

Kaiserin-Theophanu-Schule

Schulinterner Lehrplan

Informatik

Inhaltsverzeichnis

1 Rechtliche Rahmenbedingungen

- 1.1 Informatik in den Jahrgangsstufen 5 und 6
- 1.2 Informatik im Wahlpflichtfach der Sekundarstufe I
- 1.3 Informatik in der Sekundarstufe II

2 Die Fachgruppe Informatik der Kaiserin-Theophanu-Schule

3 Entscheidungen zum Unterricht

3.1 Jahrgangsstufe 6

- 3.1.1 Übersichtsraster Unterrichtsvorhaben
- 3.1.2 Konkretisierte Unterrichtsvorhaben

3.2 Wahlpflichtbereich

- 3.2.1 Übersichtsraster Unterrichtsvorhaben WP II
 - 3.2.1.1 Übersichtsraster Unterrichtsvorhaben WP II Stufe 9
 - 3.2.1.2 Übersichtsraster Unterrichtsvorhaben WP II Stufe 10
- 3.2.2 Konkretisierte Unterrichtsvorhaben WP II
 - 3.2.2.1 Konkretisierte Unterrichtsvorhaben WP II Stufe 9
 - 3.2.2.2 Konkretisierte Unterrichtsvorhaben WP II Stufe 10

3.3 Sekundarstufe II

- 3.3.1 Übersichtsraster Unterrichtsvorhaben
 - 3.3.1.1 Übersichtsraster Unterrichtsvorhaben in der EF
 - 3.3.1.2 Übersichtsraster Unterrichtsvorhaben in der Q1
 - 3.3.1.3 Übersichtsraster Unterrichtsvorhaben in der Q2
- 3.3.2 Konkretisierte Unterrichtsvorhaben
 - 3.3.2.1 Konkretisierte Unterrichtsvorhaben in der EF
 - 3.3.2.2 Konkretisierte Unterrichtsvorhaben in der Q1
 - 3.3.2.3 Konkretisierte Unterrichtsvorhaben in der Q2

4 Grundsätze der fachmethodischen und fachdidaktischen Arbeit

5 Grundsätze der Leistungsbewertung und Leistungsrückmeldung

- 5.1 Beurteilungsbereich Klausuren
- 5.2 Beurteilungsbereich Sonstige Mitarbeit
- 5.3 Grundsätze der Leistungsrückmeldung und Beratung
- 5.4 Individuelle Förderung
- 5.5 Bildung der Zeugnisnote

6 Entscheidungen zu fach- und unterrichtsübergreifenden Fragen

7 Qualitätssicherung und Evaluation

8 Eingeführte Lehrwerke

9 Bezug zum Medienkompetenzrahmen

10 Bezug zur Verbraucherbildung

1 Rechtliche Rahmenbedingungen

1.1 Informatik in den Jahrgangsstufen 5 und 6

- Kernlehrplan Informatik für die Sekundarstufe I - Klasse 5 und 6 in Nordrhein-Westfalen, Heft 5028, 1. Auflage 2021

Der Kernlehrplan tritt zum 01.08.2021 aufsteigend ab Klasse 5 in Kraft.

1.2 Informatik im Wahlpflichtfach der Sekundarstufe I

- Kernlehrplan für die Sekundarstufe I Gymnasium in Nordrhein-Westfalen, Wahlpflichtfach Informatik, Heft 34241, 1. Auflage 2019

Der Kernlehrplan tritt zum 01.08.2022 für alle Schülerinnen und Schüler, die in den Wahlpflichtunterricht eintreten, aufsteigend in Kraft.

1.3 Informatik in der Sekundarstufe II

- Kernlehrplan für die Sekundarstufe II Gymnasium/Gesamtschule in Nordrhein-Westfalen Informatik, Heftnummer 4725, 1. Auflage 2014

Der Kernlehrplan tritt zum 01.08.2014, beginnend mit der Einführungsphase, in Kraft.

2 Die Fachgruppe Informatik der Kaiserin-Theophanu-Schule

Bei der Kaiserin-Theophanu-Schule handelt es sich um ein Gymnasium in Köln-Kalk mit zurzeit ca. 1000 Schülerinnen und Schülern und etwa 90 Lehrerinnen und Lehrern. Das Einzugsgebiet der Schule umfasst Köln-Kalk sowie die östlich davon liegenden Stadtgebiete. Im Bereich der Sekundarstufe II kooperiert die Kaiserin-Theophanu-Schule mit dem Heinrich-Heine-Gymnasium in Köln-Ostheim.

Ab dem Schuljahr 2022-2023 unterrichten wir das Fach Informatik verpflichtend zweistündig in der Jahrgangsstufe 6. Ebenfalls ab dem Schuljahr 2022-2023 wird Informatik jeweils 3-stündig im Rahmen des dann wieder neunjährigen gymnasialen Bildungsgangs im Wahlpflichtbereich in den Jahrgangsstufen 9 und 10 angeboten. In der Sekundarstufe II bietet das Kaiserin-Theophanu-Gymnasium in den Jahrgangsstufen EF, Q1 und Q2 jeweils einen 3-stündigen Grundkurs in Informatik an.

Um insbesondere Schülerinnen und Schülern gerecht zu werden, die im Wahlpflichtbereich keinen Informatikunterricht besucht haben, wird in den Kursen der Einführungsphase besonderer Wert darauf gelegt, dass keine Vorkenntnisse aus der Sekundarstufe I zum erfolgreichen Durchlaufen der Oberstufe erforderlich sind.

Der Unterricht der Sekundarstufe II wird mit Hilfe der Programmiersprache Java durchgeführt.

Durch projektartiges Vorgehen, offene Aufgaben und Möglichkeiten, Problemlösungen zu verfeinern oder zu optimieren, entspricht der Informatikunterricht in besonderem Maße den Erziehungszielen, Leistungsbereitschaft zu fördern, ohne zu überfordern.

Die gemeinsame Entwicklung von Materialien und Unterrichtsvorhaben, die Evaluation von Lehr- und Lernprozessen sowie die stetige Überprüfung und eventuelle Modifikation des schulinternen Curriculums durch die Fachkonferenz Informatik stellen einen wichtigen Beitrag zur Qualitätssicherung und -entwicklung des Unterrichts dar.

Zurzeit besteht die Fachschaft Informatik der Kaiserin-Theophanu-Schule aus zwei Lehrkräften, denen fünf Computerräume mit insgesamt ca.100 Computerarbeitsplätzen und zusätzlich ein Selbstlernzentrum mit 8 Plätzen zur Verfügung stehen. Alle Arbeitsplätze sind an das schulinterne Rechnernetz angeschlossen, so dass Schülerinnen und Schüler alle Arbeitsplätze der Schule zum Zugriff auf ihre eigenen Daten, zur Recherche im Internet oder zur Bearbeitung schulischer Aufgaben verwenden können.

Der Unterricht erfolgt im 45-Minuten-Takt.

3 Entscheidungen zum Unterricht

Die Darstellung der Unterrichtsvorhaben im schulinternen Lehrplan besitzt den Anspruch, sämtliche im Kernlehrplan angeführten Kompetenzen abzudecken. Dies entspricht der Verpflichtung jeder Lehrkraft, Schülerinnen und Schülern Lerngelegenheiten so zu ermöglichen, so dass alle Kompetenzerwartungen des Kernlehrplans von ihnen erfüllt werden können. Die entsprechende Umsetzung erfolgt auf zwei Ebenen: der Übersichts- und der Konkretisierungsebene.

Im „Übersichtsraster Unterrichtsvorhaben“ wird die für alle Lehrerinnen und Lehrer gemäß Fachkonferenzbeschluss verbindliche Verteilung der Unterrichtsvorhaben dargestellt. Das Übersichtsraster dient dazu, den Kolleginnen und Kollegen einen schnellen Überblick über die Zuordnung der Unterrichtsvorhaben zu den einzelnen Jahrgangsstufen sowie den im Kernlehrplan genannten Kompetenzen, Inhaltsfeldern und inhaltlichen Schwerpunkten zu verschaffen. Der ausgewiesene Zeitbedarf versteht sich als grobe Orientierungsgröße, die nach Bedarf über- oder unterschritten werden kann. Um Freiraum für Vertiefungen, besondere Schülerinteressen, aktuelle Themen bzw. die Erfordernisse anderer besonderer Ereignisse (z.B. Praktika, Kursfahrten, o.ä.) zu erhalten, wurden im Rahmen dieses schulinternen Lehrplans ca. 85 Prozent der Bruttounterrichtszeit verplant.

Während der Fachkonferenzbeschluss zum „Übersichtsraster Unterrichtsvorhaben“ zur Gewährleistung vergleichbarer Standards sowie zur Absicherung von Lerngruppenübertritten und Lehrkraftwechseln für alle Mitglieder der Fachkonferenz Bindekraft entfalten soll, beinhaltet die Ausweisung „konkretisierter Unterrichtsvorhaben“ Beispiele und Materialien, die empfehlenden Charakter haben. Referendarinnen und Referendaren sowie neuen Kolleginnen und Kollegen dienen diese vor allem zur Orientierung in der neuen Schule, aber auch zur Verdeutlichung von unterrichtsbezogenen fachgruppeninternen Absprachen zu didaktisch-methodischen Zugängen, fächerübergreifenden Kooperationen, Lernmitteln und -orten sowie vorgesehenen Leistungsüberprüfungen.

Da in den folgenden Unterrichtsvorhaben Inhalte in der Regel anhand von Problemstellungen in Anwendungskontexten bearbeitet werden, werden in einigen Unterrichtsvorhaben jeweils mehrere Inhaltsfelder angesprochen.

3.1 Jahrgangsstufe 6

Inhaltsfelder:

- Information und Daten
- Algorithmen
- Automaten und künstliche Intelligenz
- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Kompetenzbereiche:

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

Übergeordnete Kompetenzerwartungen:

<p>Argumentieren (A)</p> <p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none">• formulieren Fragen zu einfachen informatischen Sachverhalten,• äußern Vermutungen zu informatischen Sachverhalten auf der Basis von Alltagsvorstellungen oder Vorwissen,• erläutern mögliche Auswirkungen des Einsatzes von Informatiksystemen,• begründen die Auswahl eines Informatiksystems,• bewerten ein Ergebnis einer informatischen Modellierung.	<p>Darstellen und Interpretieren (DI)</p> <p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none">• beschreiben einfache Darstellungen von informatischen Sachverhalten,• stellen informatische Sachverhalte in geeigneter Form dar,• interpretieren informatische Darstellungen.
<p>Modellieren und Implementieren (MI)</p> <p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none">• erstellen informatische Modelle zu gegebenen Sachverhalten,• implementieren informatische Modelle unter Verwendung algorithmischer Grundstrukturen,• überprüfen Modelle und Implementierungen.	<p>Kommunizieren und Kooperieren (KK)</p> <p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none">• beschreiben einfache informatische Sachverhalte unter Verwendung von Fachbegriffen sachgerecht,• anstelle der vorherigen KE: erläutern informatische Sachverhalte unter Verwendung von Fachbegriffen sachgerecht,• kooperieren in verschiedenen Formen der Zusammenarbeit bei der Bearbeitung einfacher informatischer Probleme,• strukturieren gemeinsam eine Lösung für ein informatisches Problem,• dokumentieren gemeinsam ihren Arbeitsprozess und ihre Ergebnisse auch mithilfe digitaler Werkzeuge,• setzen bei der Bearbeitung einer informatischen Problemstellung geeignete digitale Werkzeuge zum kollaborativen Arbeiten ein.

3.1.1 Übersichtsraster Unterrichtsvorhaben

Jahrgangsstufe 6	
<p>Unterrichtsvorhaben 0</p> <p>Thema: Einführung und Grundlagen</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Kommunizieren und Kooperieren - Argumentieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Informatiksysteme <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Organisatorisches - Regeln im Informatikraum - Was ist Informatik - Informatiksysteme der Schule <p>Zeitbedarf: 2 Stunden</p>	<p>Unterrichtsvorhaben I</p> <p>Thema: Informatiksysteme</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Modellieren und Implementieren - Darstellen und Interpretieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Informatiksysteme <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Aufbau und Funktionsweise von Informatiksystemen - Anwendung von Informatiksystemen <p>Zeitbedarf: 8 Stunden</p>
<p>Unterrichtsvorhaben II</p> <p>Thema: Information und Daten</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Modellieren und Implementieren - Darstellen und Interpretieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Information und Daten <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Daten und Codierung - Informationsgehalt von Daten <p>Zeitbedarf: 8 Stunden</p>	<p>Unterrichtsvorhaben III</p> <p>Thema: Verschlüsselungsverfahren</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Darstellen und Interpretieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Information und Daten <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Verschlüsselungsverfahren <p>Zeitbedarf: 6 Stunden</p>

Jahrgangsstufe 6	
<p>Unterrichtsvorhaben IV</p> <p>Thema: Algorithmen</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Modellieren und Implementieren - Darstellen und Interpretieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Algorithmen <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Algorithmen und algorithmische Grundkonzepte <p>Zeitbedarf: 8 Stunden</p>	<p>Unterrichtsvorhaben V</p> <p>Thema: Programmieren mit einer visuellen Programmierumgebung</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Modellieren und Implementieren - Darstellen und Interpretieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Algorithmen <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Implementation von Algorithmen <p>Zeitbedarf: 11 Stunden</p>
<p>Unterrichtsvorhaben VI</p> <p>Thema: Automatisierung und künstliche Intelligenz</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"> - Automaten und künstliche Intelligenz <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Aufbau und Wirkungsweise einfacher Automaten - Maschinelles Lernen mit Entscheidungsbäumen - Maschinelles Lernen mit neuronalen Netzen <p>Zeitbedarf: 9 Stunden</p>	<p>Unterrichtsvorhaben VII</p> <p>Thema: Informatik, Mensch und Gesellschaft</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Informatik, Mensch und Gesellschaft <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Informatiksysteme in der Lebens- und Arbeitswelt - Datenbewusstsein - Datensicherheit und Sicherheitsregeln <p>Zeitbedarf: 8 Stunden</p>

Summe Jahrgangsstufe 6: 60 Stunden

3.1.2 Konkretisierte Unterrichtsvorhaben

Im Folgenden sollen die im letzten Abschnitt aufgeführten Unterrichtsvorhaben konkretisiert werden. Diese Konkretisierung hat vorschlagenden Charakter, ohne die pädagogische Freiheit des Lehrenden einschränken zu wollen.

Unterrichtsvorhaben 0: Grundlagen (ca. 2 Stunden)

Unterrichtssequenzen	Kompetenzerwartungen und inhaltliche Schwerpunkte
0.1 Regeln für den Informatikraum 0.2 Computertastatur und Computermaus 0.3 Richtig Sitzen vor dem Bildschirm	Auch wenn der Kernlehrplan Informatik 5/6 es nicht explizit fordert, kann es Sinn machen, gewisse Grundlagen zum Arbeiten mit Computern mit den Lernenden zu thematisieren bzw. den Wissensstand der Lerngruppe diesbezüglich zu prüfen. Im Zeitalter der Smartphones und Tablets ist nicht zwangsläufig jeder Schülerin bzw. Schüler gewohnt, mit einem Computer zu arbeiten.

Unterrichtsvorhaben I: Informatiksysteme (ca. 8 Stunden)

Unterrichtssequenzen	Kompetenzerwartungen und inhaltliche Schwerpunkte
1.1 Das EVA-Prinzip 1.2 Projekt: Woraus besteht ein Computer? 1.3 Informatiksysteme, Hardware und Software 1.4 Benutzerkonten und sichere Passwörter 1.5 Das Speichern von Dokumenten und Dateien 1.6 Ordnung muss sein – Der Dateimanager 1.7 Ordnerstrukturen darstellen und anpassen 1.8 Arten der Datenspeicherung	<p>Inhaltliche Schwerpunkte:</p> <ul style="list-style-type: none"> • Aufbau und Funktionsweise von Informatiksystemen • Anwendung von Informatiksystemen <p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> • benennen Beispiele für (vernetzte) Informatiksysteme aus ihrer Erfahrungswelt (DI), • benennen Grundkomponenten von (vernetzten) Informatiksystemen und beschreiben ihre Funktionen (DI), • beschreiben das Prinzip der Eingabe, Verarbeitung und Ausgabe (EVA-Prinzip) als grundlegendes Prinzip der Datenverarbeitung (DI), • vergleichen Möglichkeiten der Datenverwaltung hinsichtlich ihrer spezifischen Charakteristika (u. a. Speicherort, Kapazität, Aspekte der Datensicherheit) (A), • setzen zielgerichtet Informatiksysteme zur Verarbeitung von Daten ein (MI), • erläutern Prinzipien der strukturierten Dateiverwaltung (A), • setzen Informatiksysteme zur Kommunikation und Kooperation ein (KK).

Unterrichtsvorhaben II: Information und Daten (ca. 8 Stunden)

Unterrichtssequenzen	Kompetenzerwartungen und inhaltliche Schwerpunkte
2.1 Kommunikation im Alltag und in der Informatik 2.2 Arten der Codierung 2.3 Bits und Bytes 2.4 Binärzahlen 2.5 Textcodierung – Der ASCII-Code	Inhaltliche Schwerpunkte: <ul style="list-style-type: none"> • Daten und ihre Codierung • Informationsgehalt von Daten Die Schülerinnen und Schüler <ul style="list-style-type: none"> • erläutern den Datenbegriff anhand von Beispielen aus ihrer Erfahrungswelt (A), • erläutern den Zusammenhang und die Bedeutung von Information und Daten (A), • stellen eine ausgewählte Information in geeigneter Form als Daten formalsprachlich oder graphisch dar (DI), • nennen Beispiele für die Codierung von Daten aus ihrer Erfahrungswelt (DI), • codieren und decodieren Daten unter Verwendung des Binärsystems (MI), • interpretieren ausgewählte Daten als Information im gegebenen Kontext (DI), • erläutern Einheiten von Datenmengen (A / KK), vergleichen Datenmengen hinsichtlich ihrer Größe mithilfe anschaulicher Beispiele aus ihrer Lebenswelt (DI).

Unterrichtsvorhaben III: Verschlüsselungsverfahren (ca. 6 Stunden)

Unterrichtssequenzen	Kompetenzerwartungen und inhaltliche Schwerpunkte
3.1 Geheimnisse bewahren mit Verschlüsselung 3.2 Verschlüsselungsverfahren – Monoalphabetische Verschlüsselung 3.3 Verschlüsselungsverfahren – Transposition 3.4 Verschlüsselungsverfahren – Steganographie	Inhaltliche Schwerpunkte: <ul style="list-style-type: none"> • Verschlüsselungsverfahren Die Schülerinnen und Schüler <ul style="list-style-type: none"> • erläutern ein einfaches Transpositionsverfahren als Möglichkeit der Verschlüsselung (DI), • vergleichen verschiedene Verschlüsselungsverfahren unter Berücksichtigung von ausgewählten Sicherheitsaspekten (DI).

Unterrichtsvorhaben IV: Algorithmen (ca. 8 Stunden)

Unterrichtssequenzen	Kompetenzerwartungen und inhaltliche Schwerpunkte
4.1 Beschreibung von Abläufen 4.2 Algorithmen im Alltag 4.3 Genaue Anweisungen und Abläufe 4.4 Beschreibungen abkürzen 4.5 Bedingte Anweisung und Verzweigung 4.6 Vom Algorithmus zum Programm	Inhaltliche Schwerpunkte: <ul style="list-style-type: none"> Algorithmen und algorithmische Grundkonzepte Die Schülerinnen und Schüler <ul style="list-style-type: none"> formulieren zu Abläufen aus dem Alltag eindeutige Handlungsvorschriften (DI), überführen Handlungsvorschriften in einen Programmablaufplan (PAP) oder ein Struktogramm (MI), führen Handlungsvorschriften schrittweise aus (MI), identifizieren in Handlungsvorschriften Anweisungen und die algorithmischen Grundstrukturen Sequenz, Verzweigung und Schleife (MI).

Unterrichtsvorhaben V: Programmieren mit einer Programmierumgebung (ca. 11 Std.)

Unterrichtssequenzen	Kompetenzerwartungen und inhaltliche Schwerpunkte
5.1 Einführung in die Programmierung mit Scratch 5.2 Sequenzen von Anweisungen an ein Objekt 5.3 Reagieren auf Ereignisse 5.4 Wiederholung mit fester Anzahl 5.5 Schleifen mit Abbruchbedingungen 5.6 Verzweigungen 5.7 Variablen 5.8 Zielgerichtetes Testen von Programmen 5.9 Projekt: Ein Projekt planen und durchführen	Inhaltliche Schwerpunkte: <ul style="list-style-type: none"> Implementation von Algorithmen Die Schülerinnen und Schüler <ul style="list-style-type: none"> implementieren Algorithmen in einer visuellen Programmiersprache (MI), implementieren Algorithmen unter Berücksichtigung des Prinzips der Modularisierung (MI), überprüfen die Wirkungsweise eines Algorithmus durch zielgerichtetes Testen (MI), ermitteln durch die Analyse eines Algorithmus dessen Ergebnis (DI), bewerten einen als Quelltext, Programmablaufplan (PAP) oder Struktogramm dargestellten Algorithmus hinsichtlich seiner Funktionalität (A).

Unterrichtsvorhaben VI: Automatisierung und künstliche Intelligenz (ca. 9 Stunden)

Unterrichtssequenzen	Kompetenzerwartungen und inhaltliche Schwerpunkte
6.1 Automaten im Alltag 6.2 Zustandsdiagramme 6.3 Projekt: Automaten mit Scratch 6.4 Künstliche Intelligenz in unserem Alltag 6.5 Entscheidungsbäume 6.6 Lernen durch Training 6.7 Neuronale Netze 6.8 Projekt: KI mit Scratch	Inhaltliche Schwerpunkte: <ul style="list-style-type: none"> • Aufbau und Wirkungsweise einfacher Automaten • Maschinelles Lernen mit Entscheidungsbäumen • Maschinelles Lernen mit neuronalen Netzen Die Schülerinnen und Schüler <ul style="list-style-type: none"> • erläutern die Funktionsweise eines Automaten aus ihrer Lebenswelt (A), • stellen Abläufe in Automaten graphisch dar (DI), • benennen Anwendungsbeispiele künstlicher Intelligenz aus ihrer Lebenswelt (A), • stellen das Grundprinzip eines Entscheidungsbaumes enaktiv als ein Prinzip des maschinellen Lernens dar (DI), • beschreiben die grundlegende Funktionsweise künstlicher neuronaler Netze in verschiedenen Anwendungsbeispielen (KK).

Unterrichtsvorhaben VII: Informatik, Mensch und Gesellschaft (ca. 8 Stunden)

Unterrichtssequenzen	Kompetenzerwartungen und inhaltliche Schwerpunkte
7.1 Kleine und große Netzwerke – Das Internet 7.2 Daten und Gefahren im Internet 7.3 Schutz von Daten mit Hilfe von Informatiksystemen 7.4 Wem gehören die Daten? – Rechte von Nutzern 7.5 Verhalten und Umgang mit sozialen Netzwerken	Inhaltliche Schwerpunkte: <ul style="list-style-type: none"> • Informatiksysteme in der Lebens- und Arbeitswelt • Datenbewusstsein • Datensicherheit und Sicherheitsregeln Die Schülerinnen und Schüler <ul style="list-style-type: none"> • beschreiben an Beispielen die Bedeutung von Informatiksystemen in der Lebens- und Arbeitswelt (KK), • erläutern an ausgewählten Beispielen Auswirkungen des Einsatzes von Informatiksystemen (A/KK), • beschreiben anhand von ausgewählten Beispielen die Verarbeitung und Nutzung personenbezogener Daten (DI), • erläutern anhand von Beispielen aus ihrer Lebenswelt Nutzen und Risiken beim Umgang mit eigenen und fremden Daten auch im Hinblick auf Speicherorte (A), • beschreiben Maßnahmen zum Schutz von Daten mithilfe von Informatiksystemen (A).

3.2 Wahlpflichtbereich WP II

Inhaltsfelder:

- Information und Daten
- Algorithmen
- Formale Sprachen
- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Kompetenzbereiche:

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

Übergeordnete Kompetenzerwartungen:

<p>Argumentieren (A)</p> <p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none">• formulieren Fragen zu informatischen Sachverhalten,• stellen informatische Sachverhalte strukturiert dar und analysieren deren Zusammenhänge,• erläutern und beurteilen informatische Modellierungen, Computerprogramme und Informatiksysteme,• begründen Entscheidungen bei der Nutzung von Informatiksystemen,• bewerten mögliche Auswirkungen des Einsatzes von Informatiksystemen.	<p>Darstellen und Interpretieren (DI)</p> <p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none">• interpretieren unterschiedliche Darstellungen von informatischen Sachverhalten,• veranschaulichen informatische Sachverhalte,• wählen geeignete Darstellungsformen aus,• interpretieren Ergebnisse von Implementierungen.
<p>Modellieren und Implementieren (MI)</p> <p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none">• erstellen informatische Modelle zu gegebenen Sachverhalten,• implementieren informatische Modelle,• analysieren Modelle und Implementierungen,• analysieren und bewerten Informatiksysteme und Anwendungen unter dem Aspekt der zugrundeliegenden Modellierung,• beurteilen Modelle, Implementierungen und die verwendeten Werkzeuge hinsichtlich der Eignung zur Erfassung eines Sachverhaltes.	<p>Kommunizieren und Kooperieren (KK)</p> <p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none">• stellen informatische Sachverhalte unter Verwendung von Fachbegriffen• mündlich und schriftlich sachgerecht dar,• kooperieren bei der Bearbeitung informatischer Probleme,• dokumentieren gemeinsam ihren Arbeitsprozess und ihre Ergebnisse.

3.2.1 Übersichtsraster Unterrichtsvorhaben WP II

3.2.1.1 Übersichtsraster Unterrichtsvorhaben WP II Stufe 9

Wahlpflichtbereich Stufe 9	
<p>Unterrichtsvorhaben 0</p> <p>Thema: Einführung und Grundlagen</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Darstellen und Interpretieren - Modellieren und Implementieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Information und Daten - Algorithmen - Formale Sprachen - Informatiksysteme - Informatik, Mensch und Gesellschaft <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Organisatorisches - Regeln im Informatikraum - Was ist Informatik - Überblick über die Inhalte der zwei Schuljahre - Informatiksysteme der Schule - Nutzung des Dateisystems <p>Zeitbedarf: 3 Stunden</p>	<p>Unterrichtsvorhaben I</p> <p>Thema: Aufbau und Arbeitsweise eines PCs</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 - Information und Daten <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Information, Daten und ihre Codierung - Aufbau und Funktionsweise von Informatiksystemen - Woraus besteht ein PC <p>Zeitbedarf: 9 Stunden</p>
<p>Unterrichtsvorhaben II</p> <p>Thema: Rechnen mit logische Schaltungen</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Modellieren und Implementieren - Darstellen und Interpretieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Information und Daten - Informatiksysteme <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Daten und Codierung - Informationsgehalt von Daten - Aufbau und Funktionsweise einfacher Informatiksysteme - Anwendung von Informatiksystemen <p>Zeitbedarf: 21 Stunden</p>	<p>Unterrichtsvorhaben III</p> <p>Thema: EVA und v. Neumann-Prinzip</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Darstellen und Interpretieren - Modellieren und Implementieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Algorithmen - Information und Daten - Informatiksysteme <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Daten und Codierung - Informationsgehalt von Daten - Aufbau und Funktionsweise einfacher Informatiksysteme - Anwendung von Informatiksystemen <p>Zeitbedarf: 9 Stunden</p>

Wahlpflichtbereich Stufe 9	
<p>Unterrichtsvorhaben IV</p> <p>Thema: Anwendersoftware - Excel</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Modellieren und Implementieren - Darstellen und Interpretieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Information und Daten - Algorithmen - Formale Sprachen - Informatiksysteme <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Daten und Codierung - Informationsgehalt von Daten - Anwendung von Informatiksystemen <p>Zeitbedarf: 21 Stunden</p>	<p>Unterrichtsvorhaben V</p> <p>Thema: Erste textorientierte Programmierung in einer visuellen Programmierumgebung</p> <ul style="list-style-type: none"> - Java-Hamster <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Modellieren und Implementieren - Darstellen und Interpretieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Information und Daten - Algorithmen - Formale Sprachen - Informatiksysteme <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Information, Daten und ihre Codierung - Erfassung, Verarbeitung und Verwaltung von Daten - Erstellung von Quelltexten - Analyse von Quelltexten - Aufbau und Funktionsweise einfacher Informatiksysteme - Anwendung von Informatiksystemen <p>Zeitbedarf: 27 Stunden</p>

Summe Jahrgangsstufe 9: 90 Stunden

3.2.2 3.2.1.2 Übersichtsraster Unterrichtsvorhaben WP II Stufe 10

Wahlpflichtbereich Stufe 10	
<p>Unterrichtsvorhaben 0</p> <p>Thema: Einführung und Ausblick</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Information und Daten - Algorithmen - Formale Sprachen - Informatiksysteme - Informatik, Mensch und Gesellschaft <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Organisatorisches - Überblick über die Inhalte des Schuljahres <p>Zeitbedarf: 3 Stunden</p>	<p>Unterrichtsvorhaben I</p> <p>Thema: Einführung in Python</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Modellieren und Implementieren - Darstellen und Interpretieren <ul style="list-style-type: none"> - Kommunizieren und Kooperieren - Argumentieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Information und Daten - Algorithmen - Formale Sprachen - Informatiksysteme <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Information, Daten und ihre Codierung - Erfassung, Verarbeitung und Verwaltung von Daten - Erstellung von Quelltexten - Analyse von Quelltexten - Aufbau und Funktionsweise einfacher Informatiksysteme - Anwendung von Informatiksystemen <p>Zeitbedarf: 12 Stunden</p>
<p>Unterrichtsvorhaben II</p> <p>Thema: Algorithmen</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Modellieren und Implementieren - Darstellen und Interpretieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Information und Daten - Algorithmen - Formale Sprachen - Informatiksysteme <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Information, Daten und ihre Codierung - Erfassung, Verarbeitung und Verwaltung von Daten - Erstellung von Quelltexten - Analyse von Quelltexten - Aufbau und Funktionsweise einfacher Informatiksysteme - Anwendung von Informatiksystemen <p>Zeitbedarf: 9 Stunden</p>	<p>Unterrichtsvorhaben III</p> <p>Thema: Algorithmen in Python</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Modellieren und Implementieren - Darstellen und Interpretieren - Argumentieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Information und Daten - Algorithmen - Formale Sprachen - Informatiksysteme <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Information, Daten und ihre Codierung - Erfassung, Verarbeitung und Verwaltung von Daten - Erstellung von Quelltexten - Analyse von Quelltexten - Aufbau und Funktionsweise einfacher Informatiksysteme - Anwendung von Informatiksystemen <p>Zeitbedarf: 18 Stunden</p>

Wahlpflichtbereich Stufe 10	
<p>Unterrichtsvorhaben IV</p> <p>Thema: Funktionsweise von Netzwerken</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Modellieren und Implementieren - Darstellen und Interpretieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Information und Daten - Formale Sprachen - Informatiksysteme <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Information, Daten und ihre Codierung - Erfassung, Verarbeitung und Verwaltung von Daten - Aufbau und Funktionsweise einfacher Informatiksysteme - Anwendung von Informatiksystemen <p>Zeitbedarf: 12 Stunden</p>	<p>Unterrichtsvorhaben V</p> <p>Thema: Webseitenerstellung mit HTML</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Modellieren und Implementieren - Darstellen und Interpretieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Information und Daten - Formale Sprachen - Informatiksysteme - Informatik, Mensch und Gesellschaft <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Information, Daten und ihre Codierung - Erfassung, Verarbeitung und Verwaltung von Daten - Erstellung von Quelltexten - Analyse von Quelltexten - Aufbau und Funktionsweise einfacher Informatiksysteme - Anwendung von Informatiksystemen <p>Zeitbedarf: 30 Stunden</p>
<p>Unterrichtsvorhaben VI</p> <p>Thema: Rechtliche Grundlagen</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"> - Informatik, Mensch und Gesellschaft <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Informatiksysteme im Kontext gesellschaftlicher und rechtlicher Normen - Chancen und Risiken bei der Nutzung von Informatiksystemen <p>Zeitbedarf: 6 Stunden</p>	

3.2.2 Konkretisierte Unterrichtsvorhaben WP II Stufe 9

Im Folgenden sollen die aufgeführten Unterrichtsvorhaben konkretisiert werden. Diese Konkretisierung hat vorschlagenden Charakter, ohne die pädagogische Freiheit des Lehrenden einschränken zu wollen.

3.2.2.1 Konkretisierte Unterrichtsvorhaben WP II Stufe 9

Unterrichtsvorhaben WP II 9 - 0

Thema: Einführung und Grundlagen

Leitfragen

Womit beschäftigt sich Informatik, Was machen wir in den zwei Jahren? Wie funktionieren die Rechner im Computerraum und wie arbeitet man mit dem Dateisystem und den Tauschordnern.

Absprachen zur vorhabenbezogene Konkretisierung

Nach den notwendigen organisatorischen Absprachen und der Einweisung in die Verhaltensweisen in einem Computerraum beschäftigen wir uns mit der Frage, womit sich Informatik eigentlich beschäftigt und welche Themen wir in den zwei Jahren im Wahlpflichtbereich Informatik behandeln werden. Anschließend werden die Schülerinnen und Schüler in die Benutzung der Rechner im Computerraum eingewiesen.

Zeitbedarf: 3 Stunden

Sequenzierung des Unterrichtsvorhabens

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Organisatorisches 2. Regeln im Computerraum 3. Was ist Informatik? 4. Überblick über die Inhalte der zwei Schuljahre 6. Nutzung der Rechner und des Dateisystems	Die Schülerinnen und Schüler <ul style="list-style-type: none"> • erläutern Prinzipien der strukturierten Dateiverwaltung und wenden diese an (MI), • kommunizieren und tauschen Daten mithilfe von Netzen aus (KK) • verarbeiten Daten mithilfe von Informatiksystemen (MI) • verarbeiten gleichartige Daten mit Hilfe eines geeigneten Werkzeuges (DI). 	Übersichtskarte Informatik

Unterrichtsvorhaben WP II 9 - I**Thema:** Aufbau und Arbeitsweise eines PCs**Leitfragen**

Woraus besteht ein Computer? Wofür benötigt er die verschiedenen Bauteile? Wie arbeiten diese zusammen? Wie baut man selbst einen Computer zusammen? Wie löst man Probleme, wenn der Computer nicht mehr richtig arbeitet?

Absprachen zur vorhabenbezogene Konkretisierung

Es werden alte Computer aufgeschraubt, zerlegt und wieder zusammengebaut. Die Wirkungsweise der Bauteile wird besprochen. Wir schauen einen Film, wie man selbst einen Computer aus Einzelteilen zusammenbauen kann.

Zeitbedarf: 9 Stunden**Sequenzierung des Unterrichtsvorhabens**

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Aufbau eines Computers 2. Arbeitsweise der Bestandteile 3. Wie baut man selbst einen Computer 4. Was sagen uns die Daten in den Verkaufsprospekten?	Die Schülerinnen und Schüler <ul style="list-style-type: none"> • benennen Grundkomponenten von Informatiksystemen und beschreiben ihre Funktionen (DI), • beschreiben Alltagsgeräte, in denen Informatiksysteme vorkommen (A). 	alte PCs alte Computerbestandteile Verkaufsprospekte Filme: Rechnerbau

Unterrichtsvorhaben WP II 9 - II**Thema:** Rechnen mit logische Schaltungen**Leitfragen**

Nachdem geklärt ist, aus welchen Bestandteilen ein Computer besteht, stellt sich die Frage, wie ein Computer mit Strom rechnen kann und wie er alle diese Rechnungen, Bilder, Töne und Videos erzeugen kann.

Absprachen zur vorhabenbezogene Konkretisierung

Wir erstellen mittels eines Programms zum Erstellen logischer Schaltungen einen Parallel-Addierer, der auch subtrahieren kann. In diesem Kontext werden die Begriffe Binärsystem, Dualsystem, Bit, Byte, KByte, usw. konkretisiert. Des Weiteren gehen wir auf das Addieren von Binärzahlen und die Bildung des Zweierkomplements ein. Am Ende steht ein komplettes Addier- und Subtrahierwerk. Auch die Codierung von Farben, Tönen und Bildern wird Thematisiert.

Zeitbedarf: 21 Stunden**Sequenzierung des Unterrichtsvorhabens**

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Einführung in LoCad 2. Erstellung einfacher logischer Schaltungen 3. Konstruktion eines Paralleladdierers 4. Darstellung von negativen Zahlen durch Zweierkomplementbildung 5. Konstruktion eines Paralleladdier- und -subtrahierwerks 6. Bits, Bytes, KBytes usw. 7. Codierung von Tönen, Bildern, etc.	Die Schülerinnen und Schüler <ul style="list-style-type: none"> • codieren Daten für die Verarbeitung mit einem Informatiksystem (DI), • interpretieren Ergebnisse eines Datenverarbeitungsprozesses (DI), • verarbeiten Daten mithilfe von Informatiksystemen (MI), • verwenden arithmetische und logische Operationen (MI), • verarbeiten gleichartige Daten mit Hilfe eines geeigneten Werkzeuges (DI), • erläutern Prinzipien der strukturierten Dateiverwaltung und wenden diese an (MI), • erläutern den Zusammenhang und die Bedeutung von Information und Daten (A), • repräsentieren Information in natürlicher Sprache, formalsprachlich und grafisch (DI). 	LoCad Filme: Wie ein Chip entsteht

Unterrichtsvorhaben WP II 9 - III**Thema:** EVA und v. Neumann-Prinzip**Leitfragen**

Nachdem wir wissen, wie ein Rechenwerk funktioniert, stellt sich die Frage, wie die verschiedenen Bestandteile eines Informatiksystems zusammenarbeiten. Wie kommen die Daten aus dem Arbeitsspeicher zum Prozessor? Wie unterscheidet der Rechner zwischen Programmen und Daten?

Absprachen zur vorhabenbezogene Konkretisierung

Anhand des Modellrechners Mops erarbeiten wir die einzelnen Schritte des von-Neumann-Prinzips und wenden uns anschließend dem EVA-Prinzip zu.

Zeitbedarf: 9 Stunden**Sequenzierung des Unterrichtsvorhabens**

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. von-Neumann-Prinzip 2. Der Modellrechner Mops 3. Kleinere Assemblerprogramme mit dem Modellrechner 4. Das EVA-Prinzip	Die Schülerinnen und Schüler <ul style="list-style-type: none"> • codieren Daten für die Verarbeitung mit einem Informatiksystem (DI), • interpretieren Ergebnisse eines Datenverarbeitungsprozesses (DI), • verarbeiten Daten mithilfe von Informatiksystemen (MI), • verwenden arithmetische und logische Operationen (MI), • verarbeiten gleichartige Daten mit Hilfe eines geeigneten Werkzeuges (DI), • entwerfen Algorithmen unter Verwendung des Variablenkonzeptes und von Kontrollstrukturen (MI), • reflektieren den Entwurfsprozess und beschreiben ihn auch fachsprachlich (A), • implementieren und kommentieren Algorithmen in einer Programmierumgebung (MI), • modifizieren Programme (MI), • überprüfen Handlungsvorschriften auf Eindeutigkeit und Terminierung (A), • analysieren und testen Algorithmen und Programme (MI), • erstellen syntaktisch korrekte Quelltexte in einer Programmiersprache (MI), • beschreiben das Prinzip der Eingabe, Verarbeitung und Ausgabe (EVA-Prinzip) als grundlegendes Prinzip der Datenverarbeitung (DI). 	Modellrechner Mops

Unterrichtsvorhaben WP II 9 - IV**Thema:** Anwendersoftware - Excel**Leitfragen**

Wir haben nun verstanden, wie ein Informatiksystem funktioniert. Kommen wir nun zu der Verwendung eines Rechners. Ein sehr sinnvolles und leistungsstarkes Programm, das gerade im beruflichen Umfeld unabdingbar ist, ist eine Tabellenkalkulation.

Absprachen zur vorhabenbezogene Konkretisierung

Die Schülerinnen und Schüler erlernen am Beispiel einer Tabellenkalkulation erneut die Verwendung einer formalen Sprache, lernen erste Datentypen kennen und die Notwendigkeit, Befehle syntaktisch korrekt einzugeben. Gerade im Praktikum helfen ihnen die Kenntnisse in Excel oder Calc sehr.

Zeitbedarf: 21 Stunden**Sequenzierung des Unterrichtsvorhabens**

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Einführung in eine Tabellenkalkulation 2. relative und absolute Bezüge 3. Fortgeschrittene Befehle 4. Simulationen mit einer Tabellenkalkulation	Die Schülerinnen und Schüler <ul style="list-style-type: none"> • interpretieren Ergebnisse eines Datenverarbeitungsprozesses (DI), • wählen geeignete Datentypen im Kontext eines Anwendungsbeispiels aus (MI), • verarbeiten Daten mithilfe von Informatiksystemen (MI), • verwenden arithmetische und logische Operationen (MI), • verarbeiten gleichartige Daten mit Hilfe eines geeigneten Werkzeuges (DI), • reflektieren den Entwurfsprozess und beschreiben ihn auch fachsprachlich (A), • überprüfen standardisierte Angaben auf formale Korrektheit (A), • erläutern die Begriffe Syntax und Semantik an Beispielen (A), • erarbeiten sich die Funktionsweise einer Anwendung selbstständig (DI). 	OpenOffice Calc

Unterrichtsvorhaben WP II 9 - V

Thema: Erste textorientierte Programmierung in einer visuellen Programmierumgebung
- Java-Hamster

Leitfragen

Nach der Arbeit mit einer Tabellenkalkulation und den ersten Befehlen an ein Informatiksystem ist es nun an der Zeit, die ersten Schritte in einer textorientierten Programmierumgebung zu gehen. Wir lernen ein Computerprogramm zu schreiben, zu compilieren, auszuführen und zu testen.

Absprachen zur vorhabenbezogene Konkretisierung

Die ersten Schritte in einer textorientierten Programmiersprache erlernen die Schülerinnen und Schüler anhand der visuellen Programmierumgebung Java-Hamster. Die Syntax ist eine reduzierte Form von Java und damit hilfreich für die Programmierung in Java in der Oberstufe. Nebenbei erlernen die Schüler die Modularisierung einer Anwendung in Unterprogramme.

Zeitbedarf: 27 Stunden

Sequenzierung des Unterrichtsvorhabens

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Einführung in die Entwicklungsumgebung 2. Erste kleine Programme 3. Prozeduren 4. Fallunterscheidungen 5. Schleifen 6. Algorithmen 7. Komplexere Programme	Die Schülerinnen und Schüler <ul style="list-style-type: none"> • interpretieren Ergebnisse eines Datenverarbeitungsprozesses (DI), • verwenden arithmetische und logische Operationen (MI), • reflektieren den Entwurfsprozess und beschreiben ihn auch fachsprachlich (A), • stellen Algorithmen in verschiedenen Repräsentationen dar (DI), • implementieren und kommentieren Algorithmen in einer Programmierumgebung (MI), • strukturieren und zerlegen Algorithmen in Teilalgorithmen (MI), • modifizieren Programme (MI). • erstellen syntaktisch korrekte Programme (MI), • überprüfen Handlungsvorschriften auf Eindeutigkeit und Terminierung (A), • beurteilen die Problemangemessenheit eines Algorithmus (A), • analysieren und testen Algorithmen und Programme (MI), • erstellen syntaktisch korrekte Quelltexte in einer Programmiersprache (MI), • analysieren Quelltexte auf syntaktische Korrektheit (A). 	Java-Hamster

3.2.2.2 Konkretisierte Unterrichtsvorhaben WP II Stufe 10**Unterrichtsvorhaben WP II 10 - 0****Thema:** Einführung und Ausblick**Leitfragen**

Was kommt dieses Schuljahr auf uns zu? Was machen wir in Informatik?

Absprachen zur vorhabenbezogene Konkretisierung

Zu Beginn des Schuljahres werden organisatorische Fragen geklärt und es wird ein Überblick über die Themenschwerpunkte des Schuljahres gegeben.

Zeitbedarf: 3 Stunden**Sequenzierung des Unterrichtsvorhabens**

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Organisation und Verwaltung 2. Überblick über die Themen des Schuljahres	Die Schülerinnen und Schüler <ul style="list-style-type: none">• bewerten auf Grundlage ihrer im Informatikunterricht erworbenen Kenntnisse Möglichkeiten der Datenverarbeitung hinsichtlich Chancen und Risiken in ausgewählten Kontexten (A),• geben Beispiele für Auswirkungen von Informatiksystemen auf die Berufswelt und die Lebenswelt im Allgemeinen (A).	

Unterrichtsvorhaben WP II 10 - I**Thema:** Einführung in Python**Leitfragen**

Wie programmiert man einen Computer? Wie funktioniert eine richtige Programmiersprache. Wie schreibt man richtige Programme?

Absprachen zur vorhabenbezogene Konkretisierung

Anhand der Programmiersprache Python erlernen die Schülerinnen und Schüler eine erste richtige Programmiersprache. Python gilt als relativ einfach und ist außerordentlich verbreitet und beliebt. Zudem ermöglicht die Wahl von Python es auch Schülerinnen und Schülern, die kein Informatik im Wahlpflichtbereich belegt haben in der Oberstufe ohne nennenswerte Nachteile mit Java zu beginnen.

Zeitbedarf: 12 Stunden**Sequenzierung des Unterrichtsvorhabens**

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Einführung in Python 2. Kennenlernen der Entwicklungsumgebung 3. Erste Schritte in Python 4. Das Variablenkonzept 5. Unterprogramme und Methoden 6. Fallunterscheidungen 7. Schleifen	Die Schülerinnen und Schüler <ul style="list-style-type: none"> • verarbeiten Daten mithilfe von Informatiksystemen (MI), • verwenden arithmetische und logische Operationen (MI), • modellieren und implementieren eine Anwendung unter Verwendung einer Datenstruktur in einer Programmiersprache (MI). • entwerfen Algorithmen unter Verwendung des Variablenkonzeptes und von Kontrollstrukturen (MI), • reflektieren den Entwurfsprozess und beschreiben ihn auch fachsprachlich (A), • stellen Algorithmen in verschiedenen Repräsentationen dar (DI), • implementieren und kommentieren Algorithmen in einer Programmierumgebung (MI), • strukturieren und zerlegen Algorithmen in Teilalgorithmen (MI), • modifizieren Programme (MI), • erstellen syntaktisch korrekte Programme (MI), • erstellen syntaktisch korrekte Quelltexte in einer Programmiersprache (MI), • analysieren Quelltexte auf syntaktische Korrektheit (A). 	Python Entwicklungsumgebung

Unterrichtsvorhaben WP II 10 - II**Thema:** Algorithmen**Leitfragen**

Was ist ein Algorithmus? Wie entwirft man einen Algorithmus? Welche Darstellungsformen gibt es?

Absprachen zur vorhabenbezogene Konkretisierung

Die Schülerinnen und Schüler lernen, was ein Algorithmus ist, wie man einen Algorithmus entwirft. Sie lernen Programmablaufpläne und Struktogramme kennen. Einfache Standardalgorithmen werden entwickelt.

Zeitbedarf: 9 Stunden**Sequenzierung des Unterrichtsvorhabens**

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Was ist ein Algorithmus? 2. Programmablaufpläne 3. Struktogramme 4. Pseudocode 5. Einfache Algorithmen	Die Schülerinnen und Schüler <ul style="list-style-type: none"> • entwerfen Algorithmen unter Verwendung des Variablenkonzeptes und von Kontrollstrukturen (MI), • reflektieren den Entwurfsprozess und beschreiben ihn auch fachsprachlich (A), • stellen Algorithmen in verschiedenen Repräsentationen dar (DI), • implementieren und kommentieren Algorithmen in einer Programmierumgebung (MI), • strukturieren und zerlegen Algorithmen in Teilalgorithmen (MI), • überprüfen Handlungsvorschriften auf Eindeutigkeit und Terminierung (A), • beurteilen die Problemangemessenheit eines Algorithmus (A), • analysieren und testen Algorithmen und Programme (MI). 	Python Entwicklungsumgebung

Unterrichtsvorhaben WP II 10 - III**Thema:** Algorithmen in Python**Leitfragen**

Wie schreibt man komplexere Programme in Python?

Absprachen zur vorhabenbezogene Konkretisierung

Mit Hilfe der Kenntnisse über Algorithmen lernen die Schülerinnen und Schüler nun komplexere Programme zu schreiben. Der Fokus liegt nun auf den Algorithmen und nicht mehr nur auf den Befehlen.

Zeitbedarf: 18 Stunden**Sequenzierung des Unterrichtsvorhabens**

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Komplexere Probleme mit Python lösen	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> • verarbeiten Daten mithilfe von Informatiksystemen (MI), • verwenden arithmetische und logische Operationen (MI), • modellieren und implementieren eine Anwendung unter Verwendung einer Datenstruktur in einer Programmiersprache (MI). • entwerfen Algorithmen unter Verwendung des Variablenkonzeptes und von Kontrollstrukturen (MI), • reflektieren den Entwurfsprozess und beschreiben ihn auch fachsprachlich (A), • stellen Algorithmen in verschiedenen Repräsentationen dar (DI), • implementieren und kommentieren Algorithmen in einer Programmierumgebung (MI), • strukturieren und zerlegen Algorithmen in Teilalgorithmen (MI), • modifizieren Programme (MI), • erstellen syntaktisch korrekte Programme (MI), • erstellen syntaktisch korrekte Quelltexte in einer Programmiersprache (MI), • analysieren Quelltexte auf syntaktische Korrektheit (A). 	<p>Python</p> <p>Entwicklungsumgebung</p>

Unterrichtsvorhaben WP II 10 - IV**Thema:** Funktionsweise von Netzwerken**Leitfragen**

Wie funktioniert ein Computernetzwerk? Was ist eine ip-Adresse? Was ist ein Router, was ein Gateway? Was ist Schadsoftware, Wie funktioniert Verschlüsselung?

Absprachen zur vorhabenbezogene Konkretisierung

Anhand der Software Filius lernen die Schülerinnen und Schüler nach und nach die Arbeitsweise eines Computernetzwerks kennen. Anschließend werden die Probleme, die sich durch Vernetzung von Informatiksystemen ergeben thematisiert und Verschlüsselung im Netz besprochen.

Zeitbedarf: 12 Stunden**Sequenzierung des Unterrichtsvorhabens**

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Einführung in Filius 2. IP-Adressen und Netzwerkmasken 3. Switche 4. Router 5. Gateways 6. DNS 7. Dienste und Protokolle 8. Webserver 9. Schadsoftware 10. Verschlüsselung	Die Schülerinnen und Schüler <ul style="list-style-type: none"> • benennen Grundkomponenten von Informatiksystemen und • beschreiben ihre Funktionen (DI), • erläutern unterschiedliche Dienste in Netzwerken (KK), • kommunizieren und tauschen Daten mithilfe von Netzen aus (KK), • erarbeiten sich die Funktionsweise einer Anwendung selbstständig (DI), • beschreiben Alltagsgeräte, in denen Informatiksysteme vorkommen (A), • beschreiben die Gefährdung von Daten durch Defekte und Schadsoftware und benennen Maßnahmen zum Schutz von Daten (A), • benennen Maßnahmen zur sicheren Kommunikation in Netzwerken und wenden diese an (DI), • erläutern die Unsicherheit eines einfachen Verschlüsselungsverfahrens (A), • bewerten auf Grundlage ihrer im Informatikunterricht erworbenen Kenntnisse Möglichkeiten der Datenverarbeitung hinsichtlich Chancen und Risiken in ausgewählten Kontexten (A). 	Filius

Unterrichtsvorhaben WP II 10 - V**Thema:** Webseitenerstellung mit HTML**Leitfragen**

Wie schreibt man einen HTML-Auftritt? Was ist CSS, was Java-Script? Was sind dynamische Webseiten? Wie stellt man seine Webseiten in das Internet?

Absprachen zur vorhabenbezogene Konkretisierung

Die Schülerinnen und Schüler schreiben selbstständig einen kompletten Webauftritt. Dazu erhalten sie eine Einführung und Erklärvideos. Am Ende soll ein Webauftritt mit ca. acht untereinander verlinkten Seiten stehen. Formatierungen sollen mittels CSS erfolgen. Nach Möglichkeit werden die Webseiten später auf einen Server hochgeladen.

Zeitbedarf: 30 Stunden**Sequenzierung des Unterrichtsvorhabens**

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Einführung in HTML 2. Einführung in Phase 5 3. Bereitstellung von Material 4. Erläuterungen der Rahmenbedingungen 5. Arbeitsphase 6. Hochladen der Seiten	Die Schülerinnen und Schüler <ul style="list-style-type: none"> • erstellen syntaktisch korrekte Quelltexte in einer geeigneten Dokumentenbeschreibungssprache (MI), • erläutern Prinzipien der strukturierten Dateiverwaltung und wenden diese an (MI), • erarbeiten sich die Funktionsweise einer Anwendung selbstständig (DI), • reflektieren den Entwurfsprozess und beschreiben ihn auch fachsprachlich (A), • analysieren Quelltexte auf syntaktische Korrektheit (A). 	HTML-Editor Phase 5 Irfan View

Unterrichtsvorhaben WP II 10 - VI**Thema:** Rechtliche Grundlagen**Leitfragen**

Was muss ich beachten, wenn ich Webseiten in das Internet einstelle? Wie sind die rechtlichen Vorgaben? Was ist das Urheberrechtsgesetz? Welche relevanten Gesetze gibt es noch? Was sollte ich bezüglich personenbezogener Daten beachten? Was bedeutet Datensparsamkeit?

Absprachen zur vorhabenbezogene Konkretisierung

Die Schülerinnen und Schüler lernen, welche rechtlichen Vorgaben es gibt, wenn Daten im Internet bereitgestellt werden. In diesem Zusammenhang werden auch Fragen zum Urheberrecht, dem Persönlichkeitsrecht, dem Markenrecht usw. besprochen.

Anschließend werden Probleme im Zusammenhang mit persönlichen Daten in sozialen Netzwerken angesprochen.

Zeitbedarf: 6 Stunden**Sequenzierung des Unterrichtsvorhabens**

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Welche rechtlichen Vorgaben gibt es? 2. Welche Strafen drohen? 3. Was darf ich verwenden? 4. Datensparsamkeit 5. Umgang mit persönlichen Daten	Die Schülerinnen und Schüler <ul style="list-style-type: none"> • beschreiben die Gefährdung von Daten durch Defekte und Schadsoftware und benennen Maßnahmen zum Schutz von Daten (A), • analysieren anhand ausgewählter Beispiele, wie personenbezogene Daten verarbeitet und genutzt werden können (DI), • bewerten auf Grundlage ihrer im Informatikunterricht erworbenen Kenntnisse Möglichkeiten der Datenverarbeitung hinsichtlich Chancen und Risiken in ausgewählten Kontexten (A), • benennen ausgewählte rechtliche Rahmenbedingungen des Einsatzes von Informatiksystemen (DI), • beurteilen an ausgewählten Beispielen die gesellschaftlichen Auswirkungen des Einsatzes von Informatiksystemen und berücksichtigen das Recht auf informationelle Selbstbestimmung (A), • geben Beispiele für Auswirkungen von Informatiksystemen auf die Berufswelt und die Lebenswelt im Allgemeinen (A). 	Gesetzestexte

3.3 Sekundarstufe II

3.3.1 Unterrichtsvorhaben

3.3.1.1 Übersichtsraster Unterrichtsvorhaben in der EF

Einführungsphase	
<p>Unterrichtsvorhaben E-I</p> <p>Thema Einführung in das Fach / die Rechner der Schule. Was ist Informatik? Was macht Informatik? Aufbau und Arbeitsweise eines Computers.</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"> - Daten und ihre Strukturierung - Algorithmen - Formale Sprachen und Automaten - Informatiksysteme - Informatik, Mensch und Gesellschaft <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Informatiksysteme der Schule - Gebiete der Informatik - Inhaltsfelder der Informatik - Wirkung der Automatisierung - Zahlensysteme - Geschichte der Datenverarbeitung - Von Neumann Architektur <p>Zeitbedarf: 9 Stunden</p>	<p>Unterrichtsvorhaben E-II</p> <p>Thema Grundlagen der Programmierung mit Java und einfache Algorithmen</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Modellieren - Implementieren - Darstellen und Interpretieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Daten und ihre Strukturierung - Algorithmen - Formale Sprachen und Automaten - Informatiksysteme <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Java-Syntax - Analyse, Entwurf und Implementierung einfacher Algorithmen - Grafische Darstellungen - Pseudocode - Syntax und Semantik einer Programmiersprache <p>Zeitbedarf: 30 Stunden</p>
<p>Unterrichtsvorhaben E-III</p> <p>Thema Entwurf und Implementierung einfacher Such- und Sortierverfahren</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Modellieren - Implementieren - Darstellen und Interpretieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Daten und ihre Strukturierung - Algorithmen - Formale Sprachen und Automaten <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Suchverfahren - Sortierverfahren - Aufwandsbetrachtungen <p>Zeitbedarf: 12 Stunden</p>	<p>Unterrichtsvorhaben E-IV</p> <p>Thema Objektorientierte Modellierung und Implementierung von Klassen- und Objektbeziehungen</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Modellieren - Implementieren - Darstellen und Interpretieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Daten und ihre Strukturierung - Algorithmen - Formale Sprachen und Automaten <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Objekte und Klassen - Analyse, Entwurf und Implementierung - Syntax und Semantik einer Programmiersp. <p>Zeitbedarf: 24 Stunden</p>

Einführungsphase	
<p>Unterrichtsvorhaben E-V</p> <p>Thema Grafische Oberflächen und Programmierung kleinerer Spiele</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none">- Argumentieren- Modellieren- Implementieren- Darstellen und Interpretieren- Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none">- Daten und ihre Strukturierung- Algorithmen- Formale Sprachen und Automaten <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none">- Einführung in GUIs- ActionListener- kleinere GUI-Programme- MVC-Konzept- Spieleprogrammierung <p>Zeitbedarf: 12 Stunden</p>	<p>Summe Einführungsphase: 87 Stunden</p>

3.3.1.2 Übersichtsraster Unterrichtsvorhaben in der Q1

Qualifikationsphase Q1	
<p>Unterrichtsvorhaben Q1-I</p> <p>Thema Wiederholung und Vertiefung der objektorientierten Modellierung</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Modellieren - Implementieren - Darstellen und Interpretieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Daten und ihre Strukturierung - Algorithmen - Informatik, Mensch und Gesellschaft <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Objekte und Klassen - Wirkung der Automatisierung <p>Zeitbedarf: 14 Stunden</p>	<p>Unterrichtsvorhaben Q1-II</p> <p>Thema Organisation und Verarbeitung von Daten I – Modellierung und Implementierung von Anwendungen mit dynamischen und linearen Datenstrukturen</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Modellieren - Implementieren - Darstellen und Interpretieren - Argumentieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Daten und ihre Strukturierung - Algorithmen <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Objekte und Klassen - Syntax und Semantik einer Programmiersprache - Analyse, Entwurf und Impl. von Algorithmen <p>Zeitbedarf: 20 Stunden</p>
<p>Unterrichtsvorhaben Q1-III</p> <p>Thema Algorithmen zum Suchen und Sortieren auf linearen Datenstrukturen</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Darstellen und Interpretieren - Modellieren - Implementieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Algorithmen - Formale Sprachen und Automaten <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Analyse, Entwurf und Implementierung von Algorithmen - Algorithmen in ausgewählten informatischen Kontexten - Syntax und Semantik einer Programmiersprache <p>Zeitbedarf: 20 Stunden</p>	<p>Unterrichtsvorhaben Q1-IV</p> <p>Thema Automaten und formale Sprachen</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Darstellen und Interpretieren - Modellieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Formale Sprachen und Automaten - Informatiksysteme <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Syntax und Semantik einer Programmiersprache - Endliche Automaten - Grammatiken regulärer Sprachen - Möglichkeiten und Grenzen von Automaten und formalen Sprachen <p>Zeitbedarf: 20 Stunden</p>

Summe Qualifikationsphase Q1: 74 Stunden

3.3.1.3 Übersichtsraster Unterrichtsvorhaben in der Q2

Qualifikationsphase Q2	
<p>Unterrichtsvorhaben Q2-I</p> <p>Thema Organisation und Verarbeitung von Daten II – Modellierung und Implementierung von Anwendungen mit dynamischen nicht-linearen Datenstrukturen</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Darstellen und Interpretieren - Modellieren - Implementieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Daten und ihre Strukturierung - Algorithmen - Formale Sprachen und Automaten <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Objekte und Klassen - Analyse, Entwurf und Implementierung von Algorithmen - Algorithmen in ausgewählten informatischen Kontexten <p>Zeitbedarf: 20 Stunden</p>	<p>Unterrichtsvorhaben Q2-II</p> <p>Thema Nutzung und Modellierung von relationalen Datenbanken in Anwendungskontexten</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Modellieren - Implementieren - Darstellen und Interpretieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Daten und ihre Strukturierung - Algorithmen - Formale Sprachen und Automaten - Informatik, Mensch und Gesellschaft <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Datenbanken - Algorithmen in ausgewählten informatischen Kontexten - Syntax und Semantik einer Programmiersprache - Sicherheit <p>Zeitbedarf: 20 Stunden</p>
<p>Unterrichtsvorhaben Q2-III</p> <p>Thema Informatik, Mensch und Gesellschaft</p> <p>Zentrale Kompetenzen</p> <ul style="list-style-type: none"> - Argumentieren - Kommunizieren und Kooperieren <p>Inhaltsfelder</p> <ul style="list-style-type: none"> - Algorithmen - Informatik, Mensch und Gesellschaft <p>Inhaltliche Schwerpunkte</p> <ul style="list-style-type: none"> - Einzelrechner und Rechnernetzwerke - Sicherheit - Nutzung von Informatiksystemen, - Wirkungen der Automatisierung <p>Zeitbedarf: 12 Stunden</p>	<p>Summe Qualifikationsphase Q2: 52 Stunden</p>

3.3.2 Konkretisierte Unterrichtsvorhaben

3.3.2.1 Konkretisierte Unterrichtsvorhaben in der EF

Im Folgenden sollen die in aufgeführten Unterrichtsvorhaben konkretisiert werden. Diese Konkretisierung hat vorschlagenden Charakter, ohne die pädagogische Freiheit des Lehrenden einschränken zu wollen.

Unterrichtsvorhaben EF - I

Thema:

Einführung in das Fach / die Nutzung der Rechner der Schule. Was ist Informatik? Was macht Informatik? Aufbau und Arbeitsweise eines Computers.

Leitfragen

Mit welchen Themen befasst sich das Fach Informatik? Welche Entwicklung durchlief die moderne Datenverarbeitung? Wie funktioniert ein moderner Computer?

Absprachen zur vorhabenbezogene Konkretisierung

Das erste Unterrichtsvorhaben stellt eine allgemeine Einführung in das Fach Informatik dar. Da einige Schülerinnen und Schüler das Fach zum ersten Mal in der Einführungsphase belegen, wird zu Beginn ein Überblick über die Themen des Schulfachs Informatik gegeben und in die Benutzung der Schulrechner eingewiesen. Anschließend werden die Teilbereiche der Informatik und die Inhaltsfelder erläutert.

Danach folgt ein Überblick über die Geschichte der Datenverarbeitung, den Aufbau heutiger Computer und das von Neumann-Prinzip, welches mit einem Modellrechner simuliert wird.

Zeitbedarf: 9 Stunden

Sequenzierung des Unterrichtsvorhabens

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
<p>1. Allgemeine Einführung</p> <ul style="list-style-type: none"> • Übersicht über das Fach • Einführung in die Informatiksysteme der Schule 	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> • nutzen die im Unterricht eingesetzten Informatiksysteme selbstständig, sicher, zielführend und verantwortungsbewusst (D), • nutzen das verfügbare Informatiksystem zur strukturierten Verwaltung und gemeinsamen Verwendung von Daten unter Berücksichtigung der Rechteverwaltung (K), • nutzen das Internet zur Recherche, zum Datenaustausch und zur Kommunikation (K). 	
<p>2. Teilbereiche und Inhaltsfelder der Informatik</p>	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> • bewerten anhand von Fallbeispielen die Auswirkungen des Einsatzes von Informatiksystemen (A). 	
<p>3. Grundlagen von Informatiksystemen und Überblick über die Geschichte der Informatik</p> <ul style="list-style-type: none"> • Geschichte der Datenverarbeitung • Rechneraufbau • Von-Neumann-Architektur 	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> • erläutern wesentliche Grundlagen der Geschichte der digitalen Datenverarbeitung (A). • beschreiben und erläutern den strukturellen Aufbau und die Arbeitsweise singulärer Rechner am Beispiel der „Von-Neumann-Architektur“ (A). 	<p>Videos zum Rechneraufbau, Modellrechner MOPS</p>

Unterrichtsvorhaben EF - II

Thema: Grundlagen der Programmierung mit Java und einfache Algorithmik

Leitfragen

Aus welchen Bausteinen besteht ein Java-Programm? Was sind Variablen? Wie werden Daten im Rechner gespeichert? Welche Datentypen und Kontrollstrukturen stehen in Java zur Verfügung und wie nutzt man diese? Wie verläuft der Entwicklungsprozess eines Java-Programms?

Absprachen zur vorhabenbezogene Konkretisierung

Die Grundlagen zur Programmieretechnik mit Java werden erarbeitet. Dazu zählen: einfache Datentypen, Ausdrücke, Variablenkonzept, Wertzuweisungen, Kontrollstrukturen, Methodenkonzept, Arrays.

Kontrollstrukturen und Algorithmen werden mit Programmablaufplänen und weiteren Diagrammen dargestellt.

Die Schülerinnen und Schüler wenden ihr Wissen kontextbezogen auf kleinere Problemstellungen an. Dabei durchlaufen sie den kontinuierlichen Prozess von Implementieren, Compilieren und Testen.

Zeitbedarf: 30 Stunden

Sequenzierung des Unterrichtsvorhabens

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Einführung in die Programmierung <ul style="list-style-type: none"> • Installation Java und Editor • Kommentare und Stylekonventionen • Variablen und Datentypen • Speicherung von Daten • Casting • IOTools zur Dateneingabe • Methoden ohne Parameter • Methoden mit Parametern • Anfragen und Aufträge • Bedingte Anweisung • Ausdrücke und boolesche Verknüpfungen • Wiederholschleifen • Zählschleifen • Programmablaufpläne und Nassi-Shneiderman-Diagramme • Pseudocode • Eindimensionale Felder • Felder und For-Schleifen • Felder sind Objekte • 2-D-Felder 	Die Schülerinnen und Schüler <ul style="list-style-type: none"> • analysieren und erläutern einfache Algorithmen und Programme (A), • modifizieren einfache Algorithmen und Programme (I), • stellen ganze Zahlen und Zeichen in Binärcodes dar (D), • interpretieren Binärcodes als Zahlen und Zeichen (D), • entwerfen einfache Algorithmen und stellen sie umgangssprachlich und grafisch dar (M), • implementieren Algorithmen unter Verwendung von Variablen und Wertzuweisungen, Kontrollstrukturen sowie Methodenaufrufen (I), • testen Programme schrittweise anhand von Beispielen (I), • implementieren einfache Algorithmen unter Beachtung der Syntax und Semantik einer Programmiersprache (I), • interpretieren Fehlermeldungen und korrigieren den Quellcode (I). • implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I) 	Java-Editor, IOTools

Unterrichtsvorhaben EF - III

Thema: Entwurf und Implementierung einfacher Such- und Sortierverfahren

Leitfragen

Wie kann man die bisherigen Erkenntnisse im Umgang mit Arrays zum Suchen und Sortieren von großen Datenmengen nutzen? Wie effizient ist ein Algorithmus hinsichtlich seines Laufzeitverhaltens?

Absprachen zur vorhabenbezogene Konkretisierung

Aufbauend auf den programmiertechnischen Grundlagen des letzten Unterrichtsvorhabens können die SuS nun ihr Wissen zu Arrays kontextbezogen anwenden: Dazu werden mindestens zwei der einfachen Sortierverfahren behandelt, indem sie simuliert, implementiert und getestet werden. In der anschließenden Analyse reflektieren die SuS das Laufzeitverhalten der Algorithmen im best, worst und average case.

Zeitbedarf: 12 Stunden

Sequenzierung des Unterrichtsvorhabens

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
<p>1. Implementierung und Analyse einfacher Such- und Sortierverfahren auf Arrays</p> <ul style="list-style-type: none"> • Suchverfahren • Sortierverfahren • Aufwandsbetrachtungen 	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> • analysieren Such- und Sortieralgorithmen und wenden sie auf Beispiele an (D), • entwerfen einen weiteren Algorithmus zum Sortieren (M), • beurteilen die Effizienz von Algorithmen am Beispiel von Sortierverfahren hinsichtlich Zeitaufwand und Speicherplatzbedarf (A). • analysieren und erläutern einfache Algorithmen und Programme (A), • implementieren Algorithmen unter Verwendung von Variablen und Wertzuweisungen, Kontrollstrukturen sowie Methodenaufrufen (I), • testen Programme schrittweise anhand von Beispielen (I), • implementieren einfache Algorithmen unter Beachtung der Syntax und Semantik einer Programmiersprache (I), • interpretieren Fehlermeldungen und korrigieren den Quellcode (I). 	<p>Java-Editor, IOTools, Mathe-Prisma, Videos</p>

Unterrichtsvorhaben EF - IV

Thema

Objektorientierte Modellierung und Implementierung von Klassen- und Objektbeziehungen mit dem Java-Editor unter Zuhilfenahme didaktischer Werkzeuge

Leitfragen

Was sind Objekte, was sind Klassen? Wie programmiert man objektorientiert? Was bedeutet Vererbung? Wie lassen sich komplexere Datenflüsse und Beziehungen zwischen Objekten und Klassen realisieren und grafisch darstellen? Wie nutzt man die Theorie der Softwareentwicklung zur Bewältigung komplexerer Aufgaben?

Absprachen zur vorhabenbezogene Konkretisierung

Im Java-Editor werden einfache Klassen und Objekte erzeugt. Gegebenenfalls wird das Prinzip Klasse und Objekt mit BlueJ vertieft.

Anschließend werden Getter, Setter und Konstruktoren eingeführt um die Klassen anschließend mit Anfragen und Aufträgen weiter zu modellieren.

Das Prinzip der Vererbung wird mittels Generalisierung und Spezifizierung erlernt.

Abschließend wird der Softwareentwicklungsprozess mittels Analyse, Entwurfsdiagramm, Implementationsdiagramm, Implementierung und Tests erläutert und an geeigneten Beispielen durchlaufen.

Zeitbedarf: 24 Stunden

Sequenzierung des Unterrichtsvorhabens

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
<p>1. Grundlagen der Entwicklungsumgebung und Einführung in die OO</p> <ul style="list-style-type: none"> • Objekte und Klassen • Konstruktoren • Getter und Setter • Aufträge und Anfragen • Geheimnisprinzip • Objektzustand, Objektdiagramme • Klassendiagramme, Klassendokumentationen • Objekte und Klassen im Kontext ermitteln inkl. Assoziationen • Kommunikation zwischen Objekten • Entwurfsdiagramm, Implementationsdiagramm, UML-Diagramm, Sequenzdiagramme 	<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), • modellieren Klassen unter Verwendung von Vererbung (M), • ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen oder lineare Datensammlungen zu (M), • ordnen Klassen, Attributen und Methoden ihren Sichtbarkeitsbereich zu (M), • stellen den Zustand eines Objekts dar (D), • stellen die Kommunikation zwischen Objekten grafisch dar (M), • stellen Klassen, Assoziations- und Vererbungsbeziehungen in Diagrammen grafisch dar (D), • dokumentieren Klassen durch Beschreibung der Funktionalität der Methoden (D). 	<p>Java-Editor, IOTools, BlueJ</p>

Unterrichtsvorhaben EF - V**Thema**

Grafische Oberflächen und Programmierung kleinerer Spiele

Leitfragen

Wie schreibt man grafische Oberflächen und verknüpft diese mit weiteren Klassen?

Absprachen zur vorhabenbezogene Konkretisierung

Ausgehend von dem bisher erlernten werden mit dem Java-Editor GUIs erzeugt und der ActionListener verwendet. Dabei ist auf die Einhaltung des MVC-Konzeptes zu achten.

Zum Abschluss soll ein kleines Spiel programmiert werden.

Zeitbedarf: 12 Stunden

Sequenzierung des Unterrichtsvorhabens

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
<p>1. Grafische Oberflächen</p> <ul style="list-style-type: none"> • Einführung in GUIs • ActionListener • kleinere GUI-Programme • MVC-Konzept • Spieleprogrammierung 	<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), • modellieren Klassen unter Verwendung von Vererbung (M), • ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen oder lineare Datensammlungen zu (M), • ordnen Klassen, Attributen und Methoden ihren Sichtbarkeitsbereich zu (M), • stellen den Zustand eines Objekts dar (D), • stellen die Kommunikation zwischen Objekten grafisch dar (M), • stellen Klassen, Assoziations- und Vererbungsbeziehungen in Diagrammen grafisch dar (D), • dokumentieren Klassen durch Beschreibung der Funktionalität der Methoden (D), • analysieren und erläutern eine objektorientierte Modellierung (A), • implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I). 	<p>Java-Editor, IOTools, Internet</p>

3.3.2.2 Konkretisierte Unterrichtsvorhaben in der Q1

Unterrichtsvorhaben Q1 - I

Thema

Wiederholung und Vertiefung der objektorientierten Modellierung

Leitfragen

Wie wird aus einem anwendungsbezogenen Sachkontext ein informatisches Klassenmodell entwickelt? Wie werden Attribute, Methoden und Beziehungen identifiziert, den Klassen zugeordnet und dargestellt?

Absprachen zur vorhabenbezogene Konkretisierung

Der bereits bekannte objektorientierte Zugang zu informatischer Modellierung wird von einer allgemeinen Betrachtung dieses informatischen Konzepts auf eine konkrete Problematik übertragen. Anhand dieser wird eine anwendungsbezogene Implementation Schritt für Schritt von der Objektidentifikation über das Entwurfs- und Implementationsdiagramm durchlaufen.

Grundlegende Modellierungskonzepte wie Sichtbarkeiten, Assoziationen, Vererbung sowie deren Darstellung in Entwurfs- und Klassendiagrammen und Dokumentationen werden wiederholt. Ebenso wird erneut die grafische Darstellung von Objektkommunikation thematisiert.

Anhand von Gütekriterien und Eigenschaften von Modellierung entwickeln und bewerten die Schülerinnen und Schüler Klassenentwürfe.

Das Konzept der objektorientierten Modellierung wird um die Idee der abstrakten Klasse sowie um das Subtyping erweitert.

Zeitbedarf: 14 Stunden

Sequenzierung des Unterrichtsvorhabens

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
<p>1. Wiederholung der grundlegenden Konzepte der objektorientierten Programmierung</p> <ul style="list-style-type: none"> • Sichtweise der objektorientierten Informatik auf die Welt • OOP als informatikspezifische Modellierung der Realität • Schritte der Softwareentwicklung <p>2. Erweiterung der objektorientierten Programmierung</p> <ul style="list-style-type: none"> • Umsetzung einer Anforderung in Entwurfs- und Klassendiagramm • Objektkommunikation im Sequenzdiagramm • Klassendokumentation • Umsetzung von Teilen der Modellierung 	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> - analysieren und erläutern objektorientierte Modellierungen (A), - modellieren Klassen mit ihren Attributen, Methoden und ihren Assoziationsbeziehungen unter Angabe von Multiplizitäten (M), - ordnen Klassen, Attributen und Methoden ihre Sichtbarkeitsbereiche zu (M), - modellieren abstrakte und nicht abstrakte Klassen unter Verwendung von Vererbung durch Spezialisieren und Generalisieren (M), - nutzen die Syntax und Semantik einer Programmiersprache bei der Implementierung und zur Analyse von Programmen (I), - wenden eine didaktisch orientierte Entwicklungsumgebung zur Demonstration, zum Entwurf, zur Implementierung und zum Test von Informatiksystemen an (I), - stellen Klassen und ihre 	<p>Schönigh Buch, 2.1 Die Welt ist voller Objekte</p> <p>Projekteinstieg: Klassenentwurf – step by step</p> <p>2.2 Gut geplant – Klassenentwurf</p> <p>2.3 Vererbungshierarchien nutzen</p>

3. Übung und Vertiefung der OOM / OOP	Beziehungen in Diagrammen grafisch dar (D), - dokumentieren Klassen (D), - stellen die Kommunikation zwischen Objekten grafisch dar (D).	
---------------------------------------	--	--

Unterrichtsvorhaben Q1 - II

Thema

Organisation und Verarbeitung von Daten I – Modellierung und Implementierung von Anwendungen mit dynamischen und linearen Datenstrukturen

Leitfragen

Wie müssen Daten linear strukturiert werden, um in den gestellten Anwendungsszenarien eine beliebige Anzahl von Objekten verwalten zu können?

Absprachen zur vorhabenbezogene Konkretisierung

Ausgehend von einigen Alltagsbeispielen werden als Erstes die Anforderungen an eine Datenstruktur erschlossen. Anschließend werden die Möglichkeiten des Arrays untersucht, lineare Daten zu verwalten und über deren Grenzen/Probleme die Vorteile einer dynamischen linearen Struktur am Beispiel der Struktur Queue erarbeitet (Anwendungskontext Warteschlange). Die Klasse *Queue* selbst wird vorgegeben, die Operationen erläutert. Zur Vertiefung der Kenntnisse wird ein weiteres Anwendungsszenario eingeführt (Polizeikontrolle), dessen Lösung modelliert und implementiert wird.

Darauf folgt die Erarbeitung der Struktur Stack, die mithilfe eines einfachen Anwendungsszenarios eingeführt (Biber/Palindrom) wird. Auch hier wird die Klasse *Stack* selbst vorgegeben und die Operationen erläutert. Weitere Aufgaben dienen der Vertiefung und Sicherung.

Um die Unterschiede der beiden Prinzipien FIFO und LIFO zu verstehen, werden zur Lösung der Aufgaben sowohl der Stack als auch die Queue benötigt.

Als letzte lineare dynamische Datenstruktur wird die Liste eingeführt. In dieser Sequenz liegt der Fokus auf der Möglichkeit, auf jedes Element zugreifen zu können. Nachdem die umfangreicheren Standardoperationen dieser Datenstruktur in einem einführenden Beispiel, erarbeitet und in einem weiteren Beispiel vertieft wurden, werden abschließend in einem Anwendungskontext verschiedene lineare Datenstrukturen angewendet. Die Modellierung erfolgt beim gesamten Vorhaben in Entwurfs- und Implementationsdiagrammen.

Zeitbedarf: 20 Stunden

Sequenzierung des Unterrichtsvorhabens

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
<p>1. Die Datenstruktur Feld</p> <ul style="list-style-type: none"> • Erarbeitung der Anforderungen an eine Datenstruktur • Wiederholung der Datenstruktur Array, Eigenschaften der Datenstruktur, Standardoperationen für ein und zweidimensionale Arrays • Modellierung und Implementierung von Anwendungen 	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> - erläutern und begründen methodische Vorgehensweisen, Entwurfs- und Implementationsentscheidungen sowie Aussagen über Informatiksysteme (A) - konstruieren zu kontextbezogenen Problemstellungen informatische Modelle (M) - ermitteln bei der Analyse von Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M) - ordnen Attributen, Parametern und Rückgaben von Methoden einfache 	<p>Kapitel:</p> <ul style="list-style-type: none"> - Anforderungen an eine Datenstruktur - Speichern mit Struktur <p><i>neue Wdh. Aufgabe entwickeln, z. B. eine Chart-Top-10, eine Aufgabe mit zweidimensionalem Array (vgl. Anforderungen KLP)</i></p>

<p>2. Die Datenstruktur Schlange</p> <ul style="list-style-type: none"> • Modellierung und Implementierung der Verknüpfung von Objekten • Generische Typen, Trennung von Verwaltung und Inhalt dyn. DS. • Erläuterung von Problemstellungen, die nach dem FIFO-Prinzip bearbeitet werden • Funktionalität der Schlange unter Verwendung der Klasse <i>Queue</i>; Erschließen der Standardoperationen • Modellierung und Implementierung einer Anwendung auf der Basis einer Anforderungsbeschreibung mit Objekten der Klasse <i>Queue</i> 	<p>Datentypen, Objekttypen sowie lineare und nichtlineare Datensammlungen zu (M),</p> <ul style="list-style-type: none"> - implementieren auf der Grundlage von Modellen oder Modellausschnitten Computerprogramme (I) - testen und korrigieren Computerprogramme (I) - interpretieren Fehlermeldungen und korrigieren den Quellcode (I), - überführen gegebene textuelle und grafische Darstellungen informatischer Zusammenhänge in die jeweils andere Darstellungsform (D) - stellen informatische Modelle und Abläufe in Texten, Tabellen, Diagrammen und Grafiken dar (D) - stellen lineare und nichtlineare Strukturen grafisch dar und erläutern ihren Aufbau (D) - modellieren Klassen mit ihren Attributen, Methoden und ihren Assoziationsbeziehungen unter Angabe von Multiplizitäten (M) 	<p>Kapitel:</p> <ul style="list-style-type: none"> - Wer zuerst kommt - Objekte miteinander verketten - Verwaltung und Inhalt - Funktionen der Queue <p>Aufgaben:</p> <ul style="list-style-type: none"> - Warteschlange Büro <p><i>Erweiterte Queue Verkehrskontrolle (Vertiefung)</i></p> <p><i>Die Polizei kontrolliert die Fahrzeuge im Hinblick auf ihre Verkehrstauglichkeit. Für die Kontrolle werden die Fahrzeuge aus dem Verkehr gewunken. Es werden so lange Fahrzeuge kontrolliert, bis eine gewisse Menge an Verstößen vorliegt oder Autos kontrolliert wurden.</i></p>
<p>3. Die Datenstruktur Stapel</p> <ul style="list-style-type: none"> • Erläuterung von Problemstellungen, die nach dem LIFO-Prinzip bearbeitet werden • Funktionalität der Klasse Stapel unter Verwendung der Klasse <i>Stack</i> • Modellierung und Implementierung einer Anwendung auf Basis einer Anforderungsbeschreibung mit Objekten der Klasse <i>Queue</i> • Modellierung und Implementierung einer Anwendung unter Verwendung verschiedener Datenstrukturen 	<ul style="list-style-type: none"> - ordnen Klassen, Attributen und Methoden ihre Sichtbarkeitsbereiche zu (M) - dokumentieren Klassen (D) - implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I) 	<p>Kapitel:</p> <ul style="list-style-type: none"> - Daten gut abgelegt – Stapel - Funktionen der Datenstruktur Stapel
<p>4. Die Datenstruktur Liste</p> <ul style="list-style-type: none"> • Analyse der Möglichkeiten bisheriger Datenstrukturen zwecks Bestimmung notwendiger Funktionalitäten für komplexere Anwendungen • Erarbeitung der Funktionalität der Liste unter Verwendung der Klasse <i>List</i> • Modellierung und Implementierung einer 		<p>Aufgaben:</p> <ul style="list-style-type: none"> - Standardoperationen/Basiskompetenz (Stapel Münzen/ CDs) zur Umsetzung der gegebenen Funktionen der Klasse <i>Stack</i> - Biber und Teller

<p>Anwendung mit Objekten der Klasse <i>List</i></p> <ul style="list-style-type: none"> • Modellierung und Implementierung einer Anwendung unter Verwendung verschiedener Datenstrukturen <p>5. Übungen und Vertiefungen zur Verwendung linearer und dynamischer Datenstrukturen anhand weiterer Problemstellungen</p>		<p>- <i>Palindrom</i></p> <p>Kapitel:</p> <ul style="list-style-type: none"> - Flexibel für alle Fälle – (die) lineare Liste - Funktionen der Datenstruktur Liste <p>Aufgaben:</p> <ul style="list-style-type: none"> - LEDs
---	--	---

Unterrichtsvorhaben Q1 - III

Thema

Algorithmen zum Suchen und Sortieren auf linearen Datenstrukturen

Leitfragen

Nach welchen Grundprinzipien können Algorithmen strukturiert werden? Welche Qualitätseigenschaften sollten Algorithmen erfüllen? Wie können mithilfe von Such- und Sortieralgorithmen Daten in linearen Strukturen effizient (wieder-) gefunden werden?

Absprachen zur vorhabenbezogene Konkretisierung

Zunächst werden anhand eines Anwendungsbeispiels übergreifende Algorithmeigenschaften (wie Korrektheit, Effizienz und Verständlichkeit) erarbeitet und Schritte der Algorithmenentwicklung wiederholt. Dabei kommen Struktogramme zur Darstellung von Algorithmen zum Einsatz.

Als besondere Struktur von Algorithmen wird die Rekursion an Beispielen veranschaulicht und gegenüber der Iteration abgegrenzt. Rekursive Algorithmen werden von den Schülerinnen und Schülern analysiert und selbst entwickelt.

In der zweiten Unterrichtssequenz geht es um die Frage, wie Daten in linearen Strukturen (lineare Liste und Array) (wieder-) gefunden werden können. Die lineare Suche als iteratives und die binäre Suche als rekursives Verfahren werden veranschaulicht und implementiert. Die Bewertung der Algorithmen erfolgt, indem jeweils die Anzahl der Vergleichsoperationen und der Speicherbedarf ermittelt wird.

Möchte man Daten effizient in einer linearen Struktur wiederfinden, so rückt zwangsläufig die Frage nach einer Sortierstrategie in den Fokus. Es wird mindestens ein iteratives und ein rekursives Sortierverfahren erarbeitet und implementiert sowie ihre Effizienz bewertet.

Zeitbedarf: 20 Stunden

Sequenzierung des Unterrichtsvorhabens

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
<p>1. Eigenschaften von Algorithmen</p> <ul style="list-style-type: none"> • Qualitätseigenschaften von Algorithmen • Strukturierung von Algorithmen mit Hilfe der Strategien „Modulari- 	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> - analysieren und erläutern Algorithmen und Programme (A), - modifizieren Algorithmen und Programme (I), - stellen iterative und rekursive 	<p>4.1 Ohne Algorithmen läuft nichts</p> <p>4.2 Teile die Arbeit – rekursive Algorithmen</p>

<p>sierung“ und „Teile und Herrsche“</p> <ul style="list-style-type: none"> • Analyse und Entwicklung von rekursiven Algorithmen <p>2. Suchen in Listen und Arrays</p> <ul style="list-style-type: none"> • Lineare Suche in Listen und Arrays • Binäre Suche in einem Array • Untersuchung der beiden Verfahren bzgl. Laufzeit und Speicherplatzbedarf <p>3. Sortieren auf Listen und Arrays</p> <ul style="list-style-type: none"> • Entwicklung und Implementierung eines iterativen Sortierverfahrens für eine Liste • Entwicklung und Implementierung eines rekursiven Sortierverfahrens für ein Array • Untersuchung der beiden Verfahren bzgl. Laufzeit und Speicherplatzbedarf 	<p>Algorithmen umgangssprachlich und grafisch dar (D),</p> <ul style="list-style-type: none"> - entwickeln iterative und rekursive Algorithmen unter Nutzung der Strategien „Modularisierung“ und „Teilen und Herrschen“ (M), - implementieren iterative und rekursive Algorithmen auch unter Verwendung von dynamischen Datenstrukturen (I), - testen Programme systematisch anhand von Beispielen (I). - implementieren und erläutern iterative und rekursive Such- und Sortierverfahren (I), - beurteilen die Effizienz von Algorithmen unter Berücksichtigung des Speicherbedarfs und der Zahl der Operationen (A), - beurteilen die syntaktische Korrektheit und die Funktionalität von Programmen (A), - nutzen die Syntax und Semantik einer Programmiersprache bei der Implementierung und zur Analyse von Programmen (I), - interpretieren Fehlermeldungen und korrigieren den Quellcode (I), - testen Programme systematisch anhand von Beispielen (I). 	<p>4.3 Suchen – iterativ und rekursiv</p> <p>4.4 Sortieren – iterativ und rekursiv</p>
---	--	--

Unterrichtsvorhaben Q1 - IV

Thema

Automaten und formale Sprachen

Leitfragen

Wie lassen sich reale Automaten durch ein Modell formal beschreiben? Wie kann die Art und Weise, wie ein Computer Zeichen (Eingaben) verarbeitet, durch Automaten dargestellt werden? Welche Eigenschaften besitzen Automaten und was können sie leisten? Wie werden sie dargestellt? Wie werden reguläre Sprachen durch eine Grammatik beschrieben? In welchem Verhältnis stehen endliche Automaten und Grammatiken? Welche Anwendungsfälle können durch endliche Automaten und Grammatiken regulärer Sprachen beschrieben werden und welche nicht?

Absprachen zur vorhabenbezogene Konkretisierung

Ausgehend von der Beschreibung und Untersuchung realer Automaten wird das formale Modell eines endlichen Automaten entwickelt. Neben dem Mealy-Automaten geht es vor allem um den erkennenden endlichen Automaten. Auf die Erarbeitung der Beschreibung folgt die Modellierung eigener Automaten und die Untersuchung bestehender, um die Eigenschaften und Grenzen eines endlichen Automaten zu erkennen. Hierbei wird dessen Verhalten auf bestimmte Eingaben analysiert.

An den Themenkomplex *Endliche Automaten* schließt sich die Erarbeitung von Grammatiken regulärer Sprachen an. Die Untersuchung beginnt bei die Erschließung der formalen Beschreibung und wird mit der Entwicklung von Grammatiken zu regulären Sprachen fortgeführt. Hierbei wird auch die Beziehung von Grammatiken regulärer Sprachen zu endlichen Automaten an Beispielen erarbeitet und analysiert.

Hierzu gehört auch die Untersuchung, welche Problemstellungen durch endliche Automaten und reguläre Grammatiken beschrieben werden können und welche nicht.

Zeitbedarf: 20 Stunden

Sequenzierung des Unterrichtsvorhabens

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
<p>1. Endliche Automaten</p> <ul style="list-style-type: none"> • Erarbeitung der formalen Beschreibung eines Mealy-Automaten und der Darstellungsformen • Erarbeitung der formalen Beschreibung eines deterministischen endlichen Automaten (DEA) sowie dessen Darstellungsformen; Erschließung der Fachbegriffe Alphabet, Wort, (akzeptierte) Sprache, Determinismus • Analyse der Eigenschaften von DEA durch die Modellierung eines Automaten zu einer gegebenen Problemstellung, der Modifikation eines Automaten sowie die Überführung der gegebenen Darstellungsform in eine andere <p>2. Grammatiken regulärer Sprachen</p> <ul style="list-style-type: none"> • Erarbeitung der formalen Beschreibung einer regulären Grammatik (Sprache, Terminal und Nicht-Terminal, Produktionen und Produktionsvorschriften) • Analyse der Eigenschaften einer regulären Grammatik durch deren Entwicklung und Modellierung zu einer gegebenen Problemstellung. <p>3. Übungen und Vertiefungen</p> <ul style="list-style-type: none"> • Verwendung endlicher Automaten und Grammatiken regulärer Sprachen 	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> - stellen informatische Modelle und Abläufe in Texten, Tabellen, Diagrammen und Grafiken dar (D) - analysieren und erläutern die Eigenschaften endlicher Automaten einschließlich ihres Verhaltens bei bestimmten Eingaben (A) - ermitteln die Sprache, die ein endlicher Automat akzeptiert (D) - entwickeln und modifizieren zu einer Problemstellung endlicher Automaten (M) - stellen endliche Automaten in Tabellen und Graphen dar und überführen sie in die jeweils andere Darstellungsform (D) - entwickeln zur Grammatik einer regulären Sprache einen zugehörigen endlichen Automaten (M) - analysieren und erläutern Grammatiken regulärer Sprachen (A) - modifizieren Grammatiken regulärer Sprachen (M) - entwickeln zu einer regulären Sprache eine Grammatik, die die Sprache erzeugt (M) - entwickeln zur akzeptierten Sprache eines Automaten eine zugehörige Grammatik (M) - beschreiben an Beispielen den Zusammenhang zwischen Automaten und Grammatiken (D) - zeigen die Grenzen endlicher Automaten und regulärer Grammatiken auf 	<p>Vom realen Automaten zum Modell Projekteinstieg: Schatzsuche Der Mealy-Automat</p> <p>Der erkennende endliche Automat</p> <p>Wort und Sprache</p> <p>Grammatiken regulärer Sprachen</p> <p>Prüfungsvorbereitung</p>

3.3.2.3 Konkretisierte Unterrichtsvorhaben in der Q2

Unterrichtsvorhaben Q2 - I

Thema

Organisation und Verarbeitung von Daten II – Modellierung und Implementierung von Anwendungen mit dynamischen nicht-linearen Datenstrukturen

Leitfragen

Wie können Daten mithilfe von Baumstrukturen verwaltet werden? Wie können mit binären Suchbäumen Inhalte sortiert verwaltet werden und welche Vor- und Nachteile bietet dies?

Absprachen zur vorhabenbezogene Konkretisierung

Anhand des Anwendungskontextes Spielbäume werden zunächst der generelle Aufbau von Baumstrukturen (auch nicht-binäre) und wichtige Grundbegriffe erarbeitet. Die Darstellung von Bäumen mit Knoten und Kanten wird eingeführt.

Anschließend rückt der Fokus auf die binären Bäume, deren rekursiver Aufbau für die Traversierung der Datenstruktur genutzt wird. Die Preorder-Traversierung wird verwendet, um einen gespeicherten Inhalt in einem Binärbaum zu finden (Tiefensuche).

Der Anwendungskontext Ahnenbaum wird mithilfe der Klasse BinaryTree (der Materialien für das Zentralabitur in NRW) modelliert und (ggf. in Teilen) implementiert. Dabei wird u. a. die Erzeugung eines Binärbaums mithilfe der Konstruktoren der Klasse BinaryTree thematisiert.

Möchte man Daten geordnet speichern, so bietet sich die Struktur des binären Suchbaums an. An Beispielen wird zunächst das Prinzip des binären Suchbaums erarbeitet. Die Operationen des Suchens, Einfügens, Löschens und der sortierten Ausgabe werden thematisiert.

Um Daten in einem Anwendungskontext mithilfe eines binären Suchbaums verwalten zu können, müssen sie in eine Ordnung gebracht werden können, d. h. sie müssen vergleichbar sein. Diese Vorgabe wird mithilfe des Interfaces oder einer abstrakten Klasse realisiert.

Zeitbedarf: 20 Stunden

Sequenzierung des Unterrichtsvorhabens

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
<p>1. Aufbau von Baumstrukturen und Grundbegriffe</p> <ul style="list-style-type: none"> • Erarbeitung der Begriffe Wurzel, Knoten, Blatt, Kante, Grad eines Knotens und eines Baumes, Pfad, Tiefe, Ebene, Teilbaum • Aufbau und Darstellung von Baumstrukturen in verschiedenen Anwendungskontexten <p>2. Binäre Bäume</p> <ul style="list-style-type: none"> • rekursiver Aufbau eines binären Baums • Traversierungen (pre-, in-, postorder) • Modellierung eines Binärbaums in einem Anwendungskontext mit Hilfe der Klasse BinaryTree • Implementation einer Anwendung der Datenstruktur binärer Baum (ggf. in Teilen) <p>3. Binäre Suchbäume</p> <ul style="list-style-type: none"> • Prinzip des binären Suchbaums, Ordnungsrelation • Operationen auf dem binären Suchbaum (Suchen, Einfügen, Löschen, sortierte Ausgabe) • Modellierung eines binären Suchbaums in einem Anwendungskontext mit Hilfe der Klasse BinarySearchTree (als Entwurfs- und Implementationsdiagramm) • Implementation einer Anwendung der Datenstruktur binärer Suchbaum 	<p>Die Schülerinnen und Schüler</p> <ul style="list-style-type: none"> - stellen lineare und nichtlineare Strukturen grafisch dar und erläutern ihren Aufbau (D), - erläutern Operationen dynamischer (linearer oder nicht-linearer) Datenstrukturen (A), - analysieren und erläutern Algorithmen und Programme (A), - stellen iterative und rekursive Algorithmen umgangssprachlich und grafisch dar (D). - beurteilen die syntaktische Korrektheit und die Funktionalität von Programmen (A), - ermitteln bei der Analyse von Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M), - ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen sowie lineare und nichtlineare Datensammlungen zu (M), - modellieren abstrakte und nicht abstrakte Klassen unter Verwendung von Vererbung durch Spezialisieren und Generalisieren (M), - verwenden bei der Modellierung geeigneter Problemstellungen die Möglichkeiten der Polymorphie (M), - entwickeln iterative und rekursive Algorithmen unter Nutzung der Konstruktionsstrategien „Modularisierung“ und „Teilen und Herrschen“ (M), - implementieren iterative und rekursive Algorithmen auch unter Verwendung von dynamischen Datenstrukturen (I), - modifizieren Algorithmen und Programme (I), - nutzen die Syntax und Semantik einer Programmiersprache bei der Implementierung und zur Analyse von Programmen (I), - interpretieren Fehlermeldungen und korrigieren den Quellcode (I), - testen Programme systematisch anhand von Beispielen (I), 	<p>6.1 Spielen mit Struktur – Baumstrukturen</p> <p>Projekteinstieg 1: Spielbäume</p> <p>6.2 Zwei Nachfolger sind genug! - Binäre Bäume</p> <p>Implementation des Projekts Ahnenbaum</p> <p>6.3 Wer Ordnung hält, spart Zeit beim Suchen – Binäre Suchbäume</p> <p>Projekteinstieg 2: Binäre Suchbäume</p> <p>Implementation des Projekts Benutzerverwaltung</p> <p>Prüfungsvorbereitung</p>

Unterrichtsvorhaben Q2 - II

Thema

Nutzung und Modellierung von relationalen Datenbanken in Anwendungskontexten

Leitfragen

Was sind Datenbanken und wie kann man mit ihnen arbeiten? Wie entwickelt man selbst eine Datenbank für einen Anwendungskontext?

Absprachen zur vorhabenbezogene Konkretisierung

Am Beispiel eines Online-Buchhandels wird der Aufbau einer Datenbank sowie wichtige Grundbegriffe erarbeitet. Die Schülerinnen und Schüler nehmen dabei zunächst die Sicht der Anwender an, die eine bestehende Datenbank beschreiben und analysieren und mithilfe von SQL-Abfragen Daten gezielt herausfiltern.

Mithilfe des Projekteinstiegs „Tabellen“ können bereits zu einem frühen Zeitpunkt des Unterrichtsvorhabens Redundanzen, Inkonsistenzen und Anomalien problematisiert werden.

Nachdem die Lernenden in der ersten Sequenz mit Datenbanken vertraut gemacht wurden, nehmen sie nun die Rolle der Entwickler an, indem sie selbst Datenbanken von Grund auf modellieren und das Modell in ein Relationenschema überführen. Sie arbeiten mit Entity-Relationship-Diagrammen, um Entitäten, zugehörige Attribute, Relationen und Kardinalitäten in Anwendungskontexten darzustellen. Gegebene ER-Diagramme werden analysiert, erläutert und modifiziert.

Der bereits in der ersten Sequenz problematisierte Begriff der Redundanz wird am Ende des Unterrichtsvorhabens wieder aufgegriffen, um die Normalisierung von Datenbanken zu thematisieren.

Bestehende Datenbankschemata werden hinsichtlich der 1. bis 3. Normalform untersucht und (soweit nötig) normalisiert.

Zeitbedarf: 20 Stunden

Sequenzierung des Unterrichtsvorhabens

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Nutzung von relationalen Datenbanken <ul style="list-style-type: none"> • Aufbau von Datenbanksystemen und Grundbegriffe • Aufgaben und Eigenschaften eines Datenbanksystems • Erarbeitung der Begriffe Tabelle, Attribut, Attributwert, Datensatz, Datentyp, Primärschlüssel, Datenbankschema • Problematisierung von Redundanzen, Anomalien und Inkonsistenzen • SQL-Abfragen • Erarbeitung der grundlegenden Sprachelemente von 	Die Schülerinnen und Schüler <ul style="list-style-type: none"> - erläutern die Eigenschaften und den Aufbau von Datenbanksystemen unter dem Aspekt der sicheren Nutzung (A), - analysieren und erläutern die Syntax und Semantik einer Datenbankabfrage (A), - analysieren und erläutern eine Datenbankmodellierung (A), - erläutern die Eigenschaften normalisierter Datenbankschemata (A), - bestimmen Primär- und Sekundärschlüssel (M), - ermitteln für anwendungsbezogene Problemstellungen Entitäten, zugehörige Attribute, Relationen und Kardinalitäten (M), - modifizieren eine Datenbankmodellierung (M), - modellieren zu einem Entity-Relationship-Diagramm ein relationales Datenbankschema (M), 	Kapitel 8 Datenbanken <ul style="list-style-type: none"> 8.1 Wissen speichern und verwalten – Datenbanksysteme 8.2 Daten anordnen mit Tabellen Beispiel: Buchhandlung Redundanzen, Anomalien und Inkonsistenzen Projekteinstieg: Tabellen 8.3 Daten filtern mit SQL 8.4 Komplexe Filter 8.5 Datenbankentwurf

<p>SQL (SELECT(DISTINCT), FROM, WHERE, JOIN)</p> <ul style="list-style-type: none"> Analyse und Erarbeitung von SQL-Abfragen (AND, OR, NOT, UNION, AS, GROUP BY, ORDER BY, ASC, DESC, COUNT, MAX, MIN, SUM, Arithmetische Operatoren: +, -, *, /, (...), Vergleichsoperatoren: =, <>, >, <, >=, <=, LIKE, BETWEEN, IN, IS NULL, geschachtelte Select-Ausdrücke) Vertiefung an einem weiteren Datenbankbeispiel <p>2. Modellierung von relationalen Datenbanken</p> <ul style="list-style-type: none"> Datenbankentwurf durch ER-Diagramme Ermittlung von Entitäten, zugehörigen Attributen, Beziehungen und Kardinalitäten in Anwendungssituationen und Modellierung eines Datenbankentwurfs in Form eines Entity-Relationship-Diagramms Erläuterung und Erweiterung einer Datenbankmodellierung Entwicklung eines relationalen Modells aus einem Datenbankentwurf Überführung eines Entity-Relationship-Diagramms in ein relationales Datenbankschema inklusive der Bestimmung von Primär- und Fremdschlüsseln Normalformen Überprüfung von Datenbankschemata hinsichtlich der 1. bis 3. Normalform und Normalisierung <p>3. Übung und Vertiefung der Nutzung und Modellierung von relationalen Datenbanken</p>	<ul style="list-style-type: none"> überführen Datenbankschemata in vorgegebene Normalformen (M), verwenden die Syntax und Semantik einer Datenbankabfragesprache, um Informationen aus einem Datenbanksystem zu extrahieren (I), ermitteln Ergebnisse von Datenbankabfragen über mehrere verknüpfte Tabellen (D), stellen Entitäten mit ihren Attributen und die Beziehungen zwischen Entitäten in einem Entity-Relationship-Diagramm grafisch dar (D), überprüfen Datenbankschemata auf vorgegebene Normalisierungseigenschaften (D). 	<p><i>Beispiel: Online-Buchhandel</i></p> <p><i>Datenanalyse und Entwurf</i></p> <p><i>8.6 Umsetzung des ER-Modells</i></p> <p><i>Entitätsmengen</i></p> <p><i>m:n-Beziehungen</i></p> <p><i>1:n-Beziehungen</i></p> <p><i>1:1-Beziehungen</i></p> <p><i>Wiederaufgriff des Projekteinstiegs</i></p> <p><i>8.7 Datenbanken verbessern durch Normalformen</i></p> <p><i>Prüfungsvorbereitung</i></p>
---	---	---

Unterrichtsvorhaben Q2 - III**Thema**

Informatik, Mensch und Gesellschaft

Leitfragen

Welche Chancen und welche Risiken gehen mit der fortschreitenden Entwicklung der Informatik einher? Was ist rechtlich erlaubt, was ist verboten? Welche ethisch und moralischen Fragen stellen sich? Wie kann ich mich schützen? Dürfen Programmierer alles was erlaubt ist?

Zeitbedarf: 12 Stunden

Sequenzierung des Unterrichtsvorhabens

Unterrichtssequenzen	zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Rechtliche Grundlagen 2. Ethische und moralische Aspekte 3. Auswirkungen auf den Einzelnen und die Gesellschaft 4. Datensparsamkeit	Die Schülerinnen und Schüler - 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), - 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).	Filme: - Citizenfour - David Kriesel: Traue keinem Scan, den du nicht selbst gefälscht hast - SpiegelMining – Reverse Engineering von Spiegel-Online (33c3)

4 Grundsätze der fachmethodischen und fachdidaktischen Arbeit

Unter Berücksichtigung des Schulprogramms hat die Fachkonferenz Informatik der Kaiserin-Theophanu-Schule die folgenden fachmethodischen und fachdidaktischen Grundsätze beschlossen. In diesem Zusammenhang beziehen sich die überfachlichen Grundsätze auf fächerübergreifende Aspekte, die auch Gegenstand der Qualitätsanalyse sind, die darauf folgenden Grundsätze sind fachspezifisch angelegt.

Überfachliche Grundsätze:

- Geeignete Problemstellungen zeichnen die Ziele des Unterrichts vor und bestimmen die Struktur der Lernprozesse.
- Inhalt und Anforderungsniveau des Unterrichts entsprechen dem Leistungsvermögen der Schüler/innen.
- Die Unterrichtsgestaltung ist auf die Ziele und Inhalte abgestimmt.
- Medien und Arbeitsmittel sind schülernah gewählt.
- Die Schüler/innen erreichen einen Lernzuwachs.
- Der Unterricht fördert eine aktive Teilnahme der Schüler/innen.
- Der Unterricht fördert die Zusammenarbeit zwischen den Schülern/innen und bietet ihnen Möglichkeiten zu eigenen Lösungen.
- Der Unterricht berücksichtigt die individuellen Lernwege der einzelnen Schüler/innen.
- Die Schüler/innen erhalten Gelegenheit zu selbstständiger Arbeit und werden dabei unterstützt.
- Der Unterricht fördert strukturierte und funktionale Partner- bzw. Gruppenarbeit.
- Der Unterricht fördert strukturierte und funktionale Arbeit im Plenum.
- Die Lehr- und Lernzeit wird intensiv für Unterrichtszwecke genutzt.
- Es herrscht ein positives pädagogisches Klima im Unterricht.

Fachliche Grundsätze:

- Der Unterricht unterliegt der Wissenschaftsorientierung und ist dementsprechend eng verzahnt mit seiner Bezugswissenschaft.
- Der Unterricht ist problemorientiert und soll von realen Problemen ausgehen und sich auf solche rückbeziehen.
- Der Unterricht folgt dem Prinzip der Exemplarizität und soll ermöglichen, informatische Strukturen und Gesetzmäßigkeiten in den ausgewählten Problemen und Projekten zu erkennen.
- Der Unterricht ist anschaulich sowie gegenwarts- und zukunftsorientiert und gewinnt dadurch für die Schülerinnen und Schüler an Bedeutsamkeit.
- Der Unterricht ist handlungsorientiert, d.h. projekt- und produktorientiert angelegt.
- Im Unterricht werden sowohl für die Schule didaktisch reduzierte als auch reale Informatiksysteme aus der Wissenschafts-, Berufs- und Lebenswelt eingesetzt.
- Der Unterricht beinhaltet reale Begegnung mit Informatiksystemen.

5 Grundsätze der Leistungsbewertung und Leistungsrückmeldung

Schulische Leistungsbewertung steht im Spannungsfeld pädagogischer und gesellschaftlicher Zielsetzung. Unter pädagogischen Gesichtspunkten hat sie vornehmlich das Individuum im Blick. Hier soll sie über den Leistungszuwachs rückmelden und dadurch die Motivation für weitere Anstrengungen erhöhen. Sie ermöglicht den Schülerinnen und Schülern ihre noch vorhandenen fachlichen Defizite wie auch ihre Stärken und Fähigkeiten zu erkennen um dadurch ein realistisches Selbstbild aufzubauen. Sie ist Basis für gezielte individuelle Förderung.

Für die Erziehungsberechtigten sind Noten eine einfache und zentrale Information zum Leistungsstand ihrer Kinder. Sie bieten den Anlass, über die Ursache von Defiziten und über die Beseitigung von Lernschwierigkeiten verschiedenster Art Rücksprache zu halten. Noten sind zudem Grundlage und Anlass, in den pädagogischen Konferenzen über die Schwierigkeiten und besonderen Probleme einzelner Schüler wie auch Klassen zu beraten und Maßnahmen zur Verbesserung zu beschließen.

Schulische Leistungsbewertung ist eingebettet in die durch das Schulgesetz § 48 (Grundsätze der Leistungsbewertung), APO-SI §6, APO - GOST §13 bis §17 sowie Kapitel 3 des Kernlehrplans Informatik vorgegebene Grundsätze und Verfahren. Daraus erwächst für die Schulen konkret die Aufgabe, sowohl die individuellen Schwächen und Stärken der Schüler zu diagnostizieren und gegebenenfalls die Defizite durch gezielte Maßnahmen zu beseitigen sowie besondere Begabungen zu fördern.

Die gesellschaftliche Funktion von Noten zu erfüllen ist der Schule aufgegeben. Noten entscheiden mit über Schullaufbahnen, Versetzungen und Abschlüsse. Zeugnisse sind mit entscheidender Parameter bei der Zuteilung von Berufs- und Lebenschancen. Daraus erwachsen für die Beurteilenden eine besondere Verantwortung und die Pflicht einer größtmöglichen Objektivität bei der Notenfindung.

Die Fachkonferenz Informatik legt die Kriterien für die Leistungsbeurteilung fest. Die Lehrerinnen und Lehrer machen diese Kriterien den Schülerinnen und Schülern transparent.

Es gelten folgende Grundsätze der Leistungsbewertung:

- Lernerfolgsüberprüfungen sind ein kontinuierlicher Prozess.
- Bewertet werden alle im Zusammenhang mit dem Unterricht erbrachten Leistungen (schriftliche Arbeiten, mündliche Beiträge, praktische Leistungen).
- Die Lehrperson gibt den Schülerinnen und Schülern im Unterricht hinreichend Gelegenheit, die entsprechenden Anforderungen der Leistungsbewertung im Unterricht in Umfang und Anspruch kennenzulernen und sich auf sie vorzubereiten.
- Bewertet werden der Umfang, die selbstständige und richtige Anwendung der Kenntnisse, Fähigkeiten und Fertigkeiten sowie die Art der Darstellung.

5.1 Beurteilungsbereich Klausuren

Klausuren dienen der schriftlichen Überprüfung der Lernergebnisse einer vorausgegangenen Unterrichtsreihe. Sie sind so anzulegen, dass Sachkenntnisse und methodische Fertigkeiten nachgewiesen werden können. Sie bedürfen einer angemessenen Vorbereitung und verlangen klare Aufgabenstellungen. Im Umfang und Anforderungsniveau sind Kursarbeiten bzw. Klausuren abhängig von den kontinuierlich ansteigenden Anforderungen entsprechend dem Lehrplan.

Es ist darauf zu achten, dass nicht nur die Richtigkeit der Ergebnisse und die inhaltliche Qualität, sondern auch die angemessene Form der Darstellung unabdingbare Kriterien der Bewertung der geforderten Leistung sind.

Die Fachkonferenz legt die Dauer der Kursarbeiten und Klausuren fest. An der Kaiserin-Theophanu-Schule gelten für die Sekundarstufe I und II folgende Regelungen:

Stufe	Klausur 1	Klausur 2	Klausur 3	Klausur 4
6	hier werden keine Klassenarbeiten geschrieben			
WP 9	1-2 stündig	1-2 stündig	1-2 stündig	1-2 stündig
WP 10	1-2 stündig	1-2 stündig	1-2 stündig	1-2 stündig
EF		2-stündig (90 min)	2-stündig (90 min)	
Q1	2-stündig (90 min)	2-stündig (90 min)	2-stündig (90 min)	2-stündig (90 min)
Q2	3-stündig (135 min)	3-stündig (135 min)		

Im Wahlpflichtbereich kann pro Schuljahr eine Klassenarbeit durch eine Projektarbeit ersetzt werden.

In der Einführungsphase wird nur eine Klausur im Schulhalbjahr geschrieben. Nach Möglichkeit soll dies die 2. und 3. Klausur sein, um den Schülern frühzeitig eine ausreichende Rückmeldung über ihren Leistungsstand zu geben, damit sie eine Entscheidungsgrundlage für die Wahl ihrer Fächer in der Qualifikationsphase haben.

In der Q1 kann die dritte Klausur durch eine Facharbeit ersetzt werden.

Die Bewertung der schriftlichen Leistungen in Klausuren erfolgt über ein Raster mit Hilfspunkten, die im Erwartungshorizont den einzelnen Kriterien zugeordnet sind.

Spätestens ab der Qualifikationsphase orientiert sich die Zuordnung der Hilfspunktsumme zu den Notenstufen an dem Zuordnungsschema des Zentralabiturs.

Von diesem kann aber im Einzelfall begründet abgewichen werden, wenn sich z.B. besonders originelle Teillösungen nicht durch Hilfspunkte gemäß den Kriterien des Erwartungshorizontes abbilden lassen oder eine Abwertung wegen besonders schwacher Darstellung (APO-GOST §13 (2)) angemessen erscheint.

5.2 Beurteilungsbereich Sonstige Mitarbeit

Den Schülerinnen und Schülern werden die Kriterien zum Beurteilungsbereich „sonstige Mitarbeit“ zu Beginn des Schuljahres genannt.

Mündliche Leistungen

- Beteiligung am Unterrichtsgespräch
- Zusammenfassungen zur Vor- und Nachbereitung des Unterrichts
- Präsentation von Arbeitsergebnissen
- Mitarbeit in Partner-/Gruppenarbeitsphasen

Praktische Leistungen am Computer

- Implementierung, Test und Anwendung von Informatiksystemen

Sonstige schriftliche Leistungen

- Lernerfolgsüberprüfung durch kurze schriftliche Übungen Schriftliche Übung dauern ca. 20 Minuten und umfassen den Stoff der letzten ca. 4–6 Stunden.
- Bearbeitung von schriftlichen Aufgaben im Unterricht

Die folgenden allgemeinen Kriterien gelten sowohl für die mündlichen als auch für die schriftlichen Formen der sonstigen Mitarbeit.

Die Bewertungskriterien stützen sich auf

- die Qualität der Beiträge,
- die Quantität der Beiträge und
- die Kontinuität der Beiträge.

Besonderes Augenmerk ist dabei auf

- die sachliche Richtigkeit,
- die angemessene Verwendung der Fachsprache,
- die Darstellungskompetenz,
- die Komplexität und den Grad der Abstraktion,
- die Selbstständigkeit im Arbeitsprozess,
- die Präzision und die Differenziertheit der Reflexion zu legen.

Bei Gruppenarbeiten auch auf

- das Einbringen in die Arbeit der Gruppe,
- die Durchführung fachlicher Arbeitsanteile und
- die Qualität des entwickelten Produktes.

Bei Projektarbeit darüber hinaus auf

- die Dokumentation des Arbeitsprozesses,
- den Grad der Selbstständigkeit,
- die Reflexion des eigenen Handelns und
- die Aufnahme von Beratung durch die Lehrkraft.

5.3 Grundsätze der Leistungsrückmeldung und Beratung

Die Grundsätze der Leistungsbewertung werden zu Beginn eines jeden Schuljahres den Schülerinnen und Schülern transparent gemacht. Leistungsrückmeldungen können erfolgen

- nach einer mündlichen Überprüfung,
- bei Rückgabe von schriftlichen Leistungsüberprüfungen,
- nach Abschluss eines Projektes,
- nach einem Vortrag oder einer Präsentation,
- bei auffälligen Leistungsveränderungen,
- auf Anfrage,
- als Quartalsfeedback und
- zu Eltern- oder Schülersprechtagen.

Die Leistungsrückmeldung kann

- durch ein Gespräch mit der Schülerin oder dem Schüler,
- durch einen Feedbackbogen,
- durch die schriftliche Begründung einer Note oder
- durch eine individuelle Lern-/Förderempfehlung

erfolgen.

Leistungsrückmeldungen erfolgen auch in der Einführungsphase im Rahmen der kollektiven und individuellen Beratung zur Wahl des Faches Informatik in der Qualifikationsphase.

5.4 Individuelle Förderung

Der Informatikunterricht lebt von der verantwortungsvollen und selbständigen Arbeit der Schülerinnen und Schüler, so dass die Lehrperson die nötige Zeit hat, bei Bedarf gezielt und individuell zu fördern. Leistungsstärkere Schülerinnen und Schüler können ihr Wissen anhand von vertiefenden Problemstellungen erweitern.

5.5 Bildung der Zeugnisnote

In die Note gehen alle im Unterricht erbrachten Leistungen ein. Dabei nimmt die Beurteilung der Klausuren den gleichen Stellenwert wie die Leistungen im Bereich der Mitarbeit im Unterricht ein. Zudem ist bei der Notenfindung die individuelle Lernentwicklung der Schülerinnen und Schüler angemessen zu berücksichtigen.

6 Entscheidungen zu fach- und unterrichtsübergreifenden Fragen

Die Fachkonferenz Informatik hat sich im Rahmen des Schulprogramms für folgende zentrale Schwerpunkte entschieden:

Zusammenarbeit mit anderen Fächern

Im Informatikunterricht werden Kompetenzen anhand informatischer Inhalte in verschiedenen Anwendungskontexten erworben, in denen Schülerinnen und Schülern aus anderen Fächern Kenntnisse mitbringen können. Diese können insbesondere bei der Auswahl und Bearbeitung von Softwareprojekten berücksichtigt werden und in einem hinsichtlich der informatischen Problemstellung angemessenem Maß in den Unterricht Eingang finden.

Projekttag

Alle zwei Jahre werden am der Kaiserin-Theophanu-Schule Projekttag angeboten. Die Fachkonferenz Informatik bietet nach Möglichkeit in diesem Zusammenhang mindestens ein Projekt für Schülerinnen und Schüler der gymnasialen Oberstufe an.

Vorbereitung auf die Erstellung der Facharbeit

Möglichst schon im zweiten Halbjahr der Einführungsphase, spätestens jedoch im ersten Halbjahr des ersten Jahres der Qualifikationsphase werden im Unterricht an geeigneten Stellen Hinweise zur Erstellung von Facharbeiten gegeben. Das betrifft u. a. Themenvorschläge, Hinweise zu den Anforderungen und zur Bewertung. Es wird vereinbart, dass nur Facharbeiten vergeben werden, die mit der eigenständigen Entwicklung eines Softwareproduktes verbunden sind.

Exkursionen

Ein Besuch der Ausstellung „Rechnen einst und heute“ des Arithmeums in Bonn ist wünschenswert.

7 Qualitätssicherung und Evaluation

Das schulinterne Curriculum stellt keine starre Größe dar, sondern ist als „lebendes Dokument“ zu betrachten. Dementsprechend sind die Inhalte stetig zu überprüfen, um ggf. Modifikationen vornehmen zu können.

Die Fachkonferenz (als professionelle Lerngemeinschaft) trägt durch diesen Prozess zur Qualitätsentwicklung und damit zur Qualitätssicherung des Faches bei.

Durch Diskussion der Aufgabenstellung von Klausuren in Fachdienstbesprechungen und eine regelmäßige Erörterung der Ergebnisse von Leistungsüberprüfungen wird ein hohes Maß an fachlicher Qualitätssicherung erreicht.

Jeweils einmal im Schuljahres werden in einer Sitzung der Fachkonferenz für die nachfolgenden Jahrgänge zwingend erforderlich erscheinende Veränderungen diskutiert und ggf. beschlossen, um erkannten ungünstigen Entscheidungen schnellstmöglich entgegenwirken zu können.

8 Eingeführte Lehrwerke

Informatik in der Sekundarstufe I:


- C. C. Buchner Informatik 5/6, ISBN: 978-3-661-38041-4

Informatik in der Sekundarstufe II:

- Informatik - Lehrwerk für die gymnasiale Oberstufe - Ausgabe 2014, ISBN: 978-3-14-037127-8

9 Bezug zum Medienkompetenzrahmen

1. BEDIENEN UND ANWENDEN 	2. INFORMIEREN UND RECHERCHIEREN 	3. KOMMUNIZIEREN UND KOOPERIEREN 
1.1 Medianausstattung (Hardware)	2.1 Informationsrecherche	3.1 Kommunikations- und Kooperationsprozesse
<p>Medianausstattung (Hardware) kennen, auswählen und reflektiert anwenden; mit dieser verantwortungsvoll umgehen</p>	<p>Informationsrecherchen zielgerichtet durchführen und dabei Suchstrategien anwenden</p>	<p>Kommunikations- und Kooperationsprozesse mit digitalen Werkzeugen zielgerichtet gestalten sowie mediale Produkte und Informationen teilen</p>
1.2 Digitale Werkzeuge	2.2 Informationsauswertung	3.2 Kommunikations- und Kooperationsregeln
<p>Verschiedene digitale Werkzeuge und deren Funktionsumfang kennen, auswählen sowie diese kreativ, reflektiert und zielgerichtet einsetzen</p>	<p>Themenrelevante Informationen und Daten aus Medienangeboten filtern, strukturieren, umwandeln und aufbereiten</p>	<p>Regeln für digitale Kommunikation und Kooperation kennen, formulieren und einhalten</p>
1.3 Datenorganisation	2.3 Informationsbewertung	3.3 Kommunikation und Kooperation in der Gesellschaft
<p>Informationen und Daten sicher speichern, wiederfinden und von verschiedenen Orten abrufen; Informationen und Daten zusammenfassen, organisieren und strukturiert aufbewahren</p>	<p>Informationen, Daten und ihre Quellen sowie dahinterliegende Strategien und Absichten erkennen und kritisch bewerten</p>	<p>Kommunikations- und Kooperationsprozesse im Sinne einer aktiven Teilhabe an der Gesellschaft gestalten und reflektieren; ethische Grundsätze sowie kulturell-gesellschaftliche Normen beachten</p>
1.4 Datenschutz und Informationssicherheit	2.4 Informationskritik	3.4 Cybergewalt und -kriminalität
<p>Verantwortungsvoll mit persönlichen und fremden Daten umgehen; Datenschutz, Privatsphäre und Informationssicherheit beachten</p>	<p>Unangemessene und gefährdende Medieninhalte erkennen und hinsichtlich rechtlicher Grundlagen sowie gesellschaftlicher Normen und Werte einschätzen; Jugend- und Verbraucherschutz kennen und Hilfs- und Unterstützungsstrukturen nutzen</p>	<p>Persönliche, gesellschaftliche und wirtschaftliche Risiken und Auswirkungen von Cybergewalt und -kriminalität erkennen sowie Ansprechpartner und Reaktionsmöglichkeiten kennen und nutzen</p>

4. PRODUZIEREN UND PRÄSENTIEREN		5. ANALYSIEREN UND REFLEKTIEREN		6. PROBLEMLÖSEN UND MODELLIEREN	
4.1 Medienproduktion und Präsentation		5.1 Medienanalyse		6.1 Prinzipien der digitalen Welt	
Medienprodukte adressatengerecht planen, gestalten und präsentieren; Möglichkeiten des Veröffentlichens und Teilens kennen und nutzen		Die Vielfalt der Medien, ihre Entwicklung und Bedeutungen kennen, analysieren und reflektieren		Grundlegende Prinzipien und Funktionsweisen der digitalen Welt identifizieren, kennen, verstehen und bewusst nutzen	
4.2 Gestaltungsmittel		5.2 Meinungsbildung		6.2 Algorithmen erkennen	
Gestaltungsmittel von Medienprodukten kennen, reflektiert anwenden sowie hinsichtlich ihrer Qualität, Wirkung und Aussageabsicht beurteilen		Die interessengeleitete Setzung und Verbreitung von Themen in Medien erkennen sowie in Bezug auf die Meinungsbildung beurteilen		Algorithmische Muster und Strukturen in verschiedenen Kontexten erkennen, nachvollziehen und reflektieren	
4.3 Quellendokumentation		5.3 Identitätsbildung		6.3 Modellieren und Programmieren	
Standards der Quellenangaben beim Produzieren und Präsentieren von eigenen und fremden Inhalten kennen und anwenden		Chancen und Herausforderungen von Medien für die Realitätswahrnehmung erkennen und analysieren sowie für die eigene Identitätsbildung nutzen		Probleme formalisiert beschreiben, Problemlösestrategien entwickeln und dazu eine strukturierte, algorithmische Sequenz planen; diese auch durch Programmieren umsetzen und die gefundene Lösungsstrategie beurteilen	
4.4 Rechtliche Grundlagen		5.4 Selbstregulierte Mediennutzung		6.4 Bedeutung von Algorithmen	
Rechtliche Grundlagen des Persönlichkeits- (u.a. des Bildrechts), Urheber- und Nutzungsrechts (u.a. Lizenzen) überprüfen, bewerten und beachten		Medien und ihre Wirkungen beschreiben, kritisch reflektieren und deren Nutzung selbstverantwortlich regulieren; andere bei ihrer Mediennutzung unterstützen		Einflüsse von Algorithmen und Auswirkung der Automatisierung von Prozessen in der digitalen Welt beschreiben und reflektieren	

Übergeordnete Kompetenzerwartungen

Schülerinnen und Schüler

- bewerten ein Ergebnis einer informatischen Modellierung, (MKR 6.4)
- implementieren informatische Modelle unter Verwendung algorithmischer Grundstrukturen, (MKR 6.1, 6.2)
- dokumentieren gemeinsam ihren Arbeitsprozess und ihre Ergebnisse auch mithilfe digitaler Werkzeuge, (MKR 1.2)
- setzen bei der Bearbeitung einer informatischen Problemstellung geeignete digitale Werkzeuge zum kollaborativen Arbeiten ein. (MKR 1.2, 3.1)

Konkretisierte Kompetenzerwartungen

Schülerinnen und Schüler

- erläutern ein einfaches Transpositionsverfahren als Möglichkeit der Verschlüsselung (DI), (MKR 1.4)
- vergleichen verschiedene Verschlüsselungsverfahren unter Berücksichtigung von ausgewählten Sicherheitsaspekten (DI), (MKR 1.4)
- identifizieren in Handlungsvorschriften Anweisungen und die algorithmischen Grundstrukturen Sequenz, Verzweigung und Schleife (MI), (MKR 6.2)
- implementieren Algorithmen in einer visuellen Programmiersprache (MI), (MKR 6.1, 6.3)
- implementieren Algorithmen unter Berücksichtigung des Prinzips der Modularisierung (MI), (MKR 6.1, 6.3)
- überprüfen die Wirkungsweise eines Algorithmus durch zielgerichtetes Testen (MI), (MKR 6.2)
- ermitteln durch die Analyse eines Algorithmus dessen Ergebnis (DI), (MKR 6.2)
- bewerten einen als Quelltext, Programmablaufplan (PAP) oder Struktogramm dargestellten Algorithmus hinsichtlich seiner Funktionalität (A), (MKR 6.3)
- erläutern die Funktionsweise eines Automaten aus ihrer Lebenswelt (A), (MKR 6.1)
- beschreiben das Prinzip der Eingabe, Verarbeitung und Ausgabe (EVA-Prinzip) als grundlegendes Prinzip der Datenverarbeitung (DI), (MKR 6.1)
- erläutern Prinzipien der strukturierten Dateiverwaltung (A), (MKR 1.3)
- setzen Informatiksysteme zur Kommunikation und Kooperation ein (MKR 3.1)
- beschreiben an Beispielen die Bedeutung von Informatiksystemen in der Lebens- und Arbeitswelt (KK), (MKR 6.4)
- benennen an ausgewählten Beispielen Auswirkungen des Einsatzes von Informatiksystemen auf ihre Lebens- und Erfahrungswelt (A/KK), (MKR 6.4)
- erläutern an ausgewählten Beispielen Auswirkungen des Einsatzes von Informatiksystemen (A/KK), (MKR 6.4)
- beschreiben Maßnahmen zum Schutz von Daten mithilfe von Informatiksystemen (A). (MKR 1.4)

10 Bezug zur Verbraucherbildung

Bereiche der Verbraucherbildung:

Übergreifender Bereich Allgemeiner Konsum			
Bereich A Finanzen, Markt- geschehen und Verbraucherrecht	Bereich B Ernährung und Gesundheit	Bereich C Medien und Information in der digitalen Welt	Bereich D Leben, Wohnen und Mobilität

Das Fach Informatik trägt zur Verbraucherbildung erwartungsgemäß deutlich weniger bei, als zum Medienkompetenzrahmen. Lediglich zum Bereich C sind nennenswerte Bezüge herstellbar.

Konkretisierte Kompetenzerwartungen

Schülerinnen und Schüler

- benennen an ausgewählten Beispielen Auswirkungen des Einsatzes von Informatiksystemen auf ihre Lebens- und Erfahrungswelt (A/KK), (VB C Z5)
- anstelle der vorherigen KE: erläutern an ausgewählten Beispielen Auswirkungen des Einsatzes von Informatiksystemen (A/KK), (VB C Z5)
- beschreiben anhand von ausgewählten Beispielen die Verarbeitung und Nutzung personenbezogener Daten (DI), (VB C Z5)
- erläutern anhand von Beispielen aus ihrer Lebenswelt Nutzen und Risiken beim Umgang mit eigenen und fremden Daten auch im Hinblick auf Speicherorte (A), (VB C Z3)
- beschreiben Maßnahmen zum Schutz von Daten mithilfe von Informatiksystemen (A). (VB C Z2)