

Arbeitsfassung vom 19.12.2014

Kerncurriculum für

das Gymnasium – gymnasiale Oberstufe
die Gesamtschule – gymnasiale Oberstufe
das Abendgymnasium
das Kolleg

Informatik



Niedersachsen

An der Erarbeitung des Kerncurriculums für das Unterrichtsfach Informatik in der gymnasialen Oberstufe waren die nachstehend genannten Personen beteiligt:

Das Kerncurriculum kann als "PDF-Datei" vom Niedersächsischen Bildungsserver (NIBIS) (<http://www.cuvo.nibis.de>) heruntergeladen werden.

Inhalt	Seite	
1	Bildungsbeitrag des Faches Informatik	5
2	Kompetenzorientierter Unterricht	7
2.1	Kompetenzbereiche	7
2.2	Kompetenzentwicklung	8
2.3	Innere Differenzierung	8
3	Erwartete Kompetenzen	10
3.1	Prozessbezogene Kompetenzen	10
3.2	Inhaltsbezogene Kompetenzen	12
3.3	Zusammenführung von Kompetenzen	14
	Lernfeld „Algorithmen und Datenstrukturen“	15
	Lernfeld „Information und Daten“	17
	Lernfeld „Automaten und Sprachen“	19
3.4	Projektorientierter Unterricht	20
4	Leistungsfeststellung und Leistungsbewertung	21
5	Aufgaben der Fachkonferenz	23
Anhang		
A1	Liste der Operatoren für das Fach Informatik	24
A2	Operationen der Klassen <i>Dynamische Reihung</i> , <i>Stapel</i> , <i>Schlange</i> und <i>Binärbaum</i>	25
A3	SQL-Syntax	27
A4	Auflistung der Kompetenzen	28

1 Bildungsbeitrag des Faches Informatik

In einem dem Bildungsauftrag verpflichteten Unterricht, wie er im Niedersächsischen Schulgesetz verankert ist, müssen zwei wesentliche Aspekte Berücksichtigung finden. Zum einen sollen Schülerinnen und Schüler befähigt werden, sich in einer dynamischen Gesellschaft unabhängig von ihrem Geschlecht oder sozioökonomischem Hintergrund selbstbestimmt und freiheitlich zu entwickeln. Zum anderen soll ihnen bewusst sein, dass sie durch ihr Handeln die Gesellschaft mitgestalten und dass sie für deren zukünftige Ausprägung Verantwortung übernehmen.

Die hohe Dynamik der Informationstechnologie sowie ihr stetig wachsender Einfluss auf gesellschaftliche und kulturelle Entwicklungen und die Lebenswirklichkeit erfordern einen selbstständigen, kompetenten und selbstbewussten Umgang mit ihr. Nur dadurch wird den Schülerinnen und Schülern ermöglicht, die gegenwärtigen und zukünftigen Auswirkungen dieser Technologien einzuschätzen. Sie werden in die Lage versetzt, aktiv, kreativ und gestaltend an der gesellschaftlichen Entwicklung teilzunehmen. Dies gilt insbesondere vor dem Hintergrund, dass in einer Informationsgesellschaft Erfolg in zunehmenden Maße nicht nur davon abhängt, was oder wie viel man weiß, sondern auch davon, ob man in der Lage ist, kreativ zu denken und zu handeln. Medienkompetenz, wie sie der Informatikunterricht in besonderer Weise vermittelt, hat damit in der heutigen Zeit eine ähnliche Bedeutung wie Lesen, Schreiben und Rechnen. Sie ist Voraussetzung für die Teilhabe an der Informations- und Wissensgesellschaft sowie an demokratischen Prozessen der Meinungsbildung.

In allen Berufen und Wissenschaften haben sich die Arbeitsweisen durch die Methoden der Informatik und den Einsatz informatischer Systeme grundlegend verändert. Dies erfordert ein vertieftes Verständnis informatischer Zusammenhänge, um über das reine Anwenden hinaus, kreativ, reflektiert und zielgerichtet mit Informatiksystemen arbeiten zu können. Die dazu notwendigen Verfahren und Kenntnisse werden insbesondere durch den Informatikunterricht vermittelt.

In der Informatik werden komplexe Probleme in kleinere Teilprobleme zerlegt. Dieses Prinzip erfordert es, im Rahmen von Teamarbeit klare Schnittstellen zu definieren und sich an Vereinbarungen zu halten, um zielorientiert zu arbeiten. In projektartigen Arbeitsphasen durchlaufen die Schülerinnen und Schüler einen schöpferischen, kreativen Prozess, an dessen Ende ein Produkt steht. Sie erleben sich dabei als selbstwirksam und konstruktiv.

Grundlegende informatische Methoden wie das Strukturieren, die systematische Analyse komplexer Systeme und das Modellieren fördern und fordern die Abstraktionsfähigkeit, sowie kreatives und strukturiertes Denken. Die Implementierung von Modellen und Lösungsideen als Programm oder technische Lösung macht sie überprüfbar, erlebbar und anfassbar. Die konkrete Umsetzung als Produkt ermöglicht es, Modelle und Verfahren auf der Basis vorgegebener Funktionalitäten zu beurteilen. Dies fördert die Fähigkeit, sich konstruktiv und kritisch mit eigenen und fremden Ideen auseinanderzusetzen.

Das Fach Informatik thematisiert soziale, ökonomische, ökologische und politische Phänomene und Probleme der nachhaltigen Entwicklung und trägt dazu bei, wechselseitige Abhängigkeiten zu erkennen und Wertmaßstäbe für eigenes Handeln sowie ein Verständnis für gesellschaftliche

Entscheidungen zu entwickeln. Mit dem Erwerb spezifischer Kompetenzen wird im Unterricht des Faches Informatik der Bezug zu verschiedenen Berufsfeldern hergestellt. Die Schule ermöglicht es damit den Schülerinnen und Schülern, Vorstellungen über Berufe und über eigene Berufswünsche zu entwickeln, die über eine schulische Ausbildung, eine betriebliche Ausbildung, eine Ausbildung im dualen System oder über ein Studium zu erreichen sind. Der Fachunterricht leistet somit auch einen Beitrag zur Berufsorientierung, ggf. zur Entscheidung für einen Beruf.

2 Kompetenzorientierter Unterricht

Im Kerncurriculum des Faches Informatik werden die Zielsetzungen des Bildungsbeitrags durch verbindlich erwartete Lernergebnisse konkretisiert und als Kompetenzen formuliert. Dabei werden im Sinne eines Kerns die als grundlegend und unverzichtbar erachteten fachbezogenen Kenntnisse und Fertigkeiten vorgegeben.

Kompetenzen weisen folgende Merkmale auf:

- Sie zielen ab auf die erfolgreiche und verantwortungsvolle Bewältigung von Aufgaben und Problemstellungen.
- Sie verknüpfen Kenntnisse, Fertigkeiten und Fähigkeiten zu eigenem Handeln. Die Bewältigung von Aufgaben setzt gesichertes Wissen und die Beherrschung fachbezogener Verfahren voraus sowie die Einstellung und Bereitschaft, diese gezielt einzusetzen.
- Sie stellen eine Zielperspektive für längere Abschnitte des Lernprozesses dar.
- Sie sind für die persönliche Bildung und für die weitere schulische und berufliche Ausbildung von Bedeutung und ermöglichen anschlussfähiges Lernen.

Die erwarteten Kompetenzen werden in Kompetenzbereichen zusammengefasst, die das Fach strukturieren. Aufgabe des Unterrichts im Fach Informatik ist es, die Kompetenzentwicklung der Schülerinnen und Schüler anzuregen, zu unterstützen, zu fördern und langfristig zu sichern. Dies gilt auch für die fachübergreifenden Zielsetzungen der Persönlichkeitsbildung.

2.1 Kompetenzbereiche

Unterricht und Lernprozesse im Fach Informatik basieren auf dem Zusammenspiel von prozess- und inhaltsbezogenen Kompetenzbereichen. Die inhaltsbezogenen Kompetenzbereiche geben an, über welches fachliche Wissen die Schülerinnen und Schüler in den verschiedenen Gebieten der Informatik verfügen sollen. Die prozessbezogenen Kompetenzbereiche beziehen sich auf Verfahren und Arbeitsweisen, die typisch und wichtig für die Informatik sind. Diese Verfahren werden im Umgang mit den informatischen Inhalten erworben und sind daher eng mit den inhaltsbezogenen Kompetenzen verzahnt.

Die erwarteten Kompetenzen lassen sich den folgenden Kompetenzbereichen zuordnen:

prozessbezogene Kompetenzbereiche	inhaltsbezogene Kompetenzbereiche
<ul style="list-style-type: none">• Strukturieren und Modellieren• Algorithmisieren und Implementieren• Kooperieren und Kommunizieren• Kreatives Schaffen und Problemlösen	<ul style="list-style-type: none">• Daten und ihre Strukturierung• Operationen auf Daten und Algorithmen• Informatiksysteme und ihre zugrundeliegenden Modelle• Möglichkeiten und Grenzen informatischer Verfahren und Systeme

2.2 Kompetenzentwicklung

Die Kompetenzen werden im Kontext informatischer Sachverhalte erworben und geschult. Daher ist es wichtig, Lernsituationen zu schaffen, die das Interesse der Schülerinnen und Schüler wecken und ihnen einen Rahmen bieten, in dem sie selbstständig arbeiten und ihre Interessen entfalten können. Dies wird unter anderem durch den Einsatz adäquater Werkzeuge, die zum Experimentieren und Entdecken einladen, wie auch durch einen lebensweltlichen Bezug erreicht.

Im Informatikunterricht bietet sich ein projekt- und produktorientiertes Arbeiten an, das neben den durch die Lehrkraft organisierten Arbeitsphasen auch Phasen des selbstorganisierten Wissens- und Kompetenzerwerbs hat und vor dem Hintergrund eines konkreten Problems erfolgt. Das Ergebnis der Arbeit kann ein physisches Produkt, z. B. eine Schaltung oder eine technische Konstruktion, oder auch ein virtuelles Produkt sein, wie dies bei einer Präsentation, einem Programm oder einem Handlungsplan der Fall ist. Die Schülerinnen und Schüler durchlaufen in jedem Fall einen kreativen und individuellen Prozess, der zu einem Kompetenzerleben führt und ihre Selbstwirksamkeit stärkt.

Bei der Umsetzung von Projekten kommen verschiedenste Informatikwerkzeuge aus dem Bereich der Software und Hardware zum Einsatz, sowohl bei der Erarbeitung von Inhalten als auch bei der Dokumentation. Der Umgang mit diesen Werkzeugen wird dabei aus dem Kontext heraus und nicht als Selbstzweck gelernt. Durch die überwiegend projektorientierte Ausrichtung des Informatikunterrichts ist eine Zusammenarbeit und Absprache im Team erforderlich und fördert sachbezogene Dialoge, den konstruktiven Umgang mit Kritik sowie die Fähigkeit zu kooperieren.

Fehler sind ein wichtiger Teil des Lernprozesses. Geeignete Software- und Hardwarewerkzeuge ermöglichen es den Schülerinnen und Schülern, Fehler leichter zu erkennen und konstruktiv mit ihnen umzugehen. Weiterhin sind Lern- und Leistungssituationen im Unterricht klar zu unterscheiden, damit Schülerinnen und Schüler lernen, offen und produktiv mit ihren Fehlern umzugehen.

2.3 Innere Differenzierung

Aufgrund der unterschiedlichen Lernvoraussetzungen, der individuellen Begabungen, Fähigkeiten und Neigungen sowie des unterschiedlichen Lernverhaltens sind differenzierende Lernangebote und Lernanforderungen für den Erwerb der vorgegebenen Kompetenzen unverzichtbar. Innere Differenzierung als Grundprinzip in jedem Unterricht zielt auf die individuelle Förderung der Schülerinnen und Schüler ab. Dabei werden Aspekte wie z. B. Begabungen und motivationale Orientierungen, Geschlecht, Alter, sozialer, ökonomischer und kultureller Hintergrund, Leistungsfähigkeit und Sprachkompetenz berücksichtigt.

Aufbauend auf einer Diagnose der individuellen Lernvoraussetzungen unterscheiden sich die Lernangebote z. B. in ihrer Offenheit und Komplexität, dem Abstraktionsniveau, den Zugangsmöglichkeiten, den Schwerpunkten, den bereitgestellten Hilfen und der Bearbeitungszeit. Geeignete Aufgaben zum Kompetenzerwerb berücksichtigen immer das didaktische Konzept des

Unterrichtsfaches. Sie lassen vielfältige Lösungsansätze zu und regen die Kreativität von Schülerinnen und Schülern an.

Vor allem leistungsschwache Schülerinnen und Schüler brauchen zum Erwerb der verpflichtend erwarteten Kompetenzen des Kerncurriculums vielfältige Übungsangebote, um bereits Gelerntes angemessen zu festigen. Die Verknüpfung mit bereits Bekanntem und das Aufzeigen von Strukturen im gesamten Kontext des Unterrichtsthemas erleichtern das Lernen.

Für besonders leistungsstarke Schülerinnen und Schüler werden Lernangebote bereitgestellt, die deutlich über die als Kern an alle Schülerinnen und Schüler bereits gestellten Anforderungen hinausgehen und einen höheren Anspruch haben. Diese Angebote dienen der Vertiefung und Erweiterung und lassen komplexe Fragestellungen zu.

Innere Differenzierung fordert und fördert fächerübergreifende Kompetenzen wie das eigenverantwortliche, selbstständige Lernen und Arbeiten, die Kooperation und Kommunikation in der Lerngruppe sowie das Erlernen und Beherrschen wichtiger Lern- und Arbeitstechniken. Um den Schülerinnen und Schülern eine aktive Teilnahme am Unterricht zu ermöglichen, ist es vorteilhaft, sie in die Planung des Unterrichts einzubeziehen. Dadurch übernehmen sie Verantwortung für den eigenen Lernprozess. Ihre Selbstständigkeit wird durch das Bereitstellen vielfältiger Materialien und durch die Möglichkeit eigener Schwerpunktsetzungen gestärkt.

Um die Selbsteinschätzung der Schülerinnen und Schüler zu fördern, stellt die Lehrkraft ein hohes Maß an Transparenz über die Lernziele, die Verbesserungsmöglichkeiten und die Bewertungsmaßstäbe her. Individuelle Lernfortschritte werden wahrgenommen und den Lernenden regelmäßig zurückgespiegelt. Im Rahmen von Lernzielkontrollen gelten für alle Schülerinnen und Schüler einheitliche Bewertungsmaßstäbe.

3 Erwartete Kompetenzen

In den Abschnitten 3.1 und 3.2 werden die prozessbezogenen und die inhaltsbezogenen Kompetenzbereiche detailliert beschrieben und erläutert. Die Ausweisung einzelner Kompetenzen ist nicht als Beschreibung einer Abfolge von Unterrichtseinheiten aufzufassen. Auch wird dadurch keine Reihenfolge der Bearbeitung vorgegeben.

Der Abschnitt 3.3 führt die prozess- und die inhaltsbezogenen Kompetenzen in drei Lernfeldern zusammen. Diese bilden den Rahmen, aus dem ein schulinternes Fachcurriculum unter Berücksichtigung der schuleigenen Bedingungen erstellt werden kann.

3.1 Prozessbezogene Kompetenzen

Die prozessbezogenen Kompetenzen beziehen sich auf Verfahren, die von Schülerinnen und Schülern verstanden und beherrscht werden sollen. Diese befähigen die Schülerinnen und Schüler, ihr erworbenes Wissen anzuwenden und auf neue Situationen zu übertragen. Es bedarf dazu allgemeiner Kompetenzen, wie *Kooperieren und Kommunizieren*, wobei auf die Verwendung von Fachsprache und -symbolik zu achten ist. Darüber hinaus gibt es spezifische informatische Kompetenzbereiche, wie *Strukturieren und Modellieren* sowie *Algorithmisieren und Implementieren*, die die Schülerinnen und Schüler in die Lage versetzen, grundlegende informatische Konzepte zu erkennen. Die Kompetenzbereiche sind miteinander verknüpft. Im Bereich *Kreatives Schaffen und Problemlösen* sind Kompetenzen beschrieben, die vor allem für die Informatik typische Aspekte der Problemlösung und Produktentwicklung berücksichtigen.

Die prozessbezogenen Kompetenzen bilden eine Grundlage für die Schülerinnen und Schüler, um im Sinne eines lebenslangen Lernens dem stetigen Wandel der Informatiksysteme in allen Lebensbereichen gerecht zu werden. Diese Kompetenzen werden in der Regel in der Auseinandersetzung mit konkreten Inhalten erworben. Der Schwerpunkt darf dabei nicht im detaillierten Erlernen einer Programmiersprache liegen, sondern im Erkennen übergeordneter Konzepte, Strukturen und Strategien. Dies und die Vermittlung von Verfahren zum selbstständigen Lernen und von Reflexionsfähigkeit führen zur Nachhaltigkeit des Informatikunterrichts.

P1 Strukturieren und Modellieren

Ein zentrales Prinzip der Informatik ist es, reale Probleme oder Systeme zu analysieren, Funktionsweisen zu veranschaulichen und Modelle zu entwickeln. Dabei kommt dem Strukturieren in vielen Bereichen eine zentrale Rolle zu, beispielsweise beim Zerlegen von Problemstellungen in Teilprobleme oder dem Aufteilen von Arbeitsabläufen in Handlungsschritte. Dies fördert das kreative und strukturierte Denken der Schülerinnen und Schüler.

Das Ziel der Modellierung ist, die für die Problemstellung wesentlichen Elemente eines Systems aufzufinden und in ihrer Funktion und ihrem Zusammenwirken zu verdeutlichen, um das System geeignet abzubilden. Modellieren fördert die Fähigkeit der Schülerinnen und Schüler zur Abstraktion und versetzt sie in die Lage, übergeordnete Zusammenhänge zu erkennen.

Die Schülerinnen und Schüler ...

- (1) beschreiben und strukturieren Handlungsabläufe.
- (2) zerlegen Problemstellungen in geeignete Teilprobleme.
- (3) analysieren und beurteilen Modelle nach vorgegebenen oder selbst gewählten Kriterien.
- (4) entwickeln Modelle für eine Problemstellung und stellen diese dar.

P2 Algorithmisieren und Implementieren

Auf der Basis von Modellen und Strukturen werden Algorithmen für ein vorliegendes Problem oder eine Problemklasse entworfen. Dabei werden auch bereits bekannte Standardalgorithmen verwendet. Der Begriff des Implementierens bezeichnet den Prozess, eine Lösungsidee bzw. ein abstraktes Modell in eine konkrete Realisierung umzusetzen. Dies kann beispielsweise der Aufbau einer elektronischen Schaltung oder die Umsetzung eines Algorithmus in einer konkreten Programmiersprache sein.

Dieser kreative Schaffensprozess beginnt häufig als zielgerichtetes Experimentieren und wird durch geeignete Entwicklungsumgebungen unterstützt. Durch die Implementierung erhalten die Schülerinnen und Schüler die Möglichkeit, ihre Ideen umzusetzen und als Produkte zu verwirklichen. Die Implementierung ermöglicht den Schülerinnen und Schülern, die gefundene Lösung zu beurteilen und ggf. zu korrigieren.

Die Schülerinnen und Schüler ...

- (1) entwerfen Algorithmen und stellen diese geeignet dar.
- (2) setzen ihre Problemlösungen unter Verwendung von geeigneten Entwicklungsumgebungen in ausführbare Prozesse um.
- (3) analysieren, erläutern und vergleichen Algorithmen.
- (4) reflektieren ihr Vorgehen bei der Entwicklung und Implementierung von ausführbaren Prozessen.

P3 Kooperieren und Kommunizieren

Die Bearbeitung von Problemstellungen erfolgt in der Informatik häufig in kooperativen Arbeitsformen. Hierzu ist es notwendig, dass die Schülerinnen und Schüler Absprachen treffen, arbeitsteilig vorgehen, Ergebnisse vergleichen und zusammenführen. Sie erläutern und begründen ihre Lösungsideen und Vorgehensweisen.

Das Kommunizieren über informatische Zusammenhänge verlangt ein verständliches Darstellen und Präsentieren von Überlegungen, Lösungswegen und Ergebnissen. Die Schülerinnen und Schüler verstehen und überprüfen die Äußerungen und Dokumentationen von anderen. Durch die Verwendung von Fachbegriffen und standardisierter Darstellungsformen wird der Austausch von Inhalten erleichtert.

Die Schülerinnen und Schüler ...

- (1) kommunizieren unter Verwendung der Fachsprache über informatische Inhalte und stellen diese sachgerecht dar.
- (2) dokumentieren ihre Lösungsansätze und Lösungen mithilfe geeigneter Darstellungsformen.
- (3) begründen Zusammenhänge im Kontext der Informatik.
- (4) organisieren, dokumentieren und reflektieren die gemeinsame Arbeit im Team.

P4 Kreatives Schaffen und Problemlösen

Problemstellungen aus dem Kontext der Informatik zeichnen sich häufig dadurch aus, dass sie offen und unscharf definiert sind. Somit sind neben dem Problemlösen Kompetenzen der Problemfindung und -beschreibung wichtig. Deshalb ist es notwendig, eine Problemsensitivität zu entwickeln und mögliche Lösungsszenarien unter Reflexion der eigenen Fähigkeiten und Möglichkeiten umzusetzen. Kreatives Schaffen bedeutet in der Informatik das Hervorbringen eines persönlich neuen, individuell oder gesellschaftlich nützlichen Produkts. Dies geschieht durch Kombination und Erweiterung bekannter Konzepte, Strategien und Methoden und nicht durch direkte Anwendungen von zuvor erlernten Standardverfahren.

Die Schülerinnen und Schüler ...

- (1) erweitern gegebene Programme, Algorithmen und Modelle um eigene zusätzliche Funktionalitäten.
- (2) finden und erläutern Problemstellungen, die mithilfe informatischer Kompetenzen gelöst werden können.
- (3) geben unterschiedliche Lösungswege für ein selbst gestelltes oder gegebenes Problem an und entscheiden sich begründet für einen Weg.
- (4) erfinden Produkte oder Verfahren, indem sie informatische Konzepte, Strategien und Methoden in eigenständigen Wegen kombinieren.

3.2 Inhaltsbezogene Kompetenzen

Auch wenn die Kommunikationstechnologie und die Informatiksysteme sich in einem stetigen Wandel befinden, gibt es beständige Grundlagen der Informatik, auf denen diese Technologien und Systeme basieren. Diese umfassen die Repräsentation von Informationen durch Daten, das Algorithmisieren von Prozessen und den allgemeinen Aufbau von Informatiksystemen. Der Erwerb zugehöriger Kompetenzen ermöglicht es, die Einflüsse der Informatik auf die Gesellschaft zu reflektieren und zu gestalten. Weiterführende Inhalte wie Verschlüsselungs- und Codierungstechniken, Datenbanken, Automaten sowie Standardalgorithmen ermöglichen einen vertieften Einblick in die Denk- und Arbeitsweisen der Informatik.

I1 Daten und ihre Strukturierung

Um Informationen zu speichern und zu verarbeiten, müssen sie zunächst in geeigneter Form durch Daten repräsentiert werden. Erst diese Repräsentation erlaubt die automatisierte Verarbeitung und damit die Erzeugung neuer Daten, aus deren Interpretation neue Informationen gewonnen werden können.

Zur Strukturierung von Daten existieren in der Informatik verschiedene Modellierungstechniken, die jeweils für unterschiedliche Kontexte und Problemstellungen geeignet sind und arbeitsteiliges Problemlösen ermöglichen.

Die Schülerinnen und Schüler ...

- (1) wählen eine für die Problemstellung geeignete Codierung.
- (2) verwenden Prinzipien eines Variablenkonzepts.
- (3) organisieren Daten mithilfe geeigneter Datenstrukturen.
- (4) speichern und verarbeiten Daten unter Verwendung des objektorientierten Modells.
- (5) strukturieren Daten mithilfe des relationalen Modells.

I2 Operationen auf Daten und Algorithmen

Ein Algorithmus ist eine Zusammensetzung von Basisoperationen und dient der Transformation von Eingaben in Ausgaben. Bei der objektorientierten Modellierung der Daten in komplexeren Softwareprojekten werden meist imperative Algorithmen verwendet. Das relationale Modell für die Verwaltung von Daten in Datenbanken erfordert eine deduktive Formulierung der Operationen.

Die Schülerinnen und Schüler lösen Problemstellungen unter Verwendung von Lösungsstrategien, die für das Datenmodell geeignet sind. Dabei entwickeln sie ein Verständnis für die Funktionsweise von Informatiksystemen und die Arbeitsweise der Informatik.

Die Schülerinnen und Schüler ...

- (1) verwenden die algorithmischen Grundkonzepte zur Konstruktion eigener Lösungsideen.
- (2) verwenden Standardalgorithmen im Rahmen eigener Problemlösungen.
- (3) entwickeln und verwenden Algorithmen zur Transformation von Codierungen.
- (4) verwenden eine Abfragesprache zum Filtern von Informationen.

I3 Informatiksysteme und ihre zugrundeliegenden Modelle

Jedes rechnergestützte System, von eingebetteten Systemen für die Steuerung von Haushaltsgeräten über den PC und mobile Endgeräte bis hin zum Zusammenschluss von Rechnern zu einem Netzwerk, stellt ein Informatiksystem dar. Informatiksysteme können auf der Ebene der Hardware, der Software sowie der Interaktion in Netzwerken betrachtet werden, wobei ein System nicht alle drei Ebenen enthalten muss.

Um den Aufbau und die Funktionsweise von Informatiksystemen zu verstehen und diese reflektiert verwenden zu können, rekonstruieren die Schülerinnen und Schüler Teile der Systeme und

abstrahieren wesentliche Eigenschaften und Prinzipien mithilfe von Modellen wie z. B. endlichen Automaten.

Die Schülerinnen und Schüler ...

- (1) erläutern die Funktionsweise von Informatiksystemen mithilfe von Maschinenmodellen.
- (2) rekonstruieren und entwerfen Teile von Informatiksystemen.
- (3) analysieren und rekonstruieren den Aufbau vernetzter Systeme.
- (4) entwerfen und erläutern Konzepte für die Kommunikation und Interaktion von Prozessen.

I4 Möglichkeiten und Grenzen informatischer Verfahren und Systeme

Die Kenntnisse über die prinzipiellen Möglichkeiten und Grenzen von Informatiksystemen ermöglichen den Schülerinnen und Schülern, die Auswirkungen, die sich durch die ständige technische Weiterentwicklung der Informatiksysteme ergeben, kompetent zu beurteilen. Beispiele aus der Vergangenheit helfen die Wirkung von Informatiksystemen auf alle gesellschaftlichen Bereiche zu verdeutlichen und den Bedarf von Regulierungen zu erkennen. Die Betrachtung aktueller Entwicklungen gestattet deren Diskussion aufgrund der erworbenen Fachkenntnisse.

Die Schülerinnen und Schüler ...

- (1) reflektieren die Chancen und Risiken des Einsatzes von Informatiksystemen auf die Gesellschaft.
- (2) diskutieren wesentliche Aspekte des Datenschutzes und Urheberrechts anhand von Anwendungsfällen.
- (3) untersuchen und beurteilen Grenzen des Problemlösens mit Informatiksystemen.

3.3 Zusammenführung von Kompetenzen

Die Zusammenführung der inhalts- und prozessbezogenen Kompetenzen erfolgt in den drei Lernfeldern *Algorithmen und Datenstrukturen*, *Information und Daten* sowie *Automaten und Sprachen*. Die Lernfelder sind jeweils tabellarisch in Module unterteilt. Die Anordnung legt weder eine Rangfolge noch eine zeitliche Reihenfolge der unterrichtlichen Umsetzung fest. Für die Unterrichtsinhalte sind geeignete Kontexte so zu wählen, dass aktuelle und lebensweltnahe Bezüge für die Schülerinnen und Schüler hergestellt werden.

In der ersten Spalte sind die Kompetenzen beschrieben, die von den Schülerinnen und Schülern bis zum Ende der Einführungsphase erworben werden sollen. In den beiden folgenden Spalten sind die Kompetenzen für die Qualifikationsphase beschrieben.

Für jedes Modul werden die enthaltenen prozessbezogenen und inhaltsbezogenen Kompetenzen anhand der in Kapitel 3.1 und 3.2 verwendeten Nummerierung ausgewiesen. Um die Zuordnung der Nummern zu den jeweiligen Kompetenzen zu erleichtern, befindet sich im Anhang eine kompakte Übersicht über alle prozessbezogenen und inhaltsbezogenen Kompetenzen.

Lernfeld: Algorithmen und Datenstrukturen

In der Informatik werden Problemstellungen häufig mithilfe von Algorithmen gelöst. Der Informatikunterricht macht Schülerinnen und Schüler in unterschiedlichen Kontexten mit den Prinzipien der algorithmischen Problemlösung und der Implementierung ihrer Lösungsideen vertraut. Kenntnisse iterativer und rekursiver Algorithmen ermöglichen es den Schülerinnen und Schülern, eigene Lösungsansätze zu entwickeln. Unter der Verwendung der elementaren Kontrollstrukturen und grundlegender Datenstrukturen werden Algorithmen entworfen, implementiert, getestet und überarbeitet. Die Verwendung von selbst entworfenen Operationen, Klassen und Objekten ermöglicht den Schülerinnen und Schülern eine strukturierte Realisierung ihrer Lösungsideen.

	Einführungsphase	Qualifikationsphase	Erweiterung eA
Grundlagen der Algorithmen	<ul style="list-style-type: none"> - benennen Anweisung, Sequenz, Schleife und Verzweigung als Grundstrukturen eines Algorithmus - entwerfen einen Algorithmus unter zielgerichteter Verwendung der elementaren Kontrollstrukturen 		<ul style="list-style-type: none"> - beurteilen die Effizienz von Algorithmen unter Berücksichtigung des Speicherbedarfs und der Zahl der Operationen
	<ul style="list-style-type: none"> - erläutern das Prinzip der Speicherung von Werten in Variablen - verwenden Variablen und Wertzuweisungen in Algorithmen 	<ul style="list-style-type: none"> - verwenden geeignete Variablentypen zur Speicherung von Werten - unterscheiden zwischen lokalen und globalen Variablen - unterscheiden zwischen primitiven und Referenz-Datentypen 	
	<ul style="list-style-type: none"> - verwenden Operationen zur strukturierten Implementierung von Algorithmen 	<ul style="list-style-type: none"> - verwenden Übergabeparameter und Rückgabewerte in Operationen 	
		<ul style="list-style-type: none"> - erläutern das Konzept der Rekursion an gegebenen Beispielen wie z. B. binäre Suche oder grafische Rekursionen 	<ul style="list-style-type: none"> - implementieren rekursive Algorithmen in gegebenen Zusammenhängen - erläutern die Strategie „Teile und herrsche“ beim Entwurf rekursiver Algorithmen
	P1.1-2 P2.1 P3.2 P4.4 I1.2 I2.1	P1.1-2 P2.1 P2.4 P3 P4.3 I1.2-3	

	Einführungsphase	Qualifikationsphase	Erweiterung eA
Klassen und Objekte		<ul style="list-style-type: none">- verwenden gegebene Klassen/Objekte zur Realisierung eigener Programme- entwerfen und implementieren Klassen/Objekte zur Realisierung eigener Programme- stellen Klassen/Objekte und deren Beziehungen durch UML-Klassen-/Objektdiagramme dar	
		P1.3-4 P2.2 P2.4 P3.2 P4 I1.3-4	
		<ul style="list-style-type: none">- erläutern das Prinzip, mehrere Daten des gleichen Typs in Reihungen zu verwalten, zu suchen und zu sortieren- verwenden ein- und zweidimensionale Reihungen zur Realisierung eigener Programme- erläutern das Prinzip der Datenstrukturen Stapel, Schlange und dynamische Reihung- verwenden die Datenstrukturen Stapel, Schlange und dynamische Reihung zur Realisierung eigener Programme	<ul style="list-style-type: none">- erläutern das Prinzip der Datenstruktur Binärbaum und verwenden diese Struktur zur Realisierung eigener Programme
		P2 P3 P4 I1.3 I2.2	
Im Zusammenhang mit diesem Lernfeld bieten sich als über den Kern hinausgehende Ergänzungen an: Algorithmen auf Graphen, genetische Algorithmen, Backtracking, Komplexitätstheorie, Polymorphie.			

Lernfeld: Informationen und Daten

Die Erfassung, Verwaltung und Auswertung von Daten spielen eine zentrale Rolle im täglichen Umgang mit Informationen. Bei der Speicherung und der Übertragung von Daten sind Aspekte der Datensicherheit und der Effizienz wesentlich. Dazu lernen die Schülerinnen und Schüler exemplarische Verfahren der Datenverschlüsselung sowie der Kompression und Fehlererkennung kennen. Kenntnisse der Strukturen von Datenbanken und deren Auswertung sind ein weiterer Schwerpunkt in diesem Lernfeld. Im erhöhten Anforderungsniveau wird an geeigneten Beispielen die Modellierung von Datenbanken mithilfe des ER-Modells behandelt.

	Einführungsphase	Qualifikationsphase	Erweiterung eA
Kryptologie	<ul style="list-style-type: none"> - beschreiben das Prinzip der Transposition und der Substitution zur Verschlüsselung von Daten, u.a. Caesar- und Skytale-Verfahren - implementieren monoalphabetische Verfahren, u. a. Caesar-Verfahren - beurteilen die Sicherheit einfacher Verschlüsselungsverfahren 	<ul style="list-style-type: none"> - beschreiben das Prinzip der polyalphabetischen Substitutionschiffren, u. a. Vigenère-Verfahren - implementieren polyalphabetische Verfahren, u.a. Vigenère-Verfahren - beschreiben und unterscheiden die Prinzipien der symmetrischen und der asymmetrischen Verschlüsselung - beschreiben Anwendungsbereiche für den Einsatz symmetrischer bzw. asymmetrischer Verschlüsselungen - beschreiben Anwendungsbereiche für den Einsatz von digitalen Signaturen und Zertifikaten - beurteilen die Sicherheit symmetrischer und asymmetrischer Verschlüsselungen 	
	P1.1 P2 P3.3 P4.3 I1.1-2 I2.1	P1.1-2 P2 P3.3 P4.1 P4.3 I1.1-2 I2.1 I2.3	
Datenschutz	<ul style="list-style-type: none"> - erläutern die rechtlichen Rahmenbedingungen für den Umgang mit ihren persönlichen Daten, wie z. B. informationelle Selbstbestimmung und Datenschutzrichtlinien - diskutieren die Chancen und Risiken der automatisierten Datenanalyse 		
	P3.1 P3.3 P3.4 I4		

	Einführungsphase	Qualifikationsphase	Erweiterung eA
Codierung und Übertragung von Daten	<ul style="list-style-type: none"> - beschreiben grundlegende Codierungen von Daten, u. a. Dualzahlen, ASCII, RGB-Modell - entwerfen und implementieren ein Protokoll zur Übertragung von Daten über einen Kommunikationskanal - beschreiben und begründen den dezentralen Aufbau des Internets - nennen die zentralen Komponenten des Internets, u.a. Client, Server, Router, DNS und erläutern ihre Funktion - beschreiben die Kommunikationswege im Internet - beschreiben Aspekte zur Sicherheit der Kommunikation im Internet 	<ul style="list-style-type: none"> - beschreiben das Prinzip, Daten zu komprimieren, u.a. Lauflängencodierung, Huffman-Codierung - implementieren Kompressionsverfahren, u. a. Lauflängencodierung - erläutern Möglichkeiten der Fehlererkennung und Fehlerkorrektur bei der Datenübertragung, u.a. Paritätsbit, (7,4)-Hamming-Code 	
	P1 P2 P3 P4 I1.1 I2.1 I2.3 I3.2-4	P1 P2 P3.1 P4.3-4 I1.1 I2.1-3	
Datenbanken		<ul style="list-style-type: none"> - unterscheiden Datensatz, Attribut, Schlüsselattribut, Tabelle und Datenbank - erläutern die Notwendigkeit, Datensätze eindeutig identifizieren zu können (Primärschlüssel) - erläutern den grundlegenden Aufbau von ER-Diagrammen - erläutern die Bedeutung von Kardinalitäten für die Beziehungen zwischen Entitäten - nennen Beispiele für Einfüge-, Änderungs- und Löschanomalien - untersuchen eine gegebene Datenbankmodellierung auf mögliche auftretende Anomalien - formulieren einfache Abfragen und Verbundabfragen über mehrere Tabellen in SQL - formulieren Abfragen an Datenbanken durch Verwendung von Aggregatfunktionen 	<ul style="list-style-type: none"> - modellieren Datenbanken unter Verwendung des ER-Modells - berücksichtigen bei der Datenbankmodellierung normalisierende Aspekte zur Vermeidung von Anomalien
		P1 P3 P4.1-2 I1.5 I2.4	
Im Zusammenhang mit diesem Lernfeld bieten sich als über den Kern hinausgehende Ergänzungen an: RSA-Verschlüsselung, Kryptoanalyse, Transaktionsverwaltung.			

Lernfeld: Automaten und Sprachen

Zustandsbasierte Modellierung ist eine der grundlegenden Techniken zur Entwicklung und Realisierung von technischen Systemen und Programmen. In der Informatik geschieht dies häufig in Form von endlichen Automaten unterschiedlicher Ausprägung. Die Schülerinnen und Schüler sollen in die Lage versetzt werden, das Verhalten technischer Systeme in Form von endlichen Automaten zu modellieren, wodurch sich ihnen die Funktionsweise und der Aufbau dieser Systeme erschließt. Gleichzeitig sollen Sie die Grenzen der jeweiligen Modelle erfahren sowie Ansätze zur Erweiterung entwickeln. Auf erhöhtem Niveau schließt dies die Beschreibung in Form von formalen Sprachen sowie deren Synthese und Analyse mithilfe von Grammatiken ein.

	Einführungsphase	Qualifikationsphase	Erweiterung eA
endliche Automaten		<ul style="list-style-type: none">- beschreiben den Aufbau und die Funktionsweise eines endlichen Automaten (DEA)- beschreiben den Aufbau und die Funktionsweise eines endlichen Automaten mit Ausgabe (Mealy-Automat)- entwickeln und implementieren Automatenmodelle in Form von Zustandsgraphen- analysieren die Funktionsfähigkeit eines durch einen Zustandsgraphen vorgegebenen Automaten- erläutern die Grenzen von endlichen Automaten bei der Analyse von Eingaben	<ul style="list-style-type: none">- beschreiben den Aufbau und die Funktionsweise eines deterministischen Kellerautomaten als Erweiterung des Modells des DEA
		P1.1 P1.3-4 P2.2 P3.1-3 P4.2-3 I3.1 I4.3	
formale Sprachen			<ul style="list-style-type: none">- nennen Eigenschaften formaler Sprachen an verschiedenen Beispielen- beschreiben die von einer Grammatik erzeugte Sprache- erstellen Grammatiken für formale Sprachen- erläutern den Zusammenhang zwischen regulären Grammatiken und endlichen Automaten
		P3.1 P3.3 P4.2-3 I3.1 I4.3	
Im Zusammenhang mit diesem Lernfeld bieten sich als über den Kern hinausgehende Ergänzungen an: Turingmaschine, Chomsky Hierarchie, zelluläre Automaten, L-Systeme, Petri-Netze, Schaltnetze, Schaltwerke.			

3.4 Projektorientierter Unterricht

Der Umfang der in den Lernfeldern dargestellten Kompetenzen gibt ausreichend Zeit, um projektorientierten Unterricht durchzuführen. Dabei arbeiten Schülerinnen und Schüler über einen längeren Zeitraum selbstständig an einem Thema. Die Selbstorganisation der Lerngruppe, die Planung der gemeinsamen Arbeit und das Erstellen eines fertigen Produkts, das präsentiert und evaluiert wird, stehen im Mittelpunkt. Als Themen für diese Unterrichtsprojekte bieten sich die in den Lernfeldern angegebenen Module sowie die Ergänzungen und Erweiterungen an. Nach Interesse der Lerngruppe können auch andere Inhalte der Informatik Gegenstand eines Unterrichtsprojektes sein.

Eine Phase des projektorientierten Arbeitens soll in einem der ersten drei Halbjahre der Qualifikationsphase stattfinden und mindestens drei Unterrichtswochen umfassen. Im gesamten Verlauf der Qualifikationsphase sollen Projektphasen im Umfang von insgesamt mindestens acht Wochen durchgeführt werden.

4 Leistungsfeststellung und Leistungsbewertung

Leistungen im Unterricht sind in allen Kompetenzbereichen festzustellen. Dabei ist zu bedenken, dass die sozialen und personalen Kompetenzen, die über das Fachliche hinausgehen, von den im Kerncurriculum formulierten erwarteten Kompetenzen nur in Ansätzen erfasst werden.

Der an Kompetenzerwerb orientierte Unterricht bietet den Schülerinnen und Schülern einerseits ausreichend Gelegenheiten, Problemlösungen zu erproben, andererseits fordert er den Kompetenznachweis in Leistungssituationen. Ein derartiger Unterricht schließt die Förderung der Fähigkeit zur Selbsteinschätzung der Leistung ein. In Lernsituationen dienen Fehler und Umwege den Schülerinnen und Schülern als Erkenntnismittel, den Lehrkräften geben sie Hinweise für die weitere Unterrichtsplanung. Das Erkennen von Fehlern und der produktive Umgang mit ihnen sind konstruktiver Teil des Lernprozesses. Für den weiteren Lernfortschritt ist es wichtig, bereits erworbene Kompetenzen herauszustellen und Schülerinnen und Schüler zum Weiterlernen zu ermutigen.

In Leistungs- und Überprüfungssituationen ist das Ziel, die Verfügbarkeit der erwarteten Kompetenzen nachzuweisen. Leistungsfeststellungen und Leistungsbewertungen geben den Schülerinnen und Schülern Rückmeldungen über die erworbenen Kompetenzen und den Lehrkräften Orientierung für notwendige Maßnahmen zur individuellen Förderung. Neben der kontinuierlichen Beobachtung der Schülerinnen und Schüler im Lernprozess und ihrer individuellen Lernfortschritte, die in der Dokumentation der individuellen Lernentwicklung erfasst werden, sind die Ergebnisse mündlicher, schriftlicher und anderer fachspezifischer Lernkontrollen zur Leistungsfeststellung heranzuziehen.

In Lernkontrollen werden überwiegend Kompetenzen überprüft, die im unmittelbar vorangegangenen Unterricht erworben werden konnten. Darüber hinaus sollen jedoch auch Problemstellungen einbezogen werden, die die Verfügbarkeit von Kompetenzen eines langfristig angelegten Kompetenzaufbaus überprüfen. In schriftlichen Lernkontrollen sind alle drei Anforderungsbereiche *„Wiedergeben und beschreiben“*, *„Anwenden und strukturieren“* sowie *„Transferieren und verknüpfen“* zu berücksichtigen. Bei schriftlichen Lernkontrollen liegt der Schwerpunkt in der Regel in den Bereichen I und II.

Mündliche und fachspezifische Leistungen gehen mit einem höheren Gewicht in die Gesamtzensur ein als die schriftlichen Leistungen. Der Anteil der schriftlichen Leistungen an der Gesamtzensur ist abhängig von der Anzahl der schriftlichen Lernkontrollen innerhalb eines Schulhalbjahres. Der Anteil der schriftlichen Leistungen darf ein Drittel an der Gesamtzensur nicht unterschreiten.

Zu mündlichen und anderen fachspezifischen Leistungen zählen z. B.:

- Beiträge zum Unterrichtsgespräch
- Mündliche Überprüfungen
- Unterrichtsdokumentationen
- Anwenden fachspezifischer Methoden und Arbeitsweisen
- Präsentationen
- Ergebnisse von Partner- oder Gruppenarbeiten und deren Darstellung

- Projekt- und Langzeitaufgaben
- Freie Leistungsvergleiche (z. B. Schülerwettbewerbe)

Bei kooperativen Arbeitsformen sind sowohl die individuelle Leistung als auch die Gesamtleistung der Gruppe in die Bewertung einzubeziehen. So werden neben methodisch-strategischen auch die sozial-kommunikativen Leistungen angemessen berücksichtigt.

Fachpraktische Leistungen im Fach Informatik sind Produkte, die mit den jeweiligen Informatikwerkzeugen von den Schülerinnen und Schülern in Einzel- oder Gruppenarbeit selbstständig hergestellt wurden. Diese Produkte werden insbesondere in Projektphasen entwickelt. Sie müssen unter fachspezifischen Aspekten gestaltet und beurteilt werden. Bei der Bewertung der Produkte darf nicht nur das Endergebnis, sondern muss auch der Entstehungsprozess und die Dokumentation Berücksichtigung finden.

Die Grundsätze der Leistungsfeststellung und -bewertung müssen für Schülerinnen und Schüler sowie für die Erziehungsberechtigten transparent sein.

5 Aufgaben der Fachkonferenz

Die Fachkonferenz erarbeitet unter Beachtung der rechtlichen Grundlagen und der fachbezogenen Vorgaben des Kerncurriculums einen fachbezogenen schuleigenen Arbeitsplan (Fachcurriculum). Die Erstellung des Fachcurriculums ist ein Prozess.

Mit der regelmäßigen Überprüfung und Weiterentwicklung des Fachcurriculums trägt die Fachkonferenz zur Qualitätsentwicklung des Faches und zur Qualitätssicherung bei.

Die Fachkonferenz ...

- legt die Themen bzw. die Struktur von Unterrichtseinheiten fest, die die Entwicklung der erwarteten Kompetenzen ermöglichen, und berücksichtigt dabei regionale Bezüge,
- legt die Zuordnung von Kompetenzen und Themen innerhalb der Schulhalbjahre fest,
- entwickelt Unterrichtskonzepte zur inneren Differenzierung,
- arbeitet fachübergreifende und fächerverbindende Anteile des Fachcurriculums heraus und stimmt diese mit den anderen Fachkonferenzen ab,
- entscheidet, welche Schulbücher und Unterrichtsmaterialien eingeführt werden sollen,
- trifft Absprachen zur einheitlichen Verwendung der Fachsprache und der fachbezogenen Hilfsmittel,
- trifft Absprachen zur Konzeption und zur Bewertung von schriftlichen, mündlichen und fachspezifischen Leistungen und bestimmt deren Verhältnis bei der Festlegung der Gesamtleistung,
- wirkt mit bei der Erstellung des fächerübergreifenden Konzepts zur Berufsorientierung und Berufsbildung und greift das Konzept im Fachcurriculum auf,
- entwickelt ein fachbezogenes Konzept zum Einsatz von Medien im Zusammenhang mit dem schulinternen Mediencurriculum,
- initiiert die Nutzung außerschulischer Lernorte, die Teilnahme an Wettbewerben etc.,
- initiiert Beiträge des Faches zur Gestaltung des Schullebens (Ausstellungen, Projekttag etc.) und trägt zur Entwicklung des Schulprogramms bei,
- ermittelt Fortbildungsbedarf innerhalb der Fachgruppe und entwickelt Fortbildungskonzepte für die Fachlehrkräfte.

Anhang

A1 Liste der Operatoren für das Fach Informatik

Für zentrale Prüfungsaufgaben müssen Vereinbarungen hinsichtlich der Formulierung von Arbeitsaufträgen und der erwarteten Leistung getroffen werden. Operatoren, die für das Fach Informatik eine besondere Bedeutung haben, werden in der unten stehenden Tabelle beschrieben. Die Verwendung weiterer Operatoren ist möglich, wenn sich der notwendige Bearbeitungsumfang deutlich aus dem Kontext oder einer ausführlicheren Beschreibung ergibt.

Operator	Definition	Beispiel
abschätzen	durch begründete Überlegungen Größenordnungen angeben	Schätzen Sie das Zeitverhalten des Verfahrens ab, wenn sich die Anzahl der zu bearbeitenden Daten verdoppelt.
analysieren / untersuchen	unter einer gegebenen Fragestellung wichtige Bestandteile oder Eigenschaften nach fachlich üblichen Kriterien herausarbeiten	Analysieren Sie die Funktionsweise des Algorithmus. Untersuchen Sie welche Eingaben durch den Automaten akzeptiert werden.
anwenden	ein gegebenes Verfahren unter Berücksichtigung konkreter Werte durchführen	Wenden Sie das Caesar-Verfahren zur Verschlüsselung des Textes ... an.
begründen / zeigen	einen Sachverhalt auf Gesetzmäßigkeiten bzw. kausale Zusammenhänge zurückführen	Begründen Sie die folgende Aussage: ... Zeigen Sie, dass die folgenden Wörter zur Sprache gehören.
berechnen	Ergebnisse durch Rechenoperationen gewinnen	Berechnen Sie die Länge des komprimierten Codes.
beschreiben	Sachverhalte oder Verfahren in Textform unter Verwendung der Fachsprache in eigenen Worten wiedergeben	Beschreiben Sie das Verfahren der asymmetrischen Verschlüsselung.
bestimmen	einen Lösungsweg darstellen und das Ergebnis formulieren	Bestimmen Sie die Anzahl der rekursiven Aufrufe.
beurteilen	zu einem Sachverhalt ein selbstständiges Urteil unter Verwendung von Fachwissen und Fachmethoden formulieren und begründen	Beurteilen Sie die Sicherheit des Verfahrens.
darstellen	Sachverhalte, Zusammenhänge und Algorithmen strukturiert in ggf. fachspezifischer Form wiedergeben	Stellen Sie den Verlauf in einer Tracetabelle dar.
erörtern	Argumente zu einer Aussage oder These einander gegenüberstellen und abwägen	Erörtern Sie Vor- und Nachteile aus der Sicht des Benutzers.
entwerfen / entwickeln	Nach vorgegebenen Bedingungen ein Modell / einen Algorithmus selbständig planen / erarbeiten	Entwerfen Sie ein ER-Modell, das ... Entwickeln Sie einen endlichen Automaten, der ...
ergänzen / erweitern / verändern	eine vorgegebene Problemlösung unter Berücksichtigung vorgegebener Kriterien anpassen	Erweitern Sie das ER-Modell so, dass ... Ergänzen Sie das Klassendiagramm um geeignete Attribute.

erläutern	einen Sachverhalt durch zusätzliche Informationen veranschaulichen und verständlich machen	Erläutern Sie das Prinzip der Häufigkeitsanalyse.
erstellen	bekannte Verfahren zur Lösung eines neuen Problems aus einem bekannten Problemkreis anwenden	Erstellen Sie aus den Vorgaben ein Klassendiagramm.
implementieren	Erarbeiten und Codieren eines Algorithmus oder einer Datenstruktur	Implementieren Sie eine Operation, die ...
nennen / angeben	ohne Erläuterungen und Begründungen aufzählen	Nennen Sie drei Beispielwerte, die ...
vergleichen	nach vorgegebenen oder selbst gewählten Gesichtspunkten Gemeinsamkeiten, Ähnlichkeiten und Unterschiede ermitteln und darstellen	Vergleichen Sie die beiden Kompressionsverfahren.
zeichnen / grafisch darstellen	die wesentlichen Eigenschaften eines Verfahrens / eines Modells übersichtlich in einer Zeichnung darstellen	Zeichnen Sie den zugehörigen Suchbaum.

A2 Operationen der Klassen *dynamische Reihung*, *Stapel*, *Schlange* und *Binärbaum*

Im Rahmen von zentralen Prüfungsaufgaben werden für die aufgeführten Klassen die folgenden Operationen verwendet. Die Klassen benutzen Inhaltsklassen bzw. -typen, die jeweils der aktuellen Aufgabenstellung angepasst werden. Die Klassen werden in den Aufgabenstellungen gegebenenfalls um Attribute und weitere Operationen ergänzt.

Dynamische Reihung

Die Nummerierung der dynamischen Reihung beginnt mit dem Index 1.

`DynArray()`

Eine leere dynamische Reihung wird angelegt.

`isEmpty(): Wahrheitswert`

Wenn die Reihung kein Element enthält, wird der Wert *wahr* zurückgegeben, sonst der Wert *falsch*.

`getItem(Ganzzahl index): Inhalt`

Der Inhalt an der Position index wird zurückgegeben.

`append(Inhalt inhalt)`

Der Inhalt wird am Ende der dynamischen Reihung eingefügt.

`insertAt(Ganzzahl index, Inhalt inhalt)`

Der Inhalt wird an der Position index in die dynamische Reihung eingefügt. Das sich vorher an dieser Position befindende Element und alle weiteren werden nach hinten verschoben. Falls die angegebene Position nicht existiert, hat die Operation keine Wirkung.

`delete(Ganzzahl index)`

Der Inhalt an der Position wird entfernt. Alle folgenden Elemente werden eine Position nach vorne geschoben. Falls die angegebene Position nicht existiert, hat die Operation keine Wirkung.

`getLength(): Ganzzahl`

Die Anzahl der Elemente der dynamischen Reihung wird zurückgegeben.

Stapel

`Stack()`

Ein leerer Stapel wird angelegt.

`isEmpty(): Wahrheitswert`

Wenn der Stapel kein Element enthält, wird der Wert *wahr* zurückgegeben, sonst der Wert *falsch*.

`top(): Inhalt`

Der Inhalt des obersten Elements des Stapels wird zurückgegeben, das Element aber nicht entfernt.

`push(Inhalt inhalt)`

Ein neues Element mit dem angegebenen Inhalt wird auf den Stapel gelegt.

`pop(): Inhalt`

Der Inhalt des obersten Elements wird zurückgegeben und das Element wird entfernt.

Schlange

`Queue()`

Eine leere Schlange wird angelegt.

`isEmpty(): Wahrheitswert`

Wenn die Schlange kein Element enthält, wird der Wert *wahr* zurückgegeben, sonst der Wert *falsch*.

`head(): Inhalt`

Der Inhalt des ersten Elements der Schlange wird zurückgegeben, das Element aber nicht entfernt.

`enqueue(Inhalt inhalt)`

Ein neues Element mit dem angegebenen Inhalt wird angelegt und am Ende an die Schlange angehängt.

`dequeue(): Inhalt`

Der Inhalt des ersten Elements wird zurückgegeben und das Element wird entfernt.

Binärbaum

`BinTree(Inhalt inhalt)`

Ein Baum wird erzeugt. Die Wurzel erhält den übergebenen Inhalt als Wert.

`isLeaf(): Wahrheitswert`

Die Anfrage liefert den Wert *wahr*, wenn der Baum keine Nachfolger hat, sonst liefert sie den Wert *falsch*.

`getLeft(): BinTree`

Die Operation gibt den linken Teilbaum zurück. Existiert kein linker Nachfolger, so ist das Ergebnis *null*.

`getRight(): BinTree`

Die Operation gibt den rechten Teilbaum zurück. Existiert kein rechter Nachfolger, so ist das Ergebnis *null*.

`setLeft(BinTree b)`

Der übergebene Baum wird als linker Teilbaum gesetzt.

`setRight(BinTree b)`

Der übergebene Baum wird als rechter Teilbaum gesetzt.

`getItem(): Inhalt`

Die Operation gibt den Inhaltswert der Wurzel des aktuellen Baumes zurück.

`setItem(Inhalt inhalt)`

Die Operation setzt den Inhaltswert der Wurzel des aktuellen Baumes.

A3 SQL-Syntax

Im Rahmen von zentralen Prüfungsaufgaben wird die folgende SQL-Syntax unter Berücksichtigung der angegebenen Operatoren und Gruppenfunktionen verwendet.

SELECT-Anweisung:

```
SELECT [ALL|DISTINCT] Select-Ausdruck {,Select-Ausdruck}  
FROM TabellenName {,TabellenName}  
[WHERE Bedingung]  
[GROUP BY SpaltenName {,SpaltenName}]  
[HAVING Bedingung]  
[ORDER BY Select-Ausdruck [ASC|DESC]{,Select-Ausdruck [ASC|DESC]}]  
[LIMIT Zeilenzahl]
```

Ein Select-Ausdruck kann ein Spaltenname, eine Spaltenfunktion oder ein Alias-Name sein.

Operatoren:

```
+, -, *, /,  
=, != (ungleich), >, <, >=, <=,  
AND, OR, NOT,  
LIKE, BETWEEN, IN, IS NULL
```

Arithmetische Gruppenfunktionen:

```
AVG( ), COUNT( ), MAX( ), MIN( ), SUM( )
```

A4 Auflistung der Kompetenzen

Prozessbezogene Kompetenzen

Strukturieren und Modellieren

- P 1.1. beschreiben und strukturieren Handlungsabläufe.
- P 1.2. zerlegen Problemstellungen in geeignete Teilprobleme.
- P 1.3. analysieren und beurteilen Modelle nach vorgegebenen oder selbst gewählten Kriterien.
- P 1.4. entwickeln Modelle für eine Problemstellung und stellen diese dar.

Algorithmisieren und Implementieren

- P 2.1. entwerfen Algorithmen und stellen diese geeignet dar.
- P 2.2. setzen ihre Problemlösungen unter Verwendung von geeigneten Entwicklungsumgebungen in ausführbare Prozesse um.
- P 2.3. analysieren, erläutern und vergleichen Algorithmen.
- P 2.4. reflektieren ihr Vorgehen bei der Entwicklung und Implementierung von ausführbaren Prozessen.

Kooperieren und Kommunizieren

- P 3.1. kommunizieren unter Verwendung der Fachsprache über informatische Inhalte und stellen diese sachgerecht dar.
- P 3.2. dokumentieren ihre Lösungsansätze und Lösungen mithilfe geeigneter Darstellungsformen.
- P 3.3. begründen Zusammenhänge im Kontext der Informatik.
- P 3.4. organisieren, dokumentieren und reflektieren die gemeinsame Arbeit im Team.

Kreatives Schaffen und Problemlösen

- P 4.1. erweitern gegebene Programme, Algorithmen und Modelle um eigene zusätzliche Funktionalitäten.
- P 4.2. finden und erläutern Problemstellungen, die mithilfe informatischer Kompetenzen gelöst werden können.
- P 4.3. geben unterschiedliche Lösungswege für ein selbst gestelltes oder gegebenes Problem an und entscheiden sich begründet für einen Weg.
- P 4.4. erfinden Produkte oder Verfahren, indem sie informatische Konzepte, Strategien und Methoden in eigenständigen Wegen kombinieren.

Inhaltsbezogene Kompetenzen

Daten und ihre Strukturierung

- I 1.1. wählen eine für die Problemstellung geeignete Codierung.
- I 1.2. verwenden Prinzipien eines Variablenkonzepts.
- I 1.3. organisieren Daten mithilfe geeigneter Datenstrukturen.
- I 1.4. speichern und verarbeiten Daten unter Verwendung des objektorientierten Modells.
- I 1.5. strukturieren Daten mithilfe des relationalen Modells.

Operationen auf Daten und Algorithmen

- I 2.1. verwenden die algorithmischen Grundkonzepte zur Konstruktion eigener Lösungsideen.
- I 2.2. verwenden Standardalgorithmen im Rahmen eigener Problemlösungen.
- I 2.3. entwickeln und verwenden Algorithmen zur Transformation von Codierungen.
- I 2.4. verwenden eine Abfragesprache zum Filtern von Informationen.

Informatiksysteme und ihre zugrundeliegenden Modelle

- I 3.1. erläutern die Funktionsweise von Informatiksystemen mithilfe von Maschinenmodellen.
- I 3.2. rekonstruieren und entwerfen Teile von Informatiksystemen.
- I 3.3. analysieren und rekonstruieren den Aufbau vernetzter Systeme.
- I 3.4. entwerfen und erläutern Konzepte für die Kommunikation und Interaktion von Prozessen.

Möglichkeiten und Grenzen informatischer Verfahren und Systeme

- I 4.1. reflektieren die Chancen und Risiken des Einsatzes von Informatiksystemen auf die Gesellschaft.
- I 4.2. diskutieren wesentliche Aspekte des Datenschutzes und Urheberrechts anhand von Anwendungsfällen.
- I 4.3. untersuchen und beurteilen Grenzen des Problemlösens mit Informatiksystemen.