



Schulinternes Curriculum I n f o r m a t i k (SII) Luisenschule – Mülheim an der Ruhr

(Stand: 7. August 2014)

Vorbemerkungen

Dieses schulinterne Curriculum strukturiert den Kompetenzerwerb im Fach Informatik der gymnasialen Oberstufe (Sekundarstufe II, kurz SII) an der Luisenschule auf Grundlage des Kernlehrplans Informatik¹. Geordnet nach den Jahrgangsstufen (G8) soll es den Akteuren des Informatikunterrichts – den Lernenden wie Lehrenden gleichermaßen – einen Überblick über die zu vermittelnden Kompetenzbereiche (z.B. Implementieren) und Inhaltsfelder (z.B. Algorithmen) verschaffen. Gleichfalls konkretisiert es die verbindlichen Vorgaben.

Das Fach Informatik wird an der Luisenschule ab der Jahrgangsstufe 8 im Wahlpflichtbereich II (WP II) bis hin zur Qualifikationsphase 2 (Q2) unterrichtet, mit der Möglichkeit, dieses als Abiturfach zu belegen und abschließend darin die schriftliche oder mündliche Abiturprüfung abzulegen. Ergänzt wird das Fach Informatik an der Luisenschule derzeit durch die Arbeitsgemeinschaften Robotik und Automatisierungstechnik.

Der Unterricht der SII wird mit Hilfe der Programmiersprache Java durchgeführt. Insbesondere in der Einführungsphase kommen dabei zusätzlich didaktische Programmierbibliotheken (z.B. GeoFaSC² oder GLOOP³) und -umgebungen (z.B. BlueJ) zum Einsatz, welche die Implementation von (grafischen) Programmen erleichtern.

1 Kernlehrplan für die Sekundarstufe II – Gymnasium/Gesamtschule, Informatik, Nordrhein-Westfalen, 2013

2 Siehe GeoFaSC von C. Wolf, <http://www.geofasc.de> (abg. am 29.7.2014)

3 Siehe GLOOP von V. Quade, <http://www.bezreg-duesseldorf.nrw.de/lerntreffs/informatik/structure/material/sek2/einfuehrungen/gloop.php> (abg. am 29.7.2014)

Der Kernlehrplan fordert (vgl. Kernlehrplan S. 14 ff.), dass die fachbezogenen Kompetenzen, die sich in die fünf Kompetenzbereiche

- Argumentieren (A),
- Modellieren (M),
- Implementieren (I),
- Darstellen und Interpretieren (D) sowie
- Kommunizieren und Kooperieren (K)

untergliedern lassen, im Rahmen der obligatorischen Inhaltsfelder

- Daten und ihre Strukturierung,
- Algorithmen,
- Formale Sprachen und Automaten,
- Informatiksysteme sowie
- Informatik, Mensch und Gesellschaft

zu entwickeln sind. Diese werden in der Einführungsphase aufgegriffen und in der Qualifikationsphase vertieft. Die inhaltlichen Schwerpunkte dieser Inhaltsfelder unterscheiden sich in ihrer Ausprägung und Schwierigkeit je nach Jahrgangsstufe.

Das folgende Übersichtsraster zeigt eine Auflistung sogenannter Unterrichtsvorhaben abgegrenzt für jede Jahrgangsstufe. Die Unterrichtsvorhaben strukturieren den Erwerb der genannten Kompetenzen in chronologisch und organisch sinnvoller Weise mit dem Anspruch zur Erfüllung der im Kernlehrplan formulierten Obligatorik.

Die Spalte *UV* bezeichnet das Unterrichtsvorhaben und ist Überbegriff der in der Spalte *Inhaltsbezogene Kompetenzen* angegebenen, zu entwickelnden Kompetenzen mit einem Fokus auf Fachinhalte. Die Spalte *Verknüpfungen zum KLP* stellt die inhaltsbezogenen Kompetenzen in Bezug zu den genannten Vorgaben des Kernlehrplans (Inhaltsfelder und Kompetenzbereiche) und legitimiert diese damit. Die Werte in der Spalte *ZB* geben den geschätzt notwendigen Zeitbedarf für das Unterrichtsvorhaben in Unterrichtsstunden (68-Minuten-Stunden gemäß der Stundentaktung an der Luisenschule) an. Um Freiraum für Vertiefungen, besondere Schülerinteressen, aktuelle Themen bzw. die Erfordernisse anderer besonderer Ereignisse (z.B. Praktika, Kursfahrten o.ä.) zu erhalten, wurden im Rahmen dieses schulinternen Lehrplans ca. 75 Prozent der Bruttounterrichtszeit verplant.

Die Spalte *KUV* (Konkretisierte Unterrichtsvorhaben), *Materialien*, *Medien & Werkzeuge* formuliert mögliche Zugänge, Ansätze und Beispiele zur Durchführung des Unterrichtsvorhabens, nennt jedoch auch geeignete Unterrichtsmaterialien und -gegenstände, z.B. geeignete Software-Projekte, Verweise auf sich empfehlende Lehrwerke oder auch didaktische Hinweise, die sich bei der Umsetzung im Unterricht als nützlich erweisen können. Da Unterrichtsvorhaben in der Regel anhand von Problemstellungen in Anwendungskontexten oder im Rahmen umfassenderer Implementierungen bearbeitet werden, können diese jeweils mehrere Inhaltsfelder ansprechen.

Die aufgelisteten Unterrichtsvorhaben sind verbindlich, die konkretisierten Unterrichtsvorhaben hingegen besitzen einen „empfehlenden/ beispielhaften“ Charakter. So wird dem Unterrichtenden ein Spektrum an Variabilität und Freiheit ermöglicht, um individuell und lerngruppengerecht agieren zu können.

Abschließend sei bemerkt, dass dieses Curriculum zukünftig noch Anpassungen unterworfen sein wird. Zukünftige, praktische Unterrichtserfahrungen wie auch mögliche Änderungen von Vorgaben ziehen gegebenenfalls Änderungen und Optimierungen nach sich.

Im August 2014,

Dipl.-Inform. Christian Wolf, StR
Peter Römer, StD

Unterrichtsvorhaben (EF)

ZB	UV	Inhaltsbezogene Kompetenzen	Verknüpfungen zum KLP	KUV, Materialien, Medien & Werkzeuge
10	Kurzeinführung in die Informatik: Wichtige Begriffe & ausgewählte Prinzipien	Die SuS... <ul style="list-style-type: none"> erörtern die Leitfragen „Was ist Informatik?“, „Womit beschäftigt sich Informatik?“ und „Welches sind die Gebiete (technische, praktische Informatik...) der Informatik?“ begreifen Informatik als Wissenschaft von der automatisierten Verarbeitung von Informationen unterscheiden die Begriffe Information und Daten wenden einige Kodierungen/ Dekodierungen an, stellen ganze Zahlen und Zeichen in Binärcodes dar und umgekehrt beschreiben und erläutern den Aufbau und die Arbeitsweise von Rechnern am Beispiel der Von-Neumann-Architektur identifizieren das grundlegende EVA-Prinzip informationsverarbeitender Systeme erläutern wesentliche Grundlagen der Geschichte der digitalen Datenverarbeitung 	Inhaltsfeld 4: Informatiksysteme (Digitalisierung, Einzelrechner, Internet) Inhaltsfeld 5: Informatik, Mensch und Gesellschaft (Wirkungen der Automatisierung, Geschichte der automatischen Datenverarbeitung) Kompetenzbereiche: A, D, K	Beispiel: Internetrecherche zu den Leitfragen und wichtigsten Begriffen rund um die Informatik; Entwicklung/ Konkretisierung einer Mindmap zu den Kategorien der Informatik. Beispiel: Kodierung und Dekodierung von Texten mit unbekanntem Zeichensätzen oder auch Bildinformationen in Raster- und Vektorgrafiken Z.B. Demontage eines Demonstrationsrechners und Zuordnung der Komponenten zu EVA; Verwendung von Programmen zur Simulationen der VNA. Z.B. Filmdokumentation „Eine Maschine verändert die Welt“.

Jahrgangsstufe:

EF Q1 Q2

ZB	UV	Inhaltsbezogene Kompetenzen	Verknüpfungen zum KLP	KUV, Materialien, Medien & Werkzeuge
4	Paradigma der Objektorientierung (OO)	<p>Die SuS...</p> <ul style="list-style-type: none"> • begreifen das OO-Paradigma als eine abstrakte und menschenähnliche Sicht auf Systeme • lernen die Elemente Objekt, Klasse, Eigenschaft/ Attribut, Attributswert, Dienst/ Methode (als Anfrage und Auftrag) kennen und gebrauchen • unterscheiden die Beziehungen zwischen Objekten/ Klassen mit kennt-, hat- und ist-Beziehung (bzw. Assoziations-, Aggregations-/ Kompositions- und Vererbungsbeziehung) • differenzieren zwischen Spezialisierung und Generalisierung bei der ist-Beziehung • identifizieren und klassifizieren Objekte aus Problembeschreibungen und erstellen einfache Objekt- und Klassen-Karten 	<p>Inhaltsfeld 1: Daten und ihre Strukturierung (Objekte und Klassen)</p> <p>Kompetenzbereiche: A, D, K</p>	<p>Z.B. OO-Ansatz in Grafik- (geometrische Figuren bestehen aus einfacheren geometrischen Figuren mit Attributen) u. Textverarbeitungssystemen (Texte bestehen aus Wörtern, die aus Zeichen bestehen; jedes Zeichen hat bestimmte Attribute und man kann Operationen auf ihm ausführen).*</p> <p>Z.B. Untersuchung der Hierarchie (Vererbung) von Automobilen („Ein PKW ist ein KFZ ist ein Automobil.“)</p> <p>Z.B. einzelne, geometrische Objekte (z.B. Fenster) mit gleichartigen Attributen aus Grafiken (z.B. Kirche) klassifizieren.*</p>

Jahrgangsstufe:

EF Q1 Q2

* Siehe z.B.: Informatik 1 – Objekte, Strukturen, Algorithmen, Klett-Verlag

ZB	UV	Inhaltsbezogene Kompetenzen	Verknüpfungen zum KLP	KUV, Materialien, Medien & Werkzeuge
6	OO-Analyse und -Modellierung (OOA/ OOM)	<p>Die SuS...</p> <ul style="list-style-type: none"> • analysieren Probleme durch die „OO-Brille“, modellieren und stellen sie mittels UML-Elementen dar • entwerfen Objektdiagramme mit Objektkarten (Attribute und Methoden) und den Beziehungen zwischen den Objekten • erfassen die Notwendigkeit und Nützlichkeit von Klassen- gegenüber Objektdiagrammen und aggregieren Objekt- zu Klassendiagrammen • erstellen Klassendiagramme mit Klassen-Karten (Attribute und Methoden) und den Beziehungen (kennt-, hat- und ist-Beziehung) • benutzen Multiplizitäten in Klassendiagrammen und formulieren deren Bedeutung 	<p>Inhaltsfeld 1: Daten und ihre Strukturierung (Objekte und Klasse)</p> <p>Kompetenzbereiche: A, M, D, K</p>	<p>Notationselemente Objekt-, Klassenkarte, Objekt- u. Klassendiagramm angelehnt an die UML.</p> <p>Editoren zum Erstellen von Diagrammen, z.B. yEd Graph Editor oder auch UMLed.</p> <p>Diverse Modellierungsaufgaben, z.B. Szenarien Fluggesellschaft⁴ oder Computer.</p> <p>Klassische Modellierungen wie z.B. Bank mit Kunde, Bankdirektor, Konto... oder auch Schule mit Schüler(in), Lehrer(in), Personaloder auch Bibliothek Bibliothekar, Buch/ Literatur und Benutzer...</p>

Jahrgangsstufe:

EF Q1 Q2

4 Siehe z.B. auch: S. 31 A. 1, 2, 3 aus Informatik 3 – Algorithmen, Objektorientierte Programmierung, Zustandsmodellierung, Klett-Verlag

ZB	UV	Inhaltsbezogene Kompetenzen	Verknüpfungen zum KLP	KUV, Materialien, Medien & Werkzeuge
6	Grundlagen der OO-Programmierung (OOP) in Java	<p>Die SuS...</p> <ul style="list-style-type: none"> • erzeugen Objekte aus gegebenen Klassen und rufen deren Dienste auf • implementieren erste eigene Klassen auf Basis einfacher Modelle und realisieren OOP vielmehr als Definition von Klassen denn Objekten • programmieren Methoden ohne und mit Parametern zur Intra-Objekt-Kommunikation und unterscheiden Anfragen von Aufträgen • benutzen Modifizierer zur Sichtbarkeitskontrolle von Attributen und Methoden • erfassen Kontrollstrukturen zur Ablaufsteuerung von Programmen (z.B. Schleifen und bedingte Anweisungen) • benutzen Kontrollstrukturen⁵ zur Programmierung kleinerer Algorithmen (z.B. Suchalgorithmen) und stellen diese dar 	<p>Inhaltsfeld 1: Daten und ihre Strukturierung (Objekte und Klassen)</p> <p>Inhaltsfeld 3: Formale Sprachen und Automaten (Syntax und Semantik einer Programmiersprache)</p> <p>Inhaltsfeld 4: Informatiksysteme (Dateisystem, Internet)</p> <p>Inhaltsfeld 5: Informatik, Mensch u. Gesellschaft (Einsatz von Informatiksystemen)</p> <p>Kompetenzbereiche: A, M, I, D, K</p>	<p>USB-Stick mit den notwendigen, portablen Applikationen⁶ zur Entwicklung: BlueJ, Java-Editor, JDK...</p> <p>Didaktische Entwicklungsbibliotheken wie z.B. GeoFaSC⁷, GLOOP⁸ oder SuM.</p> <p>Software-Projekte⁹ z.B.:</p> <ul style="list-style-type: none"> • Mini-Golf • Dart mit Pfeil und Dartscheibe • Grafischer-LKW • ... <p>Darstellung über Struktogramme (z.B. mit dem Werkzeug Structorizer⁹)/ Aktivitätsdiagramme.</p>

Jahrgangsstufe:

EF Q1 Q2

5 Einführung in die Aussagenlogik und boolesch'e Algebra zur Thematisierung von verknüpften Bedingungen u.U. erforderlich

6 Siehe Informatik on Stick von T. Hempel, <http://www.tinohempel.de/info/info/loStick/index.html> (abg. am 28.8.2012)

7 Siehe GeoFaSC von C. Wolf, <http://www.geofasc.de> (abg. am 29.7.2014)

8 Siehe GLOOP von V. Quade, <http://www.bezreg-duesseldorf.nrw.de/lerntreffs/informatik/structure/material/sek2/einfuehrungen/gloop.php> (abg. am 29.7.2014)

9 Siehe auch: B. Schriek: Informatik mit Java, Band I, Nili-Verlag Werl

ZB	UV	Inhaltsbezogene Kompetenzen	Verknüpfungen zum KLP	KUV, Materialien, Medien & Werkzeuge
12	Erweiterte OO-Programmierung (OOP) in Java	<p>Die SuS...</p> <ul style="list-style-type: none"> • implementieren eigene Programme auf Basis erweiterter Modelle und stellen die kennt-/ hat-Beziehung eines Objekts als ein Attribut von einem Klassentypen her • programmieren Methoden ohne und mit Parametern (auch Objekte als Parameter) zur Inter-Objekt-Kommunikation • unterscheiden bei Methodenaufrufen „Call-by-value-“ und „Call-by-reference-Parameter“ • verwenden beim Programmieren die ist-Beziehung bei Klassen zur Spezialisierung einer Klasse und Wiederverwendung von Funktionalität • überschreiben geerbte Methoden, um abgeleitete Objekte auch spezialisiert auf Nachrichten reagieren zu lassen (Polymorphie) • fakultativ: verwenden beim Programmieren die ist-Beziehung zur Generalisierung von Klassen hin zu einer abstrakten Klasse und Bündelung gemeinsamer Funktionalität • fakultativ: implementieren einfache interaktive Programme (z.B. mit Tasten-/ Maussteuerung) mit Ereignisbehandlung 	<p>Inhaltsfeld 1: Daten und ihre Strukturierung (Objekte und Klassen)</p> <p>Inhaltsfeld 3: Formale Sprachen und Automaten (Syntax und Semantik einer Programmiersprache)</p> <p>Inhaltsfeld 4: Informatiksysteme (Dateisystem, Internet)</p> <p>Inhaltsfeld 5: Informatik, Mensch u. Gesellschaft (Einsatz von Informatiksystemen)</p> <p>Kompetenzbereiche: A, M, I, D, K</p>	<p>Software-Projekte¹⁰ z.B.:</p> <ul style="list-style-type: none"> • Fliegende Kugeln, bei der Kugeln voneinander abprallen; dazu muss jede Kugel alle anderen kennen und es ist eine Kollisionsbehandlung erforderlich • Ampel/ Disco-Licht, bei dem ein Disco-Licht eine Menge von LED-Bars und eine LED-Bar eine Menge von LEDs hat • Blinking-LED, bei der eine solche eine LED ist, die zusätzlich blinken kann • Windradpark • Taschenrechner mit GUI oder interaktives Spiel (wie Ping-Pong)

Jahrgangsstufe:

EF **Q1** **Q2**

¹⁰Siehe z.B.: B. Schriek: Informatik mit Java, Band I, Nili-Verlag Werl

ZB	UV	Inhaltsbezogene Kompetenzen	Verknüpfungen zum KLP	KUV, Materialien, Medien & Werkzeuge
8	Such- und Sortieralgorithmen ¹¹	<p>Die SuS...</p> <ul style="list-style-type: none"> • analysieren, erläutern und modifizieren einfache Algorithmen und Programme • beschäftigen sich insbesondere mit einfachen Such- und Sortieralgorithmen, wozu die Behandlung 1-dim. Felder (Arrays) erforderlich und auch sinnvoll erscheint, wie z.B.: <ul style="list-style-type: none"> ◦ Bubble-Sort ◦ Insertion-Sort ◦ Lineare Suche ◦ Prinzip der binären Suche (z.B. anhand des Erratens ausgedachter Zahlen) • entwerfen auch einfache Algorithmen (z.B. Modifikationen einfacher Such- und Sortieralgorithmen) und stellen sie umgangssprachlich und grafisch dar (z.B. mittels Aktivitäts- oder Flussdiagrammen) 	<p>Inhaltsfeld 2: Algorithmen (Analyse, Entwurf und Implementierung einfacher Algorithmen; Algorithmen zum Suchen und Sortieren)</p> <p>Kompetenzbereiche: A, M, I</p>	<p>Beispiel: Sortieren mit Waage SuS sollen kleine, optisch identische Kunststoffbehälter aufsteigend nach ihrem Gewicht zu sortieren. Dazu steht ihnen eine Balkenwaage zur Verfügung, mit deren Hilfe sie das Gewicht zweier Behälter vergleichen können¹².</p> <p>Beispiel: Simulationsspiel zur binären Suche nach Tischtennisbällen. Mehrere Bälle sind nummeriert, sortiert und unter Bechern verdeckt. Mit Hilfe der binären Suche kann sehr schnell ein bestimmter Tischtennisball gefunden werden¹³.</p>

Jahrgangsstufe:

EF **Q1** **Q2**

¹¹ Bemerkung: Durch die vorangehenden Unterrichtsvorhaben mit dem Fokus auf die OOP ist es sehr wahrscheinlich, dass die SuS bereits einfache Algorithmen analysiert, entwickelt und auch implementiert haben.

¹² Siehe: Computer science unplugged – Sorting Algorithms, <http://www.csunplugged.org/sorting-algorithms> (abg. am 30.03.2014)

¹³ Siehe: Computer science unplugged – Searching Algorithms, <http://www.csunplugged.org/searching-algorithms> (abg. am 30.03.2014)

Grundsätze zur Leistungsbewertung

Auf der Grundlage von §13 - §16 der APO-GOST sowie Kapitel 3 des Kernlehrplans Informatik für die gymnasiale Oberstufe hat die Fachkonferenz der Luisenschule 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. Bezogen auf die einzelne Lerngruppe kommen ergänzend weitere der in den Folgeabschnitten genannten Instrumente der Leistungsüberprüfung zum Einsatz.

Beurteilungsbereich Klausuren

Verbindliche Absprachen: Bei der Formulierung von Aufgaben werden die für die Abiturprüfungen geltenden Operatoren des Faches Informatik schrittweise eingeführt, erläutert und dann im Rahmen der Aufgabenstellungen für die Klausuren benutzt.

Instrumente

- Einführungsphase: 1 Klausur je Halbjahr
Dauer der Klausur: 2 Unterrichtsstunden (á 45 Minuten)
- Grundkurse Q1: 2 Klausuren je Halbjahr (anstelle einer Klausur kann gemäß dem Beschluss der Lehrerkonferenz in Q1.2 eine Facharbeit geschrieben werden)
Dauer der Klausuren: 2-3 Unterrichtsstunden
- Grundkurse Q2.1: 2 Klausuren
Dauer der Klausuren: 3 Unterrichtsstunden
- Grundkurse Q 2.2: bis zu 1 Klausur (sog. Abiturvorklausur) unter Abiturbedingungen
Dauer der Klausur: 3 Unterrichtsstunden

Die Aufgabentypen, sowie die Anforderungsbereiche I-III sind entsprechend den Vorgaben in Kapitel 3 des Kernlehrplans zu beachten.

Kriterien

Die Bewertung der schriftlichen Leistungen in Klausuren erfolgt idealerweise über ein Raster mit Hilfspunkten, die einzelnen Kriterien/ erwarteten (Teil)Lösungen zugeordnet sind (sog. Erwartungshorizont). Spätestens ab der Qualifikationsphase orientiert sich die Zuordnung der Hilfspunktsumme zu ihren Notenstufen an dem Zuordnungsschema des Zentralabiturs.

Von diesem kann aber im Einzelfall begründet abgewichen werden, wenn sich z.B. besonders originelle Teillösungen nicht durch Hilfspunkte gemäß den Kriterien des Erwartungshorizontes abbilden lassen oder eine Abwertung wegen besonders schwacher Darstellung (APO-GOST §13 (2)) angemessen erscheint.

Die Note ausreichend (5 Punkte) soll bei Erreichen von 45 % der Hilfspunkte erteilt werden.

Beurteilungsbereich sonstige Mitarbeit

Den Schülerinnen und Schülern werden die Kriterien zum Beurteilungsbereich „sonstige Mitarbeit“ zu Beginn des Schuljahres genannt.

Mündliche Leistungen

- Beteiligung am Unterrichtsgespräch
- Zusammenfassungen zur Vor- und Nachbereitung des Unterrichts
- Präsentation von Arbeitsergebnissen
- Referate
- Mitarbeit in Partner-/Gruppenarbeitsphasen

Praktische Leistungen am Computer

- Implementierung, Test, Debugging, Optimierung und Anwendung von Informatiksystemen
- Strukturierung und Kommentierung von Quellcodes
- Präsentation von Ergebnissen mit Erläuterungen des Informatiksystems (auch am Quellcode)

Sonstige schriftliche Leistungen

Die folgenden sonstigen schriftlichen Leistungen können, müssen jedoch nicht zwingend durchgeführt werden:

- Z.B. Arbeitsmappe und Arbeitstagebuch zu einem durchgeführten Unterrichtsvorhaben
- Kompendium für im Laufe der Unterrichtszeit erarbeitete informatische Begriffe und Kürzel
- Dokumentation von „Unterrichtserzeugnissen“, wie z.B. Tafelanschrieb, Quelltexte von Programmen mit Beachtung von „Code-Conventions“ (wie eine anständige Kommentierung, Syntaxeinrückung usw.)
- Lernerfolgsüberprüfung durch kurze schriftliche Übungen sind möglich
- In Kursen, in denen höchstens 50% der Kursmitglieder eine Klausur schreiben, können schriftliche Übungen mindestens einmal pro Kurshalbjahr stattfinden. Schriftliche Übungen dauern ca. 20 Minuten und umfassen den Stoff der letzten Stunden.
- Bearbeitung von schriftlichen Aufgaben im Unterricht

Kriterien

Die folgenden allgemeinen Kriterien gelten sowohl für die mündlichen als auch für die schriftlichen Formen der sonstigen Mitarbeit.

Die Bewertungskriterien stützen sich auf die Qualität der Beiträge, die Quantität der Beiträge und insbesondere auch die Kontinuität der Beiträge. Besonderes Augenmerk ist dabei 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 zu legen. Bei Gruppenarbeiten auch auf das Einbringen in die Arbeit der Gruppe, die Durchführung fachlicher Arbeitsanteile und die Qualität des entwickelten Produktes. Bei Projektarbeit darüber hinaus auf die Dokumentation des Arbeitsprozesses, den Grad der Selbstständigkeit, die Reflexion des eigenen Handelns und die Aufnahme von Beratung durch die Lehrkraft.