

---

**Prüfungsteilnehmer**

**Prüfungstermin**

**Einzelprüfungsnummer**

---

**Kennzahl:** \_\_\_\_\_

**Kennwort:** \_\_\_\_\_

**Arbeitsplatz-Nr.:** \_\_\_\_\_

**Frühjahr  
2014**

**66118**

---

**Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen  
— Prüfungsaufgaben —**

---

**Fach:** Informatik (vertieft studiert)

**Einzelprüfung:** Fachdidaktik

**Anzahl der gestellten Themen (Aufgaben):** 3

**Anzahl der Druckseiten dieser Vorlage:** 6

---

**Bitte wenden!**

## Thema Nr. 1

Im Abschnitt 12.1 (Formale Sprachen, ca. 16 Std.) des Lehrplans für Informatik am Gymnasium in Bayern erarbeiten Schülerinnen und Schüler der Jahrgangsstufe 12 Beschreibungsformen für formale Sprachen, um die Möglichkeiten der Kommunikation zwischen Mensch und Maschine besser beurteilen zu können. Dabei lernen Sie, die Begriffe Syntax und Semantik einer Sprache zu unterscheiden. Die Jugendlichen erkennen, dass jeder formalen Sprache eine Grammatik zugrunde liegt, die alle zulässigen Wörter der Sprache mit Hilfe von Ableitungsregeln beschreibt. Die Zweckmäßigkeit dieser streng formalen Beschreibung zeigt sich den Schülerinnen und Schülern bei der automatischen Überprüfung der syntaktischen Korrektheit von Zeichenketten mithilfe von endlichen Automaten.

1. Eine formale Sprache, die beispielhaft im Unterricht untersucht werden kann, ist die Sprache der (vereinfachten) email-Adressen. Wie würden Sie auf eine in diesem Rahmen gestellte Schülerfrage nach der Semantik dieser Sprache antworten? Wählen Sie eine weitere formale Sprache aus und illustrieren Sie auch an dieser Beispielsprache Ihre Überlegungen zur Semantik einer formalen Sprache anschaulich!
2. Planen Sie eine Unterrichtsstunde zu dem Grobziel „Schülerinnen und Schüler lernen, die Begriffe Syntax und Semantik einer Sprache zu unterscheiden“! Nehmen Sie ausführlich Stellung zu den Punkten:
  - Thema der vorhergehenden Unterrichtsstunde
  - Feinziele der geplanten Stunde
  - fachdidaktisches Vorgehen
  - Verlaufsplan (Phase/Zeit, Schüler- und Lehreraktivität, Sozialform/Medien)
3. Geben Sie einige Aufgaben an (einschließlich Lösungsskizze), die Schülerinnen und Schüler in einer Doppelstunde zum Thema „Formale Sprachen durch Grammatiken definieren“ bearbeiten sollen und begründen Sie kurz fachdidaktisch, warum Sie diese Aufgaben für geeignet erachten! Versuchen Sie, einige typische Fehler zu antizipieren, die Schülerinnen und Schüler bei der Bearbeitung dieser Aufgaben machen könnten. Ordnen Sie ihre Aufgaben in eine Taxonomie (beispielsweise Bloom oder Krathwohl et al.) oder ein pragmatisches Kompetenzstufenmodell (nach H. Meyer) ein und begründen Sie ihre Einordnung möglichst genau.
4. Reguläre Sprachen können mit Hilfe von endlichen Automaten erkannt werden. Erläutern Sie detailliert einen problemorientierten Einstieg in das neue Thema „Endlicher Automat als Beschreibungsform für reguläre Sprachen“. Die Jugendlichen lernen den endlichen Automaten neben (regulären) Grammatiken als eine weitere Beschreibungsform für reguläre Sprachen kennen. Dies verlangt nach einer Begründung. Warum ist das neue Thema es wert, unterrichtet zu werden? Beziehen Sie in Ihre Argumentation sowohl Sichtweisen des Informatikers / der Informatikerin als auch Sichtweisen der Lehrperson und der Lernenden selbst mit ein.
5. In der Jahrgangsstufe 12 beschäftigen sich Schülerinnen und Schüler in einem späteren Lehrplanabschnitt mit der Kommunikation von Prozessen. Beschreiben sie nötige Vorkenntnisse aus dem Lehrplanabschnitt 12.1 in Form von Feinzielen und begründen Sie jeweils kurz, wozu das Vorwissen im Laufe des Abschnitts „Kommunikation von Prozessen“ benötigt wird!

**Thema Nr. 2**

In der Jahrgangsstufe 11 beschäftigen sich die Schülerinnen und Schüler unter anderem mit rekursiven Datenstrukturen. Der Lehrplan legt für den Bereich Listen folgendes fest:

**Inf 11.1.1 Listen (ca. 29 Std.)**

Die Schüler untersuchen die grundlegenden Eigenschaften der Datenstruktur Schlange, deren grundsätzlichen Aufbau sie bereits aus ihrem Alltag, z. B. von Warteschlangen, kennen. Eine erste Implementierung mit einem Feld zeigt schnell die Grenzen dieser statischen Lösung auf und führt die Jugendlichen zu einer dynamischen Datenstruktur wie der einfach verketteten Liste. Sie erarbeiten deren prinzipielle Funktionsweise sowie deren rekursiven Aufbau und wenden hierbei das Prinzip der Referenz auf Objekte an. Die Jugendlichen erkennen, dass die rekursive Struktur der Liste für viele ihrer Methoden einen rekursiven Algorithmus nahelegt. Sie verstehen, dass eine universelle Verwendbarkeit der Klasse Liste nur möglich ist, wenn auf eine klare Trennung von Struktur und Daten geachtet wird. An einfachen Beispielen aus der Praxis und deren Implementierung vertiefen die Schüler ihr Wissen und erfahren die flexible Verwendbarkeit dieses Datentyps.

- Methoden der Datenstruktur Schlange: Anfügen am Ende, Entfernen am Anfang
- allgemeines Prinzip und rekursive Struktur einer einfach verketteten Liste; graphische Veranschaulichung der Methoden zum Einfügen (auch an beliebiger Stelle), Suchen und Löschen
- rekursive Abläufe: rekursiver Methodenaufruf, Abbruchbedingung, Aufrufsequenz
- Implementierung einer einfach verketteten Liste als Klasse mittels Referenzen unter Verwendung eines geeigneten Softwaremusters (Composite); Realisierung der Methoden zum Einfügen, Suchen und Löschen
- Einsatz der allgemeinen Datenstruktur Liste bei der Bearbeitung eines Beispiels aus der Praxis: Verwaltung von Elementen verschiedener Datentypen mittels Vererbung
- Stapel und Schlange als spezielle Formen der allgemeinen Datenstruktur Liste

**1. Klärung von Lehrplaninhalten**

- a) Der Lehrplan sieht einen Einstieg in das Thema Schlangen durch eine Implementierung mit Hilfe der Datenstruktur Feld vor.  
Zeigen Sie die Grenzen dieser Umsetzung auf und formulieren Sie Anforderungen an eine geeignetere Datenstruktur!
- b) Veranschaulichen Sie, wie die „klare Trennung von Struktur und Daten“ ausgehend von einer einfach verketteten Liste erreichbar ist.

**2. Welche Rolle spielen Modellierung und Programmierung im Bereich „Rekursive Datenstrukturen/Listen“? Begründen Sie die Notwendigkeit beider Schritte für den Unterricht!**

**Fortsetzung nächste Seite!**

### 3. Rekursive Abläufe

- a) Rekursive Abläufe sind für die Schülerinnen und Schüler in der Regel neu. Erläutern Sie ein geeignetes Einstiegsproblem ausgehend von einer einfach verketteten Liste!
- b) Welche grundsätzlichen Unterschiede bestehen zwischen den rekursiven Abläufen aus Ihrem vorherigen Beispiel und denen bei der rekursiven Berechnung der Fakultätsfunktion bzw. des größten gemeinsamen Teilers zweier natürlicher Zahlen?
- c) Begründen Sie mit didaktischen Argumenten, ob Sie in rekursive Abläufe direkt mit Ihrem Einstiegsproblem aus Teilaufgabe a) oder mit der Fakultätsfunktion bzw. dem größten gemeinsamen Teiler einsteigen!
- d) Welche Möglichkeiten haben Sie im Unterricht, die rekursiven Abläufe bei einfach verketteten Listen zu visualisieren?
- e) Erstellen Sie die Feinplanung für eine Unterrichtseinheit (mindestens eine Doppelstunde) zum Einstieg in rekursive Abläufe im Themenkomplex Liste! Formulieren Sie dazu zunächst die entsprechenden Grob- und Feinziele! Stellen Sie danach detailliert und nachvollziehbar den Unterrichtsfortgang dar! Gehen Sie dabei auf die geplante Lehrer-Schüler-Interaktion, Methoden, Sozialformen und eingesetzte Medien ein!

### Thema Nr. 3

#### Softwareentwicklung und Projektmanagement

Im Rahmen des Informatikunterrichtes wollen Sie mit Ihren Schülerinnen und Schülern ein Softwareprojekt durchführen. Die Schülerinnen und Schüler haben sich entschieden, im Rahmen dieses Projektes eine Software zur **Verwaltung eines Sportvereins** zu erstellen.

Die Softwareentwicklung im Rahmen dieses Unterrichtsprojektes kann man aus drei verschiedenen Sichten betrachten:

- S1: Die Sichtweise eines professionellen Softwareentwicklers, der von einem Sportverein den Auftrag zur Entwicklung dieses Programmpakets erhält und damit Geld verdienen will. Diese Sicht werden Sie Ihren Schülerinnen und Schülern immer wieder zur Motivierung sowie als Bezugsrahmen für deren Arbeit vorstellen.
- S2: Die Sichtweise der Schülerinnen und Schüler, die im Rahmen ihres Unterrichts als Lernende nach Lehrplanvorgabe zu Ausbildungszwecken Software entwickeln, die naturgemäß in Qualität und Umfang nicht die Anforderungen einer professionellen Ausführung erfüllen kann.
- S3: Ihre didaktische und pädagogische Sicht als Lehrkraft auf den Verlauf des projektartigen Unterrichts, aus der heraus Sie vor allem auf optimalen Lernerfolg Ihrer Schülerinnen und Schüler achten.

#### 1. Unterrichtliche Voraussetzungen

Geben Sie im Sinne des Berliner Modells der allgemeinen Didaktik möglichst präzise und umfassend aus der Sicht S3 die (hypothetischen) Voraussetzungen für Ihr Unterrichtsprojekt an! Gehen Sie dabei insbesondere ein auf: Lehrplanbezug, Alter, Vorkenntnisse, Geschlechtsverteilung, Leistungsstand Ihrer Schülerinnen und Schüler sowie auf den organisatorischen und technischen Rahmen des Projektes!

#### 2. Vorgehensmodell

Als Reaktion auf die damals oft sehr schlechte Qualität industrieller Software entstand 1970 ein erstes systematisches Vorgehensmodell zur Softwareentwicklung, das unter dem Namen „Wasserfallmodell“ bekannt wurde und auch heute noch große Bedeutung hat. Im Laufe der Zeit haben sich viele verschiedene Varianten dieses Modells gebildet. Heute verwendet man meist eine fünfstufige Variante mit den folgenden Phasen: (1) Analyse, (2) Entwurf, (3) Implementierung, (4) Test mit Integration, (5) Einsatz mit Wartung.

- a) Beschreiben Sie aus der Sicht S1 kurz die Zielsetzung und die idealisierten Ergebnisse jeder dieser fünf Phasen sowohl im Allgemeinen als auch speziell auf Ihr Projekt bezogen! Achten Sie bei den Ergebnissen Ihres Projektes darauf, diese möglichst konkret anzugeben!
- b) Geben Sie an, welche Einschränkungen Sie im Ergebnis der jeweiligen Phase aus der Sicht S2 erwarten!
- c) Geben Sie an, welchen Verlauf und welche Ergebnisse Sie für jede dieser fünf Phasen, sowohl allgemein als auch speziell auf Ihr Projekt bezogen, erwarten würden!

**Fortsetzung nächste Seite!**

### 3. Artikulationsschema von Roth

Anfang der 60er Jahre entwickelte Heinrich Roth sein berühmtes Artikulationsschema, das folgende sechs Phasen (er nannte sie „Stufen“) des Unterrichtsgangs identifizierte: (1) Motivation, (2) Schwierigkeiten, (3) Lösung, (4) Tun und Ausführen, (5) Behalten und Einüben, (6) Bereitstellen, Übertragung und Integration.

- a) Geben Sie für jede dieser „Stufen“ eine beispielhafte Unterrichtsaktivität aus dem Informatik-Unterricht an! Beschreiben Sie dabei genau, was die Schülerinnen und Schüler tun und was die Lehrkraft dabei tut!
- b) Beschreiben Sie aus der Sicht S3, wie Sie die o. g. Phasen Ihres Unterrichtsprojektes (wie sie sich aus dem „Wasserfallmodell“ ergeben, siehe Aufgabe 2), mit den sechs „Stufen“ nach Roth verzahnen würden! Begründen Sie Ihre Antwort jeweils mit pädagogischen bzw. didaktischen Argumenten!