

Schulinternes Fachcurriculum Informatik

Sekundarstufe II

(grundlegendes Niveau)

Gemeinschaftsschule mit Oberstufe Faldera Neumünster

Inhaltsbezogene Kompetenzen

- Betriebs- und Kommunikationssysteme (BK)
- Algorithmen und Datenstrukturen (AD)
- Datenbanksysteme (DB)
- Softwareentwicklung (SE)

Querschnittsbereiche zum Beitrag des Faches Informatik zur Allgemeinbildung

- Big Data & KI (BDK)
- Informatik, Mensch und Gesellschaft (IMG)

Grobübersicht über die Themen und Inhalte des Unterrichts

(Der Querschnittsbereich wird themenbegleitend unterrichtet.)

Schuljahr	Betriebs- und Kommunikationssysteme	Algorithmen und Datenstrukturen	Datenbanken	Softwareentwicklung	Big Data & KI	Informatik, Mensch und Gesellschaft
E-Jahrgang	X	X			(x)	(x)
Q1-Jahrgang	X	X	X		(x)	(x)
Q2-Jahrgang		X	X	X	(x)	(x)

Anzahl der schriftlichen Leistungsnachweise: Pro Halbjahr wird eine Klausur- oder Klausur-Ersatzleistung erbracht.

Klassenstufe 11 (E-Jahrgang)
(grundlegendes Niveau | 3-stündig)

ca. Zeit	Inhaltsbezogene Kompetenzbereiche	Verbindliche Inhalte	Hinweise
	Beschreibung des Ergebnisses eines Algorithmus (AD1)	Algorithmus-Begriff als endliche effektive Ausführung von Arbeitsschritten	Verschiedene Beispiele aus der realen Welt zur Beschreibung nutzen
	Interpretation und Kommentierung einfacher Algorithmen in geeigneter Programmierumgebung (AD2)	Elementare Anweisung Sequenz von Anweisungen Kontrollstrukturen (bedingte Anweisung / Verzweigung, Wiederholung mit Abbruchbedingung) Verschachtelung von Kontrollstrukturen Variablen Bezeichner Datentyp (Zahl, Zeichen(kette)) Wert Definition / Initialisierung Zuweisung	Differenzierung nach Vorwissen / Können möglich, wobei es hier um die Kommentierung von vorgegebenen Algorithmen gehen soll; eine Wiederholung der Syntax einer bekannten Sprache aus der Sek 1 wäre sinnvoll; evtl. kann (AD8) und sogar auch (AD9) mit eingebunden werden Blockumgebung (z.B. APP-inventor, Scratch) Übergang Blockumgebung → Quellcode (z.B. Blockly) evtl. JAVA, Python (spielerische Umgebung z.B. Hamstersimulator, BigFoot)
	Formulierung von Handlungsvorschriften unter Nutzung algorithmischer Grundbausteine (AD8)	Pseudocode mit festen Schlüsselwörtern zu den Inhalten aus (AD2)	Differenzierung – Pseudocode nahe einer formalen Sprache oder eher umgangssprachlich aber mit festen Schlüsselwörtern Geeignet wäre die Wiederholung von Programmablaufplänen (PAP) aus der Sek 1
	Anwendung einer objektorientierten Programmierumgebung (AD1, AD2, AD8)	Code-Erstellung zu den Inhalten aus (AD2) an einfachen Beispielen	Geeignet ist eine Umgebung, mit der die Objektorientiertheit dargestellt werden kann

	Verwendung vorgegebener Klassen und Objekten bei der Programmierung (AD17)	Umsetzung in Programmen Klassen und Objekte Attribute und Methoden	An alltäglichen Beispielen Klassen, Objekte, Methoden, Attribute und die Idee der Vererbung darstellen Erlernen / Vertiefung einer geeigneten evtl. spielerischen Programmierumgebung
	Entwurf und Implementation von Algorithmen zur Lösung einer gegebenen Problemstellung (AD9)	Algorithmische Konzepte: Warten auf ein Ereignis, (mehrfach) verschachtelte Wiederholungen Rollen von Variablen in Programmtexten: Zählvariable, Wertspeicher, Akkumulator, Indikator, Index Modularisierung Parameterlose Unterprogramme Parameter und Argument Rückgabewert Lokale Variablen / Geltungsbereiche	Es sollten eher einfache kleine Probleme betrachtet werden, um die einzelnen Inhalte anzuwenden; die Zusammensetzung und damit das Zusammenspiel der Komponenten erfolgt dann an einem praktischen Projekt
	Praktisches Projekt (AD1, AD2, AD8, AD9, AD17)	Erstellung einer Lösung zu differenzierten Problemstellungen unter Zuhilfenahme vorgegebener Methoden / Klassen	Durch eine differenzierte bzw. modifizierte Problemstellung kann auf dem entsprechenden Niveau gearbeitet werden Diese Einheit kann als schriftliche Ersatzleistung eingebunden werden z.B. mit anschließendem Kolloquium
	Beschreibung der binären Repräsentation von Daten (BK1)	Bit Byte Wahrheitswerte Zahlen	Bitdarstellung als Schalter; Wiederholung aus der Sek 1 von KB, MB, GB, TB, auch mit den tatsächlichen Größen $2^{10} = 1024$; Klärung der Frage warum 1024 und nicht „nur“ 1000 Wahrheitstafeln für UND, ODER, NICHT und auch Kombinationen erstellen

		Zeichen und Zeichenketten Grafiken (Farbwerte)	Evtl. Wiederholung von „typischen“ Zahlensystemen: z.B. 2er-System Bei Farbwerten ist eine Wiederholung von Zahlensystemen am Beispiel des 16er-Systems sinnvoll
	Überführung der Informationen in eine Datenrepräsentation im Speicher (BK2)	Ganzzahlen	Betrachtung der Zahlen im 2er-System zur Festlegung der benötigten Byte Betrachtung eines Vorzeichenwechsels Zahlenbereiche der Variablen aus dem AD-Bereich hier konkret betrachten (für JAVA z.B. int, byte, long)
	Benennung technischer Kriterien, um Seriosität und Authentizität von Informationen zu beurteilen (BK3)	Metadaten in Bild- und Textdokumenten Erkennen von manipulierten Inhalten anhand technischer Details (u.a. Mailheader)	Konkrete E-Mails als aktuelles Beispiel verwenden
	Erklärung der grundlegenden Funktionsweise von Hardwarekomponenten und deren Zusammenwirken (BK4)	Sensoren und Eingabegeräte Prozessoren Speicher Netzwerkkomponenten Aktoren und Ausgabegeräte	
	Beschreibung der Funktion und des Zusammenwirkens von Rechner-, Steuer- und Speicherwerk in einem Von-Neumann-Rechner (BK5)	Von-Neumann-Maschinenmodell Steuerwerk Rechenwerk (ALU) Speicherwerk Fetch-Zyklus	z.B. Johnny-Modellrechner oder vn-simulator (jeweils auch als webbasierte Version)
	Beschreibung der Funktionen eines Betriebssystems (BK8)	Ressourcenverwaltung Dateienverwaltung Speicherverwaltung Prozessverwaltung Benutzerverwaltung	Beispiele von Betriebssystemen Verdeutlichung der grundlegenden Aufgabe an einem geeigneten Schalenmodell

		Rechteverwaltung	
	Beschreibung von Prozessen, deren Bestandteile und Eigenschaften (BK12)	Zustände Ready Running Blocked	Am Philosophenmodell mögliche Probleme erkennen und intuitiv lösen (z.B. Semaphore, Monitoring)
	Untersuchung von Prozessen während des Betriebs (BK13)	Zustände Speichernutzung Starten/Stoppen Öffnen/Schließen	Betrachtung des Task-Managers
	Erläuterung des Netzwerkbetriebs anhand eines Schichtenmodells (BK14)	Adressierung Paketvermittlung verbindungslos/verbindungsorientiert Protokolle der Anwendungsschicht	Schichten unter Filius betrachten Evtl. eignet sich ein Rollenspiel, um die Arbeitsaufgaben /-schritte zu verstehen
	Erläuterung des Internets als Verbund von Netzwerken (BK15)	IP-Adresse URL DNS HTTP(S) IMAP/AMTP SSH	Erstellung einer Simulation unter Filius: Router, Webserver, DNS-Server, E-Mail-Server
	Optional: Erstellung einer Website	HTML CSS	Dieser optionale Einschub könnte in die Simulation des Internets eingebunden werden
	Diskussion zu den Folgen netzbasierter Kommunikation für Entwicklungen in Politik und Gesellschaft (BK20) / Querschnittsbereich	soziale Netzwerke Manipulation der Meinungsbildung	z.B. Betrachtung von Kommunikationsprozessen in totalitären Systemen (Web-Zensur), anonymen Foren weitere Möglichkeiten: Nutzerordnung, Hilfsmöglichkeiten für Opfer, aktuelle Beispiele zur Manipulation der Meinungsbildung

Klassenstufe 12 (Q1-Jahrgang)
(grundlegendes Niveau | 3-stündig)

ca. Zeit	Inhaltsbezogene Kompetenzbereiche	Verbindliche Inhalte	Hinweise
	Erläuterung von Standardalgorithmen (AD3)	Suchen und Sortieren: Suchalgorithmen der Komplexitätsklassen $O(n)$ und $O(\log(n))$ Sortieralgorithmen der Komplexitätsklassen $O(n^2)$ und $O(n \cdot \log(n))$	Beispiele: Selection-Sort, Insertion-Sort, Bubble-Sort, Comb-Sort, Shell-Sort, Shaker-Sort log-Laufzeit: Mergesort, Quicksort Vergleich intuitiver Sortierung mit den vorgegebenen Algorithmen Darstellung durch Sortierung von haptischen Elementen in der Gruppe
	Vergleich und Beurteilung von Algorithmen zum Lösen eines Problems (AD4)	Korrektheit Problemangemessenheit Effizienz Laufzeit Speicherbedarf	Laufzeituntersuchung einiger o.g. Algorithmen z.B. durch Implementierung in vorgegebener Klasse, welche eine Zählung der Vergleiche und Tausche sowie eine zeitliche Auswertung bereitstellt
	Unterscheidung der Größenordnungen von Laufzeiten von Programmen (AD6)	Laufzeitmessung und -prognose sublineare lineare, polynomielle, logarithmische, exponentielle Laufzeit	s.o. Beispiele anhand von Suchalgorithmen
	Formulierung von Handlungsvorschriften unter Nutzung algorithmischer Grundbausteine (AD8)	Kontrollstruktur: Wiederholung mit fester Anzahl ggf. Erweiterung der Datentypen Felder	Z.B. Beschreibung des Sortier-Algorithmus durch ein PAP, Struktogramm o.ä. (Wiederholung)
	Entwurf und Implementation von Algorithmen zur Lösung	Algorithmische Konzepte: einfache Iteration über ein Feld, erschöpfende Suche, Aufzählen und Testen	Z.B. bei der Implementierung eines Suchalgorithmus

	einer gegebenen Problemstellung (AD9)		
	Entwurf und Implementation rekursiver Algorithmen (AD11)	Rekursion Abbruchbedingung Rekursionstiefe	z.B. bei Mergesort oder Quicksort
	Verwendung von geeigneten algorithmischen Strategien zur Implementation eines Algorithmus (AD13)	Divide & Conquer	z.B. bei Mergesort oder Quicksort
	Verwendung von vorgegebenen Klassen und Objekten bei der Programmierung (AD17)	Referenzen Beziehungen	s.o. Implementation von Suchalgorithmen (AD4)
	Beschreibung und Beurteilung von Authentifizierungsverfahren (BK9)	Benutzername und Passwort Zertifikate biometrische Authentifizierung Zweifaktoraauthentifizierung	z.B. Passwort (Aufbau und Länge, Sicherheit) AusweisApp Fingerabdruck, Gesichtserkennung; an Hotlines reicht manchmal auch Straße und Wohnort Online-Banking
	Beschreibung der Implementation von Dateien und Verzeichnissen in einem Dateisystem (BK10)	Dateisystem Hierarchie Verknüpfungen	z.B. ISO 9660 (altes CD-Dateisystem), FAT32, NTFS, exFAT (Flash-Dateisystem) Dateibaum (Ausblick auf Baumstrukturen) Dateiverknüpfung (Desktop, Bildschirm, etc.)
	Untersuchung und situationsangemessene Auswahl von Zugriffsrechten auf Dateien (BK11)	Benutzer(-gruppen) Berechtigungen	Berechtigungen können auch in Cloud-Anwendungen oder Lernmanagementsystemen betrachtet werden
	Beschreibung von Prozessen sowie deren Bestandteile und Eigenschaften (BK12)	Speicherbelegung Programmbereich Datenbereich Besitzer, Rechte	Arbeitsspeicher, Programm- und Datenspeicher, Stack (Kellerspeicher) (evtl. Auslagerung auf andere verfügbare Speicher)

	Diskussion über Sicherheitsrisiken und Abwehrmaßnahmen in Netzwerken (BK16)	Sicherheitsziele (Geheimhaltung, Nachrichten- und Teilnehmer-authentizität, Integrität) Risiken (Man-in-the-Middle, Phishing, Viren und Trojaner) Abwehrmaßnahmen (Firewall für Hard- und Software, sicherer Kanal, Public-Key-Infrastruktur, Virens Scanner)	evtl. falscher DHCP-Server; E-Mails zu Bankdaten
	Implementierung und Beurteilung der Wegefindung in einem dezentralen Netzwerk (BK17)	Bestandteile der Wegefindung Adresse Adressräume Switch Router	Erstellung einer Simulation unter Filius (Betrachtung von Adressen, Protokollen)
	Entwurf von Protokollen an Anwendungsbeispielen (BK18)	Sequenzdiagramme (UML) Nachrichtenformat	TCP/IP-Protokollfamilie: http, ftp, smtp, pop, dhcp, tcp, udp, ip-v4/6, Ethernet, Token Bus/Ring Protokoll-Schichten (Wiederholung Anwendungsschicht)
	Beschreibung von Webanwendungen und ihrer Architektur (BK19)	Architektur Client-Server Peer-to-Peer	z.B. Thin Client, Fat/Rich Client, Jamstack z.B. Datenbankserver, Client als Anwendungsprogramm z.B. Gnutella für Filesharing
	Diskussion zu den Folgen netzbasierter Kommunikation für Entwicklungen in Politik und Gesellschaft (BK20) / Querschnittsbereich	sichere Kommunikation Backdoor Anonymität im Internet	z.B. Betrachtung von Kommunikationsprozessen in totalitären Systemen (Web-Zensur), Darknet; weitere Möglichkeiten: Strafrecht Backdoor lässt sich gut mit Trojanern verbinden
	Entwicklung einer passenden Tabellenstruktur aus einer Problemstellung heraus (DB1) Verwendung von Ausdrücken zur Auswertung von Daten (DB2)	Repräsentation von Daten mithilfe von Tabellen Kalkulationstabellen (Attribute und Werte, Datentypen, Werte und Ausdrücke, Bezüge (relativ, absolut), logische / arithmetische und Text-Operatoren und -Funktionen, Aggregation, Diagramme	Grundmodell z.B. ER-Modell und daraus ein relationales Modell entwickeln (zur Begriffsverwendung s. unten DB5)

Erstellung und Anwendung von Datenbankabfragen (DB3)	Datenbankabfragen mithilfe geführter Abfragen mithilfe von SQL-Befehlen (select ... from ... where, joins)	z.B. Abfrage-Assistent unter MS Access https://dbup2date.uni-bayreuth.de https://sql-island.informatik.uni-kl.de
Anwendung von Operatoren zur Änderung von Datenbankinhalten (DB4)	Manipulation von Datenbankinhalten	z.B. https://sql-island.informatik.uni-kl.de
Erstellung eines Klassendiagramms aus gegebener Datenbankstruktur (DB5)	UML-Klassendiagramm Objekte, Objektmenge und Klasse Attribute Datentypen Beziehungen Kardinalitäten	Wiederholung/Anwendung der Begriffe aus der Objektorientierung: Klassendiagramm statt Relationsschema, Objekte statt Tupel, Objektmenge statt Relation, Klasse statt Relationstyp
Untersuchung eines Datenbankschemas (DB7)	Relationale Datenbanken Primär- und Fremdschlüssel	Hier kann z.B. auf den pers. Schlüssel eingegangen werden: Name, Vorname, Geb.-Dat, Geb.-Ort
Analyse von Redundanzen in einer Tabelle und in einem Satz von Tabellen (DB9)	Redundanz	
Diskussion der ökonomischen Verwertbarkeit von Daten (DB16)	Daten als Wirtschaftsfaktor	Bewertung der freiwilligen und unfreiwilligen Verwendung von Verhaltensdaten, z.B. personalisierte Werbung, automatisierte Preisanpassung
Diskussion des Einflusses von Datenfilterung auf Wahrnehmung und Meinungsbildung (DB17)	Filterblase	

Klassenstufe 13 (Q2-Jahrgang)
(grundlegendes Niveau | 3-stündig)

ca. Zeit	Inhaltsbezogene Kompetenzbereiche	Verbindliche Inhalte	Hinweise
	Erläuterung von Standardalgorithmen (AD3)	Algorithmen auf Bäumen und Graphen Datenstruktur von Graphen Datenstruktur von Suchbäumen	Beispiele: Backtracking, TSP, MST, Dijkstra-Algorithmus, Heap-Sort Beispiele: Tiefensuche, Breitensuche Beispiele: AVL-Bäume, SR-Bäume
	Beurteilung praktischer Grenzen der Algorithmisierung (AD5)	Korrektheit; Problemangemessenheit; Effizienz; Laufzeit; Speicherbedarf	
	Bewertung des Einflusses von Algorithmen auf Entscheidungsfindungen (AD7)	Algorithmische Entscheidungsfindung	Beispiele: autonomes Fahren, KI, Machine Learning
	Erläuterung einer zufallsgeleiterten Lösung für ein algorithmisches Problem (AD12)	Durchschnittliche Laufzeit oder Güte des Ergebnisses	
	Anwendung geeigneter algorithmischer Strategien bei der Implementierung von Algorithmen (AD13)	Greedy	
	Verwendung von Strategien zur zeitlichen Steuerung von Threads (AD14)	Nebenläufigkeit Semaphoren Verklemmungen	Koordination von Prozessen in Betriebssystemen Einstieg mit dem Philosophenmodell möglich

	Analyse einfacher künstlicher neuronaler Netze (AD15)	Forward Propagation Multi-Layer-Perzeptron Trainings- und Validierungsdaten	Beim Einsatz von Software können auch komplexere künstliche neuronale Netze betrachtet werden
	Erläuterung von typischen Strategien maschinellen Lernens (AD16)	überwachtes Lernen unüberwachtes Lernen Verstärkungslernen	Strategien für typische Verfahren anwenden evtl. mit Rollenspielen oder einfachen überschaubaren Situationen
	Beschreibung von Szenarien und Verwendungszwecken für Wörterbücher (AD18)	Wörterbücher als abstrakter Datentyp mit den Operationen	
	Ausführung von Operationen auf Listen und Bäumen (AD18)	Einfügen, Löschen, Suchen	
	Implementierung von Listen und Bäumen als Klassen und Verwendung dieser in einem Anwendungsbeispiel (AD19)	Listen und Bäume als Implementation von Wörterbüchern mit weiterführenden Operationen wie Filtern und Vereinigen	Einfach- und Doppelt-Verkettete-Listen Binärbäume
	Beurteilung der Unterschiede von Listen und Bäumen hinsichtlich der Effizienz (AD20)		Gegenüberstellung von vor und Nachteilen evtl. durch Zählen von Arbeitsschritten
	Analyse der Anforderungen, die von Dritten gestellt wurden, an ein zu entwickelndes System (SE1)	Anforderungsanalyse	offene, mehrdeutige Beschreibungen von Systemen als Ausgangsbasis für eine Projektplanung Vernetzen mit dem Projektlernen
	Untersuchung eines Datenbankschemas (DB7)	Referentielle Integrität Funktionale Abhängigkeiten	Wiederholung und Weiterführung aus Q1
	Analyse von Redundanzen in einer Tabelle und in einem Satz von Tabellen (DB9)	Anomalien Löschanomalien, Einfügeanomalien, Änderungsanomalien	Einfache Tabellen auf dem Papier Komplexere Tabellen in einer Simulation

	Erläuterung der Möglichkeiten zur Ableitung spezifischer Informationen aus einer großen Datenmenge (DB14)	Informationsgewinnung aus großen Datenmengen, z.B. aus Meta-Daten Sensordaten	Exemplarisch an Datensätzen durchführen
	Kommunikation mit Dritten und innerhalb eines Teams über einen Entwicklungsprozess und geplante Systeme (SE2)	Pflichtenheft Mock-Ups User stories	arbeitsteilige Gruppenarbeit, die aber eng geführt und abgesprochen wird einfache Methoden der Zeit-/Kostenschätzung können thematisiert werden
	Nennung von Vorgehensmodellen und Bewertung der Eignung für ein Vorhaben (SE3)	Wasserfallmodell Spiralmodell agile Methoden	
	Erstellung von Zeit- und Arbeitsplänen für die Projektarbeit (SE4)	Projektplanung: Gruppenbildung, Zuständigkeiten, Meilensteine	Rückgriff auf die gemachten Erfahrungen während des Profisseminars in Q1
	Präsentation des Ergebnisses eines Entwicklungsprozesses (SE5)	Präsentation Bereitstellung	Die Präsentation kann auch die Übergabe einer Dokumentation erhalten
	Entwicklung von Modellen eines Systems basierend auf gestellte Anforderungen (SE7)	Identifizieren von Anwendungsfällen Auslöser, Ablauf, Analyse der statischen Struktur eines Systems, Komponenten, Klassen, Analyse des dynamischen Systemverhaltens, Ereignisse und Aktionen	Mögliche thematische Bezüge: „Usability“ (Bereich Kunst/Design), „Zugänglichkeit“ von Systemen (gesellschaftliche Aspekte)
	Verwendung geeigneter Beschreibungsformate, um solche Modelle darzustellen (SE8)	UML-Diagramm Klassendiagramm	Sequenz- und Zustandsdiagramme können bei Bedarf auch hinzugezogen werden
	Implementation auf Basis einer Modellierung von einer	Modularisierung und Schnittstellen Vorhandene Bibliotheken	Umsetzung des in der Theorie geplanten Projekts Weiterführung des Projektplans

	Software unter Verwendung geeigneter objektorientierter Techniken (SE10)		
	Beschreibung von Verfahren um Code zu testen (SE11)	Codeabdeckung Grenzfälle	Es soll auf automatisierte Testverfahren wie Unit-Tests zurückgegriffen werden, und diese sollen auch praktisch erprobt werden