

**Schulinterner Lehrplan
des Gymnasiums Hohenlimburg zum Kernlehrplan für die
Sekundarstufen I und II (G9)
Stand: 20.09.2025
- Informatik –**



Inhalt

1 Rahmenbedingungen für den Fachbereich Informatik am Gymnasium Hohenlimburg	2
1.1 Leitbild	2
1.2 Fachliche Bezüge zu den Rahmenbedingungen des schulischen Umfelds	2
1.2.1 Fachliche Bezüge zum Leitbild der Schule.....	3
1.3 Fachliche Bezüge zu schulischen Standards zum Lehren und Lernen	4
1.3.1 Allgemein	4
1.3.2 Exkursionen und Wettbewerbe	5
1.3.3 Medien	5
2 Unterrichtsvorhaben.....	6
2.1 Unterrichtsvorhaben Jahrgangsstufe 6	7
2.2 Unterrichtsvorhaben Jahrgangsstufe 9	25
2.3 Unterrichtsvorhaben Jahrgangsstufe 10.....	26
3 Grundsätze der fachdidaktischen und fachmethodischen Arbeit	27
3.1 Lehr- und Lernprozesse	28
3.2 Individuelles Lernen und Umgang mit Heterogenität	29
3.4 Grundsätze der Leistungsbewertung und Leistungsrückmeldung.....	30
3.5 Lehr- und Lernmittel	31
4. Qualitätssicherung und Evaluation	32
5. Unterrichtsvorhaben der Oberstufe	33
5.1 Unterrichtsvorhaben.....	33
5.1.1 Unterrichtsvorhaben Jahrgangsstufe EF	34
5.1.1.2 Konkretisierung der Unterrichtsvorhaben der EF.....	36
5.1.2 Unterrichtsvorhaben Jahrgangsstufe Q1	44
5.1.2.2 Konkretisierung der Unterrichtsvorhaben der Q1.....	46
5.1.3 Unterrichtsvorhaben Jahrgangsstufe Q2	56
5.1.3.2 Konkretisierung der Unterrichtsvorhaben der Q2.....	57

1 Rahmenbedingungen für den Fachbereich Informatik am Gymnasium Hohenlimburg

1.1 Leitbild

Das Gymnasium Hohenlimburg versteht sich als starke Schule (Mitglied der BuG-Schulen NRW s. Schulprogramm). Daraus folgt, dass das 724 Mädchen und Jungen in den naturwissenschaftlichen Fächern auch stark gemacht werden sollen. Das Gymnasium sieht seinen Schwerpunkt durchaus in den naturwissenschaftlichen Fächern. Es bietet jedes Jahr einen Biologie- und Physikleistungskurs an und es gibt immer mehrere Grundkurse in Informatik, Chemie, Physik und Biologie. Von den insgesamt 64 Kolleginnen und Kollegen haben 5 KollegInnen die Facultas Informatik. Wir nehmen regelmäßig an naturwissenschaftlich ausgerichteten Wettbewerben teil und beziehen die Angebote der nahestehenden Institutionen (bspw. Stadtbücherei Roboter) mit in den Unterricht und in die außerunterrichtlichen Aktivitäten mit ein. Die einmalige Stiftung des 1974 verstorbenen Kaufmanns Arthur-Loose ermöglicht es den besten Schülerinnen und Schüler eines Jahrgangs eine Auszeichnung zu erhalten, die den künftigen Studierenden im naturwissenschaftlichen Bereich eine finanzielle Unterstützung bietet. Im Rahmen der Berufsorientierung erkunden die Schülerinnen und Schüler verschiedene, oft technische Betriebe. Der naturwissenschaftliche Unterricht ist grundlegend für viele Ausbildungsberufe in diesem Bereich.

1.2 Fachliche Bezüge zu den Rahmenbedingungen des schulischen Umfelds

An der Schule unterrichten fünf Lehrpersonen das Fach Informatik. Informatikunterricht findet in der Regel in Doppelstunden im Fachraum statt und wird in den Jahrgängen 6, 9 und 10 zweistündig unterrichtet. In allen Themenfeldern sollen Schülerinnen und Schüler die Möglichkeit haben, objektorientiertes Programmieren und eine informatorische Grundidee zu entwickeln, was mit der vorhandenen Ausstattung fast durchgehend möglich ist. Dabei sind wir bemüht die Ausstattung der Computerräume immer aktuell zu halten.

Die Schule verfügt über eine gute Ausstattung für den Informatikunterricht. Hierzu zählen zwei Informatikräume, in denen in der Regel der Informatikunterricht stattfindet, die aber auch für andere Unterrichtsfächer mit Blick auf die Umsetzung des Medienkompetenzrahmen NRW genutzt werden. Einer der Räume verfügt über genügend Arbeitsplätze für ungefähr 32 Schüler, der andere für ungefähr 24 Schüler.

Neben den Computern gehört zur Ausstattung auch eine Reihe von Calliope Minis, Lego Mindstorms und weiteren Geräten, an denen schon früh praxisorientiert Konzepte informatischer Grundbildung erprobt und erfahren werden können. Sie sind weiter mit einem Beamer ausgestattet und Klimaanlage ausgerüstet.

Funktionsinhaber der Fachgruppe

Fachvorsitz: Schlese

Stellvertreter:in: Syan

1.2.1 Fachliche Bezüge zum Leitbild der Schule

Unter dem Schulmotto „Eine Schule, die stark macht“ sollen die Lernenden als Individuen mit jeweils besonderen Fähigkeiten, Stärken und Interessen in den Blick genommen werden. Es ist ein wichtiges Anliegen, durch gezielte Unterstützung des Lernens die Potenziale jeder Schülerin und jedes Schülers in allen Bereichen optimal zu entwickeln. In einem längerfristigen Entwicklungsprozess arbeitet das Fach Informatik daran, die Bedingungen für individuelles und erfolgreiches Lernen zu verbessern.

1.3 Fachliche Bezüge zu schulischen Standards zum Lehren und Lernen

1.3.1 Allgemein

Der Informatikunterricht am Gymnasium Hohenlimburg setzt das Lern- und Unterrichtsverständnis um, das aus Sicht aktueller Unterrichtsdiagnostik geboten ist. Neben der Berücksichtigung der allgemeinen Qualitätsmerkmale von Unterricht, wie etwa die beiden Unterrichtsforscher Andreas Helmke und Hilbert Meyer sie formulieren, wird der Forderung nach kompetenzorientiertem Lernen Rechnung getragen. Die aktuellen Kernlehrpläne für die Sekundarstufen I an Gymnasien, ebenso wie neurophysiologische Erkenntnisse zum Lernen, unterstreichen den Perspektivwechsel von der Input- zur Outcome-Orientierung: Schüler:innen sollen systematisch Grundlagenwissen (intelligentes Wissen) erwerben und erweitern, und dieses in sinnvollen, begründeten und authentischen Anforderungssituationen kreativ anwenden. (bspw. Programmierung Mindstormroboter) Durch diese Anwendung manifestiert sich "Kompetenz". Um den Ansprüchen eines kompetenzorientierten Unterrichts genüge zu leisten, beachten wir in unserem Fach folgende Prinzipien: kognitive Aktivierung, lebensweltliche Anwendung, individuelle Lernbegleitung, Wissensvernetzung. Dabei wird auch darauf geachtet, eine möglichst hohe Schüler:innenorientierung und breite Schüler:innenaktivierung anzustreben. Sie kann durch

- a) die Initiierung sinnstiftender kognitiver, aber auch ganzheitlicher (kreativ, meditativ, handlungsorientiert, in Sek II allerdings verstärkt mit wissenschafts-propädeutischer Zielorientierung) Lernprozesse (Lernen „mit Kopf, Herz und Hand“, Pestalozzi).
- b) den Einsatz sinnvoller, progressiv angelegter kompetenzorientierter Lernaufgaben mit konkretem Lebensweltbezug, die die Schüler und Schülerinnen nicht nur im fachlich-inhaltlichen Lernbereich, sondern auch in im persönlichen Lernbereich fördern.
- c) den angemessenen Einsatz kooperativer Lernformen bzw. dialogischer Unterrichtsstrukturen, die es ermöglichen, den sozial-kommunikativen Lernbereich auszubauen.
- d) den Aufbau eines grundlegenden Repertoires fachlicher Methoden, die den Schüler:innen ermöglichen, in zunehmend selbstständigerer, inhaltlich-komplexerer und wissenschaftspropädeutischer Weise mit spezifischen Inhalten und Fragestellungen unseres Faches umzugehen und den methodisch-strategischen Lernbereich über bereichsspezifische Kompetenzen hinausgehend auszubauen, erreicht werden.

Der Informatikunterricht knüpft an die Alltagserfahrungen der Schüler:innen an. Dazu werden Schülervorstellungen im Unterricht erfasst und weiterentwickelt. Durch kooperative Lernformen wird eine hohe Schüleraktivität erreicht und werden kommunikative sowie soziale Kompetenzen weiterentwickelt. Trotz festgelegter Sitzreihen, bedingt durch Computerarbeitsplätze, ist die Sitzordnung so flexibel, dass jeder Zeit Einzel-, Partner und Gruppenarbeiten möglich sind.

In einem längerfristigen Entwicklungsprozess arbeitet unser Fach daran, die Bedingungen für erfolgreiches und individuelles Lernen zu verbessern. Dabei greift das Fach Informatik in allen Inhaltsbereichen aktuelle und für Schüler:innen Themen z.B. des Verbraucherschutzes und der ökologischen Bildung auf. Durch das Lernen mit verschiedenen, gerade auch digitalen Medien in unterschiedlichen Sozialformen und unter Berücksichtigung individueller Lernwege werden

altersgerecht Aufgeschlossenheit und Neugier geweckt und Schüler:innen zu eigenständigem Handeln angeleitet. Eine verstärkte Zusammenarbeit und Koordinierung der Fachbereiche ermöglicht komplexe Lerngegenstände umfassend darzustellen und Bezüge zwischen Inhalten der Fächer herzustellen, sodass ein wesentlicher Beitrag zur vertieften Allgemeinbildung geleistet werden kann. An Problemstellungen werden vorhandene Kenntnisse selbstständiger Lern- und Denkstrategien aufgegriffen und weiterentwickelt.

1.3.2 Exkursionen und Wettbewerbe

Die Fachschaft Informatik ist daran interessiert, die Schüler:innen auf die universitäre Laufbahn vorzubereiten. Daher nehmen die Schüler:innen regelmäßig an Exkursionen und Wettbewerben teil, die z.T. von Universitäten angeboten werden. Hier eine Auflistung der Exkursionen und Wettbewerbe, die regelmäßig besucht werden:

-To DO-

1.3.3 Medien

z.B. Medienkonzept, Methodencurriculum (Lernen lernen), Selbstlernzentrum, Förderkonzept, Wettbewerbskultur

2 Unterrichtsvorhaben

In der nachfolgenden *Übersicht über die Unterrichtsvorhaben* wird der Fachkonferenzbeschluss zur verbindlichen Verteilung der Unterrichtsvorhaben dargestellt. Die Übersicht dient dazu, für die einzelnen Jahrgangsstufen allen am Bildungsprozess Beteiligten einen schnellen Überblick über Themen bzw. Fragestellungen der Unterrichtsvorhaben unter Angabe besonderer Schwerpunkte in den Inhalten und in der Kompetenzentwicklung zu verschaffen. Der einzelnen Lehrkraft bleibt die Reihenfolge der Themenbereiche in enger Absprache mit den parallel unterrichtenden Lehrkräften freigestellt.

2.1 Unterrichtsvorhaben Jahrgangsstufe 6

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
UV 6.1: Wir präsentieren uns als Avatar <i>Was ist ein Informatiksystem und wie kann ich es für ein projektartiges Vorhaben nutzen?</i> ca. 6 Ustd.	IF: Information und Daten - Informationsgehalt von Daten IF: Informatiksysteme - Aufbau und Funktionsweise von Informatiksystemen - Anwendung von Informatiksystemen IF: Informatik, Mensch und Gesellschaft - Informatiksysteme in der Lebens- und Arbeitswelt - Datenbewusstsein	Argumentieren (A) - formulieren Fragen zu einfachen informatischen Sachverhalten - äußern Vermutungen zu informatischen Sachverhalten auf der Basis von Alltagsvorstellungen oder Vorwissen - begründen die Auswahl eines Informatiksystems Modellieren und Implementieren (MI) - erstellen informatische Modelle zu gegebenen Sachverhalten Darstellen und Interpretieren (DI) - beschreiben einfache Darstellungen von informatischen Sachverhalten	- stellen eine ausgewählte Information in geeigneter Form als Daten formalsprachlich oder graphisch dar (DI) - interpretieren ausgewählte Daten als Information im gegebenen Kontext (DI) - benennen Beispiele für (vernetzte) Informatiksysteme aus ihrer Erfahrungswelt (DI) - benennen Grundkomponenten von (vernetzten) Informatiksystemen und beschreiben ihre Funktionen (DI) - beschreiben das Prinzip der Eingabe, Verarbeitung und Ausgabe (EVA-Prinzip) als grundlegendes Prinzip der Datenverarbeitung (DI)

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
		Kommunizieren und Kooperieren(KK) • erläutern informatische Sachverhalte unter Verwendung von Fachbegriffen sachgerecht • setzen bei der Bearbeitung einer informatischen Problemstellung geeignete digitale Werkzeuge zum kollaborativen Arbeiten ein (MKR 1.2, 3.1)	(MKR 6.1) • vergleichen Möglichkeiten der Datenverwaltung hinsichtlich ihrer spezifischen Charakteristika (u. a. Speicherort, Kapazität, Aspekte der Datensicherheit) (A) • setzen zielgerichtet Informatiksysteme zur Verarbeitung von Daten ein (MI) • erläutern Prinzipien der strukturierten Dateiverwaltung (A) (MKR 1.3) • setzen Informatiksysteme zur Kommunikation und Kooperation ein (KK) (MKR 3.1) • beschreiben an Beispielen die Bedeutung von Informatiksystemen in der Lebens- und Arbeitswelt (KK) (MKR 6.4)

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
			• erläutern an ausgewählten Beispielen Auswirkungen des Einsatzes von Informatiksystemen (A/KK) • erläutern anhand von Beispielen aus ihrer Lebenswelt Nutzen und Risiken beim Umgang mit eigenen und fremden Daten auch im Hinblick auf Speicherorte (A)
Weitere Hinweise, Vereinbarungen und Absprachen: ... zur Vernetzung: • Informatiksysteme werden im weiteren Unterrichtsverlauf in Informatik immer wieder eine Rolle spielen, z. B. in den Unterrichtsvorhaben „<i>Von der Anweisung zum Algorithmus</i>“ (UV 6.5) und „<i>Automaten in unserer Lebenswelt</i>“ (UV 6.4) sowie „<i>Eigene Programme mit Scratch</i>“ (UV 6.6) ... zu Synergien: • Eigenverantwortliches Lernen – der erste Zugang zu der schuleigenen Lernplattform und der Umgang mit dieser kommt der Arbeit in allen weiteren Unterrichtsfächern zugute. Ebenfalls wird durch den Zugang die Möglichkeit zur Einsicht in den im Internet einsehbaren Vertretungsplan gegeben.			

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
UV 6.2: Codierungen zum Austausch und zur Verarbeitung von Nachrichten ca. 8 Ustd.	IF: Information und Daten - Daten und ihre Codierung - Informationsgehalt von Daten	Argumentieren (A) - formulieren Fragen zu einfachen informatischen Sachverhalten - äußern Vermutungen zu informatischen Sachverhalten auf der Basis von Alltagsvorstellungen oder Vorwissen Modellieren und Implementieren (MI) - erstellen informatische Modelle zu gegebenen Sachverhalten Darstellen und Interpretieren (DI) - beschreiben einfache Darstellungen von informatischen Sachverhalten - stellen informatische Sachverhalte in geeigneter Form dar - interpretieren informatische Darstellungen	- erläutern den Datenbegriff anhand von Beispielen aus ihrer Erfahrungswelt (A) - erläutern den Zusammenhang und die Bedeutung von Information und Daten (A) - stellen eine ausgewählte Information in geeigneter Form als Daten formalsprachlich oder graphisch dar (DI) - nennen Beispiele für die Codierung von Daten aus ihrer Erfahrungswelt (DI) - codieren und decodieren Daten unter Verwendung des Binärsystems (MI) - interpretieren ausgewählte Daten als Information im gegebenen Kontext (DI) - erläutern Einheiten von

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
		Kommunizieren und Kooperieren (KK) • erläutern informatische Sachverhalte unter Verwendung von Fachbegriffen sachgerecht • kooperieren in verschiedenen Formen der Zusammenarbeit bei der Bearbeitung einfacher informatischer Probleme	Datenmengen (A / KK) • vergleichen Datenmengen hinsichtlich ihrer Größe mit Hilfe anschaulicher Beispiele aus ihrer Lebenswelt
Weitere Hinweise, Vereinbarungen und Absprachen: ... zur Vernetzung: • Informationen aus Daten zu entnehmen wird im weiteren Unterrichtsverlauf in Informatik immer wieder eine Rolle spielen, z. B. in den Unterrichtsvorhaben „Detektivarbeit – Auseinandersetzung mit verschiedenen Verfahren zur Codierung und Verschlüsselung sowie deren Anwendung“ (UV 6.3), „Künstliche Intelligenz und maschinelles Lernen - Wie können Menschen und Informatiksysteme auf Grundlage von Daten Vorhersagen treffen?“ (UV 6.7) oder „Datenbewusstsein - Welche Informationen kann man aus meinen Daten oder großen Datenmengen über mich ableiten? Was bedeutet dies für mein Datenbewusstsein“ (UV 6.8) ... zu Synergien: • Mathematik – Stellenwertsysteme (Übernahme der Fachbegriffe aus dem Mathematikbuch); Physik: Rechnen mit Einheiten • Musik – Notationssystem sowie deren Informationsgehalt			

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
UV 6.3: Detektivarbeit – Auseinandersetzung mit verschiedenen Verfahren zur Codierung und Verschlüsselung sowie deren Anwendung ca. 10 - 12 Ustd.	IF: Information und Daten - Daten und ihre Codierung - Verschlüsselungsverfahren IF: Algorithmen - Algorithmen und algorithmische Grundkonzepte IF: Informatik, Mensch und Gesellschaft - Datensicherheit und Sicherheitsregeln	Argumentieren (A) - formulieren Fragen zu einfachen informatischen Sachverhalten - äußern Vermutungen zu informatischen Sachverhalten auf der Basis von Alltagsvorstellungen oder Vorwissen Modellieren und Implementieren (MI) - erstellen informatische Modelle zu gegebenen Sachverhalten Darstellen und Interpretieren (DI) - beschreiben einfache Darstellungen von informatischen Sachverhalten - stellen informatische Sachverhalte in geeigneter Form dar - interpretieren informatische Darstellungen	- nennen Beispiele für die Codierung von Daten aus ihrer Erfahrungswelt (DI) - codieren und decodieren Daten unter Verwendung des Binärsystems (MI) - erläutern Einheiten von Datenmengen (A/KK) - erläutern ein einfaches Transpositionsverfahren als Möglichkeit der Verschlüsselung (DI) (MKR 1.4) - vergleichen verschiedene Verschlüsselungsverfahren unter Berücksichtigung von ausgewählten Sicherheitsaspekten (DI) (MKR 1.4) - führen Handlungsvorschriften schrittweise aus (MI) - beschreiben Maßnahmen zum Schutz von Daten mithilfe von Informatiksystemen (A)

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
		Kommunizieren und Kooperieren (KK) • erläutern informatische Sachverhalte unter Verwendung von Fachbegriffen sachgerecht • strukturieren gemeinsam eine Lösung für ein informatisches Problem • dokumentieren gemeinsam ihren Arbeitsprozess und ihre Ergebnisse auch mithilfe digitaler Werkzeuge (MKR 1.2)	
Weitere Hinweise, Vereinbarungen und Absprachen: ... zur Vernetzung: • Informationen aus Daten zu erhalten und diese zu entschlüsseln, spielt im Unterrichtsverlauf in Informatik immer wieder eine Rolle, z. B. in dem Unterrichtsvorhaben „Codierungen zum Austausch und zur Verarbeitung von Nachrichten“ (UV 6.2)			

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
UV 6.4: Automaten in unserer Lebenswelt ca. 6 Ustd.	IF: Automaten und künstliche Intelligenz - Aufbau und Wirkungsweise einfacher Automaten IF: Informatiksysteme - Aufbau und Funktionsweise von Informatiksystemen	Argumentieren (A) - formulieren Fragen zu einfachen informatischen Sachverhalten - äußern Vermutungen zu informatischen Sachverhalten auf der Basis von Alltagsvorstellungen oder Vorwissen Modellieren und Implementieren (MI) - erstellen informatische Modelle zu gegebenen Sachverhalten Darstellen und Interpretieren (DI) - beschreiben einfache Darstellungen von informatischen Sachverhalten	- erläutern die Funktionsweise eines Automaten aus ihrer Lebenswelt (A) (MKR 6.1) - stellen Abläufe in Automaten graphisch dar (DI) - benennen Grundkomponenten von (vernetzten) Informatiksystemen und beschreiben ihre Funktionen (DI) - beschreiben das Prinzip der Eingabe, Verarbeitung und Ausgabe (EVA-Prinzip) als grundlegendes Prinzip der Datenverarbeitung (DI)

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
		Kommunizieren und Kooperieren (KK) erläutern informatische Sachverhalte unter Verwendung von Fachbegriffen sachgerecht	
Weitere Hinweise, Vereinbarungen und Absprachen: ... zur Vernetzung: Aufbau und Wirkungsweise von Automaten werden im weiteren Unterrichtsverlauf in Informatik immer wieder eine Rolle spielen, z. B. im Unterrichtsvorhaben „<i>Künstliche Intelligenz und maschinelles Lernen - Wie können Menschen und Informatiksysteme auf Grundlage von Daten Vorhersagen treffen?</i>“ (UV 6.7)			

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
UV 6.5: Von der Anweisung zum Algorithmus ca. 12 Ustd.	IF: Information und Daten - Daten und ihre Codierung - Informationsgehalt von Daten IF: Algorithmen - Algorithmen und algorithmische Grundkonzepte - Implementation von Algorithmen	Argumentieren (A) - formulieren Fragen zu einfachen informatischen Sachverhalten - äußern Vermutungen zu informatischen Sachverhalten auf der Basis von Alltagsvorstellungen oder Vorwissen Modellieren und Implementieren (MI) - erstellen informatische Modelle zu gegebenen Sachverhalten Darstellen und Interpretieren (DI) - beschreiben einfache Darstellungen von informatischen Sachverhalten	- stellen eine ausgewählte Information in geeigneter Form als Daten formalsprachlich oder graphisch dar (DI) - formulieren zu Abläufen aus dem Alltag eindeutige Handlungsvorschriften (DI) - überführen Handlungsvorschriften in einen Programmablaufplan (PAP) oder ein Struktogramm (MI) - führen Handlungsvorschriften schrittweise aus (MI) - identifizieren in Handlungsvorschriften Anweisungen und die algorithmischen Grundstrukturen Sequenz, Verzweigung und Schleife (MI) (MKR 6.2) - implementieren Algorithmen in einer

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
		Kommunizieren und Kooperieren (KK) • erläutern informatische Sachverhalte unter Verwendung von Fachbegriffen sachgerecht • setzen bei der Bearbeitung einer informatischen Problemstellung geeignete digitale Werkzeuge zum kollaborativen Arbeiten ein	visuellen Programmiersprache (MI) (MKR 6.1, 6.3) • implementieren Algorithmen unter Berücksichtigung des Prinzips der Modularisierung (MI), (MKR 6.1, 6.3) • überprüfen die Wirkungsweise eines Algorithmus durch zielgerichtetes Testen (MI) (MKR 6.2)
Weitere Hinweise, Vereinbarungen und Absprachen: ... zur Vernetzung: • Algorithmen werden im weiteren Unterrichtsverlauf in Informatik immer wieder eine Rolle spielen, z. B. in den Unterrichtsvorhaben „Codierungen zum Austausch und zur Verarbeitung von Nachrichten (UV 6.2) oder „Automaten in unserer Lebenswelt“ (UV 6.4)			

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
UV 6.6: Eigene Programme mit Scratch ca. 10 - 12 Ustd.	IF: Information und Daten - Informationsgehalt von Daten IF: Algorithmen - Algorithmen und algorithmische Grundkonzepte - Implementation von Algorithmen IF: Informatiksysteme: - Aufbau und Funktionsweise von Informatiksystemen	Argumentieren (A) - bewerten ein Ergebnis einer informatischen Modellierung (MKR 6.4) Modellieren und Implementieren (MI) - erstellen informatische Modelle zu gegebenen Sachverhalten - implementieren informatische Modelle unter Verwendung algorithmischer Grundstrukturen (MKR 6.1, 6.2) - überprüfen Modelle und Implementierungen Darstellen und Interpretieren (DI) - beschreiben einfache Darstellungen von informatischen Sachverhalten - stellen informatische Sachverhalte in geeigneter Form dar	- erläutern den Zusammenhang und die Bedeutung von Information und Daten (A), - interpretieren ausgewählte Daten als Information im gegebenen Kontext (DI) <i>überführen Handlungsvorschriften in einen Programmablaufplan (PAP) oder ein Struktogramm (MI)</i> - identifizieren in Handlungsvorschriften Anweisungen und die algorithmischen Grundstrukturen Sequenz, Verzweigung und Schleife (MI) - implementieren Algorithmen in einer visuellen Programmiersprache (MI) - implementieren Algorithmen unter Berücksichtigung des Prinzips der Modularisierung (MI) - überprüfen die Wirkungsweise eines Algorithmus durch

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
		Kommunizieren und Kooperieren (KK) • erläutern informatische Sachverhalte unter Verwendung von Fachbegriffen sachgerecht • kooperieren in verschiedenen Formen der Zusammenarbeit bei der Bearbeitung einfacher informatischer Probleme • strukturieren gemeinsam eine Lösung für ein informatisches Problem	zielgerichtetes Testen (MI) • ermitteln durch die Analyse eines Algorithmus dessen Ergebnis (DI) (MKR 6.2) • bewerten einen als Quelltext, Programmablaufplan (PAP) oder Struktogramm dargestellten Algorithmus hinsichtlich seiner Funktionalität (A) (MKR 6.3) • benennen Grundkomponenten von (vernetzten) Informatiksystemen und beschreiben ihre Funktionen (DI) • beschreiben das Prinzip der Eingabe, Verarbeitung und Ausgabe (EVA-Prinzip) als grundlegendes Prinzip der Datenverarbeitung (DI)
Weitere Hinweise, Vereinbarungen und Absprachen: Das Unterrichtsvorhaben kann unabhängig von der technischen Ausstattung der Schule durchgeführt werden. Die Anschaffung eines Mikrocontrollers ist nicht zwingend erforderlich. Alle Kompetenzen sind auch durch Nutzung der Simulationsumgebung zu erreichen. Auf den Arbeitsblättern des Leitprogramms sind kurze Videos auf YouTube verlinkt, welche den Lerninhalt jeweils mit einem kurzen Beispiel erklären und den Schülerinnen und Schülern optional als Hilfe dienen sollen.			

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
... zur Vernetzung: - Bezüge zu den Grundkomponenten eines Informatiksystems aus Unterrichtsvorhaben „<i>Wir präsentieren uns als Avatar</i>“ (UV 6.1) sowie zu den <i>Automaten in der Lebenswelt</i> (UV 6.4); Weiterführung der Kompetenzen aus Unterrichtsvorhaben „<i>Von der Anweisung zum Algorithmus</i>“ (UV 6.5) ... zu Synergien: - Weiterführende Projekte mit Calliope Minis oder Lego Mindstorm			

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
UV 6.7: Künstliche Intelligenz und maschinelles Lernen <i>Wie können Menschen und Informatiksysteme auf Grundlage von Daten Vorhersagen treffen?</i> ca. 8 - 10 Ustd.	IF: Automaten und künstliche Intelligenz - Maschinelles Lernen mit Entscheidungsbäumen - Maschinelles Lernen mit neuronalen Netzen	Argumentieren (A) - formulieren Fragen zu einfachen informatischen Sachverhalten - äußern Vermutungen zu informatischen Sachverhalten auf der Basis von Alltagsvorstellungen oder Vorwissen - erläutern mögliche Auswirkungen des Einsatzes von Informatiksystemen Modellieren und Implementieren (MI) - erstellen informatische Modelle zu gegebenen Sachverhalten	- benennen Anwendungsbeispiele künstlicher Intelligenz aus ihrer Lebenswelt (A) - stellen das Grundprinzip eines Entscheidungsbaumes enaktiv als ein Prinzip des maschinellen Lernens dar (DI) - beschreiben die grundlegende Funktionsweise künstlicher neuronaler Netze in verschiedenen Anwendungsbeispielen (KK)

		Darstellen und Interpretieren (DI) • beschreiben einfache Darstellungen von informatischen Sachverhalten • stellen informatische Sachverhalte in geeigneter Form dar • interpretieren informatische Darstellungen Kommunizieren und Kooperieren (KK) • erläutern informatische Sachverhalte unter Verwendung von Fachbegriffen sachgerecht	
Weitere Hinweise, Vereinbarungen und Absprachen: ... zur Vernetzung: • Vorhersagungen auf Grundlage von Daten spielen im Unterrichtsverlauf in Informatik immer wieder eine Rolle, z. B. in dem Unterrichtsvorhaben „Detektivarbeit – Auseinandersetzung mit verschiedenen Verfahren zur Codierung und Verschlüsselung sowie deren Anwendung“ (UV 6.3)			

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
UV 6.8: Datenbewusstsein <i>Welche Informationen kann man aus meinen Daten oder großen Datenmengen über mich ableiten? Was bedeutet dies für mein Datenbewusstsein?</i> ca. 4-6 Ustd.	IF: Informatik, Mensch und Gesellschaft • Informatiksysteme in der Lebens- und Arbeitswelt • Datenbewusstsein • Datensicherheit und Sicherheitsregeln	Argumentieren (A) • formulieren Fragen zu einfachen informatischen Sachverhalten • äußern Vermutungen zu informatischen Sachverhalten auf der Basis von Alltagsvorstellungen oder Vorwissen • erläutern mögliche Auswirkungen des Einsatzes von Informatiksystemen Darstellen und Interpretieren (DI) • beschreiben einfache Darstellungen von informatischen Sachverhalten • stellen informatische Sachverhalte in geeigneter Form dar	• erläutern an ausgewählten Beispielen Auswirkungen des Einsatzes von Informatiksystemen • beschreiben anhand von ausgewählten Beispielen die Verarbeitung und Nutzung personenbezogener Daten (DI) • erläutern anhand von Beispielen aus ihrer Lebenswelt Nutzen und Risiken beim Umgang mit eigenen und fremden Daten auch im Hinblick auf Speicherorte (A) • beschreiben Maßnahmen zum Schutz von Daten mithilfe von Informatiksystemen (A),

JAHRGANGSSTUFE 6			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Übergeordnete Kompetenzerwartungen <i>Schülerinnen und Schüler</i>	Konkretisierte Kompetenzerwartungen <i>Schülerinnen und Schüler</i>
		- interpretieren informatische Darstellungen Kommunizieren und Kooperieren (KK) - erläutern informatische Sachverhalte unter Verwendung von Fachbegriffen sachgerecht - kooperieren in verschiedenen Formen der Zusammenarbeit bei der Bearbeitung einfacher informatischer Probleme	(MKR 1.4)
Weitere Hinweise, Vereinbarungen und Absprachen: ... zur Vernetzung: - Informationen aus Datenmengen abzuleiten und ein Bewusstsein für Daten zu entwickeln, spielt im Unterrichtsverlauf in Informatik immer wieder eine Rolle, z. B. in den Unterrichtsvorhaben „Codierungen zum Austausch und zur Verarbeitung von Nachrichten“ (UV 6.2) und „Detektivarbeit – Auseinandersetzung mit verschiedenen Verfahren zur Codierung und Verschlüsselung sowie deren Anwendung“ (UV 6.3) ... zu Synergien: - Anknüpfungspunkte mit Blick auf Wirtschaft/Politik			

2.2 Unterrichtsvorhaben Jahrgangsstufe 9

JAHRGANGSSTUFE 9			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Schwerpunkte der Kompetenzentwicklung	Weitere Vereinbarungen
	•	•	
	•	•	
	•	•	
	•	•	

2.3 Unterrichtsvorhaben Jahrgangsstufe 10

JAHRGANGSSTUFE 10			
Unterrichtsvorhaben	Inhaltsfelder Inhaltliche Schwerpunkte	Schwerpunkte der Kompetenzentwicklung	Weitere Vereinbarungen
	•	•	
	•	•	
	•	•	
	•	•	

3 Grundsätze der fachdidaktischen und fachmethodischen Arbeit

In Absprache mit der Lehrerkonferenz sowie unter Berücksichtigung des Schulprogramms hat die Fachkonferenz Informatik bezüglich ihres schulinternen Lehrplans die folgenden fachdidaktischen und fachmethodischen Grundsätze beschlossen. In diesem Zusammenhang beziehen sich die Grundsätze 1 bis 14 auf fächerübergreifende Aspekte, die auch Gegenstand der Qualitätsanalyse sind, die Grundsätze 15 bis 21 sind fachspezifisch angelegt.

Überfachliche Grundsätze:

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

Fachliche Grundsätze:

15. Der Unterricht unterliegt der Wissenschaftsorientierung und ist dementsprechend eng verzahnt mit seiner Bezugswissenschaft.
16. Der Unterricht ist problemorientiert und soll von realen Problemen ausgehen und sich auf solche rückbeziehen.
17. Der Unterricht folgt dem Prinzip der Exemplarizität und soll ermöglichen, informatische Strukturen und Gesetzmäßigkeiten in den ausgewählten Problemen und Projekten zu erkennen
18. Der Unterricht ist anschaulich sowie gegenwarts- und zukunftsorientiert und gewinnt dadurch für die Schülerinnen und Schüler an Bedeutsamkeit.
19. Der Unterricht ist handlungsorientiert, d.h. projekt- und produktorientiert angelegt.
20. Im Unterricht werden sowohl für die Schule didaktisch reduzierte als auch reale Informatiksysteme aus der Wissenschafts-, Berufs- und Lebenswelt eingesetzt.
21. Der Unterricht beinhaltet reale Begegnung mit Informatiksystemen.

3.1 Lehr- und Lernprozesse

- Schwerpunktsetzungen nach folgenden Kriterien:
 - Herausstellung zentraler Ideen und Konzepte, auch unter Nutzung von Synergien zwischen den naturwissenschaftlichen Fächern
 - Zurückstellen von Verzichtbarem bzw. eventuell späteres Aufgreifen, Orientierung am Prinzip des exemplarischen Lernens
 - Anschlussfähigkeit (fachintern und fachübergreifend)
 - Herstellen von Zusammenhängen statt Anhäufung von Einzelfakten
- Lehren und Lernen in sinnstiftenden Kontexten nach folgenden Kriterien:
 - Eignung des Kontextes zum Erwerb spezifischer Kompetenzen („Was kann man an diesem Thema besonders gut lernen“?)
 - klare Schwerpunktsetzungen bezüglich des Erwerbs spezifischer Kompetenzen, insbesondere auch bezüglich physikalischer Denk- und Arbeitsweisen
 - eingegrenzte und altersgemäße Komplexität
 - authentische, motivierende und tragfähige Problemstellungen
 - Nachvollziehbarkeit/Schülerverständnis der Fragestellung
 - Kontexte und Lernwege sollten nicht unbedingt an fachsystematischen Strukturen, sondern eher an Erkenntnis- und Verständnisprozessen der Lernenden ansetzen.
- Variation der Lernaufgaben und Lernformen mit dem Ziel einer kognitiven Aktivierung aller Lernenden nach folgenden Kriterien:
 - Aufgaben auch zur Förderung von vernetztem Denken mit Hilfe von übergreifenden Prinzipien, grundlegenden Ideen und Basiskonzepten
 - Einsatz von digitalen Medien und Werkzeugen zur Verständnisförderung und zur Unterstützung und Beschleunigung des Lernprozesses.
 - Einbindung von Phasen der Metakognition, in denen zentrale Aspekte von zu erwerbenden Kompetenzen reflektiert werden, explizite Thematisierung der erforderlichen Denk- und Arbeitsweisen und ihrer zugrundeliegenden Ziele und Prinzipien, Vertraut machen mit dabei zu verwendenden Begrifflichkeiten
 - Vertiefung der Fähigkeit zur Nutzung erworbener Kompetenzen beim Transfer auf neue Aufgaben und Problemstellungen durch hinreichende Integration von Reflexions-, Übungs- und Problemlösephasen in anderen Kontexten
 - ziel- und themengerechter Wechsel zwischen Phasen der Einzelarbeit, Partnerarbeit und Gruppenarbeit unter Berücksichtigung von Vielfalt durch Elemente der Binnendifferenzierung
 - Beachtung von Aspekten der Sprachsensibilität bei der Erstellung von Materialien.
 - bei kooperativen Lernformen: insbesondere Fokussierung auf das Nachdenken und den Austausch von naturwissenschaftlichen Ideen und Argumenten

3.2 Individuelles Lernen und Umgang mit Heterogenität

Die Fachschaft Informatik hat Absprachen über die Förderung der individuellen Kompetenzentwicklung getroffen. Die Gestaltung von Lernprozessen beschränkt sich nicht auf eine angenommene mittlere Leistungsfähigkeit einer Lerngruppe, sondern soll Lerngelegenheiten sowohl für stärkere als auch schwächere Schülerinnen und Schüler bieten. Im Folgenden werden die bereits abgesprochenen und erstellten Lernarrangements für binnendifferenzierende Maßnahmen beschrieben.

- unterrichtsbegleitende Testaufgaben zur Diagnose individueller Kompetenzentwicklung in allen Kompetenzbereichen
Jahrgangsstufe 6:
Jahrgangsstufe 9:
Jahrgangsstufe 10:
- komplexere Lernaufgaben mit gestuften Lernhilfen für unterschiedliche Leistungsanforderungen
Jahrgangsstufe 6:
Jahrgangsstufe 9:
Jahrgangsstufe 10:
- unterstützende zusätzliche Maßnahmen für erkannte oder bekannte Lernschwierigkeiten
Jahrgangsstufe 6:
Jahrgangsstufe 9:
Jahrgangsstufe 10:
- herausfordernde zusätzliche Angebote für besonders leistungsstarke Schülerinnen und Schüler (auch durch Helfersysteme oder Unterrichtsformen wie „Lernen durch Lehren“)
Jahrgangsstufe 6:
Jahrgangsstufe 9:
Jahrgangsstufe 10:

3.4 Grundsätze der Leistungsbewertung und Leistungsrückmeldung

Die Fachkonferenz hat Vereinbarungen zu Bewertungskriterien und deren Gewichtung getroffen. Ziele dabei sind, innerhalb der gegebenen Freiräume sowohl eine Transparenz von Bewertungen als auch eine Vergleichbarkeit von Leistungen zu gewährleisten.

Grundsätzliche Absprachen:

Erbrachte Leistungen werden auf der Grundlage transparenter Ziele und Kriterien in allen Kompetenzbereichen benotet, sie werden den Schüler:innen jedoch auch mit Bezug auf diese Kriterien rückgemeldet und erläutert. Auf dieser Basis sollen die Schüler:innen ihre Leistungen zunehmend selbstständig einschätzen können. Die individuelle Rückmeldung erfolgt stärkenorientiert und nicht defizitorientiert, sie soll dabei den tatsächlich erreichten Leistungsstand weder beschönigen noch abwerten. Sie soll Hilfen und Absprachen zu realistischen Möglichkeiten der weiteren Entwicklung enthalten.

Die Bewertung von Leistungen berücksichtigt Lern- und Leistungssituationen. Einerseits soll dabei Schüler:innen deutlich gemacht werden, in welchen Bereichen aufgrund des zurückliegenden Unterrichts stabile Kenntnisse erwartet und bewertet werden. Andererseits dürfen sie in neuen Lernsituationen auch Fehler machen, ohne dass sie deshalb Geringschätzung oder Nachteile in ihrer Beurteilung befürchten müssen.

Überprüfung und Beurteilung der Leistungen

Die Leistungen im Unterricht werden in der Regel auf der Grundlage einer kriteriengeleiteten, systematischen Beobachtung von Unterrichtshandlungen beurteilt.

Weitere Anhaltspunkte für Beurteilungen lassen sich mit kurzen schriftlichen, auf stark eingegrenzte Zusammenhänge begrenzten Tests gewinnen.

Kriterien der Leistungsbeurteilung:

Die Bewertungskriterien für Leistungsbeurteilungen müssen den Schüler:innen bekannt sein. Die folgenden Kriterien gelten allgemein und sollten in ihrer gesamten Breite für Leistungsbeurteilungen berücksichtigt werden:

- für Leistungen, die zeigen, in welchem Ausmaß Kompetenzerwartungen des Lehrplans bereits erfüllt werden. Beurteilungskriterien können hier u.a. sein:
 - die inhaltliche Geschlossenheit und sachliche Richtigkeit sowie die Angemessenheit fachtypischer qualitativer und quantitativer Darstellungen bei Erklärungen, beim Argumentieren und beim Lösen von Aufgaben,
 - die zielgerechte Auswahl und konsequente Anwendung von Verfahren beim Planen und programmieren und bei der Nutzung von Modellen,
 - die Genauigkeit und Zielbezogenheit beim Analysieren, Interpretieren und Erstellen von Texten, Graphiken oder Diagrammen.

- für Leistungen, die im Prozess des Kompetenzerwerbs erbracht werden. Beurteilungskriterien können hier u.a. sein:
 - die Qualität, Kontinuität, Komplexität und Originalität von Beiträgen zum Unterricht (z. B. beim Generieren von Fragestellungen und Begründen von Ideen und Lösungsvorschlägen, Darstellen, Argumentieren, Strukturieren und Bewerten von Zusammenhängen),
 - die Vollständigkeit und die inhaltliche und formale Qualität von Lernprodukten (z. B. Materialsammlungen, Hefte, Mappen, Portfolios, Lerntagebücher, Dokumentationen, Präsentationen, Lernplakate, Programmen),
 - Lernfortschritte im Rahmen eigenverantwortlichen, schüleraktiven Handelns (z. B. Vorbereitung und Nachbereitung von Unterricht, Lernaufgabe, Referat, Rollenspiel, Befragung, Erkundung, Präsentation),
 - die Qualität von Beiträgen zum Erfolg gemeinsamer Gruppenarbeiten.

Verfahren der Leistungsrückmeldung und Beratung

Die Leistungsrückmeldung kann in mündlicher und schriftlicher Form erfolgen.

- Intervalle
Eine differenzierte Rückmeldung zum erreichten Lernstand sollte mindestens einmal pro Quartal erfolgen. Aspektbezogene Leistungsrückmeldung erfolgt anlässlich der Auswertung benoteter Lernprodukte.
- Formen
Schülergespräch, individuelle Beratung, schriftliche Hinweise und Kommentare (Selbst-Evaluationsbögen; Gespräche beim Elternsprechtag]

3.5 Lehr- und Lernmittel

Lehrwerke, die an Schülerinnen und Schüler für den ständigen Gebrauch ausgeliehen werden:

- Klasse 6: -
- Klasse 9: -
- Klasse 10: -

4. Qualitätssicherung und Evaluation

Das schulinterne Curriculum stellt keine starre Größe dar, sondern ist als „dynamisches Dokument“ zu betrachten. Die Fachkonferenz trägt durch eine ständige Überprüfung und Weiterentwicklung der Lehrinhalte zur Qualitätsentwicklung und damit zur Qualitätssicherung des Faches bei.

Maßnahmen der fachlichen Qualitätssicherung:

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

Kolleg: innen der Fachschaft (ggf. auch die gesamte Fachschaft) nehmen regelmäßig an Fortbildungen teil, um fachliches Wissen zu aktualisieren und pädagogische sowie didaktische Handlungsalternativen zu entwickeln. Zudem werden die Erkenntnisse und Materialien aus fachdidaktischen Fortbildungen und Implementationen zeitnah in der Fachgruppe vorgestellt und für alle verfügbar gemacht.

Feedback von Schülerinnen und Schülern wird als wichtige Informationsquelle zur Qualitätsentwicklung des Unterrichts angesehen. Sie sollen deshalb Gelegenheit bekommen, die Qualität des Unterrichts zu evaluieren.

Überarbeitungs- und Planungsprozess:

Eine Evaluation erfolgt jährlich. In den Dienstbesprechungen der Fachgruppe zu Schuljahresbeginn werden die Erfahrungen des vorangehenden Schuljahres ausgewertet und diskutiert sowie eventuell notwendige Konsequenzen formuliert. Nach der jährlichen Evaluation werden die Änderungsvorschläge für den schulinternen Lehrplan eingearbeitet. Insbesondere verständigen wir uns über alternative Materialien, Kontexte und die Zeitkontingente der einzelnen Unterrichtsvorhaben.

5. Unterrichtsvorhaben der Oberstufe

5.1 Unterrichtsvorhaben

Die Darstellung der Unterrichtsvorhaben im schulinternen Lehrplan besitzt den Anspruch, sämtliche im Kernlehrplan angeführten Kompetenzen abzudecken. Dies entspricht der Verpflichtung jeder Lehrkraft, Schülerinnen und Schüler Lerngelegenheiten zu ermöglichen, so dass alle Kompetenzerwartungen des Kernlehrplans von ihnen erfüllt werden können.

Im folgenden Übersichtsraster (**Fehler! Verweisquelle konnte nicht gefunden werden.**) wird die für alle Lehrer:innen gemäß Fachkonferenzbeschluss verbindliche Verteilung der Unterrichtsvorhaben dargestellt. Das Übersichtsraster dient dazu, den KollegInnen einen schnellen Überblick über die Zuordnung der Unterrichtsvorhaben zu den einzelnen Jahrgangsstufen sowie den im Kernlehrplan genannten Kompetenzen, Inhaltsfeldern und inhaltlichen Schwerpunkte zu schaffen.

Der ausgewiesene Zeitbedarf versteht sich als grobe Orientierungsgröße, die nach Bedarf über- oder unterschritten werden kann. Um Freiraum für Vertiefungen, besondere Schülerinteressen, aktuelle Themen bzw. die Erfordernisse anderer besonderer Ereignisse (z.B. Praktika, Kursfahrten o.ä.) zu erhalten, wurden im Rahmen dieses schulinternen Lehrplans ca. 75 Prozent der Bruttounterrichtszeit verplant.

Der Fachkonferenzbeschluss zum „Übersichtsraster Unterrichtsvorhaben“ dient zur Gewährleistung vergleichbarer Standards sowie zur Absicherung von Lerngruppenübertritten und Lehrkraftwechseln für alle Mitglieder der Fachkonferenz.

5.1.1 Unterrichtsvorhaben Jahrgangsstufe EF

JAHRGANGSSTUFE EF			
Unterrichtsvorhaben	Inhaltsfelder	Schwerpunkte der Kompetenzentwicklung	Inhaltliche Schwerpunkte
Einführung in Aufbau und Nutzung von Informatiksystemen und in grundlegende Begrifflichkeiten bis zu ersten Javaprogrammen. Zeitbedarf: 12 Ustd.	• Informatiksysteme • Daten und ihre Strukturierung • Formale Sprachen und Automaten	• Argumentieren • Darstellen und Interpretieren • Kommunizieren und Kooperieren	• Einzelrechner • Dateisystem • Digitalisierung • Daten und Bezeichner • Programme und Übersetzer
Grundlagen der objektorientierten Analyse, Modellierung und Implementierung mit einigen exemplarischen Klassen und Methoden ohne Verwendung von Kontrollstrukturen. Zeitbedarf: 8 Ustd.	• Daten und ihre Strukturierung • Formale Sprachen und Automaten	• Modellieren • Implementieren • Darstellen und Interpretieren • Kommunizieren und Kooperieren	• Objekte und Klassen • Attribute und Methoden • Syntax und Semantik einer Programmiersprache
Grundlagen der objektorientierten Programmierung und algorithmischer Grundstruktur (insbesondere Kontrollstrukturen) in Java. Zeitbedarf: 20 Ustd.	• Daten und ihre Strukturierung • Algorithmen • Formale Sprachen und Automaten	• Argumentieren • Modellieren • Implementieren • Kommunizieren und Kooperieren	• Kontrollstrukturen • Datenfelder • Objekte und Klassen • Syntax und Semantik einer Programmiersprache

JAHRGANGSSTUFE EF			
Unterrichtsvorhaben	Inhaltsfelder	Schwerpunkte der Kompetenzentwicklung	Inhaltliche Schwerpunkte
			Analyse, Entwurf und Implementierung einfacher Algorithmen
Modellierung und Implementierung von Klassen- und Objektbeziehungen im Kontext grafisch repräsentierbarer Objekte. Zeitbedarf: 18 Ustd.	- Daten und ihre Strukturierung - Algorithmen	- Argumentieren - Modellieren - Implementieren - Darstellen und Interpretieren - Kommunizieren und Kooperieren	- Klassenbeziehungen: Hat-, Ist- und Kennt-Beziehung - Unterschiedliche Vererbungsszenarien - Objekte und Klassen - Syntax und Semantik einer Programmiersprache - Analyse, Entwurf und Implementierung einfacher Algorithmen
Analyse von Such- und Sortieralgorithmen und Auswirkungen von Informatiksystemen auf die Gesellschaft Zeitbedarf: Ustd.			

5.1.1.2 Konkretisierung der Unterrichtsvorhaben der EF

Einführung in die Nutzung von Informatiksystemen und in grundlegende Begrifflichkeiten		
Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele und Materialien
1. Speicherung von Informationen • Informatik als Wissenschaft der Verarbeitung von Informationen • Speichern von Daten mit informatischen System am Beispiel der Schulrechner	Die Schüler:innen • nutzen die im Unterricht eingesetzten Informatiksysteme selbstständig, sicher, zielführend und verantwortungsbewusst (D).	
2. Informations- und Datenübermittlung in Netzen • Informatische Kommunikation in Rechnernetzen (lokal) am Beispiel des Schulnetzwerks (Benutzer, Netzwerkordner, Zugriffsrechte) • Informatische Kommunikation in Rechnernetzen (global) am Beispiel einer Internet-Kommunikationsplattform (Benutzer, logische Struktur der Plattform) • Grundlagen der technischen Umsetzung der Rechnerkommunikation (Netzwerkadresse, Paketvermittlung, Protokoll, Client-Server) • Richtlinien zum verantwortungsvollen Umgang mit dem Internet (Proxy-Filter, Datensammlungen)	• nutzen das verfügbare Informatiksystem zur strukturierten Verwaltung und gemeinsamen Verwendung von Daten unter Berücksichtigung der Rechteverwaltung (K). • nutzen das Internet zur Recherche, zum Datenaustausch und zur Kommunikation (K).	
3. Aufbau informatischer Systeme • Komponenten informatischer Systeme, von-Neumann-Architektur und EVA-Prinzip	• beschreiben und erläutern den strukturellen Aufbau und die Arbeitsweise singulärer Rechner am Beispiel der „Von-Neumann-Architektur“ (A).	Recherche im Internet Kurzvortrag

Grundlagen der objektorientierten Programmierung und der Algorithmik anhand von grafischen Beispielen		
Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele und Materialien
1. Einführung in eine Programmierumgebung und Nutzen von Klassenbibliotheken - Vorstellung einer Programmierumgebung und der Programmiersprache Java - Grundbegriffe der Objektorientierung (Klasse, Objekt, Nachricht, Methode, Auftrag, Anfrage) - Nutzung von Klassenbibliotheken und Verwendung von Dokumentationen	Die Schüler:innen - implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I). - implementieren einfache Algorithmen unter Beachtung der Syntax und Semantik einer Programmiersprache (I). - interpretieren Fehlermeldungen und korrigieren den Quellcode (I).	Projekt: Programmgesteuertes Zeichnen kennenlernen von BlueJ SuM-Klassen Stift, Bildschirm mit Dokumentationen Nachrichten an Objekte in der Werkbank Direkteingabe von Java- Befehlen Erste Programme als Nachrichtenfolge
2. Modellierung von Klassen - Modellierung benötigter Klassen aufgrund einer Aufgabenbeschreibung - Erstellung der Dokumentationen der Klassen unter Berücksichtigung von Parametern und Rückgabewerten	- ermitteln bei der Analyse einfacher Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M). - modellieren Klassen mit ihren Attributen, ihren Methoden und Assoziationsbeziehungen (M). - ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen oder lineare Datensammlungen zu (M). - dokumentieren Klassen durch Beschreibung der Funktionalität t der Methoden (D).	Projekt: Pfeilwurf Klassenwurf (Pfeil, Zielscheibe) als Auftrag an die Lehrkraft

3. Einfache Algorithmen, Verwendung von Kontrollstrukturen • Implementierung einfacher Algorithmen unter Benutzung der selbst modellierten Klassen • Verwendung und Dokumentation von Kontrollstrukturen • Verwendung weiterer Klassenbibliotheken Der Tastaturpuffer als Beispiel einer Schlange	• analysieren und erläutern einfache Algorithmen und Programme (A). • entwerfen einfache Algorithmen und stellen sie umgangs- sprachlich und grafisch dar (M). • implementieren Algorithmen unter Verwendung von Variablen und Wertzuweisungen, Kontrollstrukturen sowie Methodenaufrufen (I)	Verwendung der von Lehrkraft realisierten Klassenbibliothek Schleife, Verzweigung Struktogramme SuM: Tastatur
4. Sicherung und Anwendung von einfachen Algorithmen und Kontrollstrukturen • Eigenständige Anwendung des gelernten Wissens auf ein neues Projekt • Verwendung weiterer Klassenbibliotheken • Beobachtung einer Vererbung • UML-Diagramme der Klassenbibliothek ohne Sichtbarkeiten	• stellen Klassen, Assoziations- und Vererbungsbeziehungen in Diagrammen grafisch dar (D).	Projekt: Freihandzeichnen SuM: Maus, Buntstift

Modellierung und Implementierung von Klassen- und Objektbeziehungen anhand von grafischen Beispielen		
Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele und Materialien
1. Entwurf eigener Klassen - Entwicklung eines ersten UML-Diagramms zum gewählten Projekt - Einführung private und öffentlicher Methoden - Hole- und Setze-Methoden für Attribute - Assoziation von Klassen am Beispiel der Hat- Beziehung	Die Schüler:innen - ermitteln bei der Analyse einfacher Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M). - modellieren Klassen mit ihren Attributen, ihren Methoden und Assoziationsbeziehungen (M). - ordnen Klassen, Attributen und Methoden ihren Sichtbarkeits- bereich zu (M). - stellen Klassen, Assoziations- und Vererbungsbeziehungen in Diagrammen grafisch dar (D).	Projekt: Billard UML-Editor für BlueJ
2. Implementation eigener Klassen - Implementation der Klassen aus der ersten Sequenz - Erweiterung um weitere Funktionalität - Einführung von Zustandsvariablen für Attribute - Verdeutlichung des Factory-Aspekts, indem zu einem Bauplan viele Exemplare (Objekte) erzeugt werden	- implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I). - implementieren Algorithmen unter Verwendung von Variablen und Wertzuweisungen, Kontrollstrukturen sowie Methodenaufrufen (I). - implementieren einfache Algorithmen unter Beachtung der Syntax und Semantik einer Programmiersprache (I). - interpretieren Fehlermeldungen und korrigieren den Quellcode (I).	Fehlermeldungen, Ursachen und mögliche Lösungen sammeln

3. Erweiterung des Modells Ergänzung um eine weitere Klasse, so dass Kennt-Beziehungen notwendig werden.		Tisch
4. Einführung spezieller Unterklassen - Einführung der Vererbung durch Spezialisierung - Entwurf und Implementation von Unterklassen	- stellen Klassen, Assoziations- und Vererbungsbeziehungen in Diagrammen grafisch dar (D). - modellieren Klassen unter Verwendung von Vererbung (M). - modifizieren einfache Algorithmen und Programme (I).	Drallkugel, Bremskugel, Nummernkugel, ...
5. Arbeitsteiliger Entwurf einer komplexeren Modellierung - Entwicklung einzelner Klassendokumentationen in Kleingruppen - Zusammenstellung aller entworfenen Klassen zu einem Gesamtmodell als Grundlage für die Implementation - Entwurf einer abstrakten Oberklasse für Klassen gemeinsamen Eigenschaften und Methoden (Vererbung durch Generalisierung)	analysieren und erläutern eine objektorientierte Modellierung (A).	Projekt: Schatzsuche Beim Entwurf der Klassen darauf achten, dass Polymorphie und späte Bindung möglich sind
6. Implementation in arbeitsteiligen Kleingruppen - Entwicklung der Dokumentation der entworfenen Klassen - Implementation der Klassen - Genauere Projektplanung (Arbeitsplan) - Polymorphie bei den Unterklassen der abstrakten Oberklasse	testen Programme schrittweise anhand von Beispielen (I).	
7. Späte Bindung - Erzeugen von beliebigen unterschiedlichen Objekten während der Laufzeit - Einsatz des Debuggers zur Veranschaulichung der Laufzeit-Situation	stellen den Zustand eines Objekts dar (D).	Debugger

Vorbereitung von dynamischen Datenstrukturen und Verwendung abstrakter Klassen		
Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele und Materialien
1. Modellierung der Ausgangssituation Projekt mit Objekten vieler gemeinsamer Eigenschaften, die zu einer Modellierung einer abstrakten Oberklasse mit mehreren Unterklassen führen. Diese Objekte müssen so gewählt sein, dass eine anschauliche Verkettung möglich ist.	Die Schüler:innen - ermitteln bei der Analyse einfacher Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M). - modellieren Klassen mit ihren Attributen, ihren Methoden und Assoziationsbeziehungen (M). - ordnen Klassen, Attributen und Methoden ihren Sichtbarkeitsbereich zu (M). - stellen Klassen, Assoziations- und Vererbungsbeziehungen in Diagrammen grafisch dar (D).	Projekt: Zug
2. Implementation der Klassen - Gemeinsame Erstellung der abstrakten Oberklasse - arbeitsteilige Erstellung der Unterklassen - Bereitstellung der class-Dateien und Zusammenstellung zu einem kombinierten Projekt	- implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I). - implementieren Algorithmen unter Verwendung von Variablen und Wertzuweisungen, Kontrollstrukturen sowie Methodenaufrufen (I). - implementieren einfache Algorithmen unter Beachtung der Syntax und Semantik einer Programmiersprache (I). - interpretieren Fehlermeldungen und korrigieren den Quellcode (I).	Waggon Spezielle Waggon Zusammenstellung der Waggon zu einem Zug

3. Verkettung von Objekten • Verkettung als Möglichkeit, beliebig viele Objekte hintereinander zu hängen • Gemeinsame Ergänzung der Oberklasse um die Verkettungseigenschaft • Zusammenstellung beliebiger Objekte unter Verwendung anonymer Objekte	• ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen oder lineare Datensammlungen zu (M).	Statt alle Waggons einzeln zu bewegen, zieht die Lok den ersten Waggon, dieser den zweiten usw.
---	--	---

Analyse von Such- und Sortialgorithmen und Auswirkungen von Informatiksystemen auf die Gesellschaft		
Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele und Materialien
1. Speicherung von Zeichen • Umwandlung von ganzen Zahlen in Binärcodes und Zeichen	Die Schüler:innen • stellen ganze Zahlen und Zeichen in Binärcodes dar (D). • interpretieren Binärcodes als Zahlen und Zeichen (D).	Tabelle des ASCII-Codes in Dezimal- und Hexadezimaldarstellung Arbeitsblatt mit „Geheimcode“ in binärer Darstellung Projekte mit Komponenten und Programmgenerator: Cäsar-Codierung Umwandlungen Datenübertragung (Parität) Weitere Beispiele: IBAN, ISBN, Barcodes

2. Such- und Sortieralgorithmen • Analyse eines fertigen Programmtextes zu Such- und Sortieralgorithmen • Das Feld als Datenstruktur mit direktem Zugriff • Umwandlungen von Zeichen in Zahlen und umgekehrt	• ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen oder lineare Datensammlungen zu (M). • analysieren Such- und Sortieralgorithmen und wenden sie auf Beispiele an (D).	Projekt: Dekodieren Arbeitsblatt zum Programmverständnis
3. Weitere Sortieralgorithmen • Entwurf zusätzlicher Sortieralgorithmen • Zählen von Vergleichs- und Vertauschoperationen an realisierten Sortieralgorithmen	• entwerfen einen weiteren Algorithmus zum Sortieren (M). • beurteilen die Effizienz von Algorithmen am Beispiel von Sortierverfahren hinsichtlich Zeitaufwand und Speicherplatzbedarf (A).	Verdecktes Sortieren von Karten Ein bestimmte Kartenfolge wird nach den Verfahren sortiert und die Operationen werden ausgezählt. Ein fertiges Programm zählt für größere Datenmengen.
4. Informatiksysteme und ihre Auswirkungen auf die Gesellschaft • Auswirkungen der Automatisierung mit historischen Beispielen	• bewerten anhand von Fallbeispielen die Auswirkungen des Einsatzes von Informatiksystemen (A). • erläutern wesentliche Grundlagen der Geschichte der digitalen Datenverarbeitung (A).	Code knacken (Enigma, NSA) Bibliotheksarchive Volkszählungen Buchausleihe

5.1.2 Unterrichtsvorhaben Jahrgangsstufe Q1

JAHRGANGSSTUFE Q1			
Unterrichtsvorhaben	Inhaltsfelder	Schwerpunkte der Kompetenzentwicklung	Inhaltliche Schwerpunkte
Wiederholung der objektorientierten Modellierung und Programmierung anhand einer kontextbezogenen Problemstellung. Zeitbedarf: 8 Ustd.	• Daten und ihre Strukturierung • Algorithmen • Formale Sprachen und Automaten • Informatiksysteme	•	• Objekte und Klassen • Analyse, Entwurf und Implementierung von Algorithmen • Syntax und Semantik einer Programmiersprache • Nutzung von Informatiksystemen
Modellierung und Implementierung von Anwendungen mit dynamischen, linearen Datenstrukturen. Zeitbedarf: 18 Ustd.	• Daten und ihre Strukturierung • Algorithmen • Formale Sprachen und Automaten	•	• Objekte und Klassen • Analyse, Entwurf und Implementierung einfacher Algorithmen • Algorithmen in ausgewählten informatischen Kontexten • Syntax und Semantik einer Programmiersprache

JAHRGANGSSTUFE Q1			
Unterrichtsvorhaben	Inhaltsfelder	Schwerpunkte der Kompetenzentwicklung	Inhaltliche Schwerpunkte
Suchen und Sortieren auf linearen Datenstrukturen. Zeitbedarf: 22 Ustd.	• Daten und ihre Strukturierung • Algorithmen • Formale Sprachen und Automaten	•	• Analyse, Entwurf und Implementierung von Algorithmen • Algorithmen in ausgewählten informatischen Kontexten • Syntax und Semantik einer Programmiersprache
Modellierung und Implementierung von Anwendungen mit dynamischen, nichtlinearen Datenstrukturen. Zeitbedarf: 24 Ustd.	• Daten und ihre Strukturierung • Algorithmen • Formale Sprachen und Automaten	•	• Objekte und Klassen • Analyse, Entwurf und Implementierung einfacher Algorithmen • Algorithmen in ausgewählten informatischen Kontexten • Syntax und Semantik einer Programmiersprache
Sicherheit und Datenschutz in Netzstrukturen Zeitbedarf: 3 Ustd.	• Informatiksysteme • Informatik, Mensch und Gesellschaft	•	• Einzelrechner und Rechnernetzwerke • Sicherheit • Nutzung von Informatiksystemen, Wirkungen der Automatisierung

5.1.2.2 Konkretisierung der Unterrichtsvorhaben der Q1

Wiederholung der objektorientierten Modellierung und Programmierung anhand einer kontextbezogenen Problemstellung		
Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele und Materialien
1. Entwicklung und Testen einer Fachklasse (Model) • Erstellen eines Implementationsdiagramms für die Klasse • Schreiben einer Dokumentation der Klasse • Wiederholen der Entwicklung von Klassen sowie Get- und Set-Methoden	Die Schüler:innen • analysieren und erläutern objektorientierte Modellierungen (A). • beurteilen die syntaktische Korrektheit und die Funktionalität von Programmen (A). • modellieren Klassen mit ihren Attributen, Methoden und ihren Assoziationsbeziehungen unter Angabe von Multiplizitäten (M). • ordnen Klassen, Attributen und Methoden ihre Sichtbarkeitsbereiche zu (M). • dokumentieren Klassen (D). • implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I). • nutzen die Syntax und Semantik einer Programmiersprache bei der Implementierung und zur Analyse von Programmen (I). • wenden eine didaktisch orientierte Entwicklungsumgebung zur Demonstration, zum Entwurf, zur Implementierung und zum Test von Informatiksystemen an (I).	Bruchrechner-Projekt Klasse Bruch Testen als Exemplar der Klasse in der Werkbank

	• testen Programme systematisch anhand von Beispielen (I). • interpretieren Fehlermeldungen und korrigieren den Quellcode (I).	
2. Entwicklung einer Rechenklasse • Kommunikation zwischen Klassen • Entwicklung von Algorithmen	• stellen Klassen und ihre Beziehungen in Diagrammen grafisch dar (D). • stellen die Kommunikation zwischen Objekten grafisch dar (D). • modellieren Klassen mit ihren Attributen, Methoden und ihren Assoziationsbeziehungen unter Angabe von Multiplizitäten (M).	Klasse Bruchrechner
3. Entwicklung einer Oberflächenklasse (Control, View)	• s. O.	
4. MVC-Konzept	• analysieren und erläutern objekt-orientierte Modellierungen (A).	Erläutern des Konzepts anhand des Beispiels

Modellierung und Implementierung von Anwendungen mit dynamischen, linearen Datenstrukturen		
Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele und Materialien
1. Die Datenstruktur Schlange im Anwendungskontext unter Nutzung der Klasse Queue - Analyse der Problemstellung, Ermittlung von Objekten, ihren Eigenschaften und Operationen - Erarbeitung der Funktionalität der Klasse Queue - Modellierung und Implementierung der Anwendung unter Verwendung eines oder mehrerer Objekte der Klasse Queue	Die Schüler:innen - erläutern Operationen dynamischer (linearer oder nicht-linearer) Datenstrukturen (A). - analysieren und erläutern Algorithmen und Programme (A). - beurteilen die syntaktische Korrektheit und die Funktionalität von Programmen (A). - ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen sowie lineare und nichtlineare Datensammlungen zu (M). - ermitteln bei der Analyse von Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M). - modifizieren Algorithmen und Programme (I). - implementieren iterative und rekursive Algorithmen auch unter Verwendung von dynamischen Datenstrukturen (I). - nutzen die Syntax und Semantik einer Programmiersprache bei der Implementierung und zur Analyse von Programmen (I).	Beispiel: Patientenwarteschlange (jeder kennt seinen Nachfolger bzw. alternativ: seinen Vorgänger) Sobald ein Patient in einer Arztpraxis eintrifft, werden sein Name und seine Krankenkasse erfasst. Die Verwaltung der Patientenwarteschlange geschieht über eine Klasse, die hier als Wartezimmer bezeichnet wird. Wesentliche Operationen sind das „Hinzufügen“ eines Patienten und das „Entfernen“ eines Patienten, wenn er zur Behandlung gerufen wird. Die Simulationsanwendung stellt eine GUI zur Verfügung, legt ein Wartezimmer an und steuert die Abläufe. Wesentlicher Aspekt des Projektes ist die Modellierung des Wartezimmers mit Hilfe der Klasse Queue. Anschließend wird der Funktionsumfang der Anwendung erweitert: Patienten können sich zusätzlich in die Warteschlange zum Blutdruckmessen einreihen. Objekte werden von zwei Schlangen verwaltet.

2. Die Datenstruktur Stapel im Anwendungskontext unter Nutzung der Klasse Stack • Analyse der Problemstellung, Ermittlung von Objekten, ihren Eigenschaften und Operationen • Erarbeitung der Funktionalität der Klasse Stack • Modellierung und Implementierung der Anwendung unter Verwendung eines oder mehrerer Objekte der Klasse Stack	• interpretieren Fehlermeldungen und korrigieren den Quellcode (I). • testen Programme systematisch anhand von Beispielen (I). • stellen lineare und nichtlineare Strukturen grafisch dar und erläutern ihren Aufbau (D). • implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I).	
3. Die Datenstruktur lineare Liste im Anwendungskontext unter Nutzung der Klasse List • Erarbeitung der Vorteile der Klasse List im Gegensatz zu den bereits bekannten linearen Strukturen • Modellierung und Implementierung einer kontextbezogenen Anwendung unter Verwendung der Klasse List.		Beispiel: Abfahrtslauf Bei einem Abfahrtslauf kommen die Skifahrer nacheinander an und werden nach ihrer Zeit in eine Rangliste eingeordnet. Diese Rangliste wird in einer Anzeige ausgegeben. Ankommende Abfahrer müssen an jeder Stelle der Struktur, nicht nur am Ende oder Anfang eingefügt werden können.
4. Vertiefung - Anwendungen von Listen, Stapeln oder Schlangen in mindestens einem weiteren Kontext		Beispiel: Skispringen Nach dem Sprung erhält der Springer eine Punktzahl und wird nach dieser Punktzahl in eine Rangliste eingeordnet. Die besten 30 Springer qualifizieren sich für den zweiten Durchgang. Sie starten in umgekehrter Reihenfolge gegenüber der Platzierung auf der Rangliste. Nach dem Sprung erhält der Springer wiederum eine Punktzahl und wird nach der Gesamtpunktzahl aus beiden Durchgängen in die endgültige Rangliste eingeordnet.

Suchen und Sortieren auf linearen Datenstrukturen		
Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele und Materialien
1. Suchen von Daten in Listen und Arrays - Lineare Suche in Listen und in Arrays (auch zweidimensional) - Binäre Suche in Arrays als Beispiel für rekursives Problemlösen (ggf. aber auch iterativ) - Untersuchung der beiden Suchverfahren hinsichtlich ihrer Effizienz (Laufzeitverhalten, Speicherbedarf)	Die Schüler:innen - analysieren und erläutern Algorithmen und Programme (A). - beurteilen die syntaktische Korrektheit und die Funktionalität von Programmen (A). - beurteilen die Effizienz von Algorithmen unter Berücksichtigung des Speicherbedarfs und der Zahl der Operationen (A). - entwickeln iterative und rekursive Algorithmen unter Nutzung der Strategien „Modularisierung“ und „Teilen und Herrschen“ (M). - modifizieren Algorithmen und Programme (I). - implementieren iterative und rekursive Algorithmen auch unter Verwendung von dynamischen Datenstrukturen (I). - implementieren und erläutern iterative und rekursive Such- und Sortierverfahren (I). - nutzen die Syntax und Semantik einer Programmiersprache bei der	Projekt: Bundesjugendspiele Die Teilnehmer an Bundesjugendspielen nehmen an drei Disziplinen teil und erreichen dort Punktzahlen. Diese werden in einer Wettkampfkarte eingetragen und an das Wettkampfbüro gegeben. Zur Vereinfachung sollte sich das Modell auf die drei Disziplinen „Lauf“, „Sprung“ und „Wurf“ beschränken. Im Wettkampfbüro wird das Ergebnis erstellt. Das Programm soll dafür zunächst den Besten einer Disziplin heraussuchen können und später das gesamte Ergebnis nach gewissen Kriterien sortieren können. Beispiel: Bundesjugendspiele (s.o.)
2. Sortieren in Listen und Arrays - Entwicklung und Implementierung von iterativen und rekursiven Sortierverfahren - Entwicklung und Implementierung eines einfachen Sortierverfahrens für eine Liste - Entwicklung eines rekursiven Sortierverfahren für eine Liste (Quicksort)		

3. Untersuchung der Effizienz der Sortierv Verfahren „Sortieren durch direktes Einfügen“ und „Quicksort“ auf linearen Listen • Grafische Veranschaulichung der Sortierv Verfahren • Untersuchung der Anzahl der Vergleichsoperationen und des Speicherbedarf bei beiden Sortierv Verfahren • Beurteilung der Effizienz der beiden Sortierv Verfahren	Implementierung und zur Analyse von Programmen (I). • interpretieren Fehlermeldungen und korrigieren den Quellcode (I). • testen Programme systematisch anhand von Beispielen (I). • stellen iterative und rekursive Algorithmen umgangssprachlich und grafisch dar (D). • implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I).	Beispiel: Bundesjugendspiele
--	--	-------------------------------------

Modellierung und Implementierung von Anwendungen mit dynamischen, nichtlinearen Datenstrukturen		
Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele und Materialien
1. Analyse von Baumstrukturen in verschiedenen Kontexten • Grundlegende Begriffe (Grad, Tiefe, Höhe, Blatt, Inhalt, Teilbaum, Ebene, Vollständigkeit) • Aufbau und Darstellung von binären Bäumen anhand von Baumstrukturen in verschiedenen Kontexten	Die Schüler:innen • erläutern Operationen dynamischer (linearer oder nicht-linearer) Datenstrukturen (A). • analysieren und erläutern Algorithmen und Programme (A). • beurteilen die syntaktische Korrektheit und die Funktionalität von Programmen (A). • ermitteln bei der Analyse von Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M). • ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen sowie lineare und nichtlineare Datensammlungen zu (M). • modellieren abstrakte und nicht abstrakte Klassen unter Verwendung von Vererbung durch Spezialisieren und Generalisieren (M). • verwenden bei der Modellierung geeigneter Problemstellungen die Möglichkeiten der Polymorphie (M). • entwickeln iterative und rekursive Algorithmen unter Nutzung der Konstruktionsstrategien • „Modularisierung“ und „Teilen und	Beispiel: Suchbäume (zur sortierten Speicherung von Daten, entwickelt aus dem Verfahren der binären Suche) Alle Inhalte, die nach einer Ordnung vor dem Inhalt im aktuellen Teilbaum stehen, sind in dessen linkem Teilbaum, alle die nach dem Inhalt im aktuellen Teilbaum stehen, sind in dessen rechtem Teilbaum. (Dies gilt für alle Teilbäume.) Beispiel: Codierungsbäume für Codierungen, deren Alphabet aus genau zwei Zeichen besteht Morse hat Buchstaben als Folge von Punkten und Strichen codiert. Diese Codierungen können in einem Binärbaum dargestellt werden, so dass ein Übergang zum linken Teilbaum einem Punkt und ein Übergang zum rechten Teilbaum einem Strich entspricht. Wenn man im Gesamtbaum startet und durch Übergänge zu linken oder rechten Teilbäumen einen Pfad zum gewünschten Buchstaben sucht, erhält man die Morsecodierung des Buchstabens.

	Herrschen“ (M). • implementieren iterative und rekursive Algorithmen auch unter Verwendung von dynamischen Datenstrukturen (I). • modifizieren Algorithmen und Programme (I). • nutzen die Syntax und Semantik einer Programmiersprache bei der Implementierung und zur Analyse von Programmen (I). • interpretieren Fehlermeldungen und korrigieren den Quellcode (I). • testen Programme systematisch anhand von Beispielen (I). • stellen lineare und nichtlineare Strukturen grafisch dar und erläutern ihren Aufbau (D). • stellen iterative und rekursive Algorithmen umgangssprachlich und grafisch dar (D). • implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I).	Beispiel: Termbaum Der Aufbau von Termen wird mit Hilfe von binären Baumstrukturen verdeutlicht Beispiel: Entscheidungsbäume Um eine Entscheidung zu treffen, werden mehrere Fragen mit ja oder nein beantwortet. Die Fragen, die möglich sind, wenn die Antwort auf eine Frage mit „ja“ beantwortet wird, befinden sich im linken Teilbaum, die Fragen, die möglich sind, wenn die Antwort „nein“ lautet, stehen im rechten Teilbaum.
2. Die Datenstruktur Binärbaum im Anwendungskontext unter Nutzung der Klasse BinaryTree • Analyse der Problemstellung, Ermittlung von Objekten, ihren Eigenschaften und Operationen im Anwendungskontext • Modellierung eines Entwurfsdiagramms und Entwicklung eines Implementationsdiagramms • Erarbeitung der Klasse BinaryTree und beispielhafte Anwendung der Operationen • Implementierung der Anwendung oder von Teilen der Anwendung • Traversierung eines Binärbaums im Pre-, In- und Postorderdurchlauf		Beispiele: s. o. Durchlaufarten: Suchbaum, Termbaum In-Order Termbaum in UPN: Post-Ordner

3. Die Datenstruktur binärer Suchbaum im Anwendungskontext unter Verwendung der Klasse BinarySearchTree • Analyse der Problemstellung, Ermittlung von Objekten, ihren Eigenschaften und Operationen • Modellierung eines Entwurfsdiagramms und Entwicklung eines Implementationsdiagramm, grafische Darstellung eines binären Suchbaums und Erarbeitung der Struktureigenschaften • Erarbeitung der Klasse BinarySearchTree und Einführung des Interface ComparableContent • zur Realisierung einer geeigneten Ordnungsrelation • Implementierung der Anwendung oder von Teilen der Anwendung inklusive einer sortierten Ausgabe des Baums		Beispiel: Informatikerbaum als Suchbaum In einem binären Suchbaum werden die Namen und die Geburtsdaten von Informatikern lexikographisch geordnet abgespeichert. Alle Namen, die nach dieser Ordnung vor dem Namen im aktuellen Teilbaum stehen, sind in dessen linkem Teilbaum, alle die nach dem Namen im aktuellen Teilbaum stehen, sind in dessen rechtem Teilbaum. (Dies gilt für alle Teilbäume.) Folgende Funktionalitäten werden benötigt: • Einfügen der Informatiker-Daten in den Baum • Suchen nach einem Informatiker über den Schlüssel Name • Ausgabe des kompletten Datenbestands in nach Namen sortierter Reihenfolge
4. Übung und Vertiefungen der Verwendung von Binärbäumen oder binären Suchbäumen anhand weiterer Problemstellungen		Beispiel: Huffman-Codierung Aufgeteilt in mehrere Teilprojekte mit BinaryTree und BinarySearchTree

Sicherheit und Datenschutz in Netzstrukturen		
Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele und Materialien
1. Daten in Netzwerken und Sicherheitsaspekte in Netzen sowie beim Zugriff auf Datenbanken - Beschreibung eines Webserverzugriffs im Netz anhand eines Anwendungskontextes und einer Client-Server-Struktur zur Klärung der Funktionsweise einer Website-Abfrage - Netztopologien als Grundlage von Client-Server-Strukturen und TCP/IP-Schichtenmodell als Beispiel für eine Paketübermittlung in einem Netz - Vertraulichkeit, Integrität, Authentizität in Netzwerken sowie symmetrische und asymmetrische kryptografische Verfahren (Cäsar-, Vigenère-, RSA-Verfahren) als Methoden Daten im Netz verschlüsselt zu übertragen	Die Schüler:innen - beschreiben und erläutern Topologien, die Client-Server-Struktur und Protokolle sowie ein Schichtenmodell in Netzwerken (A). - analysieren und erläutern Eigenschaften und Einsatzbereiche symmetrischer und asymmetrischer Verschlüsselungsverfahren (A). - untersuchen und bewerten anhand von Fallbeispielen die Auswirkungen des Einsatzes von Informatiksystemen, die Sicherheit von Informatiksystemen sowie die Einhaltung der Datenschutzbestimmungen und des Urheberrechts (A). - untersuchen und bewerten Problemlagen, die sich aus dem Einsatz von Informatiksystemen ergeben, hinsichtlich rechtlicher Vorgaben, ethischer Aspekte und gesellschaftlicher Werte unter Berücksichtigung unterschiedlicher Interessenlagen (A). - nutzen bereitgestellte Informatiksysteme und das Internet reflektiert zum Erschließen, zur Aufbereitung und Präsentation - fachlicher Inhalte (D).	Materialien: Protokolle: HTTP, HTTPS, POP3 Dabei Analyse eines fertigen POP3-Clients
2. Fallbeispiele zur Datenschutzproblematik und zum Urheberrecht		Materialien: Materialblatt Datenschutzgesetz

5.1.3 Unterrichtsvorhaben Jahrgangsstufe Q2

JAHRGANGSSTUFE Q2			
Unterrichtsvorhaben	Inhaltsfelder	Schwerpunkte der Kompetenzentwicklung	Inhaltliche Schwerpunkte
Modellierung und Nutzung von relationalen Datenbanken in Anwendungskontexten Zeitbedarf: 20 Ustd.	• Daten und ihre Strukturierung • Algorithmen • Formale Sprachen und Automaten • Informatik, Mensch und Gesellschaft	•	• Datenbanken • Algorithmen in ausgewählten informatischen Kontexten • Syntax und Semantik einer Programmiersprache • Sicherheit
Endliche Automaten und formale Sprachen Zeitbedarf: 24 Ustd.	• Endliche Automaten und formale Sprachen • Algorithmen	•	• Endliche Automaten • Grammatiken regulärer Sprachen • Möglichkeiten und Grenzen von Automaten und formalen Sprachen
Prinzipielle Arbeitsweise eines Computers und Grenzen der Automatisierbarkeit Zeitbedarf: 12 Ustd.	• Informatiksysteme • Informatik, Mensch und Gesellschaft	•	• Einzelrechner und Rechnernetzwerke • Grenzen der Automatisierung

5.1.3.2 Konkretisierung der Unterrichtsvorhaben der Q2

Modellierung und Nutzung von relationalen Datenbanken in Anwendungskontexten		
Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele und Materialien
1. Nutzung von relationalen Datenbanken - Aufbau von Datenbanken und Grundbegriffe - Entwicklung von Fragestellungen zur vorhandenen Datenbank - Analyse der Struktur der vorgegebenen Datenbank und Erarbeitung der Begriffe Tabelle, Attribut, Datensatz, Datentyp, Primärschlüssel, Fremdschlüssel, Datenbankschema - SQL-Abfragen - Analyse vorgegebener SQL-Abfragen und Erarbeitung der Sprachelemente von SQL (SELECT (DISTINCT) ... FROM, WHERE, AND, OR, NOT) auf einer Tabelle - Analyse und Erarbeitung von SQL- Abfragen auf einer und mehrerer Tabelle zur Beantwortung der Fragestellungen (JOIN, UNION, AS, GROUP BY, ORDER BY, ASC, DESC, COUNT, MAX, MIN, SUM, Arithmetische Operatoren: +, -, *, /, (...), Vergleichsoperatoren: =, <>, >, <, >=, <=, LIKE, BETWEEN, IN, IS NULL)	Die Schüler:innen - erläutern die Eigenschaften und den Aufbau von Datenbanksystemen unter dem Aspekt der sicheren Nutzung (A). - analysieren und erläutern die Syntax und Semantik einer Datenbankabfrage (A). - analysieren und erläutern eine Datenbankmodellierung (A). - erläutern die Eigenschaften normalisierter Datenbankschemata (A). - bestimmen Primär- und Sekundärschlüssel (M). - ermitteln für anwendungsbezogene Problemstellungen Entitäten, zugehörige Attribute, Relationen und Kardinalitäten (M). - modifizieren eine Datenbankmodellierung (M). - modellieren zu einem Entity-Relationship-Diagramm ein relationales Datenbankschema (M).	Beispiel: VideoCenter VideoCenter ist die Simulation einer Online-Videothek für den Informatik-Unterricht mit Webfrontends zur Verwaltung der Kunden, der Videos und der Ausleihe. Außerdem ist es möglich direkt SQL-Abfragen einzugeben. Es ist auch möglich, die Datenbank herunter zu laden und lokal zu installieren. Unter http://dokumentation.videocenter.schule.de (abgerufen: 30. 03. 2014) findet man den Link zu dem VideoCenter-System sowie nähere Informationen. Lesenswert ist auch die dort verlinkte „Dokumentation der Fallstudie“ mit didaktischem Material, welches alternativ bzw. ergänzend zu der im Folgenden beschriebenen Durchführung verwendet werden kann.

2. Modellierung von relationalen Datenbanken - Entity-Relationship-Diagramm - Ermittlung von Entitäten, zugehörigen Attributen, Relationen und Kardinalitäten in Anwendungssituationen und Modellierung eines Datenbankentwurfs in Form eines Entity-Relationship-Diagramms - Erläuterung und Modifizierung einer Datenbankmodellierung - Entwicklung einer Datenbank aus einem Datenbankentwurf - Modellierung eines relationalen Datenbankschemas zu einem Entity-Relationship-Diagramm inklusive der Bestimmung von Primär- und Sekundärschlüsseln - Redundanz, Konsistenz und Normalformen - Untersuchung einer Datenbank hinsichtlich Konsistenz und Redundanz in einer Anwendungssituation - Überprüfung von Datenbankschemata hinsichtlich der 1. bis 3. Normalform und Normalisierung (um Redundanzen zu vermeiden und Konsistenz zu gewährleisten)	- überführen Datenbankschemata in die 1. bis 3. Normalform (M). - verwenden die Syntax und Semantik einer Datenbankabfragesprache, um Informationen aus einem Datenbanksystem zu extrahieren (I). - ermitteln Ergebnisse von Datenbankabfragen über mehrere verknüpfte Tabellen (D). - stellen Entitäten mit ihren Attributen und die Beziehungen zwischen Entitäten in einem Entity-Relationship-Diagramm grafisch dar (D). - überprüfen Datenbankschemata auf vorgegebene Normalisierungseigenschaften (D).	Beispiel: viele kleine Beispiele, als großes Beispiel Sportverein Anmerkung: Ein alternativer Verlauf, der bei der Modellierung beginnt, dann von ER-Diagrammen über die Relationenalgebra zur Realisierung als Relationen und zur Programmierung mit SQL führt, ist ebenso denkbar.
--	---	---

Endliche Automaten und formale Sprachen		
Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele und Materialien
1. Endliche Automaten - Vom Automaten in den Schülerinnen und Schülern bekannten Kontexten zur formalen Beschreibung eines endlichen Automaten - Untersuchung, Darstellung und Entwicklung endlicher Automaten - Umwandlung von nichtdeterministischen Akzeptoren in deterministische Akzeptoren - Simulation eines Akzeptors in der Programmiersprache	Die Schüler:innen - analysieren und erläutern die Eigenschaften endlicher Automaten einschließlich ihres Verhaltens auf bestimmte Eingaben (A). - analysieren und erläutern Grammatiken regulärer Sprachen (A). - zeigen die Grenzen endlicher Automaten und regulärer Grammatiken im Anwendungszusammenhang auf (A).	Als roter Faden der Reihe dient die Idee, einen „Taschenrechner“ für Terme zu entwickeln. Beispiele: Eingabe eines Taschenrechners, Akzeptor für bestimmte Zahlen, Akzeptor für Teilwörter in längeren Zeichenketten (auch nichtdeterministisch), Cola-Automat, Geldspielautomat, Zahlakzeptor in Java
2. Untersuchung und Entwicklung von Grammatiken regulärer Sprachen - Erarbeitung der formalen Darstellung regulärer Grammatiken - Untersuchung, Modifikation und Entwicklung von Grammatiken - Entwicklung von endlichen Automaten zum Erkennen regulärer Sprachen die durch Grammatiken gegeben werden - Entwicklung regulärer Grammatiken zu endlichen Automaten	- ermitteln die formale Sprache, die durch eine Grammatik erzeugt wird (A). - entwickeln und modifizieren zu einer Problemstellung endliche Automaten (M). - entwickeln zur akzeptierten Sprache eines Automaten die zugehörige Grammatik (M). - entwickeln zur Grammatik einer regulären Sprache einen zugehörigen endlichen Automaten (M).	Beispiele: reguläre Grammatik für Wörter mit ungerader Parität, Grammatik für Wörter, die bestimmte Zahlen repräsentieren, Satzgliedergrammatik

3. Grenzen endlicher Automaten	• modifizieren Grammatiken regulärer Sprachen (M). • entwickeln zu einer regulären Sprache eine Grammatik, die die Sprache erzeugt (M). • stellen endliche Automaten in Tabellen oder Graphen dar und überführen sie in die jeweils andere Darstellungsform (D), • ermitteln die Sprache, die ein endlicher Automat akzeptiert (D). • beschreiben an Beispielen den Zusammenhang zwischen Automaten und Grammatiken (D).	Beispiele: Klammerausdrücke, $a^n b^n$ im Vergleich zu $(ab)^n$
4. Kontextfreie und kontextsensitive Grammatiken, Chomsky-Hierarchie • Syntaxbäume • Eindeutigkeit • Grenzen kontextfreier Grammatiken		Grammatiken für Klammerterme, Satzgliedergrammatiken, Grammatik für if-Abfragen, $a^n b^n c^n$
5. Compilerbau • Stufen eines Compilers • Realisierung des Parsers und des Codierers in der Programmiersprache		Arbeitsblatt Compilerfehler analysieren Rollenspiele für Parser und Codierer Beispiel: Termcompiler Der Scanner wurde schon oben behandelt, beim Codierer wird ein Termbaum benutzt, siehe Q1-IV

Prinzipielle Arbeitsweise eines Computers und Grenzen der Automatisierbarkeit		
Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele und Materialien
1. Von-Neumann-Architektur und die Ausführung maschinennaher Programme • prinzipieller Aufbau einer von Neumann-Architektur mit CPU, Rechenwerk, Steuerwerk, Register und Hauptspeicher • maschinennahe Befehle und ihre Repräsentation in einem Binär- Code, der in einem Register gespeichert werden kann • Analyse und Erläuterung der Funktionsweise eines einfachen maschinennahen Programms	Die Schüler:innen • erläutern die Ausführung eines einfachen maschinennahen Programms sowie die Datenspeicherung auf einer „Von-Neumann-Architektur“ (A). • untersuchen und beurteilen Grenzen des Problemlösens mit Informatiksystemen (A).	Beispiel: Addition von 4 zu einer eingegeben Zahl mit einem Rechnermodell Materialien: Verwenden eines Von-Neumann-Simulators, z. B: Rechner-Baukasten, MIPS-Generator
2. Grenzen der Automatisierbarkeit • Vorstellung des Halteproblems • Unlösbarkeit des Halteproblems • Beurteilung des Einsatzes von Informatiksystemen hinsichtlich prinzipieller Möglichkeiten und prinzipieller Grenzen		Beispiel: Halteproblem