

11.2.2005

Ein Entwurf für

Standards für den Informatikunterricht in der Sekundarstufe I des Gymnasiums

Der vorgelegte Vorschlag beruht auf den Erfahrungen mit dem „Freiwilligen Schulzweig Informationstechnik“ für die Klassen 7-10 am Max-Planck-Gymnasium in Göttingen, der seit 1999 besteht, dem Entwurf für ein niedersächsisches Curriculum „Informatik in der Sekundarstufe I - mit Informatiksystemen gestalten“ von 2003, der Sichtung der Rahmenlehrpläne Informatik mehrerer Bundesländer, den „Bildungsstandards im Fach Mathematik für den Mittleren Schulabschluss“ und den Diskussionen auf dem Dagstuhl-Seminar „Concepts of Empirical Research and Standardisation of Measurement in the Area of Didactics of Informatics“ im September 2004.

Vorbemerkungen:

- Da Niedersachsen Informatik in der Sek. I in verschiedenen Schulformen erproben will, wären Bildungsstandards hilfreich. Obwohl nicht absehbar ist, ob und wann diese existieren werden, ist es sinnvoll, die Erprobung schon an ersten Entwürfen solcher Standards auszurichten.
- Standards müssen überprüfbar sein. Da Unterricht nicht nur zu überprüfbaren Ergebnissen führt, sondern weitere, wichtige Aufgaben hat, dürfen die Standards nicht so umfangreich sein, dass sie den Unterricht dominieren und Unterrichtsformen sowie –organisation bestimmen, ggf. zeitaufwändige ausschließen. Damit müssen parallel zu den Standards verschiedene mögliche Unterrichtsgänge skizziert werden, die eine angemessene **Zeitplanung** ermöglichen.
- Der Schulinformatik ist es, bedingt durch die Mächtigkeit der benutzten Informatiksysteme, eigen, **Formen des selbstständigen und problemorientierten Lernens** fördern zu können. Folgerichtig sollten informatische Standards so gefasst werden, dass sie sich in einem problem- und ergebnisorientierten Unterricht aus den gewählten Problemstellungen in unterschiedlicher Weise ergeben können.
- Die Standards selbst dürfen nicht nur aus fachimmanenten Überlegungen folgen. An allgemein bildenden Schulen sind die spezifischen **Beiträge des Faches zur Allgemeinbildung** im Kontext der anderen Fächer zu berücksichtigen.

Zur Allgemeinbildung:

Ohne hier eine eingehendere Diskussion über Allgemeinbildung führen zu wollen, werden doch Rahmenbedingungen benötigt. Dazu gehören die Aufgaben allgemein bildender Schulen

- **Orientierungswissen** zu vermitteln, hier: Informationen über die Grundlagen einer zunehmend durch Informationstechnologien geprägten und vernetzten Welt,
- **grundlegende Fertigkeiten** bereitzustellen, hier: den sachgerechten Umgang mit Informatiksystemen,

- **Erfahrungen** sowie **Selbsterprobung** in unterschiedlichen Bereichen zu ermöglichen, die unverzichtbar z. B. für die Auswahl späterer Berufsfelder sind, hier: z. B. den Zugang zu technischem Denken und Handeln am Beispiel des Einsatzes komplexer Informatiksysteme zu gewinnen
- und **Chancengleichheit** herzustellen, hier: z. B. aus dem sozialen Umfeld herrührende Defizite auszugleichen und Mädchen den Zugang zu den Informationstechnologien zu erleichtert, wie es in anderen Nationen, z. B. den angloamerikanischen bzw. romanischen Ländern offensichtlich der Fall ist.

Da wesentliche Teile dieser Aufgaben nicht oder nur schwer zu testbaren Ergebnissen führen, müssen die zur Erreichung der Standards erforderlichen Unterrichtsabschnitte entsprechenden Raum lassen. Eben dafür ist eine realistische Zeitplanung erforderlich (s.o.).

Die letzten beiden Punkte bedürfen für die Schulinformatik einer Erläuterung: In weiten Bereichen der didaktischen Literatur wird eine „Enttechnisierung“ der Informatik gefordert, auch um Hürden für die Mädchen abzubauen. (Als Beispiel mag der Verzicht auf technische Informatik in den aktuell gültigen EPAs gelten.) Ich halte diese Denkrichtung für falsch. Erstens wird durch den Verzicht auf Technik ein wesentliches Argument für das Schulfach Informatik aufgegeben – nämlich das Spektrum der Schulfächer um technische Aspekte zu erweitern –, zweitens ist die Distanz der Mädchen gegenüber der Informationstechnik offensichtlich kein geschlechtsspezifisches Phänomen, sondern an den Kulturraum gebunden – das wird z. B. im Vergleich zu den romanischen Ländern deutlich (s.o.), und drittens erscheint ein „enttechnisierter“ Orientierungsprozess, der u.a. in der Entscheidung für ein technisches Fach enden kann, mehr als problematisch, weil er irreführend ist. Da der Zugang zur Informationstechnik nicht über technische Geräte erfolgt, sondern über Informatikanwendungen im Bereich der Text- und Bildgestaltung, der Informationsgewinnung und -aufbereitung sowie der Kommunikation, also in Gebieten, an denen Mädchen der betrachteten Altersgruppe starkes Interesse haben und wo sie den gleichaltrigen Jungen klar überlegen sind, scheint mir die **Betonung der technischen Aspekte** der Informatik ein sinnvoller und Erfolg versprechender Weg, gerade auch den Mädchen einen Zugang zu (informations-)technischen Berufen zu eröffnen.

Zur Realisierbarkeit und zur Lehrerausbildung:

Die folgenden Standards erfordern nicht zwingend sofort ein eigenes Fach. Sie erfordern aber angemessene Unterrichtszeit, die wohl am effizientesten innerhalb eines besonderen Faches genutzt werden kann. Eine globale Verantwortlichkeit aller Fächer funktioniert nicht, wie sich z. B. am Scheitern der ITG gezeigt hat. Die Standards bis Klasse 7 lassen sich auch über Unterrichtsprojekte, ein Medienkonzept der Schule, epochalen Fachunterricht etc. erreichen. Die Standards bis Klasse 10 erfordern m. E. mindestens epochal Fachunterricht. Ob dieser vorläufig unter dem Etikett eines anderen Faches erfolgt, in – wenigen – anderen Fächern oder als Informatikunterricht, ist für die Standards gleichgültig. In jedem Fall muss für jede Schulform bzw. Schule festgelegt werden, was wann in welchem Fach zu erfolgen hat.

Für ein eigenes Fach Informatik spricht in erster Linie der Mangel an Fachlehrer/innen (s.u.). Für eine Schule wird es organisatorisch wesentlich einfacher sein, alle Schülerinnen und Schüler den Unterricht weniger Fachlehrer/innen durchlaufen zu lassen, als alle Unterrichtenden mehrerer anderer Fächer auf angemessenem Niveau fortzubilden.

Ein Hauptproblem der Schulinformatik sind die fehlenden Fachlehrer/innen. Die derzeit Unterrichtenden stammen überwiegend aus einer Altersstufe, die kurz vor der Pensionierung steht bzw. schon pensioniert worden ist. Sie werden derzeit ersetzt bzw. sind ersetzt worden von Kolleginnen und Kollegen, die zwar wesentlich bessere Anwenderkenntnisse besitzen als die Älteren, aber kaum über spezifische Informatikkenntnisse. Da die Nachfolger/innen der

jetzt ausscheidenden Generation sich weitgehend schon in der Ausbildung befinden, ist auch keine Besserung in Sicht. **Es ist deshalb davon auszugehen, dass sich der Mangel an Informatiklehrer/innen in den nächsten Jahren noch verschärft.**

Man kann den laufenden Umbruch allerdings auch als Chance begreifen. Nach den vorliegenden Erfahrungen wird die Verkürzung der (gymnasialen) Schulzeit zu einem **quantitativen Rückgang** der Wahlfächer in der Oberstufe, insbesondere der Informatik führen. Wenn es gelingt, neue grundständig ausgebildete Fachlehrer/innen zu gewinnen bzw. vorhandene jüngere Kolleginnen und Kollegen auf entsprechendem Niveau auszubilden, dann kann dieser Rückgang durch einen **qualitativen Fortschritt** kompensiert werden. Zusammen mit einer Fundierung des Faches in der Sek. I sollte sich damit ein auf breiter Basis qualitativ angemessener Informatikunterricht entwickeln lassen. Voraussetzung für die Gewinnung informatisch gebildeter Kolleginnen und Kollegen ist allerdings eine gesicherte berufliche Perspektive, also die Existenz eines eigenen Faches.

In den folgenden Überlegungen wird diese Situation berücksichtigt. Es wird davon ausgegangen, dass für längere Zeit die wenigen vorhandenen Informatiklehrer/innen weitgehend in der Oberstufe der Sek. II benötigt werden. Die Anforderungen in der Sek. I werden deshalb so konzipiert, dass aufbauend auf Anwendungen die informatischen Erweiterungen in den betrachteten Alterstufen auf angemessenem Niveau auch von Kolleginnen und Kollegen mit guten Anwenderkenntnissen unterrichtet werden können, die entsprechend fortgebildet worden sind.

Informatische Kompetenzen:

Im Folgenden werden informatische Kompetenzen in Anlehnung an die mathematischen Bildungsstandards formuliert und kurz erläutert.

(K 1) Informatisch argumentieren

Dazu gehört:

- Fragen stellen, die für die Informatik charakteristisch sind („Welches sind die wesentlichen Objekte?“, „Welche Beziehungen bestehen?“, „Welche beschreibenden Größen sind erforderlich?“, „Was kann weggelassen werden?“, „Welche Hilfsmittel stehen bereit?“, ...),
- Lösungsansätze und den Einsatz der dafür erforderlichen Informatiksysteme beschreiben, begründen und gegeneinander abwägen.

(K 2) Informatisch modellieren

Dazu gehört:

- vorgegebene und selbst formulierte Probleme bearbeiten,
- den Bereich oder die Situation, die modelliert werden soll, in informatische Begriffe übersetzen,
- Strukturen erkennen und die Problemstellung strukturiert zerlegen,
- geeignete Modellierungsmethoden und –werkzeuge auswählen und anwenden,
- geeignete algorithmische Verfahren auswählen, modifizieren und ggf. neu entwickeln,
- Ergebnisse in dem entsprechenden Bereich oder der entsprechenden Situation interpretieren und prüfen.

(K 3) Mit formalen und technischen Elementen der Informatik umgehen

Dazu gehört:

- mit unterschiedlichen Darstellungsformen und den entsprechenden Werkzeugen arbeiten,
- symbolische und formale Sprache in natürliche Sprache übersetzen und umgekehrt,
- unterschiedliche Darstellungsformen je nach Situation und Zweck auswählen und zwischen ihnen wechseln.

- Lösungs- und Kontrollverfahren vergleichen, ggf. geeignet modifizieren oder entwickeln und ausführen,
- informatische Werkzeuge sinnvoll und verständlich einsetzen.

(K 4) Kommunizieren

Dazu gehört:

- Arbeit sinnvoll aufteilen und Arbeitsfortschritte zwischen den Arbeitsgruppen kommunizieren und koordinieren,
- Überlegungen, Lösungswege bzw. Ergebnisse dokumentieren, verständlich darstellen und präsentieren,
- die Fachsprache adressatengerecht verwenden,
- Äußerungen von anderen und Texte zu informatischen Inhalten verstehen und überprüfen.

Zum Ordnungsschema der Kompetenzen:

Es ist sinnvoll, Kompetenzen unterschiedlichen Bereichen (oder „Bändern“) zuzuordnen. Diese Ordnung kann nach ganz unterschiedlichen Kategorien erfolgen. Eine Möglichkeit besteht darin, wie in der Mathematik Leitideen anzugeben. Für die Schulinformatik existiert ein vergleichbares Schema in den **fundamentalen Ideen**. Es liegt nahe, schon aus Gründen der Vergleichbarkeit zur Mathematik diese als Ordnungskriterium zu wählen. Ich habe aber aus verschiedenen Gründen davon Abstand genommen: Die fundamentalen Ideen der Informatik

- sind mitten in der Diskussion und in der derzeitigen Form noch nicht allgemein akzeptiert,
- beschreiben eher die Behandlung komplexer Problemstellungen und Lösungen als die in der Sek. I zu findenden einfachen Systeme
- und beziehen sich weitgehend auf die Oberstufeninformatik, decken den in der Sek. I stärker zu betonenden Anwendungsaspekt ungenügend ab.

Werden Kompetenzen fachlich geordnet, dann finden sich in diesen Ordnungen wenigstens teilweise die bekannten Fachgebiete der Oberstufen-Schulinformatik wieder. Wenn das so ist, dann sollte man m. E. dieses vertraute Schema ganz übernehmen, schon um die Fortschreibung für die Oberstufe zu erleichtern und den derzeit noch meist unzureichend ausgebildeten Unterrichtenden möglichst viel Bekanntes vorzusetzen.

Obwohl in Zeiten der objektorientierten Modellierung (OOM) die Trennung von Daten und Algorithmen etwas antiquiert erscheint, ist festzuhalten, dass die OOM zwar aktuell ist, aber nicht das Ende der Entwicklung darstellt. Sie ist auch nicht umfassend. In den veröffentlichten Beispielen für die Sek. I stellt die oft zu findende Klassifizierung von Objekten allein nach den beschreibenden Daten nur eine unzureichende Annäherung an OOM dar. Deshalb wird hier an der Trennung der Bereiche „Daten“ und „Algorithmen“ festgehalten werden. Unbeschadet davon ist die OOM in weiten Teilen eine wichtige Methode und sollte, wo sinnvoll, auch eingesetzt werden.

Als Konsequenz aus diesen Überlegungen erfolgt die Zuordnung zu den folgenden **Bändern**:

- **Informatiksysteme und Gesellschaft**

Dazu gehören Standards in der Nutzung von Anwendersystemen, Kenntnisse von deren Einsatz und dessen Auswirkungen.

- **Daten und Datenstrukturen**

Unterschiedliche Datenstrukturen sollen auf unterschiedlichem Niveau als Beschreibungs- und Ordnungsschema kennen gelernt werden.

- **Algorithmen**

Algorithmische Grundstrukturen und einige Standardalgorithmen sollen bekannt sein. Teile bekannter Systeme werden modellhaft rekonstruiert.

- **Technische Informatik**

Der Aufbau von Computern und Netzwerken, Grundkenntnisse von deren Zusammenwirken, ein technisches Computermodell usw. gehören in dieses Band.

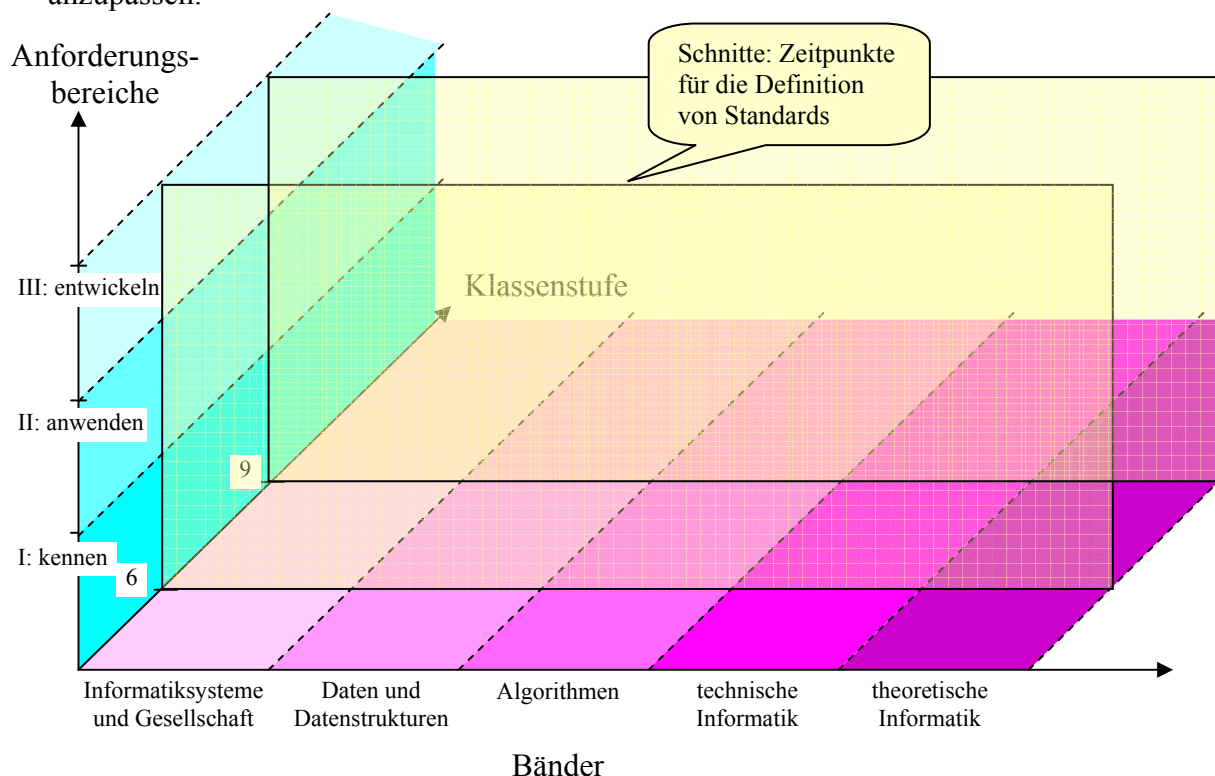
- **Theoretische Informatik**

Hier findet man Codes sowie Notationsformen und Modelle für dynamische Prozesse, insbesondere das Modell der endlichen Automaten.

Zu den Anforderungsbereichen:

Die Anforderungsbereiche werden entsprechend den Bildungsstandards Mathematik formuliert. Die Zusatzbezeichnungen betonen den handlungsorientierten Unterricht.

- **Anforderungsbereich I: Reproduzieren („Kennen“)** legt die Grundlage für die anderen beiden Niveaus. Aus der Kenntnis der grundlegenden Begriffe, Systeme und Zusammenhänge ergibt sich Orientierung im informatischen Bereich. Das Niveau umfasst die Wiedergabe und direkte Anwendung von grundlegenden Begriffen und Verfahren in einem abgegrenzten Gebiet und einem wiederholenden Zusammenhang.
- **Anforderungsbereich II: Zusammenhänge herstellen („Anwenden“)** erfolgt mit angemessener Kenntnis bezogen auf die Möglichkeiten der benutzten Systeme und die Erfordernisse der Problemstellung. Das Niveau umfasst das Bearbeiten bekannter Sachverhalte, indem Kenntnisse, Fertigkeiten und Fähigkeiten verknüpft werden, die in der Auseinandersetzung mit Informatik auf verschiedenen Gebieten erworben wurden.
- **Anforderungsbereich III: Verallgemeinern und reflektieren („Entwickeln“)** umfasst das Bearbeiten komplexer Gegebenheiten u. a. mit dem Ziel, zu eigenen Problemformulierungen, Lösungen, Begründungen, Folgerungen, Interpretationen oder Wertungen zu gelangen. Der Begriff „Entwickeln“ ist bewusst „technisch“ gefasst. Damit wird die Erweiterung des Spektrums der Schulfächer um ein „technisches Fach“ Informatik betont (s.o.). Das Niveau erfordert Kenntnisse und Verständnis für die vorhandenen Anwendungsmöglichkeiten, um diese um neue Möglichkeiten zu erweitern bzw. den Erfordernissen gezielt anzupassen.



Standards

Weil es erst einmal gilt, einen inhaltlichen Rahmen abzustecken, werden die folgenden Standards nur tabellarisch aufgeführt, nicht ausformuliert in der Form „Die Schülerinnen und Schüler ...“.

Standards, die bis Ende der Klassenstufe 6 erreicht sein sollen:

Ähnlich wie in den Naturwissenschaften wird es in diesen Altersstufen eine wesentliche Aufgabe des Unterrichts sein, vorhandene außerschulische Erfahrungen der Schülerinnen und Schüler zu sammeln, zu ordnen, Defizite zu erkennen und auszugleichen, Strukturen und Zusammenhänge deutlich zu machen und die Fachsprache zu entwickeln. Es muss eine informatische Begriffsbildung erfolgen, wobei die Begriffe aus Erfahrungen abstrahiert werden. Dafür ist es notwendig, verschiedene Informatiksysteme zum Vergleich kennen zu lernen. Entwickelt werden sollen auch die Grundbegriffe der objektorientierten Beschreibung von Informatiksystemen (Objekt, Attribut, Attributwert, Methode, Klasse und Beziehung, Ereignis und Ereignisbehandlung), deren Anwendung zur Erklärung des prinzipiellen Aufbaus und der Funktionsweise von gebräuchlichen Standardsoftwaresystemen und die Verwendung hierarchischer Strukturen zur Ordnung von Informationen sowie von vernetzten Strukturen zur Darstellung inhaltlicher Zusammenhänge von Einzelinformationen.

Ausgegangen wird von Unterrichtsprojekten, entweder verteilt im Fachunterricht, im Fachunterricht Informatik oder in Arbeitsgemeinschaften durchgeführt, in denen die folgenden Werkzeuge benutzt werden:

Werkzeug	Beispiele für Anwendungsfälle
Textverarbeitungssystem (incl. Tabellen)	<ul style="list-style-type: none">• Klassenzeitung, Projekttagzeitung• Steckbrief (Vorstellung der eigenen Person)• kreatives Schreiben
Präsentationssystem und/oder HTML-Site (incl. Tabellen und Aktionen)	<ul style="list-style-type: none">• Augenfehler und ihre Korrekturen• eigene Homepage• das Leben der Koalas
Mailsystem	<ul style="list-style-type: none">• Informationsverteilung innerhalb der Klasse• Kontakte zu Partnerschulen
Browser/WWW	<ul style="list-style-type: none">• Recherchen für den Fachunterricht• Informationssuche zur Klassenfahrt (Fahrplanauskünfte, Möglichkeiten am Zielort, ...)
Grafiksystem	<ul style="list-style-type: none">• Verfremden von Digitalfotos• Bildbearbeitung für WWW, Textverarbeitung, Präsentation

Band 1: Informatiksysteme und Gesellschaft

Anmelden	<ul style="list-style-type: none">• Authentifizieren am Computer, Netzwerk, Handy, Mailer, ...• Datenschutz bzgl. der Preisgabe persönlicher Daten
Starten von Programmen	<ul style="list-style-type: none">• Oberfläche, Symbole, (Kontext-)Menüs
Speichern, Laden, Drucken von Dokumenten	<ul style="list-style-type: none">• Geräte, Verzeichnisse, Dokumenttypen, Überschreiben und Kopieren, verschiedene Drucker
Umgang mit Texten	<ul style="list-style-type: none">• Erstellen, Ändern, Kopieren, Ausschneiden, Einfügen• Formatieren
Umgang mit Grafiken	<ul style="list-style-type: none">• Erstellen, Ändern, Kopieren, Ausschneiden, Einfügen• Erfassen von Grafiken (Scannen, Digitalfoto, ...)

Umgang mit Multimedia	<ul style="list-style-type: none"> • multimediale Elemente zusammenfügen (Text, Bild, Ton, Tabelle, ...), anordnen, formatieren • Zusammenwirken multimedialer Elemente (zeitliche Abfolge, Aktionen, Reaktionen) • Vernetzung (HTML-System, Präsentation, ...)
Umgang mit Email	<ul style="list-style-type: none"> • Erstellen, Senden, Empfangen und Beantworten von Nachrichten • Versenden von Dokumenten • Netiquette • Spam, Viren, Trojaner
Umgang mit dem WWW	<ul style="list-style-type: none"> • Dienste (Email, WWW, FTP) • Nutzung von Suchmaschinen • Datensicherheit, Schlüssel • Zitieren • Urheberrecht

Band 2: Daten und Datenstrukturen

einfache Datentypen	<ul style="list-style-type: none"> • Bit, Byte bei Farben (RGB, ...) • Zahl, Zeichen bei Formatierungen in Tabellen • Zeichen, Wort, Satz, Absatz, Abschnitt, Dokument bei Texten
strukturierte Typen (OOM)	<ul style="list-style-type: none"> • Eigenschaften von Grafikobjekten, bei Elementen einer Präsentation oder Formatierungen in Texten
Feld (Reihung)	<ul style="list-style-type: none"> • Tabellen • Pixelgrafiken
Liste (Reihung)	<ul style="list-style-type: none"> • Zeichenketten, Texte, Bildfolgen, Folienfolgen
Baum	<ul style="list-style-type: none"> • Verzeichnisstrukturen • Adressierung im WWW • Gruppierung von z. B. grafischen Objekten
Graph	<ul style="list-style-type: none"> • in HTML-Sites • in Präsentationen

Band 3: Algorithmen

Ereignissteuerung	<ul style="list-style-type: none"> • Reaktion auf „Buttonclicks“ in HTML-Seiten, Präsentationen
Steuerung durch Setzen von Objekteigenschaften	<ul style="list-style-type: none"> • in Präsentationen, Vektorgrafiken

Band 4: Technische Informatik

Arbeitsplatzcomputer	<ul style="list-style-type: none"> • Komponenten und deren Funktionen • Zusammenspiel der Komponenten • Starten eines Computers
lokales Netzwerk	<ul style="list-style-type: none"> • Struktur des Netzwerks • Client-Server-Prinzip (bei der Anmeldung im Netzwerk)
Internet	<ul style="list-style-type: none"> • Struktur des Internets (Router, DNS)

Band 5: Theoretische Informatik

Codes	<ul style="list-style-type: none">• Dualsystem• RGB• Zeichencodes (ASCII, Verschlüsseln, ...)
Daten und Information	<ul style="list-style-type: none">• Inhalt von Pixelgrafiken• bei Anfragen an Suchmaschinen

Standards, die bis Ende der Klassenstufe 9 erreicht sein sollen:

Nachdem in den vorangegangenen Klassenstufen die Nutzung von Informatiksystemen im Vordergrund stand, sollen jetzt Zusammenhänge gezeigt und Strukturen erkannt werden. Die Möglichkeiten und Gefahren des Einsatzes von Informatiksystemen werden anhand von Fallbeispielen thematisiert. Die algorithmische Beschreibung einfacher, automatisierbarer Vorgänge sowie die Rekonstruktion von bekannten Teilsystemen dienen einerseits dem Verständnis, andererseits der Hinführung zur Entwicklung neuer (Modell-)Systeme. Dafür ist die strukturierte Zerlegung und Beschreibung von Informatiksystemen erforderlich. Die explizite Behandlung technischer Details im Kontext komplexer Fragestellungen schafft Bewusstsein für die gesellschaftliche Bedeutung technischer Systeme.

Es werden die folgenden Werkzeuge benutzt:

Werkzeug	Beispiele für Anwendungsfälle
HTML-Entwicklungssystem	<ul style="list-style-type: none">• dynamische HTML-Seiten• Skriptsprache
Digitalbausteine oder Digitalsimulator	<ul style="list-style-type: none">• Entwurf von Schaltungen• Addier-/Subtrahierwerke• elementare programmierbare Rechenwerke• von-Neumann-Rechner
Tabellenkalkulation	<ul style="list-style-type: none">• Planung von Klassenfahrten (Preiskalkulation, ...)• Zufallsexperimente• funktionale Zusammenhänge / grafische Darstellungen• Auswertung von Daten (Mittelwerte, ...)
Datenbanksystem / SQL	<ul style="list-style-type: none">• Planung der Projektstage• Anfragen an Netzdatenbanken
Programmentwicklungssystem	<ul style="list-style-type: none">• LEGO-Roboter, Kara• IDE mit höherer Programmiersprache• Skriptsprache (PHP, ...)• Fujaba

Band 1: Informatiksysteme und Gesellschaft

Anmelden	<ul style="list-style-type: none">• bei Datenbanksystemen, Servern• Datenschutz bzgl. personenbezogener Daten
Umgang mit Texten	<ul style="list-style-type: none">• Layout und Formatvorlagen• Anfertigen von Facharbeiten (Quellenangaben, Literaturangaben, Fußnoten, ...)
Umgang mit Grafiken	<ul style="list-style-type: none">• Farbtiefe, Dateiformat, Kompression• DPI
Umgang mit Multimedia	<ul style="list-style-type: none">• Entwurf und Organisation von Präsentationen / HTML-Sites• Navigation in Präsentationen / HTML-Sites• Ereignissteuerung von Präsentationen / HTML-Sites

Umgang mit Datenbanksystemen	<ul style="list-style-type: none"> • Entwurf einer Datenbank im ER-Modell • Anlegen von Datenbanken und Tabellen • Eingeben, Ändern und Löschen von Daten • Setzen von Zugriffsrechten /Sichten • Befragen einer Datenbank in SQL • Datenschutzfragen
------------------------------	---

Band 2: Daten und Datenstrukturen

einfache Datentypen	<ul style="list-style-type: none"> • boolesche Größen, z. B. in der Tabellenkalkulation oder bei Datenbanken („ist vorhanden“, ...)
strukturierte Typen (OOM)	<ul style="list-style-type: none"> • am Beispiel von Datenbanken • in Programmen, abhängig vom Projekt
Feld (Reihung)	<ul style="list-style-type: none"> • Relationen, Tabellen (Funktionswerttabellen)
Baum	<ul style="list-style-type: none"> • als Möglichkeit in einer Präsentation oder HTML-Site
Graph	<ul style="list-style-type: none"> • Verweise in einer Datenbank, Tabellenkalkulation

Band 3: Algorithmen

Ereignissteuerung	<ul style="list-style-type: none"> • Reaktion auf Ereignisse in HTML-Seiten, Präsentationen, Skripten oder Programmen, abhängig vom Projekt
Steuerung durch Setzen von Objekteigenschaften	<ul style="list-style-type: none"> • in einer IDE oder im Programmtext, abhängig vom Projekt
Sequenz, Alternative	<ul style="list-style-type: none"> • in Tabellenkalkulation, Präsentation, Skript oder Programm
Wiederholung	<ul style="list-style-type: none"> • als Programmstruktur oder durch Ereignisse ausgelöst
Suchen und Sortieren	<ul style="list-style-type: none"> • an verschiedenen Beispielen, abhängig vom Projekt • in Datenbanken • in Tabellen
Darstellung von Algorithmen	<ul style="list-style-type: none"> • durch Struktogramme • Beschreibung von Objektmethoden, abhängig vom Projekt
Umsetzen von Algorithmen in imperative Programmstrukturen	<ul style="list-style-type: none"> • an verschiedenen Beispielen, abhängig vom Projekt

Band 4: Technische Informatik

Arbeitsplatzcomputer	<ul style="list-style-type: none"> • Aufgaben und Oberflächen von Betriebssystemen • Speichern auf Festplatte, optischer Platte, RAM, ROM
lokales Netzwerk	<ul style="list-style-type: none"> • logische Laufwerke, Zugriffsrechte • Sicherheit im Netzwerk
Internet	<ul style="list-style-type: none"> • TCP/IP, Protokolle (HTTP, FTP) und Ports
(Digital-)Bausteine	<ul style="list-style-type: none"> • UND, ODER, NICHT, Schalter, LED
Anwendungen von Digitalbausteinen	<ul style="list-style-type: none"> • Vergleicher, Addierer • Steuerung von Komponenten (Alarmschaltung)
Rechnerstruktur	<ul style="list-style-type: none"> • von-Neumann-Rechner

Band 5: Theoretische Informatik

Codes	<ul style="list-style-type: none"> • Verschlüsseln/Entschlüsseln mithilfe endlicher Automaten
endliche Automaten	<ul style="list-style-type: none"> • Transitionsgraph • akzeptierte Sprache

Daten und Information	<ul style="list-style-type: none"> • am Beispiel des Inhalts von Datenbanken
Syntax und Semantik von Sprachen	<ul style="list-style-type: none"> • formale Sprache von endlichen Automaten • HTML-Quelltext und seine Darstellung • Bedeutung von Sprachkonstrukten

Beispiele für die Anforderungsbereiche

Verzeichnisstrukturen als Baum		
Kennen	Anwenden	Entwickeln
<ul style="list-style-type: none"> • in Verzeichnissen navigieren • Dateien speichern und finden 	<ul style="list-style-type: none"> • Unterverzeichnisse anlegen • Dateien organisiert ablegen 	<ul style="list-style-type: none"> • für ein Projekt mit mehreren Teilnehmer/inne/n eine Verzeichnisstruktur planen

Umgang mit Grafiken		
Kennen	Anwenden	Entwickeln
<ul style="list-style-type: none"> • Fotos erfassen, laden und speichern 	<ul style="list-style-type: none"> • Grafiken in Texte oder Präsentationen einbinden, und formatieren 	<ul style="list-style-type: none"> • Grafiken verändern • Objekte freistellen • mit Ebenen arbeiten

Ereignissteuerung		
Kennen	Anwenden	Entwickeln
<ul style="list-style-type: none"> • in HTML-Sites navigieren 	<ul style="list-style-type: none"> • vorgefertigte Knöpfe in Präsentationen einsetzen • vorgefertigte Aktionen in Präsentationen benutzen 	<ul style="list-style-type: none"> • eine Präsentation planen und Ereignisse mit vorgegebenen Aktionen angemessen beantworten

RGB		
Kennen	Anwenden	Entwickeln
<ul style="list-style-type: none"> • den RGB-Farbraum kennen • Farben RGB-Werte grob zuordnen und umgekehrt 	<ul style="list-style-type: none"> • Farben in Grafiken mithilfe bekannter Werkzeuge gezielt anpassen 	<ul style="list-style-type: none"> • einen RGB-Farbmixer rekonstruieren