



Schulcurriculum Informatik

Sekundarbereich II

Stand Juni 2024

Vorbemerkungen

Unterrichtsumfang

Der Unterricht wird in der Einführungsphase zweistündig und in der Qualifikationsphase fünfstündig erteilt.

Leistungsbewertung

Die folgende Tabelle zeigt die Anzahl und den Umfang, der in Informatik zu schreibenden Klausuren sowie die Gewichtung der schriftlichen zu den sonstigen Leistungen:

| | 11.1 | 11.2 | 12.1 | 12.2 | 13.1 | 13.2 |
|--|----------------|----------------|-------------|-------------|--------------|---------------|
| Anzahl der Klausuren und Umfang | 1 60-90 min | 1 60-90 min | 1 90 min | 2 90 min | 1 210 min | 1* 300 min |
| Gewichtung der schriftlichen zu den sonstigen Leistungen | 40:60 | 40:60 | 40:60 | 50:50 | 50:50 | 50:50 |

*Klausur unter abiturähnlichen Bedingungen

Für die Bewertung der Klausuren wird der Bewertungsschlüssel aus dem Abitur verwendet:

| | | | | | | | | | | | | | | | |
|------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| Ab Prozent | 95 | 90 | 85 | 80 | 75 | 70 | 65 | 60 | 55 | 50 | 45 | 40 | 33 | 27 | 20 |
| Punkte | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 |

Die sonstigen Leistungen umfassen u. A.

- Beiträge zum Unterrichtsgespräch,
- mündliche Überprüfungen,
- Unterrichtsdokumentationen,
- Anwenden fachspezifischer Methoden und Arbeitsweisen,
- Präsentationen,
- Ergebnisse von Partner- oder Gruppenarbeiten und deren Darstellung,
- Projekt- und Langzeitaufgaben,
- Schülerwettbewerbe (z. B. Bundeswettbewerb Informatik)

Lehrbuch und verwendete Software

In der Qualifikationsphase wird das Schulbuch „Informatik 2“ (Schöningh-Verlag) eingesetzt. Bei der im Informatikunterricht verwendeten Software handelt es sich um freie Software, die kostenfrei genutzt werden darf und den Schüler:innen damit auch zu Hause zur Verfügung steht.

Einführungsphase

11.1

| Stunden | Thema | Erworbene Kompetenzen Die Schüler:innen ... | mögliches Material/ Software/Ideen | Bezug zum KC Die Schüler:innen ... |
|---|---|--|---|---|
| Webgestaltung (ca. 7 Wochen) | | | | |
| 2 | Aufrufen von Webseiten (tw. Wdhlg) | <ul style="list-style-type: none"> • beschreiben den Aufruf einer Webseite und die Aufgaben der dabei beteiligten Komponenten. • unterscheiden die Protokolle http und https. • erläutern den Aufbau einer URL. • stellen die (vereinfachte) Kommunikation zwischen Client und Webserver im Sequenzdiagramm dar. | | <ul style="list-style-type: none"> • beschreiben zentrale Komponenten eines Informatiksystems und deren Zusammenspiel. • beschreiben und begründen den dezentralen Aufbau des Internets. • nennen die zentralen Komponenten des Internets, u. a. Client, Server, Router, DNS und erläutern ihre Funktion. • beschreiben die Kommunikationswege im Internet. |
| 2 | Entwickeln eines Kommunikationsprotokolls | <ul style="list-style-type: none"> • entwickeln ein Kommunikationsprotokoll für mehrschrittige Client-Server-Kommunikation und stellen dieses in (vereinfachten) Sequenzdiagrammen dar. z. B. Geldüberweisung bei Online-Banking, Ticketkauf für Kino/Konzert, Einkauf im Online-Shop, etc. | https://weiterbildung-informatik.wollw.de/chapter4/part1/ | <ul style="list-style-type: none"> • entwerfen und implementieren ein Protokoll zur Übertragung von Daten über einen Kommunikationskanal. • beschreiben Aspekte zur Sicherheit der Kommunikation im Internet. |
| 1 | Bilder im Internet | <ul style="list-style-type: none"> • erläutern die Bedeutung von Bildern im Internet. • entwickeln ein Problembewusstsein für urheberrechtlich geschützte Bilder und berücksichtigen dies bei Bildersuchen in Internet. | | [KC-Sek I] <ul style="list-style-type: none"> • benennen Eigentumsrechte an digitalen Werken und das Recht am eigenen Bild. |
| 8 | Webseiten erstellen | <ul style="list-style-type: none"> • erstellen einfache Webseiten mit HTML und CSS. • unterscheiden bei der Erstellung Struktur und Inhalt (HTML) von Layout und Design (CSS). | https://wiki.selfhtml.org Notepad++, Phase 5 | [KC-Sek I] <ul style="list-style-type: none"> • erstellen Dokumente unter Verwendung von Auszeichnungssprachen, z. B. HTML und CSS. • überprüfen anhand eines selbst erstellten medialen Produkts rechtliche Aspekte der Veröffentlichung. |
| Textuelles Programmieren (ca. 7 Wochen) | | | | |
| In Ergänzung zu dem vorangegangenen Thema kann der Einstieg ins textuelle Programmieren zu dynamischen Webseiten führen. Wesentlich sind unabhängig vom verwendeten Werkzeug und der verwendeten Sprache, die erworbenen Kompetenzen. | | | | |
| 8 | Einstieg ins textuel- | <ul style="list-style-type: none"> • verwenden mathematische Operationen, Variablen, Fallun- | <i>Webprogrammierung:</i> | <ul style="list-style-type: none"> • verwenden und erstellen Operati- |

| Stunden | Thema | Erworbene Kompetenzen Die Schüler:innen ... | mögliches Material/ Software/Ideen | Bezug zum KC Die Schüler:innen ... |
|---------|------------------|---|--|--|
| | le Programmieren | <ul style="list-style-type: none"> terscheidungen und Schleifen zur Erstellung einfacher Programme. Erstellen und verwenden eigene Operationen, z. B. für Zahlenraten, Quiz, Malfolgen-Trainer, physik/math. Berechnungen <i>Webprogrammierung: nutzen Eingabefelder und Buttons, um selbstgeschriebene Operationen in JavaScript aufzurufen.</i> | <i>Notepad++, Phase 5, Vorlagen für html-Seiten zur Ein- und Ausgabe, damit Fokus auf JavaScript</i> | onen zur strukturierten Implementierung von Algorithmen. |
| 6 | Übungsprojekt | <ul style="list-style-type: none"> SuS nutzen erworbene Kompetenzen zur leistungsdifferenzierten Erstellung eines Programms <i>Webprogrammierung: verschiedene fachliche Schwerpunkte HTML/CSS bzw. HTML/CSS/JavaScript möglich</i> | | |

11.2

| Stunden | Thema | Erworbene Kompetenzen Die Schüler:innen ... | mögliches Material/ Software | Bezug zum KC Die Schüler:innen ... |
|------------------------------------|---------------------------------------|--|---------------------------------|---|
| Kryptologie (ca. 10 Wochen) | | | | |
| 1 | Sichere Kommunikation (Wdhlg) | <ul style="list-style-type: none"> unterscheiden zwischen Maßnahmen zum Datenschutz (rechtl. Vorgaben) und Maßnahmen zur Datensicherheit (technische, organisatorisch). erläutern Aspekte der sicheren Kommunikation (Integrität, Authentizität, Verbindlichkeit). beschreiben anhand von Protokollen (z. B. http und https) Unterschiede in der Sicherheit der Kommunikationswege. | | <ul style="list-style-type: none"> beschreiben Aspekte zur Sicherheit der Kommunikation im Internet. |
| 4 | Verschlüsselungsverfahren im Altertum | <ul style="list-style-type: none"> unterscheiden Transpositions- und Substitutionsverfahren. erläutern Beispiele für mindestens zwei Transpositionsverfahren (Gartenzaun, Skytale, Fleißnersche Schablone) und mindestens zwei Substitutionsverfahren (Caesar(!), Freimaurer, Playfair). vergleichen und beurteilen die Sicherheit der Verfahren. | SpionCamp | <ul style="list-style-type: none"> beschreiben das Prinzip der Transposition und der Substitution zur Verschlüsselung von Daten. beurteilen die Sicherheit einfacher Verschlüsselungsverfahren. |
| 2 | Caesar-Algorithmus | <ul style="list-style-type: none"> erstellen einen Algorithmus zur Caesar-Verschlüsselung in Form eines Struktogramm. | Structorizer | <ul style="list-style-type: none"> stellen Algorithmen in standardisierter Form dar. |
| 2 | Caesar-Implementierung | <ul style="list-style-type: none"> wenden Zeichenkettenoperationen an. implementieren das Caesar-Verfahren. | | <ul style="list-style-type: none"> entwerfen und implementieren Algorithmen unter Verwendung elementarer Zeichenkettenoperationen. implementieren monoalphabetische Verfahren (Caesar-Verfahren). |
| 2 | Caesar-Brechen | <ul style="list-style-type: none"> erstellen ein Programm zum „Knacken“ der Caesar-Verschlüsselung (durch Erstellen aller möglichen Klartexte). | | |

| Stunden | Thema | Erworbene Kompetenzen Die Schüler:innen ... | mögliches Material/ Software | Bezug zum KC Die Schüler:innen ... |
|--|---|--|--|---|
| 4 | Häufigkeitsanalyse | <ul style="list-style-type: none"> • führen eine Häufigkeitsanalyse für einen einfach verschlüsselten Text durch. • erstellen mit JavaScript ein Programm zur Ermittlung der Buchstabenhäufigkeiten eines Textes. | | <ul style="list-style-type: none"> • erläutern das Prinzip der Häufigkeitsanalyse. |
| 4 | Verschlüsselungsverfahren implementieren | <ul style="list-style-type: none"> • entwerfen ein weiteres monoalphabetisches Substitutionsverfahren in Form eines Struktogramms und implementieren dieses. • analysieren als Quelltext und Struktogramm gegebene Algorithmen zu Verschlüsselungen mit Hilfe von Tracetabellen. | | <ul style="list-style-type: none"> • entwerfen und implementieren Algorithmen unter Verwendung elementarer Zeichenkettenoperationen. • implementieren monoalphabetische Verfahren (Caesar-Verfahren). • stellen die Belegung von Variablen bei der Ausführung eines Algorithmus in Form einer Tracetabelle dar. • stellen Algorithmen in standardisierter Form dar. |
| Künstliche Intelligenz (ca. 5 Wochen) | | | | |
| 2 | Begriffsbildung | <ul style="list-style-type: none"> • erläutern die Begriffe künstliche Intelligenz und maschinelles Lernen. • nennen Teilgebiete und Anwendungsgebiete der künstlichen Intelligenz. | Material „MENSCH-Maschine“ | <ul style="list-style-type: none"> • reflektieren die Chancen und Risiken des Einsatzes von Informatiksystemen auf die Gesellschaft und auf das Individuum. |
| 2 | Verfahren der KI/ Maschinelles Lernen* | <ul style="list-style-type: none"> • erproben und analysieren anhand eines Spiels das verstärkende Lernen als eine Technik des maschinellen Lernens. • erläutern weitere Verfahren maschinellen Lernens. | Spiel „MENSCH-Maschine“ | <ul style="list-style-type: none"> • erläutern die Funktionsweise von Informatiksystemen mithilfe von Maschinenmodellen. • rekonstruieren Teile von Informatiksystemen. |
| 6 | Exemplarische Anwendungen der KI | <ul style="list-style-type: none"> • verwenden KI-Tools zu einem oder zwei der folgenden Bereiche und ordnen Chancen und Risiken ein. • diskutieren, ob Maschinen intelligent sein können. <p>Bilderstellung und Bildbearbeitung</p> <ul style="list-style-type: none"> • erproben fertige KI-Programme zur Veränderung und Erstellung von Bildern. • unterscheiden Veränderungen von Bildern zur Bildverbesserung und zur Bildmanipulation. • erkennen, wie leicht Bilder mithilfe geeigneter Tools verbessert bzw. manipuliert werden können. | Film: Das manipulierte Bild | <ul style="list-style-type: none"> • bearbeiten und verfremden Fotos und Grafiken mit einer (einfachen) Bildbearbeitungssoftware |
| | | <p>Bildererkennung mit Neuronalen Netzen</p> <ul style="list-style-type: none"> • erproben fertige KI-Programme zur Bildererkennung. • machen sich mit Einsatzgebieten solcher Systeme vertraut. • diskutieren Grenzen und gehen dabei auf problematische | Teachablemachine.withgoogle.com , Playground.tensorflow.org | |

| Stunden | Thema | Erworbene Kompetenzen Die Schüler:innen ... | mögliches Material/ Software | Bezug zum KC Die Schüler:innen ... |
|--|----------------------------|---|--|---------------------------------------|
| | | Lerndaten (z. B. Algorithmic Bias) ein. | | |
| | | Textgenerierende KI <ul style="list-style-type: none"> • machen sich mit Einsatzmöglichkeiten von LLMs vertraut und entwickeln Grundverständnis der Funktionsweise. • erproben LLMs anhand vorher abgestimmter Problemstellungen und bewerten so die Einsatzmöglichkeiten. • diskutieren Fragen des Urheberrechts KI-generierter Texte | ChatGPT u.ä. | |
| 2 | Weitere Anwendungen der KI | <ul style="list-style-type: none"> • <i>informieren sich, ggf. arbeitsteilig, zu einem konkreten Anwendungsbeispiel der künstlichen Intelligenz und diskutieren den Chancen und Risiken unter verschiedenen Gesichtspunkten.</i> <i>z. B. Autonomes Fahren, Verhaltenserkennung zur Gefahrenabwehr, Predictive Policing, Chat-GPT, Medizin, etc.</i> | Gewissensbits https://gewissensbits.gi.de/ | |
| *Der vorgeschlagene Ablauf orientiert sich in Teilen am Arbeitsmaterial „MENSCH-Maschine“. Der Schwerpunkt der KI kann aber auch bei Such-/Problemlösealgorithmen oder neuronalen Netzen gesetzt werden. | | | | |

Qualifikationsphase

12.1

| Stunden | Thema | Erworbene Kompetenzen Die Schüler:innen ... | mögliches Material/ Software | Bezug zum KC Die Schüler:innen ... |
|-----------------------------------|---|--|--|--|
| Datenbanken (ca. 6 Wochen) | | | | |
| 2 | Datenbanken und Datenbanksysteme | <ul style="list-style-type: none"> nennen Anwendungsbereiche von Datenbanken im Alltag. verwenden Alltags-Datenbanken und untersuchen Anforderungen an Datenbanken aus Anwender- und Entwicklersicht (Datensicherheit, Abfragemöglichkeit, Visualisierung der Ergebnisse, Datenintegrität, Mehrbenutzerfähigkeit, ..., <i>Codd'sche Regeln</i>). strukturieren Daten in Tabellen (Relationen) und unterscheiden dabei Datensatz, Datenwert, Attribut, Attributwert, Primärschlüssel und (Tabellen-) Schema. | Produktsuche (Amazon, Ebay, etc.) Bahnverbindung, Music-Map (https://www.music-map.de) , Bücherei (http://www.stadt.lue-neburg.de/ratsbuecherei) | <ul style="list-style-type: none"> erläutern den Aufbau relationaler Datenbanken unter Verwendung der Begriffe Datensatz, Attribut, Primärschlüssel, [Fremdschlüssel] und Tabelle. |
| 2 | Redundanzen, Anomalien und Inkonsistenzen | <ul style="list-style-type: none"> analysieren Fehler und Probleme, die bei der Verwaltung von Daten in Tabellen auftreten können und entwickeln erste Lösungsideen. | | <ul style="list-style-type: none"> nennen Beispiele für Einfüge-, Änderungs- und Löschanomalien. untersuchen ein gegebenes Datenbankschema auf Anomalien und Redundanzen. |
| 10 | SQL-Abfragen | <ul style="list-style-type: none"> erläutern Projektion, Selection, Vereinigung und Verbund und formulieren Anfragen mit SQL. <i>Der Verbund soll dabei nur mit Inner Join in der Form SELECT * FROM Tabelle1, Tabelle2 WHERE Tabelle1.Att = Tabelle2.Att; erfolgen.</i> verwenden bereitgestellte Datenbanken, um SQL-Abfragen zu formulieren. beachten die Auswertungsreihenfolge einer SQL-Anweisung beim Formulieren und Analysieren von SQL-Abfragen (FromWhereGroupbyHavingSelectOrder bzw. FWGHOS). <i>Hinweis: Having fehlt im Buch, muss aber behandelt werden.</i> formulieren und analysieren einfache geschachtelte Abfragen. | phpMyAdmin, MySQLQueryBrowser Mögliche DB: Fußball-Datenbank, Wetterdatenbank (beide https://dbup2date.uni-bayreuth.de/), VLIN-Materialien, Terra, SQL-Island (http://www.lgis.informatik.uni-kl.de/extra/game/) https://www.imoodle.de/sqltutorial/index.html Zur Syntax die ergänzenden Hinweise zum Kerncurriculum beachten | <ul style="list-style-type: none"> formulieren einfache Abfragen und Verbundabfragen über mehrere Tabellen. formulieren Abfragen an Datenbanken unter Verwendung von Aggregatfunktionen. |
| 5 | ER-Modelle | <ul style="list-style-type: none"> erläutern die Phasen eines Entwurfsprozesses (konzeptionell, logisch, physisch). verwenden die Begriffe Entität, Entitätsmenge, Attribut, Beziehung, Kardinalitäten zur Beschreibung und Interpretation von ER-Modellen. unterscheiden 1:1-, 1:n- und m:n-Beziehungen. erstellen und erweitern ER-Modelle zu gegebenen Sachkontexten. | Dia | <ul style="list-style-type: none"> interpretieren ein gegebenes ER-Diagramm. modellieren Datenbanken unter Verwendung des ER-Modells. |

| Stunden | Thema | Erworbene Kompetenzen Die Schüler:innen ... | mögliches Material/ Software | Bezug zum KC Die Schüler:innen ... |
|---------|---------------------------|---|---------------------------------|--|
| 3 | ER-Modelle und Relationen | <ul style="list-style-type: none"> überführen ER-Modelle in Relationen und formulieren geeignete Regeln. identifizieren Fremdschlüssel und Fremdschlüssel-Beziehungen. | | <ul style="list-style-type: none"> setzen ein ER-Modell in ein relationales Schema um. Erläutern [...] unter Verwendung der Begriffe [...], Fremdschlüssel, [...]. |
| 4 | Normalformen | <ul style="list-style-type: none"> verwenden Normalformen zur Beurteilung der Qualität von Datenbankmodellen und unterscheiden dabei funktionale und transitive Abhängigkeiten. verändern Datenbankmodelle so, dass sie den Normalformen entsprechen. <p><i>Hinweis: Das systematische Überführen in die 1. bis 3. Normalform ist nicht im KC nicht gefordert, hilft aber insbesondere schwächeren SuS.</i></p> <ul style="list-style-type: none"> erkennen, dass gute ER-Modellierungen eine Normalisierung i. A. überflüssig machen. | | <ul style="list-style-type: none"> beurteilen und verändern eine gegebene Datenbankmodellierung. |
| 5 | Komplexe Übung | <ul style="list-style-type: none"> erstellen und beurteilen ER-Modelle. überführen ER-Modelle in Relationen und überprüfen diese. verwenden SQL zur Abfrage von Datenbanken. | | |

| Stunden | Thema | Erworbene Kompetenzen Die Schüler:innen ... | mögliches Material/ Software/Ideen | Bezug zum KC Die Schüler:innen ... |
|-----------------------------------|-----------------------------------|---|---------------------------------------|--|
| Algorithmen (ca. 7 Wochen) | | | | |
| 8 | Textuelle Programmierung mit Java | <ul style="list-style-type: none"> • verwenden in <i>Java</i> zielgerichtet Kontrollstrukturen und Variablen zur Implementation von Algorithmen. Mögliche Algorithmen können z. B. das Arbeiten mit Zeichenketten (auftrennen, einfügen, suchen, ersetzen, etc.) oder das Thema Codierung und Netzwerke aus der Einführungsphase aufgreifen: Zahlensysteme (binär→ dezimal, dezimal → binär), Prüfziffern (Paritätsbit, ISBN, EAN), etc. Datentypen: Wahrheitswert (boolean), Ganzzahl (int), Fließkommazahl (double), Zeichen (char) und Zeichenkette (String) Zahloperationen: Modulo, Unterscheidung zwischen Division und ganzzahliger Division • implementieren Algorithmen, die als Struktogramm gegeben sind, in <i>Java</i>. • analysieren systematisch ihre erstellten Programme, z. B. mit Debugging oder Tracetabellen. | Java-Editor | <p>[aus 11]</p> <ul style="list-style-type: none"> • benennen Anweisung, Sequenz, Schleife und Verzweigung als Grundbausteine eines Algorithmus. • entwerfen und implementieren Algorithmen unter zielgerichteter Verwendung der elementaren Kontrollstrukturen. • erläutern das Prinzip der Speicherung von Werten in Variablen. • verwenden Variablen und Wertzuweisungen in Algorithmen. • stellen die Belegung von Variablen bei der Ausführung eines Algorithmus in Form einer Tracetabelle dar • entwerfen und implementieren Algorithmen unter Verwendung elementarer Zeichenkettenoperationen <p>[Neu]</p> <ul style="list-style-type: none"> • analysieren die Funktionsweise eines gegebenen Algorithmus. • verwenden geeignete Variablentypen zur Speicherung von Werten. |
| 1 | Algorithmen | <ul style="list-style-type: none"> • erläutern die Eigenschaften eines Algorithmus und unterscheiden verschiedene Darstellungsarten von Algorithmen. | | |
| 4 | Operationen in Java | <ul style="list-style-type: none"> • verwenden Methoden mit und ohne Rückgabewert sowie mit und ohne Übergabeparameter zur Implementation von Operationen zur Zeichenkettenverarbeitung. Dazu können die zuvor verwendeten Programme und Algorithmen zu Methoden erweitert werden. • unterscheiden Operationen anhand der Methodensignatur. | | <p>[aus 11]</p> <ul style="list-style-type: none"> • verwenden und erstellen Operationen zur strukturierten Implementierung von Algorithmen. <p>[Neu]</p> <ul style="list-style-type: none"> • verwenden Übergabeparameter und Rückgabewerte in Operationen. |

| | | | | |
|---|---------------------------|--|---|--|
| 4 | Suchen | <ul style="list-style-type: none"> • verwenden eindimensionale Reihungen in Algorithmen zu verschiedenen Sachzusammenhängen. • implementieren Algorithmen zur Suche in Reihungen von Zahlen und Zeichenketten. • entwickeln Algorithmen zur linearen und binären Suche. | | <ul style="list-style-type: none"> • erläutern das Prinzip, mehrere Daten des gleichen Typs in Reihungen zu verwalten, zu suchen ... |
| 6 | Sortieren | <ul style="list-style-type: none"> • entwerfen einen eigenen Sortieralgorithmus. • verwenden und analysieren die Sortierverfahren BubbleSort, InsertionSort und SelectionSort. • entwickeln zu mind. einem Sortierverfahren ein Struktogramm. • implementieren mind. ein Sortierverfahren. <i>Die Implementation soll mit statischen Reihungen erfolgen.</i> | Für BubbleSort, InsertionSort und SelectionSort ist Material vorhanden. | <ul style="list-style-type: none"> • ... und zu sortieren. • entwerfen und implementieren Algorithmen unter Verwendung von eindimensionalen Reihungen. • stellen Algorithmen in standardisierter Form und in schriftlich verbalisierter Form dar. |
| 3 | Effizienz von Algorithmen | <ul style="list-style-type: none"> • untersuchen praktisch und theoretisch den Aufwand der verschiedenen Sortieralgorithmen. • erkennen, dass der Aufwand der Sortieralgorithmen mitunter von der Vorsortierung der Daten abhängt und unterscheiden <i>best case</i>, <i>average case</i> und <i>worst case</i>. • beurteilen die Effizienz der Sortieralgorithmen. • unterscheiden verschiedene Komplexitätsklassen von Algorithmen. • bestimmen und beurteilen die Effizienz weiterer Algorithmen. | | <ul style="list-style-type: none"> • beurteilen die Effizienz von Algorithmen unter Abschätzung des Speicherbedarfs und der Zahl der Operationen. |
| 6 | Rekursive Algorithmen | <ul style="list-style-type: none"> • identifizieren das Prinzip der Rekursion in verschiedenen Sachzusammenhängen. • unterscheiden rekursive und iterative Algorithmen. • untersuchen gegebene rekursive Algorithmen (in verschiedenen Darstellungsformen) und identifizieren dabei Abbruchbedingungen/Rekursionsanker und Rekursionsschritt). • entwickeln und implementieren zu bekannten Problemen oder gegebenen iterativen Algorithmen einen rekursiven Algorithmus (z. B. binäre Suche, Zeichenkette umkehren, etc.). • entwickeln zu Türme von Hanoi einen rekursiven Algorithmus und untersuchen die Effizienz des Algorithmus. | Türme von Hanoi Rollenspiel zu Summen- oder Fakultätsbildung | <ul style="list-style-type: none"> • erläutern das Konzept der Rekursion an gegebenen Beispielen. • entwerfen und implementieren rekursive Algorithmen. • erläutern die Strategie „Teile und herrsche“ beim Entwurf rekursiver Algorithmen. |
| 2 | Rekursives Sortieren | <ul style="list-style-type: none"> • verwenden und analysieren Quicksort. <i>Die Implementation muss nicht vollständig erfolgen.</i> | | |

12.2

| Stunden | Thema | Erworbene Kompetenzen Die Schüler:innen ... | mögliches Material/ Software | Bezug zum KC Die Schüler:innen ... |
|--|-------------------------------------|---|---|---|
| Projekt 1: Datenbanken und Algorithmen (ca. 3 Wochen) <ul style="list-style-type: none"> Inhaltlich: Datenbankverbindungen aus Java heraus herstellen, Abfragen generieren und Ergebnisse verarbeiten. Mögliche Projektthemen: Fußball-Analysen (Ergebnisse auswerten um daraus Gewinnwahrscheinlichkeiten zu ermitteln) Visualisierung von Wetterdaten (Wetterdaten abfragen und mit <i>JavaSwing</i> grafisch darstellen) Dating-Portal (persönliche Nutzereingaben verwenden, um Partner-Vorschläge zu unterbreiten) | | | | |
| Codierung (ca. 4 Wochen) | | | | |
| 3 | Fehlererkennung und Fehlerkorrektur | <ul style="list-style-type: none"> erläutern das Prinzip von Prüfsummen anhand einer Alltagscodierung (ISBN, EAN, IBAN, Eurobanknoten, etc.) und entwickeln zu einem Verfahren ein Struktogramm. erweitern die 1-dimensionale Paritätsprüfung zu einer 2-dimensionalen Paritätsprüfung und untersuchen die Möglichkeiten der Fehlererkennung und Fehlerkorrektur. unterscheiden fehlererkennende und fehlerkorrigierende Codes. | | <ul style="list-style-type: none"> stellen Algorithmen in standardisierter Form dar. erläutern Möglichkeiten der Fehlererkennung und der Fehlerkorrektur bei der Datenübertragung, u. a. Paritätsbit, ... |
| 4 | Hamming-Abstand und Hamming-Code | <ul style="list-style-type: none"> bestimmen den Hamming-Abstand verschiedener Codierungen und beurteilen damit die fehlererkennende und fehlerkorrigierende Eigenschaften der Codierung. verwenden den (7,4)-Hamming-Code, um beliebige Codes zu einem fehlerkorrigierenden Code zu erweitern. <i>Hinweis: Zu empfehlen ist die graphische Darstellung des Codes mit Kreisen</i> überprüfen die fehlerfreie Speicherung/Übertragung von Daten mithilfe der kennengelernten Codierungen. | https://en.wikipedia.org/wiki/Hamming(7,4) | <ul style="list-style-type: none"> ... (7,4)-Hamming-Code. |
| 2 | Kommunikationsprotokoll | <ul style="list-style-type: none"> verwenden Paritätsbits und Hamming-Code zur Datenübertragung. | GoldCity/NewTown SaltLake/SilverLake | <ul style="list-style-type: none"> entwerfen und implementieren ein Protokoll zur Übertragung von Daten über einen Kommunikationskanal |
| 2 | Kompressionsverfahren | <ul style="list-style-type: none"> vergleichen den theoretischen und tatsächlichen Speicherbedarf von Bild- und Textdateien. erläutern verschiedene Möglichkeiten in Texten und Bildern Daten einzusparen und beurteilen dabei, ob es zu einem Informationsverlust kommt. | Abenteuer Informatik, Kapitel 5 | <ul style="list-style-type: none"> beschreiben Möglichkeiten, Daten zu komprimieren. |

| Stunden | Thema | Erworbene Kompetenzen Die Schüler:innen ... | mögliches Material/ Software | Bezug zum KC Die Schüler:innen ... |
|---|-------------------------------------|---|--|---|
| 4 | Lauf längencodierung | <ul style="list-style-type: none"> • erläutern das Prinzip der Lauf längencodierung. • entwickeln eine Lauf längencodierung für Pixelgrafiken und implementieren diese. | | <ul style="list-style-type: none"> • entwerfen und implementieren Algorithmen unter Verwendung von zweidimensionalen Reihungen. • entwerfen und implementieren ein Kompressionsverfahren zu einem gegebenen Sachverhalt. • beschreiben Möglichkeiten, Daten zu komprimieren, u. a. Lauf längencodierung. |
| 4 | Huffman-Codierung | <ul style="list-style-type: none"> • entwickeln ausgehend vom bekannten Morsecode die Idee eines Kompressionscode mit unterschiedlich langen Codewörtern. • beurteilen mithilfe der Fano-Bedingung, ob Codierungen eindeutig sind. • stellen Codierungen in Codebäumen dar. • verwenden die Huffman-Codierung zur Kompression von Daten. | | <ul style="list-style-type: none"> • beschreiben Möglichkeiten, Daten zu komprimieren, u. a. Huffman-Codierung. |
| 4 | Vergleich von Kompressionsverfahren | <ul style="list-style-type: none"> • erörtern verschiedene Möglichkeiten Audio- und Video-Daten zu komprimieren. • beschreiben ein modernes Kompressionsverfahren (zip, mp3, jpeg, etc.) • vergleichen verlustfreie und verlustbehaftete Kompressionsverfahren. | | <ul style="list-style-type: none"> • erläutern die Vor- und Nachteile verlustfreier und verlustbehafteter Kompression von Daten. |
| Klassen und Objekte, dynamische Datenstrukturen (ca. 8 Wochen) | | | | |
| 2 | Softwareentwicklung | <ul style="list-style-type: none"> • vergleichen verschiedene Vorgehensmodelle der Softwareentwicklung. • erläutern einen Software-Lebenszyklus. | | |
| 10 | Objekte und Klassen | <ul style="list-style-type: none"> • beschreiben Basiskonzepte der objektorientierten Modellierung (Klasse, Objekt, Attribut, Methode). • stellen Klassen als UML-Klassenkarte dar. • verwenden die Zugriffsattribute <i>private</i> und <i>public</i> für Attribute und Methoden zur Umsetzung der Datenkapselung. • entwerfen und implementieren eigene Klassen und Methoden zur Manipulation von Objekten. • entwerfen und verwenden zielgerichtet Konstruktoren. | Java-Editor (mit UML-Klassenmodellierer) Zur Darstellung von UML „Ergänzenden Hinweise zum KC“ beachten | <ul style="list-style-type: none"> • entwerfen und implementieren Algorithmen unter Verwendung von gegebenen und eigenen Klassen/Objekten. • unterscheiden zwischen lokalen und globalen Variablen. • unterscheiden zwischen primitiven Datentypen und Objektreferenzen |

| Stunden | Thema | Erworbene Kompetenzen Die Schüler:innen ... | mögliches Material/ Software | Bezug zum KC Die Schüler:innen ... |
|--------------|----------------------------|---|---|--|
| 10 | Klassenmodelle | <ul style="list-style-type: none"> modellieren Beziehungen zwischen Klassen mithilfe von Vererbungen und Assoziationen und stellen diese in UML-Klassendiagrammen dar. erstellen zu einfachen Sachkontexten objektorientierte Klassenmodelle, z. B. mithilfe der Verb-Substantiv-Methode. verändern gegebene Klassenmodelle, z. B. mithilfe der CRC-Methode. verwenden Reihungen von Objekten (<i>Objektarrays</i>) als Datentyp für Attribute. | | <ul style="list-style-type: none"> entwerfen Klassen und deren Beziehungen (Assoziation, Vererbung) und stellen diese durch Klassendiagramme dar. erläutern das Prinzip, mehrere Daten des gleichen Typs in Reihungen zu verwalten. |
| 8 | Dynamische Datenstrukturen | <ul style="list-style-type: none"> nennen unterschiedliche Anwendungsbeispiele der Datenstrukturen Stapel, Schlange und dynamische Reihung und beachten dabei auch die verschiedenen Inhaltstypen der Datenstrukturen. erläutern die Arbeitsweise der Datenstrukturen anhand der Operationen. vergleichen Stapel, Schlange, dynamische Reihung und statische Reihungen im jeweiligen Sachkontext. verwenden die Operationen der Datenstrukturen zur Entwicklung von Algorithmen und implementieren diese. | alte Prüfungsaufgaben zeigen unterschiedlichste Anwendungsbeispiele mit interessanten Fragestellungen auf „Ergänzenden Hinweise zum KC“ beachten | <ul style="list-style-type: none"> erläutern das Prinzip der Datenstrukturen Stapel, Schlange und dynamische Reihung. entwerfen und implementieren Algorithmen unter Verwendung der Datenstrukturen Stapel, Schlange und dynamische Reihung. |
| 8 | Binärbaum | <ul style="list-style-type: none"> verwenden die Fachbegriffe Binärbaum, Suchbaum, Teilbaum, Pfad, Knoten, Wurzel, Blatt, Ebene und Tiefe zur Beschreibung von Bäumen. vergleichen die Datenstrukturen Binärbaum mit linearen Datenstrukturen insbesondere hinsichtlich des Einfügens eines Elementes und des Suchens eines Elementes. erstellen und implementieren neben iterativen auch rekursive Algorithmen zur Traversierung sowie zur Suche und zum Einfügen. | | <ul style="list-style-type: none"> erläutern das Prinzip der Datenstruktur Binärbaum. entwerfen und implementieren Algorithmen zur Ausgabe der Daten eines Binärbaums in pre-, post- und inorder Reihenfolge. entwerfen und implementieren Algorithmen zur Suche und zum Einfügen in binäre Suchbäume. entwerfen und implementieren rekursive Algorithmen. |
| je nach Zeit | Übungsprojekt | <p><i>Inhaltlich:</i></p> <ul style="list-style-type: none"> Durchlaufen der verschiedenen Phasen der Softwareentwicklung sowie Anwenden von Fachmethoden (z. B. Verb-Substantiv-Methode). Arbeitsteiliges Implementieren von Software. Der Schwerpunkt kann bei der Objektorientierung oder einer der Datenstrukturen gesetzt werden. <p><i>Mögliche Projektthemen</i></p> <ul style="list-style-type: none"> Entwicklung eines objektorientierten textbasierten Adventure-Spiels. Entwicklung eines objektorientierten Strategie-Spiels (Dame, Halma, etc.) ... | | |

13.1

| Stunden | Thema | Erworbene Kompetenzen Die Schüler:innen ... | mögliches Material/ Software | Bezug zum KC Die Schüler:innen ... |
|---|--|--|---------------------------------|---|
| Kryptologie und Datenschutz (ca. 7 Wochen) | | | | |
| 2 | Substitution und Transposition [Wdhlg] | <ul style="list-style-type: none"> • erläutern und unterscheiden Substitutions- und Transpositionsverfahren. • verwenden die Begriffe Klartext, Geheimtext/Chiffre, Schlüssel, monoalphabetisch. • beschreiben Möglichkeiten der Kryptoanalyse monoalphabetischer Substitutionsverfahren. | | [aus 11] <ul style="list-style-type: none"> • beschreiben das Prinzip der Transposition und der Substitution zur Verschlüsselung von Daten. • erläutern das Prinzip der Häufigkeitsanalyse. • beurteilen die Sicherheit einfacher Verschlüsselungsverfahren. |
| 8 | Polyalphabetische Substitution | <ul style="list-style-type: none"> • erläutern das Prinzip der polyalphabetischen Substitution. • verwenden das Vigenère-Verfahren zur Ver- und Entschlüsselung von Texten. • entwickeln für das Vigenère-Verfahren ein Struktogramm und implementieren dieses. • beschreiben Möglichkeiten der Kryptoanalyse polyalphabetischer Verfahren (Kasiski-Test, Autokorrelation) und führen dieses durch. • beschreiben das Prinzip der Vernam-Verschlüsselung/OneTimePad als Spezialfall einer polyalphabetischen Verschlüsselung. • implementieren eine eigene Variante eines OneTimePad mit zufällig generiertem Schlüssel. | CrypTool | <ul style="list-style-type: none"> • beschreiben das Prinzip der polyalphabetischen Substitution, u. a. am Beispiel des Vigenère-Verfahrens. • entwerfen und implementieren ein symmetrisches Verschlüsselungsverfahren. |
| 5 | Symmetrische Verfahren | <ul style="list-style-type: none"> • beschreiben am Beispiel der Enigma die Funktionsweise von Chiffriermaschinen. • untersuchen anhand von mind. zwei Beispielen die Funktionsweise moderner symmetrischer Verschlüsselungsverfahren. • unterscheiden Strom- und Blockchiffreverfahren und erläutern das Prinzip von Blockchiffreverfahren. • erläutern das Schlüsseltauschproblem symmetrischer Verfahren. | CrypTool | <ul style="list-style-type: none"> • beurteilen die Sicherheit eines gegebenen symmetrischen Verschlüsselungsverfahrens. • erläutern die prinzipielle Funktionsweise eines modernen symmetrischen Blockchiffreverfahrens. |

| Stunden | Thema | Erworbene Kompetenzen Die Schüler:innen ... | mögliches Material/ Software | Bezug zum KC Die Schüler:innen ... |
|---------|-------------------------|---|---|--|
| 5 | Asymmetrische Verfahren | <ul style="list-style-type: none"> • erläutern das Prinzip asymmetrischer Verfahren, untersuchen exemplarisch das RSA-Verfahren und wenden dieses für einfache Beispiele an. • unterscheiden und vergleichen symmetrische und asymmetrische Verschlüsselungsverfahren. • beurteilen die Sicherheit asymmetrischer Verfahren unter Berücksichtigung aktueller und zukünftiger Rechnerleistungen. | CrypTool | <ul style="list-style-type: none"> • beschreiben und unterscheiden die Prinzipien der symmetrischen und asymmetrischen Verschlüsselung. |
| 5 | Digitale Signaturen | <ul style="list-style-type: none"> • erläutern Geheimhaltung, Integrität und Vertraulichkeit als zentrale Aspekte digitaler Kommunikation. • erläutern das Erstellen und Verifizieren digitaler Signaturen. • beschreiben das Prinzip digitaler Zertifikate, z. B. im Kontext der Webkommunikation. • verwenden PGP zum Ver- und Entschlüsseln von Nachrichten sowie zum digitalen Signieren von Nachrichten. • entwickeln Lösungen für das Schlüsseltauschproblem durch Kombination symmetrischer und asymmetrischer Verfahren (hybride Verfahren). | CrypTool | <ul style="list-style-type: none"> • erläutern das Prinzip von digitalen Signaturen und Zertifikaten. • beschreiben Anwendungsbereiche für symmetrische bzw. asymmetrische Verschlüsselungsverfahren. |
| 7 | Datenschutz | <ul style="list-style-type: none"> • unterscheiden technisch und organisatorische Maßnahmen zum Schutz von Daten (Datensicherheit) und rechtlichen Rahmenbedingungen (Datenschutz). • erläutern anhand fiktiver und realer Fallbeispiele Grundprinzipien des Datenschutzes und beziehen dabei die DSGVO, das Recht auf informationelle Selbstbestimmung sowie das IT-Grundrecht mit ein. • nennen Anwendungsbeispiele bei denen eine automatisierte Datenanalyse erfolgt und erörtern dabei Chancen und Risiken. • <i>diskutieren Vor- und Nachteile einer KI-basierten automatisierten Datenanalyse.</i> | Filme: Citizen Four, Quarks & CO -BigData | <p>[aus 11]</p> <ul style="list-style-type: none"> • erläutern die rechtlichen Rahmenbedingungen für den Umgang mit ihren persönlichen Daten, wie z. B. informationelle Selbstbestimmung und Datenschutzrichtlinien. <p>[Neu]</p> <ul style="list-style-type: none"> • diskutieren die Chancen und Risiken der automatisierten Datenanalyse. |

Projekt 2: (ca. 5 Wochen)

- Inhalte: Codierung, Verschlüsselung, Softwareentwicklung
- Projektidee: Verschlüsselungssoftware: Entwerfen eigenes Verschlüsselungsverfahren (für Texte), implementieren Software zum Ver- und Entschlüsseln.
- Projektidee: Kompressionssoftware: Entwickeln eigener (verlustbehafteter) Algorithmen zur Bildkompression und implementieren entsprechende Software.
- ...

4. Semester

| Stunden | Thema | Erworbene Kompetenzen Die Schüler:innen ... | mögliches Material/ Software | Bezug zum KC Die Schüler:innen ... |
|---|---|--|---|--|
| Endliche Automaten und Formale Sprachen (ca. 6 Wochen) | | | | |
| 4 | Deterministische endliche Automaten und Mealy-Automat | <ul style="list-style-type: none"> • geben einen DEA durch ein Eingabealphabet, einen Startzustand, eine Zustandsmenge, Übergangsfunktionen und Endzustände an (formale Definition unnötig). • geben einen Mealy-Automaten durch ein Eingabe- und ein Ausgabealphabet, einen Startzustand, eine Zustandsmenge, Übergangsfunktionen und Ausgabefunktionen an (formale Definition unnötig). • unterscheiden Akzeptoren (DEA) und Transduktoren (Mealy-Automat). • verwenden Zustandsgraphen/Übergangsgraphen und Zustandsübergangstabelle zur Darstellung von Automaten und wechseln zwischen diesen. • untersuchen die Funktion von Automaten, indem sie Eingaben mit dem Automaten verarbeiten. | AtoCC, SIMA zur Darstellung von Automaten „Ergänzenden Hinweise zum KC“ beachten | <ul style="list-style-type: none"> • beschreiben den Aufbau und die Funktionsweise eines deterministischen endlichen Automaten (DEA). • beschreiben den Aufbau und die Funktionsweise eines endlichen Automaten mit Ausgabe (Mealy-Automat). • analysieren die Funktion eines durch einen Zustandsgraphen vorgegebenen Automaten. |
| 4 | Modellieren mit endlichen Automaten | <ul style="list-style-type: none"> • entscheiden sich je nach Problemstellung für ein Automatenmodell und entwickeln Modelle für eine Problemstellung. Mögliche Problemstellungen: Silbentrennung, Parser, Serieller Addierer/Vergleicher, Simulation von Informatiksystemen. • erweitern gegebene Automaten um neue Funktionen. • untersuchen Problemstellungen, die nicht mit endlichen Automaten gelöst werden können. | | <ul style="list-style-type: none"> • entwickeln und implementieren Automatenmodelle in Form von Zustandsgraphen. • erläutern die Grenzen endlicher Automaten bei der Problemlösung. |
| 5 | Kellerautomaten | <ul style="list-style-type: none"> • geben einen Kellerautomaten durch ein Eingabealphabet, ein Kelleralphabet, einen Startzustand, eine Zustandsmenge, Übergangsfunktionen und Endzustände an (formale Definition unnötig). • verwenden zur Darstellung Zustandsgraphen und Übergangstabellen. • modellieren zuvor als unlösbar erkannte Problemstellungen mit Kellerautomaten. | AtoCC zur Darstellung von Kellerautomaten „Ergänzenden Hinweise zum KC“ beachten, die Notation im Buch weicht davon ab. | <ul style="list-style-type: none"> • beschreiben den Aufbau und die Funktionsweise eines Kellerautomaten als Erweiterung des Modells des endlichen Automaten. |
| 5 | Implementation mit Java | <ul style="list-style-type: none"> • <i>implementieren DEA und Mealy-Automaten in Java. Dabei kann die switch-case-Verzweigung verwendet werden.</i> • <i>implementieren Kellerautomaten unter Verwendung von Stapeln in Java.</i> | | <ul style="list-style-type: none"> • |

| Stunden | Thema | Erworbene Kompetenzen Die Schüler:innen ... | mögliches Material/ Software | Bezug zum KC Die Schüler:innen ... |
|---------|---------------------------|--|---------------------------------|---|
| 4 | Formale Sprachen | <ul style="list-style-type: none"> • unterscheiden künstliche und natürliche Sprachen, nennen Beispiele und Eigenschaften formaler Sprachen. • geben Sprachen mithilfe der erzeugenden Grammatik an und erstellen aus gegebenen Grammatiken Wörter einer Sprache. • überprüfen mithilfe von Ableitungsbäumen, ob Wörter durch eine Grammatik erzeugt werden können. | AtoCC | <ul style="list-style-type: none"> • nennen Eigenschaften formaler Sprachen im Vergleich zu natürlichen Sprachen. • beschreiben die von einer Grammatik erzeugte Sprache. |
| 3 | <i>Chomsky-Hierarchie</i> | <ul style="list-style-type: none"> • unterscheiden Sprachen und dabei insbesondere reguläre und kontextfreie Sprachen anhand ihrer Grammatiken <i>mithilfe der Chomsky-Hierarchie</i>. • entwickeln Grammatiken für reguläre und kontextfreie Sprachen. | | <ul style="list-style-type: none"> • entwerfen reguläre und kontextfreie Grammatiken für formale Sprachen. |
| 4 | Grammatiken und Automaten | <ul style="list-style-type: none"> • analysieren die Umsetzung von Grammatiken mit Automaten. • erstellen zu gegebenen regulären und kontextfreien Grammatiken endliche Automaten bzw. Kellerautomaten, die die gleiche Sprache akzeptieren. | | <ul style="list-style-type: none"> • erläutern den Zusammenhang zwischen regulären Grammatiken und endlichen Automaten. |