

Gymnasium Adolfinum Bückeberg

Lulu-v.-Strauß-u.-Torney-Str. 30

31675 Bückeberg

Schuleigener Arbeitsplans für das Fach Informatik

Unterricht im Fach Informatik wird am Gymnasium Adolfinum Bückeberg in der Einführungsphase der gymnasialen Oberstufe (Jg. 11) sowie in der Qualifikationsphase (Jg. 12 und 13) in Kursen auf grundlegendem und zukünftig auf erhöhtem Anforderungsniveau erteilt.

Ergänzt wird das Unterrichtsangebot in Informatik durch eine Roboter-Arbeitsgemeinschaft, in der sich die Schülerinnen und Schüler des 5. und 6. Jahrgangs im Rahmen der Begabtenförderung mit der Konstruktion und Programmierung von Lego Mindstorms Robotern beschäftigen.

Darüber hinaus nehmen alle Schülerinnen und Schüler des 6. Jahrgangs an Unterricht zur informationstechnischen Grundbildung teil. In dem halbjährigen Kurs, der im Umfang von zwei Wochenstunden unterrichtet wird, erlernen die Kinder die Grundlagen der Computerbedienung sowie erste Schritte im Umgang mit Standard-Office-Anwendungen wie Word, Powerpoint und Excel.

Informatikunterricht in der gymnasialen Oberstufe

Grundlage des Unterrichts bildet das mit dem Schuljahr 2018/19 erstmals gültige „Kerncurriculum für das Gymnasium – gymnasiale Oberstufe“. Neben den allgemeinen Zielen des Fachs Informatik und den zu entwickelnden Kompetenzen sind dort im Rahmen von Lernfeldern bestimmte Themenfelder festgelegt. Eine genaue Festlegung der inhalts- und prozessbezogenen Kompetenzen und ihre Zuordnung zu den einzelnen Lernfeldern ist in den folgenden Tabellen aufgeführt.

Unterricht in der Einführungsphase

Mithilfe graphischer und textbasierter Entwicklungsumgebungen werden Kenntnisse zum algorithmischen Problemlösen vermittelt und die Schüler in die Lage versetzt, selbstständig Algorithmen für Problemstellungen zu entwickeln und zu testen. Insbesondere sollen die Schüler auf die Anforderungen in der Qualifikationsphase vorbereitet werden.

Weiterhin erhalten sie einen Einblick in die Codierung, Verschlüsselung und Übertragung von Daten und erleben dabei, dass Informatik nicht nur Programmierung umfasst, sondern vielfältige Themengebiete aufweist. So erfahren sie, dass Informatik in ihrer Lebenswirklichkeit einen breiten Raum einnimmt.

Lernfeld: Algorithmen und Datenstrukturen	Modul: Grundlagen der Algorithmik
<i>Die Schülerinnen und Schüler ...</i>	<i>Methodische Hinweise / Materialien</i>
<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.- stellen Algorithmen in standardisierter Form dar. <p>[Kompetenzen laut KC: PK1.1, PK1.2, PK2.1, PK2.2, PK3.2 ; IK2.1]</p>	<p><i>10 Doppelstunden</i></p> <p>Als Einstieg bietet sich eine Erarbeitung des Algorithmus-Begriffs anhand von Algorithmen aus dem Alltag an.</p> <p>Durchgängig genutzt wird die Programmiersprache Java. Erste Programme werden mit dem Java-Hamstersimulator von Dietrich Boles (oder einer ähnlichen Mikrowelt) entwickelt.</p>
<ul style="list-style-type: none">- erläutern das Prinzip der Speicherung von Werten	Systematisches Probieren ist bei der

<p>in Variablen.</p> <ul style="list-style-type: none"> - verwenden Variablen und Wertzuweisungen in Algorithmen. - stellen die Belegung von Variablen bei der Ausführung eines Algorithmus in Form einer Tracetabelle dar. <p>[Kompetenzen laut KC: PK1.1, PK1.2, PK2.2, PK3.2; IK1.2]</p>	<p>Programmentwicklung ein legitimes Mittel, sollte im Unterrichtsgang aber durch die strukturierte Entwicklung von Algorithmen ersetzt werden.</p> <p>Im Hinblick auf zentrale Aufgabenstellungen muss die Arbeit mit Struktogrammen intensiv behandelt werden. Schreibtischtests mit Tracetabellen müssen permanenter Bestandteil der Arbeit mit Algorithmen sein.</p>
--	--

Lernfeld: Informationen und Daten	Modul: Codierung und Übertragung von Daten
<i>Die Schülerinnen und Schüler ...</i>	<i>Methodische Hinweise / Materialien</i>
<ul style="list-style-type: none"> - beschreiben grundlegende Codierungen von Daten, u. a. Dualzahlen, ASCII, RGB-Modell. <p>[Kompetenzen laut KC: PK3.3, PK4.3 ; IK1.1]</p>	<p><i>3 Doppelstunden</i></p> <p>Ziel ist die Beschreibung von Codierungen zur Vorbereitung der folgenden Module.</p> <p>Behandlung der Dualzahlen, Umwandlung von und in Dezimalzahlen, Stellenabschätzungen, Addition im Dualsystem. U.a. im Hinblick auf das RGB-Modell ist vertiefend auch eine Behandlung des Hexadezimalsystems sinnvoll.</p>

Lernfeld: Algorithmen und Datenstrukturen	Modul: statische und dyn. Datenstrukturen
<i>Die Schülerinnen und Schüler ...</i>	<i>Methodische Hinweise / Materialien</i>
<ul style="list-style-type: none"> - entwerfen und implementieren Algorithmen unter Verwendung elementarer Zeichenkettenoperationen. <p>[Kompetenzen laut KC: PK1.2, PK2.2 ; IK2.2]</p>	<p><i>5 Doppelstunden</i></p> <p>Spätestens hier erfolgt der Umstieg vom Hamstersimulator auf eine (nicht zu komplexe) Entwicklungsumgebung, z. B. Geany.</p> <p>Der sichere Umgang mit Zeichen und Zeichenketten in Java ist u. a. Voraussetzung für die Implementierung von Verschlüsselungsverfahren im Modul Kryptologie.</p>

Lernfeld: Informationen und Daten	Modul: Kryptologie
<i>Die Schülerinnen und Schüler ...</i>	<i>Methodische Hinweise / Materialien</i>
<ul style="list-style-type: none"> - beschreiben das Prinzip der Transposition und der Substitution zur Verschlüsselung von Daten. - implementieren monoalphabetische Verfahren, u. a. Caesar-Verfahren. - erläutern das Prinzip der Häufigkeitsanalyse. - beurteilen die Sicherheit einfacher Verschlüsselungsverfahren. <p>[Kompetenzen laut KC: PK2.2, PK3.3 ; IK2.3]</p>	<p><i>6 Doppelstunden</i></p> <p>Caesar und mind. ein weiteres einfaches Verfahren werden in Java implementiert.</p> <p>Behandlung von Vigenère ohne Programmierung. Weitere Verfahren als optionale Ergänzung.</p> <p>Empfehlenswert sind u. a. die Verfahren aus dem Spioncamp.</p>

Lernfeld: Informationen und Daten	Modul: Codierung und Übertragung von Daten
<i>Die Schülerinnen und Schüler ...</i>	<i>Methodische Hinweise / Materialien</i>
<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. - beschreiben Aspekte zur Sicherheit der Kommunikation im Internet. <p>[Kompetenzen laut KC: PK3.1, PK3.3 ; IK3.1, IK3.3]</p>	<p><i>3 Doppelstunden</i></p> <p>Nutzung des Spiels der Uni Oldenburg zur Klärung der Funktion der einzelnen Komponenten.</p> <p>Nutzung des Netzwerksimulationstools Filius.</p> <p>Auseinandersetzung mit verschiedenen Fragestellungen: z.B. Kabel oder Funk? Wie wichtig sind (See-)Kabel? Wem gehört das Internet?</p> <p>Ist genügend Zeit vorhanden, bieten sich die Entwicklung einer HTML-Seite sowie Hands-On-Experimente zur Informationsübertragung im Internet, z. B. mit dem traceroute-Befehl, an.</p>

Lernfeld: Informationen und Daten	Modul: Datenschutz
<i>Die Schülerinnen und Schüler ...</i>	<i>Methodische Hinweise / Materialien</i>
<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>[Kompetenzen laut KC: PK3.3 ; IK4.2]</p>	<p><i>2 Doppelstunden</i></p> <p>Nutzung von Daten durch Firmen für die Profilbildung</p> <p>Persönlicher Umgang mit seinen Daten</p> <p>Mögliche Ergänzung: Webseite Datendealer</p>

Unterricht in der Qualifikationsphase

In der Qualifikationsphase werden die Kenntnisse zum algorithmischen Problemlösen vertieft und die Schüler in die Lage versetzt, selbstständig Algorithmen für Problemstellungen zu entwickeln und zu testen. Sie machen sich in unterschiedlichen Kontexten mit den Prinzipien der algorithmischen Problemlösung und der Implementierung ihrer Lösungsideen vertraut. Dazu entwerfen sie eigene Operationen, Klassen und Objekte, um zu einer strukturierten Realisierung ihrer Lösungsideen zu gelangen.

Weiterhin vertiefen sie ihre Kenntnisse in der Codierung, Verschlüsselung und Übertragung von Daten und begreifen dabei, dass Informatik in der Lebenswirklichkeit ein unverzichtbarer Bestandteil ist.

Strukturen von Datenbanken und deren Auswertung sind ein weiterer Schwerpunkt. Die automatisierte Datenanalyse hat eine hohe gesellschaftliche Relevanz. So ist es notwendig, sowohl die rechtlichen Aspekte als auch die technischen Möglichkeiten zum Schutz persönlicher oder sensibler Daten zu kennen.

Neu in der Qualifikationsphase ist die zustandsbasierte Modellierung zur Entwicklung und Realisierung von technischen Systemen und Programmen. Hier soll das Verhalten technischer Systeme in Form von endlichen Automaten modelliert werden, um ihren Aufbau und ihre Funktionsweise zu erschließen. Auf erhöhtem Niveau schließt dies die Beschreibung in Form von formalen Sprachen sowie deren Synthese und Analyse mit Hilfe von Grammatiken ein.

Kursiv gedruckt: nur relevant für Kurse auf erhöhtem Anforderungsniveau

Lernfeld: Algorithmen und Datenstrukturen	Modul: Grundlagen der Algorithmik
<i>Die Schülerinnen und Schüler ...</i>	<i>Methodische Hinweise / Materialien</i>
<ul style="list-style-type: none"> - analysieren die Funktionsweise eines gegebenen Algorithmus. - stellen Algorithmen in schriftlich verbalisierter Form dar. - <i>beurteilen die Effizienz von Algorithmen unter Abschätzung des Speicherbedarfs und der Zahl der Operationen.</i> <p>[Kompetenzen laut KC: PK1.2, PK2 IK1.2, IK2.2, <i>PK1.1, PK1.2, PK2.2, PK2.3 IK2.2</i>]</p>	<p>ca. 12 Wochen in 12.1 (inklusive Projektarbeit)</p> <p>Einstieg mit einfachen klassischen Algorithmen wie ggT und kgV</p> <p>Sortierverfahren, z.B.:</p> <ul style="list-style-type: none"> - Bubblesort - Mergesort <p><i>Rekursive Algorithmen, z.B.</i></p> <ul style="list-style-type: none"> - Fakultät - Binomialkoeffizient - Quicksort - <i>hinsichtlich Laufzeitanalyse: Laufzeit in Abhängigkeit der Anzahl der zu sortierenden Elemente</i> <p>Projektarbeit: Programmierprojekt, z. B. Strategiespiel mit Computergegner</p> <ul style="list-style-type: none"> - 2 Wochenstunden Unterricht - 3 Wochenstunden Projektarbeit
<ul style="list-style-type: none"> - verwenden geeignete Variablentypen zur Speicherung von Werten. - unterscheiden zwischen lokalen und globalen Variablen. - unterscheiden zwischen primitiven Datentypen und Objektreferenzen. <p>[Kompetenzen laut KC: PK1.2, PK2 IK1.2, IK2.2]</p>	
<ul style="list-style-type: none"> - verwenden Übergabeparameter und Rückgabewerte in Operationen. - <i>erläutern das Konzept der Rekursion an gegebenen Beispielen.</i> - <i>entwerfen und implementieren rekursive Algorithmen.</i> - <i>erläutern die Strategie „Teile und herrsche“ beim Entwurf rekursiver Algorithmen.</i> <p>[Kompetenzen laut KC: PK1.2, PK2 IK1.2, IK2.2, <i>PK1.1, PK1.2, PK2.2, PK2.3 IK2.2</i>]</p>	

Lernfeld: Informationen und Daten	Modul: Codierung und Übertragung von Daten
<i>Die Schülerinnen und Schüler ...</i>	<i>Methodische Hinweise / Materialien</i>
<ul style="list-style-type: none"> - beschreiben Möglichkeiten, Daten zu komprimieren, u. a. Lauflängencodierung, Huffman-Codierung. - <i>entwerfen und implementieren ein Kompressionsverfahren zu einem gegebenen Sachverhalt.</i> - <i>erläutern die Vor- und Nachteile verlustfreier und verlustbehafteter Kompression von Daten.</i> - <i>erläutern Möglichkeiten der Fehlererkennung und der Fehlerkorrektur bei der Datenübertragung, u. a. Paritätsbit, (7,4)-Hamming-Code.</i> <p>[Kompetenzen laut KC: PK1.1, PK2.1, PK2.2, PK3.3 IK1.1, IK2.3, IK3.4, <i>PK1.1, PK2.1, PK2.2, PK3.3 IK1.1, IK2.3</i>]</p>	<p>ca. 6 Wochen in 12.1 (gA: 4 Wochen)</p> <p>Es bietet sich an, Lauflängencodierung und Huffman-Codierung anhand von Bilddateien zu unterrichten.</p> <p>Nur wenig Implementierungen, z.B. Lauflängencodierung.</p>
<ul style="list-style-type: none"> - entwerfen und implementieren ein Protokoll zur Übertragung von Daten über einen 	

Kommunikationskanal	
[Kompetenzen laut KC: PK1.1, PK2.1, PK2.2, PK3.3 IK1.1, IK2.3, IK3.4]	

Lernfeld: Informationen und Daten	Modul: Kryptologie
<i>Die Schülerinnen und Schüler ...</i>	<i>Methodische Hinweise / Materialien</i>
<ul style="list-style-type: none"> - beschreiben das Prinzip der polyalphabetischen Substitution, u.a. am Beispiel des Vigenère-Verfahrens. - beurteilen die Sicherheit eines gegebenen symmetrischen Verschlüsselungsverfahrens. - beschreiben und unterscheiden die Prinzipien der symmetrischen und asymmetrischen Verschlüsselung. - beschreiben Anwendungsbereiche für symmetrische bzw. asymmetrische Verschlüsselungsverfahren. - erläutern das Prinzip von digitalen Signaturen und Zertifikaten. - <i>entwerfen und implementieren ein symmetrisches Verschlüsselungsverfahren.</i> - <i>erläutern die prinzipielle Funktionsweise eines modernen symmetrischen Blockchiffreverfahrens.</i> <p>[Kompetenzen laut KC: PK3.3 IK4.3; <i>PK2.1, PK2.2, PK3.3 IK1.1, IK2.3</i>]</p>	<p><i>ca. 8 Wochen in 12.1/12.2</i></p> <p>Abgrenzung der Begriffe Codierung und Verschlüsselung</p> <p>Sinnvolle Ergänzungen:</p> <ul style="list-style-type: none"> - AES - RSA-Verschlüsselung - Diffie-Hellman-Schlüsseltausch <p>Implementierung:</p> <ul style="list-style-type: none"> - 2-dimensionale Reihungen nutzen (abiturrelevant) - Zeichenkettenoperationen wiederholen

Lernfeld: Informationen und Daten	Modul: Datenbanken
<i>Die Schülerinnen und Schüler ...</i>	<i>Methodische Hinweise / Materialien</i>
<ul style="list-style-type: none"> - erläutern den Aufbau relationaler Datenbanken unter Verwendung der Begriffe Datensatz, Attribut, Primärschlüssel, Fremdschlüssel und Tabelle. - nennen Beispiele für Einfüge-, Änderungs- und Löschanomalien. - untersuchen ein gegebenes Datenbankschema auf Anomalien und Redundanzen. - formulieren einfache Abfragen und Verbundabfragen über mehrere Tabellen. - formulieren Abfragen an Datenbanken unter Verwendung von Aggregatfunktionen. - <i>interpretieren ein gegebenes ER-Diagramm.</i> - <i>modellieren Datenbanken unter Verwendung des ER-Modells.</i> - <i>setzen ein ER-Modell in ein relationales Schema um.</i> - <i>beurteilen und verändern eine gegebene Datenbankmodellierung.</i> <p>[Kompetenzen laut KC: PK1.2, PK1.3, PK2.2,</p>	<p><i>ca. 8 Wochen in 12.2</i></p> <p>Tutorial mit Lösungen: https://www.luo-darmstadt.de/sqltutorial/index.html</p>

PK3.1, IK2.4, <i>PK1.4, PK1.5, PK3.1, PK3.2</i> IK1.5]	
--	--

Lernfeld: Informationen und Daten	Modul: Datenschutz
<i>Die Schülerinnen und Schüler ...</i>	<i>Methodische Hinweise / Materialien</i>
- diskutieren die Chancen und Risiken der automatisierten Datenanalyse.	ca. 2 Wochen in 12.2
[Kompetenzen laut KC: PK3.3 ; IK4.1]	

Lernfeld: Automaten und Sprachen	Modul: Automatenmodelle
<i>Die Schülerinnen und Schüler ...</i>	<i>Methodische Hinweise / Materialien</i>
- beschreiben den Aufbau und die Funktionsweise eines deterministischen endlichen Automaten (DEA).	ca. 5-6 Wochen in 13.1 (gA: ca. 6 Wochen in 13.1/13.2)
- beschreiben den Aufbau und die Funktionsweise eines endlichen Automaten mit Ausgabe (Mealy-Automat).	Einstieg mit Schatzinsel-Spiel (Schöningh-Buch)
- entwickeln und implementieren Automatenmodelle in Form von Zustandsgraphen.	Simulation der verschiedenen Automatenmodelle mit JFLAP
- analysieren die Funktion eines durch einen Zustandsgraphen vorgegebenen Automaten.	Mögliche Ergänzungen:
- erläutern die Grenzen endlicher Automaten bei der Problemlösung.	- Turingmaschine
- <i>beschreiben den Aufbau und die Funktionsweise eines Kellerautomaten als Erweiterung des Modells des endlichen Automaten.</i>	- zelluläre Automaten
	- Petri-Netze
	- NEAs, Konvertierung von NEAs in DEAs
	- Implementierung von NEAs in Java
[Kompetenzen laut KC: PK1.1, PK1.5, PK2.2, PK2.3, PK3.1, PK3.2 IK3.1, IK3.2, IK4.3, <i>IK3.1</i>]	

Lernfeld: Algorithmen und Datenstrukturen	Modul: Klassen und Objekte
<i>Die Schülerinnen und Schüler ...</i>	<i>Methodische Hinweise / Materialien</i>
- entwerfen und implementieren Algorithmen unter Verwendung von gegebenen und eigenen Klassen/Objekten.	ca. 6-7 Wochen in 13.1 (gA: ca. 8-10 Wochen)
- <i>entwerfen Klassen und deren Beziehungen (Assoziation, Vererbung) und stellen diese durch Klassendiagramme dar.</i>	Projektarbeit: Programmierprojekt, z. B. Entwicklung eines Bildbearbeitungsprogramms
	- 2 Wochenstunden Unterricht
	- 3 Wochenstunden Projektarbeit
[Kompetenzen laut KC: PK1.2, PK1.3, PK2.1 IK1.4, <i>PK1.4, PK1.5, PK3.1, PK3.2</i> IK1.4]	

Lernfeld: Algorithmen und Datenstrukturen	Modul: Statische und dyn. Datenstrukturen
<i>Die Schülerinnen und Schüler ...</i>	<i>Methodische Hinweise / Materialien</i>
- erläutern das Prinzip, mehrere Daten des gleichen Typs in Reihungen zu verwalten, zu suchen und zu	ca. 5-6 Wochen in 13.1 (gA: ca. 8 Wochen)

<p>sortieren.</p> <ul style="list-style-type: none"> - entwerfen und implementieren Algorithmen unter Verwendung von ein- und zweidimensionalen Reihungen. - erläutern das Prinzip der Datenstrukturen Stapel, Schlange und dynamische Reihung. - entwerfen und implementieren Algorithmen unter Verwendung der Datenstrukturen Stapel, Schlange und dynamische Reihung. <p>[Kompetenzen laut KC: PK1.3, PK2.1, PK2.2 IK1.3, IK2.2]</p>	
---	--

Lernfeld: Automaten und Sprachen	Modul: Formale Sprachen
<i>Die Schülerinnen und Schüler ...</i>	<i>Methodische Hinweise / Materialien</i>
<ul style="list-style-type: none"> - <i>nennen Eigenschaften formaler Sprachen im Vergleich zu natürlichen Sprachen.</i> - <i>beschreiben die von einer Grammatik erzeugte Sprache.</i> - <i>entwerfen reguläre und kontextfreie Grammatiken für formale Sprachen.</i> - <i>erläutern den Zusammenhang zwischen regulären Grammatiken und endlichen Automaten.</i> <p>[Kompetenzen laut KC: PK3.1 IK3.4]</p>	<p>ca. 2-3 Wochen in 13.1/13.2 (entfällt auf gA)</p> <p>Mögliche Ergänzungen:</p> <ul style="list-style-type: none"> - Chomsky Hierarchie - L-Systeme

Lernfeld: Algorithmen und Datenstrukturen	Modul: Statische und dyn. Datenstrukturen
<i>Die Schülerinnen und Schüler ...</i>	<i>Methodische Hinweise / Materialien</i>
<ul style="list-style-type: none"> - <i>erläutern das Prinzip der Datenstruktur Binärbaum.</i> - <i>entwerfen und implementieren Algorithmen zur Ausgabe der Daten eines Binärbaums in pre-, post- und inorder Reihenfolge.</i> - <i>entwerfen und implementieren Algorithmen zur Suche und zum Einfügen in binäre Suchbäume.</i> <p>[Kompetenzen laut KC: PK1.3, PK2.2 IK1.3, IK2.2]</p>	<p>ca. 4 Wochen in 13.2 (entfällt auf gA)</p> <p>Mögliche Ergänzungen:</p> <ul style="list-style-type: none"> - Algorithmen auf Graphen - genetische Algorithmen - Komplexitätstheorie