

RAHMENRICHTLINIEN GYMNASIUM

INFORMATIK

Wahlpflichtfach: Schuljahrgänge 10 - 12



KULTUSMINISTERIUM

An der Überarbeitung der Rahmenrichtlinien haben mitgewirkt:

Eschrich, Mario	Schochwitz
Dr. Gutzer, Hannes	Halle (betreuender Dezernent des LISA)
Dr. Herper, Henry	Magdeburg (fachwissenschaftlicher Berater)
Dr. Lehmann, Hans	Osterburg
Zuber, Jörn	Blankenburg

Verantwortlich für den Inhalt:

Kultusministerium des Landes Sachsen-Anhalt

Vorwort

Bildung und Ausbildung sind Voraussetzungen für die Entfaltung der Persönlichkeit eines jeden Menschen wie auch für die Leistungsfähigkeit von Staat, Wirtschaft und Gesellschaft. Schule ist also kein Selbstzweck, sondern hat die jeweils junge Generation so gründlich und umfassend wie möglich auf ihre persönliche, berufliche und gesellschaftliche Zukunft nach der Schulzeit vorzubereiten. Dazu soll sie alle Schülerinnen und Schüler fördern, wo sie Schwächen haben, und in ihren Stärken fordern. Jede(r) soll die ihr bzw. ihm mögliche Leistung erbringen können und die dafür gebührende Anerkennung erhalten.

Dies gilt grundsätzlich nicht nur für Lerninhalte, sondern für alle Bereiche der persönlichen Entwicklung einschließlich des Sozialverhaltens. Gleichwohl haben gerade Rahmenrichtlinien die Schule als Ort ernsthaften und konzentrierten Lernens zu begreifen und darzustellen. Lernen umfasst dabei über solides Grundwissen hinaus alles, was dazu dient, die Welt in ihren verschiedenen Aspekten und Zusammenhängen besser zu verstehen und sich selbst an sinnvollen Zielen und Aufgaben zu entfalten.

Die Rahmenrichtlinien weisen verbindliche Unterrichtsziele und -inhalte aus. Sie können und sollen jedoch nicht die pädagogische Verantwortung der einzelnen Lehrerin und des einzelnen Lehrers ersetzen:

- Die Vermittlung der verbindlichen Unterrichtsinhalte füllt keineswegs alle Unterrichtsstunden aus. Daneben besteht auch Zeit für frei ausgewählte Themen oder Schwerpunkte. Dies bedeutet nicht zwangsläufig neue oder mehr Unterrichtsinhalte. Weniger kann unter Umständen mehr sein. Entscheidend für eine erfolgreiche Vermittlung von Wissen und Schlüsselkompetenzen ist, dass dem Erwerb elementarer Grundkenntnisse und -fertigkeiten ausreichend Zeit und Raum gewidmet wird. Soweit erforderlich, ist länger daran zu verweilen und regelmäßig darauf zurück zu kommen.
- Rahmenrichtlinien beschreiben nicht alles, was eine gute Schule braucht. Ebenso bedeutsam für die Qualität einer Schule ist die Lern- und Verhaltenskultur, die an ihr herrscht. Eine Atmosphäre, die die Lernfunktion der Schule in den Vordergrund stellt und die Einhaltung von Regeln des Miteinanders beachtet, kann nicht über Vorschriften, sondern nur durch die einzelne Lehrkraft und das Kollegium in enger Zusammenarbeit mit Eltern und Schülern erreicht werden.

Die Rahmenrichtlinien haben ein Anhörungsverfahren durchlaufen, an dem sich sehr viele Institutionen und Einzelpersonen beteiligten. Nach dieser Anhörung erfolgte eine redaktionelle Bearbeitung der vorliegenden Rahmenrichtlinien.

Die in diesem Heft enthaltenen Rahmenrichtlinien treten am 1. August 2003 in Kraft. Sie unterliegen einer vierjährigen Erprobungszeit. In dieser Zeit bitte ich alle Lehrerinnen und Lehrer um Hinweise oder Stellungnahmen, damit wir die Rahmenrichtlinien evaluieren, Probleme erkennen und Verbesserungen erarbeiten können.

Allen, die an der Entstehung dieser erneuerten Rahmenrichtlinien mitgewirkt haben, danke ich herzlich.

Ich wünsche allen Lehrerinnen und Lehrern bei der Planung und Gestaltung ihres Unterrichts viel Erfolg und Freude bei der pädagogischen Arbeit.



Prof. Dr. Jan-Hendrik Olbertz
Kultusminister

Magdeburg, im Mai 2003

Inhaltsverzeichnis

	Seite
1 Aufgaben des Wahlpflichtfaches Informatik am Gymnasium.....	6
2 Qualifikationen und fachdidaktische Konzeption	7
3 Zur Arbeit mit den Rahmenrichtlinien	10
4 Grundsätze der Unterrichtsgestaltung.....	11
4.1 Didaktische Grundsätze.....	11
4.2 Leistungen und ihre Bewertung	12
5 Inhalte.....	13
5.1 Übersichten	13
5.1.1 Themen im Schuljahrgang 10 (Einführungsphase)	13
5.1.2 Themen in den Schuljahrgängen 11/12 (Qualifikationsphase)	13
5.2 Darstellung der Themen	15
5.2.1 Schuljahrgang 10 (Einführungsphase).....	15
5.2.2 Schuljahrgänge 11/12 (Qualifikationsphase).....	23

1 Aufgaben des Wahlpflichtfaches Informatik am Gymnasium

Der Informatik als Wissenschaft kommt beim Übergang zu einer Informationsgesellschaft eine Schlüsselrolle zu, da sie systematisch Möglichkeiten der automatischen Informationsverarbeitung und Wissensrepräsentation erforscht und in Informatiksystemen nutzbar macht. Das Wahlpflichtfach Informatik am Gymnasium ist hinsichtlich seines Gegenstandes und seiner Lern- und Arbeitsmethoden als integrativer Bestandteil des mathematisch-naturwissenschaftlichen Unterrichts anzusehen. Seine Aufgaben gehen im Rahmen des fächerübergreifend zu betrachtenden Bildungs- und Erziehungsprozesses über den nahe liegenden Bezug zu den naturwissenschaftlichen Fächern hinaus. In den Schuljahrgängen 7 bis 9 werden im Rahmen der Informations- und Kommunikationstechnologischen Grundbildung (IKG) und in Wahlpflichtkursen (zum Beispiel „Einführung in die Arbeit am Computer“, „Moderne Medienwelten/Informatik“) Grundlagen zur Nutzung des Computers in Verbindung mit Standardsoftware vermittelt. Die dynamische Entwicklung der Informatik und Informationstechnologie und deren Anwendung bedingen eine ständige Erweiterung und Wichtung der Unterrichtsinhalte und Arbeitsmethoden.

Die Aufgabe des Informatikunterrichtes in den Schuljahrgängen 10 bis 12 besteht darin, Schülerinnen und Schülern einen Zugang zu informatischen Denk- und Arbeitsweisen sowie Informatiksystemen zu öffnen. Informatische Bildung ist das Ergebnis von Lernprozessen, in denen Grundlagen, Methoden, Anwendungen, Denk- und Arbeitsweisen sowie die gesellschaftliche Bedeutung von Informatiksystemen Unterrichtsgegenstände und damit in Hinblick auf das wissenschaftspropädeutische Arbeiten in der Kursstufe bedeutsam sind. In allen Phasen der Lernprozesse im Wahlpflichtfach Informatik wird die Vermittlung von grundlegendem Wissen der informatischen Bildung realisiert. Dabei werden Grundkenntnisse aus den Bereichen der theoretischen Informatik, technischen Informatik, angewandten Informatik sowie deren gesellschaftliche Auswirkungen vermittelt. Damit wird auch deutlich, dass jedes Informatiksystem als Kombination von Hard- und Software-Komponenten das Ergebnis des informatischen Modellierens ist. In allen Phasen der informatischen Bildung ist die Informatik die Bezugswissenschaft. Vielfältige und wechselseitige Beziehungen bestehen zu Gegenständen der anderen Naturwissenschaften, der Technik, aber auch zu den gesellschaftswissenschaftlichen Fächern. In diesem Rahmen ordnet sich das Fach Informatik in das Gesamtkonzept zum fächerübergreifenden Unterricht ein.

Im Unterricht werden die Vorteile aber auch die Risiken, Gefahren und Grenzen mit den gesellschaftlichen Auswirkungen bei der Nutzung von Informatiksystemen verdeutlicht. Das Erkennen und das Analysieren von Informatiksystemen in der realen Welt gehört damit zu den grundlegenden Aufgaben des Wahlpflichtfaches Informatik.

2 Qualifikationen und fachdidaktische Konzeption

Qualifikationen

Der Informatikunterricht als Kern einer allgemein bildenden informatischen Bildung in den Schuljahrgängen 10 bis 12 zielt auf die Herausbildung solcher Kompetenzen ab, die gegenwärtig und zukünftig bei der Arbeit mit Informatiksystemen von wesentlicher Bedeutung sind.

Fachkompetenz

Die Schülerinnen und Schüler

- besitzen Kenntnisse zu ausgewählten Methoden und Verfahren der Informatik und erhalten einen Einblick in formale Konzepte der Informatik,
- verfügen über Fähigkeiten und Fertigkeiten bei der Lösung von Aufgaben durch die Anwendung von Informatiksystemen,
- beherrschen grundlegende Denk- und Arbeitsweisen der Informatik, die sie sich durch Anwendung, Analyse, Modifikation und Bewertung ausgewählter Informatiksysteme angeeignet haben,
- haben grundlegende Kenntnisse über Aufbau und Einsatz von Informatiksystemen,
- können Informatiksysteme bezüglich ihrer Funktion und Struktur historisch einordnen,
- kennen theoretische Grundlagen der Arbeitsweise von Computern,
- kennen wesentliche Funktionen von Betriebssystemen und können diese anwenden,
- kennen grundlegende Funktionen und Strukturen von Rechnernetzen und deren theoretische Grundlagen,
- können Algorithmen entwerfen und diese mit einer Programmiersprache implementieren,
- besitzen umfassende Kenntnisse in einem der angebotenen Wahlthemen,
- kennen die Phasen des informatischen Modellierens und können diese an einem Projekt umsetzen.

Methodenkompetenz

Die Schülerinnen und Schüler

- können Informationen beschaffen, strukturieren, bearbeiten, speichern, darstellen, bewerten und in geeigneter Form präsentieren,
- können fächerübergreifend die grundlegenden Denk- und Arbeitsweisen der Informatik mit den Inhalten anderer Fächer verknüpfen,
- können mit produktbezogenen Hilfesystemen arbeiten und sind in der Lage, sich eigenständig in unbekannte Systeme einzuarbeiten.

Sozialkompetenz

Die Schülerinnen und Schüler

- sind in der Lage, miteinander zu lernen, zu kommunizieren und selbst als Mitglied einer Lerngruppe Verantwortung zu übernehmen sowie Kritik anzunehmen und konsensfähige Lösungen anzustreben,
- besitzen Kenntnisse und Fähigkeiten zur Auseinandersetzung mit gesellschaftlichen, sozialen und ethischen Grundfragen.

Selbstkompetenz

Die Schülerinnen und Schüler

- erfahren im Umgang mit Informatiksystemen persönlich bedeutsame Werte und entdecken dabei individuelle Neigungen, Begabungen und Interessen.

Fachdidaktische Konzeption

Die in den Aufgaben des Faches Informatik am Gymnasium charakterisierte informatische Bildung orientiert sich an den nachstehenden Leitlinien

- Interaktion mit Informatiksystemen,
- Wirkprinzipien von Informatiksystemen,
- Informatische Modellierung und
- Wechselwirkungen zwischen Informatiksystemen, Individuum und Gesellschaft.

Die unter diesen Leitlinien strukturierten Kenntnisse und Fertigkeiten werden, auf unterschiedlichem Ausgangsniveau basierend, im Informatikunterricht in den Schuljahrgängen 10 bis 12 erworben, wobei stets an die Erfahrungs- und Lebenswelt der Schülerinnen und Schüler anzuknüpfen ist. Der Informatikunterricht ist so zu gestalten, dass die Schülerinnen

und Schüler durch vielfältige geistige Tätigkeiten und praktische Arbeiten mit Informatiksystemen ein anwendungsbereites Wissen im Umgang mit digital dargestellten Informationen und die Beherrschung von Informatiksystemen als unverzichtbare Ergänzungen der traditionellen Kulturtechniken erwerben. Zu diesen zählen

- die Beschaffung von Informationen,
- die Darstellung von Informationen in maschinell verarbeitbaren Zeichen (Daten),
- die maschinelle Verarbeitung und Verteilung der Daten und
- die Gewinnung neuer Informationen durch Interpretation der gewonnenen Daten, die zusammen mit dem Vorwissen zu neuem Wissen führt.

Der Gegenstand der Informatik als junge und dynamische Wissenschaft stellt in Anbetracht des zur Verfügung stehenden Zeitvolumens hohe Anforderungen an die didaktische Reduktion der Lerngegenstände. Daraus resultiert eine weitgehend exemplarische Behandlung der Inhalte. Die Rahmenrichtlinien im Schuljahrgang 10 enthalten eine Einführungsphase (Gelenkfunktion), um die Schaffung eines gesicherten Basisniveaus für alle Schülerinnen und Schüler zu ermöglichen. Hierzu ist eine starke Differenzierung im Unterricht notwendig. Die vorliegenden Rahmenrichtlinien gehen davon aus, dass jeder Schülerin und jedem Schüler ein eigenständiger Computerarbeitsplatz zur Verfügung steht.

Die im 10. Schuljahrgang behandelten zentralen Themen „Informatik und Gesellschaft“ und „Computer-Netzwerke“ haben eine besondere Bedeutung für den Informatikunterricht, da Computer heute in der Regel keine eigenständigen Arbeitsstationen, sondern Bestandteil von lokalen und globalen Netzwerken sind und in vielen gesellschaftlichen Bereichen Einzug gehalten haben.

In der Qualifikationsphase erfahren die Schülerinnen und Schüler die grundlegenden Techniken des informatischen Modellierens als eine wesentliche Grundlage für die Entwicklung von Informatiksystemen. Das schließt das Erlernen einer Programmiersprache ein. Da die Informatik über ein sehr breites Spektrum an Wissensgebieten verfügt, werden im Kurshalbjahr 12/1 Wahlthemen angeboten. Die Auswahl des Themas sollte sich an den Interessen der Schülerinnen und Schüler sowie an den schulischen Gegebenheiten orientieren. Entsprechend der Profilierung der Schule oder auf Grund aktueller Entwicklungen der Fachwissenschaft Informatik ist es möglich, abweichend von den vorgegebenen Wahlthemen ein eigenes Thema zu entwickeln, das sich an dem vorgegebenen Anforderungsniveau der anderen Wahlthemen orientieren muss.

Die Projektarbeit im Kurshalbjahr 12/2 dient der Annäherung an die angestrebten Qualifikationen und Kompetenzen.

3 Zur Arbeit mit den Rahmenrichtlinien

Die Rahmenrichtlinien stellen die Grundlage für die Gestaltung des gesamten Informatikunterrichtes in der Einführungs- und Qualifikationsphase dar. Für die schulische Planungsarbeit bilden die Qualifikationen den verbindlichen Rahmen.

Die in den Schuljahrgängen 10 und 11 vorgegebenen Themen sind Pflichtthemen. Die Verbindlichkeit bezieht sich auf die Themen, die zugeordneten Ziele und Qualifikationen sowie die jeweiligen Inhalte. Die Reihenfolge der Themen innerhalb der Jahrgänge 10 und 11 kann von der Lehrkraft verändert werden, soweit es die fachliche Logik zulässt. Die themenbezogenen Zeitrichtwerte sind Empfehlungen. Die verbleibende Unterrichtszeit sollte

- zur ergänzenden bzw. vertiefenden Behandlung von Inhalten entsprechend der Interessenlage der Schülerinnen und Schülern und
- zur Wiederholung, Zusammenfassung und Systematisierung

verwendet werden.

Im Kurshalbjahr 12/1 ist eines der vorgegebenen Wahlthemen verbindlich. Die Bearbeitung eines anderen Wahlthemas ist im Rahmen der im Abschnitt 2 getroffenen Regelungen möglich.

Die Hinweise zum Unterricht enthalten Anregungen zu inhaltlichen Schwerpunktsetzungen, Ausprägungen von Sach-, Methoden- und Sozialkompetenz, Abstimmungen zu anderen Fächern sowie zur Nutzung von Lehr- und Lernmitteln. Sie tragen ausdrücklich Empfehlungscharakter und stellen eine Planungshilfe dar. Durch die Auswahl aus den Hinweisen in Verbindung mit eigenen Vorstellungen und Erfahrungen der Lehrkraft kann der konkreten Lern- und Unterrichtssituation Rechnung getragen werden.

4 Grundsätze der Unterrichtsgestaltung

4.1 Didaktische Grundsätze

Im Interesse einer schülerorientierten Unterrichtsgestaltung greift der Informatikunterricht vorhandene Erfahrungen und Vorstellungen auf. Ausgehend davon soll eine Korrektur, Erweiterung, Vertiefung und Systematisierung von Kenntnissen, Fähigkeiten und Fertigkeiten zu den Grundlagen der Informatik, verbunden mit aktuellen Erkenntnissen und Forschungsvorhaben erreicht werden. Besonderer Wert wird auf die Ausprägung von praktischen Fähigkeiten und Fertigkeiten gelegt – als Ausdruck des handlungsorientierten Ansatzes dieser Rahmenrichtlinien. Dabei werden die Schülerinnen und Schüler insbesondere in der Projektphase in ihrer zunehmend selbstständigen Arbeitsweise differenziert begleitet. Hierzu ist es erforderlich, dass jedem Lernenden in der praktischen Arbeit ein Computerarbeitsplatz zur Verfügung steht.

Im Umgang mit Informatiksystemen spielen Hilfesysteme und ähnliche elektronische Nachschlagewerke eine zentrale Rolle. Aus diesem Grund sollen die Schülerinnen und Schüler die hierfür notwendigen Techniken erlernen und bewusst im Verlauf des gesamten Grundkurses anwenden. Insbesondere sei hier auch auf den fächerübergreifenden Aspekt dieser speziellen Nachschlagetechnik hingewiesen.

Ausgewählte Konzepte und Komplexitätsbetrachtungen fördern die Fähigkeit der Schülerinnen und Schüler zur Bewertung von Aufgabenlösungen.

Das Aufzeigen der Struktur eines Problems durch Modellierung führt zu geeigneten Lösungsmodellen. In das informatische Modellieren wird mit den Phasen Problemgewinnung, informelle Problembeschreibung, formale Modellierung, Realisierung von Lösungsansätzen und Bewertung eingeführt. Verschiedene Modellierungsverfahren führen dabei zu verschiedenen Problemlösestrategien.

Komplexere Fragestellungen und Probleme und die in diesem Zusammenhang zu entwickelnden Lösungsstrategien können durch fächerübergreifendes Herangehen vermittelt werden. Die einzelnen Projektphasen eignen sich zur Verbindung mit den Inhalten anderer Fächer. Somit besteht die Möglichkeit, komplexe Probleme fächerübergreifend zu bearbeiten.

4.2 Leistungen und ihre Bewertung

Leistungsbewertungen geben den Schülerinnen und Schülern sowie den Erziehungsberechtigten Auskunft über den Leistungsstand und der Lehrkraft die Möglichkeit zu einem Vergleich zwischen den angestrebten und den erreichten Lernzielen. Leistungsnachweise sind die Grundlage einer Leistungsbewertung. Sie umfassen mündliche und schriftliche Nachweise sowie praktische Arbeiten am Computer und die Projektarbeit. Die Fach-, Methoden-, Sozial- und Selbstkompetenzen der Schülerinnen und Schüler sind beim Leistungsnachweis und der Leistungsbewertung in angemessener Weise zu berücksichtigen. Über Formen und Anzahl von Leistungsnachweisen entscheidet die Lehrkraft eigenverantwortlich auf der Grundlage der gültigen Erlasse sowie der Beschlüsse der Fach- und Gesamtkonferenzen der Schule. Das betrifft auch das Thema „Projektarbeit zur Softwareentwicklung“ im Kurshalbjahr 12/2, zu welchem Facharbeiten angefertigt werden.

5 Inhalte

5.1 Übersichten

5.1.1 Themen im Schuljahrgang 10 (Einführungsphase)

Thema	Zeitrictwert
Grundlagen der Informationstechnik	16 Std.
Projektarbeit unter Nutzung von Standardsoftware	10 Std.
Informatik und Gesellschaft	6 Std.
Computer-Netzwerke	20 Std.

5.1.2 Themen in den Schuljahrgängen 11/12 (Qualifikationsphase)

Kurshalbjahr 11/1

Thema	Zeitrictwert
Algorithmenstrukturen und ihre Implementierung	26 Std.

Kurshalbjahr 11/2

Thema	Zeitrictwert
Strukturierte Datentypen	10 Std.
Informatisches Modellieren	16 Std.

Kurshalbjahr 12/1

Wahlthemen	Zeitrictwert
1. Modellbildung und Simulation	26 Std.
2. Analyse und Design eines Informatiksystems	26 Std.
3. Computergrafik	26 Std.
4. Abstrakte Datentypen und ihre Implementierung	26 Std.
5. Suchen und Sortieren von Daten	26 Std.
6. Endliche Automaten und formale Sprachen	26 Std.
7. Kryptologie	26 Std.
8. Einsatz von Datenbanken zur dynamischen Webseitengenerierung	26 Std.

Kurshalbjahr 12/2

Thema	Zeitrictwert
Projektarbeit zur Softwareentwicklung	16 Std.

5.2 Darstellung der Themen

5.2.1 Schuljahrgang 10 (Einführungsphase)

Thema: Grundlagen der Informationstechnik

ZRW: 16 Std.

Vorbemerkungen/Qualifikationen

Schwerpunkt dieses Themas ist die Vermittlung eines vertiefenden Verständnisses über den Aufbau und die Funktionsweise eines Informatiksystems. Es werden elementare Fähigkeiten und Fertigkeiten aus dem Bereich der Technischen Informatik herausgebildet. Die eigentlichen technischen Komponenten werden hierbei als funktionale Einheit (Black Box) benutzt und es wird nur Wert auf ihre Logik gelegt. Das Betriebssystem wird als zentrale Software des Rechners behandelt. Hierbei werden zentrale Strukturen und Funktionen vermittelt, ebenso vertiefende Fertigkeiten und Fähigkeiten im Umgang mit Betriebssystemen.

Die Schülerinnen und Schüler

- kennen Persönlichkeiten und ihre Leistungen, sowie technische und theoretische Entwicklungen aus der Historie der Informatik und können diese in den soziotechnischen Kontext einordnen,
- kennen den elementaren Aufbau und die prinzipielle Arbeitsweise einer Rechenanlage nach J. v. Neumann,
- kennen elementare Schaltfunktionen (Grundgatter) als elementare Bestandteile der Baugruppen eines J. v. Neumann-Rechners,
- können einfache logische Schaltnetze aus Grundgattern entwerfen und analysieren,
- können die Funktionsweise eines Halbaddierers, Volladdierers und FlipFlops erklären,
- kennen die Aufgaben von Betriebssystemen und sind in der Lage, mittels geeigneter Diagnose- und Verwaltungssoftware laufende Prozesse des Computers festzustellen und die Systemressourcen zu untersuchen,
- kennen das Schichtenmodell einer Rechneranlage und verstehen das Betriebssystem als Mittler zwischen Hard- und Software,
- kennen die Funktionsweise eines geeigneten Dateimanagementsystems und können dieses anwenden.

Inhalte	Hinweise zum Unterricht
<p>Historie</p> <ul style="list-style-type: none"> – Persönlichkeiten (Leibniz, Babbage, Hollerith, Zuse, von Neumann) – Hardware- und Softwareentwicklungen <ul style="list-style-type: none"> • Computergenerationen • Generationen von Softwareentwicklungswerkzeugen 	<ul style="list-style-type: none"> – Meilensteine der Informatik und ihre Einordnung im Kontext ihres sozio-technischen Umfeldes (Schülervorträge)
<ul style="list-style-type: none"> – Aufbau eines Rechners nach J. v. Neumann <ul style="list-style-type: none"> • Rechenwerk • Steuerwerk • Speicherwerk • Ein- und Ausgabekanäle – Funktionsmerkmale des Rechenwerks, Steuerwerks und Speicherwerks und ihr prinzipielles Zusammenwirken 	<ul style="list-style-type: none"> – Ausblick auch auf andere Rechnerarchitekturen – Funktionsdemonstration mit einem Modellrechner
<ul style="list-style-type: none"> – Begriffe Kodieren und Dekodieren <ul style="list-style-type: none"> • Umrechnung zwischen Zahlensystemen • ASCII/ANSI-Code 	<ul style="list-style-type: none"> – Wiederholung Dualzahlen – Binärsystem, Dezimalsystem, Hexadezimalsystem – Begriffe Bit, Byte
<ul style="list-style-type: none"> – grundlegende logische Schaltfunktionen (Grundgatter) und ihre Schaltbelegungstabellen – Analyse und Entwurf einfacher Schaltnetze bestehend aus Grundgattern 	<ul style="list-style-type: none"> – Simulation der Schaltfunktionen mittels Tabellenkalkulation oder mittels Simulationssoftware – Demonstration der realen Schaltfunktion
<ul style="list-style-type: none"> – Umsetzung der Addition mit Hilfe logischer Schaltelemente <ul style="list-style-type: none"> • Halbaddierer mit Schaltbelegungstabelle • Volladdierer mit Schaltbelegungstabelle 	<ul style="list-style-type: none"> – Wiederholung Operationen mit binären Zahlen
<ul style="list-style-type: none"> – Speicherung eines Bits mit Hilfe logischer Schaltelemente – FlipFlop mit Schaltbelegungstabelle 	

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none"> – Definition Betriebssystem – Aufgaben von Betriebssystemen <ul style="list-style-type: none"> • Abstraktion von Hardware und Schnittstelle für Anwendungsprogramme • Verwaltung von Prozessen (z. B. Programme in Ausführung) und Betriebsmitteln (z. B. Arbeitsspeicher und periphere Geräte) 	<ul style="list-style-type: none"> – Demonstration an einem realen Betriebssystem
<ul style="list-style-type: none"> – Schichtenmodell (Schalenmodell) einer Rechenanlage <ul style="list-style-type: none"> • Anwendersoftware • Betriebssystem • ROM-Software • Hardware 	<ul style="list-style-type: none"> – Einordnung von Softwareprodukten in das Schichtenmodell – Schichten eines Betriebssystems – Beispiel: BIOS beim PC
<ul style="list-style-type: none"> – Funktionsweise eines Dateimanagementsystems – Arbeit mit Diagnose-, Verwaltungs- und Wartungsprogrammen 	<ul style="list-style-type: none"> – Beispiel: Windows-Explorer – Beispiele aus der Windows-Welt: Taskmanager, Taskplaner, Scandisk, Defrag, Systemmonitor

Thema: Projektarbeit unter Nutzung von Standardsoftware ZRW: 10 Std.

Vorbemerkungen/Qualifikationen

Die im Rahmen der informatischen Grundbildung gewonnenen Kenntnisse, Fähigkeiten und Fertigkeiten im Umgang mit Standardsoftware sollen weiter gefestigt und aus informatischer Sicht systematisiert werden.

Die Schülerinnen und Schüler

- können die Grundfunktionen von Standardsoftware systematisieren und in einen informatischen Kontext einordnen,
- können produktbezogene Hilfesysteme zur Lösung ihrer Aufgaben nutzen,
- können mit geeigneter Standardsoftware thematische Belegarbeiten/Präsentationen anfertigen und vorstellen,
- kennen Objekte innerhalb von Standardsoftware,
- kennen Eigenschaften dieser Objekte und können diese verändern.

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none"> – Grundfunktionen von Standardsoftware <ul style="list-style-type: none"> • Dateiarbeit und Dokumentenverwaltung • Formatierung von Objekten – Nutzung von produktbezogenen Hilfesystemen – Erstellen einer thematischen Belegarbeit/Präsentation unter Verwendung mehrerer Standardsoftwareprodukte und elektronischer Nachschlagewerke 	<ul style="list-style-type: none"> – systematisches Herausarbeiten wesentlicher und übereinstimmender Grundfunktionen – Textobjekte (Zeichen, Absatz, Abschnitt, Dokument) – Externe Objekte (z. B. Grafik, Video, Sound, Formeln) – die thematische Belegarbeit/Präsentation kann im Laufe des Schuljahres angefertigt und in entsprechender Form verteidigt und bewertet werden – es sind nicht nur informatische Themen zulässig, sondern es ist auch das fächerübergreifende Arbeiten zu ermöglichen
<ul style="list-style-type: none"> – objektorientierte Betrachtungsweisen von Standardsoftware <ul style="list-style-type: none"> • Begriffe Klasse, Objekt, Methode, Eigenschaft (Attribut) • Identifikation der Objekte • Eigenschaften dieser Objekte • Analyse der Methoden auf Objekte 	<ul style="list-style-type: none"> – die Behandlung hat an mehreren Objekten zu erfolgen – Beispiel: <ul style="list-style-type: none"> • Klasse: Grafik • Objekt: konkretes Bild • Methoden: kopieren, verschieben, Methoden des Kontextmenüs • Eigenschaften: Breite, Höhe, Rahmen, Position

Thema: Informatik und Gesellschaft

ZRW: 6 Std.

Vorbemerkungen/Qualifikationen

Erst durch die Kenntnis von Voraussetzungen und Folgen, Chancen und Risiken des Einsatzes komplexer Informatiksysteme werden Schülerinnen und Schüler in die Lage versetzt, sich verantwortungsbewusst an der Gestaltung und am Einsatz dieser Technologie zu beteiligen und ihre Zukunft menschengerecht zu gestalten. Dazu setzen sie sich auch mit normativen und ethischen Fragen auseinander, die zum Beispiel den Zugriff auf personenbezogene Daten oder den Umgang mit dem Urheberrecht betreffen. Aus der Kenntnis der Wirkungen des Einsatzes von Informatiksystemen auf Individuum und Gesellschaft heraus sollen sie Kriterien für eine menschengerechte Technikgestaltung und

deren sozialverträglichen Einsatz entwickeln können. Überhöhten Erwartungen an das Machbare sollen sie ebenso entgegentreten wie fatalistischen Einstellungen des Ausgeliefertseins gegenüber Informatiksystemen. Weiterhin werden rechtliche und sicherheitsrelevante Fragestellungen behandelt, die sich aus der Nutzung von Informatiksystemen im soziotechnischen Umfeld ergeben.

Die Schülerinnen und Schüler

- kennen die Teilgebiete der Informatik und haben einen Einblick in ihre Aufgaben als Wissenschaft,
- kennen Einsatzmöglichkeiten von Informatiksystemen,
- erkennen Informatiksysteme in ihrer täglichen Umwelt,
- sind mit der Problematik des Datenschutzes und der Datensicherheit vertraut,
- können zwischen Datenschutz und Datensicherheit unterscheiden,
- kennen wesentliche gesetzliche Grundlagen des Datenschutzes und können daraus Persönlichkeitsrechte und Pflichten ableiten,
- wissen um grundlegende strafrechtliche Konsequenzen,
- kennen die Bedeutung der Datensicherheit und sind in der Lage, entsprechende Softwareprodukte zur Wahrung dieser Datensicherheit einzurichten und zu nutzen,
- erkennen die Notwendigkeit des Softwarerechts und kennen verschiedene lizenzrechtliche Modelle,
- kennen Voraussetzungen, Folgen, Chancen, Risiken und Grenzen des Einsatzes von Informatiksystemen.

Inhalte	Hinweise zum Unterricht
Information, Informatik und Informatiksystem – Begriffsbestimmung – Teilgebiete <ul style="list-style-type: none"> • Technische Informatik • Theoretische Informatik • Angewandte Informatik • Praktische Informatik 	– Informatiksystem als Einheit aus Hard- und Software – Begriff Information – Einsatzgebiete von Informatiksystemen <ul style="list-style-type: none"> • elektronische Zahlungssysteme (Registrierkasse) • elektronische Zugangs-, Identifikations- und Arbeitszeiterfassungssysteme
– Begriffe Datenschutz, Datensicherheit – Persönlichkeitsrecht und Datenschutz – Datenschutzgesetz – Teledienstgesetz	– aktuelle Beispiele aus den Medien – Datenschutz an der Schule – Informationspflicht (Impressum) bei Veröffentlichungen im Internet

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none"> - Datensicherheit <ul style="list-style-type: none"> • Backup • Computerviren/ Antivirenprogramme 	<ul style="list-style-type: none"> - Haltbarkeit von Datenträgern - Infektionswege - Arten von Computerviren - Schutzmaßnahmen - Verantwortlichkeiten bei Virenverbreitung
<ul style="list-style-type: none"> - Urheberrecht und Softwarerecht <ul style="list-style-type: none"> • Autoren- und Verwertungsrecht • Urheberrecht von Softwareprodukten • Lizenzformen 	<ul style="list-style-type: none"> - beachten der Rechtslage beim Anfertigen von Schülerprojekten - Kauflizenzen - rechner-spezifische Lizenzierung - Public Domain - Shareware - Freeware
<ul style="list-style-type: none"> - Grenzen von Informatiksystemen 	<ul style="list-style-type: none"> - ethische Aspekte - Datenschutz - Berechenbarkeit - physikalische und technische Grenzen

Thema: Computer-Netzwerke

ZRW: 20 Std.

Vorbemerkungen/Qualifikationen

Alle im Thema aufgeführten Inhalte sollten vorrangig aus der Sicht des in der Schule vorhandenen Netzwerkes und dessen Möglichkeiten behandelt und praktisch realisiert werden.

Die Schülerinnen und Schüler

- kennen wesentliche Abschnitte der historischen Entwicklung von Netzwerken, speziell des Internets,
- kennen Aufgaben von Netzwerken,
- können Datenschutz und Datensicherheit auf Netzwerke bewusst anwenden,
- kennen Einteilungsmöglichkeiten für Netzwerke und können das vorhandene schulische Netzwerk einordnen,

- wissen, was ein Protokoll aus informationstechnischer Sicht ist,
- kennen wesentliche Netzwerkdienste und die zugehörigen Protokolle,
- kennen Gemeinsamkeiten und Unterschiede zwischen Intranet und Internet,
- können eigene Webseiten und ein eigenes Webprojekt planen, erstellen und auf einem Webserver implementieren.

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none"> – globale und lokale Netzwerke <ul style="list-style-type: none"> • Begriffsbestimmung • historische Aspekte 	
<ul style="list-style-type: none"> – Aufgaben von Netzwerken <ul style="list-style-type: none"> • Datenfernübertragung • gemeinsame Nutzung von Ressourcen und von Peripherie • Arbeitsteilung in Netzen 	<ul style="list-style-type: none"> – am Beispiel des eigenen Netzwerks erläutern und bewusst nutzen
<ul style="list-style-type: none"> – Datenschutz und Datensicherheit in Netzwerken 	<ul style="list-style-type: none"> – Datenschutz <ul style="list-style-type: none"> • Bestandsdaten • Nutzungsdaten • Abrechnungsdaten • Inhaltsdaten – Datensicherheit <ul style="list-style-type: none"> • Zugangskontrolle • Zugriffsrechte (Freigaben und Berechtigungen) • Firewalls • Signaturen
Einteilungsmöglichkeiten von Netzwerken <ul style="list-style-type: none"> – nach Aufgabenverteilung – nach Topologien – nach Übertragungsweiten – nach Abgeschlossenheit 	<ul style="list-style-type: none"> – Client-Server, Peer to Peer – Bustopologie, Ringtopologie, Sterntopologie – LAN, MAN, WAN, GAN – Intranet, Internet – die unterschiedlichen Sichtweisen auf Netzwerke sollten am Beispiel des eigenen Netzwerkes behandelt werden

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none"> – Protokolle und Dienste <ul style="list-style-type: none"> • Begriffsbestimmung • Einteilung und Aufgaben • Zusammenhang zwischen Protokoll und Diensten 	<ul style="list-style-type: none"> – der Zusammenhang zwischen Protokoll und Diensten ist an geeigneten Beispielen aufzuzeigen
<ul style="list-style-type: none"> – Adressierung in TCP/IP-Netzwerken <ul style="list-style-type: none"> • IP-Adresse • DNS-Dienst (Domain Name Server) • Netzwerkklassen • Routing 	<ul style="list-style-type: none"> – feste Adressierung – dynamische Adressierung – DHCP-Server (Dynamic Host Configuration Protocol) – Klasse A, B, C-Netze
<ul style="list-style-type: none"> – Erstellung eines Webprojekts <ul style="list-style-type: none"> • Syntax und Semantik einer Seitenbeschreibungssprache • Nutzung eines geeigneten Editors • Integration clientseitiger Funktionalität 	<ul style="list-style-type: none"> – beim Einsatz visueller (X)HTML-Editoren ist der erzeugte Quelltext zu interpretieren und zu bearbeiten – Auswertung von Formularen (z. B. JavaScript, VBScript)

5.2.2 Schuljahrgänge 11/12 (Qualifikationsphase)

Kurshalbjahr 11/1

Thema: **Algorithmenstrukturen und ihre Implementierung** **ZRW: 26 Std.**

Vorbemerkungen/Qualifikationen

In diesem Thema werden ausgehend vom Algorithmusbegriff Algorithmenstrukturen eingeführt und in einer Programmiersprache implementiert. Die gewählte Programmiersprache sollte den objektorientierten Softwareentwicklungsansatz unterstützen.

Die Schülerinnen und Schüler

- kennen einen Algorithmusbegriff und Eigenschaften von Algorithmen,
- können Algorithmen verbal formulieren und formal darstellen,
- kennen den Begriff der Variablen in der Informatik,
- können Algorithmen analysieren, interpretieren und in einer Programmiersprache codieren,
- sind in der Lage, Programmabschnitte zu strukturieren und zu entsprechenden Einheiten zusammenzufassen,
- können Programmoberflächen gestalten und die Eigenschaften der visuellen Komponenten bearbeiten und ihre Methoden (Ereignisse) mit entsprechendem Quelltext versehen,
- kennen grafische Basisroutinen der verwendeten Programmiersprache und können diese anwenden.

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none">– intuitive Definition des Algorithmusbegriffs– Eigenschaften<ul style="list-style-type: none">• Endlichkeit• Eindeutigkeit• Ausführbarkeit• Allgemeingültigkeit	<ul style="list-style-type: none">– Handlungsvorschriften aus dem Alltag

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none"> – Beschreibungsmittel für Algorithmen <ul style="list-style-type: none"> • verbale Beschreibung • formale Beschreibung – Algorithmenstrukturen <ul style="list-style-type: none"> • Folge • Verzweigungen • Schleife 	<ul style="list-style-type: none"> – die Beschreibungsmittel sind an geeigneten Beispielen vorzustellen – Struktogramm bzw. PAP – die Algorithmenstrukturen sollen in der verwendeten Programmiersprache implementiert werden
<ul style="list-style-type: none"> – Begriff der Variablen in der Informatik – Analyse, Darstellung und Implementierung von einfachen Algorithmen 	<ul style="list-style-type: none"> – Name, Typ, Wert, Speicheradresse – Trockentest – Tausch von Variableninhalten – allgemeine quadratische Gleichung mittels Lösungsformel – Wachstumsfunktionen – weitere Beispiele aus anderen Fächern
<p>Komponenten- und ereignisorientierte Softwareentwicklung</p> <ul style="list-style-type: none"> – Datenmodell <ul style="list-style-type: none"> konkrete Datentypen – visuelle Darstellung <ul style="list-style-type: none"> Ein- und Ausgabekonzept – Methoden/Algorithmenimplementierung <ul style="list-style-type: none"> • Anweisungen • Unterprogramme und Parameterübergabe 	<ul style="list-style-type: none"> – Nutzung von visuellen Entwicklungsumgebungen – Zugang über die vordefinierten visuellen Objektklassen – Klasse, Objekt, Eigenschaft, Methode, Ereignis – Entwicklung einer Benutzeroberfläche – Referenz- und Wertübergabe
<ul style="list-style-type: none"> – grafische Basisroutinen <ul style="list-style-type: none"> • Aufruf von Grafikroutinen des gewählten Grafikpaketes in der verwendeten Programmiersprache 	<ul style="list-style-type: none"> – Veranschaulichung von Daten

Kurshalbjahr 11/2

Thema: Strukturierte Datentypen

ZRW: 10 Std.

Vorbemerkungen/Qualifikationen

Im Rahmen dieses Themas werden Datentypen zur Verwaltung von komplexen Datenstrukturen eingeführt. Hierzu werden Methoden zur permanenten Verwaltung auf externen Datenträgern vermittelt.

Die Schülerinnen und Schüler

- kennen Strukturen zur Verwaltung komplexer Daten und zur rechnerinternen Repräsentation,
- können Daten in Dateien verschiedenen Typs verwalten.

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none">– Strukturierte Datentypen und ihre Implementierung<ul style="list-style-type: none">• Reihung (Felder)• Datenverbund (Datensatz)• Dateien	<ul style="list-style-type: none">– indizierter Zugriff auf Felder– Beispiel: Minimum, Maximum, Mittelwert einer Messwertfolge– Erstellen von Datensätzen aus einem Applikationsgebiet
<ul style="list-style-type: none">– Operationen auf Dateien<ul style="list-style-type: none">• Speichern und Öffnen von Dateien• Suchen und Editieren von Daten• Fehlerinterpretation	<ul style="list-style-type: none">– Zusammenwirken von Betriebssystem und Dateizugriffskonzept– Verwaltung von Datensätzen aus einem Applikationsgebiet– Navigieren in einer Datei

Vorbemerkungen/Qualifikationen

Die im Umgang mit Algorithmierung und Programmiersprache erlernten Fähigkeiten und Fertigkeiten werden an einem Beispiel gefestigt und dabei wird ein vollständiger Softwareentwicklungszyklus (Software life cycle) durchlaufen. Die Komplexität des Beispiels ist so zu wählen, dass eine Realisierung in der zur Verfügung stehenden Zeit möglich ist.

Die Schülerinnen und Schüler

- sind in der Lage, aus einer Aufgabenstellung heraus die Ziele und Anforderungen für ein Informatiksystem abzuleiten,
- erkennen das Informatiksystem als Abbild (Modell) eines realen Systems,
- können die erforderlichen Daten bestimmen und in der geforderten Genauigkeit erfassen,
- können unter Anleitung ein komplexes Problem in Teilprobleme zerlegen und die Funktionen dieser Bausteine und ihre Schnittstellen implementieren sowie die Teillösungen zu einer Gesamtlösung zusammenfassen,
- sind in der Lage, eine projektbegleitende Dokumentation zu erstellen und diese darzulegen,
- begreifen den Softwarelebenszyklus als iterativen Prozess und erkennen dessen wirtschaftliche Bedeutung.

Inhalte	Hinweise zum Unterricht
Phasen des Softwareentwicklungszyklus – Problemformulierung und Problemanalyse Ergebnis: Anforderungsdefinition	– vollständiges und eindeutiges Erfassen des realen Systems bzw. Problems und aller wichtigen Umgebungsbedingungen (Systemeigenschaften, Pflichtenheft)
– Entwurf <ul style="list-style-type: none"> • Ergebnis: Modellbildung • Ergebnis: Spezifikation 	– Beschreibung des zu erzeugenden Systems mit informalen und formalen Mitteln und Erzeugung eines abstrakten Modells – Zerlegung des komplexen Gesamtsystems in überschaubare Teilprobleme (Module) und ihre Bearbeitung in Gruppen – Zusammenführen dieser Teilprobleme durch Beschreibung der Modulfunktionen und einer eindeutigen Schnittstellenbeschreibung sowie Aufzeigen vorhandener Modulabhängigkeiten
– Implementierung Ergebnis: lauffähiges dokumentiertes Programm	– Programm entspricht dem Ein-/Ausgabeverhalten der Anforderungsdefinition
– Funktions- und Leistungsüberprüfung <ul style="list-style-type: none"> • Ergebnisanalyse und Bewertung • Ergebnis: modifiziertes Programm 	– Verifikation und Testen der Einzelmodule – Integrationstest – Installationstest – Laufzeit- und Speicherverhalten
– Installation und Abnahme	– Ergebnis: fertiges Softwareprodukt entsprechend der Anforderungsdefinition
– Dokumentation	– die Dokumentation ist über den gesamten Zeitraum dieser Unterrichtseinheit zu führen – die Dokumentation kann als Hilfesystem erstellt werden

Kurshalbjahr 12/1

Wahlthema 1: Modellbildung und Simulation

ZRW: 26 Std.

Vorbemerkungen/Qualifikationen

Im Rahmen dieses Themas werden die Grundlagen der Modellbildung und Simulation vermittelt. Aufbauend auf den in der Qualifizierungsphase bereits gewonnenen Erkenntnissen aus dem Bereich der Softwareentwicklung werden diese am Beispiel der Entwicklung von Simulationsstudien gefestigt.

Die Schülerinnen und Schüler

- können geeignete Abstraktionstechniken zur Erstellung von Simulationsmodellen anwenden,
- erkennen, dass ein Modell einen Ausschnitt aus der realen Welt entsprechend des gewählten Abstraktionsniveaus beschreibt,
- erfahren den Erkenntnisgewinn durch Analogieschlussverfahren am Beispiel von Simulationsexperimenten mit diesen Modellen,
- erlernen den Umgang mit einer Simulationssprache bzw. einem Simulationswerkzeug und sind in der Lage, einfache Computermodelle selbst zu implementieren,
- erlernen Methoden der Visualisierung von Resultaten,
- sind in der Lage, Resultate eines Simulationslaufes und einer Simulationsstudie verbal zu interpretieren und kritisch zu bewerten,
- nutzen die Kenntnisse des wissenschaftspropädeutischen Arbeitens aus anderen Unterrichtsfächern.

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none">– Grundbegriffe<ul style="list-style-type: none">• Modellbegriff• Simulationsmodell• reales System	<ul style="list-style-type: none">– Klassifikation von Modellen nach unterschiedlichen Kriterien

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none"> – Bediensystem als Simulationsmodell <ul style="list-style-type: none"> • Auswahl und Beschreibung eines Bediensystems • Definition des Systembegriffs • Methoden von Abstraktion und Reduktion • Erfassung und Aufbereitung empirischer Daten 	<ul style="list-style-type: none"> – Auswahl eines Bediensystems aus dem Erfahrungshorizont der Schüler, zum Beispiel Dienstleistungsbetrieb, Supermarkt, Tankstelle – Wachstumsmodelle als Beispiel für kontinuierliche Modelle – verbale Beschreibung des realen Systems – Auftrag zur Datenerhebung – Definition des Ziels der Simulationsstudie, zum Beispiel Untersuchung von Strategien
<ul style="list-style-type: none"> • Rollenspiel des zu untersuchenden Systems • formale Ablaufbeschreibung • Simulation als Prozess • Zeitraffung und Zeitstreckung im Modell • Begriff der Validierung 	<ul style="list-style-type: none"> – Entwicklung und Vertiefung des Systemverständnisses – Validierung des abstrakten Modells – Erzeugen einer Ablaufskizze
<ul style="list-style-type: none"> – Modellimplementierung <ul style="list-style-type: none"> • Anweisungen, Datentypen und Abarbeitungsstrukturen einer Simulationssprache • Abbildung zeitparalleler Abläufe • Verwaltung der Simulationsuhr • Implementierung des gewählten Modells in der Simulationssprache • Verifikation und Validierung des Computermodells 	
<ul style="list-style-type: none"> – Simulationsexperimente <ul style="list-style-type: none"> • Interpretation abstrakter Resultate • Simulation als iterativer Prozess 	<ul style="list-style-type: none"> – Ableitung von systematischen Experimentierstrategien – Durchführen von Experimentserien mit unterschiedlichen Zufallszahlen
<ul style="list-style-type: none"> – Modifikation des Modells <ul style="list-style-type: none"> • Vertiefen der Anweisungen der Simulationssprache • Modifikation des Modells 	<ul style="list-style-type: none"> – Änderung der Priorität von Agenten (Forderungen) – Einführung erweiternder Sprach-elemente

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none"> – Visualisierung der Resultate <ul style="list-style-type: none"> • Visualisierungs- und Präsentationstechniken • Nutzung von Schnittstellen zur Standardsoftware 	<ul style="list-style-type: none"> – Aufbereitung und Bewertung der Resultatdaten – Anwendung von Standardsoftware zur Präsentation – Nutzung von Animationssystemen
<ul style="list-style-type: none"> – Modellkritik <ul style="list-style-type: none"> • Validierungstechniken • Ergebnisanalyse 	<ul style="list-style-type: none"> – abschließende Betrachtung und Wertung des Modellbildungsprozesses – kritische Betrachtung des Modellbildungsprozesses – Simulationsresultate als Hilfsmittel zur Entscheidungsunterstützung anwenden

Wahlthema 2: Analyse und Design eines Informatiksystems

ZRW: 26 Std.

Vorbemerkungen/Qualifikationen

In diesem Thema werden wesentliche Grundkenntnisse im Bereich der Datenerfassung, der Datenaufbereitung, der Datenverarbeitung und der Steuerung eines Informatiksystems an Hand eines Modells vermittelt. Das gewählte Modell sollte sich an einem realen System aus der Umwelt orientieren. Hierfür geeignet sind funktionale Modelle aus dem Bereich der Mess-, Steuerungs- und Regelungstechnik.

Die Schülerinnen und Schüler

- kennen Möglichkeiten der Datenerfassung und Datenaufbereitung bezogen auf das gewählte Modell,
- gelangen, ausgehend von der Problemstellung über die Modellierung des Systems zu problemadäquaten Datenstrukturen und Algorithmen,
- können diese in einer hierfür geeigneten Sprache implementieren,
- verstehen das Modell als eine Einheit aus mechanischer Funktionalität, Hard- und Software,
- kennen Sensoren und können deren Signale erfassen und auswerten,
- kennen Aktoren und können diese zur Modellsteuerung nutzen,
- können das gewählte Modell steuern.

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none"> - Automatisierung und Einsatz von Robotern in der Industrie und die damit verbundenen gesellschaftlichen Auswirkungen 	<ul style="list-style-type: none"> - interdisziplinäre Aspekte (Arbeits- und Sozialwissenschaften, Maschinenbau, Elektro- und Kommunikationstechnik) - fachübergreifende Aspekte (physikalische Wirkprinzipien)
<ul style="list-style-type: none"> - Steuerung - Regelung 	<ul style="list-style-type: none"> - Darstellen und Erläutern einer Steuerkette - Darstellen und Erläutern eines Regelkreises
<ul style="list-style-type: none"> - elementare Algorithmen zum Steuern und Regeln und ihre Implementierung 	<ul style="list-style-type: none"> - Kontrollstrukturen - Steuerungsalgorithmen
<ul style="list-style-type: none"> - Digitalisierung analoger Größen sowie die Umwandlung digitaler in analoge Signale 	<ul style="list-style-type: none"> - AD- und DA-Wandler - Skalierung analoger Messwerte - Vergleich der Effizienz von Steuerungsalgorithmen
<ul style="list-style-type: none"> - Messwernerfassung und -verarbeitung Abfrage von Sensoren - Prozessbeeinflussung Ansteuern von Aktoren 	<ul style="list-style-type: none"> - Einfluss von Aktoren auf die Gerätetechnik - Kopplungen (galvanisch, optisch, magnetisch)
<ul style="list-style-type: none"> - Modellierung einfacher Steuerungsprozesse und ihre Implementierung 	
<ul style="list-style-type: none"> - Gestaltung von Benutzungsschnittstellen 	<ul style="list-style-type: none"> - Funktionalität, Anwenderfreundlichkeit, Robustheit („Absturzsicherheit“) von Bedienoberflächen
<ul style="list-style-type: none"> - Einblick in objektorientierte Arbeitsweisen 	<ul style="list-style-type: none"> - Entwurf von Klassen und Methoden - Wiederverwendbarkeit der Software - Zerlegung der Aufgabe in Teilprobleme (Softwareentwicklung in der Gruppe)

Wahlthema 3: Computergrafik

ZRW: 26 Std.

Vorbemerkungen/Qualifikationen

Die Visualisierung von Daten und Prozessen gewinnt immer mehr an Bedeutung. In diesem Wahlthema soll, ausgehend von technischen und mathematischen Grundlagen, das prinzipielle Vorgehen beim Erstellen von virtuellen Welten an geeigneten Beispielen nachempfunden werden.

Die Schülerinnen und Schüler

- kennen Anwendungsgebiete der Computergrafik,
- können Raster- und Vektorgrafiken unterscheiden,
- kennen ein additives und ein subtraktives Farbmodell und deren Anwendungsgebiete,
- kennen 2D- und 3D-Transformationen und können 2D-Transformationen implementieren,
- können einfache virtuelle Welten modellieren und diese mittels eines geeigneten Grafiksystems implementieren,
- berücksichtigen im Rahmen dieses Faches ästhetische Aspekte bei der Gestaltung.

Inhalte	Hinweise zum Unterricht
Computergrafik <ul style="list-style-type: none"> – geschichtliche Entwicklung – Anwendungsgebiete <ul style="list-style-type: none"> • Bilderzeugung • Bildverarbeitung 	<ul style="list-style-type: none"> – Beispiele für Bilderzeugungssysteme <ul style="list-style-type: none"> • CAD-Systeme • Computerspiele • Schriftzeichen – Beispiele für Bildverarbeitungssysteme <ul style="list-style-type: none"> • Bildverbesserung • Mustererkennung • Szenenerkennung • optische Bilderfassung
Rastergrafik und Vektorgrafik <ul style="list-style-type: none"> – Begriffe – Vergleich von Raster- und Vektorgrafik – Gerätetechnik für Ein- und Ausgabe von Raster- bzw. Vektorgrafik – Grafikdateiformate – Farbmodelle 	<ul style="list-style-type: none"> – BMP, GIF, JPG, PNG, CDR, WMF – RGB, CMYK
<ul style="list-style-type: none"> – Koordinatensysteme – 2D - Transformationen <ul style="list-style-type: none"> • Translation • Skalierung • Rotation • Komposition • Implementierung in der gewählten Programmiersprache 	<ul style="list-style-type: none"> – Verschiebung – Dehnung/Stauchung – Drehung – Nacheinanderausführung – Erweiterung auf 3D-Transformationen
<ul style="list-style-type: none"> – Modellierung und Darstellung virtueller Welten – Behandlung einer geeigneten Sprache 	<ul style="list-style-type: none"> – ist in dem Umfang zu behandeln, wie es für das Projekt in 12/2 notwendig ist – VRML, OpenGL, Direkt3D, Java3D

Vorbemerkungen/Qualifikationen

Im Rahmen dieses Themas werden ausgewählte Datentypen ausführlich behandelt. Zur formalen Beschreibung werden abstrakte Datentypen verwendet. Die Implementierung erfolgt mit einer geeigneten Programmiersprache.

Die Schülerinnen und Schüler

- kennen den Begriff des abstrakten Datentyps,
- können auf der Basis von abstrakten Datentypen Grundfunktionen auf die dynamische Datenstruktur Liste beschreiben und implementieren,
- kennen die Spezialformen Stapel und Warteschlange.

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none"> – Begriff und Eigenschaften des abstrakten Datentyps (ADT) 	<ul style="list-style-type: none"> – Universalität – Geheimnisprinzip
<ul style="list-style-type: none"> – Prinzipien der Verkettung der ADT-Liste <ul style="list-style-type: none"> • einfach verkettete Liste • doppelt verkettete Liste • Ringliste 	<ul style="list-style-type: none"> – Implementierung durch dynamische Datenstrukturen oder Reihung (Felder)
<ul style="list-style-type: none"> – Implementierung der Grundfunktionen – Methoden auf Listen <ul style="list-style-type: none"> • Initialisierung • Einketten • Ausketten • Suchen • Speichern in Dateien • Laden aus Dateien 	<ul style="list-style-type: none"> – Realisierung an einem komplexen Beispiel
<ul style="list-style-type: none"> – Anwendung von Listen <ul style="list-style-type: none"> • Stack (Stapel) • Queue (Warteschlange) 	<ul style="list-style-type: none"> – LIFO-Prinzip – FIFO-Prinzip – Nachbildung eines Bediensystemmodells

Vorbemerkungen/Qualifikationen

Im Rahmen dieses Themas werden Iteration und Rekursion als fundamentale Verfahren zur Beschreibung von Zyklen behandelt. Es werden Such- und Sortieralgorithmen als Basisalgorithmen der Informatik analysiert, implementiert und bewertet.

Die Schülerinnen und Schüler

- kennen unterschiedliche Formen der Gestaltung von Zyklen,
- können iterative und rekursive Algorithmen erkennen und entwerfen,
- kennen unterschiedliche Methoden der Terminierung der Zyklenform,
- kennen elementare und rekursive Sortieralgorithmen, können diese implementieren und bezüglich ihrer Zeitkomplexität bewerten,
- kennen Suchalgorithmen und können diese implementieren.

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none"> – Iteration und Rekursion <ul style="list-style-type: none"> • Vergleich beider Verfahren • Terminierungsbedingungen • Beispielalgorithmen entwerfen und implementieren 	<ul style="list-style-type: none"> – Vergleich der Verfahren an Hand der binären Suche – Fakultät, Fibonacci-Folgen, Turm von Hanoi
<ul style="list-style-type: none"> – Elementare Sortieralgorithmen <ul style="list-style-type: none"> • Selectionsort • Insertionsort • Bubblesort 	<ul style="list-style-type: none"> – Sortieren von Feldern mit Zahlen – Sortieren von Feldern mit Datensätzen – Sortieren von Feldern nach verschiedenen Elementen des Datensatzes
<ul style="list-style-type: none"> – Divide and Conquer Sortierverfahren Quicksort 	
<ul style="list-style-type: none"> – Suchverfahren <ul style="list-style-type: none"> • sequentielles Suchen in Dateien • iteratives und rekursives Suchen am Beispiel des binären Suchens 	<ul style="list-style-type: none"> – dynamische Reihung (Felder)
<ul style="list-style-type: none"> – Zeitkomplexität von Algorithmen am Beispiel der zuvor behandelten Algorithmen als eine Möglichkeit der Bewertung von Algorithmen – Grenzen der Berechenbarkeit von Algorithmen 	<ul style="list-style-type: none"> – Abschätzen der Zeitkomplexität – Messen der benötigten Zeit – siehe Thema „Informatik und Gesellschaft“

Vorbemerkungen/Qualifikationen

Ausgehend von der Arbeitsweise eines realen Automaten aus der Erfahrungswelt der Schülerinnen und Schüler und dessen abstrakter Beschreibung sind Modelle endlicher Automaten zu beschreiben und zu untersuchen sowie formale Sprachen zu behandeln.

Die Schülerinnen und Schüler

- kennen Einsatzfelder für Automaten aus ihrer Erfahrungswelt,
- können einen endlichen Automaten definieren,
- können endliche Automaten mit unterschiedlichen Mitteln beschreiben,
- kennen die Eigenschaften von speziellen endlichen Automaten (Akzeptor/Parser),
- wissen, was eine formale Sprache ist,
- können den Inhalt der Begriffe Terminalsymbol, Nichtterminalsymbol, Alphabet, Wort, Grammatik, Syntax, Syntaxdiagramme, Scanner, Parser und Interpreter im Zusammenhang mit formalen Sprachen erläutern,
- können Worte W_i einer formalen Sprache $L(G)$ – aus der Kenntnis ihrer Grammatik G – erzeugen oder entscheiden, ob ein Wort W aus Terminalsymbolen eines Alphabetes T zu einer vorgegebenen Sprache $L(G)$ gehört oder nicht,
- können Sprachen $L(G)$ zur Beschreibung von einfachen Aufgaben entwickeln und deren Syntax sowie Semantik beschreiben.

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none"> – Automaten <ul style="list-style-type: none"> • Automatenbegriff • Einsatzfelder von Automaten • Beispiel für einen endlichen Automaten • Grundmodell von Automaten 	<ul style="list-style-type: none"> – Der Automatenbegriff wird aus der Behandlung realer Automaten durch Modellbildung entwickelt.
<ul style="list-style-type: none"> – Modell endlicher Automat <ul style="list-style-type: none"> • Definition endlicher Automat (Eingabemenge, Ausgabemenge, Menge der definierten Automatenzustände, Ausgabefunktion und Zustandsüberföhrungsfunktion) • Beschreibung endlicher Automaten • Simulation endlicher Automaten mit Computerprogrammen 	<ul style="list-style-type: none"> – Beschreibungsmittel für das Modell endlicher Automat <ul style="list-style-type: none"> • formale Beschreibung wesentlicher Modellparameter • Automatentafeln

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none"> – Automaten ohne Ausgabe als erkennende Automaten – Definition der besonderen Zustände als Fehlerzustand oder Zielzustand 	<ul style="list-style-type: none"> – Automaten nehmen besondere Zustände ein, wenn eine bestimmte Folge von Eingaben (Wort) erkannt und akzeptiert wird.
<ul style="list-style-type: none"> – Parser als spezielle erkennende Automaten 	<ul style="list-style-type: none"> – Parser als endliche Automaten, bei denen die Folge von Eingabezeichen Worte sind
<ul style="list-style-type: none"> – formale Sprachen Begriffe und Definitionen (Sprache $L(G)$, Grammatik G, Nichtterminal, Terminalsymbol, Alphabet T als Menge der Terminalsymbole) – Erzeugen eines Wortes W bei gegebener Grammatik G – Nachweis, ob vorgegebene Worte zur Sprache $L(G)$ gehören 	<ul style="list-style-type: none"> – Zur Beschreibung einer (formalen) Sprache werden neben der abstrakten Beschreibung vor allem Syntaxdiagramme eingesetzt.
<ul style="list-style-type: none"> – Entwurf und Definition (Syntax und Semantik) einer nichtformalen Sprache $L(G)$, Syntax 	<ul style="list-style-type: none"> – Bei den nichtformalen Sprachen werden den Worten W_i der Sprache Bedeutungen zugeordnet und es gilt: Wenn eine Aufgabe mit Hilfe einer Sprache $L(G)$ vollständig beschrieben werden kann, dann ist die Aufgabe prinzipiell gelöst.
<ul style="list-style-type: none"> – Lösung von Aufgaben durch die Beschreibung dieser Aufgaben in einer selbstentwickelten Programmiersprache 	<ul style="list-style-type: none"> – Einsatz von nichtformalen Sprachen <ul style="list-style-type: none"> – auch selbstdefinierter – zur Lösung einfacher Aufgaben <ul style="list-style-type: none"> • Sprache zur Programmierung einfacher Assemblerprogramme • Sprache zur Umsetzung graphischer Basisalgorithmen

Wahlthema 7: Kryptologie

ZRW: 26 Std.

Vorbemerkungen/Qualifikationen

In diesem Thema wird ausgehend von historischen Beispielen die gesellschaftliche Bedeutung und Notwendigkeit kryptologischer Verfahren dargestellt. Voraussetzung hierzu sind die Fähigkeiten und Fertigkeiten im Umgang mit der verwendeten Programmiersprache. Bei der mathematischen Beschreibung und Formalisierung sind bei einigen Verfahren Grenzen durch die Schulmathematik gesetzt.

Die Schülerinnen und Schüler

- kennen Beispiele aus der historischen Entwicklung und wissen um die modernen gesellschaftlichen Aspekte der Kryptologie,
- kennen wesentliche Aufgaben der Kryptologie in der Vergangenheit und Gegenwart,
- kennen verschiedene Verfahren der Kryptographie und der Kryptoanalyse und können diese implementieren,
- kennen Vor- und Nachteile ausgewählter Chiffrierverfahren,
- können Klartexte in der gewählten Programmiersprache nach verschiedenen Algorithmen chiffrieren,
- können fremde, verschlüsselte Texte analysieren und in der verwendeten Programmiersprache dechiffrieren.

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none"> – Grundlagen <ul style="list-style-type: none"> • Begriffsbestimmung und Bestandteile (Kryptographie, Kryptoanalyse, Steganographie) der Kryptologie • Entwicklung der Kryptologie 	<ul style="list-style-type: none"> – Aufzeigen wichtiger Meilensteine der Kryptologie und deren grundlegender Wirkprinzipien Beispiele: Skytale von Sparta, Cäsar-Scheibe, Chiffrierzylinder und Rotormaschinen, ENIGMA, Chip- und EC-Karten
<ul style="list-style-type: none"> – gesellschaftliche und technische Notwendigkeit der Verschlüsselung von Daten <ul style="list-style-type: none"> • Datenschutzgesetze und ihre Geltungsbereiche • Grundprobleme von Datenschutz und Datensicherheit in offenen und verteilten Informatiksystemen • Sicherheitsanforderungen in offenen und verteilten Informatiksystemen 	<ul style="list-style-type: none"> – Ausgehend von dem im Schuljahrgang 10 erarbeiteten Wissen und eigenen Erfahrungen der Lernenden bieten sich, untersetzt mit Beispielen, folgende kryptographische Schutzmaßnahmen an: <ul style="list-style-type: none"> • Sicherung der Anonymität bei der Datenübertragung • verschlüsselte Kommunikation • digitale Signaturen • Authentisierung von Kommunikationspartnern • Sicherung der Integrität und der Verbindlichkeit

Inhalte	Hinweise zum Unterricht
<p>Grundlagen der Kryptologie</p> <ul style="list-style-type: none"> – Algorithmen und Schlüssel – Funktionsprinzipien von Chiffrieralgorithmen – Kryptoanalyse <ul style="list-style-type: none"> • Sicherheit von Chiffrieralgorithmen • Kryptoanalyse und deren Grenzen – Verbergen von Informationen mit Hilfe der Steganographie 	<ul style="list-style-type: none"> – Hilfsmittel zur Verschlüsselung (zum Beispiel Muster und Häufigkeiten in einer Sprache, Leerzeichen, Koinzidenzindex Kappa) – Monoalphabetische Chiffrierung und Polyalphabetische Chiffrierung – Transpositionschiffren, Substitutionschiffren und deren Kombination – symmetrische und asymmetrische Algorithmen – es sollte den Schülerinnen und Schülern verdeutlicht werden, dass Hilfsmittel zur Verschlüsselung gleichzeitig beim Dechiffrieren eingesetzt werden – Hilfsmittel zur Kryptoanalyse (Kasiski-Test, Friedman-Test, Häufigkeitsanalyse)
<ul style="list-style-type: none"> – praktische Grundlagen der Kryptographie sowie der Kryptoanalyse und Implementierung ausgewählter Verfahren in der gewählten Programmiersprache 	<ul style="list-style-type: none"> – Erzeugung von Zufallszahlenfolgen – Algorithmen zur Primfaktorenzerlegung – Chiffrier- und Dechiffrieralgorithmen – Lösen von Zahlenkreuzworträtseln

Wahlthema 8: Einsatz von Datenbanken zur dynamischen Webseitengenerierung

ZRW: 26 Std.

Vorbemerkungen/Qualifikationen

Im Rahmen dieses Themas werden Lösungen für den Datenbankeinsatz erstellt. Die Implementierung der Abfragen in eine Webseite ist ebenso Schwerpunkt dieses Themas wie die Pflege und Verwaltung von Datenbanken.

Die Schülerinnen und Schüler

- kennen Beispiele für dynamische Webseiten, deren Einsatzgründe, Möglichkeiten und Grenzen,

- kennen die einzelnen Komponenten eines Formulars und dessen Ereignisse, die für den Aufbau dynamischer Webseiten notwendig sind,
- können in einer (X)HTML-Seite Formulare einbinden und mit der verwendeten Programmiersprache mit der Benutzerin oder dem Benutzer in Interaktion treten,
- können eine Datenbank erstellen,
- können Abfragen mit einer hierfür geeigneten Sprache (SQL) in dieser Datenbank mit geeigneten Tools vornehmen,
- kennen die Grundlagen des Verbindungsaufbaus zwischen der Datenbank und der Webseite (z. B. ODBC, JDBC) und können vorhandene Treiber implementieren,
- können die Abfrageergebnisse mit einer hierfür geeigneten Sprache (z. B. PHP, ASP, JSP, Java-Servlets) in eine Webseite implementieren.

Inhalte	Hinweise zum Unterricht
<ul style="list-style-type: none"> – Generierung einfacher dynamischer Webseiten (Formulare, Ereignisse) 	<ul style="list-style-type: none"> – Einsatz von JavaScript, VBScript, CSS – verschiedene Methoden (post, get), Parameterübergabe
<ul style="list-style-type: none"> – typische Einsatzbereiche für Datenbanken – Entwurf einer Datenbank 	<ul style="list-style-type: none"> – Datenschutz und Datenmissbrauch – MySQL, Access – die Administration einer Datenbank sollte der Lehrkraft oder dem Systemadministrator obliegen – Beachtung der Grundsätze der Datenbankentwicklung – Festlegen des Datentyps der definierten Felder – Festlegen der Primär- und Fremdschlüssel – Indizierung
<ul style="list-style-type: none"> – Abfragen in Datenbanken – Grundlagen des Verbindungsaufbaus 	<ul style="list-style-type: none"> – Abfragetechniken – Auswahl-, Update- und Löschanfragen – Programmieren der Abfragen und Test mit Hilfe geeigneter Tools – Erläuterung und Beispiel
<ul style="list-style-type: none"> – Implementierung der Datenbankabfrage in eine Webseite 	<ul style="list-style-type: none"> – Erstellung der Webseite, Auswertung der Formularelemente, Erstellung der Abfrage, Darstellung der Ergebnisse in der folgenden (dynamisch erstellten) Seite

Kurshalbjahr 12/2

Thema: Projektarbeit zur Softwareentwicklung

ZRW: 16 Std.

Vorbemerkungen/Qualifikationen

Im Rahmen des Projektes sind Facharbeiten anzufertigen. Die Themen sind in der Regel aus dem im Kurshalbjahr 12/1 behandelten Wahlthema abzuleiten und sollen Möglichkeiten bieten, die im Thema „Informatisches Modellieren“ erlernten Phasen der Softwareentwicklung anzuwenden.

Die Schülerinnen und Schüler

- sind in der Lage, ein Projektthema in Bezug auf Inhalt und Zeitvorgaben sinnvoll einzugrenzen,
- können ein Projekt unter Anleitung in Teilprojekte zerlegen und die erwarteten Arbeitsergebnisse der Beteiligten genau darstellen,
- können Teilprojekte eigenständig bearbeiten und die Schnittstellen der Softwaremodule formal beschreiben,
- sind in der Lage, in geeigneter Form den Nachweis ihrer Teilergebnisse am Projektthema zu erbringen,
- können im fachlichen Dialog aller Beteiligten die Ergebnisse der Teilprojekte zu einem Gesamtergebnis zusammenfassen und in einer Facharbeit beschreiben,
- erkennen, dass Kooperation und Kommunikation in der Gruppe zu Synergieeffekten führen,
- können die Ergebnisse der Projektarbeit im Rahmen des Unterrichts präsentieren.

Der Umfang der Projekte orientiert sich an der zur Verfügung stehenden Zeit und an der Leistungsfähigkeit der Gruppen. Die Ergebnisse der Projektarbeit sind im Rahmen des Unterrichts vorzustellen. Um eine realistische Projektbewertung zu ermöglichen, sind die Projektleitung und Verantwortlichkeiten in der Gruppe festzulegen. Dokumentationen sind als Nachweis für die Eigenleistung von jedem Gruppenmitglied zu führen. Die Festlegung der Projektthemen sowie die Gruppeneinteilung sollte spätestens im zweiten Drittel des Kurshalbjahres 12/1 erfolgen. Grundlage der Leistungsbewertung sind die Facharbeit und die Präsentation.