



Curriculum für das Fach Informatik in der Sekundarstufe II

- "In der Informatik geht es genauso wenig um Computer wie in der Astronomie um Teleskope" Edsger W. Dijkstra -

Das Fach Informatik dient nicht allein der Beschäftigung mit dem "Wie" in der Datenverarbeitung sondern vielmehr der Schulung des sogenannten "Informatischen Denkens", welches in der heutigen Zeit zu den alltagswichtigen Grundkompetenzen wie Lesen, Schreiben oder Rechnen gehört. Da dieses für den Erziehungsauftrag der Schule wichtige Lernfeld in den anderen Fächern bestenfalls am Rande behandelt wird, kommt dem Informatikunterricht eine besondere Bedeutung zu.

Ausstattung der Schule

Die IGS Bonn Beuel verfügt über drei Computerräume: C212, C213 und B033. Der Raum B033 verfügt über 13 Computerarbeitsplätze für die SuS an den Seiten und der Rückwand sowie einen Arbeitsbereich ohne Computer im zentralbereich des Raums. Zusätzlich gibt es einen Beamer und ein Smartboard, welches nicht an einen PC angeschlossen ist, passende Laptops sind nicht verfügbar. Einen Lehrer-PC gibt es nicht Eine Tafel ist nicht vorhanden. An der Vorderseite sind Verschlussysteme für die Lernroboter "Lego Mindstorms", in der Version NXT und EV3, der Schule verbaut. Der Raum C 213 verfügt über 15 Computerarbeitsplätze in Einzelboxen entlang der Wände sowie einen Beamer und einen daran angeschlossenen Lehrer-PC. Eine Tafel ist nicht vorhanden. Der Raum C 212 verfügt über 15 Computerarbeitsplätze auf den Tischen entlang der Wände sowie einen Beamer und einen daran angeschlossenen Lehrer-PC. Dies ist der einzige PC-Raum mit eingebauter Tafel in Form eines Whiteboards.

Lehrmittel

Da es aktuell kein Umfassendes Lehrwerk für Informatik gibt, das den qualitativen Anforderungen gerecht wird, werden ausschließlich Materialien der unterrichtenden Lehrkraft in (ggf. digitaler) Kopie verwendet. Sollte ein adäquates Lehrwerk auf den Markt kommen wird dies auf Antrag eines Mitgliedes in der Fachkonferenz vorgestellt und dessen Einführung ggf. diskutiert.

Zuordnung der Curricularen Vorgaben zu den Jahrgangsstufen

Das Curriculum stützt sich auf die Anforderungen nach dem Kernlehrplan für die Sekundarstufe II des Ministeriums für Schule und Weiterbildung des Landes NRW aus dem Jahr 2014 sowie den jeweiligen Anforderungen an das Zentralabitur in NRW. Als Programmiersprache wird Java eingesetzt.

Aufgrund der Heterogenität der Lernbedürfnisse der SuS bei informatischen Themen soll bewusst auf das Setzen verbindlicher Zeiträume für die Unterrichtseinheiten verzichtet werden. Die Unterrichtende Lehrkraft entscheidet über die Zeitbedarfe nach den individuellen Lernfortschritten der Gruppe. Besonders schnellen SuS soll hierbei durch frei wählbare Vertiefungsaufgaben (u.a. in Form von Projekten) die Möglichkeit gegeben werden, ihre individuellen Stärken weiter zu entwickeln.

Kursinhalte der Kurse "Informatik" in der EF:

Thema	Unterthemen	Kompetenzen Kernlehrplan	lt.
Was ist Informatik?	<ul style="list-style-type: none"> • Begriffsdefinition • Geschichte des Fachs • Grundprobleme des Fachs • Überblick über verschiedene Programmierkonzepte 	<p>Die SuS</p> <ul style="list-style-type: none"> • beschreiben und erläutern den strukturellen Aufbau und die Arbeitsweise singulärer Rechner am Beispiel der "Von-Neumann-Architektur" (A) • erläutern wesentliche Grundlagen der Geschichte der digitalen Datenverarbeitung(A) • stellen ganze Zahlen und Zeichen in Binärcodes dar (D) • interpretieren Binärcodes als Zahlen und Zeichen (D) • nutzen das Internet zu Recherche, Datenaustausch und Kommunikation (K) • bewerten anhand von Fallbeispielen die Auswirkungen des Einsatzes von Informatiksystemen(A) 	
Objektorientierte Modellierung	<ul style="list-style-type: none"> • Grundbegriffe Objekt, Klasse, Instanz, etc. • Klassendiagramm, Entwurfsdiagramm • Relationen zwischen Klassen • Differenzierung verschiedener Datentypen (Zahl, Text, Wahrheitswert) 	<p>Die SuS</p> <ul style="list-style-type: none"> • ermitteln bei der Analyse einfacher Problemstellungen Objekte, ihre Eigenschaften ihre Operationen und ihre Beziehungen(M) • modellieren Klassen mit ihren Attributen, ihren Methoden und Assoziationsbeziehungen (M) • modellieren Klassen unter Verwendung von Vererbung(M) • ordnen Klassen, Attributen und Methoden ihren Sichtbarkeitsbereichen zu(M) • stellen den Zustand eines Objekts dar(D) • nutzen die im Unterricht eingesetzten Informatiksysteme selbstständig, sicher, zielführend und verantwortungsbewusst(D) • stellen die Kommunikation zwischen Objekten grafisch dar(M) • stellen Klassen, Assoziations- und Vererbungsbeziehungen in Diagrammen grafisch dar(D) • dokumentieren Klassen durch Beschreibung der Funktionalität der Methoden(D) • analysieren und erläutern eine objektorientierte Modellierung (A) • verwenden Fachausdrücke bei der Kommunikation über 	

		informatische Sachverhalte (K)
Grundlagen der Programmierung	<ul style="list-style-type: none"> • Aufbau einer Klasse in Java • "einfache" Datentypen (int, boolean...) • erste Programme als Methode innerhalb einer Klasse • Kontrollstrukturen • Programmablaufplan, Struktogramm • Parameter und Rückgabewerte • Der Algorithmusbegriff 	<p>Die SuS</p> <ul style="list-style-type: none"> • analysieren und erläutern einfache Algorithmen und Programme(A) • ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen oder lineare Datensammlungen zu(M) • entwerfen einfache Algorithmen und stellen sie umgangssprachlich und grafisch dar(M) • nutzen die im Unterricht eingesetzten Informatiksysteme selbstständig, sicher, zielführend und verantwortungsbewusst(D) • implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken(I) • modifizieren einfache Algorithmen und Programme(I) • implementieren Algorithmen unter Verwendung von Variablen und Wertzuweisungen, Kontrollstrukturen sowie Methodenaufrufen(I) • testen Programme schrittweise anhand von Beispielen(I) • implementieren einfache Algorithmen unter Beachtung der Syntax und Semantik einer Programmiersprache (I) • interpretieren Fehlermeldungen und korrigieren den Quellcode(I) • nutzen das verfügbare Informatiksystem zur strukturierten Verwaltung und gemeinsamen Verwendung von Daten unter Berücksichtigung der Rechteverwaltung(K)
Programmierung über mehrere Klassen	<ul style="list-style-type: none"> • Interaktion von Objekten • Datentypen im Speicher; Seiteneffekte • Modularisierung • Zerlegung von Problemen • Rekursion • Arrays als erste Datensammlung <p>hier kann auch ein komplexeres Projekt umgesetzt werden, in</p>	<p>Die SuS</p> <ul style="list-style-type: none"> • nutzen die im Unterricht eingesetzten Informatiksysteme selbstständig, sicher, zielführend und verantwortungsbewusst(D) • implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken(I) • interpretieren Fehlermeldungen und korrigieren den Quellcode(I) • nutzen das verfügbare

	dem die SuS ein gegebenes Problem größeren Ausmaßes lösen.	Informatiksystem zur strukturierten Verwaltung und gemeinsamen Verwendung von Daten unter Berücksichtigung der Rechteverwaltung(K) <ul style="list-style-type: none"> • nutzen das Internet zu Recherche, Datenaustausch und Kommunikation (K)
Suchen und Sortieren	<ul style="list-style-type: none"> • Anforderungen von Menschen an Informatiksysteme • Suchen über Arrays • verschiedene Sortierverfahren quadratischer Komplexität im Vergleich • Komplexitätsanalyse • Einführung eines Sortierverfahrens logarithmischer Komplexität 	Die SuS <ul style="list-style-type: none"> • analysieren Such- und Sortieralgorithmen und wenden sie auf Beispiele an(D) • entwerfen einen weiteren Algorithmus zum Sortieren(M) • nutzen die im Unterricht eingesetzten Informatiksysteme selbstständig, sicher, zielführend und verantwortungsbewusst(D) • beurteilen die Effizienz von Algorithmen am Beispiel von Sortierverfahren hinsichtlich Zeitaufwand und Speicherplatzbedarf (A) • bewerten anhand von Fallbeispielen die Auswirkungen des Einsatzes von Informatiksystemen(A) • verwenden Fachausdrücke bei der Kommunikation über informatische Sachverhalte (K)

Zusätzlich sollte in angemessenem Abstand zu den Kurswahlen für die Qualifikationsstufen (Q1/Q2) mit jedem SuS ein Beratungsgespräch geführt werden, bei dem eine Empfehlung bezüglich einer möglichen Abwahl, Umwahl oder Leistungskurswahl im Fach Informatik auf Grundlage der bisherigen Leistungen in der Einführungsphase ausgesprochen wird.

Die Qualifikationsphase:

Die Kurse in der Qualifikationsphase werden nach den Vorgaben des Zentralabiturs NRW umgesetzt. Hierbei werden folgende Themen abgedeckt:

Kursinhalte der Kurse "Informatik" in der **Q1**:

Thema	Unterthemen	Kompetenzen Kernlehrplan	lt.
Lineare Datenstrukturen	<ul style="list-style-type: none"> • ArrayList • LinkedList • (ggf. als Ausblick: DoubleLinkedList) • Suchen und Sortieren über Listen • Queue • Stack 	<p>Die SuS</p> <ul style="list-style-type: none"> • erläutern Operationen dynamischer (linearer oder nichtlinearer) Datenstrukturen (A), • analysieren und erläutern Algorithmen und Programme (A), • untersuchen und beurteilen Grenzen des Problemlösens mit Informatiksystemen (A). • beurteilen die Effizienz von Algorithmen unter Berücksichtigung des Speicherbedarfs und der Zahl der Operationen (A), • beurteilen die syntaktische Korrektheit und die Funktionalität von Programmen (A), • ermitteln bei der Analyse von Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M), • entwickeln iterative und rekursive Algorithmen unter Nutzung der Strategien „Modularisierung“ und „Teilen und Herrschen“ (M), • implementieren und erläutern iterative und rekursive Such- und Sortierverfahren (I), • implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I), • modifizieren Algorithmen und Programme (I), • implementieren iterative und rekursive Algorithmen auch unter Verwendung von dynamischen Datenstrukturen (I), • testen Programme systematisch anhand von Beispielen (I). • ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen sowie lineare und nichtlineare Datensammlungen zu (M), • verwenden bei der Modellierung geeigneter 	

		<p>Problemstellungen Möglichkeiten der Polymorphie (M),</p> <ul style="list-style-type: none"> • nutzen die Syntax und Semantik einer Programmiersprache bei der Implementierung und zur Analyse von Programmen (I), • interpretieren Fehlermeldungen und korrigieren den Quellcode (I), • wenden eine didaktisch orientierte Entwicklungsumgebung zur Demonstration, zum Entwurf, zur Implementierung und zum Test von Informatiksystemen an (I). • stellen die Kommunikation zwischen Objekten grafisch dar (D), • stellen Klassen und ihre Beziehungen in Diagrammen grafisch dar (D), • dokumentieren Klassen (D), • stellen iterative und rekursive Algorithmen umgangssprachlich und grafisch dar (D), • stellen lineare und nichtlineare Strukturen grafisch dar und erläutern ihren Aufbau (D), • nutzen bereitgestellte Informatiksysteme und das Internet reflektiert zur Erschließung, Aufbereitung und Präsentation fachlicher Inhalte (D), • nutzen das verfügbare Informatiksystem zur strukturierten Verwaltung von Dateien unter Berücksichtigung der Rechteverwaltung (K),
<p>Verzweigte Datenstrukturen</p>	<ul style="list-style-type: none"> • Bäume • Binärbaum • Binärer Suchbaum 	<p>Die SuS</p> <ul style="list-style-type: none"> • erläutern Operationen dynamischer (linearer oder nichtlinearer) Datenstrukturen (A), • analysieren und erläutern Algorithmen und Programme (A), • untersuchen und beurteilen Grenzen des Problemlösens mit Informatiksystemen (A). • beurteilen die Effizienz von Algorithmen unter Berücksichtigung des Speicherbedarfs und der Zahl der Operationen (A), • beurteilen die syntaktische Korrektheit und die Funktionalität von Programmen (A), • entwickeln iterative und rekursive Algorithmen unter Nutzung der Strategien „Modularisierung“ und „Teilen und Herrschen“ (M), • ordnen Attributen,

Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen sowie lineare und nichtlineare Datensammlungen zu (M),

- verwenden bei der Modellierung geeigneter Problemstellungen Möglichkeiten der Polymorphie (M),

- implementieren und erläutern iterative und rekursive Such- und Sortierverfahren (I),

- modifizieren Algorithmen und Programme (I),

- implementieren Klassen in einer Programmiersprache auch unter Nutzung dokumentierter Klassenbibliotheken (I).

- implementieren iterative und rekursive Algorithmen auch unter Verwendung von dynamischen Datenstrukturen (I),

- testen Programme systematisch anhand von Beispielen (I).

- nutzen die Syntax und Semantik einer Programmiersprache bei der Implementierung und zur Analyse von Programmen (I),

- interpretieren Fehlermeldungen und korrigieren den Quellcode (I),

- wenden eine didaktisch orientierte Entwicklungsumgebung zur Demonstration, zum Entwurf, zur Implementierung und zum Test von Informatiksystemen an (I).

- stellen die Kommunikation zwischen Objekten grafisch dar (D),

- stellen Klassen und ihre Beziehungen in Diagrammen grafisch dar (D),

- dokumentieren Klassen (D),

- stellen lineare und nichtlineare Strukturen grafisch dar und erläutern ihren Aufbau (D),

- stellen iterative und rekursive Algorithmen umgangssprachlich und grafisch dar (D),

- nutzen bereitgestellte Informatiksysteme und das Internet reflektiert zur Erschließung, Aufbereitung und Präsentation fachlicher Inhalte (D),

- nutzen das verfügbare Informatiksystem zur strukturierten Verwaltung von Dateien unter

		Berücksichtigung der Rechteverwaltung (K),
Objektorientierte Modellierung	<ul style="list-style-type: none"> • Entwurfsdiagramm und Implementationsdiagramm vs. UML-Diagramm • Assoziationen • Transformation von Modellen untereinander sowie zwischen Modell und Javacode 	<p>Die SuS</p> <ul style="list-style-type: none"> • analysieren und erläutern objektorientierte Modellierungen (A), • untersuchen und beurteilen Grenzen des Problemlösens mit Informatiksystemen (A). • ermitteln bei der Analyse von Problemstellungen Objekte, ihre Eigenschaften, ihre Operationen und ihre Beziehungen (M), • modellieren Klassen mit ihren Attributen, Methoden und ihren Assoziationsbeziehungen unter Angabe von Multiplizitäten (M), • modellieren abstrakte und nicht abstrakte Klassen unter Verwendung von Vererbung durch Spezialisieren und Generalisieren (M), • ordnen Attributen, Parametern und Rückgaben von Methoden einfache Datentypen, Objekttypen sowie lineare und nichtlineare Datensammlungen zu (M), • verwenden bei der Modellierung geeigneter Problemstellungen Möglichkeiten der Polymorphie (M), • ordnen Klassen, Attributen und Methoden ihre Sichtbarkeitsbereiche zu (M), • interpretieren Fehlermeldungen und korrigieren den Quellcode (I), • nutzen die Syntax und Semantik einer Programmiersprache bei der Implementierung und zur Analyse von Programmen (I), • stellen die Kommunikation zwischen Objekten grafisch dar (D), • stellen Klassen und ihre Beziehungen in Diagrammen grafisch dar (D), • dokumentieren Klassen (D), • nutzen bereitgestellte Informatiksysteme und das Internet reflektiert zur Erschließung, Aufbereitung und Präsentation fachlicher Inhalte (D), • nutzen das verfügbare Informatiksystem zur strukturierten Verwaltung von Dateien unter Berücksichtigung der Rechteverwaltung (K),

Netzwerke	<ul style="list-style-type: none"> • Netzwerktopologien • Protokolle • Client-/Server - Architektur • Schichtenmodell der Kommunikation 	<ul style="list-style-type: none"> • Stellen lineare und nichtlineare Strukturen grafisch dar und erläutern ihren Aufbau (D). • beschreiben und erläutern Netzwerk-Topologien, die Client-Server-Struktur und Protokolle sowie ein Schichtenmodell in Netzwerken (A)
-----------	---	--

Kursinhalte der Kurse "Informatik" in der Q2:

Thema	Unterthemen	Kompetenzen Kernlehrplan	lt.
Automatentheorie und Formale Sprachen	<ul style="list-style-type: none"> • DFA • NFA • (ggf. Ausblick auf andere Automaten (Mealy A., Kellerautomat, Unendliche A., etc.)) • Definition Regulärer Sprachen • Chomsky Hierarchie • Grammatiken 	<p>Die SuS</p> <ul style="list-style-type: none"> • analysieren und erläutern die Eigenschaften endlicher Automaten einschließlich ihres Verhaltens bei bestimmten Eingaben (A), • untersuchen und beurteilen Grenzen des Problemlösens mit Informatiksystemen (A). • analysieren und erläutern Algorithmen und Programme (A), • beurteilen die Effizienz von Algorithmen unter Berücksichtigung des Speicherbedarfs und der Zahl der Operationen (A), • analysieren und erläutern Grammatiken regulärer Sprachen (A), • ermitteln die formale Sprache, die durch eine Grammatik erzeugt wird (A), • zeigen die Grenzen endlicher Automaten und regulärer Grammatiken im Anwendungszusammenhang auf (A). • entwickeln und modifizieren zu einer Problemstellung endliche Automaten (M), • entwickeln zur Grammatik einer regulären Sprache einen zugehörigen endlichen Automaten (M). • modifizieren Grammatiken regulärer Sprachen (M), • entwickeln zu einer regulären Sprache eine Grammatik, die die Sprache erzeugt (M), • entwickeln zur akzeptierten Sprache eines Automaten eine zugehörige Grammatik (M), • testen Programme systematisch anhand von Beispielen 	

		<p>(I).</p> <ul style="list-style-type: none"> • beschreiben an Beispielen den Zusammenhang zwischen Automaten und Grammatiken (D). • ermitteln die Sprache, die ein endlicher Automat akzeptiert (D), • stellen lineare und nichtlineare Strukturen grafisch dar und erläutern ihren Aufbau (D), • stellen die Kommunikation zwischen Objekten grafisch dar (D), • stellen endliche Automaten in Tabellen oder Graphen dar und überführen sie in die jeweils andere Darstellungsform (D), • stellen iterative und rekursive Algorithmen umgangssprachlich und grafisch dar (D), • nutzen bereitgestellte Informatiksysteme und das Internet reflektiert zur Erschließung, Aufbereitung und Präsentation fachlicher Inhalte (D), • nutzen das verfügbare Informatiksystem zur strukturierten Verwaltung von Dateien unter Berücksichtigung der Rechteverwaltung (K),
<p>Database Management Systems (DBMS)</p>	<ul style="list-style-type: none"> • Datenbanksysteme und ihre Geschichte • Entity-Relationship-Diagramm • Relationale Algebra • SQL • Verantwortungsvoller Umgang mit Daten 	<p>Die SuS</p> <ul style="list-style-type: none"> • analysieren und erläutern Algorithmen und Programme (A), • analysieren und erläutern eine Datenbankmodellierung (A), • erläutern die Eigenschaften normalisierter Datenbankschemata (A), • beurteilen die syntaktische Korrektheit und die Funktionalität von Programmen (A), • untersuchen und bewerten anhand von Fallbeispielen Auswirkungen des Einsatzes von Informatiksystemen sowie Aspekte der Sicherheit von Informatiksystemen, des Datenschutzes und des Urheberrechts (A), • untersuchen und bewerten Problemlagen, die sich aus dem Einsatz von Informatiksystemen ergeben, hinsichtlich rechtlicher Vorgaben, ethischer Aspekte und gesellschaftlicher Werte unter Berücksichtigung unterschiedlicher Interessenlagen (A). • analysieren und erläutern die Syntax und Semantik einer

Datenbankabfrage (A),

- beurteilen die Effizienz von Algorithmen unter Berücksichtigung des Speicherbedarfs und der Zahl der Operationen (A),
- erläutern Eigenschaften, Funktionsweisen und Aufbau von Datenbanksystemen unter dem Aspekt der sicheren Nutzung (A),
- ermitteln für anwendungsbezogene Problemstellungen Entitäten, zugehörige Attribute, Relationen und Kardinalitäten (M),
- modifizieren eine Datenbankmodellierung (M),
- modellieren zu einem Entity-Relationship-Diagramm ein relationales Datenbankschema (M),
- bestimmen Primär- und Sekundärschlüssel (M),
- überführen Datenbankschemata in die 1. bis 3. Normalform (M).
- testen Programme systematisch anhand von Beispielen (I).
- verwenden die Syntax und Semantik einer Datenbankabfragesprache, um Informationen aus einem Datenbanksystem zu extrahieren (I).
- wenden eine didaktisch orientierte Entwicklungsumgebung zur Demonstration, zum Entwurf, zur Implementierung und zum Test von Informatiksystemen an (I).
- stellen Entitäten mit ihren Attributen und die Beziehungen zwischen Entitäten mit Kardinalitäten in einem Entity-Relationship-Diagramm grafisch dar (D),
- ermitteln Ergebnisse von Datenbankabfragen über mehrere verknüpfte Tabellen (D)
- überprüfen Datenbankschemata auf vorgegebene Normalisierungseigenschaften (D),
- stellen iterative und rekursive Algorithmen umgangssprachlich und grafisch dar (D),
- nutzen bereitgestellte Informatiksysteme und das Internet reflektiert zur Erschließung, Aufbereitung und Präsentation fachlicher Inhalte (D),

		<ul style="list-style-type: none"> • nutzen das verfügbare Informatiksystem zur strukturierten Verwaltung von Dateien unter Berücksichtigung der Rechteverwaltung (K),
Abiturvorbereitung / Wiederholung	<ul style="list-style-type: none"> • Zusammenhang verschiedener Teilthemen anhand Fundamentaler Ideen • Prüfungen im Abitur im Fach Informatik 	<p>Die SuS</p> <ul style="list-style-type: none"> • analysieren und erläutern Algorithmen und Programme (A), • beurteilen die Effizienz von Algorithmen unter Berücksichtigung des Speicherbedarfs und der Zahl der Operationen (A), • analysieren und erläutern Eigenschaften und Einsatzbereiche symmetrischer und asymmetrischer Verschlüsselungsverfahren (A). • erläutern die Ausführung eines einfachen maschinennahen Programms sowie die Datenspeicherung auf einer „Von-Neumann-Architektur“ (A), • beschreiben und erläutern Netzwerk-Topologien, die Client-Server-Struktur und Protokolle sowie ein Schichtenmodell in Netzwerken (A). • modifizieren Algorithmen und Programme (I), • stellen iterative und rekursive Algorithmen umgangssprachlich und grafisch dar (D), • nutzen bereitgestellte Informatiksysteme und das Internet reflektiert zur Erschließung, Aufbereitung und Präsentation fachlicher Inhalte (D), • nutzen das verfügbare Informatiksystem zur strukturierten Verwaltung von Dateien unter Berücksichtigung der Rechteverwaltung (K),

Sobald die Vorgaben zum Zentralabitur andere Kursinhalte vorschreiben sind diese zu unterrichten, die obige Aufstellung wird in diesem Fall schnellstmöglich überarbeitet.

Bei der Erstellung der konkreten Unterrichtsinhalte sollte die Identifikation der Fundamentalen Ideen hinter den jeweiligen Einzelproblemen stehen. Die SuS sollen ein Bewusstsein für die Reduktion von Alltagsproblemen auf Standardsituationen entwickeln. Um die Allgemeingültigkeit dieser Fundamentalen Ideen zu verdeutlichen sollten nach Möglichkeit Alltagsbeispiele mit Bezug zur Lebenswirklichkeit der SuS gefunden werden.

Bewertungsgrundlagen

Die Grundlage für Messung und Bewertung von Lernfortschritten der SuS sollen immer die fünf Kompetenzbereiche des Faches Informatik Argumentieren (A), Modellieren (M), Implementieren (I), Darstellen und Interpretieren (D) sowie Kommunizieren und Kooperieren (K) sein. Verpflichtende Bewertungsgegenstände sind hierbei die Kommunikation im Unterricht UND die Arbeit am PC. Hierzu können noch angekündigte und unangekündigte Lernstandsüberprüfungen sowie Projektarbeiten kommen.

Das Fach Informatik kann schriftlich und mündlich gewählt werden. Bei der Ermittlung der Endnote aus den schriftlichen Noten und den sonstigen Mitarbeitsnoten darf nicht nur eine rechnerische Mittelwertnote gebildet werden, sondern es sind pädagogisch begründete Endnoten im von den Einzelnoten gebildeten Notenbereich festzusetzen.

Kooperationen

Im Rahmen der Berufsfeldpraktika sollte den SuS die Gelegenheit gegeben werden, in Betriebe mit der Spezialisierung im Aufgabenfeld EDV und Informatik hineinzugehen und die dortige Arbeit und die Anforderungen an informatiknahe Berufe in der Realität kennenlernen. Wiederkehrende Partner sind beispielsweise Eaton und die SER.

Da die Fachinhalte des Faches Informatik viele Parallelen zu den Fachinhalten der Fächer Mathematik (z.B. im Bereich der Algorithmik oder das gemeinsame Konzept der "Methode") und Physik (z.B. im Bereich der Technischen Informatik oder der Rechnernetze) aufweist, sollte hier wenn möglich Koordiniert vorgegangen werden, der Umfang der Kooperation liegt im Ermessen der unterrichtenden Fachkollegen.

Aufgaben innerhalb der Fachkonferenz werden situationsabhängig zugeteilt. Jedes Mitglied kann Beiträge zur Weiterentwicklung des Faches Informatik in Absprache mit dem Fachvorsitzenden entwickeln, über eine Umsetzung entscheidet die Fachkonferenz. Zu Beginn jeden Schuljahres wird das Curriculum in der Fachkonferenz zur Disposition gestellt.

21.08.18