

Fachlehrplan Gymnasium

01.08.2022



SACHSEN-ANHALT

Ministerium für Bildung

Informatik

Der Lehrplan für das Gymnasium ist eine Einheit aus Grundsatzband und Fachlehrplänen.

An der Erarbeitung des Fachlehrplans haben mitgewirkt:

Burgemeister, Maik	Bernburg
Degen, Antje	Hettstedt
Eschrich, Mario	Halle
Grötzsch, Steffi	Halle (Leitung der Fachgruppe)
Dr. Herper, Henry	Magdeburg (wissenschaftliche Beratung)
Prof. Dr. Müller-Hannemann, Matthias	Halle (wissenschaftliche Beratung)

An der Anpassung des Fachlehrplans zu den Schwerpunkten Nachhaltigkeit, Bildung in der digitalen Welt, Stärkung bildungssprachlicher Kompetenzen und den Ergebnissen nach der Erprobung des Fachlehrplans haben mitgewirkt:

Blanke, Kathrin	Halle (Leitung der Fachgruppe)
Burgemeister, Maik	Bernburg
Degen, Antje	Hettstedt
Eschrich, Mario	Halle
Dr. Herper, Henry	Magdeburg (wissenschaftliche Beratung)

Herausgeber: Ministerium für Bildung des Landes Sachsen-Anhalt
 Turmschanzenstr. 32
 39114 Magdeburg

In Kraft seit 2017, Anpassung 2022

Inhaltsverzeichnis

Seite

1	Bildung und Erziehung im Wahlpflichtfach Informatik	4
2	Kompetenzentwicklung im Fach Informatik	6
3	Kompetenzentwicklung in den Schuljahrgängen	15
3.1	Übersicht.....	15
3.2	Schuljahrgang 9	16
3.3	Schuljahrgang 10 (Einführungsphase)	20
3.4	Schuljahrgänge 11/12 (Qualifikationsphase)	24

1 Bildung und Erziehung im Wahlpflichtfach Informatik

In den letzten Jahrzehnten hat die rasante Entwicklung auf dem Gebiet der Informatik wie keine andere Wissenschaft zu gravierenden Veränderungen im gesamtgesellschaftlichen Bereich geführt, deren Auswirkungen auf das tägliche Leben unverkennbar sind. Die Fortschritte in den Teilgebieten der Informatik – der theoretischen, praktischen, angewandten und technischen Informatik – beeinflussen alle Forschungsgebiete. Es entstanden zahlreiche neue interdisziplinäre Wissenschaften wie zum Beispiel Wirtschaftsinformatik, Geoinformatik, Bioinformatik und Medieninformatik. Durch die Wechselwirkung von Informatik und Gesellschaft sind auch in Zukunft tiefgreifende Veränderungen in allen Lebensbereichen durch die Digitalisierung zu erwarten.

*Teilhabe und
Teilnahme am
gesellschaftlichen
Leben*

Im Fach Informatik erwerben die Schülerinnen und Schüler informatische Kompetenzen, welche die Grundlage für eine vertiefte berufliche Bildung in allen Berufsfeldern und für die allgemeine Studierfähigkeit darstellen.

Anhand exemplarischer Praxisbeispiele werden vielfältige Bezüge zu Berufen und Studienrichtungen hergestellt.

Die kompetente Nutzung von Informatiksystemen und das Verständnis ihrer Funktionsweise ermöglichen den Schülerinnen und Schülern eine aktive Teilhabe an der gesellschaftlichen Entwicklung auf dem Weg zur global vernetzten Informationsgesellschaft. Sie ist Voraussetzung für die Mitgestaltung der Informationsgesellschaft mit all ihren Chancen und Risiken, Möglichkeiten und Grenzen.

*Lebenswelt-
bezogenes
Lernen*

Die informatische Bildung ermöglicht den Schülerinnen und Schülern in unserer gegenwärtigen und zukünftigen Gesellschaft informatische Sachverhalte zu verstehen und diese sowohl selbstständig als auch im Team zu bewältigen. Sie schließt das grundlegende Verständnis moderner digitaler Hilfsmittel sowie die Beherrschung grundlegender Methoden und Werkzeuge der Informatik, die für die Bewältigung unterschiedlichster Berufs- und Alltagssituationen notwendig sind, ein. Mit der interdisziplinären Anwendung dieser informatischen Bildung und der Einbettung fachlicher Zusammenhänge in lebensweltliche Bezüge der Schülerinnen und Schüler wird ein wesentlicher Beitrag zum Verständnis der Lebenswelt und zur Studien- und Berufsorientierung geleistet.

Das Wahlpflichtfach Informatik trägt zur Ausprägung der Studierfähigkeit und damit zur Allgemeinen Hochschulreife bei, indem die Schülerinnen und Schüler

*Allgemeine
Hochschulreife*

- sich mit dem Konzept, der Implementierung und dem Einsatz von Informatiksystemen für unterschiedliche Anwendungsgebiete beschäftigen,
- eigenständig oder im Team informatische Frage- und Problemstellungen erkennen, analysieren, Lösungswege planen, Lösungen erarbeiten, kontrollieren und auswerten,
- längerfristige Lernprozesse, z. B. bei der Erstellung von Facharbeiten oder der Durchführung von Projekten, sowohl praxisnah als auch ergebnisorientiert planen und realisieren sowie
- eigene Problemlösungsprozesse dokumentieren sowie Arbeitsergebnisse wissenschaftlichen Normen entsprechend darstellen und präsentieren, wobei sie Fachsprache und Bildungssprache in bewusster Abgrenzung zur Alltagssprache ziel-, sach- und adressatengerecht verwenden.

Die Informatik ist sowohl Grundlagenwissenschaft als auch angewandte Wissenschaft.

*Wissenschafts-
propädeutisches
Arbeiten*

Das Fach Informatik führt in der Sekundarstufe I und im Besonderen in der Qualifikationsphase in die Wissenschaft Informatik, deren Fachsprache und Denkweisen ein, indem

- die Prinzipien der Informatik exemplarisch an vorhandenen Informatiksystemen betrachtet werden,
- Informations- und Kommunikationstechniken als Medium, Werkzeug und Inhalt des Lernprozesses genutzt werden,
- die Vorgehensweise des Problemlösens unter Verwendung informatischer Methoden erlernt wird und
- praktische Erfahrungen beim Implementieren gefundener Lösungen erworben werden.

2 Kompetenzentwicklung im Fach Informatik

Kompetenzmodell Im Wahlpflichtfach Informatik werden die zu erwerbenden Kompetenzen mit dem Kompetenzmodell basierend auf den Empfehlungen der Gesellschaft für Informatik (GI) e. V.¹ beschrieben.



Abb. 1: Verflechtung der Kompetenzbereiche mit Inhaltsbereichen.

Fünf Kompetenzbereiche bilden die Fachkompetenz ab:

- Modellieren und Implementieren,
- Begründen und Bewerten,
- Strukturieren und Vernetzen,
- Kommunizieren und Kooperieren,
- Darstellen und Interpretieren.

Die Kompetenzbereiche sind mit den folgenden Inhaltsbereichen verknüpft:

- Information und Daten,
- Algorithmen,
- Sprachen und Automaten,
- Informatiksysteme,
- Informatik, Mensch und Gesellschaft.

¹ Gesellschaft für Informatik (GI) e. V. (Hrsg.) (2016): Bildungsstandards Informatik für die Sekundarschule II.

Die Schülerinnen und Schüler wenden informatische Denk- und Arbeitsweisen auf konkrete Aufgabenstellungen aus ihrer Erfahrungswelt an. Sie erfassen Situationen, erstellen informatische Modelle, setzen diese mit geeigneten Werkzeugen um und konfigurieren Werkzeuge aufgabenangemessen. Sie beziehen die Lösungen wieder auf die Situation und reflektieren so die informatische Modellierung.

Modellieren und Implementieren

Am Ende der	
Einführungsphase	Qualifikationsphase
können die Schülerinnen und Schüler in der Regel:	
– Handlungsabläufe unter Verwendung algorithmischer Grundbausteine beschreiben	– Prozesse unter Verwendung einer Modellierungssprache beschreiben
– einfache informatische Modelle zu Sachverhalten aus der Lebenswelt unter Verwendung objektorientierter Techniken entwickeln	– komplexe informatische Modelle unter Verwendung objektorientierter Techniken entwickeln
– Modelle mit grafischen und textbasierten Werkzeugen implementieren	– Modelle mit textbasierten Werkzeugen objektorientiert implementieren
– Modelle und ihre Implementation analysieren	– Modelle und ihre Implementation verifizieren, validieren und reflektieren
– Informationen auf Daten abbilden, mittels geeigneter Datentypen darstellen und zugehörige Operationen anwenden	– Daten in Datenbanken unter Verwendung von Entity-Relationship-Modellen interpretieren, beschreiben und implementieren

Begründen und Bewerten

Die Schülerinnen und Schüler erkennen Informatiksysteme, äußern sich begründet über informatische Zusammenhänge unterschiedlicher Komplexität und leiten Fragestellungen und Hypothesen zur Wirkungsweise sowie Funktionalität ab. Sie erklären unter Verwendung der Fachsprache Beziehungen und Gesetzmäßigkeiten auf unterschiedlichen Niveaustufen. Die Schülerinnen und Schüler bewerten kriteriengeleitet informatische Sachverhalte und begründen ihre Bewertung.

Am Ende der Einführungsphase	Qualifikationsphase
können die Schülerinnen und Schüler in der Regel:	
– Entscheidungen im Rahmen eines Problemlösungszyklus begründen	– Entscheidungen im Rahmen eines Problemlösungszyklus begründen und bewerten
– gewählte Objekt- und Klassenstrukturen begründen	– einen objektorientierten Entwurf auf Basis von Qualitätskriterien erläutern und begründen
– Testergebnisse bewerten und daraus Veränderungen ableiten	– eigene Lösungen bewerten und weitere Arbeitsschritte ableiten
– gegebene Informatiksysteme bezüglich ihrer Praxisrelevanz bewerten	– Chancen, Risiken und Missbrauchsmöglichkeiten von Informatiksystemen bewerten
– exemplarisch Grenzen von Informatiksystemen benennen und reflektieren	– Möglichkeiten und Grenzen in einer von Informatiksystemen dominierten Welt erkennen und bewerten
– Schutz- und Sicherheitsmaßnahmen in Netzwerken begründen, anwenden und bewerten	– Datenhaltung unter den Gesichtspunkten des Datenschutzes sowie der Datensicherheit beurteilen
– Nachhaltigkeitsindikator eines Informatiksystems exemplarisch reflektieren	– Folgen der Digitalisierung im Sinne der Nachhaltigkeit erkennen, bewerten und Handlungsstrategien ableiten

Die Schülerinnen und Schüler wenden informatische Prinzipien zum Strukturieren von Sachverhalten an. Sie modularisieren und hierarchisieren diese Sachverhalte und verknüpfen sie in fach- und erfahrungsweltbezogenen Anwendungsbereichen.

Strukturieren und Vernetzen

Am Ende der	
Einführungsphase	Qualifikationsphase
können die Schülerinnen und Schüler in der Regel:	
– Zusammenhänge zwischen Daten und Information darstellen	– Daten in geeigneter Form strukturieren und verwalten
– Algorithmen strukturieren	– Abläufe und Zusammenhänge objektorientiert strukturieren
– Informatiksysteme exemplarisch analysieren sowie Daten- und Funktionalitätsstrukturen ableiten	– Informatiksysteme aus technischer, informatischer, öko-logischer und sozialer Sicht analysieren
– Problemstellungen exemplarisch analysieren, in Teilprobleme zerlegen, Lösungsansätze strukturieren	– Problemstellungen analysieren, in Teilprobleme zerlegen, notwendige Schnittstellen vereinbaren und Lösungsansätze in Hinblick auf Synergieeffekte und Softwareergonomie strukturieren

*Kommunizieren
und Kooperieren*

Die Schülerinnen und Schüler tauschen sich konstruktiv aus, vertreten, reflektieren und korrigieren gegebenenfalls den eigenen Standpunkt. Sie kommunizieren über informatische Sachverhalte, Gegenstände und Beziehungen unter Verwendung der Fachsprache und in geeigneten Darstellungsformen. Die Schülerinnen und Schüler kooperieren bei der Bearbeitung informatischer Probleme und präsentieren ihre Lern- und Arbeitsergebnisse sach-, adressaten- und situationsgerecht.

Am Ende der	
Einführungsphase	Qualifikationsphase
können die Schülerinnen und Schüler in der Regel:	
– einen Problemlösungszyklus exemplarisch reflektieren	– komplexe Projekte organisieren, kooperativ bearbeiten, reflektieren und dokumentieren
– Algorithmen, deren Nutzen und Auswirkungen sowie ihre notwendigen Ressourcen diskutieren, dokumentieren, präsentieren und in ihrer Lebenswelt reflektieren	– einen objektorientierten Entwurf sprachlich und fachsprachlich präzise reflektieren und dabei UML – Diagramme als Kommunikationsmittel nutzen
– Informatiksysteme in ihrer Lebenswelt untersuchen, beschreiben und gemeinsam anwenden	– gesellschaftliche Auswirkungen der Einführung und Nutzung von Informatiksystemen diskutieren
– lokale und netzwerkbasierte Datenverwaltung unter den Aspekten des Datenschutzes und der Datensicherheit vergleichen und gemeinsam anwenden	– Datenbankmodellierung unter den Aspekten des Datenschutzes und der Datensicherheit kooperativ anwenden und diskutieren

Die Schülerinnen und Schüler stellen ihre Vorgehensweisen angemessen und nachvollziehbar unter Verwendung von Beschreibungsmitteln der Informatik dar. Sie beschreiben und interpretieren fachspezifisch unterschiedliche Systeme und Prozesse unter Nutzung der Fachsprache der Informatik.

Darstellen und Interpretieren

Am Ende der	
Einführungsphase	Qualifikationsphase
können die Schülerinnen und Schüler in der Regel:	
– Daten darstellen und bestehende Formen interpretieren	– komplexe Datenstrukturen darstellen und bestehende Formen interpretieren
– ausgewählte Parameter von Informatiksystemen darstellen und interpretieren	– Parameter von Informatiksystemen darstellen, reflektieren und interpretieren
– Handlungsabläufe unter Verwendung algorithmischer Grundbausteine darstellen und interpretieren	– Projekte in sprachlich differenzierter und grafischer Form dokumentieren sowie Projektdokumentationen interpretieren
– verschiedene Darstellungsformen zur Beschreibung der Funktionalität von Systemen interpretieren und ineinander überführen	– spezifische Diagramme zur Beschreibung des Verhaltens von Systemen anwenden
– Fehlermitteilungen der verwendeten Entwicklungssysteme interpretieren	– Fehlermitteilungen interpretieren und unter Verwendung geeigneter Hilfesysteme Lösungsstrategien entwerfen

*Beitrag zur
Entwicklung der
Schlüssel-
kompetenzen*

Im Wahlpflichtfach Informatik werden über die fachspezifischen Kompetenzen hinaus übergreifende Schlüsselkompetenzen entwickelt, die für eine aktive Teilnahme und Teilhabe am gesellschaftlichen Leben notwendig sind.

Die Auseinandersetzung mit den Wechselwirkungen zwischen Informatiksystemen und ihrer gesellschaftlichen Einbettung stärkt das Verständnis für wirtschaftliche, soziale und gesellschaftliche Zusammenhänge auch unter dem Aspekt der nachhaltigen Entwicklung. Der Umgang mit Informatiksystemen erfolgt in Übereinstimmung mit gesellschaftlichen Normen und berücksichtigt die Persönlichkeitsrechte anderer. Damit wird ein Beitrag zur Entwicklung eines Rechtsbewusstseins und zur Einhaltung gesellschaftlicher Normen geleistet.

Schülerinnen und Schüler erwerben umfassende Kompetenzen im logischen Denken und stärken ihre Abstraktionsfähigkeit. Sie sind in der Lage, Lösungsstrategien und komplexe Abläufe formal exakt darzustellen und zu interpretieren.

Die Arbeit in den verschiedensten Projekten in allen Schuljahrgängen erfordert den norm-, sach-, situations- und adressatengerechten Einsatz von Fach- und Bildungssprache in mündlicher und schriftlicher Form und trägt somit zur Entwicklung der Kommunikationskompetenz bei. Schülerinnen und Schüler entwickeln für sich im Rahmen von Projekten ein effizientes Zeitmanagement. Das schließt Selbstorganisation, Reflexion des Lehrens und Lernens sowie das Ableiten von Rückschlüssen für sich selbst und für die Lerngruppe bezüglich der Ziele, Strategien, Ergebnisse und Lernhaltungen im Projektverlauf ein. Die Bereitschaft, Hindernisse im Verlauf des Problemlösungszyklus in Kooperation mit anderen zu überwinden, stärkt die Sozialkompetenz. Das Bearbeiten komplexer Aufgabenstellungen erfordert und fördert kreative Denkansätze.

Im Informatikunterricht nutzen die Schülerinnen und Schüler grundsätzlich verständig, zielgerichtet und reflektiert digitale Werkzeuge und Endgeräte zum Gewinnen, Verarbeiten, Bewerten sowie Präsentieren von Informationen und zum Austauschen (Kollaborieren). Sie erwerben zielgerichtet Anwendungskompetenzen und darüber hinaus ein vertieftes Verständnis für die den digitalen Werkzeugen zugrunde liegenden Prinzipien, den Aufbau und die Funktionsweise von Informatiksystemen. Diese Kompetenzen bilden die Basis für mögliche Entwicklungen eigener neuer digitaler Werkzeuge.

*Beitrag zur Bildung
in der digitalen
Welt*

Die Schülerinnen und Schüler entwickeln dabei folgende Kompetenzen:

- Zusammenhang zwischen Daten und Informationen beschreiben,
- Daten und Informationen analysieren, interpretieren und kritisch Bewerten,
- Informationen zusammenfassen und in Daten codieren,
- Daten organisieren und strukturiert verwalten,
- technische Grundlagen der Datenverwaltung analysieren und anwenden,
- technische Systeme als Informatiksysteme analysieren, modellieren, simulieren und realisieren,
- Funktionsweisen und grundlegende Prinzipien der digitalen Welt verstehen und reflektieren,
- algorithmische Strukturen für Informatiksysteme erkennen und darstellen,
- eine strukturierte, algorithmische Beschreibung zur Lösung eines Problems planen und implementieren,
- Möglichkeiten und Grenzen von Informatiksystemen in einer digital dominierten Welt erkennen und bewerten,
- Folgen der Digitalisierung im Sinne der Nachhaltigkeit erkennen, bewerten und Handlungsstrategien ableiten.

*Beitrag zur Bildung
für nachhaltige
Entwicklung*

Der Informatikunterricht leistet einen Beitrag zum Erwerb von übergreifenden Kompetenzen für eine zukunftsfähige Entwicklung entsprechend der Weltnachhaltigkeitsziele² durch die Umsetzung von Projekten in Kontexten globaler Entwicklungsdimensionen (ökologisch, ökonomisch, sozial und politisch).

Schülerinnen und Schüler nutzen Informatiksysteme in unterschiedlichen Lebensbereichen und entwickeln Kompetenzen für ein eigenverantwortliches Handeln im gesellschaftlichen Kontext. Dabei erkennen sie den permanenten Ressourcenverbrauch der Informatiksysteme. Die Schülerinnen und Schüler bewerten Chancen und Risiken der weltweiten Informationsverbreitung, entwickeln ein Verständnis der Mechanismen des kollaborativen Arbeitens in der Lern- und Arbeitswelt und reflektieren die Auswirkungen von Algorithmen auf ihre Lebenswelt.

² Vereinte Nationen (Hrsg.) (2016): Ziele für eine nachhaltige Entwicklung. New York, S. 3–17.

3 Kompetenzentwicklung in den Schuljahrgängen

3.1 Übersicht

Schuljahrgänge	Kompetenzschwerpunkte
Schuljahrgang 9	– Algorithmen interpretieren und entwickeln
	– Daten codieren und ihre Verarbeitung verstehen
Schuljahrgang 10 (Einführungsphase)	– Algorithmen- und Datenstrukturen implementieren und testen
	– Technische und theoretische Grundlagen von Informatiksystemen verstehen
Schuljahrgänge 11/12 (Qualifikationsphase)	– Kurs 1: Objektorientiertes Modellieren
	– Kurs 2: Daten erfassen, strukturieren und verarbeiten
	– Kurs 3: Software Engineering und Projektarbeit
	– Kurs 4: Aktuelle Entwicklungen der Informatik im gesellschaftlichen Kontext

3.2 Schuljahrgang 9

Kompetenzschwerpunkt: Algorithmen interpretieren und entwickeln	
Modellieren und Implementieren	<ul style="list-style-type: none"> – Handlungsabläufe formulieren – Algorithmen entwerfen und verbal darstellen – Variable als Speicherplatz beschreiben und anwenden – Algorithmen in einer visuellen Programmiersprache implementieren – reale endliche Automaten analysieren, zugehörige Automatenmodelle entwerfen
Begründen und Bewerten	<ul style="list-style-type: none"> – Algorithmen testen und weiterentwickeln – gegebene zustandsbasierte Systeme bezüglich ihrer Praxisrelevanz sowie auch unter ethischen Aspekten diskutieren und bewerten – Entscheidungen im Rahmen eines Problemlösungszyklus begründen
Strukturieren und Vernetzen	<ul style="list-style-type: none"> – Problemstellungen analysieren, in Teilprobleme zerlegen und Lösungsansätze strukturieren – Algorithmen strukturieren – Auswirkungen von Algorithmen auf die Lebenswelt exemplarisch erkennen sowie für die Meinungsbildung und Entscheidungsfindung nutzen
Kommunizieren und Kooperieren	<ul style="list-style-type: none"> – einen Problemlösungszyklus reflektieren – Algorithmen diskutieren, dokumentieren und präsentieren – Nutzen und Gefahren von Algorithmen in ihrer eigenen Lebenswelt reflektieren – notwendige Ressourcen von Algorithmen diskutieren – reale Automaten beschreiben
Darstellen und Interpretieren	<ul style="list-style-type: none"> – Algorithmen in der Lebenswelt in unterschiedlichen Darstellungsformen erkennen und interpretieren – algorithmische Bausteine benennen, unter Verwendung digitaler Werkzeuge darstellen und auf Handlungsabläufe anwenden – Automatentafel und Zustandsdiagramm endlicher Automaten interpretieren und wechselseitig übertragen

Grundlegende Wissensbestände	
<ul style="list-style-type: none"> – Begriffe: Algorithmus, Programmiersprache, Entwicklungsumgebung – Eigenschaften von Algorithmen – algorithmische Grundbausteine (Anweisung, Anweisungsfolge, Verzweigung, Wiederholung) – Problemlösungszyklus – zustandsbasierte Systeme (Definition, Darstellungsformen der Zustandsüberföhrungsfunktion) – Modell endlicher Automat 	
Mögliche Projekte	
<ul style="list-style-type: none"> – Algorithmen erleben – algorithmische Problemlösungen implementieren – mit externen Geräten kommunizieren (Verwendung von Sensoren und Aktoren) – endliche Automaten implementieren 	
Bezüge zu den fächerübergreifenden Themen (gemäß Grundsatzband Kap. 4)	
Produktion und Konsum	Deutsch, Französisch, Biologie, Chemie, Sozialkunde, Evangelischer Religionsunterricht, Katholischer Religionsunterricht, Geographie, Kunst

Kompetenzschwerpunkt: Daten codieren und ihre Verarbeitung verstehen	
Modellieren und Implementieren	<ul style="list-style-type: none"> – Informationen auf Daten abbilden – reale Rechner analysieren und ein Funktionsmodell erstellen
Begründen und Bewerten	<ul style="list-style-type: none"> – Dateiformate benennen und analysieren sowie diesen Anwendungen zuordnen – Lebensdauer von Daten technisch und rechtlich beurteilen – technische Parameter von Hardwarekomponenten bewerten – korrekte Funktionsweise von Hardwarekomponenten beurteilen – Nachhaltigkeitsindikator eines Informatiksystems exemplarisch reflektieren – Informatiksysteme im historischen Kontext einordnen – logische Grundschaltungen analysieren und deren Funktionalität bewerten
Strukturieren und Vernetzen	<ul style="list-style-type: none"> – Zusammenhänge zwischen Daten und Information darstellen – wesentliche Hardware-, Software- und Netzwerkkomponenten benennen sowie deren Aufgaben erläutern
Kommunizieren und Kooperieren	<ul style="list-style-type: none"> – Informatiksysteme in ihrer Lebenswelt erkennen, benennen, untersuchen und anwenden – Daten lokal und in Netzwerken verwalten – Aufgaben gemeinsam innerhalb von Netzwerken bearbeiten – grundlegende Verhaltensnormen in Netzwerken beachten – die Funktionsweise einfacher logischer Grundschaltungen beschreiben und erklären
Darstellen und Interpretieren	<ul style="list-style-type: none"> – Einheiten für die Datenmenge auf die Speicherkapazität von Datenträgern, Dateigrößen und Übertragungsvolumen anwenden – Betriebssysteme vergleichen und Grundfunktionen erläutern

Grundlegende Wissensbestände	
<ul style="list-style-type: none"> – Meilensteine der Informatik – Beispiele für Informatiksysteme – Betriebssysteme und ausgewählte Aufgaben (z. B. Nutzerverwaltung, Dateiverwaltung, Prozessverwaltung) – EVA–Prinzip und zugehörige Hardwarekomponenten – Rechnermodell nach J. v. Neumann – Daten (Begriff, Darstellungsformen, Nutzerschnittstellen, Speicherung, Sicherheit) – Datenverwaltung in Netzwerken – Maßeinheiten und Codierung von Daten (Zahlensysteme) – Informationsbegriff (Information als kontextbezogene Interpretation von Daten) – logische Funktionen (NICHT, UND, ODER) und deren Anwendung 	
Mögliche Projekte	
<ul style="list-style-type: none"> – logische Grundschaltungen simulieren und implementieren (Schaltnetz, Schaltwerk) – Dateiformate analysieren und transformieren – ein reales Informatiksystem analysieren 	
Bezüge zu den fächerübergreifenden Themen (gemäß Grundsatzband Kap. 4)	
Klimawandel und Klimaschutz	Deutsch, Latein, Biologie, Chemie, Physik, Sozialkunde, Evangelischer Religionsunterricht, Katholischer Religionsunterricht, Geographie, Sport
Innovation, Infrastruktur und Digitalität	Deutsch, Latein, Griechisch, Biologie, Chemie, Physik, Geographie
Produktion und Konsum	Deutsch, Französisch, Biologie, Chemie, Sozialkunde, Evangelischer Religionsunterricht, Katholischer Religionsunterricht, Geographie, Kunst

3.3 Schuljahrgang 10 (Einführungsphase)

Kompetenzschwerpunkt: Algorithmen- und Datenstrukturen implementieren und testen	
Modellieren und Implementieren	<ul style="list-style-type: none"> – Klassen, Methoden und Attribute in einer textbasierten Programmiersprache unter Verwendung der algorithmischen Grundbausteine implementieren und testen – Instanzen von implementierten Klassen erzeugen – Daten mittels geeigneter Datentypen abbilden und zugehörige Operationen anwenden – zwischen einfachen und strukturierten Datentypen unterscheiden und diese anwenden – das Variablenkonzept anwenden
Begründen und Bewerten	<ul style="list-style-type: none"> – gewählte Klassenstrukturen begründen – Testergebnisse werten und daraus Veränderungen ableiten – Entscheidungen im Rahmen eines Problemlösungszyklus am Beispiel eines Such- und eines Sortierverfahrens begründen und in Teilen bewerten
Strukturieren und Vernetzen	<ul style="list-style-type: none"> – Klassen als Möglichkeit der Datenstrukturierung in Form von Eigenschaften nutzen – Objekte als Instanzen einer Klasse mit konkreten Daten beschreiben – Problemlösungsstrategien strukturieren und beschreiben
Kommunizieren und Kooperieren	<ul style="list-style-type: none"> – Algorithmen sachgerecht dokumentieren und präsentieren – einfache Formen der Fehlerbehandlung in Programmen diskutieren und nutzen
Darstellen und Interpretieren	<ul style="list-style-type: none"> – Klasse, Objekt, Attribut und Methode mittels der Modellierungssprache UML darstellen – Algorithmen unter Verwendung algorithmischer Grundbausteine grafisch darstellen – Fehlermitteilungen interpretieren

Grundlegende Wissensbestände	
<ul style="list-style-type: none"> – algorithmische Grundbausteine sowie deren formale Beschreibungsformen und Entsprechungen in einer Programmiersprache – Syntax einer Programmiersprache und formale Beschreibungsformen – Variablenkonzept (Speichermodell) – Zuweisungs- und Vergleichsoperatoren, arithmetische und logische Operatoren – Grundbegriffe der objektorientierten Programmierung: Klasse, Objekt, Instanz, Attribut, Methode (mit und ohne Rückgabewert) – UML-Diagramme (Klassendiagramm, Objektdiagramm) 	
Mögliche Projekte	
<p>Phasen eines Problemlösungszyklus an einem der folgenden Verfahren umsetzen:</p> <ul style="list-style-type: none"> – Suchverfahren – Sortierverfahren – Verschlüsselungsverfahren 	
Bezüge zu den fächerübergreifenden Themen (gemäß Grundsatzband Kap. 4)	
Innovation, Infrastruktur und Digitalität	Deutsch, Englisch, Französisch, Italienisch, Latein, Griechisch, Biologie, Physik, Evangelischer Religionsunterricht, Geographie

Kompetenzschwerpunkt: Technische und theoretische Grundlagen von Informatiksystemen verstehen	
Modellieren und Implementieren	<ul style="list-style-type: none"> – Regeln für einfache formale Sprachen aufstellen und diese als Akzeptor implementieren – Worte einer formalen Sprache aus der Kenntnis ihrer Grammatik erzeugen – Infrastruktur von Heim- bzw. Schulnetzwerken exemplarisch unter Beachtung aktueller Sicherheitsaspekte modellieren – Heim- bzw. Schulnetzwerke simulieren – ausgewählte Netzwerkdienste unter Nutzung zugehöriger Protokolle in einer geeigneten Umgebung einrichten und nutzen
Begründen und Bewerten	<ul style="list-style-type: none"> – ein Wort aus Terminalsymbolen eines Alphabetes zu einer vorgegebenen Sprache als zugehörig erkennen oder verneinen – die Existenz von Fehlerzuständen in Akzeptoren begründen – die Notwendigkeit einer auf exakten Regeln basierenden Sprache, welche eindeutig maschinell ausführbar ist, begründen – exemplarisch Grenzen automatisierter Systeme benennen und reflektieren – die Notwendigkeit von Schutz- und Sicherheitsmaßnahmen in Netzwerken begründen und zielgerichtet anwenden – Verhaltensweisen in Netzwerken reflektieren und Schlussfolgerungen für das eigene Handeln ableiten
Strukturieren und Vernetzen	<ul style="list-style-type: none"> – gegebene Informatiksysteme analysieren sowie Daten- und Funktionsstrukturen ableiten – zwischen natürlichen und formalen Sprachen unterscheiden – die Funktionsweise von TCP/IP-Netzwerken exemplarisch erläutern – TCP/IP-Netzwerke exemplarisch strukturieren und simulieren – Dienste, zugehörige Protokolle und Ports exemplarisch erläutern
Kommunizieren und Kooperieren	<ul style="list-style-type: none"> – Anwendungsfälle formaler Sprachen nennen und deren Beitrag zur Kooperation erkennen – Bedeutung der Vernetzung von digitalen Systemen sowie deren Anwendungsfälle erläutern – Funktionalität von Diensten unter Nutzung eines Schichtenmodells exemplarisch beschreiben – lokale und netzwerkbasierte Datenverwaltung auch unter dem Aspekt der Datensicherheit vergleichen und anwenden
Darstellen und Interpretieren	<ul style="list-style-type: none"> – die Funktionalität von Informatiksystemen mittels geeigneter Diagramme abbilden – die formale Sprache, die ein endlicher Automat akzeptiert, in geeigneter Form darstellen – einfache formale Sprachen exemplarisch mittels Syntaxdiagrammen darstellen

Grundlegende Wissensbestände	
<ul style="list-style-type: none"> – Grenzen automatisierter Systeme (z. B. Turing Test) – formale Sprachen (Anwendungsgebiete, Vergleich mit natürlichen Sprachen, Semantik und Syntax einer Sprache, Beschreibung mittels Syntaxdiagrammen) – TCP/IP–Netzwerke (Aufbau, Funktionsweise, Schichtenmodell) – aktive und passive Netzwerkkomponenten – Adressierung, Adressauflösung und Routing in Netzwerken – Kommunikation in kabelgebundenen und drahtlosen Netzwerken – Peer-to-Peer-Netzwerk, Client-Server-Netzwerk – Schutz und Sicherheit in Netzwerken (Firewall, Virenschutz, Backup) – grundlegende Verhaltensregeln in Netzwerken 	
Mögliche Projekte	
<ul style="list-style-type: none"> – Netzwerkstrukturen mit geeigneter Software simulieren oder real aufbauen – Smart-Home analysieren – Modell eines Automaten (Akzeptor) zur Analyse von formalen Sprachen entwerfen (z. B. MiniHTML, MiniAssembler, ...) 	
Bezüge zu den fächerübergreifenden Themen (gemäß Grundsatzband Kap. 4)	
Innovation, Infrastruktur und Digitalität	Deutsch, Englisch, Französisch, Italienisch, Latein, Griechisch, Biologie, Physik, Evangelischer Religionsunterricht, Geographie

3.4 Schuljahrgänge 11/12 (Qualifikationsphase)

Kurs 1: Objektorientiertes Modellieren	
Modellieren und Implementieren	<ul style="list-style-type: none"> – Problemstellungen unter Verwendung von Anwendungsfalldiagrammen analysieren und dokumentieren – objektorientierte Analysen durchführen sowie beteiligte Objekte, ihre Eigenschaften, Operationen und vorhandene Beziehungen erkennen und Klassen ableiten – Prozesse mithilfe von Aktivitäts- oder Zustandsdiagrammen modellieren – Klassen mit ihren Methoden in eine Programmiersprache implementieren – Objekte erzeugen, in dynamischen Datenstrukturen verwalten, auf externen Datenträgern speichern und von diesen laden – die Kommunikation zwischen und innerhalb von Objekten organisieren – Module und Bibliotheken in eigene Problemlösungen integrieren
Begründen und Bewerten	<ul style="list-style-type: none"> – die gewählte Abstraktion begründen – eigene Lösungen bewerten und weitere Arbeitsschritte ableiten – gegebene Programme hinsichtlich ihrer Grundkonzepte analysieren
Strukturieren und Vernetzen	<ul style="list-style-type: none"> – einen objektorientierten Entwurf erstellen – das Geheimnisprinzip und das Vererbungskonzept von Klassen auf einfache Sachverhalte anwenden – Probleme in Teilprobleme exemplarisch zerlegen
Kommunizieren und Kooperieren	<ul style="list-style-type: none"> – UML–Diagramme als Kommunikationsmittel in der Gruppe nutzen – einen objektorientierten Entwurf sprachlich und fachsprachlich präzise reflektieren
Darstellen und Interpretieren	<ul style="list-style-type: none"> – Klassen und deren Beziehungen in UML darstellen – Klassendefinitionen und Klassenbeschreibungen interpretieren – Fehlermitteilungen interpretieren und unter Verwendung geeigneter Hilfesysteme Lösungsstrategien entwerfen

Grundlegende Wissensbestände	
<ul style="list-style-type: none"> – UML-Diagramme (Anwendungsfalldiagramm sowie Aktivitäts- oder Zustandsdiagramm) – Parameterlisten von Methoden (Wert- und Referenzparameter) – Erzeugen und Entfernen von Objekten (Konstruktor: Aufbau, Funktion, Überladung und ggf. Destruktor) – Testverfahren und deren Anwendung 	
Mögliche Projekte	
<ul style="list-style-type: none"> – eine Simulation aufbauen (z. B. Ampelschaltung, endlicher Automat, Warteschlangensysteme) – ein Computerspiel entwickeln – eine verteilte Anwendung entwickeln 	
Bezüge zu den fächerübergreifenden Themen (gemäß Grundsatzband Kap. 4)	
<p>Globale Partnerschaften und Europa</p>	<p>Englisch, Französisch, Spanisch, Russisch, Latein, Physik, Geschichte, Sozialkunde, Evangelischer Religionsunterricht, Geographie</p>

Kurs 2: Daten erfassen, strukturieren und verarbeiten	
Modellieren und Implementieren	<ul style="list-style-type: none"> – für anwendungsrelevante Problemstellungen Entitäten, zugehörige Attribute, Relationen und Kardinalitäten ermitteln – relationale Datenbankschemata unter Nutzung des Entity-Relationship-Modells entwerfen – ein Entity-Relationship-Modell modifizieren – Primär- und Sekundärschlüssel bestimmen – relationale Datenbankschemata bis zur dritten Normalform überführen – eine Abfragesprache anwenden – Daten rechtskonform erfassen und verarbeiten
Begründen und Bewerten	<ul style="list-style-type: none"> – Funktionen von Datenbankmanagementsystemen exemplarisch begründen – Datenbankmodellierung und objektorientierte Modellierung auch bezüglich der Methodik und Begriffswelt vergleichen und bewerten – die Verwendung von Datenstrukturen begründen – Datenbankmodelle bewerten – Schlüsselattribute bewerten
Strukturieren und Vernetzen	<ul style="list-style-type: none"> – Daten in geeigneter Form strukturieren und verwalten – Relationen zwischen Entitäten erkennen und diese in geeigneter Form darstellen
Kommunizieren und Kooperieren	<ul style="list-style-type: none"> – Entity-Relationship-Diagramme diskutieren und erstellen – eine Datenbankmodellierung analysieren und erläutern – die Eigenschaften eines normalisierten Datenbankschemas erläutern
Darstellen und Interpretieren	<ul style="list-style-type: none"> – Daten in verschiedenen Formen darstellen und bestehende Formen interpretieren – Entitäten, ihre Attribute und die Beziehungen zwischen Entitäten mit Kardinalitäten in einem Entity-Relationship-Diagramm grafisch darstellen

Grundlegende Wissensbestände	
<ul style="list-style-type: none"> – Zusammenhang von Informationen und Daten – Datenbanksystem (Datenbank, Datenbankmanagementsystem) – Funktionen von Datenbankmanagementsystemen – Datenbankentwurf (Anforderungsanalyse, konzeptioneller-, logischer- und physischer Entwurf) am Beispiel eines Entity-Relationship-Datenbankmodells – Überführung des ER-Modells in ein relationales Schema – Normalisierung einer relationalen Datenbank bis zur dritten Normalform – Datenbankanomalien (Lösch-, Einfüge-, Aktualisierungsanomalie) – Abfragesprache zur Ausführung von Operationen auf Daten 	
Mögliche Projekte	
<ul style="list-style-type: none"> – Datenbankentwurf zu einer geeigneten Miniwelt erstellen oder erweitern 	
Bezüge zu den fächerübergreifenden Themen (gemäß Grundsatzband Kap. 4)	
<p>Innovation, Infrastruktur und Digitalität</p>	<p>Deutsch, Englisch, Französisch, Italienisch, Biologie, Chemie, Physik, Geschichte, Sozialkunde, Ethikunterricht, Evangelischer Religionsunterricht, Geographie</p>

Kurs 3: Software Engineering und Projektarbeit	
Modellieren und Implementieren	<ul style="list-style-type: none"> – selbstständig Analysen im Rahmen eines Problemlösungszyklus durchführen und daraus benötigte Objekte und Klassen ableiten – die Implementierung der Teillösungen selbstständig oder mithilfe der Teampartner realisieren – Teillösungen zu einer Gesamtlösung zusammenführen – Teillösungen sowie Endprodukt verifizieren und validieren
Begründen und Bewerten	<ul style="list-style-type: none"> – Entscheidungen im Rahmen eines Problemlösungszyklus begründen und bewerten – dem Auftraggeber den objektorientierten Entwurf sowie die zugehörigen Spezifikationen auf Basis von Qualitätskriterien erläutern und begründen – die Nachhaltigkeit des Softwareprojektes abschätzen und begründen – notwendige Abstraktionen selbstständig vornehmen und begründen – die Auswahl der genutzten Algorithmen begründen und hinsichtlich der Zielführung sowie deren Effizienz bewerten – die Datenhaltung in vernetzten Systemen analysieren und diese unter den Gesichtspunkten des Datenschutzes sowie der Datensicherheit beurteilen
Strukturieren und Vernetzen	<ul style="list-style-type: none"> – ein Problem in Teilprobleme zerlegen und notwendige Schnittstellen vereinbaren – die Teilprobleme so aufteilen, dass Synergieeffekte nutzbar werden – Aspekte der Softwareergonomie umsetzen
Kommunizieren und Kooperieren	<ul style="list-style-type: none"> – Projekte organisieren und bearbeiten – geeignete Quellen zur Vorbereitung und Durchführung der Projektarbeit selbstständig auswählen und rechtskonform nutzen – Abstimmung mit dem Auftraggeber (z. B. Lehrkraft) durchführen und daraus Zielsetzungen für das Endprodukt ableiten – Regeln zur kooperativen Lösung von Teilproblemen, zur Festlegung von Klassen und deren Schnittstellen sowie zum Zeitmanagement diskutieren und vereinbaren – projektbegleitend Zwischenergebnisse dokumentieren, kommunizieren und daraus Schlussfolgerungen für das weitere Vorgehen ableiten
Darstellen und Interpretieren	<ul style="list-style-type: none"> – Abläufe des Software-Engineerings mittels verschiedener Diagramme veranschaulichen und interpretieren – eine begleitende Dokumentation des Projektes in sprachlich differenzierter und grafischer Form erstellen

Grundlegende Wissensbestände	
<ul style="list-style-type: none"> – Methoden des Software-Engineerings – Dokumentation von Projekten – Projektplanung und -organisation – Spezifikation der Anforderungen – technischer Entwurf der Software – Implementierung und Test – Qualitätssicherung 	
Mögliche Projekte	
<ul style="list-style-type: none"> – ein Projekt aus der Erfahrungswelt der Schülerinnen und Schüler umsetzen 	
Bezüge zu den fächerübergreifenden Themen (gemäß Grundsatzband Kap. 4)	
Gesundheit und Wohlergehen	Deutsch, Englisch, Latein, Biologie, Physik, Geschichte, Ethikunterricht, Sport
Arbeit und Wirtschaft	Englisch, Französisch, Spanisch, Russisch, Italienisch, Chemie, Geschichte, Sozialkunde, Ethikunterricht, Katholischer Religionsunterricht, Geographie
Innovation, Infrastruktur und Digitalität	Deutsch, Englisch, Französisch, Italienisch, Biologie, Chemie, Physik, Geschichte, Sozialkunde, Ethikunterricht, Evangelischer Religionsunterricht, Geographie
Produktion und Konsum	Deutsch, Englisch, Spanisch, Russisch, Italienisch, Latein, Biologie, Chemie, Geschichte, Sozialkunde, Evangelischer Religionsunterricht, Katholischer Religionsunterricht, Geographie, Kunst, Sport

Kurs 4: Aktuelle Entwicklungen der Informatik im gesellschaftlichen Kontext	
Modellieren und Implementieren	<ul style="list-style-type: none"> – wesentliche Aspekte von komplexen Informatikinhalten erkennen und modellieren
Begründen und Bewerten	<ul style="list-style-type: none"> – Chancen, Risiken und Missbrauchsmöglichkeiten von Informatiksystemen beschreiben und bewerten – Möglichkeiten und Grenzen in einer von Informatiksystemen dominierten Welt erkennen und bewerten – Potenziale der Digitalisierung im Sinne der Nachhaltigkeit (z. B. Ressourcenverbrauch, Umweltschutz, soziale Auswirkungen) erkennen, bewerten und Handlungsstrategien ableiten
Strukturieren und Vernetzen	<ul style="list-style-type: none"> – gegebene Informatiksysteme aus technischer, informatischer, ökologischer und sozialer Sicht analysieren und diskutieren – Rückschlüsse auf das eigene Verhalten beim Einsatz von Informatiksystemen ziehen
Kommunizieren und Kooperieren	<ul style="list-style-type: none"> – gesellschaftliche Folgen der Einführung und Nutzung von Informatiksystemen sprachlich differenziert diskutieren und Handlungsstrategien auch für lebenslanges Lernen ableiten – Potenziale der Digitalisierung im Sinne sozialer Integration und sozialer Teilhabe erkennen, analysieren und reflektieren
Darstellen und Interpretieren	<ul style="list-style-type: none"> – Wechselwirkungen zwischen Informatiksystemen, Individuen, Gesellschaft und Umwelt (z. B. Ressourcenverbrauch, Umweltschutz, soziale Auswirkungen) darstellen und beschreiben
Grundlegende Wissensbestände	
<p>aus einem oder mehreren Themenfeldern zu aktuellen Entwicklungen in der Informatik, z. B.</p> <ul style="list-style-type: none"> – aktuelle Informatiksysteme und Technikfolgenabschätzung – Gestaltung von Informatiksystemen – virtuelle Welten (z. B. 3D-Modellierung) – Systeme der künstlichen Intelligenz und deren Anwendung – soziale Netzwerke und kollaboratives Arbeiten – Ressourcenverbrauch von Informatiksystemen 	

Mögliche Projekte	
<ul style="list-style-type: none"> – ein Smart-Home aufbauen oder simulieren – ein System der künstlichen Intelligenz implementieren oder trainieren – ein Lernmanagementsystem analysieren – eine Simulationsstudie durchführen 	
Bezüge zu den fächerübergreifenden Themen (gemäß Grundsatzband Kap. 4)	
Keine Armut und kein Hunger	Deutsch, Englisch, Spanisch, Russisch, Biologie, Chemie, Geschichte, Sozialkunde, Ethikunterricht, Evangelischer Religionsunterricht, Katholischer Religionsunterricht, Geographie
Gesundheit und Wohlergehen	Deutsch, Englisch, Latein, Biologie, Physik, Geschichte, Ethikunterricht, Sport
Bildung, Kultur und lebenslanges Lernen	Deutsch, Englisch, Französisch, Spanisch, Russisch, Italienisch, Latein, Griechisch, Biologie, Geschichte, Sozialkunde, Ethikunterricht, Evangelischer Religionsunterricht, Katholischer Religionsunterricht, Geographie, Kunst, Musik, Sport
Klimawandel und Klimaschutz	Deutsch, Englisch, Französisch, Spanisch, Russisch, Latein, Biologie, Chemie, Geschichte, Sozialkunde, Ethikunterricht, Evangelischer Religionsunterricht, Katholischer Religionsunterricht, Geographie, Sport
Energie	Englisch, Spanisch, Russisch, Biologie, Chemie, Physik, Geschichte, Sozialkunde, Geographie
Arbeit und Wirtschaft	Englisch, Französisch, Spanisch, Russisch, Italienisch, Chemie, Geschichte, Sozialkunde, Ethikunterricht, Katholischer Religionsunterricht, Geographie
Innovation, Infrastruktur und Digitalität	Deutsch, Englisch, Französisch, Italienisch, Biologie, Chemie, Physik, Geschichte, Sozialkunde, Ethikunterricht, Evangelischer Religionsunterricht, Geographie
Weniger Ungleichheit	Deutsch, Englisch, Spanisch, Biologie, Chemie, Geschichte, Sozialkunde, Ethikunterricht, Evangelischer Religionsunterricht, Geographie
Städte und Gemeinden	Englisch, Spanisch, Russisch, Italienisch, Latein, Physik, Geschichte, Geographie, Kunst
Produktion und Konsum	Deutsch, Englisch, Spanisch, Russisch, Italienisch, Latein, Biologie, Chemie, Geschichte, Sozialkunde, Evangelischer Religionsunterricht, Katholischer Religionsunterricht, Geographie, Kunst, Sport