

NIKOLAUS-EHLEN
GYMNASIUM

SCHULINTERNER LEHRPLAN ZUM KERNLEHRPLAN FÜR DIE SEKUNDARSTUFE I

im Fach Informatik

Inhaltsverzeichnis

1	Rahmenbedingungen der fachlichen Arbeit am NEG.....	3
1.1	Die Fachgruppe Informatik.....	3
1.2	Bedingungen des Unterrichts.....	3
1.3	Vorgaben für den Informatikunterricht in der Sekundarstufe I.....	3
2	Entscheidungen zum Unterricht.....	4
2.1	Unterrichtsvorhaben	4
2.2	Übersichtsraster Unterrichtsvorhaben	5
2.3	Konkretisierte Unterrichtsvorhaben	11
2.3.1	Unterrichtsvorhaben in der Klasse 5.1	11
2.3.2	Unterrichtsvorhaben in der Klasse 5.2	13
2.3.3	Unterrichtsvorhaben in der Klasse 6.1	16
2.3.4	Unterrichtsvorhaben in der Klasse 6.2	19
2.3.5	Unterrichtsvorhaben in der Klasse 9.....	21
2.3.6	Unterrichtsvorhaben in der Klasse 10.....	29
3	Grundsätze der fachmethodischen und fachdidaktischen Arbeit im Informatikunterricht der gymnasialen Unter- und Mittelstufe	37
3.1	Vereinbarungen zur fachmethodischen und fachdidaktischen Arbeit im Informatikunterricht.....	37
3.2	Individualisierungsprozesse im Rahmen des Informatikunterrichts.....	37
3.2.1	Individuelle Aufgabenstellungen im Rahmen des Unterrichts.....	38
3.2.2	Erweiterte Projektarbeit.....	38
4	Grundsätze der Leistungsbewertung und Leistungsrückmeldung.....	38
4.1	Beurteilungsbereiche und Überprüfungsformen.....	38
4.1.1	Beurteilungsbereich Sonstige Mitarbeit.....	38
4.1.2	Beurteilungsbereich Kursarbeiten.....	39
4.2	Grundsätze der Leistungsrückmeldung und Beratung.....	40
5	Lehr- und Lernmittel.....	40
6	Entscheidungen zu fach- und unterrichtsübergreifenden Fragen	41
7	Qualitätssicherung und Evaluation	41

1 Rahmenbedingungen der fachlichen Arbeit am NEG

1.1 Die Fachgruppe Informatik

Die Fachgruppe Informatik umfasst derzeit zwei Lehrkräfte. Beide Lehrkräfte besitzen die Fakultas für die Sekundarstufe I und die Sekundarstufe II. Zwischen den Lehrkräften besteht eine enge Zusammenarbeit sowie ein reger Material- und Erfahrungsaustausch.

Die Fachkonferenz tritt mindestens einmal pro Schulhalbjahr zusammen, um notwendige Absprachen zu treffen. In der Regel nehmen auch jeweils bis zu zwei Vertreter der Elternschaft sowie der Schülerschaft beratend an den Sitzungen teil.

1.2 Bedingungen des Unterrichts

Der Unterricht im Fach Informatik wird als fester Bestandteil des Stundenplans in den Klassen 5 und 6, im Wahlpflicht II-Bereich für Teile der Mittelstufe (9. und 10. Klasse) und in Grundkursen in der Oberstufe angeboten. Um einen guten Übergang von der Sekundarstufe I in die Sekundarstufe II sowie die Möglichkeit eines Neueinstiegs zu gewährleisten, werden im Rahmen des Unterrichts verschiedene Programmiersprachen und unterschiedliche theoretische Aspekte des Fachs Informatik behandelt.

In der Sekundarstufe I umfasst der Unterricht im Fach Informatik für die Klassen 5 und 6 je eine 45-Minuten-Stunde pro Woche (ohne Daltonstunde) und im Wahlpflicht II-Bereich drei 45-Minuten-Stunden pro Woche. Diese sind aufgeteilt in eine Doppelstunde á 90 Minuten, in der der Fachunterricht für den Kurs stattfindet, sowie eine Daltonstunde á 45 Minuten, in der die Schülerinnen und Schüler eine gestellte Aufgabe eigenverantwortlich bearbeiten sollen.

Der Fachunterricht findet in der Regel in einem der drei Computerräume der Schule statt, die jeweils mit 16 Schüler- und einem Lehrerrechner ausgestattet sind. Je nach Kursgröße arbeiten die Mitglieder der Lerngruppe in Einzel- oder Partnerarbeit an verschiedenen Problemstellungen.

Unabhängig von der Jahrgangsstufe und der Wahl des Schulfachs Informatik, werden alle Schülerinnen und Schüler der Sekundarstufen I und II motiviert, an dem jährlich stattfindenden Wettbewerb „Informatik-Biber“ teilzunehmen.

1.3 Vorgaben für den Informatikunterricht in der Sekundarstufe I

Für das Schulfach Informatik in den Klassen 5 und 6 orientiert sich dieses schulinterne Curriculum am zuvor erteilten Digi-Unterricht, da zum momentanen Zeitpunkt noch kein Kernlehrplan für den Informatikunterricht in der Erprobungsstufe erschienen ist.

Im Wahlpflichtfach-Bereich des Faches Informatik steht ein aktueller Kernlehrplan für die Sekundarstufe I an Gymnasien in NRW aus dem Jahr 2019 als Grundlage zur Verfügung.

Als weitere Grundlage für diesen schulinternen Lehrplan konnten die Bildungsstandards der Gesellschaft für Informatik [GI-Bildungsstandards, 2008] herangezogen werden.

2 Entscheidungen zum Unterricht

2.1 Unterrichtsvorhaben

Die Unterrichtsvorhaben werden auf zwei Ebenen, der Übersichts- und der Konkretisierungsebene, beschrieben.

Im *Übersichtsraster Unterrichtsvorhaben* wird die für alle Lehrerinnen und Lehrer gemäß Fachkonferenzbeschluss verbindliche Verteilung der Unterrichtsvorhaben dargestellt. Das Übersichtsraster dient dazu, für die einzelnen Jahrgangsstufen allen Akteuren einen schnellen Überblick über Themen bzw. Fragestellungen der Unterrichtsvorhaben unter Angabe besonderer Schwerpunkte in den Inhalten und in der Kompetenzentwicklung zu verschaffen. Dadurch soll verdeutlicht werden, welches Wissen und welche Fähigkeiten in den jeweiligen Unterrichtsvorhaben besonders gut zu erlernen sind und welche Aspekte deshalb im Unterricht hervorgehoben thematisiert werden sollten. In der Hinweisspalte des Übersichtsrasters werden u. a. mögliche Entlastungen im Hinblick auf thematische Fokussierungen und interne Verknüpfungen ausgewiesen.

Der ausgewiesene Zeitbedarf versteht sich als grobe Orientierungsgröße, die nach Bedarf über- oder unterschritten werden kann. Um Spielraum für Vertiefungen, besondere Schülerinteressen, aktuelle Themen bzw. die Erfordernisse anderer besonderer Ereignisse (z. B. Klassenfahrten o. Ä.) zu erhalten, wurden im Rahmen dieses schulinternen Lehrplans ca. 75 Prozent der Bruttounterrichtszeit verplant.

In den *konkretisierten Unterrichtsvorhaben* werden die Unterrichtsvorhaben und die diesbezüglich getroffenen Absprachen detaillierter dargestellt. In dieser Darstellung wird ebenfalls deutlich, welche Kompetenzen als Schwerpunkt im Fokus stehen, aber auch, welche Kompetenzen im Unterrichtsgeschehen begleitend angesprochen werden. In der Konkretisierung der jeweiligen Unterrichtsvorhaben wird das Zusammenspiel der Kompetenzbereiche verdeutlicht. Außerdem werden Absprachen und Hinweise zur Vernetzung, Entlastung und Schwerpunktsetzung näher ausgeführt. Abweichungen von Vorgehensweisen der konkretisierten Unterrichtsvorhaben über die als verbindlich bezeichneten notwendigen Absprachen hinaus sind im Rahmen der pädagogischen Freiheit der Lehrkräfte möglich. Sicherzustellen bleibt allerdings auch hier, dass im Rahmen der Umsetzung der Unterrichtsvorhaben insgesamt alle Kompetenzerwartungen des Kernlehrplans Berücksichtigung finden.

2.2 Übersichtsraster Unterrichtsvorhaben

Unterrichtsvorhaben der Klasse 5.1 (ehemals Digi)			
Thema und Zeitbedarf	Inhaltsfelder	Kompetenzbereiche	Hinweise
<p>Einstieg in die Nutzung von Hard- / und Software</p> <p>Grundsätzlicher Umgang mit Betriebs- / und Dateisystem</p> <p>(ca. 8 Wochen)</p>	<ul style="list-style-type: none"> - Information und Daten - Informatiksysteme 	<ul style="list-style-type: none"> - Darstellen und Interpretieren - Argumentieren - Kommunizieren und Kooperieren 	Anhand der jeweils aktuellen Hardwareausstattung der Schule
<p>Einführung in eine Textverarbeitung</p> <p>(ca. 8 Wochen)</p>	<ul style="list-style-type: none"> - Information und Daten 	<ul style="list-style-type: none"> - Darstellen und Interpretieren - Kommunizieren und Kooperieren 	Microsoft Word oder vergleichbare Open Source Software
<p>Teilnahme am Informatik-Biber-Wettbewerb</p> <p>(ca. 2 Wochen)</p>	<ul style="list-style-type: none"> - Algorithmen - Informatiksysteme - Information und Daten - Informatik, Mensch und Gesellschaft - Formale Sprachen 	<ul style="list-style-type: none"> - Darstellen und Interpretieren - Argumentieren - Kommunizieren und Kooperieren - Modellieren und Implementieren 	Zeitraum meistens im November

Unterrichtsvorhaben der Klasse 5.2 (ehemals Digi)

Thema und Zeitbedarf	Inhaltsfelder	Kompetenzbereiche	Hinweise
Einstieg in die Nutzung des Internets: - Aufbau - Zugangsmöglichkeiten - Wie finde ich mich zurecht? (ca. 6 Wochen)	- Information und Daten - Informatiksysteme - Informatik, Mensch und Gesellschaft	- Argumentieren - Kommunizieren und Kooperieren - Modellieren und Implementieren	Anhand der jeweils aktuellen Hardwareausstattung der Schule
Rechtliche Aspekte: - Datenschutz - Copyright (ca. 6 Wochen)	- Information und Daten - Informatiksysteme - Informatik, Mensch und Gesellschaft	- Argumentieren - Kommunizieren und Kooperieren	
Datensicherheit: - Passwörter - Verschlüsselungsverfahren - Schutz vor Phishing, etc. (ca. 6 Wochen)	- Algorithmen - Informatiksysteme - Information und Daten - Informatik, Mensch und Gesellschaft	- Darstellen und Interpretieren - Argumentieren - Kommunizieren und Kooperieren - Modellieren und Implementieren	

Unterrichtsvorhaben der Klasse 6.1 (ehemals Digi)			
Thema und Zeitbedarf	Inhaltsfelder	Kompetenzbereiche	Hinweise
Einführung in die Programmierung einer Platine und Steuerung verschiedener Sensoren (ca. 12 Wochen)	- Algorithmen - Informatiksysteme	- Modellieren und Implementieren - Kommunizieren und Kooperieren	Verwenden der Calliope-Mini-Platine in Verbindung mit dem Einsatz am Computer
Kennenlernen einer blockbasierten Programmiersprache am Beispiel von Scratch (ca. 5 Wochen)	- Algorithmen - Informatiksysteme	- Modellieren und Implementieren - Kommunizieren und Kooperieren	Einsatz von Computern
Teilnahme am Informatik-Biber-Wettbewerb (ca. 1 Woche)	- Algorithmen - Informatiksysteme - Information und Daten - Informatik, Mensch und Gesellschaft - Formale Sprachen	- Darstellen und Interpretieren - Argumentieren - Kommunizieren und Kooperieren - Modellieren und Implementieren	Zeitraum meistens im November

Unterrichtsvorhaben der Klasse 6.2 (ehemals Digi)			
Thema und Zeitbedarf	Inhaltsfelder	Kompetenzbereiche	Hinweise
Einführung in eine Tabellenkalkulation (ca. 8 Wochen)	- Information und Daten	- Darstellen und Interpretieren - Kommunizieren und Kooperieren	Microsoft Excel oder vergleichbare Open Source Software
Einführung in eine Präsentationssoftware (ca. 8 Wochen)	- Information und Daten	- Darstellen und Interpretieren - Kommunizieren und Kooperieren	Microsoft Powerpoint oder vergleichbare Open Source Software
Optional: Ausblick auf Mathematik-Software (ca. 2 Woche)	- Information und Daten	- Darstellen und Interpretieren - Kommunizieren und Kooperieren - Modellieren und Implementieren	z.B. mit Geogebra oder vergleichbarer Software

Unterrichtsvorhaben der Klasse 9			
Thema und Zeitbedarf	Inhaltliche Kompetenzen	Prozessbezogene Kompetenzen	Hinweise
Einführung in die Nutzung von Informatiksystemen und in grundlegende Begrifflichkeiten anhand der Programmiersprache Scratch (ca. 8 Wochen)	- Algorithmen - Informatiksysteme	- Modellieren und Implementieren - Kommunizieren und Kooperieren	Einsatz von Computern

Unterrichtsvorhaben der Klasse 9			
Thema und Zeitbedarf	Inhaltliche Kompetenzen	Prozessbezogene Kompetenzen	Hinweise
Vertiefung in die Nutzung von Informatiksystemen und Übertragen bekannter Strukturen aus der Einführung anhand der Programmiersprache Python (ca. 16 Wochen)	- Algorithmen - Informatiksysteme	- Modellieren und Implementieren - Kommunizieren und Kooperieren - Strukturieren und Vernetzen - Begründen und Bewerten	Einsatz von Computern
Kennenlernen von und Rechnen in anderen Stellenwertsystemen (ca. 6 Wochen)	- Information und Daten - Informatiksysteme	- Strukturieren und Vernetzen - Kommunizieren und Kooperieren - Darstellen und Interpretieren	Baut auf den Kenntnissen über Binärzahlen (Mathematik, Klasse 5) auf

Unterrichtsvorhaben der Klasse 10			
Thema und Zeitbedarf	Inhaltliche Kompetenzen	Prozessbezogene Kompetenzen	Hinweise
Weitere Vertiefung in die Nutzung von Informatiksystemen und Ausbau der Programmierfertigkeiten und -fähigkeiten anhand der Programmiersprache Python (ca. 16 Wochen)	- Algorithmen - Informatiksysteme - Information und Daten - Informatik, Mensch und Gesellschaft	- Modellieren und Implementieren - Kommunizieren und Kooperieren - Strukturieren und Vernetzen - Begründen und Bewerten	Einsatz von Computern Beinhaltet das Themengebiet „Datenverschlüsselung“
Einführung in die Aussagenlogik (ca. 8 Wochen)	- Information und Daten	- Strukturieren und Vernetzen - Kommunizieren und Kooperieren	Schließt an das Thema „Kennenlernen von und Rechnen in anderen Stellenwertsystemen“ aus Klasse 8 an

Unterrichtsvorhaben der Klasse 10			
Thema und Zeitbedarf	Inhaltliche Kompetenzen	Prozessbezogene Kompetenzen	Hinweise
Erstellen von Webseiten mit HTML und CSS ¹ (ca. 6 Wochen)	<ul style="list-style-type: none"> - Algorithmen - Informatiksysteme - Information und Daten - Informatik, Mensch und Gesellschaft 	<ul style="list-style-type: none"> - Modellieren und Implementieren - Kommunizieren und Kooperieren - Strukturieren und Vernetzen - Begründen und Bewerten 	Projekt: Erstellen einer eigenen Webseite
Einführung in formale Sprachen und Grammatiken ¹ (ca. 6 Wochen)	<ul style="list-style-type: none"> - Sprachen und Automaten 	<ul style="list-style-type: none"> - Darstellen und Interpretieren 	Bildet die Grundlagen für das Thema „endliche Automaten“ (Q-Phase)

¹ Optionales Thema

2.3 Konkretisierte Unterrichtsvorhaben


Im Folgenden werden die einzelnen Themen aus den Übersichtsrastern der Unterrichtsvorhaben durch Angabe der detaillierteren Unterrichtssequenzen, der zu erwerbenden Kompetenzen und entsprechenden Beispielen und Hinweisen konkretisiert.




2.3.1 Unterrichtsvorhaben in der Klasse 5.1

2.3.1.1 Thema „Einstieg in die Nutzung von Hard- und Software“

Leitfragen: Was ist der Unterschied zwischen Hard- und Software? Was ist das EVA-Prinzip? Wie bediene ich die zur Verfügung stehenden Computer? Welche Ein- und Ausgabegeräte gibt es? Was leistet ein Betriebssystem? Wie ist die Dateiverwaltung organisiert?

Vorhabensbezogene Konkretisierung: Im Rahmen der Einführung in das Unterrichtsfach Informatik werden die wesentlichen Bestandteile, die bei der Datenverarbeitung eine Rolle spielen, überblicksartig nahegebracht. Dabei werden grundlegende Hardwarekomponenten benannt, nach ihren Funktionen kategorisiert und ihr Zusammenwirken untersucht. Auch die Bedeutung und Anwendung ersten typischer und bekannter Softwarekomponenten sollen an dieser Stelle behandelt werden.

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Hardwarekomponenten und ihre Funktionen kennenlernen	... benennen die verschiedenen Hardwarekomponenten korrekt und können ihre Funktionen beschreiben. ... verstehen das EVA-Prinzip und können diverse Hardwarekomponenten korrekt zuordnen.	1.1 Medianausstattung (Hardware) - Medianausstattung kennen, auswählen und anwenden 

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Verwenden und verstehen grundlegender Funktionen eines Computers	<p>... können den Schulcomputer sicher bedienen.</p> <p>... lernen verschiedene Verwendungsmöglichkeiten eines Computers kennen.</p> <p>... finden sich in der Dateioorganisation gut zurecht.</p>	<p>1.1 Medienaustattung (Hardware)</p> <p>- mit Medienaustattung verantwortungsvoll umgehen </p> <p>1.2 Digitale Werkzeuge</p> <p>- verschiedene digitale Werkzeuge und deren Funktionsumfang kennen und einsetzen </p> <p>1.3 Datenorganisation </p>

2.3.1.2 Thema „Einführung in eine Textverarbeitung“

Leitfragen: Wozu wird ein Textverarbeitungsprogramm benötigt? Welche Möglichkeiten bietet ein Textverarbeitungsprogramm? Welche Textverarbeitungsprogramme gibt es?

Vorhabensbezogene Konkretisierung: Die Schülerinnen und Schüler erlernen anhand eines ausgewählten Produktes die grundlegenden Funktionen und Möglichkeiten eines Textverarbeitungsprogramms. Dazu gehören das Erstellen, Speichern und Laden von Texten sowie verschiedene Formatierungs-, Einstellungs- und Einfügeoptionen.

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Grundlagen einer Textverarbeitung	... lernen grundlegende Bedienelemente wie Texteingabe, Laden und Speichern kennen. ... machen sich schrittweise mit verschiedenen Formatierungsarten vertraut.	1.2 Digitale Werkzeuge - Einsatz eines Textverarbeitungsprogramms
Erweiterte Funktionen einer Textverarbeitung	... können ihre Dokumente mithilfe verschiedener Einstellungsmöglichkeiten anpassen. ... fügen ihren Dokumenten Bilder, Links etc. hinzu.	Verwenden von Word, LibreOffice.writer oder analogen Programmen



2.3.1.3 Thema „Teilnahme am Informatik-Biber-Wettbewerb“

Leitfragen: Wozu wird Informatik gebraucht? Wie kann ich durch logisches Denken Probleme lösen?

Vorhabensbezogene Konkretisierung: Im Rahmen einer Unterrichtsstunde sollen die Schülerinnen und Schüler der sechsten Klassen am Informatik-Biber-Wettbewerb teilnehmen. Hier kann man auf spielerische Art und Weise vielfältige Einblicke in die Anwendungen und Möglichkeiten der Informatik erhalten sowie seine Fähigkeiten im logischen Denken testen.



Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Teilnahme am Biber-Wettbewerb	... testen ihr logisches Denkvermögen. ... erhalten Einblicke in die Welt der Informatik.	Muss innerhalb des Wettbewerbszeitraums (Anfang/Mitte November) erfolgen

2.3.2 Unterrichtsvorhaben in der Klasse 5.2

2.3.2.1 Thema „Einstieg in die Nutzung des Internets“

Leitfragen: Was ist das Internet und wie ist es aufgebaut? Welche Zugangsmöglichkeiten habe ich? Was ist ein Browser? Wie finde ich mich im Internet zu-recht?



Vorhabensbezogene Konkretisierung: In der Einführungssequenz zum Bereich „Internet“ soll den Schülerinnen und Schülern die grundlegende Funktionsweise des Internets vermittelt und eine erste Orientierung im „Worldwide Web“ gegeben werden. Hierzu lernen sie unterschiedliche Browser und Suchmaschinen kennen und testen deren vielfältige Funktionen aus.

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Aufbau und Funktionsweise des Internets	... lernen die grundlegende Struktur und Funktionsweise des Internets kennen.	6.1 Prinzipien der digitalen Welt - Grundlegende Prinzipien und Funktionsweisen der digitalen Welt kennen und verstehen 
Verwenden verschiedener Programme zur Orientierung im Internet	... können erklären, was ein Internetbrowser ist. ... verwenden eine Suchmaschine, um an die gewünschten Informationen zu gelangen. ... kennen diverse Browser und Suchmaschinen.	2.1 Informationsrecherche - Informationsrecherchen zielgerichtet durchführen und Suchstrategien anwenden 

2.3.2.2 Thema „Rechtliche Aspekte bei der Nutzung des Internets“

Leitfragen: Was bedeuten die Begriffe „Datenschutz“ und „Copyright“? Wie gelangen Informationen über mich ins Internet? Welche rechtlichen Aspekte muss ich im Internet beachten? Was muss ich beim Verwenden von Informationen und Materialien aus dem Internet beachten?



Vorhabensbezogene Konkretisierung: Die Schülerinnen und Schüler beleuchten ihr Verhalten im Internet vom persönlichen und rechtlichen Standpunkt aus gesehen. Zum einen werden Aspekte des Datenschutzes behandelt, zum anderen wird geprüft, in wie weit Daten aus dem Internet kopiert, verändert bzw. weiterverbreitet werden dürfen.

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Aspekte des Datenschutzes	<p>... tragen Sorge für die Daten, die sie im Internet hinterlassen.</p> <p>... sind sich der Konsequenzen, die aus der Internetnutzung entstehen, bewusst.</p>	<p>1.4 Datenschutz und Informationssicherheit</p> <p>- Verantwortungsvoller Umgang mit Daten; Datenschutz</p> 
Umgang mit Daten aus dem Internet	<p>... können zwischen verschiedenen Arten urheberrechtlicher Lizenzen unterscheiden.</p> <p>... wissen, wie sie mit Daten - je nach Lizenztyp - umgehen müssen.</p>	<p>4.4. Rechtliche Grundlagen</p> <p>- Grundlagen des Urheber- und Nutzungsrechts</p> 

2.3.2.3 Thema „Datensicherheit online und offline“

Leitfragen: Wie schütze ich wichtige Daten? Was ist Kryptographie und wozu braucht man sie? Welche Verschlüsselungsmethoden gibt es? Welche Gefahren gibt es im Internet?

Vorhabensbezogene Konkretisierung: Sowohl offline als auch online sollten wichtige Daten gut geschützt werden. Dazu lernen die Schülerinnen und Schüler verschiedene Möglichkeiten kennen, wie Informationen verschlüsselt werden können. Mithilfe des Beispiels der „Cäsar-Kodierung“ kann ein einfaches Verschlüsselungsverfahren konkret veranschaulicht und nachvollzogen werden. Außerdem soll auf die Gefahren des Datendiebstahls im Internet hingewiesen und aufmerksam gemacht werden.




Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Datensicherheit und Verschlüsseln von Daten	<p>... lernen die Cäsar-Kodierung als einfache und anschauliche Verschlüsselungsmethode kennen.</p> <p>... können weitere Verschlüsselungsmethoden benennen und nachvollziehen.</p> <p>... entwickeln einen bewussten Umgang mit persönlichen Daten.</p>	<p>1.4 Informationssicherheit</p> <p>- Beachten der Informationssicherheit </p> <p>6.2 Algorithmen erkennen</p> <p>- (Verschlüsselungs-) Algorithmen nachvollziehen und reflektieren </p>

2.3.3 Unterrichtsvorhaben in der Klasse 6.1

2.3.3.1 Thema „Einführung in die Programmierung einer Platine und Steuerung verschiedener Sensoren“

Leitfragen: Was ist Programmieren? Wie kommt der Quelltext zur Platine? Welche Möglichkeiten stellt eine Platine mit ihren Sensoren zur Verfügung? Wie kann ich die Fähigkeiten einer Platine beeinflussen?



Vorhabensbezogene Konkretisierung: Mithilfe der Calliope-Mini-Platine, die eine Vielzahl von Sensoren bereitstellt, und dem „Makecode“-Editor soll ein erster Einstieg in die Programmierung einer Platine erfolgen. Hierzu sollen die Schülerinnen und Schüler der sechsten Klassen über das Bearbeiten verschiedener, spannender Projekte an die zur Verfügung stehenden Möglichkeiten spielerisch und auf experimentelle Art und Weise herangeführt werden. Durch den intuitiv zu bedienenden Editor erhält die Lerngruppe auch einen Einblick in Programmabläufe.

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Kennenlernen der Calliope-Mini-Platine	<p>... lernen den Aufbau, die Bestandteile und Sensoren der Calliope-Mini-Platine kennen.</p> <p>... testen die bereits vorhandenen Funktionen der Calliope-Mini-Platine.</p>	<p>1.1 Medienausstattung (Hardware)</p> <p>- Hardware anwenden, verantwortungsvoller Umgang </p> <p>Calliope-Mini-Orakel, Schere-Stein-Papier, das kleine 1x1, Krach-o-Meter</p>
Einstieg in die Programmierung der Calliope-Mini-Platine mithilfe des Makecode-Editors	<p>... stellen eine Verbindung zwischen PC und Calliope-Mini her und übertragen Daten.</p> <p>... bearbeiten erste, einfachere Projekte zur Steuerung des Calliope-Mini durch Erstellen von kurzen Quellcode-Sequenzen im Editor-Fenster.</p>	<p>6.3 Modellieren und Implementieren</p> <p>- Lösungsstrategien entwickeln und algorithmische Sequenz planen </p> <p>- Programmieren</p> <p>Nutzen der LED-Anzeige, Abspielen von Tönen, etc.</p>
Vertiefung der Programmierkenntnisse des Calliope-Mini	<p>... bearbeiten weiterführende Projekte zur Steuerung des Calliope-Mini.</p> <p>... führen kleine Experimente mithilfe der Calliope-Mini-Platine durch und werten ihre Ergebnisse aus.</p> <p>... nutzen Alltagsgegenstände zur Erweiterung der Fähigkeiten des Calliope-Mini.</p> <p>... entwickeln eigene Ideen im Team und in der Gruppe und setzen diese gemeinsam um.</p>	<p>6.3 Modellieren und Implementieren</p> <p>- Lösungsstrategien entwickeln und algorithmische Sequenz planen </p> <p>- Programmieren</p> <p>Verwenden von Krokodilklemmen und anderen Hilfsmitteln</p>

2.3.3.2 Thema „Kennenlernen einer blockbasierten Programmiersprache am Beispiel von Scratch“

Leitfragen: Wie programmiert man ohne Platine? Welche Möglichkeiten bietet eine visuelle Programmiersprache? Wie kann ein einfaches Spiel selbst programmiert und erweitert werden?

Vorhabensbezogene Konkretisierung: Da die Schülerinnen und Schüler sich zu diesem Zeitpunkt bereits mit dem Makecode-Editor vertraut gemacht haben, soll in den letzten Unterrichtseinheiten hauptsächlich ein Ausblick auf das Programmieren mit der visuellen Sprache Scratch erfolgen, deren Aufbau und Funktionsweise dem schon bekannten Editor ähnelt. Durch die schülerfreundliche Programmieroberfläche kann die Lerngruppe nach kurzer Anleitung hier schnell selbstständig erste Lernerfolge erzielen. Hierdurch sollen die Schülerinnen und Schüler einen kurzen Einblick in die Unterrichtsthemen des WP II-Bereichs im Fach Informatik erhalten und zum weiterprogrammieren animiert und motiviert werden.

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Einstieg in die Programmiersprache Scratch	<ul style="list-style-type: none"> ... sammeln erste Erfahrungen in einer visuellen Programmiersprache. ... ändern verschiedene Eigenschaften und testen die Auswirkungen. 	<p>6.2 Algorithmen erkennen</p> <ul style="list-style-type: none"> - Algorithmische Muster und Strukturen in verschiedenen Kontexten erkennen und nachvollziehen <p style="text-align: right;"></p> <p>6.3 Modellieren und Programmieren</p> <ul style="list-style-type: none"> - Lösungsstrategien entwickeln und algorithmische Sequenz planen - Programmieren <p style="text-align: right;"></p> <p>Entwerfen von Kostümen und Bühnen, einfache Projekte mit kurzem Quelltext</p>

2.3.3.3 Thema „Teilnahme am Informatik-Biber-Wettbewerb“

Leitfragen: Wozu wird Informatik gebraucht? Wie kann ich durch logisches Denken Probleme lösen?

Vorhabensbezogene Konkretisierung: Im Rahmen einer Unterrichtsstunde sollen die Schülerinnen und Schüler der sechsten Klassen am Informatik-Biber-Wettbewerb teilnehmen. Hier kann man auf spielerische Art und Weise vielfältige Einblicke in die Anwendungen und Möglichkeiten der Informatik erhalten sowie seine Fähigkeiten im logischen Denken testen.


Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Teilnahme am Biber-Wettbewerb	... testen ihr logisches Denkvermögen. ... erhalten Einblicke in die Welt der Informatik.	Muss innerhalb des Wettbewerbszeitraums (Anfang/Mitte November) erfolgen

2.3.4 Unterrichtsvorhaben in der Klasse 6.2

2.3.4.1 Thema „Einführung in eine Tabellenkalkulation“

Leitfragen: Wozu wird ein Tabellenkalkulationsprogramm benötigt? Welche Möglichkeiten bietet ein Tabellenkalkulationsprogramm? Welche Tabellenkalkulationsprogramme gibt es?


Vorhabensbezogene Konkretisierung: Die Schülerinnen und Schüler erlernen anhand eines ausgewählten Produktes die grundlegenden Funktionen und Möglichkeiten eines Tabellenkalkulationsprogramms. Dazu gehört auch die Ausführung mathematischer Rechnungen und das Erstellen verschiedenartiger Diagramme.

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Einführung in eine Tabellenkalkulation	<p>... können erklären, wozu Tabellenkalkulationsprogramme verwendet werden.</p> <p>... beherrschen grundlegende Fähigkeiten und Fertigkeiten im Umgang mit einer Tabellenkalkulation.</p> <p>... können mithilfe einer Tabellenkalkulation Rechnungen durchführen und Diagramme erstellen.</p>	<p>1.2 Digitale Werkzeuge</p> <p>- Einsetzen eines Tabellenkalkulationsprogramms</p> <p>Verwenden von Excel, LibreOffice.calc oder analogen Programmen</p> 

2.3.4.2 Thema „Einführung in eine Präsentationssoftware“

Leitfragen: Wozu wird ein Präsentationsprogramm benötigt? Welche Möglichkeiten bietet ein Präsentationsprogramm? Welche Präsentationsprogramme gibt es?

Vorhabensbezogene Konkretisierung: Die Schülerinnen und Schüler erlernen anhand eines ausgewählten Produktes die grundlegenden Funktionen und Möglichkeiten einer Präsentationssoftware. Anhand konkreter Aufgabenstellungen sollen verschiedene Präsentationen erstellt und diverse Einstellungen und Eigenschaften ausprobiert werden. Außerdem werden Regeln zum Erstellen eigener Präsentationen diskutiert.

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Einführung in eine Präsentationssoftware	<p>... erstellen eine eigene Präsentation zu einem ausgewählten Thema.</p> <p>... halten sich an die vereinbarten Regeln bei der Präsentationserstellung.</p> <p>... kennen verschiedene Möglichkeiten, eine Präsentation sinnvoll und ansprechend zu gestalten.</p>	<p>1.2 Digitale Werkzeuge</p> <p>- Einsetzen eines Präsentationsprogramms</p> <p>Verwenden von Powerpoint, LibreOffice.base oder analogen Programmen</p> 

2.3.4.3 Thema „Arbeiten mit Mathesoftware“

Leitfragen: Was ist eine „dynamische Geometriesoftware“? Wie kann ich Graphen und Zeichnungen am Computer erstellen?

Vorhabensbezogene Konkretisierung: Mithilfe „dynamischer Geometriesoftware“ können Zeichnungen von geometrischen Figuren, Körpern und Konstruktionen sowie Graphen digital realisiert werden. Im Rahmen dieser optionalen Unterrichtssequenz sollen die Schülerinnen und Schüler einen kleinen Einblick in die vielfältigen Möglichkeiten erhalten.

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Kennenlernen von Mathesoftware	... haben eine Idee von den Möglichkeiten bei der Verwendung dynamischer Geometriesoftware.	1.2 Digitale Werkzeuge - Einsetzen von Mathematiksoftware z.B. Geogebra







2.3.5 Unterrichtsvorhaben in der Klasse 9

2.3.5.1 Thema: „Einführung in die Nutzung von Informatiksystemen und in grundlegende Begrifflichkeiten anhand der Programmiersprache Scratch“

Leitfragen: Was ist Programmieren? Wie setzt ein Computer eine Befehlsfolge um? Wie ist der Quelltext eines Computerprogramms aufgebaut? Aus welchen Bestandteilen kann ein Programm zusammengesetzt sein?

Vorhabensbezogene Konkretisierung: Anhand der intuitiv zu bedienenden Programmiersprache /-oberfläche „Scratch“ soll den Schülerinnen und Schülern ein leichter und schneller Einstieg in das Themenfeld der Programmierung ermöglicht werden. Über die Einführung immer komplexer werdender, spielerischer Problemstellungen sollen die Kursmitglieder dazu motiviert werden, Programmabläufe kennenzulernen und zu analysieren sowie neue Programmierkonzepte und -strukturen in ihre Programme einzubauen.

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Einführung in die Programmieroberfläche von Scratch	<p>... lernen den grundlegenden Umgang mit einer Programmiersprache kennen.</p> <p>... finden sich in der Benutzeroberfläche von Scratch zurecht.</p> <p>... erstellen und testen einfache Algorithmen.</p> <p>... eignen sich mithilfe von Tutorials neues Wissen an.</p>	<p>6.2 Algorithmen erkennen</p> <ul style="list-style-type: none"> - Algorithmische Muster und Strukturen in verschiedenen Kontexten erkennen und nachvollziehen <p style="text-align: right;"></p> <p>6.3 Modellieren und Programmieren</p> <ul style="list-style-type: none"> - Lösungsstrategien entwickeln und algorithmische Sequenz planen - Programmieren <p>Spiele „Flucht vor dem Drachen“, „Pong“</p> <p style="text-align: right;"></p>



Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
<p>Grundlegende Elemente einer (beliebigen) Programmiersprache und Scratch-spezifische Elemente</p>	<p>... lernen grundlegende Bestandteile von Programmen wie Schleifen, Verzweigungen, Variablen etc. kennen.</p> <p>... befassen sich mit Scratch-spezifischen Bestandteilen wie Kostümen, Spezialeffekten, Nachrichten etc.</p> <p>... modifizieren und ergänzen Quelltexte von Programmen.</p> <p>... verwenden bei der Implementierung algorithmische Bausteine.</p> <p>... begründen Vorgehensweisen bei der Modellierung informatischer Sachverhalte.</p> <p>... erkennen Reihenfolgen in Handlungsabläufen.</p>	<p>6.2 Algorithmen erkennen</p> <ul style="list-style-type: none"> - Algorithmische Muster und Strukturen in verschiedenen Kontexten erkennen und nachvollziehen <p>6.3 Modellieren und Programmieren</p> <ul style="list-style-type: none"> - Lösungsstrategien entwickeln und algorithmische Sequenz planen - Programmieren <p>Spiele „Sternsammler“, „Käse-Labyrinth“, „Krieg der Kreise“, „Affespring“, „Wüstenralley“</p>  



2.3.5.2 Thema: „Vertiefung in die Nutzung von Informatiksystemen und Übertragen bekannter Strukturen aus der Einführung anhand der Programmiersprache Python“





Für die folgenden Ausführungen zum Programmieren gilt: Die Programmiersprache Python kann ggf. auch durch eine äquivalente Programmiersprache (wie z.B. JavaScript etc.) ersetzt werden. Die angegebenen Unterrichtsinhalte und -konzepte sind unabhängig von der eingesetzten Programmiersprache.

Leitfragen: Was ist eine höhere (objektorientierte) Programmiersprache? Wie programmiert man klassisch, also ohne vorhandene, vorgefertigte Blöcke?

Vorhabensbezogene Konkretisierung: Mithilfe der (objektorientierten) Programmiersprache „Python“ sollen die Schülerinnen und Schüler ihre bisherigen Kenntnisse auf eine mächtigere Programmiersprache übertragen und kontinuierlich erweitern. Der Übergang zu Python erfolgt dabei zunächst durch Anwenden von Zeichenbefehlen („Turtlegrafik“). Anhand verschiedener Aufgabenstellungen sollen die Kursmitglieder schrittweise immer neue (Zeichen-)Befehle, Strukturen und Konzepte in Python kennenlernen.

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Einführung in die Programmieroberfläche von Python	<ul style="list-style-type: none"> ... lernen den grundlegenden Aufbau der Programmiersprache Python kennen. ... achten auf eine korrekt angewendete Syntax. ... verwenden verschiedene Zeichenbefehle zur Lösung einer Aufgabenstellung ... erstellen und testen einfache Programme mit „Turtlegrafik“. ... lernen das eigenständige Anwenden und Kombinieren bekannter und neuer Befehle. 	<p>6.2 Algorithmen erkennen</p> <ul style="list-style-type: none"> - Algorithmische Muster und Strukturen in verschiedenen Kontexten erkennen und nachvollziehen  <p>6.3 Modellieren und Programmieren</p> <ul style="list-style-type: none"> - Lösungsstrategien entwickeln und algorithmische Sequenz planen - Programmieren <p>Erstellen von verschiedenen Zeichnungen</p> 

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Variablen, Wertzuweisungen, eigene Funktionen, Verzweigungen und Schleifen in Python	<ul style="list-style-type: none"> ... lernen, wie man in Python Variablen deklariert, initialisiert und durch Wertzuweisungen verändert. ... binden Benutzereingaben in Programme ein und verallgemeinern so ihre Lösungen. ... programmieren eigene Funktionen mit und ohne Parameter. ... übertragen ihr Wissen über Verzweigungen und Schleifen auf die neue Programmiersprache. ... lernen verschiedene Ausprägungen von Verzweigungen und Schleifen kennen. ... überprüfen verschiedene Lösungen hinsichtlich Übersichtlichkeit und Effizienz. 	<p>6.2 Algorithmen erkennen</p> <ul style="list-style-type: none"> - Algorithmische Muster und Strukturen in verschiedenen Kontexten erkennen und nachvollziehen <p>6.3 Modellieren und Programmieren</p> <ul style="list-style-type: none"> - Lösungsstrategien entwickeln und algorithmische Sequenz planen - Programmieren <p>Erstellen von immer komplexer werdenden Zeichnungen mithilfe von Verzweigungen und Schleifen etc.</p> <div style="text-align: right;">   </div>

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Programmierverfahren: Top-Down und Bottom-Up	<p>... lernen die Unterschiede zwischen den Programmiermethoden „Top-Down“ und „Bottom-Up“ kennen.</p> <p>... setzen ihre Kenntnisse in den verschiedenen Programmierverfahren ein, um entsprechende Aufgabenstellungen zu bearbeiten.</p>	<p>6.2 Algorithmen erkennen</p> <ul style="list-style-type: none"> - Algorithmische Muster und Strukturen in verschiedenen Kontexten erkennen und nachvollziehen <p>6.3 Modellieren und Programmieren</p> <ul style="list-style-type: none"> - Lösungsstrategien entwickeln und algorithmische Sequenz planen - Programmieren <p>„Ying-Yang-Symbol“</p>  
Operatoren für Zeichenketten in Python	<p>... lernen erste Befehle zum Einlesen, Verändern und Ausgeben von Zeichenketten kennen.</p> <p>... erstellen erste Programme, die auf Texteingabe reagieren.</p>	<p>6.2 Algorithmen erkennen</p> <ul style="list-style-type: none"> - Algorithmische Muster und Strukturen in verschiedenen Kontexten erkennen und nachvollziehen <p>6.3 Modellieren und Programmieren</p> <ul style="list-style-type: none"> - Lösungsstrategien entwickeln und algorithmische Sequenz planen - Programmieren <p>Einfaches Quiz-Spiel</p>  

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Zufallsgenerator in Python	<p>... setzen den Zufallsgenerator ein, um ganze Zufallszahlen zu erzeugen.</p> <p>... verwenden Zufallszahlen zur Lösung von Problemen und Erstellung von Spielen.</p>	<p>6.2 Algorithmen erkennen</p> <ul style="list-style-type: none"> - Algorithmische Muster und Strukturen in verschiedenen Kontexten erkennen und nachvollziehen <p>6.3 Modellieren und Programmieren</p> <ul style="list-style-type: none"> - Lösungsstrategien entwickeln und algorithmische Sequenz planen - Programmieren <p>„Random-Walk“, „Torkelnde Turtle“</p>



2.3.5.3 Thema: „Kennenlernen von und Rechnen in anderen Stellenwertsystemen“

Leitfragen: Wie verarbeitet ein Computer Informationen? Wie rechnet man mit binären Zahlen?

Vorhabensbezogene Konkretisierung: In diesem Modul geht es darum den Schülerinnen und Schülern aufzuzeigen, wie ein Computer grundlegend Informationen verarbeitet. Dazu werden die Kenntnisse über binäre Zahlen aus Klasse 5 wiederholt und um die Darstellung negativer Binärzahlen und das Rechnen mit Binärzahlen erweitert. Außerdem soll an dieser Stelle das Hexadezimalsystem eingeführt werden.

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Wiederholung der Kenntnisse über Binärzahlen	... frischen anhand von Beispielen und Übungsaufgaben ihre Kenntnisse über Binärzahlen auf.	Umwandlungen von Binärzahlen in Dezimalzahlen (und umgekehrt)
Negative Binärzahlen und Rechnen mit Binärzahlen	<p>... lernen den Zahlenkreis als logische Veranschaulichung negativer Binärzahlen kennen.</p> <p>... erarbeiten sich das Umwandeln negativer Binärzahlen in Dezimalzahlen (und umgekehrt)</p> <p>... lernen einfache Rechnungen mit Binärzahlen kennen.</p>	<p>Zahlenkreis, Addition von Binärzahlen</p> <p>Optional: Weitere Rechenarten mit Binärzahlen</p>
Das Hexadezimalsystem	... lernen das Hexadezimalsystem als weiteres Stellenwertsystem kennen.	Umwandlungen von Hexadezimalzahlen in Dezimalzahlen (und umgekehrt)

2.3.6 Unterrichtsvorhaben in der Klasse 10

2.3.6.1 Thema: „Weitere Vertiefung in die Nutzung von Informatiksystemen und Ausbau der Programmierfertigkeiten und -fähigkeiten anhand der Programmiersprache Python“

Leitfragen: Wie lassen sich Programme noch weiter verbessern? Was sind Objekte und Methoden? Wie kann man mithilfe einer Programmiersprache Text verarbeiten? Wie lassen sich Informationen möglichst sicher austauschen?

Vorhabensbezogene Konkretisierung: Im Rahmen des Informatikunterrichts der neunten Klasse soll die Programmierkenntnisse der Schülerinnen und Schüler in Python noch weiter vertieft werden. Zu diesem Zweck werden innerhalb dieses Moduls Funktionen mit Rückgabewert, Objekte und Methoden sowie der Umgang mit Zeichenketten im Mittelpunkt des Informatikunterrichts zum Thema „Programmierung“ stehen. Speziell wird in diesem Bereich auch das Thema „Kodierung von Daten“ behandelt.

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Funktionen mit Rückgabewert	<p>... unterscheiden zwischen Funktionen mit und ohne Rückgabewert.</p> <p>... definieren eigene Funktionen, die Zahlenwerte oder Zeichenketten zurückgeben.</p> <p>... modellieren mithilfe von Funktionen bekannte mathematische Operatoren.</p>	<p>6.2 Algorithmen erkennen</p> <ul style="list-style-type: none"> - Algorithmische Muster und Strukturen in verschiedenen Kontexten erkennen und nachvollziehen <p>6.3 Modellieren und Programmieren</p> <ul style="list-style-type: none"> - Lösungsstrategien entwickeln und algorithmische Sequenz planen - Programmieren <p>Flächeninhalte und Umfänge von geom. Figuren, Nachfolger/Vorgänger, Fakultät</p>





Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
<p>Objekte und Methoden als grundlegende Bestandteile einer objektorientierten Programmiersprache</p>	<p>... lernen die Begriffe „Objekt“ und „Methode“ kennen.</p> <p>... verwenden Python als objektorientierte Programmiersprache.</p> <p>... binden mehrere, gleichartige Objekte in ein Programm ein.</p> <p>... verwenden Strukturen/Sequenzen wie Tupel und Listen.</p>	<p>6.2 Algorithmen erkennen</p> <ul style="list-style-type: none"> - Algorithmische Muster und Strukturen in verschiedenen Kontexten erkennen und nachvollziehen <p>6.3 Modellieren und Programmieren</p> <ul style="list-style-type: none"> - Lösungsstrategien entwickeln und algorithmische Sequenz planen - Programmieren <p>Programm „Tanzende Turtles“</p>



Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Vertiefung der Verarbeitung von Zeichenketten und Informationen	<p>... lernen den Datentyp „String“ als neue Ausprägung einer Sequenz kennen.</p> <p>... vergleichen und unterscheiden die verschiedenen Typen von Sequenzen.</p> <p>... lernen neue Befehle zum Umgang mit Zeichenketten kennen.</p> <p>... nutzen die neuen Befehle, um komplexere Programme zum Verarbeiten von Zeichenketten zu schreiben.</p> <p>... verwenden den Datentyp „Dictionary“ um zusammenhängende Informationen zu modellieren.</p> <p>... beschäftigen sich mit Methoden zur Kodierung von Daten, z.B. der Cäsar-Kodierung.</p>	<p>6.2 Algorithmen erkennen</p> <ul style="list-style-type: none"> - Algorithmische Muster und Strukturen in verschiedenen Kontexten erkennen und nachvollziehen <p>6.3 Modellieren und Programmieren</p> <ul style="list-style-type: none"> - Lösungsstrategien entwickeln und algorithmische Sequenz planen - Programmieren <p>Programm „Star-Wars-Name“, Überarbeitung des Quiz-Spiels aus Klasse 8, Simulation eines „Online-Shops“, Programm zur Cäsar-Kodierung</p>



Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
<p>Weitere Vertiefung der Programmierkenntnisse in Python</p>	<p>... lernen weitere Möglichkeiten der (objektorientierten) Programmierung kennen, wie z.B. das Definieren eigener Klassen, das Verwenden weiterer Python-Module oder Einbinden von Bildern in Python-Programme.</p>	<p>6.2 Algorithmen erkennen</p> <ul style="list-style-type: none"> - Algorithmische Muster und Strukturen in verschiedenen Kontexten erkennen und nachvollziehen <p>6.3 Modellieren und Programmieren</p> <ul style="list-style-type: none"> - Lösungsstrategien entwickeln und algorithmische Sequenz planen - Programmieren <p>z.B. Simulation einer Uhr</p> <div style="display: flex; justify-content: space-around; align-items: center;">   </div>

2.3.6.2 Thema: „Einführung in die Aussagenlogik“

Leitfragen: Was ist eine Aussage im Sinne der Aussagenlogik? Wie lassen sich verschiedene Aussagen miteinander verknüpfen? Wo findet die Aussagenlogik eine Anwendung?




Vorhabensbezogene Konkretisierung: Die Aussagenlogik ist ein Bestandteil der theoretischen Informatik. Innerhalb dieses Moduls sollen die Schülerinnen und Schüler sich damit auseinandersetzen, wie man Umgangssprache und informatische Konzepte miteinander verbinden kann. Dazu lernen sie zunächst grundlegende Begriffe und Operatoren der Aussagenlogik kennen, untersuchen aussagenlogische Verknüpfungen und übertragen ihr Wissen auf Anwendungsbeispiele.

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Grundlagen der Aussagenlogik	... lernen grundlegende Definitionen und Operatoren der Aussagenlogik kennen. ... untersuchen umgangssprachliche Formulierungen auf logische Eigenschaften.	Operatoren: nicht, und, oder, entweder...oder, wenn...dann, genau dann...wenn
Untersuchen von aussagenlogischen Ausdrücken mithilfe von Wahrheitstabelle	... verknüpfen einzelne aussagenlogische Ausdrücke mit unterschiedlichen Operatoren. ... untersuchen zusammengesetzte Ausdrücke mithilfe von Wahrheitstabelle. ... leiten verschiedene Darstellungsformen aussagenlogischer Ausdrücke (z.B. KNF oder DNF) her.	
Anwendungen der Aussagenlogik	... übertragen ihr Wissen über aussagenlogische Ausdrücke auf Anwendungsbeispiele. ... lösen Alltagsprobleme mithilfe der Aussagenlogik.	

2.3.6.3 Optionales Thema: „Erstellen von Webseiten mit HTML und CSS“

Leitfragen: Wie ist das Internet aufgebaut? Wie sind Webseiten strukturiert? Welche Gestaltungselemente gibt es?

Vorhabensbezogene Konkretisierung: Das Internet ist in den letzten Jahrzehnten ein unverzichtbares alltägliches Medium geworden. Die grundlegende strukturgebende Sprache ist weiterhin HTML und die dazugehörige Erweiterung zur optischen Ausgestaltung CSS. Webseiten müssen heutzutage so gestaltet werden, dass sie auf allen möglichen Endgeräten, vom klassischen PC bis hin zum Handybildschirm, lesbar sind. Neben den Strukturelementen soll auch der gestalterische Aspekt eine Rolle spielen.

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Aufbau des Internets	<p>...lernen den dezentralen Aufbau des Internets kennen.</p> <p>...erarbeiten den Aufbau einer Server-Client-Struktur.</p> <p>...können Plattformunabhängigkeit als zentrales Merkmal von Webseiten definieren.</p>	<p>6.1 Prinzipien der digitalen Welt</p> <p>- Grundlegende Prinzipien und Funktionsweisen der digitalen Welt identifizieren, kennen, verstehen und bewusst nutzen</p> 
Strukturelemente von HTML	<p>...erarbeiten den Aufbau von HTML-Tags.</p> <p>...erarbeiten die Unterteilung einer Webseite in head und body.</p> <p>...erarbeiten exemplarische Meta-Angaben.</p> <p>...erarbeiten grundlegende Elemente zur Text-Strukturierung wie z.B. p, div und h.</p> <p>...erarbeiten den Link-Tag a und planen den Aufbau einer Webseite in verschiedene Unterseiten.</p> <p>...erarbeiten Tabellen und Listen.</p>	<p>6.3 Modellieren und Programmieren</p> <p>- Lösungsstrategien entwickeln und algorithmische Sequenz planen</p> <p>- Programmieren</p> <p>Schrittweise Entwicklung einer eigenen Webseite zu einem selbstgewählten Thema.</p> 
Grundlegende Elemente von CSS	<p>...erlernen den Aufbau von Stylesheets.</p> <p>...erlernen die Integration von Stylesheets in HTML-Dokumente.</p> <p>...erlernen CSS-Elemente zur Bildschirmteilung.</p> <p>...erlernen gestalterische Elemente zur Farbgebung / Schriftgestaltung / etc.</p>	<p>6.3 Modellieren und Programmieren</p> <p>- Lösungsstrategien entwickeln und algorithmische Sequenz planen</p> <p>- Programmieren</p> <p>Schrittweise Entwicklung einer eigenen Webseite zu einem selbstgewählten Thema.</p> 

2.3.6.4 Optionales Thema: „Einführung in formale Sprachen und Grammatiken“

Leitfragen: Was ist eine formale Sprache/Grammatik? Wozu werden formale Sprachen und Grammatiken verwendet?

Vorhabensbezogene Konkretisierung: Auch das Gebiet der formalen Sprachen und Grammatiken gehört thematisch zum Bereich der theoretischen Informatik. In diesem Modul sollen die Schülerinnen und Schüler sowohl den Zusammenhang zwischen formalen Sprachen und der Entwicklung von Programmiersprachen erfahren als auch Grundlagen für das weiterführende Thema der „Automatentheorie“ erhalten.

Unterrichtssequenzen	Kompetenzen Die Schülerinnen und Schüler...	Beispiele / Hinweise
Einführung und Grundlagen in das Thema formale Sprachen und Grammatiken	<p>... lernen die grundlegenden Definitionen im Bereich „formale Sprachen und Grammatiken“ kennen.</p> <p>... beschreiben die Bestandteile von formalen Grammatiken unter Verwendung von Fachbegriffen.</p>	
Aufstellen und Untersuchen formaler Sprachen und Grammatiken	<p>... stellen die Syntax von formalen Sprachen mithilfe von Ableitungsbäumen und Syntaxdiagrammen dar.</p> <p>... leiten aus einer formalen Grammatik die erzeugte Sprache ab (und umgekehrt).</p> <p>... ordnen formale Grammatiken den verschiedenen Typen zu.</p>	Ausblick: Automatentheorie

3 Grundsätze der fachmethodischen und fachdidaktischen Arbeit im Informatikunterricht der gymnasialen Unter- und Mittelstufe

3.1 Vereinbarungen zur fachmethodischen und fachdidaktischen Arbeit im Informatikunterricht

Unter Berücksichtigung des Schulprogramms hat die Fachkonferenz Informatik die folgenden fachmethodischen und fachdidaktischen Grundsätze beschlossen:

1. Geeignete Problemstellungen zeichnen die Ziele des Unterrichts vor und bestimmen die Struktur der Lernprozesse.
2. Inhalt und Anforderungsniveau des Unterrichts entsprechen dem Leistungsvermögen der Schüler/innen.
3. Die Unterrichtsgestaltung ist auf die Ziele und Inhalte abgestimmt.
4. Medien und Arbeitsmittel sind schülernah gewählt.
5. Die Schüler/innen erreichen einen Lernzuwachs.
6. Der Unterricht fördert eine aktive Teilnahme der Schüler/innen.
7. Der Unterricht fördert die Zusammenarbeit zwischen den Schülern/innen und bietet ihnen Möglichkeiten zu eigenen Lösungen.
8. Der Unterricht berücksichtigt die individuellen Lernwege der einzelnen Schüler/innen.
9. Die Schüler/innen erhalten Gelegenheit zu selbstständiger Arbeit und werden dabei unterstützt.
10. Der Unterricht fördert strukturierte und funktionale Partner- bzw. Gruppenarbeit.
11. Der Unterricht fördert strukturierte und funktionale Arbeit im Plenum.
12. Die Lernumgebung ist vorbereitet; der Ordnungsrahmen wird eingehalten.
13. Die Lehr- und Lernzeit wird intensiv für Unterrichtszwecke genutzt.
14. Es herrscht ein positives pädagogisches Klima im Unterricht. Der Unterricht unterliegt der Wissenschaftsorientierung und ist dementsprechend eng verzahnt mit seiner Bezugswissenschaft.
15. Der Unterricht ist problemorientiert und soll von realen Problemen ausgehen und sich auf solche rückbeziehen.
16. Der Unterricht folgt dem Prinzip der Exemplarität und soll ermöglichen, informatische Strukturen und Gesetzmäßigkeiten in den ausgewählten Problemen und Projekten zu erkennen.
17. Der Unterricht ist anschaulich sowie gegenwarts- und zukunftsorientiert und gewinnt dadurch für die Schülerinnen und Schüler an Bedeutsamkeit.
18. Der Unterricht ist handlungsorientiert, d.h. projekt- und produktorientiert angelegt.
19. Im Unterricht werden sowohl für die Schule didaktisch reduzierte als auch reale Informatiksysteme aus der Wissenschafts-, Berufs- und Lebenswelt eingesetzt.

3.2 Individualisierungsprozesse im Rahmen des Informatikunterrichts

Für die Fachkonferenz Informatik beschreibt der Begriff „Individualisierung“ den „Prozess, der die LernerInnen von der Fremdbestimmung zur Selbstbestimmung ihres Lernprozesses führt, indem Differenzierungsmaßnahmen umgesetzt werden, um die Eigenaktivität, die Motivation und die Eigenverantwortung zu steigern“. Zur Umsetzung im Informatikunterricht in den Klassen 5 und 6 sollen dazu folgende Maßnahmen erweitert bzw. vertieft werden:

3.2.1 Individuelle Aufgabenstellungen im Rahmen des Unterrichts

Da die Schülerinnen und Schüler im Informatikunterricht „von Natur aus“ (d.h. aufgrund der Gegebenheiten in den Computerräumen und den deutlich unterschiedlichen Fähigkeiten und Fertigkeiten im Umgang mit Hardware/Software) mit recht individuellem Tempo in Einzel- oder Partnerarbeit die gestellten Aufgaben bearbeiten, sind wir als Fachlehrer daran gewöhnt, Aufgaben mit verschiedenen (ansteigenden) Schwierigkeitsgraden vorzubereiten und im Rahmen des Unterrichts individuell einzusetzen. Dabei ist es wichtig transparent zu machen, welche aus dem Unterricht erwachsenen Aufgaben von allen oder nur einem Teil der Lerngruppe zu bearbeiten sind.

3.2.2 Erweiterte Projektarbeit

Im Rahmen des Fachs Informatik bieten sich vielfältige Möglichkeiten an, projektorientiert zu arbeiten: Ein Individualisierungsprozess kann hier durch exakt zugeschnittene Absprachen zwischen Lehrperson und Schülergruppe hinsichtlich Zielrichtung und Komplexität eines Programmierprojektes, theoretischer Aspekte oder einer historischen Betrachtung umgesetzt werden.

4 Grundsätze der Leistungsbewertung und Leistungsrückmeldung

Auf der Grundlage des Kernlehrplans Informatik hat die Fachkonferenz des Nikolaus-Ehlen-Gymnasiums im Einklang mit dem entsprechenden schulbezogenen Konzept die nachfolgenden Grundsätze zur Leistungsbewertung und Leistungsrückmeldung beschlossen. Die nachfolgenden Absprachen stellen die Minimalanforderungen an das lerngruppenübergreifende gemeinsame Handeln der Fachgruppenmitglieder dar und beziehen sich auf den Unterricht in den Klassen 5 und 6 sowie den Wahlpflichtfach-Bereich. Bezogen auf die einzelne Lerngruppe kommen ergänzend weitere der in den Folgeabschnitten genannten Instrumente der Leistungsüberprüfung zum Einsatz.

4.1 Beurteilungsbereiche und Überprüfungsformen

Da im Informatikunterricht in den Klassen 5 und 6 keine Klassenarbeiten geschrieben werden, resultiert die Zeugnisnote ausschließlich aus dem Bereich der „sonstigen Mitarbeit“.

Für den Wahlpflichtfach-Bereich gilt: Die Note am Ende eines Halbjahres setzt sich zu jeweils ca. 50 % aus den Leistungen der „sonstigen Mitarbeit“ und den Ergebnissen der Kursarbeiten zusammen. Die folgenden beiden Abschnitte stellen zu beiden Beurteilungsbereichen eine detaillierte Übersicht einzelner Leistungsbewertungsaspekte dar.

4.1.1 Beurteilungsbereich Sonstige Mitarbeit

Den Schülerinnen und Schülern werden die Kriterien zum Beurteilungsbereich „sonstige Mitarbeit“ zu Beginn des Schuljahres genannt. Dabei setzt sich die Gesamtleistung in diesem Beurteilungsbereich aus der „mündlichen Mitarbeit“ (Beteiligung am Unterrichtsgespräch, Präsentation von Arbeitsergebnissen, Referate, Mitarbeit in Partner- und Gruppenarbeitsphasen), den „praktischen Leistungen am Computer“ (z.B. Implementierung, Test und Anwendung von Programmen, die regelmäßig als Datei abgegeben werden), „Ergebnisse der Dalton-Aufgaben“ (z.B. Besprechen schriftlicher Dalton-Aufgaben oder erstellter Programme) und „sonstige schriftliche Leistungen“ (z.B. Lernerfolgsüberprüfung durch kurze schriftliche Übungen, Bearbeitung von schriftlichen Aufgaben im Unterricht) zusammen.

Als allgemeine Kriterien sowohl für die mündlichen als auch für die schriftlichen Formen der sonstigen Mitarbeit werden herangezogen:

- die Qualität der Beiträge,
- die Quantität der Beiträge und
- die Kontinuität der Beiträge.

Besonderes Augenmerk wird dabei gelegt auf:

- die sachliche Richtigkeit,
- die angemessene Verwendung der Fachsprache,
- die Darstellungskompetenz,
- die Komplexität und den Grad der Abstraktion,
- die Selbstständigkeit im Arbeitsprozess,
- die Präzision und
- die Differenziertheit der Reflexion.

Darüber hinaus gelten für Gruppenarbeiten die Kriterien:

- Einbringen in die Arbeit der Gruppe,
- Durchführung fachlicher Arbeitsanteile,
- Qualität des entwickelten Produktes,

bei der Bearbeitung von Dalton-Aufgaben:

- Vollständigkeit der angefertigten Lösungen,
- Bereitschaft zur Vorstellung und tatsächliche Präsentation der vorbereiteten Dalton-Aufgabe.

und bei Projektarbeiten:

- Dokumentation des Arbeitsprozesses,
- Grad der Selbstständigkeit,
- Reflexion des eigenen Handelns sowie
- Aufnahme von Beratung durch die Lehrkraft.

4.1.2 Beurteilungsbereich Kursarbeiten

Für den Wahlpflicht-Bereich gilt: In jedem Halbjahr erfolgen zwei Leistungsüberprüfungen in Form von schriftlichen Arbeiten mit einer Dauer von bis zu 90 Minuten (zwei Schulstunden). Dabei kann eine Kursarbeit pro Schuljahr durch eine mehrwöchige, benotete Projektarbeit ersetzt werden.

Die Bewertung der schriftlichen Leistungen in Kursarbeiten erfolgt über ein Raster mit Punkten, die im Erwartungshorizont den einzelnen Aufgaben und Unteraufgaben zugeordnet sind. Die Note „ausreichend“ soll bei Erreichen von knapp der Hälfte der maximalen Punktzahl erteilt werden.

4.2 Grundsätze der Leistungsrückmeldung und Beratung

Leistungsrückmeldungen können erfolgen

- nach einer mündlichen Überprüfung,
- bei Rückgabe von schriftlichen Leistungsüberprüfungen,
- nach Abschluss eines Projektes,
- nach einem Vortrag oder einer Präsentation,
- bei auffälligen Leistungsveränderungen,
- auf Anfrage und
- zu Elternsprechtagen.

Die Leistungsrückmeldung kann

- durch ein Gespräch mit der Schülerin oder dem Schüler,
- durch einen Feedbackbogen,
- durch die schriftliche Begründung einer Note oder
- durch eine individuelle Lern-/Förderempfehlung

erfolgen.

5 Lehr- und Lernmittel

Die Fachkonferenz Informatik hat sich darauf verständigt, derzeit ohne ein eingeführtes Lehrwerk im Unterricht zu arbeiten. Die Schülerinnen und Schüler, die das Fach Informatik in der Sekundarstufe I im Wahlpflicht II-Bereich belegen, sollten jedoch nach Möglichkeit zu Hause Zugang zu einem Rechner haben. Die im Unterricht verwendete Software (derzeit die Programmiersprachen bzw. Entwicklungsumgebungen „Scratch“, „Python“ sowie ggf. „HTML“, „CSS“ und „Javascript“) kann über Downloads von den offiziellen Webseiten scratch.mit.edu/ und www.python.org/ installiert werden.

Für den Informatikunterricht in den Klassen 5 und 6 stellt die Schule neben den verwendeten Softwareprodukten die benötigte Hardware in Form von PCs und Calliope-Mini-Platinen zur Verfügung. Diese verbleiben jedoch in der Schule und dürfen nicht mit nach Hause genommen werden.

6 Entscheidungen zu fach- und unterrichtsübergreifenden Fragen

Die unterrichtlichen Vorhaben zum Thema „Arbeiten mit Anwendungen aus einem Office-Paket“, mit den Aspekten „Einführung in ein Textverarbeitungsprogramm“, „Einführung in ein Tabellenkalkulationsprogramm“ sowie „Einführung in eine Präsentationssoftware“ (in Klasse 5 und 6) sind u.a. als Vorbereitung auf die Verwendung in anderen Fächern wie Deutsch, Geschichte, Politik und Mathematik anzusehen.

In unregelmäßigen Abständen können Exkursionen zum „Heinz-Nixdorf-Museum“ in Paderborn unternommen werden.

7 Qualitätssicherung und Evaluation

Das schulinterne Curriculum des Fachs Informatik stellt keine starre Größe dar, sondern ist als „lebendes Dokument“ zu betrachten. Dementsprechend sind die Inhalte stetig zu überprüfen, um ggf. Modifikationen vornehmen zu können. Die Fachkonferenz (als professionelle Lerngemeinschaft) trägt durch diesen Prozess zur Qualitätsentwicklung und damit zur Qualitätssicherung des Faches bei. Dieser Prozess wird durch einen kontinuierlichen Erfahrungsaustausch zwischen den Fachlehrkräften sowie regelmäßige Diskussionen zwischen allen Beteiligten im Rahmen der Fachkonferenzsitzungen gewährleistet. Darüber hinaus soll die Qualitätssicherung durch Beratungen zu Aufgabenstellungen von Klausuren und eine Erörterung der Ergebnisse von Leistungsüberprüfungen ausgebaut werden.