

Viktoriaschule Aachen



**Curriculum im Fach Informatik
für die S I und S II
mit Leistungsbewertungskon-
zept**

Stand: November 2025

Inhaltsverzeichnis

1. FACHGRUPPE INFORMATIK.....	4
1.1. FACHGRUPPENARBEIT	5
1.2. ENTSCHEIDUNGEN ZU FACH- UND UNTERRICHTSÜBERGREIFENDEN FRAGEN	5
1.3. ZUSAMMENARBEIT MIT ANDEREN FÄCHERN	6
1.4. EVALUATION DES SCHULINTERNEN CURRICULUMS.....	6
2. SCHULINTERNER LEHRPLAN SEKUNDARSTUFE I	6
2.1. VORBEMERKUNGEN	6
2.1.1. KLASSE 5, 1. HALBJAHR (DigiKit)	7
2.1.2. KLASSE 5, 2. HALBJAHR.....	7
2.1.3. MODELLVORHABEN KLASSE 7 UND 8.....	8
2.1.4. WAHLPFLICHTFACH (WP II)	9
2.2. ÜBERSICHT UNTERRICHTSVERTEILUNG SEKUNDARSTUFE 1.....	10
2.2.1. DigiKit	10
2.2.2. KLASSE 5 (2. HALBJAHR)	11
2.2.3. KLASSE 6 (1. UND 2. HALBJAHR)	13
2.2.4. UMSETZUNG DES MEDIENKOMPETENZRAHMENS IN DER 5. UND 6. KLASSE	15
2.2.5. WAHLPFLICHTKURS II (9. UND 10. KLASSE).....	16
3. SCHULINTERNER LEHRPLAN SEKUNDARSTUFE II	23
3.1. VORBEMERKUNG	23
3.2. EINFÜHRUNGSPHASE (EF)	23
3.2.1. ÜBERSICHT UNTERRICHTSVORHABEN.....	23
3.2.2. KONKRETISIERTE UNTERRICHTSVORHABEN EF	26
3.3. QUALIFIKATIONSPHASE	35
3.3.1. ÜBERSICHT UNTERRICHTSVORHABEN Q1	35
3.3.2. ÜBERSICHT UNTERRICHTSVORHABEN Q2	37
3.3.3. KONKRETISIERTE UNTERRICHTSVORHABEN Q1 UND Q2	38
4. LEISTUNGSBEWERTUNG	50
4.1. RECHTLICHE GRUNDLAGEN UND PRINZIPIEN.....	50
4.1.1. TRANSPARENZ DER LEISTUNGSBEURTEILUNG	50
4.1.2. GRUNDSÄTZE DER LEISTUNGSBEWERTUNG	51
4.2. BEURTEILUNGSBEREICH SCHRIFTLICHE LEISTUNGEN/KLASSENARBEITEN.....	51
4.2.1. BEWERTUNG VON KURSARBEITEN IN DER SEKUNDARSTUFE I	51
4.2.2. BEWERTUNG VON KLAUSUREN IN DER SEKUNDARSTUFE II	53
4.2.3. KRITERIEN	54
4.2.4. GRUNDSÄTZE DER LEISTUNGSRÜCKMELDUNG UND BERATUNG	55
4.3. BEWERTUNG DER SONSTIGEN MITARBEIT IN DER SEKUNDARSTUFE I	55
4.4. INDIVIDUELLE FÖRDERUNG	57
4.5. BILDUNG DER ZEUGNISNOTE.....	57
4.6. BEWERTUNG DER SONSTIGEN MITARBEIT IN DER SEKUNDARSTUFE II	57
4.6.1. VERBINDLICHE ABSPRACHEN DER FACHKONFERENZ	57

4.6.2. LEISTUNGSASPEKTE	58
4.6.3. KRITERIEN	58
4.7. BEWERTUNG VON FACHARBEITEN UND BESONDEREN LERNLEISTUNGEN	59
<u>5. QUALITÄTSSICHERUNG UND EVALUATION</u>	<u>59</u>
<u>6. ANHANG</u>	<u>60</u>
6.1. BEURTEILUNGSBOGEN ZUR FACHARBEIT	60

1. Fachgruppe Informatik

Die Viktoriaschule ist ein Gymnasium in Trägerschaft der Evangelischen Kirche im Rheinland. Sie liegt in der Aachener Innenstadt nahe dem Hauptbahnhof. Der Einzugsbereich der Schülerschaft umfasst das gesamte Stadtgebiet und die angrenzenden Gemeinden der Städteregion Aachen.

In der Sekundarstufe I gehört seit dem Schuljahr 2021/2022 Informatik zu den Pflichtfächern in der 5. und 6. Klasse. Des Weiteren wird ein Informatik-Kurs des Wahlpflichtkurses II (Stufen 9 und 10) zum durchgängigen Angebot der Schule angeboten. Im Wahlverhalten der SchülerInnen der Stufe 9 ist die Nachfrage für diesen Kurs gut, so dass die maximal 20 Plätze meistens ausgeschöpft werden. Unbefriedigend ist bisher der Anteil der Mädchen in den Kursen, der meist unter 20% liegt.

Aufbauend auf der dreizügigen Sekundarstufe I kommt in der Einführungsphase der Sekundarstufe II in der Regel ein Grundkurs Informatik zustande, der bis zum Abitur Bestand hat. Zu diesen Kursen gehören sowohl SchülerInnen, die Informatik im Wahlpflichtkurs II belegt hatten, als auch solche, die noch keine Erfahrung mit Inhalten der Informatik oder dem Programmieren gemacht haben. Die Bildung eines Leistungskurses ist wegen der Größe der Jahrgänge und wegen des Wahlverhaltens der SchülerInnen sehr unwahrscheinlich und daher bisher nicht in Betracht gekommen. Dennoch können die SchülerInnen das Fach Informatik sowohl als 3. Abiturfach wählen als auch als 4. mündliches Fach.

Das Angebot an Arbeitsgemeinschaften im Bereich der Informatik befindet sich noch im Aufbau. Seit dem Schuljahr 2023/2024 wird ein Modellvorhaben für die 7. Klasse angeboten. Über das Modellvorhaben „Informatik in den Jahrgangsstufen 7/8 des Gymnasiums“ soll erprobt werden, inwieweit diese „Lücke“ durch das Angebot einer einstündigen Arbeitsgemeinschaft in den Jahrgangsstufen 7 und 8 geschlossen werden kann. Das Modellvorhaben umfasst eine Laufzeit von zwei Jahren. Es beginnt im Schuljahr 2023/24 in der Jahrgangsstufe 7 und wird im Schuljahr 2024/25 mit den Jahrgangsstufen 7 und 8 fortgeführt. Dieses AG-Angebot ist von Bedeutung, um die Kenntnisse der Informationstechnologie unter den Schülern breiter zu streuen und insbesondere die Schülerinnen besser auf ihre Fächerwahl für WP II und die Oberstufe vorzubereiten.

Die Ausstattung der Viktoriaschule für den Informatik-Unterricht ist ab dem Schuljahr 2022/2023 mit nur noch einem festen Computerraum und einem mobilen Computerraum, die über 18 bzw. 20 Computerarbeitsplätze verfügen, gut. Ergänzt werden diese Plätze durch zusätzliche Laptops, sodass eine Klassenstärke von ca. 30 SchülerInnen jeweils an einem Computer arbeiten kann. Bei den im Unterricht eingesetzten Programmen legen wir Wert auf frei verfügbare und möglichst quelloffene Werkzeuge, die den SchülerInnen auch außerhalb des Unterrichts auf verschiedenen Programmen zur Verfügung stehen. Insbesondere gehören dazu LibreOffice, Filius, Python (TigerJython), Greenfoot, Visual Studio Code, BlueJ, und Eclipse.

1.1. Fachgruppenarbeit

Die Fachkonferenz überprüft kontinuierlich, inwieweit die im schulinternen Lehrplan vereinbarten Maßnahmen zum Erreichen der im Kernlehrplan vorgegebenen Ziele geeignet sind. Dazu dienen beispielsweise auch der regelmäßige Austausch sowie die gemeinsame Konzeption von Unterrichtsmaterialien, welche hierdurch mehrfach erprobt und bezüglich ihrer Wirksamkeit beurteilt werden.

Kollegen der Fachschaft nehmen regelmäßig an Fortbildungen der Bezirksregierung, der Universitäten und des Fachverbandes teil, um fachliches Wissen zu aktualisieren und pädagogische sowie didaktische Handlungsalternativen zu entwickeln. Zudem werden die Erkenntnisse und Materialien aus fachdidaktischen Fortbildungen und Implementationen zeitnah in der Fachgruppe vorgestellt und für alle verfügbar gemacht. Das Feedback von SchülerInnen wird als wichtige Informationsquelle zur Qualitätsentwicklung des Unterrichts angesehen, sodass diese Gelegenheit bekommen, die Qualität zu evaluieren.

Ziele der Fachgruppe für die weitere Entwicklung des Lehrplans sind die stärkere Nutzung von Fortbildungen, Zusammenarbeit mit InfoSphere der RWTH, Ausbau der AGs, ein Konzept für Schnupperstunden in Stufe 7 und die Berücksichtigung geschlechtsspezifischer Aspekte. So ist u.a. eine Intensivierung kollegiumsinterner Fortbildungen geplant.

1.2. Entscheidungen zu fach- und unterrichtsübergreifenden Fragen

Die Fachkonferenz Informatik hat sich im Rahmen des Schulprogramms für folgende fachdidaktische/fachmethodische Grundsätze bzw. zentrale Schwerpunkte entschieden:

- Der Unterricht orientiert sich am aktuellen Stand der Informatik. Dazu wird sich auch mit aktuellen Informatiksystemen und deren Weiterentwicklung beschäftigt.
- Der Unterricht soll von realen Problemen ausgehen, sich auf solche rückbeziehen und knüpft an die Interessen/Erfahrungen der SchülerInnen an (problemorientiert).
- Der Unterricht ist anschaulich sowie gegenwarts- und zukunftsorientiert und gewinnt dadurch für die SchülerInnen an Bedeutsamkeit.
- Der Unterricht ist handlungsorientiert, d.h. projekt- und produktorientiert angelegt.
- Der Unterricht folgt dem Prinzip der Exemplarizität und soll ermöglichen, informatische Strukturen und Gesetzmäßigkeiten in den ausgewählten Problemen und Projekten zu erkennen.
- Der Unterricht fördert vernetzendes Denken und wird deshalb, falls möglich, fach- und lernbereichsübergreifend ggf. auch projektartig angelegt.
- Der Unterricht beinhaltet reale Begegnung sowohl an inner- als auch an außerschulischen Lernorten.
- Im Unterricht werden sowohl für die Schule didaktisch reduzierte als auch reale Informatiksysteme aus der Berufs- und Lebenswelt eingesetzt.
- Der Unterricht leistet einen wichtigen Beitrag zur Vorbereitung auf Ausbildung und Beruf und zeigt informatikaffine Berufsfelder auf.

1.3. Zusammenarbeit mit anderen Fächern

Durch die unterschiedliche Belegung von Fächern können SchülerInnen Aspekte aus anderen Kursen mit in den Informatikunterricht einfließen lassen. Es wird Wert daraufgelegt, dass in bestimmten Fragestellungen die Expertise einzelner SchülerInnen gesucht wird, die aus einem von ihnen belegten Fach genauere Kenntnisse mitbringen und den Unterricht dadurch bereichern. Dieses Wissen soll auch aus dem Informatikunterricht in die jeweiligen anderen Fächer übertragen werden.

1.4. Evaluation des schulinternen Curriculums

Das schulinterne Curriculum stellt keine starre Größe dar, sondern ist als „dynamisches Dokument“ zu betrachten. Dementsprechend werden die Inhalte stetig überprüft, um ggf. Modifikationen vornehmen zu können. Die Fachkonferenz trägt durch diesen Prozess zur Qualitätsentwicklung und damit zur Qualitätssicherung des Faches Informatik bei. Die Evaluation erfolgt jährlich. Zu Schuljahresbeginn werden die Erfahrungen des vergangenen Schuljahres in der Fachschaft gesammelt, bewertet und eventuell notwendige Konsequenzen und Handlungsschwerpunkte formuliert.

2. Schulinterner Lehrplan Sekundarstufe I

2.1. Vorbemerkungen

In der Information für Eltern zur Sekundarstufe I in NRW von 2004 heißt es: „In den Klassen 7 bis 9 mit dem Schwerpunkt in der Klasse 8 wird eine informations- und kommunikations-technologische Grundbildung vermittelt.“ (S. 20) Dementsprechend sollte das Fach Informatik nicht die Aufgabe übernehmen, in allgemeine Anwendungsprogramme (Office, Dateiverwaltung, Internet) einzuführen, jedoch sollen erste Grundkenntnisse vermittelt werden. Dieses Vorhaben bezieht sich natürlich auf die 9. Und 10. Klasse und soll selbstverständlich auch auf das Modellvorhaben Informatik bezogen werden.

Da Informatik ein aufstrebendes Fach im Schulalltag geworden ist, wurden sukzessive passende Lehrwerke veröffentlicht. Im Lehrplan-Navigator des Ministeriums für Sekundarstufe I sind im Fach Informatik keine bis kaum welche gelistet. Infolgedessen wird sich an den jeweiligen unterrichtsspezifischen Inhalten an den Lehrwerken *DAT Klett Einfach Informatik* orientiert. Dementsprechend fehlen in NRW zugelassene Lehrwerke für das Fach Informatik. Ein Lehrwerk kann deshalb nicht auf dieselbe Weise wie in anderen Fächern die Grundlage für das schulinterne Curriculum bilden.

Nicht nur die Inhalte, sondern auch die Methoden der Informatik unterliegen einem stetigen Wandel. Die Lebenswelt der SchülerInnen verändert sich hinsichtlich ihres Umgangs mit IT-Medien schon innerhalb weniger Jahre stark, obwohl diese zuweilen vermehrt mit digitalen Medien aufwachsen und im alltäglichen Gebrauch nutzen. Man betrachte hier zum Beispiel die Entwicklung des „Web 2.0“ bezüglich sozialer Netzwerke und aktiver

Mitgestaltung der Online-Welt. Je konkreter ein Curriculum also Inhalte und Methoden festlegt, desto schneller veraltet es auch.

Das 60-Minuten-Modell der Viktoriaschule hat zur Folge, dass die WP-II-Fächer mit zweimal 60 Minuten pro Woche unterrichtet werden, also 15 Minuten weniger als im herkömmlichen 45-Minuten-Modell. Bezogen auf die beiden Schuljahre bedeutet das eine Reduzierung um ca. 20 Zeitstunden oder etwa acht Unterrichtswochen. Um diese verlorene Zeit dennoch aufzufangen, wird pro Schuljahr in einem Halbjahr das Fach Informatik im Wahlpflichtkurs mit je drei Unterrichtsstunden pro Woche unterrichtet.

2.1.1. Klasse 5, 1. Halbjahr (DigiKit)

Ab dem Schuljahr 2025/2026 wird im ersten Halbjahr in der Klasse 5 das Fach DigiKit (Einführung digitaler Kompetenzen) unterrichtet, indem bereits Themen aus dem Kernlehrplan der Klasse 5 vorab gelehrt werden. Dabei handelt es sich um die Kapitel 0 und 1 aus dem Lehrbuch Informatik 5/6 mit Ausnahme der Themen 1.1 „Das Eva-Prinzip“ und 1.3 „Informatiksysteme, Hardware und Software“. Dadurch sollen den SchülerInnen erste Kompetenzen im Umgang mit dem Computer vermittelt werden und der Einstieg in das zweite Halbjahr erleichtert werden. Dieses Fach wird von den jeweiligen MathematiklehrerInnen bzw. KlassenlehrerInnen betreut und erfolgt in Abstimmung mit den Informatik-lehrerInnen.

2.1.2. Klasse 5, 2. Halbjahr

Die im Folgenden dargestellten Unterrichtsvorhaben decken die im Kernlehrplan angeführten Kompetenzen so ab, dass Lerngelegenheiten zur Erfüllung aller Kompetenzerwartungen gegeben sind.

Die Übersichtstabelle stellt die Verteilung der für alle LehrerInnen gemäß Fachkonferenzbeschluss verbindlichen Unterrichtsvorhaben dar. Der ausgewiesene Zeitbedarf (bezogen auf Einheiten von 60 Minuten) versteht sich als grobe Orientierungsgröße, die nach Bedarf über- oder unterschritten werden kann. Da in den folgenden Unterrichtsvorhaben Inhalte in der Regel anhand von Problemstellungen in Anwendungskontexten bearbeitet werden, sprechen einige Unterrichtsvorhaben jeweils mehrere Inhaltsfelder an. Für die 5. und 6. Klasse legen wir gemäß Beschluss der Fachkonferenz in weiten Strecken das Lehrwerk Informatik 5/6 aus dem C.C. Buchner Verlag zugrunde.

Hinsichtlich des schulinternen Curriculums für die Jahrgangsstufe 5/6 wurde beschlossen, dass Thema 7 „Informatik, Mensch und Gesellschaft“ vorzuziehen und direkt an die Thematik I „Informatiksysteme“ anzuschließen. Als Erweiterung zu Thema 3 „Information und Daten - Verschlüsselungsverfahren“ wird ein vertiefender Exkurs in das schulinterne Curriculum aufgenommen. Dabei werden weitere Verschlüsselungsverfahren erarbeitet und in Bezug zu den schon gelehrt gestellt. Weitere Verschlüsselungsverfahren werden in dem Modellvorhaben 7/8 thematisiert.

2.1.3. Modellvorhaben Klasse 7 und 8

Mit der Einführung des Pflichtfachs Informatik in den Jahrgangsstufen 5 und 6 aller Schulformen wurde der deutlich zunehmenden Bedeutung des Faches Rechnung getragen.

Darüber hinaus haben die SchülerInnen der Schulformen Hauptschule, Realschule, Sekundarschule und Gesamtschule die Möglichkeit, das Fach Informatik im Rahmen des Wahlpflichtangebots ab Klasse 7 auch durchgehend in der Mittelstufe zu belegen.

Für die SchülerInnen an den Gymnasien gibt es diese Option bislang nicht. An den Gymnasien wird das Fach Informatik erst wieder in den Jahrgangsstufen 9 und 10 im Wahlpflichtbereich II angeboten. Infolgedessen bieten wir an der Viktoriaschule eine einstündige Arbeitsgemeinschaft in den Jahrgangsstufen 7 und 8 an, sodass die „Lücke“ durch das Angebot geschlossen werden kann.

Die Arbeitsgemeinschaft umfasst eine Laufzeit von zwei Jahren. Der erste Durchgang startete im Schuljahr 2023/24 in der Jahrgangsstufe 7 und wird im kommenden Schuljahr 2024/25 in der Jahrgangsstufe 8 fortgesetzt. Das Modellvorhaben soll keine einmalige Veranstaltung sein. Die aktuelle Sequenzierung wird Grundlage für den Lehrplan des möglicherweise kommenden Pflichtfachs Informatik in den Stufen 7 und 8. Dieses Vorhaben wird in einer AG umgesetzt.

Die zu lehrende Inhalte werden eine Vertiefung der bekannten Themen aus den Jahrgangsstufen 5 und 6 darstellen, sodass auf Grundlage dessen auch eine weitere Vorbereitung auf das Wahlpflichtfach II erfolgen kann. Diese Themen umfassen Stand jetzt folgende Aspekte:

- Kryptologie (z.B. monoalphabetische und polyalphabetische Substitution)
- Datenbewusstsein (z.B. Weitergabe von Metadaten, Gefahren im Internet)
- Visuelle Programmierung (z.B. durch Scratch)
- Textuelle Programmierung (z.B. durch TigerJython inklusive eines kleinen Roboters + Grundstrukturen der Programmierung mit Python)
- Textuelle Programmierung mithilfe der Programmiersprache Python
- Textuelle Programmierung durch blockbasierte Bausteine
- Teilnahme am Jugendwettbewerb Informatik
- Erstellung eines Escape Rooms
- Prinzip eines Automaten untersuchen
- Logische Schaltungen
- Bildbearbeitung per Gimp; Unterschied Pixelgrafik/Vektorgrafik
- Untersuchung eines KI-gesteuerten Spiels und deren Auswirkung

2.1.4. Wahlpflichtfach (WP II)

Der Lehrplan für den Wahlpflichtbereich II für die Stufen 9 und 10 wurde Ende des Jahres 2025 aktualisiert und auf den Stand des Landesweiten Lehrplans angepasst. Er berücksichtigt somit den Aufbau des Internets bzw. World-Wide-Webs in angemessener Form sowie die Verbreitung mobiler IT und deren tägliche Handhabung durch die SchülerInnen. Dabei werden durch das Handlungsfeld *Informatik, Mensch und Gesellschaft* die SchülerInnen auf eine datenkonforme Verbreitung von Inhalten und Geheimhaltung privater Daten sensibilisiert. Weiterhin sollen sie auf Gefahren im Internet vorbereitet werden. Um erste Kenntnisse in einer textbasierten Programmierung zu erlangen, erlernen die SchülerInnen die Programmiersprache Python. Diese längere Programmierphase endet mit einer abschließenden Programmierung eines Roboters, um die erworbenen Kenntnisse sinngemäß anwenden zu können. Obwohl in der Sekundarstufe II die Programmierung im Vordergrund steht, sind Vorkenntnisse durch den Wahlpflichtbereich wünschenswert, jedoch nicht relevant, sodass jeder in der Sekundarstufe II ohne dem Wahlpflichtkurs auch einsteigen kann. Somit sind die im Wahlpflichtbereich II vermittelten Inhalte bzw. erworbenen Kompetenzen nicht Voraussetzung für die Wahl des Fachs Informatik in der Oberstufe. Die dort v.a. verwendete Programmiersprache Java wird im Wahlpflichtbereich II nicht verwendet.

Der Zeitpunkt der Kursarbeiten wird durch die Mittelstufenkoordination zentral für alle Wahlpflicht-II-Kurse festgelegt und lässt sich daher nicht an didaktische Erfordernisse anpassen. Die Abstände zwischen den Arbeiten und gegenüber dem jeweiligen Beginn der Halbjahre schwanken stark und ändern sich auch von Schuljahr zu Schuljahr. Die Anzahl der Unterrichtsstunden und die Inhalte für die Themenreihen müssen daher immer wieder angepasst werden, damit zum jeweiligen Termin der Arbeiten ein sinnvoller Einschnitt erfolgen kann. Pro Schuljahr kann dabei eine Kursarbeit durch eine Projektarbeit ersetzt werden. Dies wird in der Regel mit einem Programmierprojekt in Bezug auf einen Mini-Roboter (Calliope oder Micro:Bit) in Verbindung gesetzt.

Zurzeit wird in dem Wahlpflichtfach das Informatik-Buch „Informatik 9/10“ vom C.C. Buchner Verlag erprobt (Stand Schuljahr 25/26). Die jeweiligen Themen decken sich mit der Übersicht zu der Stoffverteilung und werden thematisch ergänzt (siehe Kapitel 2.6.). Weitere Informationen und mögliche Vertiefungsangebote werden zeitnah ergänzt, sodass dieser Teil des schulinternen Kernlehrplans noch aktualisiert wird.

2.2. Übersicht Unterrichtsverteilung Sekundarstufe 1

2.2.1.DigiKit

Kapitel 0 Grundlagen (ca. 1 Stunde)

Informatik 5/6	
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 gewohnt, mit einem Computer zu arbeiten. Der Tipp-Kurs wird als AG in der Klasse 5 am Nachmittag angeboten.

Kapitel 1 Informatiksysteme (ca. 8 Stunden)

Informatik 5/6	Kompetenzerwartungen und inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen
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	<u>Inhaltliche Schwerpunkte:</u> - Aufbau und Funktionsweise von Informatiksystemen - Anwendung von Informatiksystemen Die SchülerInnen... ➤ 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), (MKR 6.1), ➤ 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), (MKR 1.3), ➤ setzen Informatiksysteme zur Kommunikation und Kooperation ein (KK), (MKR 3.1).	- Argumentieren (A) - Modellieren und Implementieren (MI) - Darstellen und Interpretieren (DI) - Kommunizieren und Kooperieren (KK)

2.2.2.Klasse 5 (2. Halbjahr)

Kapitel 7 Informatik, Mensch und Gesellschaft (ca. 6 Stunden)

Informatik 5/6	Kompetenzerwartungen und inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen
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 Nutzer 7.5 Verhalten und Umgang mit sozialen Netzwerken	<u>Inhaltliche Schwerpunkte:</u> - Informatiksysteme in der Lebens- und Arbeitswelt - Datenbewusstsein, Datensicherheit und Sicherheitsregeln Die SchülerInnen... ➤ beschreiben an Beispielen die Bedeutung von Informatiksystemen in der Lebens- und Arbeitswelt (KK), (MKR 1.2), ➤ benennen an ausgewählten Beispielen Auswirkungen des Einsatzes von Informatiksystemen auf ihre Lebens- und Erfahrungswelt (A/KK), (MKR 6.4), ➤ anstelle der vorherigen KE: erläutern an ausgewählten Beispielen Auswirkungen des Einsatzes von Informatiksystemen (A/KK), ➤ beschreiben anhand von ausgewählten Beispielen die Verarbeitung und Nutzung personenbezogener Daten (DI), (MKR 1.2) ➤ 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), (MKR 1.2), (MKR 1.2, 3.1), (MKR 1.4).	- Argumentieren (A) - Kommunizieren und Kooperieren (KK)

Kapitel 2 Information und Daten – Informationsgehalt von Daten und ihre Codierung (ca. 8 Stunden)

Informatik 5/6	Kompetenzerwartungen und inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen
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	<u>Inhaltliche Schwerpunkte:</u> - Daten und ihre Codierung - Informationsgehalt von Daten Die SchülerInnen... ➤ erläutern den Datenbegriff anhand von Beispielen aus ihrer Erfahrungswelt (A), ➤ erläutern den Zusammenhang und die Bedeutung von Information und Daten (A), (MKR 1.2), ➤ 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),	- Argumentieren (A) - Modellieren und Implementieren (MI) - Darstellen und Interpretieren (DI) - Kommunizieren und Kooperieren (KK)

Kapitel 3 Information und Daten – Verschlüsselungsverfahren (ca. 7 Stunden)

Informatik 5/6	Kompetenzerwartungen und inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen
3.1 Geheimnisse bewahren mit Verschlüsselung 3.2 Verschlüsselungsverfahren – Monoalphabetische Verschlüsselung 3.3 Verschlüsselungsverfahren – Transposition 3.4 Verschlüsselungsverfahren – Steganographie	<u>Inhaltliche Schwerpunkte:</u> - Verschlüsselungsverfahren Die SchülerInnen... ➤ erläutern ein einfaches Transpositionsverfahren als Möglichkeit der Verschlüsselung (DI), (MKR 6.4), (MKR 1.4), ➤ vergleichen verschiedene Verschlüsselungsverfahren unter Berücksichtigung von ausgewählten Sicherheitsaspekten (DI), (MKR 1.4).	- Darstellen und Interpretieren (DI)

2.2.3.Klasse 6 (1. und 2. Halbjahr)

Kapitel 4 Algorithmen (ca. 6 Stunden)

Informatik 5/6	Kompetenzerwartungen und inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen
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	<u>Inhaltliche Schwerpunkte:</u> - Algorithmen und algorithmische Grundkonzepte Die SchülerInnen... ➤ 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), (MKR 6.2),	- Modellieren und Implementieren (MI) - Darstellen und Interpretieren (DI)

Kapitel 5 Programmieren mit einer visuellen Programmierumgebung (ca. 8 Stunden)

Informatik 5/6	Kompetenzerwartungen und inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen
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	<u>Inhaltliche Schwerpunkte:</u> - Implementation von Algorithmen Die SchülerInnen... ➤ implementieren Algorithmen in einer visuellen Programmiersprache (MI), (MKR 6.1, 6.3), ➤ implementieren Algorithmen unter Berücksichtigung des Prinzips der Modularisierung (MI), (MKR 6.1, 6.2), (MKR 1.2, 3.1), (MKR 6.1, 6.3), ➤ überprüfen die Wirkungsweise eines Algorithmus durch zielgerichtetes Testen (MI), ➤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).	- Argumentieren (A) - Modellieren und Implementieren (MI) - Darstellen und Interpretieren (DI)

Kapitel 6 Automatisierung und künstliche Intelligenz (ca. 7 Stunden)

Informatik 5/6	Kompetenzerwartungen und inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen
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	<u>Inhaltliche Schwerpunkte:</u> - Aufbau und Wirkungsweise einfacher Automaten - Maschinelles Lernen mit Entscheidungsbäumen und mit neuronalen Netzen Die SchülerInnen... ➤ erläutern die Funktionsweise eines Automaten aus ihrer Lebenswelt (A), (MKR 6.1), ➤ 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), (MKR 1.2), (MKR 1.2, 3.1).	- Argumentieren (A) - Darstellen und Interpretieren (DI) - Kommunizieren und Kooperieren (KK)

2.2.4. Umsetzung des Medienkompetenzrahmens in der 5. Und 6. Klasse

Medienkompetenzen		Beispiele zur Umsetzung in Informatik 5/6	
1. Bedienen und Anwenden – Digitale Werkzeuge			
1.1 Medienausstattung (Hardware)	➤ 1.2 Projekt: Woraus besteht ein Computer? ➤ 1.3 Informatiksysteme, Hardware und Software	S.16f S.18f/A1 und A2	
1.3 Datenorganisation	➤ 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	S.22/A2, Aufgabe 1 S.25/A3, Aufgabe 1 und 2 S.26f/A1, A2, Aufgabe 1 und 2 S.28f	
1.4 Datenschutz und Informationssicherheit	➤ 7.3 Schutz von Daten mit Hilfe von Informatiksystemen ➤ 7.4 Wem gehören die Daten? – Rechte von Nutzern	S.132f S.134f	
2. Informieren und Recherchieren			
2.1 Informationsauswertung	➤ 2.1 Kommunikation im Alltag und in der Informatik	S.37/Aufgabe 3	
2.4 Informationskritik	➤ 7.5 Verhalten und Umgang mit sozialen Medien	S.136/A2	
3. Kommunizieren und Kooperieren			
3.1 Kommunikations- und Kooperationsprozesse	➤ 1.3 Informatiksysteme, Hardware und Software ➤ 2.5 Textcodierung – Der ASCII-Code ➤ 4.2 Algorithmen im Alltag	S.19/Aufgabe 1 S.45/Aufgabe 2d S.69/Aufgabe 3b	
3.4 Cybergewalt und -kriminalität	➤ 7.2 Daten und Gefahren im Internet	S.130f	
4. Produzieren und Präsentieren			
4.1 Medienproduktion und Präsentation	➤ 2.1 Kommunikation im Alltag und in der Informatik	S.37/Aufgabe3	
4.4 Rechtliche Grundlagen	➤ 7.4 Wem gehören die Daten? – Rechte von Nutzern	S.134f	
5. Analysieren und Reflektieren			
5.4 Selbstregulierte Mediennutzung	➤ 7.5 Verhalten und Umgang mit sozialen Medien	S.136/A1	
6. Problemlösen und Modellieren			
6.2 Algorithmen erkennen	➤ 4 Algorithmen	S.65 – 82	
6.3 Modellieren und Programmieren	➤ 5 Programmieren mit einer visuellen Programmierumgebung	S.83 – 106	
6.4 Bedeutung von Algorithmen	➤ 6.4 Künstliche Intelligenz in unserem Alltag	S.114f	

2.2.5. Wahlpflichtkurs II (9. Und 10. Klasse)

Planungsgrundlage: 100 Zeitstunden (2,5 Stunden pro Woche, 40 Wochen) → 75 % entsprechen 75 Unterrichtsstunden pro Schuljahr.

Unterrichtsvorhaben	Inhaltliche Schwerpunkte/ Konkretisierung	Inhaltsfelder	methodische Schwerpunkte/Kompetenzen
UV 9.1.: Mein digitaler Fußabdruck – wo hinterlasse ich Daten, was kann daraus geschlossen werden und wie kann ich mich schützen? Leitfragen: • Aus welchen Quellen werden Informationen über Personen zusammengestellt? • Zu welchem Zweck werden personenbezogene Informationen aus verschiedenen Quellen verknüpft? • Welche Probleme ergeben sich aus der unkontrollierten Nutzung verknüpfter Datenbestände? • Wie wird unser Alltag durch allgegenwärtige Informationstechnologien jetzt und in Zukunft geprägt? • Welche typischen Funktionalitäten und technischen Grundlagen nutzen computergestützte Alltagsgegenstände? • Welche rechtlichen Aspekte spielen im Zusammenhang mit Datenerhebungen und -verknüpfungen eine Rolle? • Welche rechtlichen Aspekte werden bei der Nutzung allgegenwärtiger Informationstechnologien berührt? • Unter welchen rechtlichen Rahmenbedingungen dürfen Videos, Bilder und Grafiken auf Internetseiten veröffentlicht werden? Zeitbedarf: 14 Stunden	• Anwendung von Informatiksystemen • Informatiksysteme im Kontext gesellschaftlicher und rechtlicher Normen • Chancen und Risiken bei der Nutzung von Informatiksystemen • Urheberrecht und Creative Commons • Gesellschaftliche Auswirkungen von Datenspeicherung, Datenschutz und Datenmissbrauch • Soziale Netzwerke • Datensicherheit	• Informatiksysteme • Informatik, Mensch und Gesellschaft	Die SchülerInnen... • erläutern den Zusammenhang und die Bedeutung von Information und Daten (A). • repräsentieren Information in natürlicher Sprache, formalsprachlich und grafisch (DI), • codieren Daten für die Verarbeitung mit einem Informatiksystem (DI), • verarbeiten Daten mithilfe von Informatiksystemen (MI), • benennen Maßnahmen zur sicheren Kommunikation in Netzwerken und wenden diese an (DI), • erläutern die Unsicherheit eines einfachen Verschlüsselungsverfahrens (A) • erläutern im Rahmen einer animierten Präsentation Aspekte der digitalen Welt dar (A/DI) • verwenden arithmetische und logische Operationen (MI) • beschreiben das Prinzip der Eingabe, Verarbeitung und Ausgabe (EVA-Prinzip) als grundlegendes Prinzip der Datenverarbeitung (DI) • benennen Grundkomponenten von Informatiksystemen und beschreiben ihre Funktionen (DI)

Unterrichtsvorhaben	Inhaltliche Schwerpunkte/ Konkretisierung	Inhaltsfelder	methodische Schwerpunkte/Kompetenzen
UV 9.2.: Das weltweite Datennetz – ein Geheimnis? Leitfragen: • Welchen geschichtlichen Ablauf nahm das Internet und welche Struktur wurde angenommen? • Wie funktioniert der Aufruf von Internetseiten? • Welchen Aufbau besitzt eine URL? • Welche Auswirkungen haben bearbeitete Bilder im Internet? • Wie kann ich meine Daten schützen? • Welche Schadsoftware kann meinen Computer befallen? Zeitbedarf: 10 Stunden	• Information, Daten und ihre Codierung • Verarbeitung und Verwaltung von Daten • Aufbau und Funktionsweise einfacher Informatikanwendung von Informatiksystemen • Informatiksysteme im Kontext gesellschaftlicher und rechtlicher Normen • Chancen und Risiken bei der Nutzung von Informatiksystemen • Gesellschaftliche Auswirkungen von Datenspeicherung, Datenschutz und Datenmissbrauch • Datenschutz in vernetzten Informationsgesellschaften	• Information und Daten • Formale Sprachen • Informatiksysteme • Informatik, Mensch und Gesellschaft	Die SchülerInnen... • erläutern unterschiedliche Dienste in Netzwerken (KK) • 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) • 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)
UV 9.3.: Erstellung einer eigenen Homepage mit HTML und CSS Leitfragen: • Wie erstellt man Internetseiten in HTML? • Wie können Hyperlinks, Videos, Bilder und Audiodateien in eine Webseite eingebettet werden? • Wie erstelle ich Formulare per HTML und lasse diese an eine E-Mailadresse verschicken? • Wie formatiert man Internetseiten mit Hilfe von CSS? • Wie erstelle ich Spiele als Webseiten mithilfe von JavaScript? Zeitbedarf: 24 Stunden	• Information, Daten und ihre Codierung • Erfassung, Verarbeitung und Verwaltung von Daten • Erstellung und Analyse von Quelltexten • Aufbau und Funktionsweise einfacher Informatikanwendung von Informatiksystemen • Informatiksysteme im Kontext gesellschaftlicher/rechtlicher Normen • Formulare/Fenster und Komponenten (z.B. Buttons, Textfelder) • Einfache Effekte mit JavaScript • Gestaltung eines Quiz mithilfe von HTML und CSS durch flexible und vielgestaltige Layouts	• Information und Daten • Formale Sprachen • Informatiksysteme • Informatik, Mensch und Gesellschaft	Die SchülerInnen... • erstellen syntaktisch korrekte Quelltexte in einer geeigneten Dokumentenbeschreibungssprache (MI) • analysieren Quelltexte auf syntaktische Korrektheit (A) • überprüfen standardisierte Angaben auf formale Korrektheit (A) • gestalten Webseiten mit strukturierten Inhalten zu einem Oberthema und verknüpfen sie untereinander (DI/MI) • benennen ausgewählte rechtliche Rahmenbedingungen des Einsatzes von Informatiksystemen (DI) • wenden zielgerichtet Prinzipien der strukturierten Dateiverwaltung an (MI) • führen anhand der gelernten Inhalte eine Projektarbeit in Form einer Erstellung eines Quizzes durch (MI)

Unterrichtsvorhaben	Inhaltliche Schwerpunkte/ Konkretisierung	Inhaltsfelder	methodische Schwerpunkte/Kompetenzen
UV 9.4.: Strategien entwickeln – Einführung von Baumdiagrammen und der Graphentheorie Leitfragen: • Wie können Zusammenhänge zwischen Daten transparent verdeutlicht werden? • Welcher Weg ist der kürzeste und welche Eigenschaften müssen diese Wege beinhalten? • Wie sieht die binäre Darstellung von Graphen aus? • Welche Optimierungsprobleme können erörtert werden? • Wie lassen sich Daten komprimieren? • Welche Fehlerquellen sind bei der Datenübertragung möglich? Zeitbedarf: 15 Stunden	• Information, Daten und ihre Codierung • Erfassung, Verarbeitung und Verwaltung von Daten • Entwurf und Analyse von Algorithmen • Daten und ihre Codierung • Erfassung, Verarbeitung und Verwaltung von Daten	• Information und Daten • Informatiksysteme	Die SchülerInnen... • erläutern den Zusammenhang und die Bedeutung von Information und Daten (A) • repräsentieren Information in natürlicher Sprache, formalsprachlich und grafisch (DI) • codieren Daten für die Verarbeitung mit einem Informatiksystem (DI) • interpretieren Ergebnisse eines Datenverarbeitungsprozesses (DI) • verarbeiten Daten mithilfe von Informatiksystemen (MI)
UV 9.5.: Jäger und Sammler – Wie werden Datensammlungen systematisch angelegt und verwaltet? Leitfragen: • Wie werden Daten sinnvoll angelegt? • Wie können Informationen effizient aus einem Datensatz gefiltert werden? • Wie können Daten sinnvoll sortiert werden? • Wie können Zusammenhänge zwischen Daten transparent verdeutlicht werden? • Wie lassen sich Datensätze grafisch miteinander verknüpfen? Zeitbedarf: 12 Stunden	• Daten und ihre Codierung • Erfassung, Verarbeitung und Verwaltung von Daten • Algorithmen zum Lösen von Aufgaben und Problemen aus verschiedenen Anwendungsgebieten • Algorithmen mit den algorithmischen Grundkonzepten entwerfen, darstellen und realisieren • Formale Sprache und Automaten • SQL-Abfragen • Durchführung eines Planbeispiels • Gesellschaftliche Auswirkungen von Datenspeicherung, -schutz, -missbrauch und -sicherheit • Rasterfahndung, Sortieren, Filtern	• Information und Daten • Formale Sprachen • Algorithmen • Informatiksysteme • Informatik, Mensch und Gesellschaft	Die Schülerinnen... • analysieren Datenbanken zur Gewinnung und Weitergabe persönlicher Daten (A) • ermitteln den Aufbau einer einfachen Datenbank (Tabelle: Schlüssel, Schema, Datensatz, Datenfeld, ...) (DI) • identifizieren Objekte in einer Datenbank (DI) • sortieren und filtern Datensätze (DI) • erstellen über Masken, Assistenten oder SQL-Abfragen (DI) • stellen Beziehungen zwischen Tabellen/Abfragen grafisch dar (Beziehungstypen) (DI) • geben Syntax und Semantik/ Entitäten (Attribute, Datentyp, Primärschlüssel, Schreibweise) an (A/DI)

Unterrichtsvorhaben	Inhaltliche Schwerpunkte/ Konkretisierung	Inhaltsfelder	methodische Schwerpunkte/Kompetenzen
UV 10.1.: Wie funktioniert unser Schul-Netzwerk? Leitfragen: • Was muss ich beachten, wenn ich Netzwerke erfolgreich nutzen will? • Welchen Aufbau haben Netzwerke? • Welchen geschichtlichen Hintergrund haben Netzwerke? • Welche Ziele und Aufgaben verfolgen Netzwerke? Zeitbedarf: 9 Stunden	• Komponenten eines einfachen Netzwerkes • Protokolle und Pakete im Internet • Struktur verbundener Netzwerke und Routing • Sicherheit in Netzwerken	• Information und Daten • Informatiksysteme	Die SchülerInnen... • experimentieren mit Netzwerkkomponenten in der Simulationsumgebung Filius. (DI) • klassifizieren und beurteilen Anforderung an Netze (A) • kennen Aufgaben in ihrer Vielfalt und machen sie sich am eigenen Schulnetz bewusst (DI) • ordnen Netze ein (z. B. Schulnetz, LAN-Party-Netz) (DI) • können Entwicklung von Computernetzen im historischen Bezug darstellen (DI)
UV 10.2.: Tabellenkalkulation – Die TK als sinnvolles Hilfsmittel bei Berechnungen und Kalkulationen Leitfragen: • Wie kann der Computer bei der Zukunftsplanung helfen? • Wie strukturiert man Programme sinnvoll? • Welche Daten muss der Computer speichern? • Wie analysiert und testet man Programmenteile? Zeitbedarf: 12 Stunden	• Information, Daten und ihre Codierung • Erfassung, Verarbeitung und Verwaltung von Daten • Chancen und Risiken der Nutzung von Informatiksystemen • Funktionen für Verweise, Texte und Suche • Programmieren mit wiederholenden und bedingten Anweisungen • Variablen als Merkzettel • Objekte und ihre Funktionen aufrufen • Erstellung und Formatierung von Tabellen • Durchführung einfacher und komplexeren Kalkulationen • Daten sammeln, eingeben und auswerten • Rechnen mit komplexen Formeln • Spielerische Anwendungsbeispiele aus der Kryptologie	• Information und Daten • Informatik, Mensch und Gesellschaft	Die SchülerInnen... • erläutern den Zusammenhang und die Bedeutung von Information und Daten (A) • repräsentieren Information in natürlicher Sprache, formalsprachlich und grafisch (DI) • codieren Daten für die Verarbeitung mit einem Informatiksystem (DI) • 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)

Unterrichtsvorhaben	Inhaltliche Schwerpunkte/ Konkretisierung	Inhaltsfelder	methodische Schwerpunkte/Kompetenzen
UV 10.3.: Computerprogramme mit System – Einstieg in die textorientierte Programmierung Leitfragen: • Wie realisiert man einen Algorithmus in der Programmiersprache Python? • Wie realisiert man Fallunterscheidungen und Wiederholungen? • Wie verwendet man Parameter und Variablen in der Programmiersprache Python? • Wie strukturiert man Programme sinnvoll? • Welche Daten muss der Computer speichern? • Wie analysiert und testet man Programmtteile? Zeitbedarf: 34 Stunden	• Information, Daten und ihre Codierung • Erfassung, Verarbeitung und Verwaltung von Daten • Entwurf und Analyse von Algorithmen • Erstellung und Analyse von Quelltexten • Variablen als Merkzettel • Datentypen • Kontrollstrukturen: Wiederholte und bedingte Anweisungen • Objekte und ihre Funktionen aufrufen • Kleine Animationen anhand grafischer Anwendungen erstellen	• Information und Daten • Algorithmen • Formale Sprachen	Die SchülerInnen... • repräsentieren Information in natürlicher Sprache, formalsprachlich und grafisch (DI) • codieren Daten für die Verarbeitung mit einem Informatiksystem (DI) • 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) • modellieren und implementieren eine Anwendung unter Verwendung eines strukturierten Datentyps in einer Programmiersprache (MI) • entwerfen Algorithmen unter Verwendung des Variablenkonzeptes und von Kontrollstrukturen und unter Berücksichtigung des Prinzips der Modularisierung (MI) • reflektieren den Entwurfsprozess und beschreiben ihn auch fachsprachlich (A) • implementieren, modifizieren und kommentieren Algorithmen in einer Programmierumgebung (MI) • strukturieren und zerlegen Algorithmen in Teilalgorithmen (MI) • modifizieren Programme (MI) • überprüfen Handlungsvorschriften auf Eindeutigkeit und Terminierung (A) • beurteilen die Problemangemessenheit eines Algorithmus (A) • erstellen syntaktisch korrekte Quelltexte in einer Programmiersprache (MI) • analysieren und testen Algorithmen und Programme (MI) • erläutern die Begriffe Syntax und Semantik an Beispielen (A) • analysieren Quelltexte auf syntaktische Korrektheit (A) • bewerten auf Grundlage im Informatikunterricht erworbenen Kenntnisse Möglichkeiten der Datenverarbeitung hinsichtlich Chancen und Risiken in ausgewählten Kontexten (A) • erläutern die Möglichkeit der Werteübergabe mittels Parametern (MI) • überprüfen die Wirkungsweise eines Algorithmus durch zielgerichtetes Testen bei der Lösung gleichartiger Probleme (MI),

Unterrichtsvorhaben	Inhaltliche Schwerpunkte/ Konkretisierung	Inhaltsfelder	methodische Schwerpunkte/Kompetenzen
UV 10.4.: Computerprogramme mit System – Steuerung und Programmierung von Roboter im Alltag und in der Arbeitswelt Leitfragen: • Was steckt hinter dem Begriff Informatiksystem? • Warum werden immer mehr Alltagsgeräte durch Informatiksysteme gesteuert? • In welchen Bereichen werden Roboter eingesetzt? • Wie arbeiten Roboter? • Aus welchen Bestandteilen sind typische Roboter aufgebaut? • Wie hat sich die Arbeitswelt durch den Einsatz von Robotern verändert und welche Veränderungen sind in der Zukunft zu erwarten? • Welche Daten können mit Sensoren erfasst und wie können diese Daten in Programmen verarbeitet werden? • Welche Effekte werden durch Aktoren ausgelöst und wie können Aktoren durch Programme angesteuert werden? • Wie können Robotermodelle durch Programme gesteuert werden? Zeitbedarf: 20 Stunden	• Information, Daten und ihre Codierung • Erfassung, Verarbeitung und Verwaltung von Daten • Entwurf und Analyse von Algorithmen • Erstellung und Analyse von Quelltexten • Aufbau und Funktionsweise einfacher Informatiksysteme • Anwendung von Informatiksystemen • Informatiksysteme im Kontext gesellschaftlicher und rechtlicher Normen • Chancen und Risiken bei der Nutzung von Informatiksystemen • Möglichkeiten und Grenzen von Robotern und künstlicher Intelligenz • Motoren, Sensoren und Aufbau des Roboterbausatzes • Komponenten des Roboters über Python-Funktionen steuern und abfragen • Parallele Prozesse für gleichzeitige Bewegungen von Motoren und Beobachtungen an Sensoren	• Information und Daten • Algorithmen • Informatiksysteme • Informatik, Mensch und Gesellschaft	Die SchülerInnen... • erläutern den Zusammenhang und die Bedeutung von Information und Daten (A) • repräsentieren Information in natürlicher Sprache, formalsprachlich und grafisch (DI) • codieren Daten für die Verarbeitung mit einem Informatiksystem (DI) • verarbeiten Daten mithilfe von Informatiksystemen (MI) • benennen Maßnahmen zur sicheren Kommunikation in Netzwerken und wenden diese an (DI) • erläutern die Unsicherheit eines einfachen Verschlüsselungsverfahrens (A) • interpretieren Ergebnisse eines Datenverarbeitungsprozesses (DI) • wählen geeignete Datentypen im Kontext eines Anwendungsbeispiels aus (MI) • modellieren und implementieren eine Anwendung unter Verwendung eines strukturierten Datentyps in einer Programmiersprache (MI) • entwerfen Algorithmen unter Verwendung des Variablenkonzeptes und von Kontrollstrukturen (MI) • reflektieren den Entwurfsprozess und beschreiben ihn auch fachsprachlich (A) • implementieren, modifizieren 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) • erstellen syntaktisch korrekte Quelltexte in einer Programmiersprache (MI) • analysieren und testen Algorithmen und Programme (MI) • analysieren Quelltexte auf syntaktische Korrektheit (A)

Unterrichtsvorhaben	Inhaltliche Schwerpunkte/ Konkretisierung	Inhaltsfelder	methodische Schwerpunkte/Kompetenzen
UV 10.5: Verstehen und Anwenden von künstlicher Intelligenz Leitfragen: • Wie arbeitet eine künstliche Intelligenz? • Welches Verfahren steckt hinter maschinellem Lernen? • Wie ist ein neuronales Netz aufgebaut? • Welche Funktionsweise steckt hinter überwachtem, unüberwachtem und bestärkendem Lernen? Zeitbedarf: 10 Stunden	• überwachtes Lernen • unüberwachtes Lernen • bestärkendes Lernen	• Information und Daten • Algorithmen • Informatiksysteme • Informatik, Mensch und Gesellschaft	• beschreiben Anwendungsbeispiele künstlicher Intelligenz zum überwachten, unüberwachten und bestärkenden Lernen (KK) • beschreiben die grundlegenden Funktionsweisen maschinellen Lernens (überwacht, unüberwacht, bestärkend) in verschiedenen Anwendungsbeispielen (KK) • ordnen begründet die Methoden des maschinellen Lernens (überwachtes Lernen, unüberwachtes, bestärkendes Lernen) verschiedenen Anwendungsbeispielen zu (A) • analysieren den Einfluss von Trainingsdaten auf die Ergebnisse eines Verfahrens maschinellen Lernens (A)

3. Schulinterner Lehrplan Sekundarstufe II

3.1. Vorbemerkung

Die im Folgenden dargestellten Unterrichtsvorhaben decken die im Kernlehrplan angeführten Kompetenzen so ab, dass Lerngelegenheiten zur Erfüllung aller Kompetenzerwartungen gegeben sind.

Die Übersichtstabelle stellt die Verteilung der für alle LehrerInnen gemäß Fachkonferenzbeschluss verbindlichen Unterrichtsvorhaben dar. Der ausgewiesene Zeitbedarf (bezogen auf Einheiten von 60 Minuten) 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, umfassen die Vorhaben dieses schulinternen Lehrplans ca. 75 Prozent der Bruttounterrichtszeit.

Da in den folgenden Unterrichtsvorhaben Inhalte in der Regel anhand von Problemstellungen in Anwendungskontexten bearbeitet werden, sprechen einige Unterrichtsvorhaben jeweils mehrere Inhaltsfelder an. Für die Einführungsphase legen wir gemäß Beschluss der Fachkonferenz in weiten Strecken das Lehrwerk Informatik 1 aus dem Schöningh-Verlag zugrunde. In der Qualifikationsphase wird dementsprechend das Lehrwerk Informatik 2 aus dem Schöningh-Verlag genutzt.

3.2. Einführungsphase (EF)

3.2.1. Übersicht Unterrichtsvorhaben

<u>Unterrichtsvorhaben E-I</u>	<u>Unterrichtsvorhaben E-II</u>
Thema: Was macht Informatik? - Einführung in die Inhaltsfelder der Informatik	Thema: Grundlagen der objektorientierten Analyse, Modellierung und Implementierung
Zentrale Kompetenzen: • Kommunizieren und Kooperieren • Darstellen und Interpretieren • Argumentieren	Zentrale Kompetenzen: • Modellieren • Implementieren • Darstellen und Interpretieren • Kommunizieren und Kooperieren
Inhaltsfelder: • Informatiksysteme • Informatik, Mensch und Gesellschaft	Inhaltsfelder: • Daten und ihre Strukturierung • Formale Sprachen und Automaten
Inhaltliche Schwerpunkte: • Einsatz, Nutzung und Aufbau von Informatiksystemen • Wirkung der Automatisierung	Inhaltliche Schwerpunkte: • Objekte und Klassen • Syntax und Semantik einer Programmiersprache
Zeitbedarf: 5 Stunden	Zeitbedarf: 6 Stunden

<u>Unterrichtsvorhaben E-III</u> Thema: Algorithmische Grundstrukturen in Java Zentrale Kompetenzen: • Argumentieren • Modellieren • Implementieren • Kommunizieren und Kooperieren Inhaltsfelder: • Daten und ihre Strukturierung • Algorithmen • Formale Sprachen und Automaten Inhaltliche Schwerpunkte: • Objekte und Klassen • Syntax und Semantik einer Programmiersprache • Analyse, Entwurf und Implementierung einfacher Algorithmen Zeitbedarf: 13 Stunden	<u>Unterrichtsvorhaben E-IV</u> Thema: Das ist die digitale Welt! – Einführung in die Grundlagen, Anwendungsgebiete und Verarbeitung binärer Codierung Zentrale Kompetenzen: • Kommunizieren und Kooperieren • Darstellen und Interpretieren • Argumentieren Inhaltsfelder: • Informatiksysteme • Informatik, Mensch und Gesellschaft Inhaltliche Schwerpunkte: • Binäre Codierung und Verarbeitung • Besondere Eigenschaften der digitalen Speicherung und Verarbeitung von Daten Zeitbedarf: 6 Stunden
<u>Unterrichtsvorhaben E-V</u> Thema: Modellierung und Implementierung von Klassen- und Objektbeziehungen anhand lebensnaher Anforderungsbeispiele Zentrale Kompetenzen: • Kommunizieren und Kooperieren • Darstellen und Interpretieren • Argumentieren • Modellieren • Implementieren Inhaltsfelder: • Daten und ihre Strukturierung • Algorithmen • Formale Sprachen und Automaten Inhaltliche Schwerpunkte: • Objekte und Klassen • Syntax und Semantik einer Programmiersprache • Analyse, Entwurf und Implementierung einfacher Algorithmen Zeitbedarf: 14 Stunden	<u>Unterrichtsvorhaben E-VI</u> Thema: Such- und Sortieralgorithmen Zentrale Kompetenzen: • Argumentieren • Modellieren • Darstellen und Interpretieren • Kommunizieren und Kooperieren Inhaltsfelder: • Algorithmen • Daten und ihre Strukturierung Inhaltliche Schwerpunkte: • Algorithmen zum Suchen und Sortieren • Analyse, Entwurf und Implementierung einfacher Algorithmen • Objekte und Klassen Zeitbedarf: 7 Stunden

Unterrichtsvorhaben E-VII

Thema:

Leben in der digitalen Welt – Immer mehr Möglichkeiten und immer mehr Gefahren!?

Zentrale Kompetenzen:

- Kommunizieren und Kooperieren
- Darstellen und Interpretieren
- Argumentieren

Inhaltsfelder:

- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte:

- Geschichte der automatischen Datenverarbeitung
- Wirkungen der Automatisierung
- Dateisystem

Zeitbedarf: 8 Stunden

Gesamter Zeitbedarf für die Einführungsphase: 59 Unterrichtsstunden (à 60 Minuten)

3.2.2. Konkretisierte Unterrichtsvorhaben EF

Didaktische Lernumgebung

Zur Einführung in die objektorientierte Programmierung verwenden wir die didaktische Lernumgebung Greenfoot. Die Lehrtexte und Aufgaben des Lehrwerks „Informatik 1“ beziehen sich auch speziell auf ein bestimmtes Greenfoot-Szenario.

Nach der Einarbeitungsphase sollte auf eine komplexere Lernumgebung umgestellt werden. Hier bieten sich verschiedene Möglichkeiten (in alphabetischer Reihenfolge) an:

- BlueJ
- Eclipse
- Java Editor
- NetBeans
- Visual Studio Code

Die folgenden Konkretisierungen von Vorhaben stammen aus einer Vorlage des Schöningh-Verlags und sind auf das eingeführte Lehrbuch „Informatik 1“ abgestimmt.

Unterrichtsvorhaben EF-I

Thema: Was macht Informatik? - Einführung in die Inhaltsfelder der Informatik

Leitfragen:

- *Was macht Informatik?*
- *Welche fundamentalen Konzepte müssen InformatikerInnen in ihre Arbeit einbeziehen, damit informatische Systeme effizient und zuverlässig arbeiten können?*
- *Wo lassen sich diese Konzepte (in Ansätzen) in dem schuleigenen Netzwerk- und Computersystem wiederfinden?*

Vorhabenbezogene Konkretisierung:

Im ersten Unterrichtsvorhaben werden die fünf Inhaltsfelder des Faches Informatik beispielhaft an einem Informatiksystem erarbeitet. Das Unterrichtsvorhaben ist so strukturiert, dass die SchülerInnen anhand bekannter Alltagstechnik die Grundideen fundamentaler informatischer Konzepte (Inhaltsfelder) größtenteils selbstständig erarbeiten und nachvollziehen. Ausgehend von dem bekannten Bedienungs- und Funktionalitätswissen eines Navigationsgerätes werden die Strukturierung von Daten, das Prinzip der Algorithmen, die Eigenheit formaler Sprachen, die Kommunikationsfähigkeit von Informatiksystemen und die positiven und negativen Auswirkungen auf Menschen und Gesellschaft thematisiert. Das am Navigationsgerät erworbene Wissen kann auf weitere den SchülerInnen bekannte Informatiksysteme übertragen werden. In einem letzten Schritt kann ausgehend von den Inhaltsfeldern das Schulnetzwerk in Ansätzen so analysiert werden, dass ein kompetenter Umgang mit diesem ermöglicht wird.

Zeitbedarf: 5 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Informatiksysteme und ihr genereller Aufbau (a) Daten und ihre Strukturierung (b) Algorithmen (c) Formale Sprachen und Automaten (d) Informatiksysteme (e) Informatik, Mensch und Gesellschaft	Die SchülerInnen... - bewerten anhand von Fallbeispielen die Auswirkungen des Einsatzes von Informatiksystemen (A) - nutzen die im Unterricht eingesetzten Informatiksysteme selbstständig, sicher, zielführend und verantwortungsbewusst (D)	Kapitel 1 „Was macht Informatik“ Als Anschauungsmaterial bieten sich Navigationsgeräte an
2. Der kompetente Umgang mit dem Schulnetzwerk (a) Erstellen und Anlegen von Ordnerstrukturen (b) Sortieren von Dateien und Ordnern (c) Eingabe von Befehlen über Eingabeaufforderung (d) Einzelrechner und Netzwerk (e) Sicherheit und Datenschutz		Kapitel 1 „Was macht Informatik“ Interview mit dem Netzwerk-administrator, Benutzer- und Datenschutzbestimmungen der Schule

Unterrichtsvorhaben EF-II

Thema: Grundlagen der objektorientierten Analyse, Modellierung und Implementierung

Leitfragen:

- *Wie lassen sich Gegenstandsbereiche informatisch modellieren und in einem Greenfoot-Szenario informatisch realisieren?*

Vorhabenbezogene Konkretisierung:

Ein zentraler Bestandteil des Informatikunterrichts der Einführungsphase ist die Objekt-orientierte Programmierung. Dieses Unterrichtsvorhaben führt in die Grundlagen der Analyse, Modellierung und Implementierung in diesem Kontext ein. Dazu werden zunächst konkrete Gegenstandsbereiche aus der Lebenswelt der SchülerInnen analysiert und im Sinne des objektorientierten Paradigmas strukturiert. Dabei werden die grundlegenden Begriffe der Objektorientierung und Modellierungswerkzeuge wie Objektdiagramme und Klassendiagramme eingeführt.

Im Anschluss wird die objektorientierte Analyse für das Greenfoot-Szenario Planetenerkundung durchgeführt. Die vom Szenario vorgegebenen Klassen werden von SchülerInnen in Teilen analysiert und entsprechende Objekte anhand einfacher Problemstellungen erprobt. Die Greenfoot-Umgebung ermöglicht es, Beziehungen zwischen Klassen zu einem späteren Zeitpunkt (Kapitel 4) zu thematisieren. So kann der Fokus hier auf Grundlagen wie der Unterscheidung zwischen Klasse und Objekt, Attribute, Methoden, Objektivität und Objektzustand gelegt werden.

Da in Kapitel 2 zudem auf die Verwendung von Kontrollstrukturen verzichtet wird und der Quellcode aus einer rein linearen Sequenz besteht, ist auf diese Weise eine Fokussierung auf die Grundlagen der Objektorientierung möglich, ohne dass algorithmische Probleme ablenken. Natürlich kann die Arbeit an diesen Projekten unmittelbar zum nächsten Unterrichtsvorhaben (Kapitel 3) führen. Dort stehen Kontrollstrukturen im Mittelpunkt.

Zeitbedarf: 6 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Identifikation von Objekten und Klassen (a) An einem lebensweltnahen Beispiel werden Objekte und Klassen im Sinne der objektorientierten Modellierung eingeführt. (b) Objekte werden durch Objektdiagramme, Klassen durch Klassendiagramme dargestellt. (c) Die Modellierungen werden einem konkreten Anwendungsfall entsprechend angepasst.	Die SchülerInnen... - ermitteln bei der Analyse einfacher Problemstellungen Objekte, ihre Eigenschaften und ihre Operationen (M), - stellen den Zustand eines Objekts dar (D), - modellieren Klassen mit ihren Attributen und ihren Methoden (M), - implementieren einfache Algorithmen unter Beachtung der Syntax und Semantik einer Programmiersprache (I), - implementieren Klassen in einer Programmiersprache, auch unter Nutzung dokumentierter Klassenbibliotheken (I).	Kapitel 2 Einführung in die Objektorientierung 2.1 Objektorientierte Modellierung
2. Analyse von Objekten und Klassen im Greenfoot-Szenario (a) Schritte der objektorientierten Analyse, Modellierung und Implementation (b) Analyse und Erprobung der Objekte im Greenfoot-Szenario		Kapitel 2 Einführung in die Objektorientierung 2.2 Das Greenfoot-Szenario „Planeten-erkundung“; Von der Realität zu Objekten; Von den Objekten zu Klassen, Klassendokumentation; Objekte inspizieren; Methoden aufrufen; Objektidentität und Objektzustand
3. Implementierung einfacher Aktionen in Greenfoot (a) Quelltext einer Java-Klasse (b) Implementation eigener Methoden, Dokumentation mit JavaDoc (c) Programme übersetzen (Aufgabe des Compilers) und testen		Kapitel 2 Einführung in die Objektorientierung 2.3 Programmierung in Greenfoot; Methoden schreiben; Programme übersetzen und testen

Unterrichtsvorhaben EF-III

Thema: Algorithmische Grundstrukturen in Java

Leitfragen:

- *Wie lassen sich Aktionen von Objekten flexibel realisieren?*

Vorhabenbezogene Konkretisierung:

Das Ziel dieses Unterrichtsvorhabens besteht darin, das Verhalten von Objekten flexibel zu programmieren. Ein erster Schwerpunkt liegt dabei auf der Erarbeitung von Kontrollstrukturen. Die Strukturen Wiederholung und bedingte Anweisung werden an einfachen Beispielen eingeführt und anschließend anhand komplexerer Problemstellungen erprobt. Da die zu entwickelnden Algorithmen zunehmend umfangreicher werden, werden systematische Vorgehensweisen zur Entwicklung von Algorithmen thematisiert.

Ein zweiter Schwerpunkt des Unterrichtsvorhabens liegt auf dem Einsatz von Variablen. Beginnend mit lokalen Variablen, die in Methoden und Zählschleifen zum Einsatz kommen, über Variablen in Form von Parametern und Rückgabewerten von Methoden, bis hin zu Variablen, die die Attribute einer Klasse realisieren, lernen die SchülerInnen die unterschiedlichen Einsatzmöglichkeiten des Variablenkonzepts anzuwenden.

Zeitbedarf: 13 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Algorithmen (a) Wiederholungen (While-Schleife) (b) bedingte Anweisungen (c) Verknüpfung von Bedingungen durch die logischen Funktionen UND, ODER und NICHT (d) Systematisierung des Vorgehens zur Entwicklung von Algorithmen zur Lösung komplexerer Probleme	Die SchülerInnen... - analysieren, erläutern und modifizieren einfache Algorithmen und Programme (A), - entwerfen einfache Algorithmen und stellen sie umgangssprachlich und grafisch dar (M), - ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen zu (M), - implementieren Algorithmen unter Verwendung von Variablen und Wertzuweisungen, Kontrollstrukturen sowie Methodenaufrufen (I), - implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I), - implementieren einfache Algorithmen unter Beachtung der Syntax und Semantik einer Programmiersprache (I), - testen Programme schrittweise anhand von Beispielen und interpretieren Fehlermeldungen und korrigieren den Quellcode (I).	Kapitel 3 Algorithmen 3.1 Wiederholungen 3.2 Bedingte Anweisungen 3.3 Logische Funktionen 3.4 Algorithmen entwickeln
2. Variablen und Methoden (a) Implementierung eigener Methoden mit lokalen Variablen, auch zur Realisierung einer Zählschleife (b) Implementierung eigener Methoden mit Parameterübergabe und/oder Rückgabewert sowie von Konstruktoren (c) Realisierung von Attributen		Kapitel 4 Variablen und Methoden 4.1 lokale Variablen 4.2 Methoden 4.3 Attribute

Unterrichtsvorhaben EF-IV

Thema: Das ist die digitale Welt! - Einführung in die Grundlagen, Anwendungsgebiete und Verarbeitung binärer Codierung

Leitfragen:

- *Wie werden binäre Informationen gespeichert und wie können sie davon ausgehend weiterverarbeitet werden?*
- *Wie unterscheiden sich analoge von digitalen Medien und Geräten?*
- *Wie ist der Grundaufbau einer digitalen Rechenmaschine?*

Vorhabenbezogene Konkretisierung:

Das Unterrichtsvorhaben hat die binäre Speicherung und Verarbeitung sowie deren Besonderheiten zum Inhalt. Im ersten Schritt erarbeiten die SchülerInnen anhand ihnen bekannter technischer Gegenstände die Gemeinsamkeiten, Unterschiede und Besonderheiten der jeweiligen analogen und digitalen Version. Nach dieser ersten grundlegenden Einordnung des digitalen Prinzips wenden die SchülerInnen das Binäre Zahlensystem mit arithmetischen und logischen Operationen an und codieren Zeichen binär. Zum Abschluss soll der grundlegende Aufbau eines Rechnersystems im Sinne der von-Neumann-Architektur erarbeitet werden.

Zeitbedarf: 8 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Analoge und digitale Aufbereitung und Verarbeitung von Daten (a) Erarbeitung der Unterschiede von analog und digital (b) Zusammenfassung und Bewertung der technischen Möglichkeiten von analog und digital	Die SchülerInnen... - bewerten anhand von Fallbeispielen die Auswirkungen des Einsatzes von Informatiksystemen (A) - stellen ganze Zahlen und Zeichen in Binärcodes dar (D), - interpretieren Binärcodes als Zahlen und Zeichen (D) - beschreiben und erläutern den strukturellen Aufbau und die Arbeitsweise singulärer Rechner am Beispiel der „Von-Neumann-Architektur“ (A) - nutzen das Internet zur Recherche, zum	Exkurs „Analog und Digital“
2. Der Umgang mit binärer Codierung von Informationen (a) Das binäre (und hexadezimale) Zahlensystem (b) Binäre Informationsspeicherung (c) Binäre Verschlüsselung (d) Implementation eines Binärumrechners	Datenaustausch und zur Kommunikation (K) - implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I)	Exkurs „Binäre Welt“
3. Aufbau informatischer Systeme (a) Identifikation des EVA-Prinzips als grundlegende Arbeitsweise informatischer Systemen (b) Nachvollziehen der von-Neumann-Architektur als relevantes Modell der Umsetzung des EVA-Prinzips		Exkurs „Arbeitsweise eines Computers“

Unterrichtsvorhaben EF-V

Thema: Modellierung und Implementierung von Klassen- und Objektbeziehungen anhand lebensnaher Anforderungsbeispiele

Leitfragen:

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

Vorhabenbezogene Konkretisierung:

Das Unterrichtsvorhaben hat die Entwicklung von Objekt- und Klassenbeziehungen zum Schwerpunkt. Dazu werden, ausgehend von der Realität, über Objektidentifizierung und Entwurf bis hin zur Implementation kleine Softwareprodukte in Teilen oder ganzheitlich erstellt. Zuerst identifizieren die SchülerInnen Objekte und stellen diese dar. Aus diesen Objekten werden Klassen und ihre Beziehungen in Entwurfsdiagrammen erstellt. Nach diesem ersten Modellierungsschritt werden über Klassendokumentationen und der Darstellung von Objektkommunikationen anhand von Sequenzdiagrammen Implementationsdiagramme entwickelt. Danach werden die Implementationsdiagramme unter Berücksichtigung der Klassendokumentationen in Javaklassen programmiert. In einem letzten Schritt wird das Konzept der Vererbung sowie seiner Vorteile erarbeitet.

Schließlich sind die SchülerInnen in der Lage, eigene kleine Softwareprojekte zu entwickeln. Ausgehend von der Dekonstruktion und Erweiterung eines Spiels wird ein weiteres Projekt von Grund auf modelliert und implementiert. Dabei können arbeitsteilige Vorgehensweisen zum Einsatz kommen. In diesem Zusammenhang wird auch das Erstellen von graphischen Benutzeroberflächen eingeführt.

Zeitbedarf: 18 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Umsetzung von Anforderungen in Entwurfsdiagramme (a) Aus Anforderungsbeschreibungen werden Objekte mit ihren Eigenschaften identifiziert (b) Gleichartige Objekte werden in Klassen (Entwurf) zusammengefasst und um Datentypen und Methoden erweitert	Die SchülerInnen... - analysieren und erläutern eine objektorientierte Modellierung (A), - stellen die Kommunikation zwischen Objekten grafisch dar (M), - ermitteln bei der Analyse einfacher Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M), - modellieren Klassen mit ihren Attributen, ihren Methoden und Assoziationsbeziehungen (M), - ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen oder lineare Datensammlungen zu (M), - ordnen Klassen, Attributen und Methoden ihren Sichtbarkeitsbereich zu (M), - modellieren Klassen unter Verwendung von Vererbung (M), - implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I), - testen Programme schrittweise anhand von Beispielen (I), - interpretieren Fehlermeldungen und korrigieren den Quellcode (I), - analysieren und erläutern einfache Algorithmen und Programme (A) - modifizieren einfache Algorithmen und Programme (I), - entwerfen einfache Algorithmen und stellen sie umgangssprachlich und grafisch dar (M). - stellen Klassen, Assoziations- und Vererbungsbeziehungen in Diagrammen grafisch dar (D), - dokumentieren Klassen durch Beschreibung der Funktionalität der Methoden (D)	Kapitel 6 Klassenentwurf 6.1. Von der Realität zum Programm 6.2. Objekte identifizieren 6.3. Klassen und Beziehungen entwerfen
2. Implementationsdiagramme als erster Schritt der Programmierung (a) Erweiterung des Entwurfsdiagramms um Konstruktoren und get- und set-Methoden (b) Festelegung von Datentypen in Java, sowie von Rückgaben und Parametern (c) Entwicklung von Klassendokumentationen (d) Erstellung von Sequenzdiagrammen als Vorbereitung Vorbereitung für die Programmierung		Kapitel 6 Klassenentwurf 6.4 Klassen und Beziehungen implementieren 6.5 Vererbung
3. Programmierung anhand der Dokumentation und des Implementations- und Sequenzdiagrammes (a) Klassen werden in Java-Quellcode umgesetzt (b) Das Geheimnisprinzip wird umgesetzt (c) Einzelne Klassen und das Gesamtsystem werden anhand der Anforderungen und Dokumentationen auf ihre Korrektheit überprüft.		Kapitel 6 Klassenentwurf 6.4 Klassen und Beziehungen implementieren 6.5 Vererbung
4. Vererbungsbeziehungen (a) Das Grundprinzip der Vererbung wird erarbeitet (b) Die Vorteile der Vererbungsbeziehungen (c) Vererbung wird implementiert		Kapitel 6 Klassenentwurf 6.5 Vererbung
5. Softwareprojekt (a) Analyse und Dekonstruktion eines Spiels (Modelle, Quelltexte) (b) Erweiterung des Spiels um weitere Funktionalitäten (c) Modellierung eines Spiels aufgrund einer Anforderungsbeschreibung, inklusive einer grafischen Benutzeroberfläche (d) (arbeitsteilige) Implementation des Spiels		Kapitel 8 Softwareprojekte 8.1 Softwareentwicklung 8.2 Oberflächen

Unterrichtsvorhaben EF-VI

Thema: Such- und Sortialgorithmen

Leitfragen:

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

Vorhabenbezogene Konkretisierung:

Der Schwerpunkt des Vorhabens liegt dabei auf der Erarbeitung von Such- und Sortialgorithmen und nicht auf deren Implementierung in einer Programmiersprache, auf die in diesem Vorhaben vollständig verzichtet werden soll. Zunächst lernen die SchülerInnen das Feld als eine erste Datensammlung kennen. Optional können nun zunächst die wesentlichen Eigenschaften von Algorithmen wie z.B. Korrektheit, Terminiertheit, Effizienz und Verständlichkeit sowie die Schritte einer Algorithmenentwicklung erarbeitet werden (Klärung der Anforderung, Visualisierung, Zerlegung in Teilprobleme). Daran anschließend lernen die SchülerInnen zunächst Strategien des Suchens (lineare Suche, binäre Suche, Hashing) und dann des Sortierens (Selection Sort, Insertion Sort, Bubble Sort) kennen. Die Projekteinstiege dienen dazu, die jeweiligen Strategien handlungsorientiert zu erkunden und intuitive Effizienzbetrachtungen der Suchalgorithmen vorzunehmen.

Zeitbedarf: 9 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Modellierung und Implementation von Datenansammlungen (a) Modellierung von Attributen als Felder (b) Deklaration, Instanziierung und Zugriffe auf ein Feld	Die SchülerInnen... - analysieren Such- und Sortialgorithmen und wenden sie auf Beispiele an (D) - entwerfen einen weiteren Algorithmus zum Sortieren (M) - beurteilen die Effizienz von Algorithmen am Beispiel von Sortierverfahren hinsichtlich Zeit und Speicherplatzbedarf (A) - ordnen Attributen lineare Datenansammlungen zu (M)	Kapitel 7 Sortieren und Suchen auf Feldern 7.1 Das Feld – Eine Sammlung von Daten
2. Explorative Erarbeitung von Suchverfahren (a) Erkundung von Strategien für das Suchen auf unsortierten Daten, auf sortierten Daten und mithilfe einer Berechnungsfunktion. (b) Vergleich der drei Verfahren durch intuitive Effizienzbetrachtungen.		Kapitel 7 Sortieren und Suchen auf Feldern Projekteinstieg 1: Suchen 7.2 Suchen mit System (Lineare Suche, Binäre Suche, Hashing)
3. Systematisierung von Algorithmen und Effizienzbetrachtungen (a) Formulierung oder Erläuterung von mehreren Algorithmen im Pseudocode (b) Anwendung von Sortialgorithmen auf verschiedene Beispiele (c) Bewertung von Algorithmen anhand der Anzahl der nötigen Vergleiche sowie bezüglich der Rechenzeit und des Speicherplatzbedarfs		Kapitel 7 Sortieren und Suchen auf Feldern Projekteinstieg 2: Sortieren 7.3 Ordnung ist das halbe Leben!? – Sortieren Selection Sort, Insertion Sort, Bubble Sort

Unterrichtsvorhaben EF-VII

Thema: Leben in der digitalen Welt – Immer mehr Möglichkeiten und immer mehr Gefahren!?

Leitfragen:

- *Welche Entwicklungen, Ideen und Erfindungen haben zur heutigen Informatik geführt?*
- *Welche Auswirkungen hat die Informatik für das Leben des modernen Menschen?*

Vorhabenbezogene Konkretisierung:

Das Unterrichtsvorhaben stellt die verschiedenen Entwicklungsstränge der Informatik in den Fokus. Darüber hinaus wird beispielhaft analysiert und bewertet, welche Möglichkeiten und Gefahren die moderne Informationsverarbeitung mit sich bringt.

Im ersten Schritt des Unterrichtsvorhabens wird anhand von Themenkomplexen entscheidende Entwicklungen der Informatik erarbeitet. Dabei werden auch übergeordnete Tendenzen identifiziert. Ausgehend von dieser Betrachtung kann die aktuelle Informatik hinsichtlich ihrer Leistungsfähigkeit analysiert werden. Dabei soll herausgestellt werden, welche positiven und negativen Folgen Informatiksysteme mit sich bringen können.

Zeitbedarf: 12 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Schriftzeichen, Rechenmaschine, Computer (a) Anhand von Schwerpunkten, wie z.B. Datenspeicherung, Maschinen, Vernetzung sollen wichtige Entwicklungen der Informatik vorgestellt werden. (b) Anhand der unterschiedlichen Schwerpunkte sollen universelle Tendenzen der Entwicklung der Informationsverarbeitung erarbeitet werden.	Die SchülerInnen... - bewerten anhand von Fallbeispielen die Auswirkungen des Einsatzes von Informatiksystemen (A), - erläutern wesentliche Grundlagen der Geschichte der digitalen Datenverarbeitung (A) - nutzen das Internet zur Recherche, zum Datenaustausch und zur Kommunikation (K)	Exkurs „Geschichte der Informatik“
2. Die Informationsverarbeitung und ihre Möglichkeiten und Gefahren (a) Ausgehend von 1. werden Tendenzen der Entwicklung der Informatik erarbeitet (b) Informatik wird als Hilfswissenschaft klassifiziert, die weit über ihren originären Bereich hinaus Effizienz- und Leistungssteigerungen erzeugt (c) Anhand von Fallbeispielen werden technische und organisatorische Vorteile, sowie deren datenschutzrechtlichen Nachteile betrachtet.		Exkurs „Informatik und Gesellschaft“

3.3. Qualifikationsphase

3.3.1. Übersicht Unterrichtsvorhaben Q1

Die Beschreibung stimmt mit dem vom Ministerium bereitgestellten Beispiel und der zum entsprechenden Lehrwerk passenden Übersicht vom Schöningh-Verlag überein. Die Reihenfolge und einzelne Schwerpunkte hat die Fachkonferenz jedoch angepasst.

<u>Unterrichtsvorhaben Q1-I</u> Thema: Wdh. der objektorientierten Modellierung und Programmierung anhand einer kontextbezogenen Problemstellung Zentrale Kompetenzen: • Argumentieren • Modellieren • Implementieren • Darstellen und Interpretieren Inhaltsfelder: • Daten und ihre Strukturierung • Algorithmen • Formale Sprachen und Automaten • Informatiksysteme Inhaltliche Schwerpunkte: • Objekte und Klassen • Analyse, Entwurf und Implementierung von Algorithmen • Syntax und Semantik einer Programmiersprache • Wirkung von Informatiksystemen Zeitbedarf: 6 Stunden	<u>Unterrichtsvorhaben Q1-II</u> Thema: Modellierung und Implementierung von Anwendungen mit dynamischen, linearen Datenstrukturen Zentrale Kompetenzen: • Modellieren • Implementieren • Darstellen und Interpretieren • Kommunizieren und Kooperieren Inhaltsfelder: • Daten und ihre Strukturierung • Algorithmen • Formale Sprachen und Automaten Inhaltliche Schwerpunkte: • Objekte und Klassen • Analyse, Entwurf und Implementierung von Algorithmen • Algorithmen in ausgewählten informatischen Kontexten • Syntax und Semantik einer Programmiersprache Zeitbedarf: 15 Stunden
--	---

<u>Unterrichtsvorhaben Q1-III</u> Thema: Suchen und Sortieren auf linearen Datenstrukturen Zentrale Kompetenzen: • Argumentieren • Darstellen und Interpretieren • Modellieren • Implementieren Inhaltsfelder: • Algorithmen • Formale Sprachen und Automaten Inhaltliche Schwerpunkte: • Analyse, Entwurf und Implementierung von Algorithmen • Algorithmen in ausgewählten informatischen Kontexten • Syntax und Semantik einer Programmiersprache Zeitbedarf: 12 Stunden	<u>Unterrichtsvorhaben Q1-IV</u> Thema: Modellierung und Nutzung von ereignisgesteuerten grafischen Benutzeroberflächen in Anwendungskontexten Zentrale Kompetenzen: • Argumentieren • Modellieren • Implementieren • Darstellen und Interpretieren • Kommunizieren und Kooperieren Inhaltsfelder: • Daten und ihre Strukturierung • Informatiksysteme • Algorithmen Inhaltliche Schwerpunkte: • Ereignisse verarbeiten • Vererbungs- und Objektstrukturen • Syntax und Semantik einer Programmiersprache Zeitbedarf: 15 Stunden
<u>Unterrichtsvorhaben Q1-V</u> Thema: Aufbau von Kommunikation in Netzwerken Zentrale Kompetenzen: • Argumentieren • Darstellen und Interpretieren • Kommunizieren und Kooperieren Inhaltsfelder: • Informatiksysteme • Informatik, Mensch und Gesellschaft Inhaltliche Schwerpunkte: • Einzelrechner und Rechnernetzwerke • Sicherheit • Nutzung von Informatiksystemen, Wirkungen der Automatisierung Zeitbedarf: 7 Stunden	

Gesamter Zeitbedarf für Q1: 55 Unterrichtsstunden (a 60 Minuten)

3.3.2. Übersicht Unterrichtsvorhaben Q2

<u>Unterrichtsvorhaben Q2-I</u> Thema: Modellierung und Implementierung von Anwendungen mit dynamischen, nichtlinearen Datenstrukturen Zentrale Kompetenzen: • Argumentieren • Darstellen und Interpretieren • Modellieren • Implementieren • Kommunizieren und Kooperieren Inhaltsfelder: • Daten und ihre Strukturierung • Algorithmen • Formale Sprachen und Automaten Inhaltliche Schwerpunkte: • Objekte und Klassen • Analyse, Entwurf und Implementierung von Algorithmen • Algorithmen in ausgewählten informatischen Kontexten • Syntax und Semantik einer Programmiersprache Zeitbedarf: 16 Stunden	<u>Unterrichtsvorhaben Q2-II</u> Thema: Nutzung und Modellierung von relationalen Datenbanken in Anwendungskontexten Zentrale Kompetenzen: • Argumentieren • Modellieren • Implementieren • Darstellen und Interpretieren • Kommunizieren und Kooperieren Inhaltsfelder: • Daten und ihre Strukturierung • Algorithmen • Formale Sprachen und Automaten • Informatik, Mensch und Gesellschaft Inhaltliche Schwerpunkte: • Datenbanken • Algorithmen in ausgewählten informatischen Kontexten • Syntax und Semantik einer Programmiersprache • Sicherheit Zeitbedarf: 15 Stunden
<u>Unterrichtsvorhaben Q2-III</u> Thema: Endliche Automaten und formale Sprachen Zentrale Kompetenzen: • Argumentieren • Modellieren • Darstellen und Interpretieren • Kommunizieren und Kooperieren Inhaltsfelder: • Endliche Automaten und formale Sprachen Inhaltliche Schwerpunkte: • Endliche Automaten • Grammatiken regulärer Sprachen • Möglichkeiten und Grenzen von Automaten und formalen Sprachen Zeitbedarf: 12 Stunden	

Gesamter Zeitbedarf für Q2: 43 Unterrichtsstunden

3.3.3. Konkretisierte Unterrichtsvorhaben Q1 und Q2

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?*
- *Welche Auswirkungen hat die informatisch-technische Entwicklung auf das Leben der Menschen?*

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

Zeitbedarf: 6 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Wiederholung der grundlegenden Konzepte der objektorientierten Programmierung a) Sichtweise der objektorientierten Informatik auf die Welt b) OOP als informatikspezifische Modellierung der Realität c) Schritte der Softwareentwicklung	Die SchülerInnen... - ermitteln bei der Analyse von Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M), - 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), - verwenden bei der Modellierung geeigneter Problemstellungen Möglichkeiten der Polymorphie - 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 Beziehungen in Diagrammen grafisch dar (D), - dokumentieren Klassen (D), - stellen die Kommunikation zwischen Objekten grafisch dar (D), - untersuchen und bewerten anhand von Fallbeispielen Auswirkungen des Einsatzes von Informatiksystemen sowie Aspekte der Sicherheit von Informatiksystemen, des Datenschutzes 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).	Kapitel 1 Konzepte des objektorientierten Modellierens 1.1 Modellierung der Realität 1.2 Die Welt ist voller Objekte Projekteinstieg: Klassendesign – step by step
2. Erweiterung der objektorientierten Programmierung a) Umsetzung einer Anforderung in Entwurfs- und Klassendiagramm b) Objektkommunikation im Sequenzdiagramm c) Klassendokumentation d) Umsetzung von Teilen der Modellierung		1.3 Gut geplant – Klassenentwurf 1.4 Hierarchien machen's einfacher – Vererbung
3. Mensch und Technik a) InformatikerInnen verändern die Welt b) Automatisierung des Alltags durch Informatik		Die digitale Welt 001 – Mensch und Technik

Unterrichtsvorhaben Q1-II

Thema: Modellierung und Implementierung dynamischer Listenstrukturen und deren Anwendungen

Leitfragen:

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

Vorhabenbezogene Konkretisierung:

Ausgehend von 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 (Vokabeltrainer) erarbeitet und in einem weiteren Beispiel vertieft (LED) wurden, werden abschließend in einem Anwendungskontext verschiedene lineare Datenstrukturen angewendet. Die Modellierung erfolgt beim gesamten Vorhaben in Entwurfs- und Implementationsdiagrammen.

Zeitbedarf: 15 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Die Datenstruktur Feld a) Erarbeitung der Anforderungen an eine Datenstruktur b) Wiederholung der Datenstruktur Array, Eigenschaften der Datenstruktur, Standardoperationen für ein und zweidimensionale Arrays c) Modellierung und Implementierung von Anwendungen	Die SchülerInnen... - erläutern Operationen dynamischer (linearer) Datenstrukturen (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), - interpretieren Fehlermeldungen und korrigieren den Quellcode (I), - 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) - ordnen Klassen, Attributen und Methoden ihre Sichtbarkeitsbereiche zu (M) - dokumentieren Klassen (D) - implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I) - verwenden bei der Modellierung geeigneter Problemstellungen Möglichkeiten der Polymorphie (M) - untersuchen und bewerten anhand von Fallbeispielen Auswirkungen des Einsatzes von Informatiksystemen sowie Aspekte der Sicherheit von Informatiksystemen, des Datenschutzes 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).	Kapitel 2 Lineare Datenstrukturen 2.1 Anforderungen an eine Datenstruktur 2.2 Datenansammlungen fester Größe – Arrays 2.3 Wer zuerst kommt ... – Schlangen 2.4 Daten gut abgelegt – Stapel 2.5 Flexibel für alle Fälle – lineare Listen
2. Die Datenstruktur Schlange a) Modellierung und Implementierung der Verknüpfung von Objekten b) Generische Typen, Trennung von Verwaltung und Inhalt dyn. DS. c) Erläuterung von Problemstellungen, die nach dem FIFO-Prinzip bearbeitet werden d) Funktionalität der Schlange unter Verwendung der Klasse <i>Queue</i> ; Erschließen der Standardoperationen e) Modellierung und Implementierung einer Anwendung auf der Basis einer Anforderungsbeschreibung mit Objekten der Klasse <i>Queue</i>		
3. Die Datenstruktur Stapel a) Erläuterung von Problemstellungen, die nach dem LIFO-Prinzip bearbeitet werden b) Funktionalität der Klasse Stapel unter Verwendung der Klasse <i>Stack</i> , Erschließen der Standardoperationen c) Modellierung und Implementierung einer Anwendung auf Basis einer Anforderungsbeschreibung mit Objekten der Klasse <i>Stack</i> d) Modellierung und Implementierung einer Anwendung unter Verwendung verschiedener Datenstrukturen (Objekte der Klassen <i>Queue</i> , <i>Stack</i> und <i>Array</i> (<i>Palindrom</i>))		
4. Die Datenstruktur Liste a) Analyse der Möglichkeiten bisheriger Datenstrukturen zwecks Bestimmung notwendiger Funktionalitäten für komplexere Anwendungen (Abgrenzung zu <i>Stack/Queue</i> , zusätzliche Fähigkeiten der Klasse <i>List</i>) b) Erarbeitung der Funktionalität der Liste unter Verwendung der Klasse <i>List</i> c) Modellierung und Implementierung einer Anwendung mit Objekten der Klasse <i>List</i> d) Modellierung und Implementierung einer Anwendung unter Verwendung verschiedener Datenstrukturen (<i>Stack</i> , <i>Queue</i> , <i>List</i>)		

Unterrichtsvorhaben Q1-III

Thema: Rekursive und iterative Algorithmen im Anwendungskontext Sortieren und Suchen

Leitfragen:

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

Vorhabenbezogene Konkretisierung:

Zunächst werden anhand eines Anwendungsbeispiels übergreifende Algorithmeigenschaften (wie Korrektheit, Effizienz und Verständlichkeit) erarbeitet und Schritte der Algorithmusentwicklung 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 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 Sortiervorgehen erarbeitet und implementiert sowie ihre Effizienz bewertet.

Zeitbedarf: 12 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Eigenschaften von Algorithmen a) Qualitätseigenschaften von Algorithmen b) Strukturierung von Algorithmen mit Hilfe der Strategien „Modularisierung“ und „Teile und Herrsche“ c) Analyse und Entwicklung von rekursiven Algorithmen	Die SchülerInnen... - analysieren und erläutern Algorithmen und Programme (A), - modifizieren Algorithmen und Programme (I), - stellen iterative und rekursive Algorithmen umgangssprachlich und grafisch dar (D), - 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 Sortierv Verfahren (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), - untersuchen und bewerten anhand von Fallbeispielen Auswirkungen des Einsatzes von Informationssystemen sowie Aspekte der Sicherheit von Informationssystemen, des Datenschutzes und des Urheberrechts (A), - untersuchen und bewerten Problemlagen, die sich aus dem Einsatz von Informationssystemen ergeben, hinsichtlich rechtlicher Vorgaben, ethischer Aspekte und gesellschaftlicher Werte unter Berücksichtigung unterschiedlicher Interessenlagen (A).	Kapitel 3 Algorithmen 3.1 Ohne Algorithmen läuft nichts 3.2 Teile die Arbeit – rekursive Algorithmen 3.3 Suchen – iterativ und rekursiv Projekteinstieg: Laufzeitanalyse experimentell 3.4 Sortieren – iterativ und rekursiv
2. Suchen in Listen und Arrays a) Lineare Suche in Listen und Arrays b) Binäre Suche in einem Array c) Untersuchung der beiden Verfahren bzgl. Laufzeit und Speicherplatzbedarf		
3. Sortieren auf Listen und Arrays a) Entwicklung und Implementierung eines iterativen Sortiervfahrens für eine Liste (Sortieren durch Einfügen) b) Entwicklung und Implementierung eines rekursiven Sortiervfahrens für eine Liste (Quicksort) c) Untersuchung der beiden Verfahren bzgl. Laufzeit und Speicherplatzbedarf d) Weitere Sortiervverfahren auf Listen und Arrays (Sortieren durch Auswählen, Mergesort)		
4. Verantwortung der Informatik a) Der Unterschied zwischen Anwender und Produzent von Informationssystemen b) Informatik und Ethik		Die digitale Welt 011 – Verantwortung der Informatik

Unterrichtsvorhaben Q1-V

Thema: Aufbau von Kommunikation in Netzwerken

Leitfragen:

- Was macht menschliche Kommunikation aus?
- Welchen Stellenwert haben technische/ informatische Hilfsmittel für die Kommunikation?
- Wie werden Daten in einem Netzwerk zwischen den Kommunikationspartnern übertragen?
- Wie ist die Arbeitsteilung in Netzwerken gestaltet?
- Wie kann sicher in Netzwerken kommuniziert werden?

Vorhabenbezogene Konkretisierung:

Ausgehend von alltäglicher Face-to-Face-Kommunikation werden die Grundprinzipien sowie die Bewertungskriterien von Kommunikation erläutert. Das Netzwerk wird als vorteilhafte Kommunikationsstruktur dargestellt und anhand von Topologien und Reichweiten kategorisiert. Ausgehend davon wird der Protokollbegriff entwickelt und anhand des TCP/IP-Schichtenmodells analysiert. Anschließend wird das Client-Server-Prinzip vorgestellt und angewandt.

Sichere Kommunikation in Netzen ist nur dank kryptografischer Verfahren möglich. Stellvertretend werden zwei symmetrische und ein asymmetrisches Verfahren erläutert, angewandt und bewertet.

Zeitbedarf: 7 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Technische Kommunikation als Fortführung natürlicher Kommunikation a) Kommunikation im Shannon-Weaver-Modell b) Kriterien von technischen Kommunikationsarten c) Die Geschichte der technischen Kommunikation	Die SchülerInnen... - beschreiben und erläutern Netzwerk-Topologien, die Client-Server-Struktur und Protokolle sowie ein Schichtenmodell in Netzwerken (A), - analysieren und erläutern Eigenschaften und Einsatzbereiche symmetrischer und asymmetrischer Verschlüsselungsverfahren (A), - nutzen bereitgestellte Informatiksysteme und das Internet reflektiert zur Erschließung, Aufbereitung und Präsentation fachlicher Inhalte (D).	Kapitel 6 Kommunikation in Netzwerken 6.1 Menschen kommunizieren – ohne und mit Technik Projekteinstieg: Kommunikation im Wilden Westen
2. Aufbau von Netzwerken und Kommunikationsregeln a) Das Netzwerk als Organisationsprinzip der Kommunikation und Möglichkeiten der Ausformung b) Geregelte technische Kommunikation durch Protokolle in Schichtenmodellen		6.2 Ohne Protokoll läuft nichts – Netzwerke
3. Aufgabenteilung in Netzwerken durch Server und Client a) Aufbau und Aufgaben der Client-Server-Struktur b) Protokolle zwischen Client und Server		6.3 Einer für alle – Client-Server-Struktur
4. Kryptologie a) Veranschaulichen und Anwenden von symmetrischen und asymmetrischen kryptographischen Verfahren (Caesar, Vigenère, RSA) b) Bewertung der Verfahren hinsichtlich ihrer Sicherheit und ihrem Aufwand		Die digitale Welt 111 – Kryptologie

Unterrichtsvorhaben Q2-I

Thema: Modellierung und Implementierung von Anwendungen mit dynamischen, nichtlinearen Datenstrukturen am Beispiel der Binärbäume und am Beispiel der Graphen

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?*

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 beiden Konstruktoren der Klasse *BinaryTree* thematisiert. Möchte man Daten geordnet speichern, 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 *ComparableContent* realisiert, das alle Klassen, dessen Objekte in einem Suchbaum verwaltet werden sollen, implementieren müssen. Auf diese Weise wird ein Anwendungskontext (Benutzerverwaltung) mithilfe der Klassen *BinarySearchTree* und *ComparableContent* modelliert und (ggf. in Teilen) implementiert.

Zeitbedarf: 16 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Aufbau von Baumstrukturen und Grundbegriffe a) Erarbeitung der Begriffe Wurzel, Knoten, Blatt, Kante, Grad eines Knotens und eines Baumes, Pfad, Tiefe, Ebene, Teilbaum b) Aufbau und Darstellung von Baumstrukturen in verschiedenen Anwendungskontexten	Die SchülerInnen... - 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).	Kapitel 5 Nicht-lineare Datenstrukturen 5.1 Spielen mit Struktur – Baumstrukturen Projekteinstieg 1: Spielbäume 5.2 Zwei Nachfolger sind genug! - Binäre Bäume Implementation des Projekts Ahnenbaum 5.3 Wer Ordnung hält, spart Zeit beim Suchen – Binäre Suchbäume Projekteinstieg 2: Binäre Suchbäume Implementation des Projekts Benutzerverwaltung
2. Binäre Bäume a) rekursiver Aufbau eines binären Baums b) Traversierungen (pre-, in-, postorder) c) Modellierung eines Binärbaums in einem Anwendungskontext mit Hilfe der Klasse BinaryTree (als Entwurfs- und Implementationsdiagramm) d) Implementation einer Anwendung der Datenstruktur binärer Baum (ggf. in Teilen)		
3. Binäre Suchbäume a) Prinzip des binären Suchbaums, Ordnungsrelation b) Operationen auf dem binären Suchbaum (Suchen, Einfügen, Löschen, sortierte Ausgabe) c) Modellierung eines binären Suchbaums in einem Anwendungskontext mit Hilfe der Klasse BinarySearchTree (als Entwurfs- und Implementationsdiagramm) und dem Interface Item d) Implementation einer Anwendung der Datenstruktur binärer Suchbaum (ggf. in Teilen)		

Unterrichtsvorhaben Q2-II

Thema: Modellierung und Nutzung relationaler Datenbanken und Implementierung

Leitfragen:

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

Vorhabenbezogene Konkretisierung:

Am Beispiel eines Online-Buchhandels wird der Aufbau einer Datenbank sowie wichtige Grundbegriffe erarbeitet. Die SchülerInnen nehmen dabei zunächst die Sicht der Anwender ein, 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: 15 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Nutzung von relationalen Datenbanken a) 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 b) SQL-Abfragen - Erarbeitung der grundlegenden Sprachelemente von SQL (SELECT(DISTINCT), FROM, WHERE, JOIN) - 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) c) Vertiefung an einem weiteren Datenbankbeispiel	Die SchülerInnen... - 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), - verwenden die Syntax und Semantik einer Datenbankabfragesprache, um Informationen aus einem Datenbanksystem zu extrahieren (I), - ermitteln Ergebnisse von Datenbankabfragen über mehrere verknüpften Tabellen (D), - ermitteln für anwendungsbezogene Problemstellungen Entitäten, zugehörige Attribute, Relationen und Kardinalitäten (M), - stellen Entitäten mit ihren Attributen und die Beziehungen zwischen Entitäten in einem Entity-Relationship-Diagramm grafisch dar (D), - modifizieren eine Datenbankmodellierung (M), - modellieren zu einem Entity-Relationship-Diagramm ein relationales Datenbankschema (M), - bestimmen Primär- und Sekundärschlüssel (M), - analysieren und erläutern eine Datenbankmodellierung (A), - erläutern die Eigenschaften normalisierter Datenbankschemata (A), - überprüfen Datenbankschemata auf vorgegebene Normalisierungseigenschaften (D), - überführen Datenbankschemata in die 1. bis 3. Normalform (M).	Kapitel 8 Datenbanken 7.1 Wissen speichern und verwalten – Datenbanksysteme 7.2 Daten anordnen mit Tabellen Projekteinstieg: Tabellen 7.3 Daten filtern mit SQL 7.4 Komplexe Filter 7.5 Datenbankentwurf 7.6 Umsetzung des ER-Modells Wiederaufgriff des Projekteinstiegs 7.7 Datenbanken verbessern durch Normalformen

Unterrichtsvorhaben Q2-III

Thema: Endliche Automaten und formale Sprache

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?
- Was sind die strukturellen Hauptbestandteile eines Computers und wie kann man sich die Ausführung eines maschinennahen Programms mit diesen Komponenten vorstellen?
- Welche Möglichkeiten bieten Informatiksysteme und wo liegen ihre Grenzen?

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 der 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: 12 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Endliche Automaten a) Erarbeitung der formalen Beschreibung eines Mealy-Automaten und der Darstellungsformen b) Erarbeitung der formalen Beschreibung eines deterministischen endlichen Automaten (DEA) sowie dessen Darstellungsformen; Erschließung der Fachbegriffe Alphabet, Wort, (akzeptierte) Sprache, Determinismus c) Analyse der Eigenschaften von DEAs durch die Modellierung eines Automaten zu einer gegebenen Problemstellung, der Modifikation eines Automaten sowie die Überführung der gegebenen Darstellungsform in eine andere	Die SchülerInnen... - 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 endliche 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), - ermitteln die formale Sprache, die durch eine Grammatik erzeugt wird (A), - entwickeln zu einer regulären Sprache eine Grammatik, die die Sprache erzeugt (M)	Kapitel 4 Endliche Automaten und formale Sprachen 4.1 Endliche Automaten Projekteinstieg: Schatzsuche
2. Grammatiken regulärer Sprachen a) Erarbeitung der formalen Beschreibung einer regulären Grammatik (Sprache, Terminal und Nicht-Terminal, Produktionen und Produktionsvorschriften) b) Analyse der Eigenschaften einer regulären Grammatik durch deren Entwicklung und Modellierung zu einer gegebenen Problemstellung.	- 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 im Anwendungszusammenhang auf (A), - erläutern die Ausführung eines einfachen maschinennahen Programms sowie die Datenspeicherung auf einer „Von-Neumann-Architektur“ (A), - untersuchen und beurteilen Grenzen des Problemlösens mit Informatiksystemen (A).	4.2 Formale Sprachen
3. Übungen und Vertiefungen Verwendung endlicher Automaten und Grammatiken regulärer Sprachen		4.5 Prüfungsvorbereitung
4. Grundsätzliche Arbeitsweise eines Computers und Grenzen der Berechenbarkeit a) Von-Neumann-Architektur und die Ausführung maschinennaher Programme b) Grenzen der Berechenbarkeit anhand des Halteproblems, nicht effizient berechenbare Probleme		Die digitale Welt 101 – Maschinennahe Programmierung Die digitale Welt 100 – Berechenbarkeit

4. Leistungsbewertung

4.1. Rechtliche Grundlagen und Prinzipien

Auf der Grundlage von §13 - §17 der APO-GOST, das Schulgesetz § 48 (Grundsätze der Leistungsbewertung) sowie Kapitel 3 des Kernlehrplans Informatik für die gymnasiale Oberstufe hat die Fachkonferenz im Einklang mit dem entsprechenden schulbezogenen Konzept die nachfolgenden Grundsätze zur Leistungsbewertung und Leistungsrückmeldung beschlossen. Daraus erwächst für die Schulen konkret die Aufgabe, sowohl die individuellen Schwächen und Stärken der SchülerInnen zu diagnostizieren und gegebenenfalls die Defizite durch gezielte Maßnahmen zu beseitigen sowie besondere Begabungen zu fördern. Zudem haben Noten eine wichtige gesellschaftliche Funktion. Diese mit Inhalt zu füllen, ist Aufgabe der Schulen. Noten dienen als zentrale Entscheidungsgrundlage über Schullaufbahnen, Versetzungen und Abschlüsse. Zeugnisse sind ein 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 nachfolgenden Absprachen stellen die Minimalanforderungen an das lerngruppenübergreifende gemeinsame Handeln der Fachgruppenmitglieder dar. Bezogen auf die einzelne Lerngruppe kommen ergänzend weitere der in den Folgeabschnitten genannten Instrumente der Leistungsüberprüfung zum Einsatz.

4.1.1. Transparenz der Leistungsbeurteilung

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 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 und SchülerInnen sind Noten eine einfache und zentrale Information über den Leistungsstand. Sie bieten 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 halbjährlich stattfindenden Konferenzen über die Schwierigkeiten und besonderen Probleme einzelner SchülerInnen wie auch Klassen zu beraten und Maßnahmen zur Verbesserung zu beschließen.

4.1.2. Grundsätze der Leistungsbewertung

Die Fachkonferenz Informatik legt die Kriterien für die Leistungsbeurteilung fest. Die LehrerInnen machen diese Kriterien den SchülerInnen 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).
- Leistungsbewertung bezieht sich auf die im Unterricht erworbenen und geförderten Kompetenzen.
- Die Lehrperson gibt den SchülerInnen 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.

4.2. Beurteilungsbereich schriftliche Leistungen/Klassenarbeiten

4.2.1. Bewertung von Kursarbeiten in der Sekundarstufe I

Schriftliche Arbeiten (Kursarbeiten oder Projektarbeiten inkl. Dokumentation) dienen der 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 schriftliche Arbeiten abhängig von den kontinuierlich ansteigenden Anforderungen entsprechend dem Lehrplan.

Die Anzahl und Dauer der schriftlichen Arbeiten im Fach Informatik hat die Fachkonferenz im Rahmen der Vorgaben der APO–SI für den Wahlpflichtbereich wie folgt festgelegt:

Anzahl und zeitlicher Umfang der Klassenarbeiten

Jahrgangsstufe	Zahl der Klassenarbeiten	Dauer der Arbeit
9.1	2	60 Minuten
9.2	2	60 Minuten
10.1	2	60 Minuten
10.2	2	60 Minuten

Die Verteilung der Arbeiten auf das Jahr ergibt sich aus der Länge der Schulhalbjahre. In der Regel werden die Termine der Kursarbeiten aller Wahlpflichtfächer zentral durch die Koordination der Mittelstufe vorgegeben. Grundsätzlich ist es möglich pro Schuljahr eine Projektarbeit als schriftliche Arbeit zu werten. Projektarbeiten können auch auf mehrere Unterrichtsstunden verteilt angefertigt werden. Grundlage der Projektbewertung ist die Dokumentation der Projektarbeit. Vorgaben hierzu werden je nach gestellter Arbeit den SchülerInnen mitgeteilt.

Die Bewertung der schriftlichen Leistungen in den Kursarbeiten erfolgt über ein Raster mit Hilfspunkten, die im Erwartungshorizont den einzelnen Teillösungen zugeordnet sind. Die Note ausreichend soll bei Erreichen von 50% der Hilfspunkte erteilt werden. Die Korrektur der schriftlichen Leistungen erfolgt transparent anhand eines Erwartungshorizontes. Klassenarbeiten werden zeitnah zurückgegeben und besprochen.

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. Es wird empfohlen, die Kursarbeiten in angemessenem Vorlauf zum Klassenarbeitstermin zu konzipieren, damit Zeit bleibt, die SchülerInnen auf alle zu überprüfenden Kompetenzen vorzubereiten – auch auf solche, die nicht Schwerpunkte der Kursarbeiten sind.

In der 5. Und 6. Klasse wird empfohlen eine schriftliche Leistungsüberprüfung in Form einer Hausaufgabenüberprüfung oder eines Tests durchzuführen. Dabei sollte die Überprüfung einen zeitlichen Rahmen von maximal 25 Minuten nicht überschreiten. Das Benotungsraster kann sich an folgendem Beispiel orientieren:

Punktesystem zur Benotung von schriftlichen Überprüfungen und Kursarbeiten

Punkte in [%]	Note
0	6
20	5 minus
28	5
37	5 plus
46	4 minus
50,5	4
55	4 plus
59,5	3 minus
64	3
68	3 plus
73	2 minus
77,5	2
82	2 plus
86,5	1 minus
91	1
95,5	1 plus

4.2.2. Bewertung von Klausuren in der Sekundarstufe II

Ab der Stufe EF werden gemäß den Vorgaben der Lehrpläne, der Vorgaben für das Abitur und des schulinternen Curriculums Klausuren geschrieben. Bei der Formulierung von Aufgaben werden die für die Abiturprüfungen geltenden Operatoren des Faches Informatik schrittweise eingeführt, erläutert und dann im Rahmen der Aufgabenstellungen für die Klausuren benutzt.

In der Einführungsphase wird aufgrund nur einer Klausur der schriftliche Anteil etwa im Verhältnis 40:60 zur sonstigen Mitarbeit gewertet. Anstelle einer Klausur kann gemäß dem Beschluss der Lehrerkonferenz in Q 1.2 eine Facharbeit geschrieben werden.

Anzahl und zeitlicher Umfang der Klausuren

Halbjahr	Grundkurs		Leistungskurs		Hinweise
	Anzahl	Dauer	Anzahl	Dauer	
EF /I	1	2	-	-	
EF/II	1	2	-	-	
Q1/I	2	2	2	3	
Q1/II	2	2	2	3	Die 1. Klausur kann durch eine Facharbeit ersetzt werden.
Q2/I	2	3	2	4	
Q2/II	1	3	1	4,15	Im GK nur für SchülerInnen, die Informatik als 3. Abiturfach gewählt

Mit Ausnahme der Klausur in Q2/II handelt es sich um Unterrichtsstunden, in Q2/II um Zeitstunden.

Den Klausuren wird folgende Verteilung der Anforderungsbereiche zugrunde gelegt:

- Anforderungsbereich I: ca. 30 % (Reproduktion – Wiedergabe von Kenntnissen und Modellen, Beschreibung unter Verwendung gelernter Arbeitstechniken (Grafen, Tabellen, etc.))
- Anforderungsbereich II: ca. 50 % (Transfer – selbständiges Auswählen, Anordnen, Verarbeiten und Darstellen bekannter Sachverhalte unter vorgegebenen, neuen Gesichtspunkten, die aber in einem eingeübten Zusammenhang stehen; Auswerten von unbekanntem Material unter bekanntem Aspekt)
- Anforderungsbereich III: ca. 20 % (problemlösendes Denken – planmäßiges Verarbeiten komplexer Gegebenheiten mit dem Ziel selbständiger Lösung, Gestaltung, Deutung, Beurteilung; dabei werden gelernte Methoden zur Lösung der Aufgabe selbständig neu kombiniert bzw. verändert, um sie der neuen Problemstellung anzupassen)

In den Aufgabenstellungen der Klausuren sollen die im Zentralabitur verwendeten Operatoren ab der Stufe EF verwendet werden, damit diese den SchülerInnen von Beginn an vertraut sind.

4.2.3.Kriterien

Die Bewertung der schriftlichen Leistungen in Klausuren erfolgt über ein Raster mit Hilfspunkten, die im Erwartungshorizont den einzelnen Kriterien zugeordnet sind. Die Note ausreichend (5 Punkte) soll bei Erreichen von 45 % der Hilfspunkte erteilt werden.

Grundsätzlich richtet sich die Korrektur nach den Vorlagen, die aus den bisher durchgeführten Klausuren des Zentralabiturs bekannt sind. Sie muss für die SchülerInnen nachvollziehbar sein. Wenn formale Korrekturzeichen nicht genügen, dann sind sie durch sachbezogene Hinweise am Rand oder am Ende der Arbeit zu ergänzen. Die Benotung der Klausuren in der S II soll sich an folgendem Punktesystem orientieren:

Punktesystem zur Benotung von Klausuren

Punkte in [%]	Punkte (Note)	Note
0	0	6
20	1	5 minus
26	2	5
33	3	5 plus
40	4	4 minus
45	5	4
50	6	4 plus
55	7	3 minus
60	8	3
65	9	3 plus
70	10	2 minus
75	11	2
80	12	2 plus
85	13	1 minus
90	14	1
95	15	1 plus

Die Begründung einer Note beinhaltet eine knappe Darstellung der positiven und negativen Anteile der Arbeit in den einzelnen Anforderungsbereichen, eine Information über Lernerfolg, Defizite und die Verwendung von Fachsprache.

Um die Leistungsbewertung durch die Fachkollegen und die Anforderungen an die SchülerInnen zu vereinheitlichen, werden die Klausuraufgaben im Fachkollegium ausgetauscht und hinsichtlich des Anforderungsprofils überprüft. Darüber hinaus findet ein regelmäßiger Austausch von parallel unterrichtenden Kollegen statt. Hierbei werden methodische Schwerpunkte und grundlegende Bewertungskriterien vereinbart, die ein einheitliches Anforderungsprofil sicherstellen.

4.2.4. Grundsätze der Leistungsrückmeldung und Beratung

Die Grundsätze der Leistungsbewertung werden zu Beginn eines jeden Halbjahres den SchülerInnen 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 (mit angemessener Bedenkzeit),
- als Quartalsfeedback und
- zu Eltern- oder SchülerInnensprechtagen.

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.

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

4.3. Bewertung der sonstigen Mitarbeit in der Sekundarstufe I

Den SchülerInnen werden die Kriterien zum Bewertungsbereich sonstige Leistungen zu Beginn des Schuljahres genannt. Bei der Unterrichtsgestaltung sind den SchülerInnen hinreichend Möglichkeiten zur Mitarbeit zu eröffnen, z.B. durch

- Beteiligung am Unterrichtsgespräch
- Zusammenfassungen zur Vor- und Nachbereitung des Unterrichts
- Präsentation von Arbeitsergebnissen
- Mitarbeit in Partner- und Gruppenarbeitsphase
- Schriftliche Bearbeitung von Aufgaben im Unterricht
- Führen eines Lernblogs zur Dokumentation der Unterrichtsinhalte
- Praktische Leistungen am Computer als Werkzeug im Unterricht
- Protokolle und Referate
- Kürzere Projektarbeiten
- Lernerfolgsüberprüfungen und schriftliche Übungen

Der Bewertungsbereich „sonstige Leistungen“ erfasst die Qualität und Kontinuität der Beiträge, die die SchülerInnen im Unterricht erbringen. Diese Beiträge sollen unterschiedliche mündliche und schriftliche Formen in enger Bindung an die Aufgabenstellung, die inhaltliche Reichweite und das Anspruchsniveau der jeweiligen Unterrichtseinheit umfassen.

Bewertungskriterien

Die Bewertungskriterien für eine Leistung müssen auch für SchülerInnen transparent, klar und nachvollziehbar sein. Die folgenden allgemeinen Kriterien gelten sowohl für die schriftlichen als auch für die sonstigen Formen der Leistungsüberprüfung:

- Qualität der Beiträge
- Kontinuität der Beiträge
- Sachliche Richtigkeit
- Angemessene Verwendung der Fachsprache
- Darstellungskompetenz
- Komplexität/Grad der Abstraktion
- Selbstständigkeit im Arbeitsprozess
- Einhaltung gesetzter Fristen
- Präzision
- Differenziertheit der Reflexion
- Bei Gruppenarbeiten
 - Einbringen in die Arbeit der Gruppe
 - Durchführung fachlicher Arbeitsanteile
- Bei Projekten
 - Dokumentation des Arbeitsprozesses
 - Grad der Selbstständigkeit
 - Qualität des Produktes
 - Reflexion des eigenen Handelns
 - Kooperation mit dem Lehrenden / Aufnahme von Beratung

Grundsätze der Leistungsrückmeldung und Beratung

Die Leistungsrückmeldung findet in mündlicher oder schriftlicher Form statt. Sie kann auch an Eltern und/oder SchülerInnensprechtagen oder in Form von individuellen Lern-/Förderempfehlungen erfolgen. Die von allen SchülerInnen verbindlich zu führende schriftliche Dokumentation (Kursnotizbuch) wird insgesamt zweimal pro Halbjahr bewertet.

4.4. Individuelle Förderung

Die LehrerInnen beobachten die individuellen Leistungen in allen Bereichen der Informatik über einen längeren Zeitraum, um auf dieser Grundlage ein Leistungsbild zu erhalten. Neben der Orientierung an den Kompetenzstandards der jeweiligen Jahrgangsstufe kann bei der Leistungsbewertung auch die jeweilige Entwicklung der Schülerin oder des Schülers, gemäß der zu beobachtenden Lern- und Denkfortschritte, berücksichtigt werden.

Der Informatikunterricht lebt von der verantwortungsvollen und selbständigen Arbeit der SchülerInnen, so dass die Lehrperson die nötige Zeit hat, bei Bedarf gezielt und individuell zu fördern.

Leistungsstärkere SchülerInnen können ihr Wissen anhand von vertiefenden Problemstellungen erweitern oder als Experten für einzelne (Teil-) Probleme im Rahmen des Lernens durch Lehren ihren Mitschülern beratend zur Seite stehen.

4.5. Bildung der Zeugnisnote

In die Note gehen alle im Unterricht erbrachten Leistungen ein. Dabei nehmen die Beurteilung der Kursarbeiten bzw. Klausuren den gleichen Stellenwert wie die Leistungen im Bereich der Mitarbeit im Unterricht ein. Zudem ist bei der Notenfindung die individuelle Lernentwicklung der SchülerInnen angemessen zu berücksichtigen.

4.6. Bewertung der sonstigen Mitarbeit in der Sekundarstufe II

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

4.6.1. Verbindliche Absprachen der Fachkonferenz

Alle SchülerInnen führen in der Einführungsphase in Kleingruppen ein Kurzprojekt durch und fertigen dazu eine Arbeitsmappe mit Arbeitstagebuch an. Dies wird in die Note für die Sonstige Mitarbeit einbezogen.

In der Qualifikationsphase erstellen, dokumentieren und präsentieren die SchülerInnen in Kleingruppen ein anwendungsbezogenes Softwareprodukt. Dies wird in die Note für die Sonstige Mitarbeit einbezogen.

4.6.2. Leistungsaspekte

Die Beiträge für den Unterricht lassen sich den folgenden drei Aspekten zuordnen. Für ihre Gewichtung soll das Verhältnis 40:40:20 als Orientierung dienen.

Mündliche Leistungen

- Beteiligung am Unterrichtsgespräch bzw. am Gespräch in der Gruppe
- Zusammenfassungen zur Vor- und Nachbereitung des Unterrichts
- Präsentation von Arbeitsergebnissen
- Referate

Praktische Leistungen am Computer

- Implementierung und Test von Informatiksystemen
- Modellierung in Form von Diagrammen (UML)

Sonstige schriftliche Leistungen

- Arbeitsmappe und Arbeitstagebuch zu einem durchgeführten Unterrichtsvorhaben (z.B. Projekt)
- Beiträge in Foren und Wiki-Seiten des Kurses
- Bearbeitung von schriftlichen Aufgaben im Unterricht
- Lernerfolgsüberprüfung durch kurze schriftliche Übungen

Über die Durchführung schriftlicher Übungen entscheidet die Lehrkraft nach Bedarf. Sie dauern ca. 20 Minuten und umfassen den Stoff der letzten ca. 4–6 Stunden.

4.6.3. Kriterien

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 Ziel- und Ergebnisorientierung,
- die angemessene Verwendung der Fachsprache und der Fachmethoden,
- die Anschaulichkeit durch Beispiele und symbolische Darstellungen,
- die Komplexität und den Grad der Abstraktion,
- und die Präzision zu legen.

Bei Gruppenarbeiten auch auf

- das Einbringen in die Arbeit der Gruppe,
- die Arbeitsteilung durch Modularisieren und Einhalten von Absprachen,
- die Durchführung fachlicher Arbeitsanteile,
- das Dokumentieren und Bereitstellen von Teilergebnissen und
- die Qualität des entwickelten Produktes.

Bei Projektarbeit darüber hinaus auf

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

4.7. Bewertung von Facharbeiten und besonderen Lernleistungen

Grundlegende Aussagen zu Facharbeiten, Projektkursen und besonderen Lernleistungen enthält das allgemeine Konzept der Leistungsbewertung der Viktoriaschule. Darüber hinaus einigte sich die Fachkonferenz Informatik auf ein mögliches Raster zur Bewertung der jeweils abzugebenden Arbeit (siehe Facharbeitsbeurteilungsbogen).

Bei Einbringen einer besonderen Lernleistung in die Berechnung der Abiturnote muss die zugrunde gelegte Arbeit einen deutlich größeren Umfang bzw. ein höheres Niveau aufweisen als eine Facharbeit.

5. Qualitätssicherung und Evaluation

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

Es werden jährlich in Fachkonferenzen Erfahrungen ausgetauscht und ggf. Änderungen für den nächsten Durchgang der Einführungsphase beschlossen, um erkannten ungünstigen Entscheidungen schnellstmöglich entgegenwirken zu können.

Außerhalb dieses Curriculums werden ausführliche Modulbeschreibungen entwickelt, die sich auf die aktuell gemachten Unterrichtserfahrungen, dort verwendete Arbeitsblätter und Lernzielkontrollen bzw. auf die entwickelten Klausuren stützen.

6. Anhang

6.1. Beurteilungsbogen zur Facharbeit

Grundsätze:

- Das Verhältnis Form und Inhalt in der Notengebung beträgt 50:50.
- Dieser Beurteilungsbogen ergänzt das schriftliche Gutachten.

Formale Leistung (50 Punkte)

Der Prüfling ...

1	... legt eine vollständige Arbeit vor: korrektes Deckblatt, Inhaltsverzeichnis, Literaturverzeichnis, Erklärung.	4	
2	... hält sich an den vereinbarten Umfang.	2	
3	... hält typographische Vorgaben ein: Einband, Seitenspiegel, Seitenangaben, gliedernde Abschnitte und Überschriften, Schriftgrad.	10	
4	... hält sich an seine Gliederung: Inhaltsverzeichnis und Textteil stimmen überein.	3	
5	... ordnet der Darstellung Quellen zu.	5	
6	... zeigt einen formal korrekten Umgang mit seinen Quellen.	4	
7	... legt ein Literaturverzeichnis mit allen Angaben zu den benutzten Hilfsmitteln vor (Literatur, Abbildungen, Materialien) und speichert alle Internet-Quellen als Anhang.	4	
8	... schreibt sprachlich richtig (Grammatik, Syntax, Orthografie, Zeichensetzung) und stilistisch sicher sowie variabel.	10	
9	... stellt Diagramme nach UML-Standard und Quelltexte in besonderer Formatierung mit Zeilennummern dar.	8	
	Summe	50	

Inhaltliche und gedankliche Leistung (50 Punkte)

Der Prüfling...

1	... wählt in Absprache mit dem Lehrer ein eigenständiges Thema und eine eigenständige Problemstellung.	4	
2	... strukturiert seinen Text schlüssig, stringent sowie gedanklich klar und stellt durchgängig und konsequent einen Themenbezug her.	5	
3	... nutzt genügend, sinnvolle und sachgemäße Quellen; recherchiert umfangreich.	5	
4	... verwertet seine Quellen im Sinne des Themas und der Problemstellung: verknüpft seine Argumentation sinnvoll mit Nachweisen.	4	
5	... argumentiert sinnvoll und nachvollziehbar, bezieht beschreibende, deutende und wertende Aussagen schlüssig aufeinander, unterscheidet gewissenhaft zwischen Faktendarstellung, fachwissenschaftlichen Positionen und der eigenen Meinung.	4	
6	... formuliert unter Beachtung der Fachsprache sachlich, präzise und begrifflich differenziert	8	
7	... kommt zu gehaltvollen Erarbeitungen sowie Teil- und Gesamtergebnissen. [Kommentar dazu: siehe Korrekturbemerkungen in der Arbeit und beiliegendes Gutachten]	20	
	Summe	50	

	Summe der formalen und inhaltlichen Leistung	100	
	Note		