p>	<p>Aktuelle Angebote für Schüler und Schülerinnen, Schülertag Informatik</p>		

Städt. Gymnasium an der Hönne, Menden: Curriculum für das Fach Informatik in der Sekundarstufe II

Jg	Thema und Unterrichtsvorhaben	Projekte, Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
	2) Fahrt zur CEBIT nach Hannover (freiwillig)			