

Schulinterner Lehrplan am Heinrich-Heine-Gymnasium

Informatik 5/6



Stand: September 2024

Fach:	Informatik	Jahrgangsstufe:	5/6
Aufgaben und Ziele des Fachs Informatik am HHG			
Aufgabe des Faches Informatik ist die Vermittlung einer informatischen Grundbildung. Das Fach Informatik nimmt einen wichtigen Stellenwert in der Entwicklung der Schülerinnen und Schüler im Hinblick auf die Herausforderung einer digitalisierten Welt. Um junge Menschen auf ein selbstbestimmtes Leben in einer durch Digitalisierung geprägten Gesellschaft vorzubereiten und deren Teilhabe zu gewährleisten, bedarf es einer informatischen Grundbildung als wichtigen Bestandteil der Allgemeinbildung. Ausgangspunkt des Informatikunterrichts der Klassen 5 und 6 am HHG sind Fragestellungen mit lebensweltlichem Bezug. Gleichzeitig wird im besonderen Maße Wert daraufgelegt, dass die Schülerinnen und Schüler durch motivierende, altersgerechte und selbstentdeckende Zugänge informatorische Zusammenhänge individuell kennen und verstehen lernen. Durch eine verstärkte Zusammenarbeit und Koordinierung der Fachbereiche werden Bezüge zwischen den Inhalten der Fächer hergestellt. Die Fachkonferenz Informatik hat beschlossen, dass im Rahmen des Unterrichtsvorhabens „Was ist Informatik?“ eine aktuelle Präsentation in Zusammenarbeit mit einem anderen Fach erstellt wird.			

Inhalte und Kompetenzerwartungen am Ende der Erprobungsstufe

Unterrichtsvorhaben Klasse 5

Unterrichtsvorhaben 5.1a: Was ist Informatik?

Inhaltliche Schwerpunkte

- Informatik – ein neues Fach
- Daten, Information, Wissen
- Informatik im Alltag & Beruf (Erstellung einer PowerPoint-Präsentation)

Inhaltsfelder (IF) nach KLP Informatik

IF: Information und Daten

- Informationsgehalt von Daten

IF: Informatiksysteme

- Anwendung von Informatiksystemen

IF: Informatik, Mensch und Gesellschaft

- Informatiksysteme in der Lebens- und Arbeitswelt

Übergeordnete Kompetenzerwartungen nach KLP Informatik

Argumentieren (A)

Schülerinnen und Schüler ...

- 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
- begründen die Auswahl eines Informatiksystems

Darstellen und Interpretieren (DI)

Schülerinnen und Schüler ...

- beschreiben einfache Darstellungen von informatischen Sachverhalten
- stellen informatische Sachverhalte in geeigneter Form dar

Kommunizieren und Kooperieren (KK)

Schülerinnen und Schüler ...

- beschreiben einfache informatische Sachverhalte unter Verwendung von Fachbegriffen sachgerecht
- kooperieren in verschiedenen Formen der Zusammenarbeit bei der Bearbeitung einfacher informatischer Probleme

Konkretisierte Kompetenzerwartungen nach KLP Informatik

Die Schülerinnen und Schüler ...

- erläutern den Datenbegriff anhand von Beispielen aus ihrer Erfahrungswelt (A)
- erläutern den Zusammenhang und die Bedeutung von Information und Daten (A)
- interpretieren ausgewählte Daten als Information im gegebenen Kontext (DI)
- beschreiben an Beispielen die Bedeutung von Informatiksystemen in der Lebens- und Arbeitswelt (KK)

benennen an ausgewählten Beispielen Auswirkungen des Einsatzes von Informatiksystemen auf ihre Lebens- und Erfahrungswelt (A/KK)

Unterrichtsvorhaben 5.1b: Umgang mit Informatiksystemen

Inhaltliche Schwerpunkte

- Regeln im Umgang mit Computern in Schule und Alltag
- Schulnetzwerk
- Hardware & Software (EVA-Prinzip)
- Dateien verwenden & Ordnerstrukturen
- Avatare

Inhaltsfelder (IF) nach KLP Informatik

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

Übergeordnete Kompetenzerwartungen nach KLP Informatik

Argumentieren (A)

Schülerinnen und Schüler ...

- äußern Vermutungen zu informatischen Sachverhalten auf der Basis von Alltagsvorstellungen oder Vorwissen
- begründen die Auswahl eines Informatiksystems

Kommunizieren und Kooperieren (KK)

Schülerinnen und Schüler ...

- beschreiben einfache informatische Sachverhalte unter Verwendung von Fachbegriffen sachgerecht
- kooperieren in verschiedenen Formen der Zusammenarbeit bei der Bearbeitung einfacher informatischer Probleme
- dokumentieren gemeinsam ihren Arbeitsprozess und ihre Ergebnisse auch mithilfe digitaler Werkzeuge
- setzen bei der Bearbeitung einer informatischen Problemstellung geeignete digitale Werkzeuge zum kollaborativen Arbeiten ein

Konkretisierte Kompetenzerwartungen nach KLP Informatik

Die Schülerinnen und Schüler ...

- interpretieren ausgewählte Daten als Information im gegebenen Kontext (DI)
- erläutern Einheiten von Datenmengen (A/KK)
- vergleichen Möglichkeiten der Datenverwaltung hinsichtlich ihrer spezifischen Charakteristika (u. a. Speicherort, Kapazität, Aspekte der Datensicherheit) (A)
- benennen Grundkomponenten von (vernetzten) Informatiksystemen und beschreiben ihre Funktionen (DI)
- benennen Beispiele für (vernetzte) Informatiksysteme aus ihrer Erfahrungswelt (DI)
- beschreiben das Prinzip der Eingabe, Verarbeitung und Ausgabe (EVA-Prinzip) als grundlegendes Prinzip der Datenverarbeitung (DI)
- erläutern Prinzipien der strukturierten Dateiverwaltung (A)
- setzen Informatiksysteme zur Kommunikation und Kooperation ein (KK)
- benennen an ausgewählten Beispielen Auswirkungen des Einsatzes von Informatiksystemen auf ihre Lebens- und Erfahrungswelt (A/KK)

Unterrichtsvorhaben 5.2: Von der Anwendung zum Algorithmus (Algorithmen I)

Inhaltliche Schwerpunkte

- Handlungen durch Ablaufpläne beschreiben
- Entscheidungen in alltäglichen Situationen (Ablaufpläne)
- Experimentieren mit den Ozobots
- Programmieren mit den Ozobots
- Algorithmen untersuchen & entwickeln
- Programmieren mit der Maus

Inhaltsfelder (IF) nach KLP Informatik

IF: Algorithmen

- Algorithmen und algorithmische Grundkonzepte

Übergeordnete Kompetenzerwartungen nach KLP Informatik

Argumentieren (A)

Schülerinnen und Schüler ...

- äußern Vermutungen zu informatischen Sachverhalten auf der Basis von Alltagsvorstellungen oder Vorwissen
- bewerten ein Ergebnis einer informatischen Modellierung

Modellieren und Implementieren (MI)

Schülerinnen und Schüler ...

- erstellen informatische Modelle zu gegebenen Sachverhalten
- überprüfen Modelle und Implementierungen

Darstellen und Interpretieren (DI)

Schülerinnen und Schüler ...

- beschreiben einfache Darstellungen von informatischen Sachverhalten
- stellen informatische Sachverhalte in geeigneter Form dar
- interpretieren informatische Darstellungen

Kommunizieren und Kooperieren (KK)

Schülerinnen und Schüler ...

- beschreiben einfache 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

Konkretisierte Kompetenzerwartungen nach KLP Informatik

Die Schülerinnen und Schüler ...

- formulieren zu Abläufen aus dem Alltag eindeutige Handlungsvorschriften (DI)
- führen Handlungsvorschriften schrittweise aus (MI)
- identifizieren in Handlungsvorschriften Anweisungen und die algorithmischen Grundstrukturen Sequenz, Verzweigung und Schleife (MI)
- überprüfen die Wirkungsweise eines Algorithmus durch zielgerichtetes Testen (MI)
- implementieren Algorithmen in einer visuellen Programmiersprache (MI)
- ermitteln durch die Analyse eines Algorithmus dessen Ergebnis (DI)

Unterrichtsvorhaben 5.3: Codierung & Kryptographie I

Inhaltliche Schwerpunkte

- Bedeutung von Codes
- Passwortsicherheit
- Winker- & Morse-Alphabet
- Steganographie

Inhaltsfelder (IF) nach KLP Informatik

IF: Information und Daten

- Daten und ihre Codierung
- Informationsgehalt von Daten
- Verschlüsselungsverfahren

Übergeordnete Kompetenzerwartungen nach KLP Informatik

Argumentieren (A)

Schülerinnen und Schüler ...

- äußern Vermutungen zu informatischen Sachverhalten auf der Basis von Alltagsvorstellungen oder Vorwissen

Modellieren und Implementieren (MI)

Schülerinnen und Schüler ...

- erstellen informatische Modelle zu gegebenen Sachverhalten

Darstellen und Interpretieren (DI)

Schülerinnen und Schüler ...

- beschreiben einfache Darstellungen von informatischen Sachverhalten
- stellen informatische Sachverhalte in geeigneter Form dar
- interpretieren informatische Darstellungen

Kommunizieren und Kooperieren (KK)

Schülerinnen und Schüler ...

- beschreiben einfache informatische Sachverhalte unter Verwendung von Fachbegriffen sachgerecht
- kooperieren in verschiedenen Formen der Zusammenarbeit bei der Bearbeitung einfacher informatischer Probleme

Konkretisierte Kompetenzerwartungen nach KLP Informatik

Die Schülerinnen und Schüler ...

- 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)
- interpretieren ausgewählte Daten als Information im gegebenen Kontext (DI)
- beschreiben Maßnahmen zum Schutz von Daten mithilfe von Informatiksystemen (A)

Unterrichtsvorhaben 5.4: Automaten in unserer Lebenswelt

Inhaltliche Schwerpunkte

- Aufbau und Wirkungsweise einfacher Automaten
- Automaten aus der Lebenswelt der SuS
- Abläufe in Automaten graphisch darstellen

Inhaltsfelder (IF) nach KLP Informatik

IF: Automaten und künstliche Intelligenz

- Aufbau und Wirkungsweise einfacher Automaten

IF: Informatik, Mensch und Gesellschaft

- Informatiksysteme in der Lebens- und Arbeitswelt
- Datenbewusstsein
- Datensicherheit und Sicherheitsregeln

Übergeordnete Kompetenzerwartungen nach KLP Informatik

Argumentieren (A)

Schülerinnen und Schüler ...

- äußern Vermutungen zu informatischen Sachverhalten auf der Basis von Alltagsvorstellungen oder Vorwissen
- erläutern mögliche Auswirkungen des Einsatzes von Informatiksystemen

Darstellen und Interpretieren (DI)

Schülerinnen und Schüler ...

- beschreiben einfache Darstellungen von informatischen Sachverhalten
- stellen informatische Sachverhalte in geeigneter Form dar
- interpretieren informatische Darstellungen

Kommunizieren und Kooperieren (KK)

Schülerinnen und Schüler ...

- beschreiben einfache informatische Sachverhalte unter Verwendung von Fachbegriffen sachgerecht
- kooperieren in verschiedenen Formen der Zusammenarbeit bei der Bearbeitung einfacher informatischer Probleme

Konkretisierte Kompetenzerwartungen nach KLP Informatik

Die Schülerinnen und Schüler ...

- erläutern die Funktionsweise eines Automaten aus ihrer Lebenswelt (A)
- stellen Abläufe in Automaten graphisch dar (DI)
- benennen an ausgewählten Beispielen Auswirkungen des Einsatzes von Informatiksystemen auf ihre Lebens- und Erfahrungswelt (A/KK)

Unterrichtsvorhaben Klasse 6

Unterrichtsvorhaben 6.1: Codierung & Kryptographie II

Inhaltliche Schwerpunkte

- Sender-Empfänger-Modell
- Caesar-Chiffre
- Zahlen im Binärsystem darstellen
- Texte und Bilder binär codieren (ASCII)

Inhaltsfelder (IF) nach KLP Informatik

IF: Information und Daten

- Daten und ihre Codierung
- Informationsgehalt von Daten
- Verschlüsselungsverfahren

Übergeordnete Kompetenzerwartungen nach KLP Informatik

Argumentieren (A)

Schülerinnen und Schüler ...

- äußern Vermutungen zu informatischen Sachverhalten auf der Basis von Alltagsvorstellungen oder Vorwissen

Modellieren und Implementieren (MI)

Schülerinnen und Schüler ...

- erstellen informatische Modelle zu gegebenen Sachverhalten

Darstellen und Interpretieren (DI)

Schülerinnen und Schüler ...

- beschreiben einfache Darstellungen von informatischen Sachverhalten
- stellen informatische Sachverhalte in geeigneter Form dar
- interpretieren informatische Darstellungen

Kommunizieren und Kooperieren (KK)

Schülerinnen und Schüler ...

- beschreiben einfache informatische Sachverhalte unter Verwendung von Fachbegriffen sachgerecht
- kooperieren in verschiedenen Formen der Zusammenarbeit bei der Bearbeitung einfacher informatischer Probleme

Konkretisierte Kompetenzerwartungen nach KLP Informatik

Die Schülerinnen und Schüler ...

- 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 Datenmengen (A/KK)
- vergleichen Datenmengen hinsichtlich ihrer Größe mithilfe anschaulicher Beispiele aus ihrer Lebenswelt (DI)
- erläutern ein einfaches Transpositionsverfahren als Möglichkeit der Verschlüsselung (DI)
- vergleichen verschiedene Verschlüsselungsverfahren unter Berücksichtigung von ausgewählten Sicherheitsaspekten (DI)
- beschreiben Maßnahmen zum Schutz von Daten mithilfe von Informatiksystemen (A)

Unterrichtsvorhaben 6.2: Vom Modell zum Programm (Algorithmen II)

Inhaltliche Schwerpunkte

- Einstieg in Scratch
- Vom Ablaufplan zum Struktogramm
- Entscheidungen und Wiederholungen
- Eigene Programme entwickeln (Witze)

Inhaltsfelder (IF) nach KLP Informatik

IF: Algorithmen

- Algorithmen und algorithmische Grundkonzepte
- Implementation von Algorithmen

IF: Informatiksysteme

- Aufbau und Funktionsweise von Informatiksystemen

Übergeordnete Kompetenzerwartungen nach KLP Informatik

Modellieren und Implementieren (MI)

Schülerinnen und Schüler ...

- erstellen informatische Modelle zu gegebenen Sachverhalten
- implementieren informatische Modelle unter Verwendung algorithmischer Grundstrukturen
- überprüfen Modelle und Implementierungen

Darstellen und Interpretieren (DI)

Schülerinnen und Schüler ...

- beschreiben einfache Darstellungen von informatischen Sachverhalten
- stellen informatische Sachverhalte in geeigneter Form dar
- interpretieren informatische Darstellungen

Kommunizieren und Kooperieren (KK)

Schülerinnen und Schüler ...

- beschreiben einfache 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
- dokumentieren gemeinsam ihren Arbeitsprozess und ihre Ergebnisse auch mithilfe digitaler Werkzeuge

Konkretisierte Kompetenzerwartungen nach KLP Informatik

Die Schülerinnen und Schüler ...

- überführen Handlungsvorschriften in einen Programmablaufplan (PAP) oder ein Struktogramm (MI)
- 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 zielgerichtetes Testen (MI)
- ermitteln durch die Analyse eines Algorithmus dessen Ergebnis (DI)
- bewerten einen als Quelltext, Programmablaufplan (PAP) oder Struktogramm dargestellten Algorithmus hinsichtlich seiner Funktionalität
- setzen zielgerichtet Informatiksysteme zur Verarbeitung von Daten ein (MI)

Unterrichtsvorhaben 6.3: Maschinelles Lernen und künstliche Intelligenz

Inhaltliche Schwerpunkte

- Alltag mit künstlicher Intelligenz
- Maschinelles Lernen

Inhaltsfelder (IF) nach KLP Informatik

IF: Automaten und künstliche Intelligenz

- Maschinelles Lernen mit Entscheidungsbäumen
- Maschinelles Lernen mit neuronalen Netzen

IF: Informatik, Mensch und Gesellschaft

- Informatiksysteme in der Lebens- und Arbeitswelt
- Datenbewusstsein
- Datensicherheit und Sicherheitsregeln

Übergeordnete Kompetenzerwartungen nach KLP Informatik

Argumentieren (A)

Schülerinnen und Schüler ...

- äußern Vermutungen zu informatischen Sachverhalten auf der Basis von Alltagsvorstellungen oder Vorwissen
- erläutern mögliche Auswirkungen des Einsatzes von Informatiksystemen

Darstellen und Interpretieren (DI)

Schülerinnen und Schüler ...

- beschreiben einfache Darstellungen von informatischen Sachverhalten
- stellen informatische Sachverhalte in geeigneter Form dar
- interpretieren informatische Darstellungen

Kommunizieren und Kooperieren (KK)

Schülerinnen und Schüler ...

- beschreiben einfache informatische Sachverhalte unter Verwendung von Fachbegriffen sachgerecht
- kooperieren in verschiedenen Formen der Zusammenarbeit bei der Bearbeitung einfacher informatischer Probleme

Konkretisierte Kompetenzerwartungen nach KLP Informatik

Die Schülerinnen und Schüler ...

- 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)
- benennen an ausgewählten Beispielen Auswirkungen des Einsatzes von Informatiksystemen auf ihre Lebens- und Erfahrungswelt (A/KK)

Unterrichtsvorhaben 6.4: Experimentieren mit Mikrocontrollern

Inhaltliche Schwerpunkte

- Der Calliope und seine Bestandteile
- Die Programmierumgebung Open Roberta Lab
- Eingabe und Ausgabe mit Calliope
- Sensorik mit dem Calliope und Calli:bot

Inhaltsfelder (IF) nach KLP Informatik

IF: Algorithmen

- Algorithmen und algorithmische Grundkonzepte
- Implementation von Algorithmen

IF: Informatiksysteme

- Aufbau und Funktionsweise von Informatiksystemen
- Anwendung von Informatiksystemen

IF: Informatik, Mensch und Gesellschaft

- Informatiksysteme in der Lebens- und Arbeitswelt
- Datenbewusstsein
- Datensicherheit und Sicherheitsregeln

Übergeordnete Kompetenzerwartungen nach KLP Informatik

Modellieren und Implementieren (MI)

Schülerinnen und Schüler ...

- erstellen informatische Modelle zu gegebenen Sachverhalten
- implementieren informatische Modelle unter Verwendung algorithmischer Grundstrukturen
- überprüfen Modelle und Implementierungen

Darstellen und Interpretieren (DI)

Schülerinnen und Schüler ...

- beschreiben einfache Darstellungen von informatischen Sachverhalten
- stellen informatische Sachverhalte in geeigneter Form dar
- interpretieren informatische Darstellungen

Kommunizieren und Kooperieren (KK)

Schülerinnen und Schüler ...

- beschreiben einfache 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
- dokumentieren gemeinsam ihren Arbeitsprozess und ihre Ergebnisse auch mithilfe digitaler Werkzeuge

Konkretisierte Kompetenzerwartungen nach KLP Informatik

Die Schülerinnen und Schüler ...

- überführen Handlungsvorschriften in einen Programmablaufplan (PAP) oder ein Struktogramm (MI)
- 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 zielgerichtetes Testen (MI)
- ermitteln durch die Analyse eines Algorithmus dessen Ergebnis (DI)
- bewerten einen als Quelltext, Programmablaufplan (PAP) oder Struktogramm dargestellten Algorithmus hinsichtlich seiner Funktionalität
- setzen zielgerichtet Informatiksysteme zur Verarbeitung von Daten ein (MI)
- setzen Informatiksysteme zur Kommunikation und Kooperation ein (KK)
- beschreiben Maßnahmen zum Schutz von Daten mithilfe von Informatiksystemen (A)
- benennen Grundkomponenten von (vernetzten) Informatiksystemen und beschreiben ihre Funktionen (DI)

Unterrichtsvorhaben 6.5: Datenbewusstsein

Inhaltliche Schwerpunkte

- Umgang mit eigenen und fremden Daten
- Speicherorte (am Beispiel OneDrive)
- Maßnahmen zum Schutz von Daten

Inhaltsfelder (IF) nach KLP Informatik

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
- Datensicherheit und Sicherheitsregeln

Übergeordnete Kompetenzerwartungen nach KLP Informatik

Argumentieren (A)

Schülerinnen und Schüler ...

- äußern Vermutungen zu informatischen Sachverhalten auf der Basis von Alltagsvorstellungen oder Vorwissen
- erläutern mögliche Auswirkungen des Einsatzes von Informatiksystemen

Darstellen und Interpretieren (DI)

Schülerinnen und Schüler ...

- beschreiben einfache Darstellungen von informatischen Sachverhalten
- stellen informatische Sachverhalte in geeigneter Form dar
- interpretieren informatische Darstellungen

Kommunizieren und Kooperieren (KK)

Schülerinnen und Schüler ...

- kooperieren in verschiedenen Formen der Zusammenarbeit bei der Bearbeitung einfacher informatischer Probleme
- dokumentieren gemeinsam ihren Arbeitsprozess und ihre Ergebnisse auch mithilfe digitaler Werkzeuge
- setzen bei der Bearbeitung einer informatischen Problemstellung geeignete digitale Werkzeuge zum kollaborativen Arbeiten ein

Konkretisierte Kompetenzerwartungen nach KLP Informatik

Die Schülerinnen und Schüler ...

- stellen eine ausgewählte Information in geeigneter Form als Daten formalsprachlich oder graphisch dar (DI)
- interpretieren ausgewählte Daten als Information im gegebenen Kontext (DI)
- beschreiben das Prinzip der Eingabe, Verarbeitung und Ausgabe (EVA-Prinzip) als grundlegendes Prinzip der Datenverarbeitung (DI)
- 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)
- setzen Informatiksysteme zur Kommunikation und Kooperation ein (KK)
- beschreiben an Beispielen die Bedeutung von Informatiksystemen in der Lebens- und Arbeitswelt (KK)

- erläutern an ausgewählten Beispielen Auswirkungen des Einsatzes von Informatiksystemen (A/KK)
- 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)

Leistungsbewertung

Der Beurteilungsbereich „Sonstige Leistungen im Unterricht“ erfasst die im Unterrichtsgeschehen durch mündliche, schriftliche und praktische Beiträge erkennbare Kompetenzentwicklung der Schülerinnen und Schüler. Bei der Bewertung berücksichtigt werden die Qualität, die Quantität und die Kontinuität der Beiträge. Die Kompetenzentwicklung im Beurteilungsbereich „Sonstige Leistungen im Unterricht“ wird sowohl durch kontinuierliche Beobachtung während des Schuljahres (Prozess der Kompetenzentwicklung) als auch durch punktuelle Überprüfungen (Stand der Kompetenzentwicklung) festgestellt. Bei der Bewertung von Leistungen, die die Schülerinnen und Schüler im Rahmen von Partner- oder Gruppenarbeiten erbringen, kann der individuelle Beitrag zum Ergebnis der Partner- bzw. Gruppenarbeit einbezogen werden.

Mögliche Überprüfungsformen:

- Darstellungs- und Dokumentationsaufgaben
 - Beschreibung informatorischer Zusammenhänge
 - Auswertung von Sensoren
- Modellierungs- und Implementationsaufgaben
 - Erstellung von Quellcodes und Algorithmen
 - Fehlersuche und -korrektur in Algorithmen und Quellcodes
 - Ablaufpläne und Struktogramme
- Präsentationsaufgaben
 - Programme vorstellen und erläutern
 - Kurzvorträge und Präsentationen
- Begründungs- und Bewertungsaufgaben
 - Begründung der eigenen Vorgehensweise
 - Stellungnahme zu Texten und Medienbeiträgen

Weitere Hinweise, Vereinbarungen und Absprachen:

- Informatik wird in Klasse 5 im ersten Halbjahr und in Klasse 6 im zweiten Halbjahr jeweils mit einer Doppelstunde pro Woche unterrichtet.
- Teilnahme Informatik-Biber (Klasse 5)
- Teilnahme Jugendwettbewerb Informatik (Klasse 6)

Bezug zum Medienkompetenzrahmen NRW (Schwerpunkte)

- | | |
|---|-----------------|
| 1. Bedienen und Anwenden | (1.1-1.4) |
| 2. Informieren und Recherchieren | (2.4) |
| 3. Kommunizieren und Kooperieren | (3.1, 3.2, 3.4) |
| 4. Produzieren und Präsentieren | (4.1, 4.4) |
| 5. Analysieren und Reflektieren | (5.4) |
| 6. Problemlösen und Modellieren | (6.1-6.4) |

Anhang: Berufsorientierung / Einsatz des iPads im Informatikunterricht

<u>Berufsorientierung</u>			
	Inhalt	Wann?	Wo?
5	Bedeutung von Informatiksystemen in der Lebens- und Arbeitswelt	5.1	Informatik: Unterrichtsvorhaben 5.1a: Was ist Informatik? Unterrichtsvorhaben 5.1b: Umgang mit Informatiksystemen
6	Umgang mit personenbezogenen Daten Bewusstes Verhalten im Internet	6.5	Informatik: Unterrichtsvorhaben 6.5: Datenbewusstsein
6	Sender-Empfänger-Modell (Kommunikation und Kommunikationsstörungen)	6.2	Informatik: Unterrichtsvorhaben 6.1: Codierung & Kryptographie II
6	Alltag mit künstlicher Intelligenz	6.2	Informatik: Unterrichtsvorhaben 6.3: Maschinelles Lernen und künstliche Intelligenz
<u>Einsatz des iPads im Informatikunterricht</u>			
<i>Vorwort:</i> <i>Aufgrund fachspezifischer Vorgaben liegt der Schwerpunkt auf dem Einsatz von Desktop-PCs. Dennoch werden die iPads im Informatikunterricht folgendermaßen verwendet:</i>			
Jahrgangsstufe 5/6: • Digitales Lehrwerk „Praxis Informatik 5/6“ → Inhalte, Lösungen, eigene Notizen, Web-Codes • PowerPoint-Projekt • Digitale Arbeitsblätter und Ordnerstruktur (OneNote)			

Schulinterner Lehrplan am Heinrich-Heine-Gymnasium

Medienbildung (MB)



Stand: September 2024

Fach:	Medienbildung	Jahrgangsstufe:	7
Aufgaben und Ziele des Fachs „Medienbildung“ am HHG			
Das Fach „Medienbildung“ (Jg. 7) soll den Grundstein für die mediale Kompetenzerweiterung als Unterstützung für sämtliche Fächer im Rahmen des Medienkonzeptes bilden. In der Jahrgangsstufe 7 wird epochal mit einem zweistündigen Umfang das Fach unterrichtet. Die Konzeption zielt darauf ab, dass die in den Jahrgangsstufe 5-6 erworbenen Fähigkeiten, die unter Einbezug sämtlicher Fächer in der Unterstufe laut dem Medienkompetenzplan NRW vermittelt werden, in der Jahrgangsstufe 7 erweitert und vertieft werden, um auf diese Weise den Herausforderungen einer zunehmend digitalisierten Welt zu begegnen. Des Weiteren soll die Vermittlung einer informatischen Grundbildung, die als Hauptaufgabe des Fachs Informatik in den Jahrgangsstufen 5-6 definiert ist, durch unterschiedliche Zugänge ergänzt werden.			

Inhalte und Zielsetzungen am Ende der Jahrgangsstufe 7

Unterrichtsvorhaben 7.1: Daten und Datenbanken

Zielsetzung

Das Themenfeld „Daten und Datenbanken“ vertieft den strukturierten Umgang mit Daten. Die Schülerinnen und Schüler erfassen und ordnen Daten in Tabellen, führen Berechnungen durch und nutzen einfache Wenn-Funktionen. Sie lernen die Bedeutung von Datenbanken für die Speicherung und Verwaltung von Informationen kennen und unterscheiden diese von der Arbeit mit einer Tabellenkalkulation.

Inhaltliche Schwerpunkte

- Mit Daten arbeiten
- Mit Daten rechnen
- Einfache Wenn-Funktionen nutzen
- Bedeutung von Datenbanken

Unterrichtsvorhaben 7.2: Textbasierte Programmierung – XLOGO

Zielsetzung

Das Themenfeld „Textbasierte Programmierung mit XLogo“ knüpft an die informatische Grundbildung der Jahrgangsstufen 5 und 6 an. Die Schülerinnen und Schüler beschreiben Algorithmen mithilfe von Struktogrammen und setzen sie in textbasierte Programme um. Wiederholungen, Unterprogramme und Variablen ermöglichen zunehmend komplexe Zeichnungen und Animationen. Projektartige Aufgaben fördern die Kreativität, das systematische Testen und die Zusammenarbeit.

Inhaltliche Schwerpunkte

- Grundlagen der Programmierung
- Programmieren mit XLogo
- Algorithmen & Struktogramme
- Wiederholungen
- Programme und Unterprogramme
- Variablen
- Animationen

Unterrichtsvorhaben 7.3: Chancen & Risiken der digitalen Welt

Zielsetzung

Im Themenfeld „Chancen & Risiken der digitalen Welt“ sollen die Schülerinnen und Schüler im Hinblick auf ihren eigenen Umgang mit digitalen Medien sensibilisiert werden. Es wird aufgezeigt, wie sich digitale Kommunikation sinnvoll einsetzen lässt und wo sie auf ihre Grenzen stößt. Dabei soll das Verhalten in sozialen Netzwerken analysiert und der Umgang mit eigenen Daten kritisch betrachtet werden. Weiterhin wird der zunehmende Einfluss von digitalen Netzwerken in allen Lebensbereichen thematisiert und hinterfragt.

Inhaltliche Schwerpunkte

- Verhalten und Umgang mit sozialen Netzwerken
- Cybermobbing
- Mediennutzung im Alltag
- Spaß und Sucht
- Urheberrecht und Datenschutz
- Digitaler Wandel durch das Internet

Unterrichtsvorhaben 7.4: Projekt Makerspace

Zielsetzung

Das Themenfeld „Projekt Makerspace“ umfasst die Planung, Gestaltung und Herstellung eigener Produkte. Die Schülerinnen und Schüler modellieren dreidimensionale Objekte und erstellen visuelle Entwürfe. Sie lernen den Umgang mit dem 3D-Drucker und dem Plotter kennen und setzen geeignete Entwürfe praktisch um. Dabei werden Kreativität, Kooperationsbereitschaft und das technische Verständnis für die eingesetzten Geräte gefördert.

Inhaltliche Schwerpunkte

- Modellierung von dreidimensionalen Objekten
- Visuelles Design
- Umgang mit dem 3D-Drucker
- Umgang mit dem Plotter

Leistungsbewertung

Der Beurteilungsbereich „Sonstige Leistungen im Unterricht“ erfasst die im Unterrichtsgeschehen durch mündliche, schriftliche und praktische Beiträge erkennbare Kompetenzentwicklung der Schülerinnen und Schüler. Bei der Bewertung berücksichtigt werden die Qualität, die Quantität und die Kontinuität der Beiträge. Die Kompetenzentwicklung im Beurteilungsbereich „Sonstige Leistungen im Unterricht“ wird sowohl durch kontinuierliche Beobachtung während des Schuljahres (Prozess der Kompetenzentwicklung) als auch durch punktuelle Überprüfungen (Stand der Kompetenzentwicklung) festgestellt. Bei der Bewertung von Leistungen, die die Schülerinnen und Schüler im Rahmen von Partner- oder Gruppenarbeiten erbringen, kann der individuelle Beitrag zum Ergebnis der Partner- bzw. Gruppenarbeit einbezogen werden.

Mögliche Überprüfungsformen:

- Darstellungs- und Dokumentationsaufgaben
 - Beschreibung informatorischer Zusammenhänge
 - Auswertung von Sensoren und Aktoren
 - Beschreibung und Erläuterung methodischer Vorgehensweisen
- Modellierungs- und Implementationsaufgaben
 - Erstellung von Quellcodes und Algorithmen
 - Fehlersuche und -korrektur in Algorithmen und Quellcodes
- Präsentationsaufgaben
 - Programme vorstellen und erläutern
 - Kurzvorträge und Präsentationen
 - Dateien (Word, Excel, PowerPoint) präsentieren
- Begründungs- und Bewertungsaufgaben
 - Begründung der eigenen Vorgehensweise
 - Stellungnahme zu Texten und Medienbeiträgen

Weitere Hinweise, Vereinbarungen und Absprachen:

- Medienbildung wird in Klasse 7 epochal wahlweise im ersten oder zweiten Halbjahr unterrichtet.
- Teilnahme Informatik-Biber (Medienbildungskurse 1. Halbjahr)
- Teilnahme Jugendwettbewerb Informatik (Medienbildungskurse 2. Halbjahr)

Bezug zum Medienkompetenzrahmen NRW (Schwerpunkte)

- | | |
|---|-----------------|
| 1. Bedienen und Anwenden | (1.1-1.4) |
| 2. Informieren und Recherchieren | (2.1-2.4) |
| 3. Kommunizieren und Kooperieren | (3.1, 3.3, 3.4) |
| 4. Produzieren und Präsentieren | (4.1-4.4) |
| 5. Analysieren und Reflektieren | (5.1, 5.3, 5.4) |
| 6. Problemlösen und Modellieren | (6.3, 6.4) |

Anhang: Berufsorientierung / Einsatz des iPads im Informatikunterricht

<u>Berufsorientierung</u>			
	Inhalt	Wann?	Wo?
7	Tabellenkalkulation und strukturierte Verwaltung von Daten	7	Medienbildung: Unterrichtsvorhaben 7.1: Daten und Datenbanken
7	Mediennutzung im Alltag	7	Medienbildung: Unterrichtsvorhaben 7.3: Chancen & Risiken der digitalen Welt
<u>Einsatz des iPads im Informatikunterricht</u>			
<i>Vorwort:</i> <i>Aufgrund fachspezifischer Vorgaben liegt der Schwerpunkt auf dem Einsatz von Desktop-PCs. Dennoch werden die iPads im Informatikunterricht folgendermaßen verwendet:</i>			
Jahrgangsstufe 7: • Office-Anwendungen, insbesondere Tabellenkalkulation • Krypto-Kids • Digitale Arbeitsblätter und Ordnerstruktur (OneNote)			

Schulinterner Lehrplan am Heinrich-Heine-Gymnasium

Informatik 9/10



Stand: September 2024

Fach:	Informatik	Jahrgangsstufe:	9/10
Aufgaben und Ziele des Fachs Informatik am HHG			
Aufgabe des Faches Informatik ist die Vermittlung einer informatischen Grundbildung. Das Fach Informatik nimmt einen wichtigen Stellenwert in der Entwicklung der Schülerinnen und Schüler im Hinblick auf die Herausforderungen einer digitalisierten Welt ein. Um junge Menschen auf ein selbstbestimmtes Leben in einer durch Digitalisierung geprägten Gesellschaft vorzubereiten und deren Teilhabe zu gewährleisten, bedarf es einer informatischen Grundbildung als wichtigen Bestandteil der Allgemeinbildung. Ausgangspunkt des Informatikunterrichts der Klassen 9 und 10 am HHG sind Fragestellungen mit lebensweltlichem Bezug. Gleichzeitig wird im besonderen Maße Wert darauf gelegt, dass die Schülerinnen und Schüler durch motivierende, altersgerechte und selbstentdeckende Zugänge informatische Zusammenhänge individuell kennen und verstehen lernen. Bei der Arbeit mit Informatiksystemen erhalten die Lernenden regelmäßig Rückmeldungen über ihren individuellen Lernfortschritt. In einem längerfristigen Entwicklungsprozess arbeitet das Fach Informatik daran, die Bedingungen für individuelles und erfolgreiches Lernen zu optimieren. Das Fach Informatik ermöglicht vertiefte Einsicht in den Aufbau und die Funktionsweise von Informatiksystemen und leistet damit einen wesentlichen Beitrag zur Bildung in einer digitalisierten Welt, der auch einen wesentlichen Punkt des Schulprogramms darstellt.			

Unterrichtsvorhaben der Jahrgangsstufe 9

Unterrichtsvorhaben 9.1: Weltweites Datennetz – HTML, CSS & Datenschutz

Schwerpunkte der Kompetenzentwicklung

- analysieren und beschreiben informatische Sachverhalte (A),
- strukturieren informatische Sachverhalte (MI),
- interpretieren unterschiedliche Darstellungen von informatischen Sachverhalten (DI),
- erläutern adressatengerecht informatische Sachverhalte (KK),
- stellen informatische Sachverhalte unter Verwendung von Fachbegriffen dar (KK),
- kooperieren im Rahmen des projektorientierten Arbeitens (KK),
- planen die Dokumentation und Präsentation ihrer Vorgehensweise und Arbeitsergebnisse eigenständig (KK).

Inhaltsfelder (IF) nach KLP Wahlpflichtfach Informatik

- Information und Daten
- Automaten und formale Sprachen
- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte

- Daten und ihre Codierung
- Erstellung und Analyse von Quelltexten
- Anwendung von Informatiksystemen
- Datenschutz und Datensicherheit

Inhaltliche Konkretisierungen

- HTML und CSS
- Gestaltung einer eigenen Internetseite
- Datenschutz & Urheberrecht
- Lizenzmodelle

Konkretisierte Kompetenzerwartungen nach KLP Wahlpflichtfach Informatik

Die Schülerinnen und Schüler ...

- interpretieren Daten aus dem Ergebnis eines Verarbeitungsprozesses (DI),
- beschreiben an ausgewählten Beispielen das Codierungsprinzip von Pixel- und Vektorgrafiken (KK),
- analysieren Quelltexte auf syntaktische Korrektheit (A/MI),
- erstellen syntaktisch korrekte Quelltexte in einer geeigneten Dokumentenbeschreibungssprache und in einer Programmiersprache (MI),
- wenden zielgerichtet Prinzipien der strukturierten Dateiverwaltung an (MI),
- bewerten verschiedene Lizenzmodelle im Hinblick auf Weiterentwicklung und Nutzung digitaler Produkte (A),
- entwickeln kriteriengeleitet Handlungsoptionen für den Umgang mit eigenen und fremden Daten (A).

Unterrichtsvorhaben 9.2: Netzwerke – Wie sie funktionieren, wozu wir sie brauchen

Schwerpunkte der Kompetenzentwicklung

- analysieren und beschreiben informatische Sachverhalte (A),
- bewerten mögliche Auswirkungen des Einsatzes von Informatiksystemen (A),
- strukturieren informatische Sachverhalte (MI),
- analysieren Modelle und Implementierungen (MI),
- implementieren informatische Modelle (MI),
- veranschaulichen informatische Sachverhalte (DI),
- erläutern adressatengerecht informatische Sachverhalte (KK),
- stellen informatische Sachverhalte unter Verwendung von Fachbegriffen dar (KK).

Inhaltsfelder (IF) nach KLP Wahlpflichtfach Informatik

- Information und Daten
- Informatiksysteme

Inhaltliche Schwerpunkte

- Erfassung, Verarbeitung und Verwaltung von Daten
- Anwendung von Informatiksystemen

Inhaltliche Konkretisierungen

- Netzwerkaufbau
- Simulation von Netzwerken mit Filius

Konkretisierte Kompetenzerwartungen nach KLP Wahlpflichtfach Informatik

Die Schülerinnen und Schüler ...

- interpretieren Daten aus dem Ergebnis eines Verarbeitungsprozesses (DI),
- identifizieren für (vernetzte) Informatiksysteme kriteriengeleitet Anwendungsbereiche in der Lebens- und Berufswelt (A),
- wenden zielgerichtet Prinzipien der strukturierten Dateiverwaltung an (MI).

Unterrichtsvorhaben 9.3: Das Spioncamp – Sichere Kommunikation mit Kryptographie
Schwerpunkte der Kompetenzentwicklung

- bewerten informatische Sachverhalte kriteriengeleitet, (A),
- entwickeln Handlungsstrategien für informatische Fragestellungen (A),
- strukturieren informatische Sachverhalte (MI),
- wenden ein informatisches Verfahren zur Lösung eines Problems an (MI).

Inhaltsfelder (IF) nach KLP Wahlpflichtfach Informatik

- Information und Daten
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte

- Daten und ihre Codierung
- Verschlüsselungsverfahren
- Erfassung, Verarbeitung und Verwaltung von Daten
- Datenschutz und Datensicherheit

Inhaltliche Konkretisierungen

- Codes und Codierung
- Binärzahlen und ASCII-Code
- Transposition (Skytale, Schablonen, Pflügen)
- Substitution (Freimaurer- und Caesar-Chiffre, Playfair, Rotoren, Vigenère-Quadrat)
- Enigma
- Häufigkeitsanalyse
- Schlüsselaustausch
- Datenschutz und Datensicherheit

Konkretisierte Kompetenzerwartungen nach KLP Wahlpflichtfach Informatik

Die Schülerinnen und Schüler ...

- verwenden Substitutionsverfahren als Möglichkeit der Verschlüsselung (MI),
- beurteilen verschiedene Verschlüsselungsverfahren unter Berücksichtigung von ausgewählten Sicherheitsaspekten (A),
- erläutern die Prinzipien der Datensicherheit (Vertraulichkeit, Integrität und Verfügbarkeit) und berücksichtigen diese beim Umgang mit Daten (A),
- entwickeln kriteriengeleitet Handlungsoptionen für den Umgang mit eigenen und fremden Daten (A).

Unterrichtsvorhaben 9.4: Überall Automaten – Vom Lichtschalter zum Marienkäfer

Schwerpunkte der Kompetenzentwicklung

- strukturieren informatische Sachverhalte (MI),
- entwickeln informatische Modelle zu gegebenen Problemstellungen (MI),
- wenden ein informatisches Verfahren zur Lösung eines Problems an (MI),
- interpretieren unterschiedliche Darstellungen von informatischen Sachverhalten (DI).

Inhaltsfelder (IF) nach KLP Wahlpflichtfach Informatik

- Automaten und formale Sprachen

Inhaltliche Schwerpunkte

- Aufbau und Wirkungsweise von Automaten

Inhaltliche Konkretisierungen

- Endliche Automaten im Alltag
- Zustandsorientierte Modellierung
- grafische Darstellung eines Automaten als Zustandsübergangsdiagramm
- Programmierumgebung „Kara“ oder „Python SpaceBug“

Konkretisierte Kompetenzerwartungen nach KLP Wahlpflichtfach Informatik

Die Schülerinnen und Schüler ...

- analysieren die Funktionsweise eines Automaten mit Hilfe eines Zustandsübergangsdiagramms (DI),
- entwickeln einen Automaten für eine konkrete Problemstellung (MI).

Unterrichtsvorhaben 9.5: Projekt: Helfer im Alltag – Sensoren und Aktoren

Schwerpunkte der Kompetenzentwicklung

- analysieren und beschreiben informatische Sachverhalte (A),
- bewerten mögliche Auswirkungen des Einsatzes von Informatiksystemen (A),
- entwickeln Handlungsstrategien für informatische Fragestellungen (A),
- entwickeln informatische Modelle zu gegebenen Problemstellungen (MI),
- implementieren informatische Modelle (MI),
- beurteilen Modelle und Implementierungen hinsichtlich der Lösung einer Problemstellung (MI),
- wenden ein informatisches Verfahren zur Lösung eines Problems an (MI),
- identifizieren informatische Sachverhalte in komplexen Anwendungsbereichen (DI),
- veranschaulichen informatische Sachverhalte (DI),
- erläutern adressatengerecht informatische Sachverhalte (KK),
- stellen informatische Sachverhalte unter Verwendung von Fachbegriffen dar (KK),
- kooperieren im Rahmen des projektorientierten Arbeitens (KK),
- planen die Dokumentation und Präsentation ihrer Vorgehensweise und Arbeitsergebnisse eigenständig (KK).

Inhaltsfelder (IF) nach KLP Wahlpflichtfach Informatik

- Algorithmen
- Automaten und formale Sprachen
- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte

- Algorithmen und algorithmische Grundkonzepte
- Variablen
- Implementation von Algorithmen
- Erstellung und Analyse von Quelltexten
- Anwendung von Informatiksystemen
- Informatiksysteme in der Lebens- und Berufswelt

Inhaltliche Konkretisierungen

- Algorithmen
- Struktogramme
- Blockbasierte vs. Textbasierte Programmierung
- Aufbau von Steuerungen mithilfe von Aktoren und Sensoren
- Einsatzbereiche von Robotern
- Einsatz von Lego EV3 bzw. Spike Prime
- Aufbau und Funktion von Robotern
- Programmierung von Robotermodellen mit „Open Roberta Lab“
- Entwicklung von Algorithmen zur Lösung von Aufgaben
- Projektorientierte Gruppenarbeit

Konkretisierte Kompetenzerwartungen nach KLP Wahlpflichtfach Informatik

Die Schülerinnen und Schüler ...

- stellen Algorithmen in verschiedenen Repräsentationen dar (DI),
- entwerfen und implementieren Algorithmen unter Verwendung von Variablen verschiedener Typen und unter Berücksichtigung des Prinzips der Modularisierung (MI),

- kommentieren, modifizieren und ergänzen Quelltexte von Programmen nach Vorgaben (MI), erläutern die Möglichkeit der Werteübergabe mithilfe von Parametern (MI),
- überprüfen die Wirkungsweise eines Algorithmus durch zielgerichtetes Testen bei der Lösung gleichartiger Probleme (MI),
- beurteilen die Problemangemessenheit verwendeter Algorithmen (MI),
- analysieren Quelltexte auf syntaktische Korrektheit (A/MI),
- erstellen syntaktisch korrekte Quelltexte in einer geeigneten Dokumentenbeschreibungssprache und in einer Programmiersprache (MI),
- identifizieren für (vernetzte) Informatiksysteme kriteriengeleitet Anwendungsbereiche in der Lebens- und Berufswelt (A).

Optionale und ergänzende Unterrichtsvorhaben

- Teilnahme an Wettbewerben (z.B. Lego-Wettbewerbe)
- Wiederholung und Vertiefung der textbasierten Programmierung am Bsp. XLOGO

Unterrichtsvorhaben der Jahrgangsstufe 10

Unterrichtsvorhaben 10.1: Wie ein Computer denkt

Schwerpunkte der Kompetenzentwicklung

- analysieren und beschreiben informatische Sachverhalte (A),
- bewerten mögliche Auswirkungen des Einsatzes von Informatiksystemen (A),
- strukturieren informatische Sachverhalte (MI),
- analysieren Modelle und Implementierungen (MI),
- entwickeln informatische Modelle zu gegebenen Problemstellungen (MI),
- beurteilen Modelle und Implementierungen hinsichtlich der Lösung einer Problemstellung (MI),
- veranschaulichen informatische Sachverhalte (DI),
- interpretieren Ergebnisse von Implementierungen (DI),
- stellen informatische Sachverhalte unter Verwendung von Fachbegriffen dar (KK).

Inhaltsfelder (IF) nach KLP Wahlpflichtfach Informatik

- Information und Daten
- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte

- Erfassung, Verarbeitung und Verwaltung von Daten
- Anwendung von Informatiksystemen
- Logische Schaltungen
- Informatiksysteme in der Lebens- und Berufswelt

Inhaltliche Konkretisierungen

- EVA-Prinzip und Zuordnung der Hardware-Komponenten
- Aussagenlogik und Wahrheitstabellen
- Zahldarstellungen und Grundrechenarten in n-adischen Systemen
- Simulation von logischen Schaltungen mit Logic Simulator
- Einsatz ausgewählter Informatiksysteme in der Berufswelt

Konkretisierte Kompetenzerwartungen nach KLP Wahlpflichtfach Informatik

Die Schülerinnen und Schüler ...

- interpretieren Daten aus dem Ergebnis eines Verarbeitungsprozesses (DI),
- identifizieren für (vernetzte) Informatiksysteme kriteriengeleitet Anwendungsbereiche in der Lebens- und Berufswelt (A),
- erstellen und simulieren logische Schaltungen mithilfe digitaler Werkzeuge (MI),
- bewerten eine logische Schaltung hinsichtlich ihrer Funktionalität (A),
- diskutieren Auswirkungen des Einsatzes von Informatiksystemen an ausgewählten Beispielen aus der Berufswelt (A/KK).

Unterrichtsvorhaben 10.2: Imperative Programmierung mit Python

Schwerpunkte der Kompetenzentwicklung

- bewerten informatische Sachverhalte kriteriengeleitet (A),
- strukturieren informatische Sachverhalte (MI),
- analysieren Modelle und Implementierungen (MI),
- entwickeln informatische Modelle zu gegebenen Problemstellungen (MI),
- implementieren informatische Modelle (MI),
- beurteilen Modelle und Implementierungen hinsichtlich der Lösung einer Problemstellung (MI),
- wenden ein informatisches Verfahren zur Lösung eines Problems an (MI),
- identifizieren informatische Sachverhalte in komplexen Anwendungsbereichen (DI),
- veranschaulichen informatische Sachverhalte (DI),
- interpretieren Ergebnisse von Implementierungen (DI),
- interpretieren unterschiedliche Darstellungen von informatischen Sachverhalten (DI),
- stellen informatische Sachverhalte unter Verwendung von Fachbegriffen dar (KK).

Inhaltsfelder (IF) nach KLP Wahlpflichtfach Informatik

- Information und Daten
- Algorithmen
- Automaten und formale Sprachen
- Informatiksysteme

Inhaltliche Schwerpunkte

- Erfassung, Verarbeitung und Verwaltung von Daten
- Algorithmen und algorithmische Grundkonzepte
- Variablen
- Implementation von Algorithmen
- Erstellung und Analyse von Quelltexten
- Anwendung von Informatiksystemen

Inhaltliche Konkretisierungen

- Grundbefehle
- Mit Python schreiben und rechnen
- Animationen
- Modulare Entwürfe
- Parameter & Variablen
- Verzweigungen und bedingte Schleifen
- Listen und Datenbanken in TigerJython (im Vergleich zu Excel)
- Relationale Datenbanken

Konkretisierte Kompetenzerwartungen nach KLP Wahlpflichtfach Informatik

Die Schülerinnen und Schüler ...

- verarbeiten Daten mit einer Programmiersprache unter Berücksichtigung logischer und arithmetischer Operationen (MI),
- wählen geeignete Datentypen im Kontext eines Anwendungsbeispiels aus (MI),
- interpretieren Daten aus dem Ergebnis eines Verarbeitungsprozesses (DI),
- überprüfen algorithmische Eigenschaften (Endlichkeit der Beschreibung, Eindeutigkeit, Terminierung) in Handlungsvorschriften (A),
- stellen Algorithmen in verschiedenen Repräsentationen dar (DI),

- entwerfen und implementieren Algorithmen unter Verwendung von Variablen verschiedener Typen und unter Berücksichtigung des Prinzips der Modularisierung (MI),
- kommentieren, modifizieren und ergänzen Quelltexte von Programmen nach Vorgaben (MI),
- erläutern die Möglichkeit der Werteübergabe mithilfe von Parametern (MI),
- überprüfen die Wirkungsweise eines Algorithmus durch zielgerichtetes Testen bei der Lösung gleichartiger Probleme (MI),
- beurteilen die Problemangemessenheit verwendeter Algorithmen (MI),
- erläutern die Begriffe Syntax und Semantik einer Programmiersprache an Beispielen (KK),
- analysieren Quelltexte auf syntaktische Korrektheit (A/MI),
- erstellen syntaktisch korrekte Quelltexte in einer geeigneten Dokumentenbeschreibungssprache und in einer Programmiersprache (MI),
- wenden zielgerichtet Prinzipien der strukturierten Dateiverwaltung an (MI).

Unterrichtsvorhaben 10.3: Künstliche Intelligenz & Maschinelles Lernen

Schwerpunkte der Kompetenzentwicklung

- analysieren und beschreiben informatische Sachverhalte (A),
- bewerten mögliche Auswirkungen des Einsatzes von Informatiksystemen (A),
- entwickeln Handlungsstrategien für informatische Fragestellungen (A),
- erläutern adressatengerecht informatische Sachverhalte (KK),
- stellen informatische Sachverhalte unter Verwendung von Fachbegriffen dar (KK).

Inhaltsfelder (IF) nach KLP Wahlpflichtfach Informatik

- Information und Daten
- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte

- überwachtes Lernen
- unüberwachtes Lernen
- bestärkendes Lernen

Inhaltliche Konkretisierungen

- Wiederholung Automaten
- Künstliche Intelligenz in Alltag und Gesellschaft
- „Internet of Things“
- Starke und schwache KI
- Maschinelles Lernen mit Entscheidungsbäumen
- Formen des Maschinellen Lernens
- Neuronale Netze

Konkretisierte Kompetenzerwartungen nach KLP Wahlpflichtfach Informatik

Die Schülerinnen und Schüler ...

- beschreiben Anwendungsbeispiele künstlicher Intelligenz zum Überwachen, unüberwachten und bestärkenden Lernen (KK),
- beschreiben die grundlegende Funktionsweise maschinellen Lernens (überwacht, unüberwacht, bestärkend) in verschiedenen Anwendungsbeispielen (KK), (MKR 6.1)
- ordnen begründet die Methoden des maschinellen Lernens (überwachtes Lernen, unüberwachtes, bestärkendes Lernen) verschiedenen Anwendungsbeispielen zu (A),
- analysieren den Einfluss von Trainingsdaten auf die Ergebnisse eines Verfahrens maschinellen Lernens (A). (MKR 6.4)

Unterrichtsvorhaben 10.4: Informatik kreativ (vertiefendes Projekt)

Schwerpunkte der Kompetenzentwicklung, Inhaltsfelder und inhaltliche Schwerpunkte sind projektabhängig. Beispiele für Projekte

- Spieleprogrammierung in Python
- Calliope und Calli:Bot
- Planung und Durchführung eines Programmierprojektes
- Programmierung Wonder Workshop Cue mit Python
- Programmierung Spike Prime mit Python
- Planung und Gestaltung eines umfangreichen 3D-Druck-Projekts
- Apps schreiben mit dem AppInventor
- Programmieren mit Canvas

Optionale und ergänzende Unterrichtsvorhaben

- Teilnahme an Wettbewerben (z.B. Lego-Wettbewerbe)

Grundsätze der fachdidaktischen und fachmethodischen Arbeit

I. Überfachliche Grundsätze

- Schülerinnen und Schüler werden in dem Prozess unterstützt, selbstständige, eigenverantwortliche, selbstbewusste, sozial kompetente und engagierte Persönlichkeiten zu werden.
- Der Unterricht nimmt insbesondere zu Beginn des Wahlpflichtunterrichts Rücksicht auf die unterschiedlichen Voraussetzungen der Schülerinnen und Schüler.
- Geeignete Problemstellungen bestimmen die Struktur der Lernprozesse.
- Die Unterrichtsgestaltung ist grundsätzlich kompetenzorientiert angelegt.
- Der Unterricht vermittelt einen kompetenten Umgang mit Medien. Dies betrifft sowohl die private Mediennutzung als auch die Verwendung verschiedener Medien zur Präsentation von Arbeitsergebnissen.
- Der Unterricht fördert das selbstständige Lernen und Finden individueller Lösungswege sowie die Kooperationsfähigkeit der Schülerinnen und Schüler.
- Die Schülerinnen und Schüler werden in die Planung der Unterrichtsgestaltung einbezogen.
- Der Unterricht wird gemeinsam mit den Schülerinnen und Schülern evaluiert.
- Die Schülerinnen und Schüler erfahren regelmäßige, kriterienorientierte Rückmeldungen zu ihren Leistungen.
- In verschiedenen Unterrichtsvorhaben werden fächerübergreifende Aspekte berücksichtigt.

II. Fachliche Grundsätze

- berufliche Orientierung der Schülerinnen und Schüler im mathematisch-naturwissenschaftlich-technischen Aufgabenfeld,
- Erkenntnisse und geschulte Wahrnehmung unserer digitalisierten, globalisierten Welt und ihre Bedeutung in allen Bereichen des Lebens,
- selbstständiges informatisches Problemlösen,
- Fähigkeiten zur kritischen und verantwortungsvollen Analyse, Modellierung und Implementierung ausgewählter Informatiksysteme,
- altersgemäße Auseinandersetzung mit der Gestaltung und der Sicherheit von Informatiksystemen sowie den Folgen und Wirkungen ihres Einsatzes für die Gesellschaft,
- reflektierte Auseinandersetzung mit textgenerierenden Systemen, die auf künstlicher Intelligenz beruhen

Leistungsbewertung

I. Beurteilungsbereich schriftliche Leistungen/Klassenarbeiten

Schriftliche Arbeiten (Klassenarbeiten oder Projektarbeiten inkl. Dokumentation) dienen der Überprüfung der Lernergebnisse einer vorausgegangenen Unterrichtsreihe. Sie werden so angelegt, dass Sachkenntnisse und methodische Fertigkeiten nachgewiesen werden können.

Im Umfang und Anforderungsniveau sind schriftliche Arbeiten abhängig von den kontinuierlich ansteigenden Anforderungen entsprechend dem Lehrplan.

Die Anzahl und Dauer der schriftlichen Arbeiten im Fach Informatik hat die Fachkonferenz im Rahmen der Vorgaben der APO-SI für den Wahlpflichtbereich wie folgt festgelegt:

Jahgangsstufe	Arbeiten pro Schuljahr	Dauer in Unterrichtsstunden
9	4	1–1,5
10	4	1–2

Grundsätzlich ist es möglich und angestrebt pro Schuljahr eine Projektarbeit als schriftliche Arbeit zu werten. Projektarbeiten können auch auf mehrere Unterrichtsstunden verteilt angefertigt werden. Grundlage der Projektbewertung ist die Dokumentation der Projektarbeit. Vorgaben hierzu werden je nach gestellter Arbeit den Schülerinnen und Schülern mitgeteilt. Die Arbeiten werden mithilfe eines Punkterasters bewertet. Die Notengebung orientiert sich an folgendem Schema:

Note	Punkteanteil
ungenügend	0–19 %
mangelhaft	20–49 %
ausreichend	50–62 %
befriedigend	63–74 %
gut	75–87 %
sehr gut	88–100 %

II. Beurteilungsbereich „Sonstige Leistungen“

Der Beurteilungsbereich „Sonstige Leistungen im Unterricht“ erfasst die im Unterrichtsgeschehen durch mündliche, schriftliche und praktische Beiträge erkennbare Kompetenzentwicklung der Schülerinnen und Schüler. Bei der Bewertung berücksichtigt werden die Qualität, die Quantität und die Kontinuität der Beiträge. Die Kompetenzentwicklung im Beurteilungsbereich „Sonstige Leistungen im Unterricht“ wird sowohl durch kontinuierliche Beobachtung während des Schuljahres (Prozess der Kompetenzentwicklung) als auch durch punktuelle Überprüfungen (Stand der Kompetenzentwicklung) festgestellt. Bei der Bewertung von Leistungen, die die Schülerinnen und Schüler im Rahmen von Partner- oder

Gruppenarbeiten erbringen, kann der individuelle Beitrag zum Ergebnis der Partner- bzw. Gruppenarbeit einbezogen werden.

Mögliche Überprüfungsformen

- Darstellungs- und Dokumentationsaufgaben
- Beschreibung informatischer Zusammenhänge
- Auswertung von Sensoren
- Modellierungs- und Implementationsaufgaben
- Erstellung von Quellcodes und Algorithmen
- Fehlersuche und -korrektur in Algorithmen und Quellcodes
- Ablaufpläne und Struktogramme
- Präsentationsaufgaben
- Programme vorstellen und erläutern
- Kurzvorträge und Präsentationen
- Begründungs- und Bewertungsaufgaben
- Begründung der eigenen Vorgehensweise
- Stellungnahme zu Texten und Medienbeiträgen

Weitere Hinweise, Vereinbarungen und Absprachen

- verpflichtende Teilnahme am Informatik-Biber (Klasse 9 & 10) im ersten Halbjahr,
- verpflichtende Teilnahme am Jugendwettbewerb Informatik (Klasse 9 & 10) im zweiten Halbjahr

Bezug zum Medienkompetenzrahmen NRW

Übergeordnete Kompetenzerwartungen bis zum Ende von Jahrgangsstufe 10

Schülerinnen und Schüler

- kooperieren im Rahmen des projektorientierten Arbeitens (KK). (MKR 3.1)

Konkretisierte Kompetenzerwartungen bis zum Ende von Jahrgangsstufe 10

Schülerinnen und Schüler

- beurteilen verschiedene Verschlüsselungsverfahren unter Berücksichtigung von ausgewählten Sicherheitsaspekten (A), (MKR 1.4)
- überprüfen algorithmische Eigenschaften (Endlichkeit der Beschreibung, Eindeutigkeit, Terminierung) in Handlungsvorschriften (A), (MKR 6.1)
- stellen Algorithmen in verschiedenen Repräsentationen dar (DI), (MKR 6.3)
- entwerfen und implementieren Algorithmen unter Verwendung von Variablen verschiedener Typen und unter Berücksichtigung des Prinzips der Modularisierung (MI), (MKR 6.1, 6.2, 6.3)
- kommentieren, modifizieren und ergänzen Quelltexte von Programmen nach Vorgaben (MI), (MKR 6.3)
- erläutern die Möglichkeit der Werteübergabe mithilfe von Parametern (MI), (MKR 6.1)
- überprüfen die Wirkungsweise eines Algorithmus durch zielgerichtetes Testen bei der Lösung gleichartiger Probleme (MI), (MKR 6.2, 6.4)
- beurteilen die Problemangemessenheit verwendeter Algorithmen (MI), (MKR 6.4)
- analysieren Quelltexte auf syntaktische Korrektheit (A/MI), (MKR 6.3)
- erstellen syntaktisch korrekte Quelltexte in einer geeigneten Dokumentenbeschreibungssprache und in einer Programmiersprache (MI), (MKR 6.3)
- analysieren die Funktionsweise eines Automaten mit Hilfe eines Zustandsübergangsdiagramms (DI), (MKR 6.3)
- entwickeln einen Automaten für eine konkrete Problemstellung (MI), (MKR 6.3)
- beschreiben die grundlegende Funktionsweise maschinellen Lernens (überwacht, unüberwacht, bestärkend) in verschiedenen Anwendungsbeispielen (KK), (MKR 6.1)
- analysieren den Einfluss von Trainingsdaten auf die Ergebnisse eines Verfahrens maschinellen Lernens (A), (MKR 6.4)
- wenden zielgerichtet Prinzipien der strukturierten Dateiverwaltung an (MI), (MKR 1.3)
- bewerten verschiedene Lizenzmodelle im Hinblick auf Weiterentwicklung und Nutzung digitaler Produkte (A), (MKR 4.4)
- erläutern die Prinzipien der Datensicherheit (Vertraulichkeit, Integrität und Verfügbarkeit) und berücksichtigen diese beim Umgang mit Daten (A), (MKR 1.4)
- entwickeln kriteriengeleitet Handlungsoptionen für den Umgang mit eigenen und fremden Daten (A). (MKR 1.3, 1.4)

Schulinterner Lehrplan am Heinrich-Heine-Gymnasium

Informatik Oberstufe



Stand: September 2025

Fach:	Informatik	Jahrgangsstufe:	Sek. II
Aufgaben und Ziele des Fachs Informatik am HHG			
Grundlegendes Wissen der informatischen Bildung nimmt in der heutigen Zeit einen immer größeren Stellenwert ein und bildet damit einen bedeutsamen Teil unserer kulturellen Identität. Daher wird im Fach Informatik nicht nur Wissen über den verantwortungsvollen Umgang mit verschiedenen Medien, sondern auch über deren Aufbau erworben. Die Schülerinnen und Schüler erlernen bzw. erleben durch die informatische Bildung auch einen offenen und kritischen Umgang mit neuen Entwicklungen, zu deren Folgen sie eine eigene Position beziehen und somit in der Lage sind, selbstständig und mündig zu handeln. Ausgangspunkt des Informatikunterrichts der Oberstufe am HHG sind Fragestellungen mit lebensweltlichem Bezug. Gleichzeitig wird im besonderen Maße Wert daraufgelegt, dass die Schülerinnen und Schüler durch motivierende, altersgerechte und selbstentdeckende Zugänge informatorische Zusammenhänge individuell kennen und verstehen lernen. Im Fach Informatik wird dabei besonders die Informationsverarbeitungskompetenz entwickelt. Bei der Arbeit mit Informatiksystem erhalten die Lernenden regelmäßig Rückmeldungen über ihren individuellen Lernfortschritt. In einem längerfristigen Entwicklungsprozess arbeitet das Fach Informatik daran, die Bedingungen für individuelles und erfolgreiches Lernen zu optimieren. Das Fach Informatik ermöglicht vertiefte Einsicht in den Aufbau und die Funktionsweise von Informatiksystemen und leistet damit einen wesentlichen Beitrag zur Bildung in einer digitalisierten Welt, der auch einen wesentlichen Punkt des Schulprogramms darstellt.			

Übersichtsraster Unterrichtsvorhaben

Einführungsphase

Unterrichtsvorhaben E-I

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

Zentrale Kompetenzen:

- Kommunizieren und Kooperieren
- Darstellen und Interpretieren
- Argumentieren

Inhaltsfelder:

- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte:

- Einsatz, Nutzung und Aufbau von Informatiksystemen
- Wirkung der Automatisierung

Zeitbedarf: 6 Stunden

Unterrichtsvorhaben E-II

Thema: *Grundlagen der objektorientierten Analyse, Modellierung und Implementierung*

Zentrale Kompetenzen:

- Modellieren
- Implementieren
- Darstellen und Interpretieren
- Kommunizieren und Kooperieren

Inhaltsfelder:

- Daten und ihre Strukturierung
- Formale Sprachen und Automaten

Inhaltliche Schwerpunkte:

- Objekte und Klassen
- Syntax und Semantik einer Programmiersprache

Zeitbedarf: 8 Stunden

Unterrichtsvorhaben E-III

Thema: *Algorithmische Grundstrukturen in Java*

Zentrale Kompetenzen:

- Argumentieren
- Modellieren
- Implementieren
- Kommunizieren und Kooperieren

Inhaltsfelder:

- Daten und ihre Strukturierung
- Algorithmen
- Formale Sprachen und Automaten

Inhaltliche Schwerpunkte:

- Objekte und Klassen
- Syntax und Semantik einer Programmiersprache
- Analyse, Entwurf und Implementierung einfacher Algorithmen

Zeitbedarf: 18 Stunden

Unterrichtsvorhaben E-IV

Thema: *Das ist die digitale Welt! – Einführung in die Grundlagen, Anwendungsgebiete und Verarbeitung binärer Codierung*

Zentrale Kompetenzen:

- Kommunizieren und Kooperieren
- Darstellen und Interpretieren
- Argumentieren

Inhaltsfelder:

- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte:

- Binäre Codierung und Verarbeitung
- Besondere Eigenschaften der digitalen Speicherung und Verarbeitung von Daten

Zeitbedarf: 8 Stunden

Unterrichtsvorhaben E-V

Thema: *Modellierung und Implementierung von Klassen- und Objektbeziehungen anhand lebensnaher Anforderungsbeispiele*

Zentrale Kompetenzen:

- Kommunizieren und Kooperieren
- Darstellen und Interpretieren
- Argumentieren
- Modellieren
- Implementieren

Inhaltsfelder:

- Daten und ihre Strukturierung
- Algorithmen
- Formale Sprachen und Automaten

Inhaltliche Schwerpunkte:

- Objekte und Klassen
- Syntax und Semantik einer Programmiersprache
- Analyse, Entwurf und Implementierung einfacher Algorithmen

Zeitbedarf: 18 Stunden

Unterrichtsvorhaben E-VI

Thema: *Such- und Sortieralgorithmen*

Zentrale Kompetenzen:

- Argumentieren
- Modellieren
- Darstellen und Interpretieren
- Kommunizieren und Kooperieren

Inhaltsfelder:

- Algorithmen
- Daten und ihre Strukturierung

Inhaltliche Schwerpunkte:

- Algorithmen zum Suchen und Sortieren
- Analyse, Entwurf und Implementierung einfacher Algorithmen
- Objekte und Klassen

Zeitbedarf: 9 Stunden

Unterrichtsvorhaben E-VII

Thema: *Leben in der digitalen Welt – Immer mehr Möglichkeiten und immer mehr Gefahren!?*

Zentrale Kompetenzen:

- Kommunizieren und Kooperieren
- Darstellen und Interpretieren
- Argumentieren

Inhaltsfelder:

- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte:

- Geschichte der automatischen Datenverarbeitung
- Wirkungen der Automatisierung
- Dateisystem

Zeitbedarf: 10 Stunden

Qualifikationsphase (Q1 und Q2) - GRUNDKURS

Qualifikationsphase 1	
<u>Unterrichtsvorhaben Q1-I</u> Thema: <i>Wiederholung der objektorientierten Modellierung und Programmierung anhand einer kontextbezogenen Problemstellung</i> Zentrale Kompetenzen: • Argumentieren • Modellieren • Implementieren • Darstellen und Interpretieren • Kommunizieren und Kooperieren Inhaltsfelder: • Daten und ihre Strukturierung • Algorithmen • Formale Sprachen und Automaten • Informatiksysteme Inhaltliche Schwerpunkte: • Objekte und Klassen • Analyse, Entwurf und Implementierung von Algorithmen • Syntax und Semantik einer Programmiersprache • Nutzung von Informatiksystemen Zeitbedarf: 8 Stunden	<u>Unterrichtsvorhaben Q1-II</u> Thema: <i>Modellierung und Implementierung von Anwendungen mit dynamischen, linearen Datenstrukturen</i> Zentrale Kompetenzen: • Argumentieren • Modellieren • Implementieren • Darstellen und Interpretieren • Kommunizieren und Kooperieren Inhaltsfelder: • Daten und ihre Strukturierung • Algorithmen • Formale Sprachen und Automaten Inhaltliche Schwerpunkte: • Objekte und Klassen • Analyse, Entwurf und Implementierung von Algorithmen • Algorithmen in ausgewählten informatischen Kontexten • Syntax und Semantik einer Programmiersprache Zeitbedarf: 20 Stunden

Unterrichtsvorhaben Q1-III

Thema:

Suchen und Sortieren auf linearen Datenstrukturen

Zentrale Kompetenzen:

- Argumentieren
- Modellieren
- Implementieren
- Darstellen und Interpretieren
- Kommunizieren und Kooperieren

Inhaltsfelder:

- Algorithmen
- Formale Sprachen und Automaten

Inhaltliche Schwerpunkte:

- Analyse, Entwurf und Implementierung von Algorithmen
- Algorithmen in ausgewählten informatischen Kontexten
- Syntax und Semantik einer Programmiersprache

Zeitbedarf: 16 Stunden

Unterrichtsvorhaben Q1-IV

Thema:

Modellierung und Nutzung von relationalen Datenbanken in Anwendungskontexten

Zentrale Kompetenzen:

- Argumentieren
- Modellieren
- Implementieren
- Darstellen und Interpretieren
- Kommunizieren und Kooperieren

Inhaltsfelder:

- Daten und ihre Strukturierung
- Algorithmen
- Formale Sprachen und Automaten
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte:

- Datenbanken
- Algorithmen in ausgewählten informatischen Kontexten
- Syntax und Semantik einer Programmiersprache
- Sicherheit

Zeitbedarf: 20 Stunden

Unterrichtsvorhaben Q1-V

Thema:

Sicherheit und Datenschutz in Netzstrukturen

Zentrale Kompetenzen:

- Argumentieren
- Darstellen und Interpretieren
- Kommunizieren und Kooperieren

Inhaltsfelder:

- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte:

- Einzelrechner und Rechnernetzwerke
- Sicherheit
- Nutzung von Informatiksystemen, Wirkungen der Automatisierung

Zeitbedarf: 10 Stunden

Qualifikationsphase 2

Unterrichtsvorhaben Q2-I

Thema:

Modellierung und Implementierung von Anwendungen mit dynamischen, nichtlinearen Datenstrukturen

Zentrale Kompetenzen:

- Argumentieren
- Modellieren
- Implementieren
- Darstellen und Interpretieren
- Kommunizieren und Kooperieren

Inhaltsfelder:

- Daten und ihre Strukturierung
- Algorithmen
- Formale Sprachen und Automaten

Inhaltliche Schwerpunkte:

- Objekte und Klassen
- Analyse, Entwurf und Implementierung von Algorithmen
- Algorithmen in ausgewählten informatischen Kontexten
- Syntax und Semantik einer Programmiersprache

Zeitbedarf: 24 Stunden

Unterrichtsvorhaben Q2-II

Thema:

Endliche Automaten und formale Sprachen

Zentrale Kompetenzen:

- Argumentieren
- Modellieren
- Darstellen und Interpretieren
- Kommunizieren und Kooperieren

Inhaltsfelder:

- Endliche Automaten und formale Sprachen

Inhaltliche Schwerpunkte:

- Endliche Automaten
- Grammatiken regulärer Sprachen
- Möglichkeiten und Grenzen von Automaten und formalen Sprachen

Zeitbedarf: 20 Stunden

Unterrichtsvorhaben Q2-III

Thema:

Prinzipielle Arbeitsweise eines Computers und Grenzen der Automatisierbarkeit

Zentrale Kompetenzen:

- Argumentieren
- Kommunizieren und Kooperieren

Inhaltsfelder:

- Informatiksysteme
- Informatik, Mensch und Gesellschaft

Inhaltliche Schwerpunkte:

- Einzelrechner und Rechnernetzwerke
- Grenzen der Automatisierung

Zeitbedarf: 12 Stunden

Konkretisierung der Unterrichtsvorhaben

Einführungsphase

Unterrichtsvorhaben EF-I

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

Leitfragen: Was macht Informatik? Welche fundamentalen Konzepte müssen Informatikerinnen und Informatiker in ihre Arbeit einbeziehen, damit informatische Systeme effizient und zuverlässig arbeiten können? Wo lassen sich diese Konzepte (in Ansätzen) in dem schuleigenen Netzwerk- und Computersystem wiederfinden?

Vorhabenbezogene Konkretisierung:

Im ersten Unterrichtsvorhaben werden die fünf Inhaltsfelder des Faches Informatik beispielhaft an einem Informatiksystem erarbeitet. Das Unterrichtsvorhaben ist so strukturiert, dass die Schülerinnen und Schüler anhand bekannter Alltagstechnik die Grundideen fundamentaler informatischer Konzepte (Inhaltsfelder) größtenteils selbstständig erarbeiten und nachvollziehen.

Ausgehend von dem bekannten Bedienungs- und Funktionalitätswissen eines Navigationsgerätes werden die Strukturierung von Daten, das Prinzip der Algorithmik, die Eigenheit formaler Sprachen, die Kommunikationsfähigkeit von Informatiksystemen und die positiven und negativen Auswirkungen auf Mensch und Gesellschaft thematisiert. Das am Beispiel Navigationsgerät erworbene Wissen kann auf weitere den Schülerinnen und Schülern bekannte Informatiksysteme übertragen werden.

In einem letzten Schritt kann ausgehend von den Inhaltsfeldern das Schulnetzwerk in Ansätzen so analysiert werden, dass ein kompetenter Umgang mit diesem ermöglicht wird.

Zeitbedarf: 6 Stunden

Sequenzierung des Unterrichtsvorhabens:		
Unterrichtssequenzen 1. Informatiksysteme und ihr genereller Aufbau (a) Daten und ihre Strukturierung (b) Algorithmen (c) Formale Sprachen und Automaten (d) Informatiksysteme (e) Informatik, Mensch und Gesellschaft 2. Der kompetente Umgang mit dem Schulnetzwerk (a) Erstellen und Anlegen von Ordnerstrukturen (b) Sortieren von Dateien und Ordnern (c) Eingabe von Befehlen über Eingabeaufforderung (d) Einzelrechner und Netzwerk (e) Sicherheit und Datenschutz	Zu entwickelnde Kompetenzen Die Schülerinnen und Schüler - bewerten anhand von Fallbeispielen die Auswirkungen des Einsatzes von Informatiksystemen (A) - nutzen die im Unterricht eingesetzten Informatiksysteme selbstständig, sicher, zielführend und verantwortungsbewusst (D)	Kapitel und Materialien <i>Kapitel 1 Was macht Informatik?</i> Als Anschauungsmaterial bieten sich Navigationsgeräte und Landkarten an. <i>Kapitel 1 Was macht Informatik?</i> Interview mit dem Netzwerkadministrator, Benutzer- und Datenschutzbestimmungen der Schule

Unterrichtsvorhaben EF-II

Thema: *Grundlagen der objektorientierten Analyse, Modellierung und Implementierung*

Leitfragen: Wie lassen sich Gegenstandsbereiche informatisch modellieren und in einem Greenfoot-Szenario informatisch realisieren?

Vorhabenbezogene Konkretisierung:

Ein zentraler Bestandteil des Informatikunterrichts der Einführungsphase ist die Objektorientierte Programmierung. Dieses Unterrichtsvorhaben führt in die Grundlagen der Analyse, Modellierung und Implementierung in diesem Kontext ein.

Dazu werden zunächst konkrete Gegenstandsbereiche aus der Lebenswelt der Schülerinnen und Schüler analysiert und im Sinne des objektorientierten Paradigmas strukturiert. Dabei werden die grundlegenden Begriffe der Objektorientierung und Modellierungswerkzeuge wie Objektdiagramme und Klassendiagramme eingeführt.

Im Anschluss wird die objektorientierte Analyse für das Greenfoot-Szenario Planetenerkundung durchgeführt. Die vom Szenario vorgegebenen Klassen werden von Schülerinnen und Schülern in Teilen analysiert und entsprechende Objekte anhand einfacher Problemstellungen erprobt. Die Lernenden implementieren und testen einfache Programme. Die Greenfoot-Umgebung ermöglicht es, Beziehungen zwischen Klassen zu einem späteren Zeitpunkt (Kapitel 3) zu thematisieren. So kann der Fokus hier auf Grundlagen wie der Unterscheidung zwischen Klasse und Objekt, Attribute, Methoden, Objektidentität und Objektzustand gelegt werden.

Da in Kapitel 2 zudem auf die Verwendung von Kontrollstrukturen verzichtet wird und der Quellcode aus einer rein linearen Sequenz besteht, ist auf diese Weise eine Fokussierung auf die Grundlagen der Objektorientierung möglich, ohne dass algorithmische Probleme ablenken. Natürlich kann die Arbeit an diesen Projekten unmittelbar zum nächsten Unterrichtsvorhaben (Weiterhin Kapitel 2) führen. Dort stehen Kontrollstrukturen im Mittelpunkt.

Zeitbedarf: 8 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Identifikation von Objekten und Klassen (a) An einem lebensweltnahen Beispiel werden Objekte und Klassen im Sinne der objektorientierten Modellierung eingeführt. (b) Objekte werden durch Objektdiagramme, Klassen durch Klassendiagramme dargestellt. (c) Die Modellierungen werden einem konkreten Anwendungsfall entsprechend angepasst.	Die Schülerinnen und Schüler - ermitteln bei der Analyse einfacher Problemstellungen Objekte, ihre Eigenschaften und ihre Operationen (M), - stellen den Zustand eines Objekts dar (D), - modellieren Klassen mit ihren Attributen und ihren Methoden (M), - implementieren einfache Algorithmen unter Beachtung der Syntax und Semantik einer Programmiersprache (I), - implementieren Klassen in einer Programmiersprache, auch unter Nutzung dokumentierter Klassenbibliotheken (I).	<i>Kapitel 2 Algorithmen</i> Projekteinstieg: Planetenerkundung
2. Analyse von Objekten und Klassen im Greenfoot-Szenario (a) Schritte der objektorientierten Analyse, Modellierung und Implementation (b) Analyse und Erprobung der Objekte im Greenfoot-Szenario		<i>Kapitel 2 Algorithmen</i> Projekteinstieg: Planetenerkundung
3. Implementierung einfacher Aktionen in Greenfoot (a) Quelltext einer Java-Klasse (b) Implementation eigener Methoden, Dokumentation mit JavaDoc (c) Programme übersetzen (Aufgabe des Compilers) und testen		<i>Kapitel 2 Algorithmen</i> 2.1 Erste Programme – Methoden schreiben

Unterrichtsvorhaben EF-III

Thema: *Algorithmische Grundstrukturen in Java*

Leitfragen: *Wie lassen sich Aktionen von Objekten flexibel realisieren?*

Vorhabenbezogene Konkretisierung:

Das Ziel dieses Unterrichtsvorhabens besteht darin, das Verhalten von Objekten flexibel zu programmieren. Ein erster Schwerpunkt liegt dabei auf der Erarbeitung von Kontrollstrukturen. Die Strukturen Wiederholung und bedingte Anweisung werden an einfachen Beispielen eingeführt und anschließend anhand komplexerer Problemstellungen erprobt. Da die zu entwickelnden Algorithmen zunehmend umfangreicher werden, werden systematische Vorgehensweisen zur Entwicklung von Algorithmen thematisiert.

Ein zweiter Schwerpunkt des Unterrichtsvorhabens liegt auf dem Einsatz von Variablen. Beginnend mit lokalen Variablen, die in Methoden und Zählschleifen zum Einsatz kommen, über Variablen in Form von Parametern und Rückgabewerten von Methoden, bis hin zu Variablen, die die Attribute einer Klasse realisieren, lernen die Schülerinnen und Schüler die unterschiedlichen Einsatzmöglichkeiten des Variablenkonzepts anzuwenden.

Zeitbedarf: 18 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Algorithmen (a) Wiederholungen (While-Schleife, Zählschleife) (b) bedingte Anweisungen (c) Verknüpfung von Bedingungen durch die logischen Funktionen UND, ODER und NICHT (d) Systematisierung des Vorgehens zur Entwicklung von Algorithmen zur Lösung komplexerer Probleme	Die Schülerinnen und Schüler - analysieren und erläutern einfache Algorithmen und Programme (A), - entwerfen einfache Algorithmen und stellen sie umgangssprachlich und grafisch dar (M), - ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen zu (M), - modifizieren einfache Algorithmen und Programme (I), - implementieren Algorithmen unter Verwendung von Variablen und Wertzuweisungen, Kontrollstrukturen sowie Methodenaufrufen (I), - implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I), - implementieren einfache Algorithmen unter Beachtung der Syntax und Semantik einer Programmiersprache (I), - testen Programme schrittweise anhand von Beispielen (I), - interpretieren Fehlermeldungen und korrigieren den Quellcode (I).	<i>Kapitel 2 Algorithmen</i> 2.2 Wiederholungen 2.3 Bedingte Anweisungen 2.4 Logische Operationen
2. Variablen und Methoden (a) Implementierung eigener Methoden mit lokalen Variablen, auch zur Realisierung einer Zählschleife (b) Implementierung eigener Methoden mit Parameterübergabe und/oder Rückgabewert (c) Implementierung von Konstruktoren (d) Realisierung von Attributen		<i>Kapitel 2 Algorithmen</i> 2.5 Lokale Variablen und Zählschleifen 2.6 Parameter und Rückgabewerte 2.7 Algorithmen entwickeln

Unterrichtsvorhaben EF-IV

Thema: *Das ist die digitale Welt! – Einführung in die Grundlagen, Anwendungsgebiete und Verarbeitung binärer Codierung*

Leitfragen: Wie werden binäre Informationen gespeichert und wie können sie davon ausgehend weiter verarbeitet werden? Wie unterscheiden sich analoge Medien und Geräte von digitalen Medien und Geräten? Wie ist der Grundaufbau einer digitalen Rechenmaschine?

Vorhabenbezogene Konkretisierung:

Das Unterrichtsvorhaben hat die binäre Speicherung und Verarbeitung sowie deren Besonderheiten zum Inhalt.

Im ersten Schritt erarbeiten die Schülerinnen und Schüler anhand ihnen bekannter technischer Gegenstände die Gemeinsamkeiten, Unterschiede und Besonderheiten der jeweiligen analogen und digitalen Version. Nach dieser ersten grundlegenden Einordnung des digitalen Prinzips wenden die Schülerinnen und Schüler das Binäre als Zahlensystem mit arithmetischen und logischen Operationen an und codieren Zeichen binär.

Zum Abschluss soll der grundlegende Aufbau eines Rechnersystems im Sinne der Von-Neumann-Architektur erarbeitet werden.

Zeitbedarf: 8 Stunden

Sequenzierung des Unterrichtsvorhabens:		
Unterrichtssequenzen 1. Analoge und digitale Aufbereitung und Verarbeitung von Daten (a) Erarbeitung der Unterschiede von analog und digital (b) Zusammenfassung und Bewertung der technischen Möglichkeiten von analog und digital 2. Der Umgang mit binärer Codierung von Informationen (a) Das binäre (und hexadezimale) Zahlensystem (b) Binäre Informationsspeicherung (c) Binäre Verschlüsselung (d) Implementation eines Binärumsrechners 3. Aufbau informatischer Systeme (a) Identifikation des EVA-Prinzips als grundlegende Arbeitsweise informatischer Systemen (b) Nachvollziehen der von-Neumann-Architektur als relevantes Modell der Umsetzung des EVA-Prinzips	Zu entwickelnde Kompetenzen Die Schülerinnen und Schüler - bewerten anhand von Fallbeispielen die Auswirkungen des Einsatzes von Informatiksystemen (A) - stellen ganze Zahlen und Zeichen in Binärcodes dar (D) - interpretieren Binärcodes als Zahlen und Zeichen (D) - beschreiben und erläutern den strukturellen Aufbau und die Arbeitsweise singulärer Rechner am Beispiel der „Von-Neumann-Architektur“ (A) - nutzen das Internet zur Recherche, zum Datenaustausch und zur Kommunikation (K) - implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I)	Kapitel und Materialien <i>Die digitale Welt 001 – Von analog zu digital</i> <i>Die digitale Welt 010 – Binäre Welt</i> <i>Die digitale Welt 100 – Der Von-Neumann-Rechner</i>

Unterrichtsvorhaben EF-V

Thema: *Modellierung und Implementierung von Klassen- und Objektbeziehungen anhand lebensnaher Anforderungsbeispiele*

Leitfragen: Wie werden realistische Systeme anforderungsspezifisch reduziert, als Entwurf modelliert und implementiert? Wie kommunizieren Objekte und wie wird dieses dargestellt und realisiert?

Vorhabenbezogene Konkretisierung:

Das Unterrichtsvorhaben hat die Entwicklung von Objekt -und Klassenbeziehungen zum Schwerpunkt. Dazu werden, ausgehend von der Realität, über Objektidentifizierung und Entwurf bis hin zur Implementation kleine Softwareprodukte in Teilen oder ganzheitlich erstellt.

Zuerst identifizieren die Schülerinnen und Schüler Objekte und stellen diese dar. Aus diesen Objekten werden Klassen und ihre Beziehungen in Entwurfsdiagrammen erstellt.

Nach diesem ersten Modellierungsschritt werden über Klassendokumentationen und der Darstellung von Objektkommunikationen anhand von Sequenzdiagrammen Implementationsdiagramme entwickelt. Danach werden die Implementationsdiagramme unter Berücksichtigung der Klassendokumentationen in Javaklassen programmiert. In einem letzten Schritt wird das Konzept der Vererbung sowie seiner Vorteile erarbeitet.

Schließlich sind die Schülerinnen und Schüler in der Lage, eigene kleine Softwareprojekte zu entwickeln. Ausgehend von der Dekonstruktion und Erweiterung eines Spiels wird ein weiteres Projekt von Grund auf modelliert und implementiert. Dabei können arbeitsteilige Vorgehensweisen zum Einsatz kommen. In diesem Zusammenhang wird auch das Erstellen von graphischen Benutzeroberflächen eingeführt.

Zeitbedarf: 18 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Umsetzung von Anforderungen in Entwurfsdiagramme (a) Aus Anforderungsbeschreibungen werden Objekte mit ihren Eigenschaften identifiziert (b) Gleichartige Objekte werden in Klassen (Entwurf) zusammengefasst und um Datentypen und Methoden erweitert	Die Schülerinnen und Schüler - analysieren und erläutern eine objektorientierte Modellierung (A), - stellen die Kommunikation zwischen Objekten grafisch dar (M), - ermitteln bei der Analyse einfacher Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M), - modellieren Klassen mit ihren Attributen, ihren Methoden und Assoziationsbeziehungen (M), - ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen oder lineare Datensammlungen zu (M), - ordnen Klassen, Attributen und Methoden ihren Sichtbarkeitsbereich zu (M), - modellieren Klassen unter Verwendung von Vererbung (M), - implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I), - testen Programme schrittweise anhand von Beispielen (I), - interpretieren Fehlermeldungen und korrigieren den Quellcode (I), - analysieren und erläutern einfache Algorithmen und Programme (A) - modifizieren einfache Algorithmen und Programme (I), - entwerfen einfache Algorithmen und stellen sie umgangssprachlich und grafisch dar (M).	Kapitel 3 Von der Idee zur Software 3.1 Der Klassenentwurf
2. Implementationsdiagramme als erster Schritt der Programmierung (a) Erweiterung des Entwurfsdiagramms um Konstruktoren und get- und set-Methoden (b) Festlegung von Datentypen in Java, sowie von Rückgaben und Parametern (c) Entwicklung von Klassendokumentationen (d) Erstellung von Sequenzdiagrammen als Vorbereitung für die Programmierung		Kapitel 3 Von der Idee zur Software 3.2 Die Klassenimplementation
3. Programmierung anhand der Dokumentation und des Implementations- und Sequenzdiagrammes (a) Klassen werden in Java-Quellcode umgesetzt. (b) Das Geheimnisprinzip wird umgesetzt. (c) Einzelne Klassen und das Gesamtsystem werden anhand der Anforderungen und		Kapitel 3 Von der Idee zur Software 3.2 Die Klassenimplementation

Dokumentationen auf ihre Korrektheit überprüft.	- stellen Klassen, Assoziations- und Vererbungs-beziehungen in Diagrammen grafisch dar (D), - dokumentieren Klassen durch Beschreibung der Funktionalität der Methoden (D)	
4. Vererbungsbeziehungen (a) Das Grundprinzip der Vererbung wird erarbeitet. (b) Die Vorteile der Vererbungsbeziehungen (c) Vererbung wird implementiert 5. Softwareprojekt (a) Analyse und Dekonstruktion eines Spiels (Modelle, Quelltexte) (b) Erweiterung des Spiels um weitere Funktionalitäten (c) Modellierung eines Spiels aufgrund einer Anforderungsbeschreibung, inklusive einer grafischen Benutzeroberfläche (d) (arbeitsteilige) Implementation des Spiels		3 Von der Idee zur Software 3.3 Vererbung Kapitel 5 Softwareprojekte 5.1 Softwareentwicklung 5.2 Oberflächen

Unterrichtsvorhaben EF-V

Thema: *Modellierung und Implementierung von Klassen- und Objektbeziehungen anhand lebensnaher Anforderungsbeispiele*

Leitfragen: Wie werden realistische Systeme anforderungsspezifisch reduziert, als Entwurf modelliert und implementiert? Wie kommunizieren Objekte und wie wird dieses dargestellt und realisiert?

Vorhabenbezogene Konkretisierung:

Das Unterrichtsvorhaben hat die Entwicklung von Objekt -und Klassenbeziehungen zum Schwerpunkt. Dazu werden, ausgehend von der Realität, über Objektidentifizierung und Entwurf bis hin zur Implementation kleine Softwareprodukte in Teilen oder ganzheitlich erstellt.

Zuerst identifizieren die Schülerinnen und Schüler Objekte und stellen diese dar. Aus diesen Objekten werden Klassen und ihre Beziehungen in Entwurfsdiagrammen erstellt.

Nach diesem ersten Modellierungsschritt werden über Klassendokumentationen und der Darstellung von Objektkommunikationen anhand von Sequenzdiagrammen Implementationsdiagramme entwickelt. Danach werden die Implementationsdiagramme unter Berücksichtigung der Klassendokumentationen in Javaklassen programmiert. In einem letzten Schritt wird das Konzept der Vererbung sowie seiner Vorteile erarbeitet.

Schließlich sind die Schülerinnen und Schüler in der Lage, eigene kleine Softwareprojekte zu entwickeln. Ausgehend von der Dekonstruktion und Erweiterung eines Spiels wird ein weiteres Projekt von Grund auf modelliert und implementiert. Dabei können arbeitsteilige Vorgehensweisen zum Einsatz kommen. In diesem Zusammenhang wird auch das Erstellen von graphischen Benutzeroberflächen eingeführt.

Zeitbedarf: 18 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Umsetzung von Anforderungen in Entwurfsdiagramme (a) Aus Anforderungsbeschreibungen werden Objekte mit ihren Eigenschaften identifiziert (b) Gleichartige Objekte werden in Klassen (Entwurf) zusammengefasst und um Datentypen und Methoden erweitert	Die Schülerinnen und Schüler - analysieren und erläutern eine objektorientierte Modellierung (A), - stellen die Kommunikation zwischen Objekten grafisch dar (M), - ermitteln bei der Analyse einfacher Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M), - modellieren Klassen mit ihren Attributen, ihren Methoden und Assoziationsbeziehungen (M), - ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen oder lineare Datensammlungen zu (M), - ordnen Klassen, Attributen und Methoden ihren Sichtbarkeitsbereich zu (M), - modellieren Klassen unter Verwendung von Vererbung (M), - implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I), - testen Programme schrittweise anhand von Beispielen (I), - interpretieren Fehlermeldungen und korrigieren den Quellcode (I), - analysieren und erläutern einfache Algorithmen und Programme (A) - modifizieren einfache Algorithmen und Programme (I), - entwerfen einfache Algorithmen und stellen sie umgangssprachlich und grafisch dar (M).	Kapitel und Materialien <i>Kapitel 5 Klassenentwurf</i> 5.1 Von der Realität zum Programm 5.2 Objekte 5.3 Klassen und Beziehungen entwerfen
2. Implementationsdiagramme als erster Schritt der Programmierung (a) Erweiterung des Entwurfsdiagramms um Konstruktoren und get- und set-Methoden (b) Festlegung von Datentypen in Java, sowie von Rückgaben und Parametern (c) Entwicklung von Klassendokumentationen (d) Erstellung von Sequenzdiagrammen als Vorbereitung für die Programmierung	(a) Erweiterung des Entwurfsdiagramms um Konstruktoren und get- und set-Methoden (b) Festlegung von Datentypen in Java, sowie von Rückgaben und Parametern (c) Entwicklung von Klassendokumentationen (d) Erstellung von Sequenzdiagrammen als Vorbereitung für die Programmierung	<i>Kapitel 5 Klassenentwurf</i> 5.4 Klassen und Beziehungen implementieren
3. Programmierung anhand der Dokumentation und des Implementations- und Sequenzdiagrammes (a) Klassen werden in Java-Quellcode umgesetzt. (b) Das Geheimnisprinzip wird umgesetzt. (c) Einzelne Klassen und das Gesamtsystem werden anhand der Anforderungen	(a) Klassen werden in Java-Quellcode umgesetzt. (b) Das Geheimnisprinzip wird umgesetzt. (c) Einzelne Klassen und das Gesamtsystem werden anhand der Anforderungen	<i>Kapitel 5 Klassenentwurf</i> 5.4 Klassen und Beziehungen implementieren

Unterrichtsvorhaben EF-VI

Thema: *Such- und Sortieralgorithmen*

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

Vorhabenbezogene Konkretisierung:

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

Zunächst lernen die Schülerinnen und Schüler das Feld als eine erste Datensammlung kennen. Alle Algorithmen dieses Kapitels arbeiten auf einem Feld.

Die Schülerinnen und Schüler lernen zunächst Strategien des Suchens (lineare Suche, binäre Suche, Hashing) und dann des Sortierens (Selection Sort, Insertion Sort, Bubble Sort) kennen.

Daran anschließend lernen die Schülerinnen und Schüler zunächst Strategien des Suchens (lineare Suche, binäre Suche, Hashing) und dann des Sortierens (Selection Sort, Insertion Sort, Bubble Sort) kennen. Die Projekteinstiege dienen dazu, die jeweiligen Strategien handlungsorientiert zu erkunden und intuitive Effizienzbetrachtungen der Suchalgorithmen vorzunehmen.

Schließlich wird die Effizienz unterschiedlicher Sortierverfahren beurteilt.

Zeitbedarf: 9 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Kapitel und Materialien
1. Modellierung und Implementation von Datenansammlungen (a) Modellierung von Attributen als Felder (b) Deklaration, Instanziierung und Zugriffe auf ein Feld	Die Schülerinnen und Schüler - analysieren Such- und Sortieralgorithmen und wenden sie auf Beispiele an (D) - entwerfen einen weiteren Algorithmus zum Sortieren (M) - beurteilen die Effizienz von Algorithmen am Beispiel von Sortierverfahren hinsichtlich Zeit und Speicherplatzbedarf (A) - ordnen Attributen lineare Datenansammlungen zu (M)	<i>Kapitel 4 Suchen und Sortieren</i> 4.1 Speichern mit Struktur – Arrays
2. Explorative Erarbeitung von Suchverfahren (a) Erkundung von Strategien für das Suchen auf unsortierten Daten, auf sortierten Daten und mithilfe einer Berechnungsfunktion (b) Vergleich der drei Verfahren durch intuitive Effizienzbetrachtungen		<i>Kapitel 4 Suchen und Sortieren</i> Projekteinstieg 1: Suchen 4.2 Suchen mit System Lineare Suche Binäre Suche Hashing
3. Systematisierung von Algorithmen und Effizienzbetrachtungen (a) Formulierung (falls selbst gefunden) oder Erläuterung von mehreren Algorithmen im Pseudocode (b) Anwendung von Sortieralgorithmen auf verschiedene Beispiele (c) Bewertung von Algorithmen anhand der Anzahl der nötigen Vergleiche (d) Analyse eines weiteren Sortieralgorithmus (sofern nicht in (a) bereits geschehen)		<i>Kapitel 4 Suchen und Sortieren</i> Projekteinstieg 2: Sortieren 4.3 Ordnung ist das halbe Leben!? – Sortieren Sortieren Selection Sort Insertion Sort Bubble Sort

Unterrichtsvorhaben EF-VII

Thema: Leben in der digitalen Welt – Immer mehr Möglichkeiten und immer mehr Gefahren!?

Leitfragen: Welche Entwicklungen, Ideen und Erfindungen haben zur heutigen Informatik geführt? Welche Auswirkungen hat die Informatik für das Leben des modernen Menschen?

Vorhabenbezogene Konkretisierung:

Das Unterrichtsvorhaben stellt die verschiedenen Entwicklungsstränge der Informatik in den Fokus. Darüber hinaus wird beispielhaft analysiert und bewertet, welche Möglichkeiten und Gefahren die moderne Informationsverarbeitung mit sich bringt.

Im ersten Schritt des Unterrichtsvorhabens wird anhand von Themenkomplexen entscheidende Entwicklungen der Informatik erarbeitet. Dabei werden auch übergeordnete Tendenzen identifiziert.

Ausgehend von dieser Betrachtung kann die aktuelle Informatik hinsichtlich ihrer Leistungsfähigkeit analysiert werden. Dabei soll herausgestellt werden, welche positiven und negativen Folgen Informatiksysteme mit sich bringen können.

Zeitbedarf: 12 Stunden

[illegible]

Qualifikationsphase – Grundkurs

Die folgenden Kompetenzen aus dem Bereich *Kommunizieren und Kooperieren* werden in allen Unterrichtsvorhaben der Qualifikationsphase vertieft und sollen aus Gründen der Lesbarkeit nicht in jedem Unterrichtsvorhaben separat aufgeführt werden:

Die Schülerinnen und Schüler

- verwenden die Fachsprache bei der Kommunikation über informatische Sachverhalte (K),
- nutzen das verfügbare Informatiksystem zur strukturierten Verwaltung von Dateien unter Berücksichtigung der Rechteverwaltung (K),
- organisieren und koordinieren kooperatives und eigenverantwortliches Arbeiten (K),
- strukturieren den Arbeitsprozess, vereinbaren Schnittstellen und führen Ergebnisse zusammen (K),
- beurteilen Arbeitsorganisation, Arbeitsabläufe und Ergebnisse (K),
- präsentieren Arbeitsabläufe und -ergebnisse adressatengerecht (K).

Unterrichtsvorhaben Q1-I:

Thema: Wiederholung der objektorientierten Modellierung und Programmierung

Leitfragen: *Wie modelliert und implementiert man zu einer Problemstellung in einem geeigneten Anwendungskontext Java-Klassen inklusive ihrer Attribute, Methoden und Beziehungen? Wie kann man die Modellierung und die Funktionsweise der Anwendung grafisch darstellen?*

Vorhabenbezogene Konkretisierung:

Zu einer Problemstellung in einem Anwendungskontext soll eine Java-Anwendung entwickelt werden. Die Problemstellung soll so gewählt sein, dass für diese Anwendung die Verwendung einer abstrakten Oberklasse als Generalisierung verschiedener Unterklassen sinnvoll erscheint und eine Klasse durch eine Unterklasse spezialisiert werden kann. Um die Aufgabe einzugrenzen, können (nach der ersten Problemanalyse) einige Teile (Modellierungen oder Teile von Java-Klassen) vorgegeben werden.

Die Schülerinnen und Schüler erläutern und modifizieren den ersten Entwurf und modellieren sowie implementieren weitere Klassen und Methoden für eine entsprechende Anwendung. Klassen und ihre Beziehungen werden in einem Implementationsdiagramm dargestellt. Dabei werden Sichtbarkeitsbereiche zugeordnet. Exemplarisch wird eine Klasse dokumentiert. Der Nachrichtenaustausch zwischen verschiedenen Objekten wird verdeutlicht, indem die Kommunikation zwischen zwei ausgewählten Objekten grafisch dargestellt wird. In diesem Zusammenhang wird das Nachrichtenkonzept der objektorientierten Programmierung wiederholt.

Zeitbedarf: 8 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Wiederholung und Erweiterung der objektorientierten Modellierung und Programmierung durch Analyse und Erweiterung eines kontextbezogenen Beispiels (a) Analyse der Problemstellung (b) Analyse der Modellierung (Implementationsdiagramm)	Die Schülerinnen und Schüler - analysieren und erläutern objektorientierte Modellierungen (A), - beurteilen die syntaktische Korrektheit und die Funktionalität von Programmen (A), - modellieren Klassen mit ihren Attributen, Methoden und ihren	<i>Beispiel:</i> Wetthüpfen Für ein Wetthüpfen zwischen einem Hasen, einem Hund und einem Vogel werden die Tiere gezeichnet. Alle Tiere springen wiederholt nach links. Die Höhe und Weite jedes Hüpfers ist zufällig. Evtl. marschieren sie anschließend hintereinander her. oder

(c) Erweiterung der Modellierung im Implementationsdiagramm (Vererbung, abstrakte Klasse) (d) Kommunikation zwischen mindestens zwei Objekten (grafische Darstellung) (e) Dokumentation von Klassen (f) Implementierung der Anwendung oder von Teilen der Anwendung	Assoziationsbeziehungen unter Angabe von Multiplizitäten (M), • ordnen Klassen, Attributen und Methoden ihre Sichtbarkeitsbereiche zu (M), • modellieren abstrakte und nicht abstrakte Klassen unter Verwendung von Vererbung durch Spezialisieren und Generalisieren (M), • 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), • interpretieren Fehlermeldungen und korrigieren den Quellcode (I), • stellen Klassen und ihre Beziehungen in Diagrammen grafisch dar (D), • dokumentieren Klassen (D), • stellen die Kommunikation zwischen Objekten grafisch dar (D).	<i>Beispiel:</i> Tannenbaum Ein Tannenbaum soll mit verschiedenen Arten von Schmuckstücken versehen werden, die durch unterschiedliche geometrische Objekte dargestellt werden. Es gibt Kugeln, Päckchen in der Form von Würfeln und Zuckerringe in Form von Toren. Ein Prototyp, der bereits mit Kugeln geschmückt werden kann, kann zur Verfügung gestellt werden. Da alle Schmuckstücke über die Funktion des Auf- und Abschmückens verfügen sollen, liegt es nahe, dass entsprechende Methoden in einer gemeinsamen Oberklasse realisiert werden. <i>Materialien:</i> Ergänzungsmaterialien zum Lehrplannavigator Unterrichtsvorhaben Q1.1-Wiederholung (Download Q1-I.1)
--	---	--

Unterrichtsvorhaben Q1-II:

Thema: Modellierung und Implementierung von Anwendungen mit dynamischen, linearen Datenstrukturen

Leitfrage: *Wie können beliebig viele linear angeordnete Daten im Anwendungskontext verwaltet werden?*

Vorhabenbezogene Konkretisierung:

Nach Analyse einer Problemstellung in einem geeigneten Anwendungskontext, in dem Daten nach dem First-In-First-Out-Prinzip verwaltet werden, werden der Aufbau von Schlangen am Beispiel dargestellt und die Operationen der Klasse Queue erläutert. Anschließend werden für die Anwendung notwendige Klassen modelliert und implementiert. Eine Klasse für eine den Anforderungen der Anwendung entsprechende Oberfläche sowie die Klasse Queue wird dabei von der Lehrkraft vorgegeben. Anschließend wird die Anwendung modifiziert, um den Umgang mit der Datenstruktur zu üben. Anhand einer Anwendung, in der Daten nach dem Last-In-First-Out-Prinzip verwaltet werden, werden Unterschiede zwischen den Datenstrukturen Schlange und Stapel erarbeitet. Um einfacher an Objekte zu gelangen, die zwischen anderen gespeichert sind, wird die Klasse List eingeführt und in einem Anwendungskontext verwendet. In mindestens einem weiteren Anwendungskontext wird die Verwaltung von Daten in Schlangen, Stapeln oder Listen vertieft. Modellierungen werden dabei in Entwurfs- und Implementationsdiagrammen dargestellt.

Zeitbedarf: 20 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Die Datenstruktur Schlange im Anwendungskontext unter Nutzung der Klasse Queue (a) Analyse der Problemstellung, Ermittlung von Objekten, ihren Eigenschaften und Operationen (b) Erarbeitung der Funktionalität der Klasse Queue (c) Modellierung und Implementierung der Anwendung unter Verwendung eines	Die Schülerinnen und Schüler • 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),	<i>Beispiel:</i> 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

oder mehrerer Objekte der Klasse Queue	• 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), • 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).	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. <i>Materialien:</i> Ergänzungsmaterialien zum Lehrplannavigator Unterrichtsvorhaben Q1.2 – Warteschlange (Download Q1-II.1)
2. Die Datenstruktur Stapel im Anwendungskontext unter Nutzung der Klasse Stack (a) Analyse der Problemstellung, Ermittlung von Objekten, ihren Eigenschaften und Operationen (b) Erarbeitung der Funktionalität der Klasse Stack (c) Modellierung und Implementierung der Anwendung unter Verwendung eines		<i>Beispiel:</i> Heftstapel In einem Heftstapel soll das Heft einer Schülerin gefunden werden. oder <i>Beispiel:</i> Kisten stapeln In einem Stapel nummerierter Kisten soll eine bestimmte Kiste gefunden und an einen Kunden geliefert werden. Dazu müssen Kisten auf

oder mehrerer Objekte der Klasse Stack		verschiedene Stapel gestapelt und wieder zurückgestellt werden.
3. Die Datenstruktur lineare Liste im Anwendungskontext unter Nutzung der Klasse List (a) Erarbeitung der Vorteile der Klasse List im Gegensatz zu den bereits bekannten linearen Strukturen (b) Modellierung und Implementierung einer kontextbezogenen Anwendung unter Verwendung der Klasse List.		<i>Beispiel:</i> 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. <i>Materialien:</i> Ergänzungsmaterialien zum Lehrplannavigator Unterrichtsvorhaben Q1.2 - Listen (Download Q1-II.2)
4. Vertiefung - Anwendungen von Listen, Stapeln oder Schlangen in mindestens einem weiteren Kontext		<i>Beispiel:</i> Skispringen Ein Skispringen hat folgenden Ablauf: 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. <i>Beispiel:</i> Terme in Postfix-Notation

		Die sog. UPN (<i>Umgekehrt-Polnische-Notation</i>) bzw. <i>Postfix-Notation</i> eines Terms setzt den Operator hinter die Operanden. Um einen Term aus der gewohnten Infixschreibweise in einen Term in UPN umzuwandeln oder um den Wert des Terms zu berechnen, kann ein Stack verwendet werden. <i>Beispiel:</i> Rangierbahnhof Auf einem Güterbahnhof gibt es drei Gleise, die nur zu einer Seite offen sind. Wagons können also von einer Seite auf das Gleis fahren und nur rückwärts wieder hinausfahren. Die Wagons tragen Nummern, wobei die Nummer jedoch erst eingesehen werden kann, wenn der Wagon der vorderste an der offenen Gleisseite ist. (Zwischen den Wagons herumzuturnen, um die anderen Wagonnummern zu lesen, wäre zu gefährlich.) Zunächst stehen alle Wagons unsortiert auf einem Gleis. Ziel ist es, alle Wagons in ein anderes Gleis zu fahren, so dass dort die Nummern der Wagons vom Gleisende aus aufsteigend in richtiger Reihenfolge sind. Zusätzlich zu diesen beiden Gleisen gibt es ein Abstellgleis, das zum Rangieren benutzt werden kann. <i>Beispiel:</i> Autos an einer Ampel zur Zufahrtsregelung Es soll eine Ampel zur Zufahrtsregelung in Java simuliert werden. An einem geradlinigen, senkrecht von unten nach oben verlaufenden
--	--	---

		Straßenstück, das von Autos nur einspurig in eine Richtung befahren werden kann, ist ein Haltepunkt markiert, an dem die Ampel steht. Bei einem Klick auf eine Schaltfläche mit der Aufschrift „Heranfahren“ soll ein neues Auto an den Haltepunkt heranfahren bzw. bis an das letzte Auto, das vor dem Haltepunkt wartet. Grünphasen der Ampel werden durch einen Klick auf eine Schaltfläche mit der Aufschrift „Weiterfahren“ simuliert. In jeder Grünphase darf jeweils nur ein Auto weiterfahren. Die anderen Autos rücken nach. <i>Materialien:</i> Ergänzungsmaterialien zum Lehrplannavigator Unterrichtsvorhaben Q1-II.3 – Anwendungen für lineare Datenstrukturen (Download Q1-II.3)
--	--	---

Unterrichtsvorhaben Q1-III:

Thema: Suchen und Sortieren auf linearen Datenstrukturen

Leitfrage: *Wie kann man gespeicherte Informationen günstig (wieder-)finden?*

Vorhabenbezogene Konkretisierung:

In einem Anwendungskontext werden zunächst Informationen in einer linearen Liste bzw. einem Feld gesucht. Hierzu werden Verfahren entwickelt und implementiert bzw. analysiert und erläutert, wobei neben einem iterativen auch ein rekursives Verfahren thematisiert wird und mindestens ein Verfahren selbst entwickelt und implementiert wird. Die verschiedenen Verfahren werden hinsichtlich Speicherbedarf und Zahl der Vergleichsoperationen miteinander verglichen.

Anschließend werden Sortierverfahren entwickelt und implementiert (ebenfalls für lineare Listen und Felder). Hierbei soll auch ein rekursives Sortierverfahren entwickelt werden. Die Implementationen von Quicksort sowie dem Sortieren durch Einfügen werden analysiert und erläutert. Falls diese Verfahren vorher schon entdeckt wurden, sollen sie hier wiedererkannt werden. Die rekursive Abarbeitung eines Methodenaufrufs von Quicksort wird grafisch dargestellt.

Abschließend werden verschiedene Sortierverfahren hinsichtlich der Anzahl der benötigten Vergleichsoperationen und des Speicherbedarfs beurteilt.

Zeitbedarf: 16 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Suchen von Daten in Listen und Arrays (a) Lineare Suche in Listen und in Arrays (b) Binäre Suche in Arrays als Beispiel für rekursives Problemlösen (c) Untersuchung der beiden Suchverfahren hinsichtlich ihrer Effizienz (Laufzeitverhalten, Speicherbedarf)	Die Schülerinnen und Schüler • analysieren und erläutern Algorithmen und Programme (A), • beurteilen die syntaktische Korrektheit und die Funktionalität von Programmen (A),	<i>Beispiel:</i> Karteiverwaltung Für ein Adressverwaltungsprogramm soll eine Methode zum Suchen einer Adresse geschrieben werden. oder <i>Beispiel:</i> Bundesjugendspiele

	• 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 Sortierv Verfahren (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 iterative und rekursive Algorithmen umgangssprachlich und grafisch dar (D).	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 herausuchen können und später das gesamte Ergebnis nach gewissen Kriterien sortieren können. <i>Materialien:</i> Ergänzungsmaterialien zum Lehrplannavigator Unterrichtsvorhaben Q1.3 - Suchen und Sortieren (Download Q1-III.1)
2. Sortieren in Listen und Arrays - Entwicklung und Implementierung von iterativen und rekursiven Sortierv Verfahren (a) Entwicklung und Implementierung eines einfachen Sortierv Verfahrens für eine Liste (b) Implementierung eines einfachen Sortierv Verfahrens für ein Feld (c) Entwicklung eines rekursiven Sortierv Verfahrens für ein Feld (z.B. Sortieren durch Mischen)		<i>Beispiel:</i> Karteiverwaltung (s.o.) oder <i>Beispiel:</i> Bundesjugendspiele (s.o.) <i>Materialien:</i> (s.o.)

3. Untersuchung der Effizienz der Sortierverfahren „Sortieren durch direktes Einfügen“ und „Quicksort“ auf linearen Listen (a) Grafische Veranschaulichung der Sortierverfahren (b) Untersuchung der Anzahl der Vergleichsoperationen und des Speicherbedarf bei beiden Sortierverfahren (c) Beurteilung der Effizienz der beiden Sortierverfahren		<i>Beispiel:</i> Karteiverwaltung (s.o.) oder <i>Beispiel:</i> Bundesjugendspiele (s.o.) <i>Materialien:</i> (s.o.)
---	--	---

Unterrichtsvorhaben Q1-IV:

Thema: Modellierung und Nutzung von relationalen Datenbanken in Anwendungskontexten

Leitfragen: *Wie können Fragestellungen mit Hilfe einer Datenbank beantwortet werden? Wie entwickelt man selbst eine Datenbank für einen Anwendungskontext?*

Vorhabenbezogene Konkretisierung:

Ausgehend von einer vorhandenen Datenbank entwickeln Schülerinnen und Schüler für sie relevante Fragestellungen, die mit dem vorhandenen Datenbestand beantwortet werden sollen. Zur Beantwortung dieser Fragestellungen wird die vorgegebene Datenbank von den Schülerinnen und Schülern analysiert und die notwendigen Grundbegriffe für Datenbanksysteme sowie die erforderlichen SQL-Abfragen werden erarbeitet.

In anderen Anwendungskontexten müssen Datenbanken erst noch entwickelt werden, um Daten zu speichern und Informationen für die Beantwortung von möglicherweise auftretenden Fragen zur Verfügung zu stellen. Dafür ermitteln Schülerinnen und Schüler in den Anwendungssituationen Entitäten, zugehörige Attribute, Relationen und Kardinalitäten und stellen diese in Entity-Relationship-Modellen dar. Entity-Relationship-Modelle werden interpretiert und erläutert, modifiziert und in Datenbankschemata überführt. Mit Hilfe von SQL-Anweisungen können anschließend im Kontext relevante Informationen aus der Datenbank extrahiert werden.

Ein Entity-Relationship-Diagramm kann auch verwendet werden, um die Entitäten inklusive ihrer Attribute und Relationen in einem vorgegebenen Datenbankschema darzustellen.

An einem Beispiel wird verdeutlicht, dass in Datenbanken Redundanzen unerwünscht sind und Konsistenz gewährleistet sein sollte. Die 1. bis 3. Normalform wird als Gütekriterium für Datenbankentwürfe eingeführt. Datenbankschemata werden hinsichtlich der 1. bis 3. Normalform untersucht und (soweit nötig) normalisiert.

Zeitbedarf: 20 Stunden

Sequenzierung des Unterrichtsvorhabens

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Nutzung von relationalen Datenbanken (a) Aufbau von Datenbanken und Grundbegriffe	Die Schülerinnen und Schüler	<i>Beispiel:</i> VideoCenter VideoCenter ist die Simulation einer Online-Videothek für den Informatik-Unterricht mit Webfrontends zur Verwaltung der Kunden, der

- Entwicklung von Fragestellungen zur vorhandenen Datenbank - Analyse der Struktur der vorgegebenen Datenbank und Erarbeitung der Begriffe Tabelle, Attribut, Datensatz, Datentyp, Primärschlüssel, Fremdschlüssel, Datenbankschema (b) 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) (c) Vertiefung an einem weiteren Datenbankbeispiel	- 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), - bestimmen Primär- und Sekundärschlüssel (M), - überführen Datenbankschemata in vorgegebene Normalformen (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	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/old/video/index.html (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. <i>Beispiel: Schulbuchausleihe</i> Unter www.brd.nrw.de/lerntreffs/informatik/structure/material/sek2/datenbanken.php (abgerufen: 30. 03. 2014) wird eine Datenbank zur Verfügung gestellt, die Daten einer Schulbuch-Ausleihe enthält (über 1000 Entleiher, 200 Bücher mit mehreren tausend Exemplaren und viele Ausleihvorgänge). Die Datenbank kann in OpenOffice eingebunden werden.
2. Modellierung von relationalen Datenbanken (a) Entity-Relationship-Diagramm Ermittlung von Entitäten, zugehörigen Attributen, Relationen und		<i>Beispiel: Fahrradverleih</i> Der Fahrradverleih <i>BTR (BikesToRent)</i> verleiht unterschiedliche Typen von Fahrrädern diverser Firmen an seine Kunden. Die Kunden sind bei <i>BTR</i> registriert (Name,

Kardinalitäten in Anwendungssituationen und Modellierung eines Datenbankentwurfs in Form eines Entity-Relationship-Diagramms • Erläuterung und Modifizierung einer Datenbankmodellierung (b) Entwicklung einer Datenbank aus einem Datenbankentwurf • Modellierung eines relationalen Datenbankschemas zu einem Entity-Relationship-Diagramm inklusive der Bestimmung von Primär- und Sekundärschlüsseln (c) 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)	einem Entity-Relationship-Diagramm grafisch dar (D), • überprüfen Datenbankschemata auf vorgegebene Normalisierungseigenschaften (D).	Adresse, Telefon). <i>BTR</i> kennt von den Fahrradfirmen den Namen und die Telefonnummer. Kunden von <i>BTR</i> können CityBikes, Treckingräder und Mountainbikes ausleihen. <i>Beispiel: Reederei</i> Die Datenverwaltung einer Reederei soll in einem Datenbanksystem umgesetzt werden. Ausgehend von der Modellierung soll mit Hilfe eines ER-Modells und eines Datenbankschemas dieser erste Entwurf normalisiert und in einem Datenbanksystem umgesetzt werden. Es schließen sich diverse SQL-Abfragen an, wobei auf die Relationenalgebra eingegangen wird. <i>Beispiel: Buchungssystem</i> In dem Online-Buchungssystem einer Schule können die Lehrer Medienräume, Beamer, Laptops, Kameras, usw. für einen bestimmten Zeitpunkt buchen, der durch Datum und die Schulstunde festgelegt ist. Dazu ist die Datenbank zu modellieren, ggf. zu normalisieren und im Datenbanksystem umzusetzen. Weiter sollen sinnvolle Abfragen entwickelt werden. Unter http://mrbs.sourceforge.net (abgerufen: 30.03. 2014) findet man ein freies Online-Buchungssystem inklusive Demo, an Hand derer man erläutern kann, worum es in dem Projekt geht.
--	--	---

		<i>Beispiel: Schulverwaltung</i> In einer Software werden die Schulhalbjahre, Jahrgangsstufen, Kurse, Klassen, Schüler, Lehrer und Noten einer Schule verwaltet. Man kann dann ablesen, dass z.B. Schüler X von Lehrer Y im 2. Halbjahr des Schuljahrs 2011/2012 in der Jahrgangsstufe 9 im Differenzierungsbereich im Fach Informatik die Note „sehr gut“ erhalten hat. Dazu ist die Datenbank zu modellieren, ggf. zu normalisieren und im Datenbanksystem umzusetzen. Weiter sollen sinnvolle Abfragen entwickelt werden und das Thema Datenschutz besprochen werden.
--	--	--

Unterrichtsvorhaben Q1-V:

Thema: Sicherheit und Datenschutz in Netzstrukturen

Leitfragen: *Wie werden Daten in Netzwerken übermittelt? Was sollte man in Bezug auf die Sicherheit beachten?*

Vorhabenbezogene Konkretisierung:

Anschließend an das vorhergehende Unterrichtsvorhaben zum Thema Datenbanken werden der Datenbankzugriff aus dem Netz, Topologien von Netzwerken, eine Client-Server-Struktur, das TCP/IP-Schichtenmodell sowie Sicherheitsaspekte beim Zugriff auf Datenbanken und verschiedene symmetrische und asymmetrische kryptografische Verfahren analysiert und erläutert. Fallbeispiele zur Datenschutzproblematik und zum Urheberrecht runden das Unterrichtsvorhaben ab.

Zeitbedarf: 10 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Daten in Netzwerken und Sicherheitsaspekte in Netzen sowie beim Zugriff auf Datenbanken (a) Beschreibung eines Datenbankzugriffs im Netz anhand eines Anwendungskontextes und einer Client-Server-Struktur zur Klärung der Funktionsweise eines Datenbankzugriffs (b) Netztopologien als Grundlage von Client-Server-Strukturen und TCP/IP-Schichtenmodell als Beispiel für eine Paketübermittlung in einem Netz (c) Vertraulichkeit, Integrität, Authentizität in Netzwerken sowie symmetrische	Die Schülerinnen und Schüler • 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	<i>Materialien:</i> Ergänzungsmaterialien zum Lehrplannavigator Unterrichtsvorhaben, Verschlüsselung Q1.5 - Zugriff auf Daten in Netzwerken (Download Q1-V.1)

und asymmetrische kryptografische Verfahren (Cäsar-, Vigenère-, RSA-Verfahren) als Methoden Daten im Netz verschlüsselt zu übertragen	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).	
2. Fallbeispiele zur Datenschutzproblematik und zum Urheberrecht		<i>Materialien:</i> <i>Ergänzungsmaterialien zum Lehrplannavigator Unterrichtsvorhaben Q1 5 - Datenschutz beim Videocenter, Materialblatt-Datenschutzgesetz (Download Q1-V.2)</i>

Unterrichtsvorhaben Q2-I:

Thema: Modellierung und Implementierung von Anwendungen mit dynamischen, nichtlinearen Datenstrukturen

Leitfragen: *Wie können Daten im Anwendungskontext mit Hilfe binärer Baumstrukturen verwaltet werden? Wie kann dabei der rekursive Aufbau der Baumstruktur genutzt werden? Welche Vor- und Nachteile haben Suchbäume für die geordnete Verwaltung von Daten?*

Vorhabenbezogene Konkretisierung:

Anhand von Beispielen für Baumstrukturen werden grundlegende Begriffe eingeführt und der rekursive Aufbau binärer Bäume dargestellt. Anschließend werden für eine Problemstellung in einem der Anwendungskontexte Klassen modelliert und implementiert. Dabei werden die Operationen der Datenstruktur Binärbaum thematisiert und die entsprechende Klasse BinaryTree (der Materialien für das Zentralabitur in NRW) der Vorgaben für das Zentralabitur NRW verwendet. Klassen und ihre Beziehungen werden in Entwurfs- und Implementationsdiagrammen dargestellt. Die Funktionsweise von Methoden wird anhand grafischer Darstellungen von Binärbäumen erläutert.

Unter anderem sollen die verschiedenen Baumtraversierungen (Pre-, Post- und Inorder) implementiert werden. Unterschiede bezüglich der Möglichkeit, den Baum anhand der Ausgabe der Bauminhalte via Pre-, In- oder Postorder-Traversierung zu rekonstruieren, werden dabei ebenfalls angesprochen, indem die fehlende Umkehrbarkeit der Zuordnung Binärbaum → Inorder-Ausgabe an einem Beispiel verdeutlicht wird.

Eine Tiefensuche wird verwendet, um einen in der Baumstruktur gespeicherten Inhalt zu suchen.

Zu einer Problemstellung in einem entsprechenden Anwendungskontext werden die Operationen der Datenstruktur Suchbaum thematisiert und unter der Verwendung der Klasse BinarySearchTree (der Materialien für das Zentralabitur in NRW) weitere Klassen oder Methoden in diesem Anwendungskontext modelliert und implementiert. Auch in diesem Kontext werden grafische Darstellungen der Bäume verwendet.

Die Verwendung von binären Bäumen und Suchbäumen wird anhand weiterer Problemstellungen oder anderen Kontexten weiter geübt.

Zeitbedarf: 24 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien, Materialien
1. Analyse von Baumstrukturen in verschiedenen Kontexten	Die Schülerinnen und Schüler	<i>Beispiel:</i> Termbaum Der Aufbau von Termen wird mit Hilfe von binären Baumstrukturen verdeutlicht.

(a) Grundlegende Begriffe (Grad, Tiefe, Höhe, Blatt, Inhalt, Teilbaum, Ebene, Vollständigkeit) (b) Aufbau und Darstellung von binären Bäumen anhand von Baumstrukturen in verschiedenen Kontexten	• 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 Herrschen“ (M), • implementieren iterative und rekursive Algorithmen auch unter Verwendung von dynamischen Datenstrukturen (I),	oder <i>Beispiel: Ahnenbaum</i> Die binäre Baumstruktur ergibt sich daraus, dass jede Person genau einen Vater und eine Mutter hat. <i>Weitere Beispiele für Anwendungskontexte für binäre Bäume:</i> <i>Beispiel: Suchbäume</i> (zur sortierten Speicherung von Daten) 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.) oder <i>Beispiel: Entscheidungsbäume</i> 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. oder
---	--	--

	• 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).	<i>Beispiel:</i> 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. <i>Materialien:</i> Ergänzungsmaterialien zum Lehrplannavigator Unterrichtsvorhaben Q2.1 – Binärbaum (Download Q2-1.1)
2. Die Datenstruktur Binärbaum im Anwendungskontext unter Nutzung der Klasse BinaryTree (a) Analyse der Problemstellung, Ermittlung von Objekten, ihren Eigenschaften und Operationen im Anwendungskontext (b) Modellierung eines Entwurfsdiagramms und Entwicklung eines Implementationsdiagramms		<i>Beispiel: Informatikerbaum als binärer Baum</i> In einem <i>binären Baum</i> 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:

(c) Erarbeitung der Klasse BinaryTree und beispielhafte Anwendung der Operationen (d) Implementierung der Anwendung oder von Teilen der Anwendung (e) Traversierung eines Binärbaums im Pre-, In- und Postorderdurchlauf		• Einfügen der Informatiker-Daten in den Baum • Suchen nach einem Informatiker über den Schlüssel Name • Ausgabe des kompletten Datenbestands in nach Namen sortierter Reihenfolge <i>Materialien:</i> Ergänzungsmaterialien zum Lehrplannavigator Unterrichtsvorhaben Q2.1 – Binärbaum (Download Q2-I.2)
3. Die Datenstruktur binärer Suchbaum im Anwendungskontext unter Verwendung der Klasse BinarySearchTree (a) Analyse der Problemstellung, Ermittlung von Objekten, ihren Eigenschaften und Operationen (b) Modellierung eines Entwurfsdiagramms und Entwicklung eines Implementationsdiagramm, grafische Darstellung eines binären Suchbaums und Erarbeitung der Struktureigenschaften (c) Erarbeitung der Klasse BinarySearchTree und Einführung des Interface Item zur Realisierung einer geeigneten Ordnungsrelation (d) Implementierung der Anwendung oder von Teilen der Anwendung inklusive einer sortierten Ausgabe des Baums		<i>Beispiel:</i> Informatikerbaum als <i>Suchbaum</i> In einem binären <i>Suchbaum</i> 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

		<i>Materialien:</i> Ergänzungsmaterialien zum Lehrplannavigator Unterrichtsvorhaben Q2.1 – Binärer Suchbaum (Download Q2-I.3)
4. Übung und Vertiefungen der Verwendung von Binärbäumen oder binären Suchbäumen anhand weiterer Problemstellungen		<i>Beispiel:</i> Codierungsbäume (s.o.) oder Huffman-Codierung <i>oder</i> <i>Beispiel:</i> Buchindex Es soll eine Anwendung entwickelt werden, die anhand von Stichworten und zugehörigen Seitenzahlen ein Stichwortregister erstellt. Da die Stichwörter bei der Analyse des Buches häufig gesucht werden müssen, werden sie in der Klasse Buchindex als Suchbaum (Objekt der Klasse BinarySearchTree) verwaltet. 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.) <i>oder</i> <i>Beispiel:</i> Entscheidungsbäume (s.o.) <i>oder</i>

		<i>Beispiel:</i> Termbaum (s.o.) <i>oder</i> <i>Beispiel:</i> Ahnenbaum (s.o.) <i>Materialien:</i> Ergänzungsmaterialien zum Lehrplannavigator Unterrichtsvorhaben Q2.1 – Anwendung Binärbaum (Download Q2-I.4)
--	--	---

Unterrichtsvorhaben Q2-II:

Thema: Endliche Automaten und formale Sprachen

Leitfragen: *Wie kann man (endliche) Automaten genau beschreiben? Wie können endliche Automaten (in alltäglichen Kontexten oder zu informatischen Problemstellungen) modelliert werden? Wie können Sprachen durch Grammatiken beschrieben werden? Welche Zusammenhänge gibt es zwischen formalen Sprachen, endlichen Automaten und regulären Grammatiken?*

Vorhabenbezogene Konkretisierung:

Anhand kontextbezogener Beispiele werden endliche Automaten entwickelt, untersucht und modifiziert. Dabei werden verschiedene Darstellungsformen für endliche Automaten ineinander überführt und die akzeptierten Sprachen endlicher Automaten ermittelt. An einem Beispiel wird ein nichtdeterministischer Akzeptor eingeführt als Alternative gegenüber einem entsprechenden deterministischen Akzeptor.

Anhand kontextbezogener Beispiele werden Grammatiken regulärer Sprachen entwickelt, untersucht und modifiziert. Der Zusammenhang zwischen regulären Grammatiken und endlichen Automaten wird verdeutlicht durch die Entwicklung von allgemeinen Verfahren zur Erstellung einer regulären Grammatik für die Sprache eines gegebenen endlichen Automaten bzw. zur Entwicklung eines endlichen Automaten, der genau die Sprache einer gegebenen regulären Grammatik akzeptiert.

Auch andere Grammatiken werden untersucht, entwickelt oder modifiziert. An einem Beispiel werden die Grenzen endlicher Automaten ausgelotet.

Zeitbedarf: 20 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien oder Materialien
1. Endliche Automaten (a) Vom Automaten in den Schülerinnen und Schülern bekannten Kontexten zur formalen Beschreibung eines endlichen Automaten (b) Untersuchung, Darstellung und Entwicklung endlicher Automaten	Die Schülerinnen und Schüler analysieren und erläutern die Eigenschaften endlicher Automaten einschließlich ihres Verhaltens auf bestimmte Eingaben (A),	<i>Beispiele:</i> Cola-Automat, Geldspielautomat, Roboter, Zustandsänderung eines Objekts „Auto“, Akzeptor für bestimmte Zahlen, Akzeptor für Teilwörter in längeren Zeichenketten, Akzeptor für Terme <i>Materialien:</i>

	• analysieren und erläutern Grammatiken regulärer Sprachen (A), • zeigen die Grenzen endlicher Automaten und regulärer Grammatiken im Anwendungszusammenhang auf (A), • ermitteln die formale Sprache, die durch eine Grammatik erzeugt wird (A), • entwickeln und modifizieren zu einer Problemstellung endliche Automaten (M), • 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), • 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).	Ergänzungsmaterialien zum Lehrplannavigator Unterrichtsvorhaben Q2.2 – Endliche Automaten, Formale Sprachen (Download Q2-II.1)
2. Untersuchung und Entwicklung von Grammatiken regulärer Sprachen (a) Erarbeitung der formalen Darstellung regulärer Grammatiken (b) Untersuchung, Modifikation und Entwicklung von Grammatiken (c) Entwicklung von endlichen Automaten zum Erkennen regulärer Sprachen die durch Grammatiken gegeben werden (d) Entwicklung regulärer Grammatiken zu endlichen Automaten		<i>Beispiele:</i> reguläre Grammatik für Wörter mit ungerader Parität, Grammatik für Wörter, die bestimmte Zahlen repräsentieren, Satzgliederungsgrammatik <i>Materialien: (s.o.)</i>
3. Grenzen endlicher Automaten		<i>Beispiele:</i> Klammerausdrücke, $a^n b^n$ im Vergleich zu $(ab)^n$

Unterrichtsvorhaben Q2-III:

Thema: Prinzipielle Arbeitsweise eines Computers und Grenzen der Automatisierbarkeit

Leitfragen: Was sind die strukturellen Hauptbestandteile eines Computers und wie kann man sich die Ausführung eines maschinenahen Programms mit diesen Komponenten vorstellen? Welche Möglichkeiten bieten Informatiksysteme und wo liegen ihre Grenzen?

Vorhabenbezogene Konkretisierung:

Anhand einer von-Neumann-Architektur und einem maschinennahen Programm wird die prinzipielle Arbeitsweise von Computern verdeutlicht. Ausgehend von den prinzipiellen Grenzen endlicher Automaten liegt die Frage nach den Grenzen von Computern bzw. nach Grenzen der Automatisierbarkeit nahe. Mit Hilfe einer entsprechenden Java-Methode wird plausibel, dass es unmöglich ist, ein Informatiksystem zu entwickeln, dass für jedes beliebige Computerprogramm und jede beliebige Eingabe entscheidet ob das Programm mit der Eingabe terminiert oder nicht (Halteproblem). Anschließend werden Vor- und Nachteile der Grenzen der Automatisierbarkeit angesprochen und der Einsatz von Informatiksystemen hinsichtlich prinzipieller Möglichkeiten und prinzipieller Grenzen beurteilt.

Zeitbedarf: 12 Stunden

Sequenzierung des Unterrichtsvorhabens:

Unterrichtssequenzen	Zu entwickelnde Kompetenzen	Beispiele, Medien oder Materialien
1. Von-Neumann-Architektur und die Ausführung maschinennaher Programme a) prinzipieller Aufbau einer von Neumann-Architektur mit CPU, Rechenwerk, Steuerwerk, Register und Hauptspeicher b) einige maschinennahe Befehle und ihre Repräsentation in einem Binär-Code, der in einem Register gespeichert werden kann	Die Schülerinnen und Schüler • 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).	<i>Beispiel:</i> Addition von 4 zu einer eingegeben Zahl mit einem Rechnermodell <i>Materialien:</i> Ergänzungsmaterialien zum Lehrplannavigator Unterrichtsvorhaben Q2.3 – Von-Neumann-Architektur und maschinennahe Programmierung (Download Q2-III.1)

c) Analyse und Erläuterung der Funktionsweise eines einfachen maschinen-nahen Programms		
2. Grenzen der Automatisierbarkeit a) Vorstellung des Halteproblems b) Unlösbarkeit des Halteproblems c) Beurteilung des Einsatzes von Informatiksystemen hinsichtlich prinzipieller Möglichkeiten und prinzipieller Grenzen		<i>Beispiel:</i> Halteproblem <i>Materialien:</i> Ergänzungsmaterialien zum Lehrplannavigator Unterrichtsvorhaben Q2.3 - Halteproblem (Download Q2-III.2)
Unterrichtsvorhaben Q2-IV: Wiederholung und Vertiefung ausgewählter Kompetenzen und Inhalte des ersten Jahrs der Qualifikationsphase		

Leistungsbewertung

I. Beurteilungsbereich schriftliche Leistungen/Klassenarbeiten

Schriftliche Arbeiten (Klassenarbeiten oder Projektarbeiten inkl. Dokumentation) dienen der Überprüfung der Lernergebnisse einer vorausgegangenen Unterrichtsreihe. Sie werden so angelegt, dass Sachkenntnisse und methodische Fertigkeiten nachgewiesen werden können. Im Umfang und Anforderungsniveau sind schriftliche Arbeiten abhängig von den kontinuierlich ansteigenden Anforderungen entsprechend dem Lehrplan.

Die Anzahl und Dauer der schriftlichen Arbeiten im Fach Informatik hat die Fachkonferenz im Rahmen der Vorgaben der APO–GOST wie folgt festgelegt:

Jahrgangsstufe	Arbeiten pro Schulhalbjahr	Dauer (in U-Stunden)
EF	1	2
Q1 (GK)	2	2
Q2.1 (GK)	2	3
Q2.2 (GK)	1 Klausur unter Abiturbedingungen	

Die Arbeiten werden mithilfe eines Punkterasters bewertet. Die Notengebung orientiert sich an folgendem Schema:

Note	ungenügend	mangelhaft	ausreichend	befriedigend	gut	sehr gut
Punkteanteil	0% – 19%	20% – 38%	39% – 54%	55% – 69%	70% – 84%	85% – 100%

II. Beurteilungsbereich „Sonstige Leistungen“:

Der Beurteilungsbereich „Sonstige Leistungen im Unterricht“ erfasst die im Unterrichtsgeschehen durch mündliche, schriftliche und praktische Beiträge erkennbare Kompetenzentwicklung der Schülerinnen und Schüler. Bei der Bewertung berücksichtigt werden die Qualität, die Quantität und die Kontinuität der Beiträge. Die Kompetenzentwicklung im Beurteilungsbereich „Sonstige Leistungen im Unterricht“ wird sowohl durch kontinuierliche Beobachtung während des Schuljahres (Prozess der Kompetenzentwicklung) als auch durch punktuelle Überprüfungen (Stand der Kompetenzentwicklung) festgestellt. Bei der Bewertung von Leistungen, die die Schülerinnen und Schüler im Rahmen von Partner- oder Gruppenarbeiten erbringen, kann der individuelle Beitrag zum Ergebnis der Partner- bzw. Gruppenarbeit einbezogen werden.

Mögliche Überprüfungsformen:

- Darstellungs- und Dokumentationsaufgaben
 - Beschreibung informatorischer Zusammenhänge
 - Auswertung von Sensoren
- Modellierungs- und Implementationsaufgaben
 - Erstellung von Quellcodes und Algorithmen
 - Fehlersuche und -korrektur in Algorithmen und Quellcodes
 - Ablaufpläne und Struktogramme
- Präsentationsaufgaben
 - Programme vorstellen und erläutern
 - Kurzvorträge und Präsentationen

- | |
|--|
| <ul style="list-style-type: none">• Begründungs- und Bewertungsaufgaben<ul style="list-style-type: none">○ Begründung der eigenen Vorgehensweise○ Stellungnahme zu Texten und Medienbeiträgen• Praktische Leistungen am Computer<ul style="list-style-type: none">○ Implementierung, Test und Anwendung von Informatiksystemen |
| Weitere Hinweise, Vereinbarungen und Absprachen: |
| <ul style="list-style-type: none">• Teilnahme Informatik-Biber• Teilnahme Jugendwettbewerb Informatik |