Schule gemeinsam erleben

STÄDTISCHES GYMNASIUM ERWITTE

Schulinternes Curriculum

Sekundarstufen I und II

Informatik



Quelle: pexels.com

Fachschaft Informatik
Entwurfsstand: Erwitte, Juni 2025

SGE . Glasmerweg 12 . 59597 Erwitte . Fon 02943|2688 . Fax 02943|486176 . info@gymnasium-erwitte.de . **www.gymnasium-erwitte.de**

Inhalt

1 Aufgaben und Ziele des Faches 1
1.1 Durchgängiges Sprachbildungskonzept des Faches Informatik 3
2 Rahmenbedingungen der fachlichen Arbeit 7
2.1 Schulprogrammatische Leitlinien des Faches Informatik 7
2.2 Unterrichtsstruktur im Fach Informatik 7
2.3 Unterrichtende im Fach Informatik 7
2.4 Unterrichtsbedingungen 7
3 Entscheidungen zum Unterricht 8
3.1 Übersichtsraster und konkretisierte Unterrichtsvorhaben in der Klassenstufe 6 9
3.2 Übersichtsraster Unterrichtsvorhaben in der Einführungsphase 12
3.3 Konkretisierte Unterrichtsvorhaben 14
3.3.1 UV EF I – Was macht Informatik? - Einführung in die Inhaltsfelder der Informatik 14
3.3.2 UV EF II – Grundlagen der objektorientierten Analyse, Modellierung und Implementierung 15
3.3.3 UV EF III – Algorithmische Grundstrukturen in Java 18
3.3.4 UV EF IV – Das ist die digitale Welt! – Einführung in die Grundlagen, Anwendungsgebiete und Verarbeitung binärer Codierung 19
3.3.5 UV EF V – Modellierung und Implementierung von Klassen- und Objektbeziehungen anhand lebensnaher Anforderungsbespiele 21
3.3.6 UV EF VI – Such- und Sortieralgorithmen 24
3.3.7 UV EF VII – Leben in der digitalen Welt – Immer mehr Möglichkeiten und immer mehr Gefahren!? 26
3.4 Übersichtsraster Unterrichtsvorhaben in der Qualifikationsphase 28
3.5 Konkretisierte Unterrichtsvorhaben 30
3.5.1 UV Q-Phase I – Wiederholung und Vertiefung der objektorientierten Modellierung 30
3.5.2 UV Q-Phase II – Organisation und Verarbeitung von Daten I – Modellierung und Implementierung 32
3.5.3 UV Q-Phase III – Algorithmen zum Suchen und Sortieren auf linearen Datenstrukturen 35
3.5.4 UV Q-Phase IV – Automaten und formale Sprachen 37

3.5.5 UV Q-Phase V – Organisation und Verarbeitung von Daten II – Modellierung und Implementierung von Anwendungen mit dynamischen nicht-linearen Datenstrukturen 39
3.5.6 UV Q-Phase VI – Aufbau von und Kommunikation in Netzwerken 42
3.5.7 UV Q-Phase VII – Nutzung und Modellierung von relationalen Datenbanken in Anwendungskontexten 44
3.6 Überfachliche Grundsätze 46
3.6.1 Fachliche Grundsätze 46
3.6.2 Anmerkung zu beruflichen Handlungskompetenzen im Sinne der Studien- und Berufsorientierung: 47
3.6.3 Fachterminologie 47
3.7 Grundsätze der Leistungsbewertung und Leistungsrückmeldung 48
3.8 Absprachen bezüglich Hausaufgaben 49
3.9 Lehr- und Lernmittel 50
4 Entscheidungen zu fach- und unterrichtsübergreifenden Fragen 50
5 Qualitätssicherung und Evaluation 50

1 Aufgaben und Ziele des Faches

Gegenstand der Fächer im mathematisch-naturwissenschaftlich-technischen Aufgabenfeld (III) sind die empirisch erfassbare, die in formalen Strukturen beschreibbare und die durch Technik gestaltbare Wirklichkeit sowie die Verfahrens- und Erkenntnisweisen, die ihrer Erschließung und Gestaltung dienen.

Innerhalb der von allen Fächern zu erfüllenden Querschnittsaufgaben tragen insbesondere auch die Fächer des mathematisch-naturwissenschaftlich-technischen Aufgabenfeldes im Rahmen der Entwicklung von Gestaltungskompetenz zur kritischen Reflexion geschlechter- und kulturstereotyper Zuordnungen, zur Werteerziehung, zur Empathie und Solidarität, zum Aufbau sozialer Verantwortung, zur Gestaltung einer demokratischen Gesellschaft, zur Sicherung der natürlichen Lebensgrundlagen, auch für kommende Generationen im Sinne einer nachhaltigen Entwicklung, und zur kulturellen Mitgestaltung bei. Darüber hinaus leisten sie einen Beitrag zur interkulturellen Verständigung, zur interdisziplinären Verknüpfung von Kompetenzen, auch mit gesellschaftswissenschaftlichen und sprachlich-literarisch-künstlerischen Feldern, sowie zur Vorbereitung auf Ausbildung, Studium, Arbeit und Beruf.

Wie kaum eine andere Fachdisziplin durchdringt die Informatik mit den von ihr entwickelten Systemen für jedermann wahrnehmbar nahezu alle Bereiche von Wirtschaft, Gesellschaft, Arbeit und Freizeit. Sie besitzt einen großen Anteil am Entwicklungsstand unserer technisierten und globalisierten Welt. Prozessorgesteuerte Geräte, Softwareprodukte und durch deren Einsatz bestimmte Verfahrensweisen und Prozesse beeinflussen und verändern unser Leben mit hoher Dynamik. Die Informatik stellt Prinzipien und Methoden zur Erforschung komplexer Phänomene und für die Entwicklung komplexer Systeme bereit, die zahlreiche andere Fachdisziplinen aufgreifen und adaptieren. Daher ist die Informatik in hohem Maße interdisziplinär ausgerichtet. Die Auseinandersetzung mit Themen und Methoden der Informatik in der Schule dient somit der Lebensvorbereitung und Orientierung in einer von der Informationstechnologie geprägten Welt.

Der Informatikunterricht der gymnasialen Oberstufe geht deutlich über eine Grundbildung im Bereich der Informations- und Kommunikationstechnologien in der Sekundarstufe I hinaus. Die Schülerinnen und Schüler erwerben über rezeptive Medienanwendungen und die interaktive Nutzung von Medienangeboten in den Fächern hinaus Fähigkeiten zur kritischen und verantwortungsvollen Analyse, Modellierung und Implementierung komplexer Informatiksysteme. Dabei konzentriert sich der Unterricht stets auf fundamentale und zeitbeständige informatische Ideen, Konzepte und Methoden und schließt auch die Auseinandersetzung mit Fragen einer menschengerechten Gestaltung und der Sicherheit von Systemen sowie der Folgen und Wirkungen des Einsatzes von Informatiksystemen ein. Schülerinnen und Schüler werden so befähigt und motiviert, auch zukünftige Entwicklungen zu nutzen, zu verstehen, hinsichtlich ihrer Wirkungen zu beurteilen und sich aktiv an der Fortentwicklung zu beteiligen.

Ausgangspunkt im Informatikunterricht ist häufig ein Problem mit lebensweltlichem Bezug. Schülerinnen und Schüler erwerben und erweitern in der aktiven Auseinandersetzung mit komplexen Problemstellungen Kompetenzen, die sie zum selbstständigen informatischen Problemlösen befähigen. Mit der Aneignung von Strategien und Techniken zur strukturierten Zerlegung im Problemlöseprozess, zur Algorithmisierung von Abläufen sowie zur formalsprachlichen, grafischen oder symbolischen Beschreibung von Sachverhalten und Zusammenhängen erwerben die Lernenden Kompetenzen zur Bewältigung von Komplexität. Die Konstruktion eines abstrakten Modells zu einer anwendungsbezogenen Problemstellung fördert das Abstraktionsvermögen sowie kreatives und strukturelles Denken. Die Umsetzung eines informatischen Modells in ein lauffähiges Informatiksystem hat für Schülerinnen und Schüler nicht nur einen hohen Motivationswert, sondern ermöglicht ihnen auch die eigenständige Überprüfung der Angemessenheit und Wirkung des Modells im Rückbezug auf die Problemstellung. Im Unterricht lassen sich umfangreiche Informatiksysteme nur in arbeitsteiliger projektorientierter Zusammenarbeit im Team erstellen. Solche Projekte können nur gelingen, wenn die gemeinsame Arbeit strukturiert geplant und organisiert wird. Insgesamt leistet das Fach Informatik in der gymnasialen Oberstufe damit einen wichtigen Beitrag zu einer erweiterten Allgemeinbildung und allgemeinen Studierfähigkeit der Schülerinnen und Schüler.

Die inhaltliche und methodische Gestaltung des Unterrichts ist entscheidend dafür, dass Schülerinnen und Schüler die ausgewiesenen Kompetenzen erwerben können. Informatikunterricht erfordert in hohem Maße die Anwendung schüleraktivierender Methoden, die selbstständiges Lernen ermöglichen und individuelle Förderung begünstigen. Unterschiedliche, auch geschlechtsspezifische Herangehensweisen, Interessen, Vorerfahrungen und fachspezifische Kenntnisse sind angemessen zu berücksichtigen.

Ausgangspunkt des Informatikunterrichts der in der Klasse 6 einsetzt, sind Fragestellungen mit lebensweltlichem Bezug. In der aktiven und altersgemäßen Auseinandersetzung mit diesen Fragen erwerben und erweitern Schülerinnen und Schüler Kompetenzen zur Lösung informatischer Probleme. Die Auseinandersetzung mit Informatiksystemen hat für die Lernenden einen hohen Motivationswert, da Informatiksysteme eine unmittelbare Rückmeldung der Implementationen hinsichtlich Korrektheit und Angemessenheit ermöglichen.

Im Informatikunterricht der Klasse 6 werden in hohem Maße schüleraktivierende Methoden eingesetzt, die selbstständiges Lernen ermöglichen und individuelle Förderung begünstigen. Unterschiedliche, auch durch Geschlechtersozialisation geprägte Herangehensweisen, Interessen und Kenntnisse werden, auch durch Maßnahmen zum individuellen Fördern und Fordern, angemessen berücksichtigt.

In der Klasse 6 soll eine informatorische Grundbildung geschehen, weshalb der Fokus hier neben den inhaltlichen Schwerpunkten auch auf der grundlegenden „Bedienung“ von Informatiksystemen liegt aber auch auf der Verwendung von lebensweltnaher Anwendungen, wie Word oder Powerpoint oder dem Durchführen einer einfachen Internetrecherche (Siehe hierzu auch Medienkonzept).

In der Einführungsphase werden ausgehend von einfachen Fragestellungen und unter Anleitung der Lehrperson zunächst einzelne Stufen eines Problemlösungsprozesses durchlaufen. Am Ende der Qualifikationsphase sollen die Schülerinnen und Schüler dann in der Lage sein, Lösungsansätze in Form von lauffähigen Informatiksystemen weitgehend selbstständig zu entwickeln.

Der Grundkurs führt unter dem Aspekt einer fundamentalen wissenschaftspropädeutischen Ausbildung in basale Sachverhalte, Probleme sowie Zusammenhänge des Faches ein und verdeutlicht die Differenz zwischen Alltagswissen und wissenschaftlich begründetem Wissen. Er fördert den Kompetenzerwerb im Bereich der wesentlichen Arbeits- und Entwicklungsmethoden der Informatik und ermöglicht das Erkennen fachlicher Zusammenhänge exemplarisch anhand geeigneter Beispiele.

Der Leistungskurs widmet sich mit Blick auf eine exemplarisch erweiterte wissenschaftspropädeutische Ausbildung einem vertieften Kompetenzerwerb hinsichtlich der Breite, der Komplexität und des Reichtums an Aspekten der Inhaltsfelder. Er zielt auf einen systematischen Überblick über wesentliche Inhalte, eine vertiefte Beherrschung der informatischen Methoden sowie die selbstständige Umsetzung, Übertragung und Reflexion. Ein kompletter Problemlöseprozess kann in einem Leistungskurs häufiger vollständig durchlaufen werden, während dies in einem Grundkurs oft nur exemplarisch erwartet werden kann.

Die Progression der Anforderungen zwischen Einführungs- und Qualifikationsphase sowie Grund- und Leistungskursen resultiert nicht in erster Linie aus den Prozessen selbst. Vielmehr leitet sie sich aus der zunehmenden Komplexität der fachlichen Inhalte sowie der unterrichtlichen Kontexte, Projektvorhaben und Beispiele ab, die den Schülerinnen und Schülern den Kompetenzerwerb ermöglichen.

1.1 Durchgängiges Sprachbildungskonzept des Faches Informatik

Allgemeine Vorbemerkungen
Sprache gilt als die entscheidende Schlüsselkompetenz für eine gelingende Bildungsbiographie und die gesellschaftliche Teilhabe. Das Städtische Gymnasium Erwitte macht es sich durch ein durchgängiges Sprachbildungskonzept, welches in allen Fachcurricula der Schule verankert ist, zur selbstverständlichen Aufgabe allen Schüler/innen und Schülern unabhängig von ihren individuellen Grundvoraussetzungen dazu zu verhelfen, die Unterschiede zwischen Register und Struktur der Alltagssprache und dem, was bildungssprachlich verlangt ist, beherrschen zu lernen.

Sprachförderung, Sprachbildung und sprachsensibler Unterricht sind am SGE ein gemeinsames Entwicklungsvorhaben aller Fächer und der ganzen Schule. Unser Ziel ist es, durch die Verankerung konkreter Vorstellungen (u.a. hinsichtlich der geforderten Zieltextformate) und das Treffen von verbindlichen Vereinbarungen eine durchgängige Sprachentwicklung in den einzelnen Fächern sowie einheitliche fächerübergreifende Standards zu entwickeln.

Als Grundlage der Erarbeitung wird der Referenzrahmen für Schulqualität herangezogen und ein Bezug zu den FörMig-Qualitätsmerkmalen sowie zur Qualtätsanalyse NRW hergestellt. Die Sicherung erfolgt im Schulprogramm[1] sowie in den einzelnen schulinternen Fachcurricula und bildet somit eine Basis des gemeinsamen kollegialen Handelns. Auf diese Weise kann eine gleichsinnige schulische Umsetzung gewährleistet werden, von der alle Schülerinnen und Schüler der Schule gleichsam und unabhängig von individuellen soziobiografischen Faktoren profitieren.

Konkrete Zielsetzungen und verbindliche Unterrichtsvereinbahrungen
Mit Bezug zum durchgängigen Sprachbildungskonzept des Schule (vgl. Absätze 4.-4.6) ergeben sich für das zugrundeliegende Unterrichtsfach folgende konkrete Unterrichtsvereinbarungen hinsichtlich sprachbildender Lerngelegenheiten in den unten dargelegten schuleigenen Unterrichtsvorhaben:

• **Sprache und Verhalten von Lehr- und pädagogischen Fachkräften** (Absatz 4.2) Die Fachlehrehrkraft agiert sprachsensibel als bewusstes Sprachvorbild für die Lernenden und verpflichtet sich...

- als Sprachvorbild zu agieren auf geschlechtergerechte und diskriminierungsfreie Sprache zu achten.
- Bewusstsein für unterschiedliche Sprachregister schaffen.
- eine bewusste Anwendung von Bildungs- und Alltagssprache zu fördern.
- In Lehr- und Lernprozessen eine fach- und entwicklungsangemessene Sprache (Artikulation, Intonation, Sprechgeschwindigkeit, Satzbau, Wortschatz, fachrelevante Begriffe und Redemittel) zu verwenden.
- sprachliche Anteile des Lernens für SuS in verständlicher Weise transparent zu machen.
- sich schülerorientiert auf vorhandene Sprachkompetenzen der SuS einzustellen.
- zum Gebrauch bildungssprachlicher Muster anzuregen.
- lebensweltliche Mehrsprachigkeit und Heterogenität zu wertschätzen und als Ressource, zu nutzen indem sie die Möglichkeiten eröffnet, mehrsprachliche Potentiale der SuS in den Unterricht und schulische Handlungsfelder einzubringen.

• **Konkrete Maßnahmen und Methoden** (Absatz 4.2.2) Der Unterricht bietet vielfältige Möglichkeiten zur sprachlichen Unterstützung des fachlichen Lernens, indem...

- die Sprachstände der Schülerinnen und Schüler bei der Planung und Gestaltung der unterrichtlichen Prozesse berücksichtigt werden.
- Sprachbewusstheit geschaffen wird und unterschiedliche Funktionen und Register von Sprache sowie die Kommunikation über digitale Medien thematisiert werden.

[1] Vgl. dazu die durch die Bezirksregierung Arnsberg vorgesehenen Konzeptteile A und B.

- die Lernsituation ein komplexes Sprachhandeln herausfordert und SuS umfassend Sprech- und Schreibgelegenheiten zur Erprobung ihrer Sprachfähigkeiten erhalten.
- eine Visualisierung des sprachlichen Anteils in der Lernumgebung besteht.
- die Möglichkeit für SuS, komplexere Äußerungen zu formulieren gegeben wird.
- Sprachgerüste („Scaffolds") mit Angebot an sprachlich relevanten Mitteln und/oder Impulsen zur kognitiven Aktivierung bereitgestellt werden.
- Anregung zum Gebrauch bildungssprachlicher Muster gegeben werden.
- Raum für sprachliche Aktivität der SuS vorhanden ist.
- Verstehenskontrollen und Reformulierungsaufgaben stattfinden.
- Lehrkräfte auf sprachliche Präzision der SuS achten.
- Lehrkräfte und SuS die Ergebnisse der sprachlichen Bildung überprüfen und bewerten. Die Lehrkräfte nutzen dafür u.a. die Ergebnisse von Lernstanderfassungen für den Unterricht, Ergebnisse der Lernstands-, Lernentwicklungs- und Lernerfolgsüberprüfungen. Sie sind kontinuierlich Anlass, die Gestaltung der Lehr- und Lernprozesse zu überprüfen und ggf. zu modifizieren.

• **Die im Unterricht eingesetzten Materialien sind sprachsensibel aufbereitet** (Absätze 4.3, 4.6)

- Es erfolgt ein Einsatz von Texten und Materialien, die den Spracherwerbsstand der SuS berücksichtigen und den Spracherwerb unterstützen.
- Die Materialien verweisen auf Unterstützungsmöglichkeiten für das sprachliche Lernen (z. B. Wörterlisten, Scaffolds).
- Die Materialien sind ggf. für unterschiedliche Sprachniveaus differenziert (z. B. durch komplexere Satzstrukturen, unterschiedlich gegliederte Texte).
- Sprachliche Hürden in Aufgabenstellungen und Unterrichtsmaterialien werden regelmäßig reflektiert.

Das folgende Schaubild zeigt zusammenfassend die verschiedenen Bereiche, die bei unserem sprachsensiblen Fachunterricht Berücksichtigung finden.



Das Sprachbildungskonzept des Städtischen Gymnasiums Erwitte unterliegt einem ständigen Qualitätssicherungskreislauf, in dem aktuelle Entwicklungsvorhaben sowohl fachschaftsarbeitsbezogen als auch systemisch erfasst, beschrieben, geplant, umgesetzt und schließlich evaluiert und bewertet werden.

Weitere Informationen können dem durchgängigen Sprachbildungskonzept auf der Homepage entnommen werden.

2 Rahmenbedingungen der fachlichen Arbeit

2.1 Schulprogrammatische Leitlinien des Faches Informatik

Das Fach Informatik soll einen wesentlichen Beitrag leisten hinsichtlich erzieherischer, naturwissenschaftlicher und medialer Aufgaben und solcher, die mit einem Informatiksystem bearbeitet werden. Der einzelne Mensch mit seinen Stärken, Begabungen, Eigenarten und Schwächen steht dabei genauso im Fokus der pädagogischen Bemühungen, wie die Förderung seiner kooperativen und sozialen Fähigkeiten.

2.2 Unterrichtsstruktur im Fach Informatik

Die Schülerinnen und Schüler erhalten im Sinne der Vorgaben der APOSI / APOSII (APO-GOS) für das Fach Informatik in der Sekundarstufen I und II Unterricht in den / der:

- Klassenstufe 6 2 Stunden durchgehend
- EF 3 Stunden durchgehend
- Q1 3 (GK) bzw. 5 (LK) Stunden durchgehend
- Q2 3 (GK) bzw. 5 (LK) Stunden durchgehend

2.3 Unterrichtende im Fach Informatik

Die Fachschaft Informatik besteht zurzeit (Schuljahr 2024/25) aus drei Lehrern mit der Fakultas Informatik für die Sek II und einer Lehrerin mit der Fakultas für die Klasse 5 und 6.

Die oder der Fachvorsitzende ist für die organisatorischen Belange des Faches in der Schule zuständig und wird von ihren bzw. seinen Kolleginnen und Kollegen dabei unterstützt.

Die Fachschaft wird gelegentlich von Aushilfskräften, Referendaren und Praktikanten unterstützt.

2.4 Unterrichtsbedingungen

Von der Fachschaft Informatik werden der Inforamtikraum und das SLZ genutzt. In beiden Räumen gibt es eine Ausstattung an PCs, die von den SuS genutzt werden können. Auf den PCs sind verschiedene Programme installiert, die von den SuS genutzt werden.

Beide Räume haben außerdem eine Leinwand mit Beamer und Apple TV bzw. eine Smart-Board-Tafel.

Die Fachschaft Informatik verfügt noch über weitere Lehrmittel, wie u. a. den Mindstorms Robotern von Lego.

3 Entscheidungen zum Unterricht

Im Folgenden sind die konkretisierten Unterrichtsvorhaben der Jahrgangsstufen 6, EF und der Q-Phase in tabellarischer Form ausführlich dargestellt. Neben den allgemeinen Angaben sind in ihnen Kompetenzerwartungen, didaktische und methodische Festlegungen und auch individuelle Spielräume angegeben.

Bei den Bereichen „Fachmethodische Arbeitsform“ und „Lernmittel“ sind jeweils nur die zentralen Formen und Medien angegeben, die für das Unterrichtsvorhaben unabdingbar sind. Das Hinzuziehen anderer Formen und Medien ist grundsätzlich aber möglich.

Die Dauern der Unterrichtsvorhaben sind etwaige Werte, die im Ermessen der Lehrkraft überstiegen und unterschritten werden können. Die Lehrkraft trägt Verantwortung dafür, dass die notierten Kompetenzen unabhängig von der Länge des Unterrichtsvorhabens die vorgesehene Anbahnung in den Lerngruppen finden. Selbiges gilt auch für die Reihenfolge der Unterrichtsvorhaben.

Neben den hier dargelegten Unterrichtsvorhaben ist auch das Durchführen von Projekten und Kleinsequenzen zu anderen Themen möglich. Auch das Verschmelzen mehrerer Unterrichtsvorhaben ist akzeptabel, wenn die Anbahnung der aufgeführten Kompetenzen und die Einhaltung der anderen Aspekte gewahrt bleibt.

Selbstverständlich orientiert sich der Unterricht stehts an den aktuellen Länder- und Abiturvorgaben, sodass geforderte Gegenstände natürlich bei der Kompetenzanbahnung zum Einsatz kommen.

Die gelb hinterlegten Textstellen beziehen sich auf den Medienkompetenzrahmen des Landes NRW und sind auch ein Beitrag für das Medienkonzept der Schule. Diese sind hier beispielhaft markiert, da das Fach Informatik mit seinem Grundwerkzeug, dem PC, von sich aus einen großen Beitrag für Medienbildung leisten kann.

Die rot hinterlegten Textstellen sind Beiträge für die Berufs- und Studienorientierung. Neben Berufsfeldern, die so angedeutet werden (vor allem im Bereich der Informatik und der SOftwareentiwcklung), leistet das Fach Informatik einen Beitrag zu Handlungskompetenzen aus der späteren Berufswelt. Diese sind unten unter dem Punkt „Anmerkungen zur Studien und Berufsorientierung“ weiter aufgeschlüsselt. Sie beziehen sich auf das StuBO-Curriculum.

Verweise zur Sprachbildung sind grün hinterlegt (s.o. 2.2.).

3.1 Übersichtsraster und konkretisierte Unterrichtsvorhaben in der Klassenstufe 6

<table>
<tr><th colspan="2">Klassenstufe 6</th></tr>
<tr>
<td><u>Unterrichtsvorhaben 6-I</u>

Thema:
Einstieg in die Informatik

Zentrale Kompetenzen:
- Erläutern des Datenbegriffs anhand von Beispielen aus der eigenen Erfahrungswelt
- Benennen von Beispielen für (vernetzte) Informatiksysteme aus der eigenen Erfahrungswelt und Beschreibung in Bezug auf ihre Funktionen
- Beschreiben des EVA Prinzips als grundlegendes Prinzip der Datenverarbeitung
- Einsetzen von Informatiksystemen zur Verarbeitung von Daten, zur Kommunikation und Kooperation und dem Erläutern der Dateiverwaltung
- Beschreibung und Bewertung der Bedeutung und Auswirkung von Informatiksystemen für die eigene Lebenswelt

Inhaltsfelder:
- Informationen und Daten
- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte:
- Fachgebiete, Inhalte und wesentliche Fragen der Informatik
- Informatiksysteme, Hardware und Von-Neumann-PC
- Software und Maschinensprache (Binär)
- EVA-Prinzip

Methodische Hinweise:
z. B. Einsatz von ABs, Lernhomepages, Auseinanderbauen eines echten PCs oder Smartphones

Leistungsüberprüfung:
LÜ z. B. im Unterrichtsgespräch, anhand schriftlicher Aufgaben und Leistungsüberprüfungen und praktischer Aufgaben

Zeitbedarf: etwa 10 Stunden</td>
<td><u>Unterrichtsvorhaben 6-II</u>

Thema:
Verschlüsselungsverfahren

Zentrale Kompetenzen:
- Erläutern von und vergleichen von verschiedenen Verschlüsselungsverfahren in Bezug auf ihre Funktion und Sicherheit
- Einsetzen von Informatiksystemen zur Verarbeitung von Daten, zur Kommunikation und Kooperation und zur Erstellung eines Medienproduktes
- Erläutern an Beispielen aus der Kryptographie Auswirkungen des Einsatzes von Informatiksystemen und bewerten deren Nutzen und Risiken

Inhaltsfelder:
- Informationen und Daten
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte:
- Analyse, Vergleich und Bewertung verschiedener Verschlüsselungs- und Kommunikationsverfahren (Codierung, Steganographie, Transposition und Substitution)

Methodische Hinweise:
z. B. Einsatz von ABs, Lernhomepages, Erstellen einer Powerpointpräsentation zu einer ausgewählten Verschlüsselungstechnik

Leistungsüberprüfung:
LÜ z. B. im Unterrichtsgespräch, anhand schriftlicher Aufgaben und Leistungsüberprüfungen und praktischer Aufgaben, Powerpointpräsentation

Zeitbedarf: etwa 12 Stunden</td>
</tr>
</table>

Unterrichtsvorhaben 6-III

Thema:
Algorithmen und Programmieren

Zentrale Kompetenzen:
- Formulierung von Handlungsvorschriften zu Alltagsabläufen und Überführen der selben in ein PAP oder Struktogramm
- Ausführen von Algorithmen und Identifizeirung zentraler Strukturmerkmale (wie Sequenz, Verzweigung, Schleife, etc.)
- Implementierung eines eigenen Algorithmus in einer visuellen Programmiersprache (mit z. B. Scratch, CODE, etc.) unter Berücksichtigung des Prinzips der Modularisierung
- Überprüfen und bewerten eines Programmes in Bezug auf seine Funktionalität durch Testen und Analyse

Inhaltsfelder:
- Informationen und Daten
- Algorithmen
- Informatiksysteme

Inhaltliche Schwerpunkte:
- Algorithmen als Handlungsvorschriften im Alltag und in einer Programmierumgebung (Schriftlich und Graphisch)
- Schrittweise Ausführung von Handlungsvorschriften
- Grundtstrukturen: u. a. Befehl, Sequenz, bedingte Anweisung, Schleifen und Ereignisse
- Erstellen und Testen eigener Programme
- Analysieren, beschreiben und erklären einfacher Programme

Methodische Hinweise:
z. B. Einsatz von ABs, Lernhomepages, Programmierumgebungen (z. B. Scratch, Programmieren mit Star Wars oder Minecraft über CODE, Bundeswettbewerb Informatik, etc.)

Leistungsüberprüfung:
LÜ z. B. im Unterrichtsgespräch, anhand schriftlicher Aufgaben und Leistungsüberprüfungen und praktischer Aufgaben (Hier: Implementationen)

Zeitbedarf: etwa 24 Stunden
(Es bietet sich an, dieses UV auf Grund der Länge und Inhaltsfülle aufzuteilen).

Unterrichtsvorhaben 6-IV

Thema:
Daten, Information und ihre Repräsentation im World Wide Web

Zentrale Kompetenzen:
- Erläutern des Datenbegriffes und des Zusammenhangs von Informationen und Daten anhand von Beispielen aus der Lebenswelt
- Darstellen und Interpretation von Daten in vorgegebenen Kontexten
- Erläutern Einheiten von Datenmengen und vergleichen diese
- Beschreibung und Bewerten von Datenverarbeitung unter der Gesichtspunkt des Datenschutzes

Inhaltsfelder:
- Informationen und Daten
- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte:
- Repräsentation von Daten grundlegender Datenytpen (Zahlen, Zeichen und Zeichenketten (ASCII) in z.B. Binär
- Speicherbedarf von Daten in Bits und Bytes
- Datenschutz, personenbezogene Daten und Besipiele für deren Nutzung und Missbrauch
- Datennutzung im Internet (Datenschutzaspekte z.B. im Zusammenhang mit der Suchmaschine Google oder stattlichen Kontexten am Bsp. China)
- Abhängigkeit und Sucht (z. B. Medien- oder Spielesucht)

Methodische Hinweise:
z. B. Einsatz von ABs, Analyse von Homepages und Apps, Lernfilmen und Dokus

Leistungsüberprüfung:
LÜ z. B. im Unterrichtsgespräch, anhand schriftlicher Aufgaben und Leistungsüberprüfungen und praktischer Aufgaben, Powerpointpräsentation

Zeitbedarf: etwa 10 Stunden

Unterrichtsvorhaben 6-VI **Thema:** *Maschinelles Lernen* **Zentrale Kompetenzen**: - Benennung von Anwendungsbeispielen von KI in der Lebenswelt - Beschreiben und Erklären der Entscheidungsbäume und neuronaler Netze als Grundprinzipien der Funktion von KI - Darstellen von Abläufen in Form von Automaten **Inhaltsfelder**: - Automaten und künstliche Intelligenz - Informatiksysteme - Informatik, Mensch und Gesellschaft **Inhaltliche Schwerpunkte**: - Grundzüge maschinellen Lernens, Grundfunktionen einer KI - Analyse von KI in Alltagsbeispielen (ChatGPT aber auch Spotify und Social Media) - Entscheidungsbäume und Neuronale Netze - Beschreibung und Bewerten von Datenverarbeitung durch KI unter der Gesichtspunkt des Datenschutzes - Darstellung von Prozessen mit Hilfe von Automaten **Methodische Hinweise:** z. B. Einsatz von ABs, Lernhomepages, Erarbeitung mit Hilfe einer KI, Analyse von Apps und Anwendungen **Leistungsüberprüfung:** LÜ z. B. im Unterrichtsgespräch, anhand schriftlicher Aufgaben und Leistungsüberprüfungen und praktischer Aufgaben **Zeitbedarf**: etwa 8 Stunden	**Summe Klasse 6:** **64 Stunden**

3.2 Übersichtsraster Unterrichtsvorhaben in der Einführungsphase

<table>
<tr><th colspan="2">Einführungsphase</th></tr>
<tr>
<td><u>Unterrichtsvorhaben EF-I</u>

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

Zentrale Kompetenzen:
- Kommunizieren und Kooperieren
- Darstellen und Interpretieren
- Argumentieren

Inhaltsfelder:
- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte:
- Einsatz, Nutzung und Aufbau von Informatiksystemen
- Wirkung der Automatisierung

Zeitbedarf: etwa 6 Stunden</td>
<td><u>Unterrichtsvorhaben EF-II</u>

Thema:
Grundlagen der objektorientierten Analyse, Modellierung und Implementierung

Zentrale Kompetenzen:
- Modellieren
- Implementieren
- Darstellen und Interpretieren
- Kommunizieren und Kooperieren

Inhaltsfelder:
- Daten und ihre Strukturierung
- Formale Sprachen und Automaten

Inhaltliche Schwerpunkte:
- Objekte und Klassen
- Syntax und Semantik einer Programmiersprache

Zeitbedarf: etwa 8 Stunden</td>
</tr>
<tr>
<td><u>Unterrichtsvorhaben EF-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: etwa 18 Stunden</td>
<td><u>Unterrichtsvorhaben EF-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: etwa 8 Stunden</td>
</tr>
</table>

<table>
<tr>
<td><u>Unterrichtsvorhaben EF-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: etwa 18 Stunden</td>
<td><u>Unterrichtsvorhaben EF-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: etwa 9 Stunden</td>
</tr>
<tr>
<td><u>Unterrichtsvorhaben EF-VII</u>

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
- KI, Neuronale Netze und Entscheidungsbäume

Zeitbedarf: etwa 10 Stunden</td>
<td>Summe Einführungsphase:
77 Stunden</td>
</tr>
</table>

3.3 Konkretisierte Unterrichtsvorhaben

3.3.1 UV EF I – Was macht Informatik? - Einführung in die Inhaltsfelder der Informatik

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

Leitfragen: Was macht Informatik? Welche fundamentalen Konzepte müssen Informatikerinnen und Informatiker 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 und Schüler 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 Algorithmik, die Eigenheit formaler Sprachen, die Kommunikationsfähigkeit von Informatiksystemen und die positiven und negativen Auswirkungen auf Mensch und Gesellschaft thematisiert. Das am Beispiel Navigationsgerät erworbene Wissen kann auf weitere den Schülerinnen und Schülern 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: 6 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	Die Schülerinnen und Schüler - 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 und Landkarten an.

Gesellschaft		
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 Datenschutzbe-stimmungen der Schule

3.3.2 UV EF II – Grundlagen der objektorientierten Analyse, Modellierung und Implementierung

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 Objektorientierte 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 und Schüler 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 und Schülern in Teilen analysiert und entsprechende Objekte anhand einfacher Problemstellungen erprobt. Die Lernenden implementieren und testen einfache Programme. Die Greenfoot-Umgebung ermöglicht es, Beziehungen zwischen Klassen zu einem späteren Zeitpunkt (Kapitel 5) zu thematisieren. So kann der Fokus hier auf Grundlagen wie der Unterscheidung zwischen Klasse und Objekt, Attribute, Methoden, Objektidentität 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: 8 Stunden

Sequenzierung des Unterrichtsvorhabens: siehe nächste Seite

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 und Schüler - 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 „Planetenerkundung“ 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

3.3.3 UV EF III – Algorithmische Grundstrukturen in Java

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 und Schüler die unterschiedlichen Einsatzmöglichkeiten des Variablenkonzepts anzuwenden.

Zeitbedarf: 18 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Algorithmen (a) Wiederholungen (While-Schleife, Zählschleife) (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 und Schüler - analysieren und erläutern einfache Algorithmen und Programme (A), - entwerfen einfache Algorithmen und stellen sie umgangssprachlich und grafisch dar (M), - ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen zu (M), - modifizieren einfache Algorithmen und Programme (I), - 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 (I),	***Kapitel 3 Algorithmen*** 3.1 Kontrollstrukturen 3.2 Wiederholungen 3.3 Zählschleifen 3.4 Bedingte Anweisungen 3.5 Logische Operationen 3.6 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 (c) Implementierung von Konstruktoren (d) Realisierung von Attributen		***Kapitel 4 Variablen und Methoden*** 4.1 Variablen 4.2 Methoden

	- interpretieren Fehlermeldungen und korrigieren den Quellcode (I).	

3.3.4 UV EF IV – Das ist die digitale Welt! – Einführung in die Grundlagen, Anwendungsgebiete und Verarbeitung binärer Codierung

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 weiter verarbeitet werden? Wie unterscheiden sich analoge Medien und Geräte 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 und Schüler 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 und Schüler das Binäre als 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 und Schüler - 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)	***Die digitale Welt 001 – Von analog zu digital***
2. Der Umgang mit binärer Codierung von Informationen		***Die digitale Welt 010 – Binäre Welt***

(a) Das binäre (und hexadezimale) Zahlensystem (b) Binäre Informationsspeicherung (c) Binäre Verschlüsselung (d) Implementation eines Binärumrechners	- nutzen das Internet zur Recherche, zum Datenaustausch und zur Kommunikation (K) - implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I)	
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		***Die digitale Welt 011 – Der Von-Neumann-Rechner***

3.3.5 UV EF V – Modellierung und Implementierung von Klassen- und Objektbeziehungen anhand lebensnaher Anforderungsbespiele

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 und Schüler 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 und Schüler 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: s. nächste Seite

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 und Schüler - 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 Vererbungs-beziehungen in Diagrammen grafisch dar (D), - dokumentieren Klassen durch Beschreibung der Funktionalität der Methoden (D)	***Kapitel 5 Klassenentwurf*** 5.1 Von der Realität zum Programm 5.2 Objekte 5.3 Klassen und Beziehungen entwerfen
2. Implementationsdiagramme als erster Schritt der Programmierung (a) Erweiterung des Entwurfsdiagramms um Konstruktoren und get- und set-Methoden (b) Festlegung von Datentypen in Java, sowie von Rückgaben und Parametern (c) Entwicklung von Klassendokumentationen (d) Erstellung von Sequenzdiagrammen als Vorbereitung für die Programmierung		***Kapitel 5 Klassenentwurf*** 5.4 Klassen und Beziehungen implementieren
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 5 Klassenentwurf*** 5.4 Klassen und Beziehungen implementieren
4. Vererbungsbeziehungen (a) Das Grundprinzip der Vererbung wird erarbeitet. (b) Die Vorteile der Vererbungsbeziehungen (c) Vererbung wird implementiert		***Kapitel 5 Klassenentwurf*** 5.5 Vererbung
5. Softwareprojekt (a) Analyse und Dekonstruktion		***Kapitel 7 Softwareprojekte***

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		7.1 Softwareentwicklung 7.2 Oberflächen

3.3.6 UV EF VI – Such- und Sortieralgorithmen

Thema: Such- und Sortieralgorithmen

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

Vorhabenbezogene Konkretisierung:
Dieses Unterrichtsvorhaben beschäftigt sich mit der Erarbeitung von Such- und Sortieralgorithmen. Der Schwerpunkt des Vorhabens liegt dabei auf den Algorithmen selbst und nicht auf deren Implementierung in einer Programmiersprache, auf die in diesem Vorhaben vollständig verzichtet werden soll.

Zunächst lernen die Schülerinnen und Schüler das Feld als eine erste Datensammlung kennen. Alle Algorithmen dieses Kapitels arbeiten auf einem Feld.
Die Schülerinnen und Schüler lernen zunächst Strategien des Suchens (lineare Suche, binäre Suche, Hashing) und dann des Sortierens (Selection Sort, Insertion Sort, Bubble Sort) kennen.

Daran anschließend lernen die Schülerinnen und Schüler 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.

Schließlich wird die Effizienz unterschiedlicher Sortierverfahren beurteilt.

Zeitbedarf: 9 Stunden

Sequenzierung des Unterrichtsvorhabens: s. nächste Seite

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 und Schüler - analysieren Such- und Sortieralgorithmen und wenden sie auf Beispiele an (D) - entwerfen einen weiteren Algorithmus zum Sortieren (M) - beurteilen die Effizienz von Algorithmen am Beispiel von Sortierverfahren hinsichtlich Zeit und Speicherplatzbedarf (A) - ordnen Attributen lineare Datenansammlungen zu (M)	***Kapitel 6 Suchen und Sortieren*** 6.1 Speichern mit Struktur – Arrays
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 6 Suchen und Sortieren*** Projekteinstieg 1: Suchen 6.2 Suchen mit System Lineare Suche Binäre Suche Hashing
3. Systematisierung von Algorithmen und Effizienzbetrachtungen (a) Formulierung (falls selbst gefunden) oder Erläuterung von mehreren Algorithmen im Pseudocode (b) Anwendung von Sortieralgorithmen auf verschiedene Beispiele (c) Bewertung von Algorithmen anhand der Anzahl der nötigen Vergleiche (d) Analyse eines weiteren Sortieralgorithmus (sofern nicht in (a) bereits geschehen)		***Kapitel 6 Suchen und Sortieren*** Projekteinstieg 2: Sortieren 6.3 Ordnung ist das halbe Leben!? – Sortieren Sortieren Selection Sort Insertion Sort Bubble Sort

3.3.7 UV EF VII – Leben in der digitalen Welt – Immer mehr Möglichkeiten und immer mehr Gefahren!?

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, u. a. KI 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. Hierbei wird auch und v. a. KI und ihre Funktionen unter die Lupe genommen.

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 und Schüler - bewerten anhand von Fallbeispielen die Auswirkungen des Einsatzes von Informatiksystemen (A), - erläutern wesentliche Grundlagen der Geschichte der digitalen Datenverarbeitung (A) - nutzen das Internet oder KI zur Recherche, zum Datenaustausch und zur Kommunikation (K)	***Die digitale Welt 100 – Von der Schrift zum Smartphone***
2. Die Informationsverarbeitung und ihre Möglichkeiten und Gefahren		***Die digitale Welt 100 – Das Leben in der digitalen Welt***

(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 daten-schutzrechtlichen Nachteile betrachtet.		

3.4 Übersichtsraster Unterrichtsvorhaben in der Qualifikationsphase

<table>
<tr><th colspan="2">Qualifikationsphase – Q1 -GK</th></tr>
<tr>
<td><u>Unterrichtsvorhaben Q1-I</u>

Thema: Wiederholung und Vertiefung der objektorientierten Modellierung

Zentrale Kompetenzen:
- Modellieren
- Darstellen und Interpretieren
- Implementieren
- Kommunizieren und Kooperieren

Inhaltsfelder:
- Daten und ihre Strukturierung
- Algorithmen
- Informatik, Mensch und Gesellschaft
- Informatiksysteme

Inhaltliche Schwerpunkte:
- Objekte und Klassen
- Wirkung der Automatisierung
- Nutzung von Informatiksystemen

Zeitbedarf: etwa 14 Stunden</td>
<td><u>Unterrichtsvorhaben Q1-II</u>

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

Zentrale Kompetenzen:
- Modellieren
- Implementieren
- Darstellen und Interpretieren
- Argumentieren

Inhaltsfelder:
- Daten und ihre Strukturierung
- Algorithmen
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte:
- Objekte und Klassen
- Syntax und Semantik einer Programmiersprache
- Analyse, Entwurf und Implementierung von Algorithmen
- Algorithmen in ausgewählten informatischen Kontexten
- Wirkungen der Automatisierung

Zeitbedarf: etwa 20 Stunden</td>
</tr>
<tr>
<td><u>Unterrichtsvorhaben Q1-III</u>

Thema: Algorithmen zum Suchen und Sortieren auf linearen Datenstrukturen

Zentrale Kompetenzen:
- Argumentieren
- Darstellen und Interpretieren
- Modellieren
- Implementieren

Inhaltsfelder:
- Algorithmen
- Formale Sprachen und Automaten
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte:
- Analyse, Entwurf und Implementierung von Algorithmen
- Algorithmen in ausgewählten informatischen Kontexten
- Syntax und Semantik einer Programmiersprache
- Wirkungen der Automatisierung

Zeitbedarf: etwa 16 Stunden</td>
<td><u>Unterrichtsvorhaben Q1-IV</u>

Thema: Automaten und formale Sprachen

Zentrale Kompetenzen:
- Argumentieren
- Darstellen und Interpretieren
- Modellieren
- Kommunizieren und Kooperieren

Inhaltsfelder:
- Formale Sprachen und Automaten
- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte:
- Syntax und Semantik einer Programmiersprache
- Endliche Automaten
- Grammatiken regulärer Sprachen
- Möglichkeiten und Grenzen von Automaten und formalen Sprachen
- Einzelrechner und Rechnernetzwerke
- Grenzen der Automatisierung

Zeitbedarf: etwa 24 Stunden</td>
</tr>
<tr><td colspan="2">Summe Qualifikationsphase 1: 74 Stunden</td></tr>
</table>

Qualifikationsphase – Q2 - GK

Unterrichtsvorhaben Q2-I

Thema: Organisation und Verarbeitung von Daten II – Modellierung und Implementierung von Anwendungen mit dynamischen nicht-linearen 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: etwa 20 Stunden

Unterrichtsvorhaben Q2-II

Thema: Aufbau von und 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: etwa 16 Stunden

Unterrichtsvorhaben Q2-III

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: etwa 20 Stunden

Summe Qualifikationsphase 2: 56 Stunden

3.5 Konkretisierte Unterrichtsvorhaben

3.5.1 UV Q-Phase I – Wiederholung und Vertiefung der objektorientierten Modellierung

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

Zeitbedarf: 14 Stunden

Sequenzierung des Unterrichtsvorhabens: s. nächste Seite

<table>
<tr><th>Unterrichtssequenzen</th><th>Zu entwickelnde Kompetenzen</th><th>Kapitel und Materialien</th></tr>
<tr><td>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</td><td rowspan="3">Die Schülerinnen und Schüler …
- 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).</td><td>Kapitel 1 Konzepte des objektorientierten Modellierens

1.1 Modellierung der Realität
1.2 Die Welt ist voller Objekte
Projekteinstieg: Klassenentwurf – step by step</td></tr>
<tr><td>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</td><td>1.3 Gut geplant – Klassenentwurf
1.4 Hierarchien machen´s einfacher – Vererbung</td></tr>
<tr><td>3. Mensch und Technik
a) Informatiker verändern die Welt
b) Automatisierung des Alltags durch Informatik</td><td>Die digitale Welt 001 – Mensch und Technik</td></tr>
</table>

3.5.2 UV Q-Phase II – Organisation und Verarbeitung von Daten I – Modellierung und Implementierung

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

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

Vorhabensbezogene 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: 20 Stunden

Sequenzierung des Unterrichtsvorhabens: s. nächste Seite

<table>
<tr><th>Unterrichtssequenzen</th><th>Zu entwickelnde Kompetenzen</th><th>Kapitel und Materialien</th></tr>
<tr><td>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</td><td rowspan="3">Die Schülerinnen und Schüler …
- 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).</td><td>Kapitel 2 Lineare Datenstrukturen
2.1 Anforderungen an eine Datenstruktur
2.2 Datenansammlungen fester Größe – Arrays</td></tr>
<tr><td>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 Queue; Erschließen der Standardoperationen
e) Modellierung und Implementierung einer Anwendung auf der Basis einer Anforderungsbeschreibung mit Objekten der Klasse Queue</td><td>2.3 Wer zuerst kommt ... – Schlangen</td></tr>
<tr><td>3. Die Datenstruktur Stapel
a) Erläuterung von Problemstellungen, die nach dem LIFO-Prinzip bearbeitet werden
b) Funktionalität der Klasse Stapel unter Verwendung der Klasse Stack, Erschließen der Standardoperationen
c) Modellierung und Implementierung einer Anwendung auf Basis einer Anforderungsbeschreibung mit Objekten der Klasse Stack
d) Modellierung und Implementierung einer Anwendung unter Verwendung verschiedener</td><td>2.4 Daten gut abgelegt – Stapel</td></tr>
</table>

Datenstrukturen (Objekte der Klassen *Queue*, *Stack* und *Array (Palindrom)*)		
4. Die Datenstruktur Liste a) Analyse der Möglichkeiten bisheriger Datenstrukturen zwecks Bestimmung notwendiger Funktionalitäten für komplexere Anwendungen (Abgrenzung zu *Stack*/*Queue*, zusätzliche Fähigkeiten der Klasse *List*) b) Erarbeitung der Funktionalität der Liste unter Verwendung der Klasse *List* c) Modellierung und Implementierung einer Anwendung mit Objekten der Klasse *List* d) Modellierung und Implementierung einer Anwendung unter Verwendung verschiedener Datenstrukturen (Stack, Queue, List)		2.5 Flexibel für alle Fälle – lineare Listen
5. Übungen und Vertiefungen zur Verwendung linearer und dynamischer Datenstrukturen anhand weiterer Problemstellungen **6. Datenschutz** a) Datenschutz als Grundrecht b) Das Datenschutzgesetz c) Datensammler		2.6 Prüfungsvorbereitung Projekteinstieg: Wartende Helden Mit dem Heldenspiel können alle im Kapitel behandelten Datenstrukturen erarbeitet werden. Das Spiel kann bis zu einem beliebigen Grad realisiert werden, sodass es sowohl als Einstieg als auch als ein umfassendes Projekt für lineare Datenstrukturen genutzt werden kann. Die digitale Welt 101 – Datenschutz

3.5.3 UV Q-Phase III – Algorithmen zum Suchen und Sortieren auf linearen Datenstrukturen

Thema: Algorithmen zum Suchen und Sortieren auf linearen Datenstrukturen

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

Vorhabenbezogene Konkretisierung:
Zunächst werden anhand eines Anwendungsbeispiels übergreifende Algorithmuseigenschaften (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 und Schülern analysiert und selbst entwickelt.
In der zweiten Unterrichtssequenz geht es um die Frage, wie Daten in linearen Strukturen (lineare Liste und Array) (wieder-)gefunden werden können. Die lineare Suche als iteratives und die binäre Suche als rekursives Verfahren werden veranschaulicht und implementiert. Die Bewertung der Algorithmen erfolgt, indem jeweils die Anzahl der Vergleichsoperationen und der Speicherbedarf ermittelt wird.
Möchte man Daten effizient in einer linearen Struktur wiederfinden, so rückt zwangsläufig die Frage nach einer Sortierstrategie in den Fokus. Es wird mindestens ein iteratives und ein rekursives Sortierverfahren erarbeitet und implementiert sowie ihre Effizienz bewertet.

Zeitbedarf: 20 Stunden

Sequenzierung des Unterrichtsvorhabens: s. nächste Seite

<table>
<tr><th>Unterrichtssequenzen</th><th>Zu entwickelnde Kompetenzen</th><th>Kapitel und Materialien</th></tr>
<tr><td>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</td><td rowspan="4">Die Schülerinnen und Schüler …
- 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 Sortierverfahren (I),
- beurteilen die Effizienz von Algorithmen unter Berücksichtigung des Speicherbedarfs und der Zahl der Operationen (A),
- beurteilen die syntaktische Korrektheit und die Funktionalität von Programmen (A),
- nutzen die Syntax und Semantik einer Programmiersprache bei der Implementierung und zur Analyse von Programmen (I),
- interpretieren Fehlermeldungen und korrigieren den Quellcode (I),
- 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).</td><td>Kapitel 3 Algorithmen
3.1 Ohne Algorithmen läuft nichts
3.2 Teile die Arbeit – rekursive Algorithmen</td></tr>
<tr><td>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</td><td>3.3 Suchen – iterativ und rekursiv
Projekteinstieg: Laufzeitanalyse experimentell</td></tr>
<tr><td>3. Sortieren auf Listen und Arrays
a) Entwicklung und Implementierung eines iterativen Sortierverfahrens für eine Liste (Sortieren durch Einfügen)
b) Entwicklung und Implementierung eines rekursiven Sortierverfahrens für eine Liste (Quicksort)
c) Untersuchung der beiden Verfahren bzgl. Laufzeit und Speicherplatzbedarf
d) Weitere Sortierverfahren auf Listen und Arrays (Sortieren durch Auswählen, Mergesort)</td><td>3.4 Sortieren – iterativ und rekursiv</td></tr>
<tr><td>4. Verantwortung der Informatik
a) Der Unterschied zwischen Anwender und Produzent von Informatiksystemen
b) Informatik und Ethik</td><td>Die digitale Welt 011 – Verantwortung der Informatik</td></tr>
</table>

3.5.4 UV Q-Phase IV – Automaten und formale Sprachen

Thema: Automaten und formale Sprachen

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

Vorhabensbezogene Konkretisierung:
Ausgehend von der Beschreibung und Untersuchung realer Automaten wird das formale Modell eines endlichen Automaten entwickelt. Neben dem Mealy-Automaten geht es vor allem um den erkennenden endlichen Automaten. Auf die Erarbeitung der Beschreibung folgt die Modellierung
eigener Automaten und die Untersuchung bestehender, um die Eigenschaften und Grenzen eines endlichen Automaten zu erkennen. Hierbei wird dessen Verhalten auf bestimmte Eingaben analysiert.
An den Themenkomplex *Endliche Automaten* schließt sich die Erarbeitung von Grammatiken regulärer Sprachen an. Die Untersuchung beginnt bei die Erschließung der formalen Beschreibung und wird mit der Entwicklung von Grammatiken zu regulären Sprachen fortgeführt. Hierbei wird auch die Beziehung von Grammatiken regulärer Sprachen zu endlichen Automaten an Beispielen erarbeitet und analysiert. Hierzu gehört auch die Untersuchung, welche Problemstellungen durch endliche Automaten und reguläre Grammatiken beschrieben werden können und welche nicht.

Zeitbedarf: 20 Std.

Sequenzierung des Unterrichtsvorhabens: s. nächste Seite

<table>
<tr><th>Unterrichtssequenzen</th><th>Zu entwickelnde Kompetenzen</th><th>Kapitel und Materialien</th></tr>
<tr><td>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</td><td rowspan="4">Die Schülerinnen und Schüler …
- 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)
- 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).</td><td><u>Kapitel 4 Endliche Automaten und formale Sprachen</u>
<u>4.1 Endliche Automaten</u>
<u>Projekteinstieg: Schatzsuche</u></td></tr>
<tr><td>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.</td><td><u>4.2 Formale Sprachen</u></td></tr>
<tr><td>3. Übungen und Vertiefungen
Verwendung endlicher Automaten und Grammatiken regulärer Sprachen</td><td><u>4.5 Prüfungsvorbereitung</u></td></tr>
<tr><td>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</td><td><u>Die digitale Welt 101 – Maschinennahe Programmierung</u>
<u>Die digitale Welt 100 – Berechenbarkeit</u></td></tr>
</table>

3.5.5 UV Q-Phase V – Organisation und Verarbeitung von Daten II – Modellierung und Implementierung von Anwendungen mit dynamischen nicht-linearen Datenstrukturen

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

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

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: 20 Stunden

Sequenzierung des Unterrichtsvorhabens: s. nächste Seite

<table>
<tr><th>Unterrichtssequenzen</th><th>Zu entwickelnde Kompetenzen</th><th>Kapitel und Materialien</th></tr>
<tr><td>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</td><td rowspan="4">Die Schülerinnen und Schüler …
- 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).</td><td>Kapitel 5 Nicht-lineare Datenstrukturen
5.1 Spielen mit Struktur – Baumstrukturen
Projekteinstieg 1: Spielbäume</td></tr>
<tr><td>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)</td><td>5.2 Zwei Nachfolger sind genug! - Binäre Bäume
Implementation des Projekts Ahnenbaum</td></tr>
<tr><td>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)</td><td>5.3 Wer Ordnung hält, spart Zeit beim Suchen – Binäre Suchbäume
Projekteinstieg 2: Binäre Suchbäume
Implementation des Projekts Benutzerverwaltung</td></tr>
<tr><td>4. Übung und Vertiefungen der Verwendung von Binärbäumen oder binären Suchbäumen anhand weiterer Problemstellungen</td><td>5.8 Prüfungsvorbereitung</td></tr>
</table>

3.5.6 UV Q-Phase VI – Aufbau von und Kommunikation in Netzwerken

Thema: Aufbau von und 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: 12 Stunden

Sequenzierung des Unterrichtsvorhabens: s. nächste Seite

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 und Schüler … - 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
5. Übung und Vertiefung des Aufbaus von und der Kommunikation in Netzwerken		6.4 Prüfungsvorbereitung

3.5.7 UV Q-Phase VII – Nutzung und Modellierung von relationalen Datenbanken in Anwendungskontexten

Thema: Nutzung und Modellierung von relationalen Datenbanken in Anwendungskontexten

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

Vorhabenbezogene Konkretisierung:
Am Beispiel eines Online-Buchhandels wird der Aufbau einer Datenbank sowie wichtige Grundbegriffe erarbeitet. Die Schülerinnen und Schüler nehmen dabei zunächst die Sicht der Anwender 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: 20 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	Die Schülerinnen und Schüler … - 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 einen Datenbanksystem zu extrahieren (I),	**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

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	- ermitteln Ergebnisse von Datenbankabfragen über mehrere verknüpfte 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 Datenbank-modellierung (A), - erläutern die Eigenschaften normalisierter Datenbankschemata (A), - überprüfen Datenbankschemata auf vorgegebene Normalisierungseigenschaften (D). - überführen Datenbankschemata in die 1. bis 3. Normalform (M).	
2. Modellierung von relationalen Datenbanken a) Datenbankentwurf durch ER-Diagramme - Ermittlung von Entitäten, zugehörigen Attributen, Beziehungen und Kardinalitäten in Anwendungssituationen und Modellierung eines Datenbankentwurfs in Form eines Entity-Relationship-Diagramms - Erläuterung und Erweiterung einer Datenbankmodellierung b) Entwicklung eines relationalen Modells aus einem Datenbankentwurf - Überführung eines Entity-Relationship-Diagramms in ein relationales Datenbankschema inklusive der Bestimmung von Primär- und Fremdschlüsseln c) Normalformen - Überprüfung von Datenbankschemata hinsichtlich der 1. bis 3. Normalform und Normalisierung (um Redundanzen zu vermeiden und Konsistenz zu gewährleisten)		7.5 Datenbankentwurf 7.6 Umsetzung des ER-Modells Wiederaufgriff des Projekteinstiegs 7.7 Datenbanken verbessern durch Normalformen

3. Übung und Vertiefung der Nutzung und Modellierung von relationalen Datenbanken		7.8 Prüfungsvorbereitung

3.6 Überfachliche Grundsätze

1. Geeignete Problem- / Fragestellungen zeichnen die Ziele des Unterrichts vor und bestimmen die Struktur der Lernprozesse.
2. Inhalt und Anforderungsniveau des Unterrichts entsprechen dem Leistungsvermögen der Schüler/innen.
3. Die Unterrichtsgestaltung ist auf die Ziele und Inhalte abgestimmt.
4. Medien und Arbeitsmittel sind schülernah gewählt.
5. Die Schüler/innen erreichen einen Lernzuwachs.
6. Der Unterricht fördert eine aktive Teilnahme der Schüler/innen.
7. Der Unterricht fördert die Zusammenarbeit zwischen den Schülern/innen und bietet ihnen Möglichkeiten zu eigenen Lösungen.
8. Der Unterricht berücksichtigt die individuellen Lernwege der einzelnen Schüler/innen.
9. Die Schüler/innen erhalten Gelegenheit zu selbstständiger Arbeit und werden dabei unterstützt.
10. Der Unterricht fördert strukturierte und funktionale Partner- bzw. Gruppenarbeit und andere Sozialformen.
11. Der Unterricht fördert strukturierte und funktionale Arbeit im Plenum.
12. Die Lernumgebung ist vorbereitet; der Ordnungsrahmen wird eingehalten.
13. Die Lehr- und Lernzeit wird intensiv für Unterrichtszwecke genutzt.
14. Es herrscht ein positives pädagogisches Klima im Unterricht.

3.6.1 Fachliche Grundsätze

1. Fachbegriffe werden den Schülern alters- und situationsbedingt angemessen vermittelt. Sie sind an Fachinhalte der Informatik gebunden und werden im Wesentlichen in ihren Kontexten anwendungsbezogen erarbeitet.
2. Fachmethoden (z.B. Analysen von Modellen, Implementation) werden immer durch die inhaltlichen Kontexte motiviert und in ihnen angewendet (keine Analyse um der Analyse willen).
3. Die Fachinhalte sollen die Vielfalt der informatorischen- und Medienlandschaft widerspiegeln.
4. Der Unterricht soll vernetzendes Denken fördern und deshalb phasenweise handlungsorientiert, fächerübergreifend und ggf. auch projektartig angelegt sein.
5. Der Unterricht ist grundsätzlich an den Kompetenzen der Schülerinnen und Schüler orientiert und knüpft an deren Vorkenntnissen, Interessen und Erfahrungen an. Dies betrifft insbesondere Vorerfahrungen im

Bereich des Programmierens. Schülerinnen und Schüler mit entsprechenden Fähigkeiten sollen im Unterricht ein adäquates Forum erhalten, ohne dass die anderen dadurch benachteiligt werden.

6. Der Unterricht ist problemorientiert und soll von realen, möglichst alltagsnahen Problemen ausgehen.
7. Lerninhalte sind so (exemplarisch) zu wählen, dass die geforderten Kompetenzen erworben und geübt werden können bzw. erworbene Kompetenzen an neuen Lerninhalten erprobt werden können. In der Benutzung der Fachterminologie ist auf schwerpunktartige und kontinuierliche Wiederholung zu achten.

3.6.2 Anmerkung zu beruflichen Handlungskompetenzen im Sinne der Studien- und Berufsorientierung:

Bei nahezu allen projektartig angelegten und eigenständig durchgeführten Arbeiten im Informatikunterricht werden die folgenden Handlungskompetenzen angebahnt und bei den SuS trainiert:
Zuverlässigkeit (A1), Selbständigkeit (A2), Selbstreflexion (A3), Motivation und Leistungsbereitschaft (A4), Konzentration (A5), Zeitmanagement (A8), Verantwortungsbereitschaft (B1), Kommunikation (B2), Kooperation (B3), Kritikfähigkeit (B4), Lern- und Arbeitsmethoden (C1), Informationsbeschaffung (C2), Strukturierung (C3) und Präsentation (C4). Auch die Fachkompetenzen Texproduktion und -rezeption (D1, D2) und Allgemeinwissen (D6) (s. StuBO-Curriculum).

Einen Beitrag zur Berufsorientierung leistet auch die Bearbeitung von Aufgaben mit Hilfe von Apps, die auch im späteren Lebensverlauf noch einmal aufkommen können (wie z. B. Word, Powerpoint, etc.).

3.6.3 Fachterminologie

Im Rahmen des Fachunterrichts in der Sek I und der Sek II wird bei den SuS ein angemessenen Fachvokabular eingeführt und im Kontext eines bildungssprachlichen Niveaus trainiert. Dies entwickelt sich kongruent im Rahmen der Lehrpläne.

Erste Begriffe werden in der Sek I angebahnt und eingeführt. Diese werden in der EF wiederaufgegriffen und vertieft. In der Q-Phase werden diese unterfüttert und mit weiterer Fachterminologie ergänzt.

3.7 Grundsätze der Leistungsbewertung und Leistungsrückmeldung

Neben der Vorgaben des Kernlehrplans Informatik verständigt sich die Fachschaft Informatik mit Bezug auf das allgemeine Leistungskonzept des Städtischen Gymnasiums Erwitte auf folgende Grundsätze und Absprachen:

Die Bewertung im Rahmen der verbindlichen Unterrichtsvorhaben orientiert sich u.a. an Evaluationsbögen, die zur individuellen Kontrolle der erreichten Kompetenzen kontinuierlich im Unterricht eingesetzt werden. Sie dienen der Transparenz der zu erlernenden Kompetenzen, der Selbstevaluation der Schülerinnen und Schüler und als Kriterien für den/die Lehrer/in. Der/die Lehrer/in trägt dafür Sorge, dass die dort aufgelisteten Kompetenzen im Unterricht zur Anwendung kommen und individuell überprüfbar sind.

Für die Schülerinnen und Schüler soll im Unterricht jeweils eine deutliche Unterscheidung von Lern- und Leistungssituation gemacht werden. Dies gilt insbesondere im Prozess von Projektaufgaben.

Allgemeine Bewertung der Mündlichen Leistungen
Neben Äußerungen im Unterrichtsgespräch (Quantität und Qualität) und die allgemeine Bereitschaft, sich am Unterricht zu beteiligen, können auch schriftlich erledigte Aufgaben aber auch praktisch-digitale Ergebnisse für die Leistungsbewertung herangezogen werden.

Mappe und Heftführung
Die Bewertung einer „Mappe“ erfolgt im digitalen Rahmen (Speicherstruktur auf dem PC oder in Teams) und nur nach vorgehender Festlegung der Kriterien. Diese umfassen in der Regel die Bereiche Vollständigkeit, Ordentlichkeit, Strukturiertheit und Richtigkeit. Sie soll individuelle Gestaltungspielräume berücksichtigen.

Schriftliche Überprüfungen
Pro Halbjahr werden in der Regel 2 Schriftliche Überprüfungen (mehr oder auch weniger oder gar keine möglich) zur Überprüfung des Verständnisses und der Anwendungsfähigkeit der in einem Unterrichtsvorhaben behandelten Fachbegriffe und Kompetenzen durchgeführt.
Bei einem Täuschungsversuch ist entsprechend den Richtlinien in der APO-Sek I oder GOSt zu verfahren. Entsprechende Maßnahmen (z.B. Teil- oder komplette Bewertung mit der Note 6 („ungenügend“)) behält sich die Lehrkraft vor.

Klassenarbeiten
In der Sekundarstufe II werden im Fach Informatik Klausuren geschrieben, wenn SuS Informatik als Klausurfach gewählt haben. In der EF gibt es pro Halbjahr eine, in der Q-Phase pro Halbjahr zwei Klausuren.

Die Fachschaft einigt sich auf die folgenden Klausurlängen unten. Abitur und Vorabitur sind vorgegeben. Die Klausurzeiten sind in Minuten angegeben und

steigern sich in der Länge sukzessive, um auf die Längen der Abiturzeiten vorzubereiten.

Stufe	GK	LK
EF	90	-
Q1.1	120	135
Q1.2	150	180
Q2.1	180	220
Q2.2 - Vorabitur	225	270
Q2.2 - Abitur	225	270

Facharbeit
Eine Klausur in der Q1.1 kann durch die Facharbeit ersetzt werden. Diese wird in enger Zusammenarbeit mit der Fachlehrkraft vorbereitet. Die Begleitgespräche sind neben der fertigen Arbeit für die Bewertung mit heranzuziehen. Weitere Informationen zur Facharbeit im Fach Informatik sind bei den entsprechenden Methodentagen oder auf Nachfrage zu bekommen.

Mündliche Abiturprüfung
Im Fach Informatik kann eine mündliche Abiturprüfung abgelegt werden. Hierzu muss das Fach in der Q-Phase durchgängig schriftliche belegt werden. Weitere Informationen dazu bekommen SuS auf Veranstaltungen der Oberstufenleitung und im Rahmen der Abiturvorbereitung in den Fächern. Eine mündliche Prüfung wird in der Q2 simuliert.

Nicht erbrachte Leistungen
Nicht erbrachte Leistungen sind mit der Note 6 („ungenügend“) zu bewerten. Die Lehrkraft behält sich außerdem vor, nicht fristgerecht abgegebene Aufgaben nicht mehr anzunehmen und ebenfalls mit der Note 6 zu bewerten.

3.8 Absprachen bezüglich Hausaufgaben

Die Fachkonferenz Informatik hat die Vereinbarung getroffen, dass Hausaufgaben zur besonders zur Sicherung und Vertiefung von Lerninhalten gegeben werden. Selbstverständlich können Hausaufgaben auch erteilt werden, um die Unterrichtsinhalte der folgenden Stunden vorzubereiten. Im Regelfall sollte der Aufwand für das Erledigen der Hausaufgaben den Aufwand von einer halben Stunde in der Woche nicht übersteigen. Vor dem Hintergrund individueller Förderung der Schülerinnen und Schüler sollen die Hausaufgaben differenziert erteilt werden. Dies kann sich im variierenden Umfang der Hausaufgaben oder auch beispielweise in unterschiedlichen Schwierigkeitsgraden der erteilten Hausaufgaben äußern.

Hausaufgaben sind außerdem ein weiteres Mittel der Sprachbildung v. a. im dem Bereich der Schriftsprache.

3.9 Lehr- und Lernmittel

Zurzeit stehen u.a. folgende Arbeitsmaterialien unten zur Verfügung. Ihre konkrete Zuordnung und weitere Aspekte sind in den ausformulierten Unterrichtsvorhaben oben notiert.

Schulbücher

- Jede/r SuS hat ein eigenes Schulbuch jeweils passend Klasse 6, zur EF oder zur Q-Phase ausgegeben.

Digitales

- Computer im SLZ und im Informatikraum
- iPads (Leihgeräte von der Schule oder standardisiert ab Klasse 8)
- Programme und Apps (z.B. Programmierumgebungen, Programme zur Darstellung von Informatiksystemen, Textverarbeitungsprogramme (Word, Open Office, etc.) und Präsentationsmedien (Powerpoint, Keynote, etc.))
- Projektionsflächen und Arbeitsbereiche (Beamer, Leinwand und Smartboards)

Sonstiges

- Selbst erstellte Arbeitsblätter und Arbeitsaufträge
- Visualisierungen und Verschriftlichungen von informatorischen Modellen und Implementationen
- Bild-, Video- und Filmmaterial

4 Entscheidungen zu fach- und unterrichtsübergreifenden Fragen

Das Ziel der Fachschaft Informatik ist es, die Koordination mit anderen Fächern in den jeweiligen Jahrgangsstufen in unterschiedlichen Formen zu erweitern. Diese kann in der thematischen Abstimmung mit einer sinnvollen Rhythmisierung bestehen, bis hin zu fächerübergreifenden und fächerverbindenden Absprachen (Beispiele s.o. UVs).

5 Qualitätssicherung und Evaluation

Das vorliegende schulinterne Curriculum im Fach Informatik ist nach den aktuellen Lehrplänen G9 entworfen worden und unterliegt einer fortlaufenden Qualitätssicherung und Evaluation.

Dies geschieht auf Basis der Erfahrung der unterrichtenden Kollegen und auf der Grundlage von statistisch erhobenen Fragebögen und anderem Feedback in ausgewählten Klassenstufen. Revisionen am Hauscurriculum Informatik sind möglichst zum Beginn des neuen Schuljahres vorzunehmen, damit sie von der Fachkonferenz beschlossen werden können.