

Curriculum

Informatik

Einführungsphase EF

Reismann-Gymnasium Paderborn

1. Rahmenbedingungen

Diesem schulinternen Curriculum liegen der Lehrplan Informatik für die gymnasiale Oberstufe sowie die Vorgaben zu den unterrichtlichen Voraussetzungen für die schriftlichen Prüfungen im Abitur des Landes NRW zugrunde. Das Fach Informatik ist ein typisches Kooperationsfach mit anderen Paderborner Gymnasien.

2. Gestaltung und Inhalte des Unterrichts

Der Informatikunterricht ist durch eine Handlungs- und Produktorientierung gekennzeichnet, da sich die Schülerinnen und Schüler mit Problemstellungen auseinandersetzen und diese unter Anwendung von fachspezifischen Methoden wie der objektorientierten Modellierung und Programmierung systematisch lösen. Auf diesem Wege wird die Problemlösefähigkeit der Schülerinnen und Schüler verbessert. Es sollten soweit möglich reale Problemstellungen aus der Lebenswelt der Schülerinnen und Schüler aufgegriffen werden. Zudem wird das selbstständige Arbeiten und insbesondere im Rahmen von Projekten die Team-, Kommunikations- sowie die Kooperationsfähigkeit gefördert. Um eine reflektierte Auseinandersetzung mit dem Fach Informatik zu gewährleisten, müssen die Schülerinnen und Schüler dazu angeregt werden, ihre Problemlösungen und bestehende Informatiksysteme zu analysieren und zu bewerten. Dabei werden die Grenzen von Informatiksystemen und auch der verantwortungsvolle Umgang mit ihnen thematisiert.

Einführungsphase

Inhaltliche Schwerpunkte	Bezüge zur Obligatorik
Objektorientierte Modellierung <ul style="list-style-type: none">• Objekte, Klassen, Attribute, Methoden mit Rückgabe- und Übergabewerten• Beziehungen zwischen Objekten und Klassen• UML-Notation• Geheimnisprinzip	LP: Ein Informatikmodell gewinnen: Probleme eingrenzen und spezifizieren, reduzierte Systeme definieren ZA: I.1 Konzepte des objektorientierten Modellierens
Objektorientierte Programmierung <ul style="list-style-type: none">• Programmierkonventionen• elementare Datentypen• Objektreferenzen• Variablen-Konzept• Konstanten• Kontrollstrukturen• Arrays	LP: Lösungen nach einem Programmierkonzept realisieren, überprüfen und weiterentwickeln ZA: I.1 Konzepte des objektorientierten Modellierens

Übersichtsraster Unterrichtsvorhaben

Allgemeines

Der Informatikunterricht in der Sekundarstufe II orientiert sich am Kernlehrplan für die Sek. II des Landes NRW, Endfassung lt. dErl. v. 04.09.2013 gültig ab 01.08.2014.

Der Grundkurs der Einführungsphase bereitet den an den Vorgaben für das Zentralabitur in NRW ausgerichteten Unterricht der Qualifikationsphase vor. Die hier aufgereihten Inhalte und Kompetenzen legen in ihrer Auflistung keine obligatorische Reihenfolge für ihre unterrichtliche Behandlung fest. Im Gegenteil, es ist eher von einer Verzahnung der Themen und einem spiralförmigem Kompetenzerwerb auszugehen.

Eingeführtes Lehrbuch: Thomas Kempe, Annika Löhr: Informatik, Schöningh-Verlag.

Einführungsphase	
<p><u>Unterrichtsvorhaben E-I</u></p> <p>Thema: <i>Einführung in die Nutzung von Informatiksystemen und in grundlegende Begrifflichkeiten</i></p> <p>Zentrale Kompetenzen:</p> <ul style="list-style-type: none"> • Argumentieren • Darstellen und Interpretieren • Kommunizieren und Kooperieren <p>Inhaltsfelder:</p> <ul style="list-style-type: none"> • Informatiksysteme • Informatik, Mensch und Gesellschaft <p>Inhaltliche Schwerpunkte:</p> <ul style="list-style-type: none"> • Einzelrechner • Dateisystem • Internet • Einsatz von Informatiksystemen <p>Zeitbedarf: 4 Stunden</p>	<p><u>Unterrichtsvorhaben E-II</u></p> <p>Thema: <i>Grundlagen der objektorientierten Analyse, Modellierung</i></p> <p>Zentrale Kompetenzen:</p> <ul style="list-style-type: none"> • Modellieren • Darstellen und Interpretieren <p>Inhaltsfelder:</p> <ul style="list-style-type: none"> • Daten und ihre Strukturierung <p>Inhaltliche Schwerpunkte:</p> <ul style="list-style-type: none"> • Objekte und Klassen <p>Zeitbedarf: 6 Stunden</p>

Einführungsphase

Unterrichtsvorhaben E-III

Thema:

Grundlagen der objektorientierten Programmierung und algorithmischer Grundstrukturen in Java

Zentrale Kompetenzen:

- Argumentieren
- Modellieren
- Implementieren

Inhaltsfelder:

- Daten und ihre Strukturierung
- Algorithmen

Inhaltliche Schwerpunkte:

- Objekte und Klassen
- Syntax und Semantik einer Programmiersprache
- Analyse, Entwurf und Implementierung einfacher Algorithmen

Zeitbedarf: 18 Stunden

Unterrichtsvorhaben E-IV

Thema:

Modellierung und Implementierung von Klassen- und Objektbeziehungen

Zentrale Kompetenzen:

- Argumentieren
- Modellieren
- Implementieren
- Darstellen und Interpretieren

Inhaltsfelder:

- Daten und ihre Strukturierung
- Algorithmen
- Formale Sprachen und Automaten

Inhaltliche Schwerpunkte:

- Objekte und Klassen
- Syntax und Semantik einer Programmiersprache
- Analyse, Entwurf und Implementierung einfacher Algorithmen

Zeitbedarf: 18 Stunden

Einführungsphase	
<p><u>Unterrichtsvorhaben E-V</u></p> <p>Thema: <i>Such- und Sortieralgorithmen</i></p> <p>Zentrale Kompetenzen:</p> <ul style="list-style-type: none"> • Argumentieren • Modellieren • Darstellen und Interpretieren <p>Inhaltsfelder:</p> <ul style="list-style-type: none"> • Algorithmen <p>Inhaltliche Schwerpunkte:</p> <ul style="list-style-type: none"> • Algorithmen zum Suchen und Sortieren • Analyse, Entwurf und Implementierung einfacher Algorithmen <p>Zeitbedarf: 12 Stunden</p>	<p><u>Unterrichtsvorhaben E-VI</u></p> <p>Thema: <i>Geschichte der automatischen Datenverarbeitung</i></p> <p>Zentrale Kompetenzen:</p> <ul style="list-style-type: none"> • Argumentieren • Darstellen und Interpretieren <p>Inhaltsfelder:</p> <ul style="list-style-type: none"> • Informatik, Mensch und Gesellschaft • Informatiksysteme <p>Inhaltliche Schwerpunkte:</p> <ul style="list-style-type: none"> • Einzelrechner • Geschichte der automatischen Datenverarbeitung • Digitalisierung <p>Zeitbedarf: 16 Stunden</p>
Summe Einführungsphase: 74	

I) Einführungsphase

Die folgenden Kompetenzen aus dem Bereich *Kommunizieren und Kooperieren* werden in allen Unterrichtsvorhaben der Einführungsphase vertieft und sollen aus Gründen der Lesbarkeit nicht in jedem Unterrichtsvorhaben separat aufgeführt werden:

Die Schülerinnen und Schüler

- verwenden Fachausdrücke bei der Kommunikation über informatische Sachverhalte (K),
- präsentieren Arbeitsabläufe und -ergebnisse (K),
- kommunizieren und kooperieren in Gruppen und in Partnerarbeit (K),
- nutzen das verfügbare Informatiksystem zur strukturierten Verwaltung und gemeinsamen Verwendung von Daten unter Berücksichtigung der Rechteverwaltung (K).

Unterrichtsvorhaben EF-I

Thema: Einführung in die Nutzung von Informatiksystemen und in grundlegende Begrifflichkeiten

Leitfragen: Womit beschäftigt sich die Wissenschaft der Informatik? Was sind die Fragestellungen und Meilensteine der Informatik? Wie kann die in der Schule vorhandene informatische Ausstattung genutzt werden?

Vorhabenbezogene Konkretisierung:

Das erste Unterrichtsvorhaben stellt eine allgemeine Einführung in das Fach Informatik und seine Teilbereiche dar. Dabei ist zu berücksichtigen, dass für manche Schülerinnen und Schüler in der Einführungsphase der erste Kontakt mit dem Unterrichtsfach Informatik stattfindet, so dass zu Beginn Grundlagen des Fachs behandelt werden müssen.

Die Schülerinnen und Schüler sollen einen Überblick über die unterschiedlichen fachwissenschaftlichen Fragestellungen sowie die Inhaltsfelder des Informatikunterrichts der Sek II erlangen.

Ferner wird eine Einführung in die Nutzung der konkreten Recherausstattung der Schule sowie in die Strukturen ihrer Vernetzung gegeben. Dabei wird insbesondere Wert auf einen verantwortungsvollen Umgang mit diesem System gelegt.

Zeitbedarf: 4 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
<p>1. Teilbereiche der Informatik</p> <p>(a) praktische, theoretische, technische Informatik sowie Informatik und Gesellschaft</p> <p>(b) Vereinbarung von Richtlinien zur Datenspeicherung auf den Schulrechnern (z.B. Ordnerstruktur, Dateibezeichner usw.)</p>	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> • erkennen und beschreiben die Teilbereiche der Informatik sowie ihre typischen Problemstellungen (D), • bewerten anhand von Fallbeispielen die Auswirkungen des Einsatzes von Informatiksystemen (A), • nutzen die im Unterricht eingesetzten Informatiksysteme selbstständig, sicher, zielführend und verantwortungsbewusst (D), 	<p>Lehrbuch, Präsentation</p>
<p>2. Einführung in die Infrastruktur der Schule</p> <p>(a) Informatische Kommunikation in Rechnernetzen am Beispiel des Schulnetzwerks (z.B. Benutzeranmeldung, Netzwerkordner, Zugriffsrechte, Client-Server)</p> <p>(b) Richtlinien zum verantwortungsvollen Umgang mit dem Internet</p>	<ul style="list-style-type: none"> • nutzen das Internet zur Recherche, zum Datenaustausch und zur Kommunikation (K), • nutzen die im Unterricht eingesetzten Informatiksysteme selbstständig, sicher, zielführend und verantwortungsbewusst (D). 	<p>Rechnerausstattung der Schule</p> <p>Lernstatt, Gruppenordner, Moodle, Zarafa</p> <p>Intranet, vpn-Zugriff</p>

Unterrichtsvorhaben EF-II

Thema: Grundlagen der objektorientierten Analyse, Modellierung

Leitfrage: *Was sind Klassen/Objekte? Wie werden sie durch Eigenschaften und Fähigkeiten charakterisiert?*

Vorhabenbezogene Konkretisierung:

Ein zentraler Bestandteil des Informatikunterrichts der Einführungsphase ist die Objektorientierte Programmierung. Dieses Unterrichtsvorhaben führt in die Grundlagen der Analyse und Modellierung in diesem Kontext ein.

Die Schülerinnen und Schüler sollen Klassen und Objekte aus ihrem realen Umfeld beschreiben, anwendungsbezogen reduzieren und in UML-Syntax darstellen. Sie erkunden in einer didaktischen Lernumgebung (z.B. Greenfoot) Objekte mit ihrer Identität und ihrem Zustand sowie Klassen mit ihren Attributen und Methoden.

Zeitbedarf: 6 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
<p>1. Identifikation von Objekten, Modellierung und Beschreibung von Klassen und Objekten</p> <p>(a) am Beispiel eines lebensweltnahen Beispiels werden Objekte im Sinne der objektorientierten Modellierung eingeführt. (Auto, Tiere, Konto...)</p> <p>(b) in einer didaktischen Lernumgebung (z.B. Greenfoot)</p> <p>2. Vererbung</p> <p>(a) Das Grundprinzip der Vererbung</p> <p>(b) Vorteile der Vererbungsbeziehung</p>	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none">• ermitteln bei der Analyse einfacher Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M),• modellieren Klassen mit ihren Attributen, ihren Methoden und Assoziationsbeziehungen (M),• analysieren und erläutern objektorientierte Modellierungen (A),• dokumentieren Klassen, indem sie die Funktionalität der Attribute und Methoden beschreiben.	<p>Lehrbuch, didaktische Programmierumgebung Greenfoot</p> <p>Objektdiagramme, Klassendiagramme, Entwurfsdiagramme</p>

Unterrichtsvorhaben EF-III

Thema: Grundlagen der objektorientierten Programmierung und algorithmischer Grundstrukturen in Java anhand von einfachen Animationen

Leitfragen: *Wie lassen sich Animationen von Objekten unter Berücksichtigung von Tastatureingaben realisieren?*

Vorhabenbezogene Konkretisierung:

Der Schwerpunkt dieses Unterrichtsvorhabens liegt auf der Entwicklung Erarbeitung von Kontrollstrukturen. Algorithmische Wiederholung und die bedingte Anweisung werden an einfachen Beispielen eingeführt und anschließend anhand komplexer Problemstellungen implementiert.

Da die zu entwickelnden Algorithmen zunehmend umfangreicher werden, werden systematische Vorgehensweisen zur Entwicklung von Algorithmen thematisiert.

Ein zweiter Schwerpunkt liegt auf dem Einsatz von Variablen. Zunächst werden lokale Variablen, die in Methoden und Zählschleifen zum Einsatz kommen, anschließend Variablen in Form von Parametern und Rückgabewerten von Methoden behandelt.

Komplexere Assoziationsbeziehungen zwischen Klassen werden in diesem Unterrichtsvorhaben zunächst nicht behandelt. Sie stellen den Schwerpunkt des folgenden Vorhabens dar.

Zeitbedarf: 18 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
<p>1. Algorithmen</p> <p>(a) Wiederholungsanweisungen (vorprüfende und nachprüfende Schleifen)</p> <p>(b) bedingte Anweisungen</p> <p>(c) Verknüpfung von Bedingungen durch logische Operatoren AND, OR, NOT</p> <p>(d) Systematisierung des Vorgehens zur Entwicklung von Algorithmen zur Lösung komplexer Probleme.</p>	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> • analysieren und erläutern einfache Algorithmen und Programme (A), • entwerfen einfache Algorithmen und stellen sie umgangssprachlich und grafisch dar (M), • ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen zu (M), • ordnen Klassen, Attributen und Methoden ihren Sichtbarkeitsbereich zu (M), • modifizieren einfache Algorithmen und Programme (I), • implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I), • implementieren Algorithmen unter Verwendung von Variablen und Wertzuweisungen, Kontrollstrukturen sowie Methodenaufrufen (I), • implementieren einfache Algorithmen unter Beachtung der Syntax und Semantik einer Programmiersprache (I), • testen Programme schrittweise anhand von Beispielen (I), • interpretieren Fehlermeldungen und korrigieren den Quellcode (I). 	<p>Lehrbuch, Arbeitsblätter</p> <p>Greenfoot</p>
<p>2. Variablen und Methoden</p> <p>(a) Implementierung eigener Methoden mit lokalen Variablen, auch zur Realisierung einer Zählschleife (for)</p> <p>(b) Implementierung eigener Methoden mit Parameterübergabe und/oder Rückgabewert</p> <p>(c) Realisierung von Attributen</p>		

Unterrichtsvorhaben EF-IV

Thema: Modellierung und Implementierung von Klassen- und Objektbeziehungen

Leitfrage: Wie werden realistische Systeme anforderungsspezifisch reduziert, als Entwurf modelliert und implementiert? Wie kommunizieren Objekte und wie wird dieses dargestellt und realisiert?

Vorhabenbezogene Konkretisierung

Der Schwerpunkt dieses Unterrichtsvorhabens liegt auf der Entwicklung von Objekt- und Klassenbeziehungen. Dazu werden, ausgehend von realen Anforderungsdefinitionen, kleine Softwareprodukte in Teilen oder ganzheitlich Schritt für Schritt (Objektidentifizierung, Entwurf, Implementation) erstellt. Ausgehend von der Dekonstruktion und Erweiterung eines vorgegebenen Projektes wird ein weiteres Projekt von Grund auf modelliert und implementiert. Dabei können arbeitsteilige Vorgehensweisen zum Einsatz kommen. In diesem Zusammenhang kann auch das Erstellen von graphischen Benutzeroberflächen eingeführt werden.

Zeitbedarf: 18 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
3. Analyse von Objekten und Klassen (a) Aus Anforderungsbeschreibungen werden Objekte mit ihren Eigenschaften identifiziert. (b) Schritte der objektorientierten Analyse, Modellierung und Implementation (c) Analyse und Erprobung von vorgegebenen Implementationen (d) Analyse und Dekonstruktion eines Softwareprojektes z.B. eines Spiels (Modelle oder Quelltext)	Die Schülerinnen und Schüler <ul style="list-style-type: none">• analysieren und erläutern eine objektorientierte Modellierung, einfache Algorithmen und Programme (A),• stellen die Kommunikation zwischen Objekten grafisch dar (M),• ermitteln bei der Analyse einfacher Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M),• modellieren Klassen mit ihren Attributen, ihren Methoden und Assoziationsbeziehungen (M),• entwerfen einfache Algorithmen und stellen sie umgangssprachlich und grafisch dar (M),• ordnen Attributen, Parametern und Rückgaben von Me-	Lehrbuch, Arbeitsblätter, Greenfoot, BlueJ, Java-Editor o.a. Teilimplementierte Softwareprojekte (Verflixte7, Würfelspiel)
4. Implementationsdigagramme als erster Schritt der Programmierung (a) Erweiterung des Entwurfsdiagramms um		

<p>Konstruktoren und get-, set-Methoden</p> <p>(b) Festlegung von Datentypen in Java, sowie von Rückgaben und Parametern</p> <p>(c) Dokumentation der Klassen des Projekts</p> <p>(d) Erstellung von Sequenzdiagrammen als Vorbereitung für die Implementierung</p> <p>(e) Modellierung eines eigenen Softwareprojekts aufgrund einer Anforderungsbeschreibung</p>	<p>thoden einfache Datentypen oder Objekttypen zu (M),</p> <ul style="list-style-type: none"> • ordnen Klassen, Attributen und Methoden ihren Sichtbarkeitsbereich zu (M), • modellieren Klassen unter Verwendung von Vererbung (M), • implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I), • testen Programme schrittweise anhand von Beispielen (I), • interpretieren Fehlermeldungen und korrigieren den Quellcode (I), • modifizieren einfache Algorithmen und Programme (I), • stellen Klassen, Assoziations- und Vererbungsbeziehungen in Diagrammen grafisch dar (D), • dokumentieren Klassen durch Beschreibung der Funktionalität der Methoden (D). 	
<p>5. Implementation von Klassen</p> <p>(a) Analyse und Erläuterung einer Java-Klasse</p> <p>(b) Implementation eigener Klassen mit Attributen und Methoden auf Basis zuvor erarbeiteter Dokumentationen.</p> <p>(c) Umsetzung des Geheimnisprinzips</p> <p>(d) Instanziierung von Objekten</p> <p>(e) Einzelne Klassen und das Gesamtsystem werden anhand der Anforderungen und Dokumentationen auf ihre Korrektheit überprüft.</p> <p>(f) Erweiterung des gegebenen Softwareprojekts und weitere Funktionalität</p> <p>(g) Implementation des eigenen Softwareprojekts auf Basis der Modellierung, ggf. einer grafischen Oberfläche (ggf. arbeitsteilig)</p>		

Unterrichtsvorhaben EF-V

Thema: Such- und Sortieralgorithmen

Leitfragen: *Wie können Objekte bzw. Daten effizient sortiert und gesucht werden?*

Vorhabenbezogene Konkretisierung:

Zunächst lernen die Schülerinnen und Schüler das Feld als eine erste Datensammlung kennen. Dann steht die Erarbeitung von einfachen Such und Sortieralgorithmen im Vordergrund. Der Schwerpunkt liegt dabei auf den Algorithmen selbst und nicht auf deren Implementierung, die höchstens exemplarisch erstellt werden soll. Erste Effizienzbetrachtungen sollen intuitiv vorgenommen und bewertet werden.

Zeitbedarf: 12 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Modellierung und Implementation von Datensammlungen (a) Modellierung von Attributen als Felder (b) Deklaration, Instanzierung und Zugriffe auf ein Feld	Die Schülerinnen und Schüler <ul style="list-style-type: none">• beurteilen die Effizienz von Algorithmen am Beispiel von Sortierverfahren hinsichtlich Zeit und Speicherplatzbedarf (A),• entwerfen einen oder mehrere Algorithmen zum Suchen und Sortieren (M),• analysieren Such- und Sortieralgorithmen und wenden sie auf Beispiele an (D),• ordnen Attributen lineare Datensammlungen zu (M),• implementieren exemplarisch Such- und Sortieralgorithmen (I)	Lehrbuch, Arbeitsblätter BlueJ, Java-Editor, Eclipse Spielkarten o.a. Gegenstände zum Sortieren/Suchen Beispiele: Sortieren durch Auswählen, durch Einfügen, durch Vertauschen
2. Explorative Erarbeitung von Such- und Sortierverfahren und ihre exemplarische Implementation (a) Erkundung von Strategien für das Suchen auf unsortierten und sortierten Daten. (b) Erkundung von Strategien für das Sortieren von Daten (c) Vergleich der Verfahren durch intuitive Effizienzbetrachtungen (d) Exemplarische Implementation von		

Such- und Sortieralgorithmen		
3. Systematisierung von Algorithmen und Effizienzbetrachtungen (a) Umgangssprachliche Formulierung (falls selbst gefunden) oder Erläuterung von vorgegebenen Algorithmen (b) Anwendung von Sortieralgorithmen auf verschiedene Beispiele (c) Bewertung von Algorithmen anhand der Anzahl der nötigen Vergleiche und erste Effizienzbetrachtungen bezüglich der Rechenzeit und des Speicherplatzbedarfes		

Unterrichtsvorhaben EF-VI

Thema: Geschichte der automatischen Datenverarbeitung

Leitfrage: *Wie werden Daten codiert? Wie werden sie grundsätzlich verarbeitet und gespeichert? Wie ist die geschichtliche Entwicklung der EDV?*

Vorhabenbezogene Konkretisierung:

Die Schülerinnen und Schüler stellen bekannte analoge und digitale technische Systeme (Fernseher, Kommunikation, Musik, Fotografie, ...) gegenüber. In diesem Zusammenhang soll eine geschichtliche Entwicklung der automatischen Datenverarbeitung erarbeitet werden (ab 20. Jhd.). Das Binärsystem als Abbildung von zwei Zuständen – an und aus – soll als grundlegendes Zahlensystem in der Datenverarbeitung erkannt werden. Des Weiteren soll der grundlegende Aufbau eines Rechnersystems im Sinne der Von-Neumann-Architektur erarbeitet werden.

Zeitbedarf: 16 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Gegenüberstellung von analogen und digitalen Systemen im geschichtlichen Kontext.	Die Schülerinnen und Schüler <ul style="list-style-type: none">• erläutern wesentliche Grundlagen der Geschichte der digitalen Datenverarbeitung (A)• stellen ganze Zahlen und Zeichen in Binärcodes dar (D),• interpretieren Binärcodes als Zahlen und Zeichen (D),• beschreiben und erläutern den strukturellen Aufbau und die Arbeitsweise singulärer Rechner am Beispiel der „Von-Neumann-Architektur“ (A)	Lehrbuch: Exkurs Referate
2. Binäre Kodierung und ihre Anwendung (z.B. ASCII, UNICODE...)		Lehrbuch: Exkurs
3. Aufbau informatischer Systeme (a) Identifikation typischer Komponenten informatischer Systeme, Beschränkung auf das Wesentliche, Herleitung der Von-Neumann-Architektur (b) EVA als Prinzip der Verarbeitung von Daten und als Grundlage der Von-Neumann-Architektur		Know-How-Computer