

- o **Hinweise zum Einsatz von Lehrwerken/ Materialien:**

kein Lehrwerk vorhanden

- o **Grundsätze der Leistungsbeurteilung:**

Mit mehr als 50 % gehen in die Zeugnisnoten stets die Unterrichtsbeiträge ein. Hierzu zählen z.B. die Mitarbeit im Unterricht (ausdrücklich auch kooperativen Lernformen), erbrachte Leistungen in Tests, Kurzvorträgen, eventuelle Umsetzung der Aufgabe in den dazugehörigen digitalen Medien oder weiteren Lernprodukte.

- o **Überblick über die Themen des Faches in den Sek I:**

in Sek I werden die Themenfelder der Fachanforderungen unterrichtet.

- o **Hinweise zum überfachlichen Kompetenzerwerb:**

Selbstkompetenzen wie Selbstwirksamkeit, Selbstreflexion und Ausdauer werden systematisch gefördert. Die Lernenden überprüfen ihre Arbeitsergebnisse, dabei entwickeln sie ihre Problemlösefähigkeiten sowie ihre Medienkompetenz weiter. Soziale Kompetenzen werden im Informatikunterricht der Sek I gezielt gefördert. Durch regelmäßig angelegte kooperative Arbeitsformen erwerben die Lernenden Fähigkeiten zur konstruktiven Zusammenarbeit sowie zu Toleranz und gegenseitigem Respekt.

- o **Hinweise zur Sprachbildung:**

Sprachbildung in der Sek I ist ein integraler Bestandteil des Informatikunterrichts. Sie umfasst die sachgerechte Verwendung fachspezifischer Terminologie. Durch mündliche und schriftliche Präsentationen informatischer Inhalte sowie durch die Teilnahme an fachbezogenen Diskussionen entwickeln die Schülerinnen und Schüler die Fähigkeit, eigene Ideen, Lösungsansätze und fachliche Kenntnisse adressatengerecht, präzise und strukturiert zu kommunizieren.

Jahrgangsstufe: 7

○ Hinweise zu außerschulischen Lernorten und Projekten bzw. Wettbewerben:

Die jährlich stattfindende Code Week bietet den SuS auf freiwilliger Basis die Möglichkeit an Workshops teilzunehmen.

○ Hinweise zur Leistungsüberprüfung:

Im Laufe des Schuljahres ist eine schriftliche Leistungsnachweise in Form einer Klassenarbeit zu erbringen. Diese kann ganz oder teilweise mit Hilfe eines Computers geschrieben werden. Weitere Leistungsnachweise sind in unterschiedlichen Formen möglich. Diese umfassen u.a. Tests, Gruppen- oder Projektarbeiten, Klassenarbeit oder gleichwertige Lernleistung, Kontrolle des Arbeitsfortschritts, Abgabe von Produkten.

Themen und Inhalte	Kompetenzbezug, didaktische Prinzipien	Mögliche Methoden	Mögliche analoge und digitale Medien
Einführung in den Informatikunterricht - Was ist Informatik? - Regeln im Fachraum - Anmeldung bei Lernplattformen - Ordnerstrukturen und Dateien	- Strategien zum Vermeiden von Datenverlust - Wenden typische Operationen auf Dateien an - Entwerfen zu einem Verwendungszweck passende Verzeichnisstrukturen und ordnen Dateien systematisch in diese ein.	<i>Dateien von Itslearning herunterladen, umbenennen, speichern, verschieben</i>	ItsLearning lerv
Informatiksysteme I - Bandbreite der Informatiksysteme - Allgegenwärtigkeit von Informatiksystemen - EVA-Prinzip - Hardwarekunde	- beschreiben einfache und komplexe Informatiksysteme und deren Einsatzbereich - nennen Anwendungsbereiche für verschiedene persönliche Geräte - nennen Hardwarekomponenten und ihre Funktion - erklären die grundlegende Funktionsweise von Hardwarekomponenten und deren Zusammenwirken	- Definition und Sammlung von Informatiksystemen - Anwendung und Beschreibung des EVA Prinzips auf typische Aktionen am digitalen Endgerät - Betrachten und Zuordnen der Bauteile im inneren eines Tower-PCs	
Textverarbeitung I Strukturelemente	untersuchen Textdokumente hinsichtlich Struktur und Format	Analysieren, verändern und erstellen von Textdokumenten	Libre Office

• Steuerzeichen • Trennung von Struktur und Layout • Tabellen und Grafiken • Verweise • automatische Gliederung	• entwickeln aus einer Problemstellung eine passende Dokumentstruktur • verwenden Formatvorlagen zweckmäßig und sparsam		
Daten und Information I • Morsecode, ISBN, Gebärdensprache .. • Binärsystem • Bits/Bytes, ASCII • Graustufen- und RGB Modell • Dateigröße in Abhängigkeit von Pixelmodell und Auflösung • Kompressionsverfahren • Farbtiefe	• reduzieren Informationen auf ihren relevanten Anteil • interpretieren Daten im Speicher als Zahlen, Zeichen, Wahrheitswerte oder Grafiken • beschreiben die binäre Repräsentation von Zeichen und ganzen Zahlen • verwenden Kompressions- und Archivierungswerkzeuge zur effizienten Speicherung von Daten • klassifizieren verlustfreie und verlustbehaftete Kompression • untersuchen und bearbeiten Rastergrafiken im Hinblick auf ein Präsentationsziel • beschreiben Merkmale von Vektor- und Rastergrafiken sowie deren Anwendungsbereiche	• Daten codieren und decodieren • Vorträge zu verschiedenen Codierungsverfahren • Rechnen mit Binärzahlen	Codierungstabellen
Algorithmen I • Grundlagen der Programmierung in Scratch (elementare Anweisung, Sequenz, Kontrollstrukturen, Variablen), Nachrichten	• nennen und beschreiben Algorithmen aus dem Alltag • Überführen Algorithmen aus dem Alltag in konkrete Handlungen • formulieren Handlungsvorschriften unter Nutzung algorithmischer Grundbausteine • interpretieren und kommentieren einfache Algorithmen in einer grafischen Programmierungsumgebung • beschreiben sowohl für einzelne Anweisungen wie auch für Algorithmen im Ganzen das Ergebnis der Ausführung	• Algorithmus in Textform verfassen (Kochrezept) • vorgegebene Algorithmen analysieren und modifizieren • vorgegebene Algorithmen beschreiben • vorgegebene Algorithmen erweitern • Fehler finden und beheben • eigene Projektarbeit	Scratch

	• implementieren einfache Algorithmen in einer grafischen Programmierumgebung • wenden das Variablenkonzept an • entwerfen und implementieren Algorithmen zur Lösung einer gegebenen Problemstellung • wenden Techniken zur schrittweisen Prüfung der Korrektheit von Algorithmen an • entwerfen, implementieren und testen (eigene) Protokolle an Anwendungsbeispielen		
--	---	--	--

Jahrgangsstufe: 8

○ Hinweise zu außerschulischen Lernorten und Projekten bzw. Wettbewerben:

Die jährlich stattfindende Code Week bietet den SuS auf freiwilliger Basis die Möglichkeit an Workshops teilzunehmen

○ Hinweise zur Leistungsüberprüfung:

Im Laufe des Schuljahres ist ein alternativer Leistungsnachweise in Form eines Projekts zu erbringen. Weitere Leistungsnachweise sind in unterschiedlichen Formen möglich. Diese umfassen u.a. Tests, Gruppen- oder Projektarbeiten, Klassenarbeit oder gleichwertige Lernleistung, Kontrolle des Arbeitsfortschritts, Abgabe von Produkten

Themen und Inhalte	Kompetenzbezug, didaktische Prinzipien	Mögliche Methoden	Mögliche analoge und digitale Medien
Informatiksysteme II • Arbeit mit dem Calliope • Sensoren und Eingabegeräte • Aktoren und Ausgabegeräte	• konstruieren Informatiksysteme mithilfe von Mikrocontroller-Boards oder Robotern	• vorgegebene Algorithmen analysieren und modifizieren • vorgegeben Algorithmen beschreiben • vorgegebene Algorithmen erweitern • Fehler finden und beheben • eigene Projektarbeit	Calliope
Textverarbeitung II • Erstellung von Seiten mithilfe von HTML • Recht am eigenen Bild • Eigentumsrechte (z.B. Creative Commons Lizenz)	• Untersuchen Textdokumente hinsichtlich Struktur und Format • Entwickeln aus einer Problemstellung eine passende Dokumentstruktur • Verwenden Formatvorlagen zweckmäßig und sparsam • Analysieren geistiges Eigentum auf freie Verwendbarkeit	• Umwandeln der HTML-Struktur in ein Textdokument in Libre Office • Erstellen einer vorgegeben Seite in HTML • Fehlersuche • Freies Projekt	HTML-Editor
Tabellenkalkulation • Kalkulationstabellen • Attribute und Werte • Datentypen • Werte und Ausdrücke • Bezüge (relativ und absolut) • Diagramme	• entwickeln aus einer Problemstellung eine passende Tabellenstruktur • verwenden Ausdrücke zur Auswertung von Daten • überführen Daten in eine andere Repräsentation		Libre Office

Netzwerke und Internet I • Authentifizierung und Autorisierung • Passwortsicherheit	• Beurteilen Situationen, in denen persönliche Daten erhoben, gespeichert und weitergegeben werden • Beurteilen die scheinbare Anonymität im Internet • Nennen und beurteilen Schutzmechanismen persönlicher Geräte • Sicherheit, Kryptographie, Caesar Verschlüsselung • Sicherheit von Passwörtern	• Präsentationen • Nutzungstagebuch • Passwortsicherheit überprüfen/Passwortanalyse • Erstellen von Material zur Verschlüsselung (Caesar-Scheibe.)	Verschlüsselungsmaterial
Netzwerk und Internet II • Kommunikationsmodell • Kommunikationsregeln • Lokale und globale Netzwerke • IP Adresse • URL • Client-Server • Netzwerktopologien	• verwenden grundlegende Funktionen des Betriebssystems zur Bewältigung typischer Aufgaben • erklären die Bestandteile eines allgemeinen Kommunikationsprozesses • erläutern das Internet als Verbund von Netzwerken • erklären die Adressierung im Internet • beschreiben Webanwendungen und ihre Architektur • erläutern wichtige Internetdienste	• Simulation eines Netzwerks mit Filius • Rollenspiel	Filius
KI • Benutzung von KI-Tools • Arten des maschinellen Lernens • Gesellschaftliche Aspekte	• Chancen und Risiken künstlicher Intelligenz • Erproben verschiedene KI-Tools und Verwendbarkeit prüfen • Kennenlernen Prinzipien des maschinellen Lernen	• Prompten mit Fobizz	Fobizz-KI

Jahrgangsstufe: 9 – Wahlpflichtunterricht

○ Hinweise zu außerschulischen Lernorten und Projekten bzw. Wettbewerben:

Besuch von oder bei Nordakademie / FH Wedel, Codeweek denkbar

○ Hinweise zur Leistungsüberprüfung:

Tests, Gruppen- oder Projektarbeiten, Klassenarbeit oder gleichwertige Lernleistung, Kontrolle des Arbeitsfortschritts.

Themen und Inhalte	Kompetenzbezug, didaktische Prinzipien	Mögliche Methoden	Mögliche analoge und digitale Medien
Bildbearbeitung	- Nutzung einer Bildbearbeitungssoftware - Arten und Zweck von Bildbearbeitung - Bilddateiformate und deren Nutzung - Manipulation mit Bildern	- Schritt-für-Schritt Anleitungen - Recherche und Präsentation von Beispielen und Methoden - Eigene Fotoprojekte	GIMP
Strukturierte Darstellung von Texten im Internet	- Erstellen einer Homepage erweitert mit CSS - Einbindung fremder und eigener Bilder	- Forschen mit den Inspektor - Lernpfad Designsteuerung CSS - Eigene thematische Website	- Internetbrowser (inkl. Inspektor) - Notepad++
Medienrecht und Mediensicherheit	- Veröffentlichung von Medien - Recht am eigenen Bild - Urheberrecht - CC-Lizenzen	- Recherche und Präsentation - Entscheidungsbäume / Infolyer erstellen - Einzelfallbeurteilungen	

• Textuelle Programmierung	• Lineares textuelles Programmieren in einfacher Simulationsumgebung	• Nachbauen und Erproben • Analyse von Programmen • Prozesse planen und umsetzen	• RobotKarol
• Strukturierte Darstellung von Algorithmen	• Struktogramme und Programmablaufpläne lesen lernen, • Alltagsalgorithmen strukturiert darstellen • planen und übertragen von Algorithmen in / aus Struktogrammen • Modularisierung, Auslagerung von Unterprogrammen	• Prozesse der Programmierung und Programmentwicklung erproben • zeichnerische Darstellungen (analog & digital) • kollaborative Gruppenarbeiten	• RobotKarol (Formatierung & Struktogrammdarstellung)
• Netzwerke und Internet	• Vertiefung des Wissens über Netzwerkstrukturen und Datenaustausch im Internet	• Paketübermittlung verstehen • Austausch über lokale Netzwerkgrenzen • Infrastruktur nachbauen	• Filius

Jahrgangsstufe: 10 – Wahlpflichtunterricht

○ Hinweise zu außerschulischen Lernorten und Projekten bzw. Wettbewerben:

Besuch von oder bei Nordakademie / FH Wedel, Codeweek denkbar

○ Hinweise zur Leistungsüberprüfung:

Tests, Gruppen- oder Projektarbeiten, Klassenarbeit oder gleichwertige Lernleistung, Kontrolle des Arbeitsfortschritts.

Themen und Inhalte	Kompetenzbezug, didaktische Prinzipien	Mögliche Methoden	Mögliche analoge und digitale Medien
• Audiotbearbeitung	• Nutzung einer Audiotbearbeitungssoftware • Sounddateiformate und deren Nutzung • Medienproduktion	• Zentrale Einführung in die Software • Recherche und Erstellung nutzbarer Medien (mit Wdh. rechtlicher Fragen) • Tagesablauf mit Geräuschen schneiden • Eigenes Podcastprojekt	• Audacity
• Akteur-Sensor-Systeme	• Software für vorhandene Hardware entwickeln • Steuerung Mikrocontroller • autonome Robotersysteme	• Sensorik und Motorik erproben • Nachbau von Programmen • Messwerteerfassung • Fehleranalyse • Partnerprojekte	• Calliope mini mit Callibot/ Mindstorms NXT • OpenRoberta Lab oder MakeCode
• Rund um Daten	• Datenmengen und Datenströme (BigData) • Datenspeicherung, -nutzung, -sicherheit (SmartHome)	• Eigene Nutzung erkennen und hinterfragen • BigData und Folgen erforschen •	

• Textuelle Programmierung II	• Lineares textuelles Programmieren	• Schritt-für-Schritt Anleitungen • von Geometrie Anwendung (Turtlegrafik) zu komplexen Problemlösungen • Integration erlernter algorithmischer Grundstrukturen	• FMS Logo
---	---	---	--

- **Hinweise zum Einsatz von Lehrwerken/ Materialien:**

kein Lehrwerk vorhanden

- **Grundsätze der Leistungsbeurteilung:**

Mit mehr als 50 % gehen in die Zeugnisnoten stets die Unterrichtsbeiträge ein. Hierzu zählen z.B. die Mitarbeit im Unterricht (ausdrücklich auch kooperativen Lernformen), erbrachte Leistungen in Tests, Kurzvorträgen, eventuelle Umsetzung der Aufgabe in den dazugehörigen digitalen Medien (Fillius, Python, Instahub, JFLAP, networkx, draw.io...) oder weiteren Lernprodukten. Die Lehrkraft entscheidet unter Berücksichtigung der eigenen Unterrichtsgestaltung über die Gewichtungen der Leistungsbestandteile.

Es werden zwei schriftliche Leistungsnachweise in Form von Klausuren im Laufe des Schuljahres geschrieben (Einsatz des PCs möglich).

- **Überblick über die Themen des Faches in den Qualifikationsphase:**

Eine kurze Wiederholung der Grundlagen des Informatikunterrichts stellt die formalen Voraussetzungen für die erfolgreiche Mitarbeit sicher.

- **Einführungsphase:**

Betriebssysteme vermitteln zwischen der Hardware und der Anwendung durch den Benutzer. Die Speicherung und Repräsentation von Daten ist die Grundlage jeglicher Informationsverarbeitung. Durch die Verwendung von Algorithmen und textueller Programmierung entwickeln und trainieren die Lernenden informatorisches Denken (*Computational Thinking*). Die Kommunikation in Netzwerken und Internet ist im Alltag allgegenwärtig. Dabei spielt insbesondere die Sicherheit in Kommunikationssystemen eine große Rolle.

- **Qualifikationsphase:**

Die Themen der Q1 und Q2 bilden die Grundlage für das Verständnis und die Gestaltung der digitalen Infrastruktur, die unser tägliches Leben prägt. Datenbanken ermöglichen die effiziente Speicherung und Verwaltung großer Datenmengen. Algorithmen sind essenziell

für die Lösung komplexer Probleme. Programmierung befähigt dazu, eigene Anwendungen zu entwickeln, während Softwareentwicklung den Prozess von der Idee zum funktionierenden Produkt beschreibt. Die Vermittlung dieser Themen fördert kritisches Denken, Problemlösungsfähigkeiten und Kreativität, die in einer technologiegeprägten Welt unerlässlich sind.

- **Hinweise zum überfachlichen Kompetenzerwerb:**

Selbstkompetenzen wie Selbstwirksamkeit, Selbstreflexion und Ausdauer werden systematisch gefördert. Lernmethodische Kompetenzen bilden eine zentrale Grundlage für erfolgreiches, selbstgesteuertes Lernen. Die Lernenden überprüfen ihre Arbeitsergebnisse, reflektieren Lernprozesse und nehmen bei Bedarf Korrekturen vor. Dabei entwickeln sie nachhaltige Lernstrategien, Problemlösefähigkeiten sowie ihre Medienkompetenz weiter. Soziale Kompetenzen werden im Informatikunterricht der Qualifikationsphase gezielt gefördert. Durch regelmäßig angelegte kooperative Arbeitsformen erwerben die Lernenden Fähigkeiten zur konstruktiven Zusammenarbeit sowie zu Toleranz und gegenseitigem Respekt.

- **Hinweise zur Sprachbildung:**

Sprachbildung in der Qualifikationsphase ist integraler Bestandteil des Informatikunterrichts. Sie umfasst die sachgerechte Verwendung fachspezifischer Terminologie sowie das Verstehen und Produzieren fachbezogener Texte. Durch mündliche und schriftliche Präsentationen informatischer Inhalte sowie durch die Teilnahme an fachbezogenen Diskussionen entwickeln die Schülerinnen und Schüler die Fähigkeit, eigene Ideen, Lösungsansätze und fachliche Kenntnisse adressatengerecht, präzise und strukturiert zu kommunizieren.

Jahrgangsstufe: 11 (Einführungsphase)

○ Hinweise zu außerschulischen Lernorten und Projekten bzw. Wettbewerben:

Als mögliche außerschulische Lernorte kommen Hochschulen wie die Fachhochschule Wedel oder die Nordakademie in Betracht. Sie bieten Einblicke in informatische Studiengänge, wissenschaftliches Arbeiten sowie praxisnahe Anwendungen der Informatik.

○ Hinweise zur Leistungsüberprüfung:

Im Laufe des Schuljahres sind mindestens zwei schriftliche Leistungsnachweise in Form von Klausuren zu erbringen. Diese können ganz oder teilweise mit Hilfe eines Computers geschrieben werden. Weitere Leistungsnachweise sind in unterschiedlichen Formen möglich.

Themen und Inhalte	Kompetenzbezug, didaktische Prinzipien	Mögliche Methoden	Mögliche analoge und digitale Medien
• Grundlagen des Informatikunterrichts	• siehe Sek.I	• Stationsarbeit [Regeln im Fachraum; Anmeldung bei Lernplattformen; Ordnerstrukturen; Dateibenennung etc.]	
• Betriebssysteme	• beschreiben die Funktionen eines Betriebssystems. • beschreiben die Implementierung von Dateien und Verzeichnissen in einem Dateisystem. • beschreiben Prozesse, deren Bestandteile und Eigenschaften.	• Untersuchungen des Betriebssystems der Schulcomputer	• Geräte- und Task-Manager
• Speicherung und Repräsentation von Daten	• beschreiben die binäre Repräsentation von Daten. • überführen Informationen in eine Datenrepräsentation im Speicher.	• Geschichte der Computer • Zahlensysteme (Binär-, Hexadezimal); Ganzzahlen im Zweierkomplement; Festkommazahlen; Bilder (s/w-, Graustufen, RGB-Farbbilder); Metadaten	• Besuch eines Computermuseums • Sammlung verschiedener (historischer) Speichermedien

• Algorithmen [Kriterien für Algorithmen; Variablenkonzept; Werte: Wahrheitswerte, Strings und Integer; Kontrollstrukturen: Anweisung, Sequenz, bedingte Anweisung bzw. Verzweigungen, Schleife; logische Verknüpfungen: AND, OR, NOT]	• beschreiben sowohl für einzelne Anweisungen wie auch für Algorithmen im Ganzen das Ergebnis der Ausführung. • formulieren Handlungsvorschriften unter Nutzung algorithmischer Grundbausteine.	• Programmierung von Mitschüler*innen • Programmablaufplan (PAP) • Struktogramme (NSD) • Berechnungen mit einer Stack-Machine	• Programmiersprache <i>Python</i> • Kommandozeileninterpreter z.B. <i>Anaconda</i> • Programmierumgebung z.B. <i>Spyder</i>
• Textbasierte Programmierung [Einführung in die Programmierumgebung; Testen von Programmen; Grenzfälle; Funktionen und Methoden; Such- und Sortieralgorithmen]	• interpretieren und kommentieren einfache Algorithmen in einer geeigneten Programmierumgebung. • erläutern Standardalgorithmen.	• Programmierung von TicTacToe • Programmtabellen • Such- und Sortieralgorithmen	• Programmiersprache <i>Python</i> • Programmierumgebung z.B. <i>Spyder</i>
• Kommunikation in Netzwerken und Internet [Simulation von Netzwerken, Internet als Verbund an Netzwerken, IP-Adresse; Gateway, DNS; URL; Protokolle: OSI-Schichtenmodell, HTTP(S); E-Mail-Konten; Vergleich von Netzwerken; Sicherheit in Netzwerken]	• erläutern den Netzbetrieb anhand eines Schichtenmodells. • erläutern das Internet als Verbund von Netzwerken. • beschreiben Webanwendungen und ihre Architektur.	• Sequenzdiagramme • Routertabellen	• <i>Filius</i>-Software

• Sicherheit in Kommunikationssystemen [Sicherheitsziele, Risiken und Abwehrmaßnahmen; Zertifikate; asymmetrische Verschlüsselung; öffentliche und private Schlüssel; Signaturen]	• diskutieren Sicherheitsrisiken und Abwehrmaßnahmen in Netzwerken. • beschreiben und beurteilen Authentifizierungsverfahren • diskutieren die Folgen netzbasierter Kommunikation für Entwicklungen in Politik und Gesellschaft.	• Personenbezogene Daten, DGSVO • Creative-Commons-Lizenzen • Hash-Funktionen • sicher(er)e digitale Kommunikationsmöglichkeiten • Aufbau des, Kriminalität und politische Opposition im Darknet • Diskussion zur Verwendung von <i>Palantir</i> durch Sicherheitsbehörden	• Film: <i>The Imitation Game – ein streng geheimes Leben</i> • Film: <i>The Great Hack</i> (Skandal um <i>Cambridge Analytica</i>)
--	--	---	--

Jahrgangsstufen: 12 und 13 (Qualifikationsphase)

○ Hinweise zu außerschulischen Lernorten und Projekten bzw. Wettbewerben:

Als mögliche außerschulische Lernorte kommen Hochschulen wie die Fachhochschule Wedel oder die Nordakademie in Betracht. Sie bieten Einblicke in informatische Studiengänge, wissenschaftliches Arbeiten sowie praxisnahe Anwendungen der Informatik.

○ Hinweise zur Leistungsüberprüfung:

Im Laufe des Schuljahres sind mindestens zwei schriftliche Leistungsnachweise in Form von Klausuren zu erbringen. Diese können ganz oder teilweise mit Hilfe eines Computers geschrieben werden. Weitere Leistungsnachweise sind in unterschiedlichen Formen möglich.

Themen und Inhalte	Kompetenzbezug, didaktische Prinzipien	Mögliche Methoden	Mögliche analoge und digitale Medien
Datenbankensysteme (Kalkulationstabellen, Datenbankabfragen, UML-Klassendiagramm, Redundanz und Anomalien)	- erläutern den Aufbau relationaler Datenbanken unter Verwendung der Begriffe Datensatz, Attribut, Primärschlüssel, Fremdschlüssel und Tabelle - nennen und erläutern Beispiele für Einfüge-, Änderungs- und Löschanomalien - untersuchen ein gegebenes Datenbankschema auf Anomalien und Redundanzen - formulieren Abfragen und Verbundabfragen über mehrere Tabellen - kennen die Unterscheidung zwischen funktionalen und transitiven Abhängigkeiten - diskutieren die Chancen und Risiken der automatisierten Datenanalyse	- SQL-Abfragen mit select, from, where, joins, sodass ein Selektieren, Aggregieren, Zusammenführen und Transformieren von Daten geschehen kann - Datenbankanwendung von Instahub <i>Querschnittsbereich Informatik, Mensch und Gesellschaft wird eingebunden</i>	Instahub
Algorithmen - entwerfen und implementieren	- erläutern Standardsuchalgorithmen der Komplexitätsklassen $O(n)$ und $O(\log(n))$ - erläutern Sortieralgorithmen der Komplexitätsklassen $O(n^2)$ und $O(n \cdot \log(n))$ - beurteilen Algorithmen zum Lösen eines Problems (Korrektheit, Effizienz, Laufzeit, Speicherbedarf) - beurteilen die Praktischen Grenzen der Algorithmierung	- Selektion-Sort, Quicksort - Rekursion über vertraute Handlungen erklären (Fakultät, Countdown, Türme von Hanoi) - Zentrale Begriffe: rekursiver Aufruf, Abbruchsbedingung	Python

	- entwerfen und implementieren rekursive Algorithmen, erläutern das Konzept Rekursion an gegebenen Beispielen - wenden bei der Implementierung von Algorithmen geeignete algorithmische Strategien an	Greedy	
Datenstrukturen - Objektorientiertes Programmieren	- beschreiben Basiskonzepte der objektorientierten Modellierung (Klasse, Objekt, Attribut, Methode, Vererbung, Kapselung) - verwenden vorgegebene Klassen und Objekte bei der Programmierung	- Grundbegriffe mit Alltagsobjekt-Methode einführen (Klassen Auto, Katze, Bankkonto) - Klassen und ihre Beziehungen visuell darstellen (UML-Diagramme) - schrittweise Programmierbeispiele in Python einführen <i>Projekte wie Winterlandschaft (fallende Schneeflocken) oder ideales Gas erweitern die Programmierfähigkeiten und das Verständnis in Bereich der OOP</i> <i>Querschnittsbereich Informatik, Mensch und Gesellschaft wird eingebunden</i>	Python Pygame
Datenstrukturen lineare Datenstrukturen: Array, Stapel, Schlange, lineare Liste nicht lineare Datenstrukturen: Binärbaum, binärer Suchbaum	- beschreiben Szenarien und Verwendungszwecke für Wörterbücher - führen Operationen auf Listen und Bäumen aus - beurteilen die Unterschiede von Listen und Bäumen hinsichtlich der Effizienz - implementieren Listen und Bäume als Klassen und verwenden diese in einem Anwendungsbeispiel	- Wörterbuch als abstrakter Datentyp mit den Methoden Einfügen, Löschen, Suchen - Listen und Bäume als Implementation von Wörterbüchern - Graphentheorie: Königsberger Brückenproblem, Soziale	Python, networkx

		Netzwerke, Straßenkarten, Travelling Salesman Problem (TSP), Dijkstra-Algorithmus, Minimaler Spannbaum (MST)	
• Endliche Automaten und formale Sprachen	• beschreiben die Automaten im Alltag • erläutern die reguläre Sprachen, reguläre Grammatiken • beschreiben deterministischer und nichtdeterministischer endlicher Automat		JFLAP
• Big Data & KI	• bewerten den Einfluss von Algorithmen auf Entscheidungen, Entscheidungsbäume • analysieren einfache künstliche neuronale Netze • erläutern typische Strategien maschinellen Lernens: überwachtes und verstärkendes Lernen	• Bedeutsame Situationen, in denen Maschinen Entscheidungen treffen z.B. autonomen Fahren <i>Querschnittsbereich Informatik, Mensch und Gesellschaft wird eingebunden</i>	