

# Schulinternes Curriculum für Informatik (Q1) Stand April 2015

## Qualifikationsphase 1 (Teil 1)

### Unterrichtsvorhaben Q1-I

**Thema:** *Wiederholung der objektorientierten Modellierung und Programmierung anhand einer kontextbezogenen Problemstellung*

**Zentrale Kompetenzen:**

- Argumentieren
- Modellieren
- Implementieren
- Darstellen und Interpretieren
- Kommunizieren und Kooperieren

**Inhaltsfelder:**

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

**Inhaltliche Schwerpunkte:**

- Objekte und Klassen
- Analyse, Entwurf und Implementierung von Algorithmen
- Syntax und Semantik einer Programmiersprache
- Nutzung von Informatiksystemen

**Zeitbedarf:** 8 Stunden

### Unterrichtsvorhaben Q1-II

**Thema:** *Modellierung und Implementierung von Anwendungen mit dynamischen, linearen Datenstrukturen*

**Zentrale Kompetenzen:**

- Argumentieren
- Modellieren
- Implementieren
- Darstellen und Interpretieren
- Kommunizieren und Kooperieren

**Inhaltsfelder:**

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

**Inhaltliche Schwerpunkte:**

- Objekte und Klassen
- Analyse, Entwurf und Implementierung von Algorithmen
- Algorithmen in ausgewählten informatischen Kontexten
- Syntax und Semantik einer Programmiersprache

**Zeitbedarf:** 20 Stunden

## Qualifikationsphase 1 (Teil 2)

### Unterrichtsvorhaben Q1-III

**Thema:** Suchen und Sortieren auf linearen Datenstrukturen

**Zentrale Kompetenzen:**

- Argumentieren
- Modellieren
- Implementieren
- Darstellen und Interpretieren
- Kommunizieren und Kooperieren

**Inhaltsfelder:**

- Algorithmen
- Formale Sprachen und Automaten

**Inhaltliche Schwerpunkte:**

- Analyse, Entwurf und Implementierung von Algorithmen
- Algorithmen in ausgewählten informatischen Kontexten
- Syntax und Semantik einer Programmiersprache

**Zeitbedarf:** 16 Stunden

### Unterrichtsvorhaben Q1-IV

**Thema:** Modellierung und Nutzung von relationalen Datenbanken in Anwendungskontexten

**Zentrale Kompetenzen:**

- Argumentieren
- Modellieren
- Implementieren
- Darstellen und Interpretieren
- Kommunizieren und Kooperieren

**Inhaltsfelder:**

- Daten und ihre Strukturierung
- Algorithmen
- Formale Sprachen und Automaten
- Informatik, Mensch und Gesellschaft

**Inhaltliche Schwerpunkte:**

- Datenbanken
- Algorithmen in ausgewählten informatischen Kontexten
- Syntax und Semantik einer Programmiersprache
- Sicherheit

**Zeitbedarf:** 21 Stunden

### Unterrichtsvorhaben Q1-V

**Thema:**

*Sicherheit und Datenschutz in Netzstrukturen*

**Zentrale Kompetenzen:**

- Argumentieren
- Darstellen und Interpretieren
- Kommunizieren und Kooperieren

**Inhaltsfelder:**

- Informatiksysteme
- Informatik, Mensch und Gesellschaft

**Inhaltliche Schwerpunkte:**

- Einzelrechner und Rechnernetzwerke
- Sicherheit
- Nutzung von Informatiksystemen, Wirkungen der Automatisierung

**Zeitbedarf:** 10 Stunden

**Summe Qualifikationsphase 1: 75**

## Unterrichtsvorhaben Q1-I: Wiederholung der objektorientierten Modellierung und Programmierung

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
<p><b>1. Wiederholung und Erweiterung der objektorientierten Modellierung und Programmierung durch Analyse und Erweiterung eines kontextbezogenen Beispiels</b></p> <p>(a) Analyse der Problemstellung</p> <p>(b) Analyse der Modellierung (Implementationsdiagramm)</p> <p>(c) Erweiterung der Modellierung im Implementationsdiagramm (Vererbung, abstrakte Klasse)</p> <p>(d) Kommunikation zwischen mindestens zwei Objekten (grafische Darstellung)</p> <p>(e) Dokumentation von Klassen</p> <p>(f) Implementierung der Anwendung oder von Teilen der Anwendung</p>	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> <li>• analysieren und erläutern objektorientierte Modellierungen (A),</li> <li>• beurteilen die syntaktische Korrektheit und die Funktionalität von Programmen (A),</li> <li>• modellieren Klassen mit ihren Attributen, Methoden und ihren Assoziationsbeziehungen unter Angabe von Multiplizitäten (M),</li> <li>• ordnen Klassen, Attributen und Methoden ihre Sichtbarkeitsbereiche zu (M),</li> <li>• modellieren abstrakte und nicht abstrakte Klassen unter Verwendung von Vererbung durch Spezialisieren und Generalisieren (M),</li> <li>• implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I),</li> <li>• nutzen die Syntax und Semantik einer Programmiersprache bei der Implementierung und zur Analyse von Programmen (I),</li> <li>• wenden eine didaktisch orientierte Entwicklungsumgebung zur Demonstration, zum Entwurf, zur Implementierung und zum Test von Informatiksystemen an (I),</li> <li>• interpretieren Fehlermeldungen und korrigieren den Quellcode (I),</li> <li>• stellen Klassen und ihre Beziehungen in Diagrammen grafisch dar (D),</li> <li>• dokumentieren Klassen (D),</li> <li>• stellen die Kommunikation zwischen Objekten grafisch dar (D).</li> </ul>	<p><i>Beispiele:</i></p> <p>- Krankenhaus und (oder) Schachspiel</p> <p><i>Materialien:</i></p> <p>Informatik 3 – Lehrwerk für Gymnasien (Ernst Klett Verlag)</p>

## Unterrichtsvorhaben Q1-II: Modellierung und Implementierung von Anwendungen mit dynamischen, linearen Datenstrukturen

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
<p><b>1. Die Datenstruktur Schlange im Anwendungskontext unter Nutzung der Klasse Queue</b></p> <p>(a) Analyse der Problemstellung, Ermittlung von Objekten, ihren Eigenschaften und Operationen</p> <p>(b) Erarbeitung der Funktionalität der Klasse Queue</p> <p>(c) Modellierung und Implementierung der Anwendung unter Verwendung eines oder mehrerer Objekte der Klasse Queue</p>	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> <li>• erläutern Operationen dynamischer (linearer oder nicht-linearer) Datenstrukturen (A),</li> <li>• analysieren und erläutern Algorithmen und Programme (A),</li> <li>• beurteilen die syntaktische Korrektheit und die Funktionalität von Programmen (A),</li> <li>• ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen sowie lineare und nichtlineare Datensammlungen zu (M),</li> <li>• ermitteln bei der Analyse von Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M),</li> <li>• modifizieren Algorithmen und Programme (I),</li> <li>• implementieren iterative und rekursive Algorithmen auch unter Verwendung von dynamischen Datenstrukturen (I),</li> <li>• nutzen die Syntax und Semantik einer Programmiersprache bei der Implementierung und zur Analyse von Programmen (I),</li> <li>• interpretieren Fehlermeldungen und korrigieren den Quellcode (I),</li> </ul>	<p><i>Beispiele:</i></p> <ul style="list-style-type: none"> <li>- Taxischlange</li> <li>- Schlange vor der Kasse</li> <li>- Patientenwarteschlange</li> </ul> <p><i>Materialien:</i></p> <ul style="list-style-type: none"> <li>- Informatik 4 – Lehrwerk für Gymnasien (Ernst Klett Verlag)</li> <li>- Eigenes Skript</li> </ul>
<p><b>2. Die Datenstruktur Stapel im Anwendungskontext unter Nutzung der Klasse Stack</b></p> <p>(a) Analyse der Problemstellung, Ermittlung von Objekten, ihren Eigenschaften und Operationen</p> <p>(b) Erarbeitung der Funktionalität der Klasse Stack</p> <p>(c) Modellierung und Implementierung der Anwendung unter Verwendung eines oder mehrerer Objekte der Klasse Stack</p>	<ul style="list-style-type: none"> <li>• implementieren iterative und rekursive Algorithmen auch unter Verwendung von dynamischen Datenstrukturen (I),</li> <li>• nutzen die Syntax und Semantik einer Programmiersprache bei der Implementierung und zur Analyse von Programmen (I),</li> <li>• interpretieren Fehlermeldungen und korrigieren den Quellcode (I),</li> </ul>	<p><i>Beispiel:</i></p> <ul style="list-style-type: none"> <li>- Klausurstapel auf dem Schreibtisch des Lehrers (Vorgänge bei der Korrektur einer Klausur)</li> <li>- Kisten stapeln ( In einem Stapel nummerierter Kisten soll eine bestimmte Kiste gefunden und an einen Kunden geliefert werden.</li> </ul>
<p><b>3. Die Datenstruktur lineare Liste im Anwendungskontext unter Nutzung der Klasse List</b></p> <p>(a) Erarbeitung der Vorteile der Klasse List im Gegensatz zu den bereits bekannten Strukturen</p> <p>(b) Modellierung und Implementierung einer kontextbezogenen Anwendung unter Verwendung der Klasse List.</p>	<ul style="list-style-type: none"> <li>• testen Programme systematisch anhand von Beispielen (I),</li> <li>• stellen lineare und nichtlineare Strukturen grafisch dar und erläutern ihren Aufbau (D).</li> </ul>	<p><i>Beispiele:</i></p> <ul style="list-style-type: none"> <li>- Girlande</li> <li>- Abfahrtslauf</li> </ul> <p><i>Materialien:</i></p> <ul style="list-style-type: none"> <li>- Eigenes Script</li> <li>- Ergänzungsmaterialien zum Lehrplannavigator</li> </ul>
<p><b>4. Vertiefung - Anwendungen von Listen, Stapeln oder Schlangen in mindestens einem weiteren Kontext</b></p>	<ul style="list-style-type: none"> <li>• testen Programme systematisch anhand von Beispielen (I),</li> <li>• stellen lineare und nichtlineare Strukturen grafisch dar und erläutern ihren Aufbau (D).</li> </ul>	<p><i>Beispiele:</i></p> <ul style="list-style-type: none"> <li>- Skispringen</li> <li>- Terme in Postfix-Notation</li> <li>- Rangierbahnhof</li> <li>- Autos an einer Ampel zur Zufahrtsregelung</li> </ul> <p><i>Materialien:</i></p> <ul style="list-style-type: none"> <li>- Eigenes Script</li> <li>- Ergänzungsmaterialien zum Lehrplannavigator</li> </ul>

## Unterrichtsvorhaben Q1-III: Suchen und Sortieren auf linearen Datenstrukturen

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
<p><b>1. Suchen von Daten in Listen und Arrays</b></p> <p>(a) Lineare Suche in Listen und in Arrays</p> <p>(b) Binäre Suche in Arrays als Beispiel für rekursives Problemlösen</p> <p>(c) Untersuchung der beiden Suchverfahren hinsichtlich ihrer Effizienz (Laufzeitverhalten, Speicherbedarf)</p>	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> <li>analysieren und erläutern Algorithmen und Programme (A),</li> <li>beurteilen die syntaktische Korrektheit und die Funktionalität von Programmen (A),</li> <li>beurteilen die Effizienz von Algorithmen unter Berücksichtigung des Speicherbedarfs und der Zahl der Operationen (A),</li> </ul>	<p><i>Beispiele:</i></p> <ul style="list-style-type: none"> <li>Karteiverwaltung</li> <li>Bundesjugendspiele</li> </ul> <p><i>Materialien:</i></p> <ul style="list-style-type: none"> <li>Eigenes Script</li> <li>Ergänzungsmaterialien zum Lehrplannavigator</li> </ul>
<p><b>2. Sortieren in Listen und Arrays - Entwicklung und Implementierung von iterativen und rekursiven Sortierverfahren</b></p> <p>(a) Entwicklung und Implementierung eines einfachen Sortierverfahrens für eine Liste</p> <p>(b) Implementierung eines einfachen Sortierverfahrens für ein Feld</p> <p>(c) Entwicklung eines rekursiven Sortierverfahrens für ein Feld (z.B. Sortieren durch Mischen)</p>	<ul style="list-style-type: none"> <li>entwickeln iterative und rekursive Algorithmen unter Nutzung der Strategien „Modularisierung“ und „Teilen und Herrschen“ (M),</li> <li>modifizieren Algorithmen und Programme (I),</li> <li>implementieren iterative und rekursive Algorithmen auch unter Verwendung von dynamischen Datenstrukturen (I),</li> <li>implementieren und erläutern iterative und rekursive Such- und Sortierverfahren (I),</li> </ul>	<p><i>Beispiele:</i></p> <ul style="list-style-type: none"> <li>Karteiverwaltung</li> <li>Bundesjugendspiele</li> </ul> <p><i>Materialien:</i></p> <ul style="list-style-type: none"> <li>Eigenes Script</li> <li>Ergänzungsmaterialien zum Lehrplannavigator</li> </ul>
<p><b>3. Untersuchung der Effizienz der Sortierverfahren „Sortieren durch direktes Einfügen“ und „Quicksort“ auf linearen Listen</b></p> <p>(a) Grafische Veranschaulichung der Sortierverfahren</p> <p>(b) Untersuchung der Anzahl der Vergleichsoperationen und des Speicherbedarf bei beiden Sortierverfahren</p> <p>(c) Beurteilung der Effizienz der beiden Sortierverfahren</p>	<ul style="list-style-type: none"> <li>nutzen die Syntax und Semantik einer Programmiersprache bei der Implementierung und zur Analyse von Programmen (I),</li> <li>interpretieren Fehlermeldungen und korrigieren den Quellcode (I),</li> <li>testen Programme systematisch anhand von Beispielen (I),</li> <li>stellen iterative und rekursive Algorithmen umgangssprachlich und grafisch dar (D).</li> </ul>	<p><i>Beispiele:</i></p> <ul style="list-style-type: none"> <li>Karteiverwaltung</li> <li>Bundesjugendspiele</li> </ul> <p><i>Materialien:</i></p> <ul style="list-style-type: none"> <li>Eigenes Script</li> <li>Ergänzungsmaterialien zum Lehrplannavigator</li> </ul>

## Unterrichtsvorhaben Q1-IV: Modellierung und Nutzung von relationalen Datenbanken in Anwendungskontexten

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
<p><b>1. Nutzung von relationalen Datenbanken</b></p> <p>(a) Aufbau von Datenbanken und Grundbegriffe</p> <ul style="list-style-type: none"> <li>Entwicklung von Fragestellungen zur vorhandenen Datenbank</li> <li>Analyse der Struktur der vorgegebenen Datenbank und Erarbeitung der Begriffe Tabelle, Attribut, Datensatz, Datentyp, Primärschlüssel, Fremdschlüssel, Datenbankschema</li> </ul> <p>(b) SQL-Abfragen</p> <ul style="list-style-type: none"> <li>Analyse vorgegebener SQL-Abfragen und Erarbeitung der Sprachelemente von SQL (SELECT (DISTINCT) ...FROM, WHERE, AND, OR, NOT) auf einer Tabelle</li> <li>Analyse und Erarbeitung von SQL-Abfragen auf einer und mehrerer Tabelle zur Beantwortung der Fragestellungen (JOIN, UNION, AS, GROUP BY, ORDER BY, ASC, DESC, COUNT, MAX, MIN, SUM, Arithmetische Operatoren: +, -, *, /, (...), Vergleichsoperatoren: =, &lt;&gt;, &gt;, &lt;, &gt;=, &lt;=, LIKE, BETWEEN, IN, IS NULL)</li> </ul> <p>(c) Vertiefung an weiteren Datenbankbeispielen</p>	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> <li>erläutern die Eigenschaften und den Aufbau von Datenbanksystemen unter dem Aspekt der sicheren Nutzung (A),</li> <li>analysieren und erläutern die Syntax und Semantik einer Datenbankabfrage (A),</li> <li>analysieren und erläutern eine Datenbankmodellierung (A),</li> <li>erläutern die Eigenschaften normalisierter Datenbankschemata (A),</li> <li>bestimmen Primär- und Sekundärschlüssel (M),</li> <li>ermitteln für anwendungsbezogene Problemstellungen Entitäten, zugehörige Attribute, Relationen und Kardinalitäten (M),</li> <li>modifizieren eine Datenbankmodellierung (M),</li> </ul>	<p><i>Beispiele:</i></p> <ul style="list-style-type: none"> <li>VideoCenter</li> <li>Bundesliga</li> <li>Stundenplan</li> <li>Schulbuchausleihe</li> </ul> <p><i>Materialien:</i></p> <ul style="list-style-type: none"> <li>Eigenes Script</li> <li>OpenOffice</li> </ul>
<p><b>2. Modellierung von relationalen Datenbanken</b></p> <p>(a) Entity-Relationship-Diagramm</p> <ul style="list-style-type: none"> <li>Ermittlung von Entitäten, zugehörigen Attributen, Relationen und Kardinalitäten in Anwendungssituationen und Modellierung eines Datenbankentwurfs in Form eines Entity-Relationship-Diagramms</li> <li>Erläuterung und Modifizierung einer Datenbankmodellierung</li> </ul> <p>(b) Entwicklung einer Datenbank aus einem Datenbankentwurf</p> <ul style="list-style-type: none"> <li>Modellierung eines relationalen Datenbankschemas zu einem Entity-Relationship-Diagramm inklusive der Bestimmung von Primär- und Sekundärschlüsseln</li> </ul> <p>(c) Redundanz, Konsistenz und Normalformen</p> <ul style="list-style-type: none"> <li>Untersuchung einer Datenbank hinsichtlich Konsistenz und Redundanz in einer Anwendungssituation</li> <li>Überprüfung von Datenbankschemata hinsichtlich der 1. bis 3. Normalform und Normalisierung (um Redundanzen zu vermeiden und Konsistenz zu gewährleisten)</li> </ul>	<ul style="list-style-type: none"> <li>modellieren zu einem Entity-Relationship-Diagramm ein relationales Datenbankschema (M),</li> <li>bestimmen Primär- und Sekundärschlüssel (M),</li> <li>überführen Datenbankschemata in vorgegebene Normalformen (M),</li> <li>verwenden die Syntax und Semantik einer Datenbankabfragesprache, um Informationen aus einem Datenbanksystem zu extrahieren (I),</li> <li>ermitteln Ergebnisse von Datenbankabfragen über mehrere verknüpfte Tabellen (D),</li> <li>stellen Entitäten mit ihren Attributen und die Beziehungen zwischen Entitäten in einem Entity-Relationship-Diagramm grafisch dar (D),</li> <li>überprüfen Datenbankschemata auf vorgegebene Normalisierungseigenschaften (D).</li> </ul>	<p><i>Beispiele:</i></p> <ul style="list-style-type: none"> <li>Fahrradverleih</li> <li><i>Beispiel:</i> Reederei</li> <li>Buchungssystem</li> <li>Schulverwaltung</li> </ul>

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
<p><b>1. Daten in Netzwerken und Sicherheitsaspekte in Netzen sowie beim Zugriff auf Datenbanken</b></p> <p>(a) Beschreibung eines Datenbankzugriffs im Netz anhand eines Anwendungskontextes und einer Client-Server-Struktur zur Klärung der Funktionsweise eines Datenbankzugriffs</p> <p>(b) Netztopologien als Grundlage von Client-Server-Strukturen und TCP/IP-Schichtenmodell als Beispiel für eine Paketübermittlung in einem Netz</p> <p>(c) Vertraulichkeit, Integrität, Authentizität in Netzwerken sowie symmetrische und asymmetrische kryptografische Verfahren (Cäsar-, Vigenère-, RSA-Verfahren) als Methoden Daten im Netz verschlüsselt zu übertragen</p>	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> <li>• beschreiben und erläutern Topologien, die Client-Server-Struktur und Protokolle sowie ein Schichtenmodell in Netzwerken (A),</li> <li>• analysieren und erläutern Eigenschaften und Einsatzbereiche symmetrischer und asymmetrischer Verschlüsselungsverfahren (A),</li> <li>• untersuchen und bewerten anhand von Fallbeispielen die Auswirkungen des Einsatzes von Informatiksystemen, die Sicherheit von Informatiksystemen sowie die Einhaltung der Datenschutzbestimmungen und des Urheberrechts (A),</li> <li>• untersuchen und bewerten Problemlagen, die sich aus dem Einsatz von Informatiksystemen ergeben, hinsichtlich rechtlicher Vorgaben, ethischer Aspekte und gesellschaftlicher Werte unter Berücksichtigung unterschiedlicher Interessenlagen (A),</li> <li>• nutzen bereitgestellte Informatiksysteme und das Internet reflektiert zum Erschließen, zur Aufbereitung und Präsentation fachlicher Inhalte (D).</li> </ul>	<p><i>Materialien:</i> Ergänzungsmaterialien zum Lehrplannavigator Unterrichtsvorhaben, Verschlüsselung Q1.5 - Zugriff auf Daten in Netzwerken</p>
<p><b>2. Fallbeispiele zur Datenschutzproblematik und zum Urheberrecht</b></p>		<p><i>Materialien:</i> Ergänzungsmaterialien zum Lehrplannavigator Unterrichtsvorhaben Q1 5 - Datenschutz beim Videocenter, Materialblatt-Datenschutzgesetz</p>