

Schulinterner Lehrplan
des Freiherr-vom-Stein-Gymnasiums
im Fach Informatik
für die Einführungsphase

Unterrichtsvorhaben 1: Motivation (Zeitbedarf ca. 6 Stunden)

Leitfrage: „Was ist und kann Informatik?“

Inhaltsfeld: Mensch, Gesellschaft, Informatik

Leitkompetenzen: Argumentieren, Kommunizieren/Kooperieren

Anzustrebende Kompetenzen aus dem Kernlehrplan:

Die SchülerInnen erläutern wesentliche Grundlagen der Geschichte der digitalen Datenverarbeitung und benennen gegenwärtige sowie potentiell zukünftige Chancen von Digitalisierung und deren Bedeutung für die (Arbeits-)Welt. Dafür nutzen sie das Internet zur Recherche, zum Datenaustausch und zur Kommunikation.

mögliche Konkretisierung des Unterrichtsvorhabens:

Der Begriff „Informatik“ wird etymologisch erläutert.

Zu dem Leitthema „Informatik in Vergangenheit, Gegenwart und Zukunft“ wird in Partnerarbeit eine Internetrecherche durchgeführt, eine PowerPoint-Präsentation erstellt sowie ein Kurzreferat vorbereitet und gehalten.

Unterrichtsvorhaben 2: Einführung in Informatiksysteme (Zeitbedarf ca. 6 Stunden)

Leitfrage: „Wie funktionieren Computer(systeme)?“

Inhaltsfeld: „Informatiksysteme“

Leitkompetenzen: Kommunizieren/Kooperieren, Darstellen und Interpretieren

Anzustrebende Kompetenzen aus dem Kernlehrplan:

Die SchülerInnen stellen ganze Zahlen und Zeichen in Binärcodes dar und interpretieren Binärcodes als Zahlen und Zeichen. Sie beschreiben und erläutern den strukturellen Aufbau und die Arbeitsweise singulärer Rechner am Beispiel der „Von-Neumann-Architektur“. Sie nutzen das verfügbare Informatiksystem zur strukturierten Verwaltung und gemeinsamen Verwendung von Daten unter Berücksichtigung der Rechteverwaltung (was auch für die folgenden Unterrichtsvorhaben gilt).

mögliche Konkretisierung des Unterrichtsvorhabens:

Die SchülerInnen beschäftigen sich mit Bits und Bytes und analysieren das Speichervolumen z.B. ihres eigenen Smartphones.

Sie lernen die Hardwarekomponenten eines Rechners kennen und setzen sich mit der Von-Neumann-Architektur auseinander.

Sie akzeptieren und unterschreiben die Benutzerordnung für die Computerräume, werden in das schuleigene Computersystem und dessen Ordnerablage eingeführt und trainieren eine sinnvolle Dateiablage.

Unterrichtsvorhaben 3: Einführung in die objektorientierte Programmierung (Zeitbedarf ca. 12 Stunden)

Leitfrage: „Wie kommt die Welt in den Rechner?“

Inhaltsfeld: Daten und ihre Strukturierung

Leitkompetenzen: Modellieren, Implementieren

Anzustrebende Kompetenzen aus dem Kernlehrplan:

Die SchülerInnen ermitteln bei der Analyse einfacher Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen. Sie modellieren Klassen mit ihren Attributen, ihren Methoden und Assoziationsbeziehungen sowie unter Verwendung von Vererbung. Sie ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen und Objekttypen zu sowie Klassen, Attributen und Methoden ihren Sichtbarkeitsbereich. Sie stellen die Kommunikation zwischen Objekten graphisch dar, stellen Klassen-, Assoziations- und Vererbungsbeziehungen in Diagrammen graphisch dar, dokumentieren Klassen durch Beschreibung der Funktionalität der Methoden, analysieren und erläutern eine objektorientierte Modellierung und implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken. Sie interpretieren Fehlermeldungen des Compilers und korrigieren den Quellcode.

mögliche Konkretisierung des Unterrichtsvorhabens:

Die SchülerInnen beschreiben zwei verschiedene Stühle hinsichtlich ihrer Unterschiede und Gemeinsamkeiten und lernen so den Unterschied zwischen einem Objekt und einer Klasse kennen. Sie erstellen UML-Diagramme der Klasse „Stuhl“ sowie anderer Klassen und lernen dabei die ersten einfachen Datentypen in der Programmiersprache Java kennen. Sie implementieren die ersten Klassen in der Programmiersprache Java in der Entwicklungsumgebung „BlueJ“. Um den verschiedenen Objekten einer Klasse verschiedene Eigenschaften zuzuweisen bzw. vorhandene Eigenschaften abzufragen, nutzen sie erstmalig die setter- und getter-Methoden und unterscheiden so zwischen sondierenden und manipulierenden Methoden. Sie implementieren die Klasse „Haus“ unter Nutzung der bereits dokumentierten, von „Shapes“ abgeleiteten Klassen „Triangle“, „Circle“ and „Square“. Schließlich kennen sie die wesentlichen Bestandteile des Quellcodes einer Klasse und erläutern diesen.

Unterrichtsvorhaben 4: Kontrollstrukturen (Zeitbedarf ca. 15 Stunden)

Leitfrage: „Wie programmiert man ein erstes Spiel?“

Inhaltsfeld: Formale Sprachen und Automaten, Algorithmen

Leitkompetenzen: Argumentieren, Modellieren, Implementieren

Anzustrebende Kompetenzen aus dem Kernlehrplan:

Die SchülerInnen analysieren und erläutern Computerprogramme. Sie implementieren einfache Algorithmen unter Beachtung der Syntax und Semantik einer Programmiersprache, modifizieren einfache Algorithmen und implementieren Algorithmen unter Verwendung von Kontrollstrukturen sowie Methodenaufrufen. Sie interpretieren Fehlermeldungen und korrigieren den Quellcode. Sie implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken.

mögliche Konkretisierung des Unterrichtsvorhabens:

Die SchülerInnen lernen die Programmierumgebung „Greenfoot“ kennen. Sie machen sich durch Testen und Quellcodeanalyse mit einem konkreten Szenario, etwa „Leaves and Wombats, vertraut: Sie prüfen mit einer if-Abfrage, ob ein Wombat ein Blatt findet, lassen ihn bis zu einer Wand laufen (while-Schleife) oder ihn z.B. acht Schritte laufen (for-Schleife). Mit den kennengelernten Kontrollstrukturen und den Klassenbibliotheken von Greenfoot sowie der Java-Klasse Random implementieren sie ein „Schatzräuber-Spiel“.

Unterrichtsvorhaben 5: Algorithmen (Zeitbedarf ca. 6 Stunden)

Leitfrage: „Wie kommt man bei einem Problem sicher ans Ziel?“

Inhaltsfeld: Algorithmen

Leitkompetenzen: Argumentieren, Darstellen und Interpretieren

Anzustrebende Kompetenzen aus dem Kernlehrplan:

Die SchülerInnen entwerfen, analysieren, erläutern und modifizieren einfache Algorithmen und Programme. Sie stellen sie umgangssprachlich und graphisch dar und implementieren sie unter Verwendung von Variablen und Wertzuweisungen, Kontrollstrukturen sowie Methodenaufrufen. Sie testen ihre Programme schrittweise anhand von Beispielen.

mögliche Konkretisierung des Unterrichtsvorhabens:

Die SchülerInnen versuchen, mit einem gewissen Rotationsprinzip Zahlen in eine richtige Reihenfolge zu bringen oder die systematischen Rotationen eines Schiebepuzzles zu erkennen. Sie lernen den Algorithmus kennen, mit dem das sicher funktioniert. Sie finden weitere Beispiele algorithmischer Problemlösungen (Euklidischer Algorithmus, Algorithmus zum Umwandeln von Dezimal- in Binärzahlen etc.), implementieren und testen diese.

Unterrichtsvorhaben 6: Arrays (Zeitbedarf ca. 9 Stunden)

Leitfrage: „Wie kann man mehrere Daten desselben Typs effizient speichern und abrufen?“

Inhaltsfeld: „Daten und ihre Strukturierung“

Leitkompetenzen: Argumentieren, Modellieren, Implementieren

Anzustrebende Kompetenzen aus dem Kernlehrplan:

Die SchülerInnen ordnen Attributen, Parametern und Rückgaben von Methoden (statische) lineare Datensammlungen zu.

mögliche Konkretisierung des Unterrichtsvorhabens:

Die SchülerInnen erkennen, dass das Abspeichern z.B. aller Namen der KursteilnehmerInnen viel Quelltext und viel Schreiarbeit erfordert. Sie lernen Arrays als lineare Datensammlungen kennen. Sie beschreiben und erläutern wesentliche Eigenschaften der Datenstruktur Array (vorgegebene Länge, vorgegebener Daten- bzw. Objekttyp, feste Positionszuweisung etc.), deklarieren und initialisieren Arrays und implementieren elementare Operationen (löschen, tauschen, suchen). Dies erfolgt zunächst mit einem primitiven Datentyp (z.B. Zahlenliste oder Würfelbecher mit Datentyp `int`), anschließend mit einem komplexen Daten- oder Objekttyp (z.B. Bundesliga-tabelle mit Datentyp `String` oder Objekttyp „Verein“).

Unterrichtsvorhaben 7: Suchen und Sortieren (Zeitbedarf ca. 9 Stunden)

Leitfrage: „Wie kann man Daten sortieren, z.B. um ausgewählte Daten schnell zu finden?“

Inhaltsfeld: Algorithmen

Leitkompetenzen: Darstellen und Interpretieren, Implementieren

Anzustrebende Kompetenzen aus dem Kernlehrplan:

Die SchülerInnen analysieren Such- und Sortieralgorithmen und wenden sie auf Beispiele an. Sie entwerfen einen weiteren Algorithmus zum Sortieren und beurteilen die Effizienz von Algorithmen am Beispiel von Sortierverfahren hinsichtlich Zeitaufwand und Speicherplatzbedarf.

mögliche Konkretisierung des Unterrichtsvorhabens:

Die SchülerInnen sortieren beispielsweise Spielkarten. Dabei überlegen sie, inwiefern sich der Sortiervorgang eines Computers von dem eines Menschen unterscheidet. Sie entwickeln einen Sortier-Algorithmus. Erfahrungsgemäß werden die Algorithmen Bubble-Sort und Min-Sort intuitiv entdeckt. Diese werden intensiv besprochen und implementiert. Andere, insbesondere rekursive, Sortierverfahren werden nur theoretisch erläutert und hinsichtlich Zeitaufwand und Speicherplatzbedarf analysiert.

Unterrichtsvorhaben 8: GUI (Zeitbedarf ca. 9 Stunden)
Leitfrage: „Wie kann man selbst eine schöne graphische Benutzeroberfläche herstellen?“
Inhaltsfeld: Formale Sprachen und Automaten
Leitkompetenzen: Modellieren, Implementieren
Anzustrebende Kompetenzen aus dem Kernlehrplan: Die SchülerInnen implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken.
mögliche Konkretisierung des Unterrichtsvorhabens: Die SchülerInnen lernen Textfelder, Eingabefelder und Knöpfe als wesentliche Bestandteile einer GUI (graphical user interface) kennen. Sie implementieren eine kreativ nach ihren eigenen Vorstellungen funktionierende GUI, z.B. eine, die bei Eingabe eines Vor- und Nachnamens automatisch eine E-Mail-Adresse generiert, oder aber eine, die bei Eingabe eines Filmtitels Informationen zum Film aufzeigt etc.

Unterrichtsvorhaben 9: Vertiefung (Zeitbedarf ca. 9 Stunden)
Leitfrage: „Wie lässt sich das Spiel ‚Hangman‘ programmieren?“
Inhaltsfeld: Daten und ihre Strukturierung, Formale Sprachen und Automaten
Leitkompetenzen: Modellieren, Implementieren
Anzustrebende Kompetenzen aus dem Kernlehrplan: Die SchülerInnen implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken. Sie ermitteln, ausgehend von dem analogen Spiel, Klassen und Objekte und modellieren Klassen mit ihren Attributen und Methoden. Sie entwerfen einfache Algorithmen und stellen sie umgangssprachlich und graphisch dar.
mögliche Konkretisierung des Unterrichtsvorhabens: Die SchülerInnen lernen die Java-Klasse <code>String</code> kennen und beschreiben und erläutern deren grundlegende Methoden. Sie identifizieren das Suchwort als Array von Zeichen (<code>chars</code>) und vertiefen die in den Unterrichtsvorhaben 6 und 7 erworbenen Kompetenzen. Sie nutzen ihre im vorherigen Unterrichtsvorhaben erworbenen Kenntnisse über graphical user interfaces, um ein zunächst ohne Graphik funktionierendes, später graphisch ansprechendes Galgenmännchen-Spiel zu programmieren (Trennung von Fachklassen und Graphik).

Unterrichtsvorhaben 10: Ausblick (Zeitbedarf ca. 6 Stunden)
Leitfrage: „Was darf Informatik und was nicht?“
Inhaltsfeld: Mensch, Gesellschaft, Informatik
Leitkompetenzen: Argumentieren, Kommunizieren/Kooperieren
Anzustrebende Kompetenzen aus dem Kernlehrplan: Die SchülerInnen bewerten anhand von Fallbeispielen die Auswirkungen des Einsatzes von Informatiksystemen.
mögliche Konkretisierung des Unterrichtsvorhabens: Nach der Chancenbetrachtung von Digitalisierung zu Beginn des ersten Unterrichtsjahres Informatik soll nun auch eine „Risiken“-Betrachtung erfolgen. Wo liegen ethische Probleme in den Möglichkeiten der Informatik? Die Fallbeispiele können dem aktuellen Tagesgeschehen und/oder der Fachliteratur entnommen werden.

Die Leitkompetenz „Kommunizieren und Kooperieren“ wird während aller Unterrichtsvorhaben quasi beiläufig geschult und ausgebaut, da die Arbeitsergebnisse und Projekte in aller Regel in kooperativen Arbeitsformen entstehen und am Ende eines Vorhabens im Plenum präsentiert werden.

Das Leistungsbewertungskonzept ist in einem gesonderten Dokument auf der Homepage verfügbar.