

Lukas Gerner

Fachinformatiker Fachinformatikerin

Fachrichtung Anwendungsentwicklung

Planen eines Softwareproduktes Entwicklung und Umsetzung von Algorithmen

Prüfungstrainer Abschlussprüfung Teil 2
Übungsaufgaben und erläuterte Lösungen

Aufgabenteil

Bestell-Nr. 2759

u-form Verlag · Hermann Ullrich GmbH & Co. KG

Deine Meinung ist uns wichtig!

Du hast Fragen, Anregungen oder Kritik zu diesem Produkt?

Das u-form Team steht dir gerne Rede und Antwort.

Einfach eine kurze E-Mail an:

feedback@u-form.de

Änderungen, Korrekturen und Zusatzinfos

findest du übrigens unter diesem Link:

www.u-form.de/addons/2759-2025.zip

Wenn der Link nicht funktioniert,
haben wir noch keine Zusatzinfos hinterlegt.



BITTE BEACHTEN:

Zu diesem Prüfungstrainer gehört auch noch ein **Lösungsteil**.



1. Auflage 2025 · ISBN 978-3-95532-759-0

Alle Rechte liegen beim Verlag bzw. sind der
Verwertungsgesellschaft Wort, Untere Weidenstr. 5, 81543 München,
Telefon 089 514120, zur treuhänderischen Wahrnehmung überlassen.
Damit ist jegliche Verbreitung und Vervielfältigung dieses Werkes –
durch welches Medium auch immer – untersagt.



© u-form Verlag | Hermann Ullrich GmbH & Co. KG
Cronenberger Straße 58 | 42651 Solingen
Telefon: 0212 22207-0 | Telefax: 0212 22207-63
Internet: www.u-form.de | E-Mail: uform@u-form.de

Hi!

cool, dass du dich für diesen Prüfungstrainer entschieden hast und dich nun auf die schriftliche Abschlussprüfung Teil 2 für Fachinformatiker der Fachrichtung Anwendungsentwicklung vorbereiten willst.

Mithilfe dieses Buches wirst du, über sieben Prüfungen verteilt sowie mithilfe zusätzlicher Übungen, alle relevanten Inhalte der Prüfungsverordnung lernen und praktisch anwenden. Wir beziehen uns hierbei vollständig auf die neuste gültige Ausbildungsordnung (AO 2020).

Wichtig zu wissen: Alle Prüfungskataloge für IT-Berufe (unter anderem auch für Fachinformatiker Anwendungsentwicklung) wurden umfassend überarbeitet und 2025 in einer neuen Fassung veröffentlicht. Einige zuvor relevanten Themen wurden gestrichen und sind somit nicht mehr Bestandteil der Abschlussprüfung Teil 2. Mit der Aktualisierung wurden aber auch zahlreiche neue Themen zum Katalog hinzugefügt.

Dieser Prüfungstrainer berücksichtigt bereits alle neuesten Änderungen des offiziellen Prüfungskataloges (gültig ab 01.01.2025) und ist daher bestens zur Prüfungsvorbereitung ab Sommer 2025 geeignet.

Alle Aufgaben dieses Prüfungstrainers könnten genau so auch in deiner tatsächlichen Abschlussprüfung Teil 2 gestellt werden.

Jede Aufgabe wird im beiliegenden Lösungsbuch mithilfe von Lösungsvorschlägen erklärt und ggf. über Schaubilder ausführlich dargestellt. Oft sind hierbei auch mehrere Lösungen möglich. Wir bitten zu beachten, dass in der Lösungserläuterung nicht immer alle möglichen Lösungen vollständig berücksichtigt werden können.

Um alle Themen des Prüfungskataloges abzudecken, empfehlen wir dir die vollständige Bearbeitung dieses Buches. In deiner anstehenden IHK-Prüfung werden nicht alle der theoretisch möglichen Themengebiete behandelt. Dennoch sind alle Themen für deine weitere berufliche Laufbahn und auch für deine betriebliche Projektarbeit sowie dein Fachgespräch relevant.

Wir wünschen dir viel Erfolg bei der Bearbeitung der Prüfungen und hoffen, dass du durch unser Buch bestmögliche Ergebnisse in deiner anstehenden Abschlussprüfung erzielen wirst!



Lukas Gerner

Mein besonderer Dank gilt Alexander Lorenz, der das fachliche Lektorat dieses Prüfungstrainers übernommen hat. Deine Unterstützung war großartig – danke dir, Alex!

Aufbau der IHK-Abschlussprüfungen im Überblick

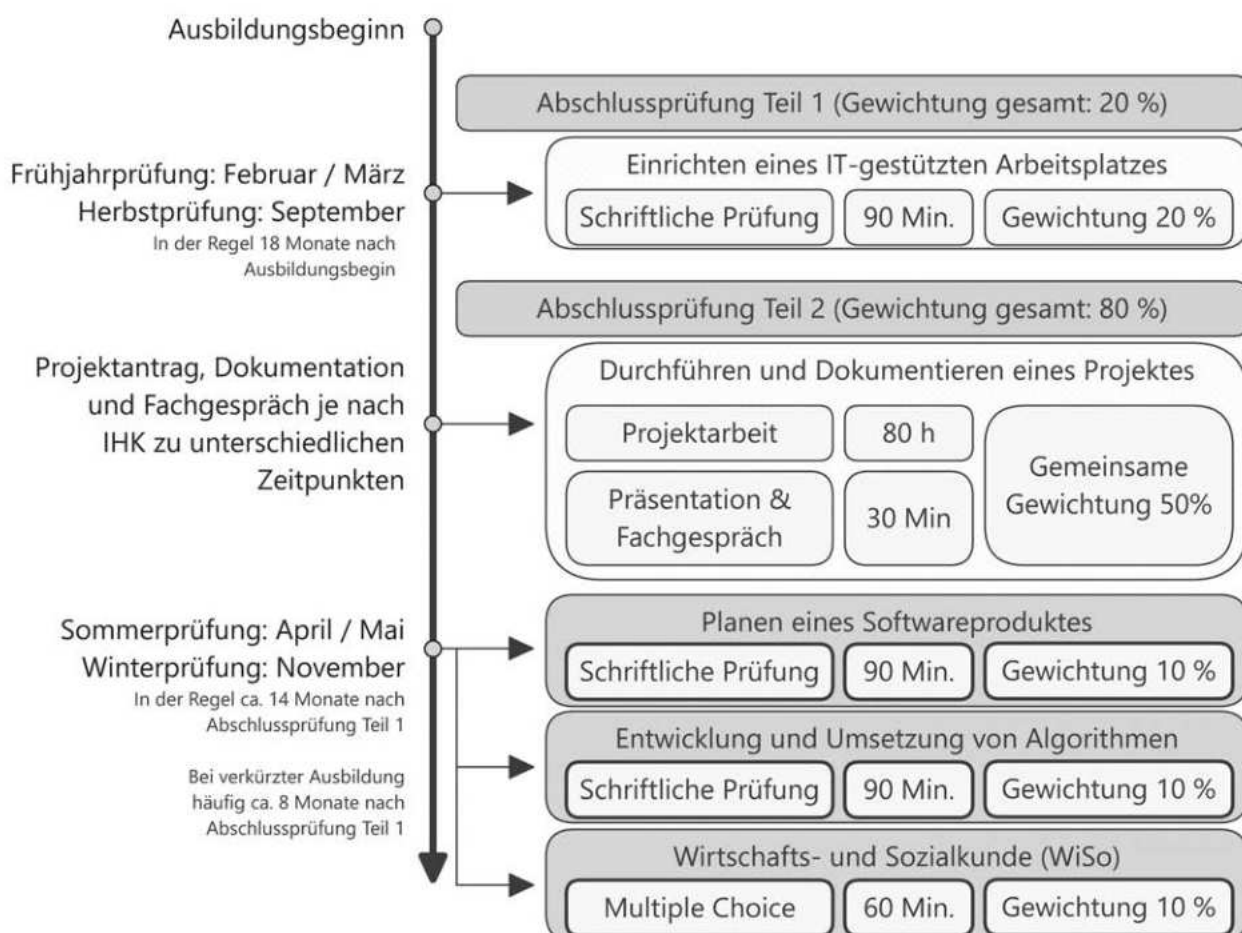
Bei deiner Abschlussprüfung handelt es sich um eine gestreckte IHK-Abschlussprüfung mit zwei großen Prüfungsteilen:

Gestreckte Abschlussprüfung Teil 1 (GAP1):

- „Einrichten eines IT-gestützten Arbeitsplatzes“

Gestreckte Abschlussprüfung Teil 2 (GAP2):

- Betriebliche Projektarbeit inkl. Projektdokumentation, Projektpräsentation sowie einem anschließenden Fachgespräch vor einem Prüfungsausschuss
- Berufsbezogene schriftliche Prüfung 1 („Planen eines Softwareproduktes“)
- Berufsbezogene schriftliche Prüfung 2 („Entwicklung und Umsetzung von Algorithmen“)
- Schriftliche Prüfung in Wirtschafts- und Sozialkunde (WiSo) im Multiple-Choice-Format



Dieses Buch bereitet dich auf die schriftlichen Prüfungen der gestreckten Abschlussprüfung Teil 2 (GAP2) vor. Die schriftlichen Prüfungen „Planen eines Softwareproduktes“, „Entwicklung und Umsetzung von Algorithmen“ sowie „Wirtschafts- und Sozialkunde (WiSo)“ werden an einem gemeinsamen Prüfungstermin nacheinander absolviert. Gemeinsam sind diese Prüfungen mit 30 % für die Berechnung der Abschlussnote zu gewichten.

Vorwort	3
Aufbau der IHK-Abschlussprüfungen	4
Einführung: Themenvorstellung und Prüfungstipps	6
Prüfungen	
Prüfung 1: Beispielbeschreibung	15
Prüfung 2: Beispielbeschreibung	37
Prüfung 3: Beispielbeschreibung	61
Prüfung 4: Beispielbeschreibung	85
Prüfung 5: Beispielbeschreibung	109
Prüfung 6: Beispielbeschreibung	133
Prüfung 7: Beispielbeschreibung	153
Zusätzliche Aufgaben	
Zusätzliche Übungen	175
Prüfungsvorbereitung Wirtschafts- und Sozialkunde (WiSo)	183
Anhang	
Übersicht Themenkomplexe gemäß Prüfungskatalog	193

Ist deine Lieferung vollständig?

Dieser Prüfungstrainer besteht aus zwei Büchern: einem Aufgabenbuch und einem Lösungsbuch. Beide Bücher gehören zusammen und werden ausschließlich als Set verkauft. Hast du nur eines der Bücher erhalten, liegt ein Fehler vor. Bitte melde in diesem Fall deine Falschlieferung:

- **Bei Bestellung über einen Dritthändler (z. B. Amazon, Thalia):** Reklamiere die Lieferung direkt beim Händler.
- **Bei Bestellung direkt bei u-form:** Teile uns deine Bestellnummer, Auftragsnummer oder Kundennummer mit und schreibe uns eine E-Mail an feedback@u-form.de. Wir helfen dir schnellstmöglich weiter.



Prüfung 1

Planen eines Softwareproduktes

4 Aufgaben

90 Minuten

100 Punkte

Entwicklung und Umsetzung von Algorithmen

4 Aufgaben

90 Minuten

100 Punkte

Viel Erfolg!

Die Aufgaben der beiden nachfolgenden Prüfungsbereiche beziehen sich auf die folgende Ausgangssituation:



Zur Abwicklung des kontinuierlich wachsenden Güterverkehrs ist die Errichtung eines neuen Güterbahnhofs geplant. Ihr Team erhält den Auftrag, für dessen Organisation diverse Software-Pakete zu entwickeln.

Für diese Prüfungssimulation dürfen folgende Inhalte des Belegsatzes genutzt werden:

- SQL-Syntax (Seite 10 im Lösungsbuch)
- UML-Sequenzdiagramm (Seite 16 im Lösungsbuch)
- UML-Klassendiagramm (Seite 15 im Lösungsbuch)

Alle übrigen Inhalte des Belegsatzes sind zur Bearbeitung dieser Prüfungssimulation nicht zugelassen.

Prüfungsbereich 1: Planen eines Software-Produktes

1. Aufgabe (29 Punkte)

- a) Für jedes der zu entwickelnden Software-Pakete wird ein Pflichtenheft erstellt. Die darin enthaltenen Anforderungsbeschreibungen müssen den folgenden drei Kriterien genügen: Vollständigkeit, Eindeutigkeit, Testbarkeit

- aa) Beschreiben Sie, welche Konsequenzen drohen, wenn die Kriterien Eindeutigkeit und Testbarkeit nicht beachtet werden.

(4 Punkte)

Konsequenzen bei Nichtbeachtung der Eindeutigkeit:

Konsequenzen bei Nichtbeachtung der Testbarkeit:

Prüfung 1

- ab) Beurteilen Sie die folgende Anforderung anhand der drei Kriterien: (6 Punkte)

Die Länge eines Güterzugs beträgt bei Einsatz einer einzelnen Lokomotive maximal 30 Waggons. Aus ökonomischen Gründen darf er aber auch nicht zu kurz sein.

Kriterien	Bewertung mit kurzer Begründung
Vollständigkeit	
Eindeutigkeit	
Testbarkeit	

- b) Die zu erstellende Software-Lösung soll mithilfe einer passenden Programmiersprache erstellt und anschließend effizient und wartbar gestaltet werden.

- ba) Nennen Sie drei allgemeingültige Kriterien für die Auswahl einer Programmiersprache. (3 Punkte)

- bb) Nennen Sie zwei exemplarische Möglichkeiten, ein Programm effizient zu gestalten. (2 Punkte)

- c) Bei der Verwaltung des Zugaufkommens fallen große Datenmengen an.

- ca) Ihr Team diskutiert darüber, ob für die Sortierung der Daten ein quadratisches oder ein logarithmisches Verfahren zu bevorzugen ist. Nennen Sie jeweils einen Vorteil, einen Nachteil und geben Sie einen beispielhaften Sortieralgorithmus für jedes der beiden Verfahren an. (6 Punkte)

	Quadratische Sortierverfahren	Logarithmische Sortierverfahren
Vorteil		
Nachteil		
Beispiel		

- cb) Sie möchten den Bubble Sort Algorithmus innerhalb eines Projektes anwenden und verwenden hierfür eine Vergleichsfunktion.
- Es existiert bereits eine Klasse *Gueterzug*, die Informationen zu einem Zug bereitstellt. Ein Array *Zuege* enthält mehrere Objekte vom Typ *Gueterzug*.
 - Die Vergleichsfunktion wird als Function-Objekt übergeben und kann im Sortieralgorithmus wie folgt aufgerufen werden: *vergleiche(ZugA, ZugB)*.
 - Die Funktion liefert einen Wert größer als 0, wenn *ZugB* vor *ZugA* einsortiert werden soll (z. B. nach Länge, Gewicht oder Priorität).

Implementieren Sie den Bubble Sort Algorithmus im Pseudocode unter Verwendung der Vergleichsfunktion.

(8 Punkte)

```
// Bubble Sort für Güterzüge mit einer Vergleichsfunktion
sort (Gueterzug[] Zuege, Function vergleiche) : void
```

Prüfungsbereich 2:

Entwicklung und Umsetzung von Algorithmen

1. Aufgabe (25 Punkte)

Die Waggon disposition beauftragt Sie mit der Erstellung einer Anwendung, die die abzufertigenden Waggon zu Zügen zusammenstellt.

Für jeden Waggon existieren folgende Angaben:

```
1 class Waggon
2 {public String zielbahnhof;
3   public int gefahrgutklasse;
4           // 0 = kein Gefahrgut
5           // 1 = mittlere Gefahrenstufe,
6           // 2 = höchste Gefahrenstufe
7   public int prioritaet;
8           // 0 = niedrigste Priorität
9           // 2 = höchste Priorität
10 }
```

Geben Sie in Pseudocode den Algorithmus für folgende Funktion an, die aus einer Liste der abzufertigenden Waggon je nach Zielbahnhof und Gefahrgutklasse einen Zug, also eine neue Liste von Waggon zusammenstellt:

```
11 public ArrayList<Waggon> zugreihung (ArrayList<Waggon> waggon,
12   String zielbahnhof, int gefahrgutklasse, int maxAnzahl)
13 {
14   ArrayList<Waggon> zug = new ArrayList<Waggon>();
15   ...
16   Rueckgabe zug;
17 }
```

Prüfung 1

Die Funktion soll folgendermaßen arbeiten:

- Zunächst werden bis zu *maxAnzahl* Waggons der als Parameter angegebenen Gefahrgutklasse mit dem angegebenen Zielbahnhof ausgewählt, d. h. aus der Liste „waggons“ gelöscht und an die Ergebnisliste angehängt.
- Sollte danach die maximale Anzahl von Waggons noch nicht erreicht sein, wird der Zug mit Waggons niedriger Gefahrgutklassen und dem korrekten Zielbahnhof aufgefüllt. Da Gefahrgutzüge langsamer sind, dürfen diese Waggons allerdings keine zu hohe Priorität haben. Es gilt:
 - In einen Zug der Gefahrgutklasse 2 dürfen auch Waggons der Klasse 1 mit Priorität 0 oder 1 und Waggons der Klasse 0 mit Priorität 0.
 - In einen Zug der Gefahrgutklasse 1 dürfen auch Waggons der Klasse 0 mit Priorität 0.
- Stehen nicht genug geeignete Waggons zur Verfügung, fährt der Zug mit weniger als *maxAnzahl* Waggons.

Für die Verarbeitung der ArrayLists stehen Ihnen folgende Methoden zur Verfügung:

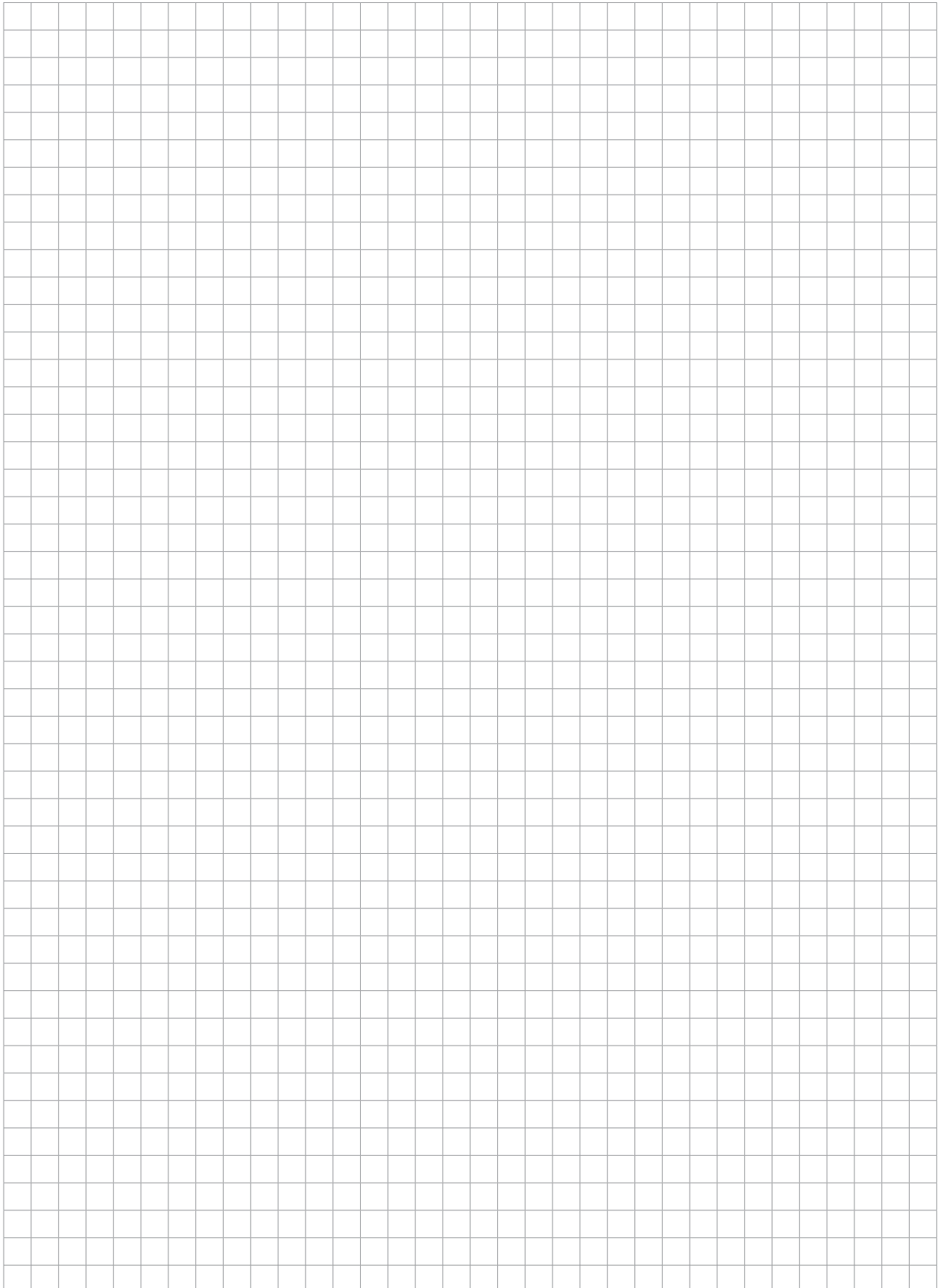
get(i)	Liefert die Referenz auf das Objekt an Position i
loeschen(i)	Löscht ein Objekt an der Position i aus der Liste, so dass alle nachfolgenden Elemente um eine Position aufrücken
hinzufuegen(Waggon)	Hängt ein neues Objekt (d. h. dessen Referenz) an eine Liste an
laenge()	Gibt die aktuelle Länge einer Liste an

Beispiel:

Indexposition	.zielbahnhof	.gefahrenklasse	.prioritaet
0	Dortmund	0	2
1	Kassel	1	0
2	Dresden	1	1
3	Dresden	2	1
4	Dresden	0	2
5	Dortmund	1	2
6	Dresden	1	0

Aus dieser Liste abzufertigender Waggons würde ein Zug nach Dresden mit der Gefahrgutklasse 2 zuerst aus Waggon 3 und dann aus den Waggons 2 und 6 bestehen. Waggon 4 hat eine zu hohe Priorität, die übrigen haben andere Ziele.

Ein Zug nach Dortmund der Gefahrgutklasse 0 würde nur aus Waggon 0 bestehen, weil Waggon 5 eine höhere Gefahrgutklasse hat.



2. Aufgabe (25 Punkte)

Nicht immer können Züge die geplante Route nehmen, weil gelegentlich Strecken gesperrt werden müssen (z. B. wegen Bauarbeiten, Unfällen, umgestürzter Bäume usw.).

Für das besonders dichte und stark befahrene Streckennetz im Ruhrgebiet existiert dazu die folgende Verbindungsmatrix, die angibt, ob eine bestimmte Strecke befahrbar (1), zurzeit nicht befahrbar (-1) oder nicht als Direktverbindung existiert (0) ist:

		0	1	2	3	4
		Duisburg	Oberhausen	Mülheim	Essen	Dortmund
0	Duisburg	0	1	1	0	0
1	Oberhausen	1	0	1	1	1
2	Mülheim	1	1	0	1	0
3	Essen	0	1	-1	0	1
4	Dortmund	0	1	0	1	0

Beispielsweise bezeichnet *verbindung[0][1]* die Strecke von Duisburg nach Oberhausen mit dem Wert 1.

- a) Verfassen Sie eine Funktion als Pseudocode, die zählt, wie viele Strecken derzeit gesperrt sind.

(7 Punkte)

[illegible]

- b) Die folgende Funktion soll eine Verbindung zwischen zwei Städten suchen, wobei die Städte durch ihre Spaltennummern dargestellt werden:

```
18 String suche(int start, int ziel) {
19     for (int i=0; i<5; i++) {
20         if (verbindung[start][i]==1) {
21             if (i==ziel) {
22                 return start + " " + ziel; }
23             else {
24                 return start + " " + suche(i,ziel); }
25         }
26     }
27 }
```

- ba)** Testen Sie diese Funktion anhand der folgenden Parameterwerte und geben Sie das jeweilige Ergebnis an.

(6 Punkte)

Start	Ziel	Ergebnis
0	1	
3	0	
0	3	

- bb)** Erläutern Sie den Fehler, der beim Entwurf dieser Funktion gemacht wurde.

(3 Punkte)

- bc)** Erläutern Sie wie die Funktion angepasst werden könnte, um den Fehler zu vermeiden.

(3 Punkte)

- bd)** Um Funktionen zukünftig ausführlicher zu testen, sollen Testdatengeneratoren eingesetzt werden.

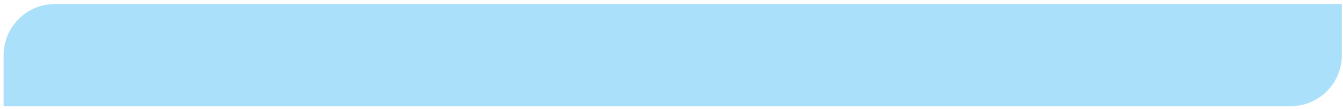
Beschreiben Sie eine Methode die Testdatengeneratoren nutzen, um eine umfassende Testabdeckung zu gewährleisten.

(2 Punkte)

- be)** Nach Ihren Tests soll ein Testprotokoll angefertigt werden.

Nennen Sie vier allgemeine Bestandteile eines Testprotokolls.

(4 Punkte)



Zusätzliche Übungen

Die statistischen Auswertungen zu den am häufigsten geprüften Themen (siehe Seite 11) sind auch den Erstellern von Prüfungsaufgaben bekannt. Umso wichtiger ist es, sich ebenfalls mit Themen auseinanderzusetzen, die bislang nur selten oder noch nicht in den Abschlussprüfungen behandelt wurden. Gerade diese Themen könnten in zukünftigen Prüfungen stärker abgefragt werden.



Einige selten abgefragte Themen sind auf die verschiedenen Prüfungssimulationen dieses Prüfungstrainers verteilt. Ergänzend findest du nachfolgend zusätzliche Übungen zu diesen Themen sowie zu ausgewählten bislang noch nicht abgefragten Themen des Prüfungskataloges. Die zusätzlichen Übungen sowie der gesamte Prüfungstrainer werden jährlich geprüft und aktualisiert.

Viel Erfolg!

1. Umfeldanalyse (10 Punkte)

Sie arbeiten in einem mittelständischen Maschinenbauunternehmen mit ca. 120 Mitarbeitenden. Zur Optimierung der betrieblichen Abläufe soll ein ERP-System eingeführt werden, das insbesondere die Bereiche Lager, Einkauf, Fertigung und Vertrieb miteinander verknüpft.

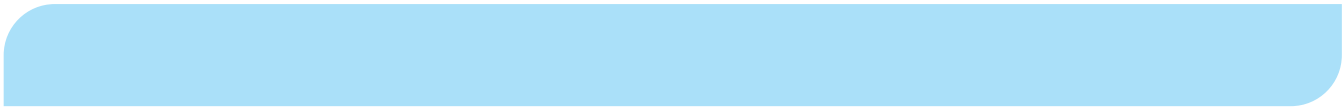
- a) Nennen Sie drei allgemeingültige Bereiche, die bei einer Umfeldanalyse zu berücksichtigen sind.

(4 Punkte)

- b) Erläutern Sie jeweils einen Aspekt zu zwei der zuvor genannten Bereiche einer Umfeldanalyse.

(6 Punkte)

Bereich der Umfeldanalyse	Erläuterung eines Aspektes



WiSo

Wirtschafts- und Sozialkunde

Prüfungsvorbereitung WiSo

Im Prüfungsbereich Wirtschafts- und Sozialkunde (WiSo) werden für jede Prüfung 30 Multiple-Choice-Fragen aus einem Aufgabenpool ausgewählt und zu einer einheitlichen Prüfung zusammengestellt.

Die meisten Fragen lassen sich mit grundlegenden Kenntnissen gut beantworten, und häufig kann die richtige Lösung durch das Ausschlussverfahren ermittelt werden.

Dennoch gibt es immer wieder besonders anspruchsvolle Fragen, die bei Prüfungsteilnehmenden zu großer Unsicherheit führen.

Auf den folgenden Seiten findest du Aufgaben, die sich an den schwierigsten Aufgaben der vergangenen Prüfungen orientieren.

In einer typischen WiSo-Prüfung gibt es in der Regel zwei bis fünf solcher besonders schwierigen Aufgaben. Auf den nachfolgenden Seiten findest du 15 Aufgaben, also so viele schwierige Aufgaben, wie sonst in mehreren Prüfungen verteilt.

Bei deiner tatsächlichen Abschlussprüfung müssen alle Ergebnisse zum Prüfungsteil WiSo in ein vorgefertigtes Formular eingetragen werden.

Für die nachfolgenden Übungen wollen wir dir das Übertragen deiner Lösungen in ein separates Formular ersparen und haben die Antwortfelder daher direkt in den Aufgabenstellungen berücksichtigt.

Viel Erfolg und gutes Gelingen für die nachfolgenden Aufgaben!

Beispielhafter Lösungsbogen einer IHK-Abschlussprüfung im Prüfungsbereich WiSo



Details zur öffentlich zugänglichen Prüfungsstatistik der vergangenen IHK-Abschlussprüfungen findest du unter pes.ihk.de

Die folgenden Aufgaben beziehen sich auf die folgende Ausgangssituation:

Die Trivanta GmbH ist ein IT-Service-Unternehmen und hat 45 Mitarbeiter. Die folgenden Aufgaben beziehen sich auf dieses Unternehmen.



1. Aufgabe

Sie beschäftigen sich mit dem Aufgabenbereich des Betriebsrates und lesen hierbei den folgenden Satz „der Betriebsrat sorgt in Unternehmen für die Umsetzung und Einhaltung von Tarifverträgen.“ Bisher hatten Sie nur Gewerkschaften mit Tarifverträgen in Verbindung gebracht.

Welche der folgenden Darstellungen von Gewerkschaften ist zutreffend?

Kreuzen Sie die zutreffende Darstellung an.

Gewerkschaften ...

- ☐ haben bei Tarifverhandlungen für die Belange aller Arbeitnehmer einzutreten.
- ☐ sind in den Betrieben für die Schlichtung von Auseinandersetzungen zwischen Arbeitnehmern und Arbeitgebern zuständig.
- ☐ haben die Aufgabe, die Betriebsratswahlen zu kontrollieren.
- ☐ können bei Streitigkeiten zwischen Arbeitgebern und Arbeitnehmern den organisierten Arbeitnehmern Rechtsbeistand leisten.
- ☐ handeln mit den Arbeitgeberverbänden auch Betriebsvereinbarungen aus.

2. Aufgabe

Innerhalb Ihres Unternehmens gibt es eine Jugend- und Auszubildendenvertretung (JAV), in der Sie sich engagieren möchten.

Welche der folgenden Sachverhalte über die JAV ist zutreffend?

Kreuzen Sie die zutreffende Darstellung an.

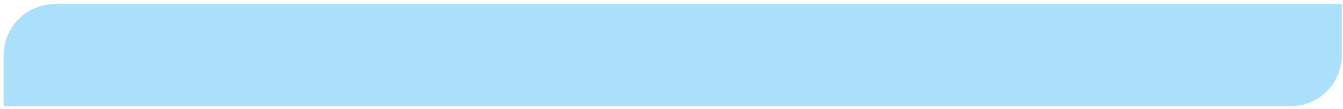
- ☐ Die regelmäßige Amtszeit der JAV beträgt vier Jahre.
- ☐ Eine JAV kann nur gewählt werden, wenn bereits ein Betriebsrat besteht.
- ☐ Das aktive Wahlrecht für die JAV endet immer mit Vollendung des 18. Lebensjahres.
- ☐ Das passive Wahlrecht für die JAV endet mit Vollendung des 18. Lebensjahres.
- ☐ Das Mindestalter für das aktive Wahlrecht beträgt 16 Jahre.

3. Aufgabe

Ihre Kollegin Frau Müller bringt nach einer Schwangerschaft ein gesundes Kind zur Welt.

Wie viele Wochen Mutterschutz stehen Ihrer Kollegin Frau Müller nach der Entbindung laut Mutterschutzgesetz zu?

Tragen Sie den zutreffenden Zeitraum in Wochen ein: _____



Lukas Gerner

Fachinformatiker Fachinformatikerin

Fachrichtung Anwendungsentwicklung

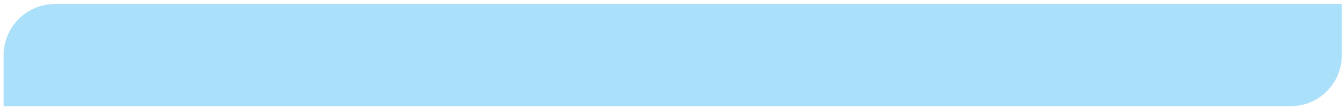
Planen eines Softwareproduktes Entwicklung und Umsetzung von Algorithmen

Prüfungstrainer Abschlussprüfung Teil 2
Übungsaufgaben und erläuterte Lösungen

Lösungsteil

Bestell-Nr. 2759

u-form Verlag · Hermann Ullrich GmbH & Co. KG



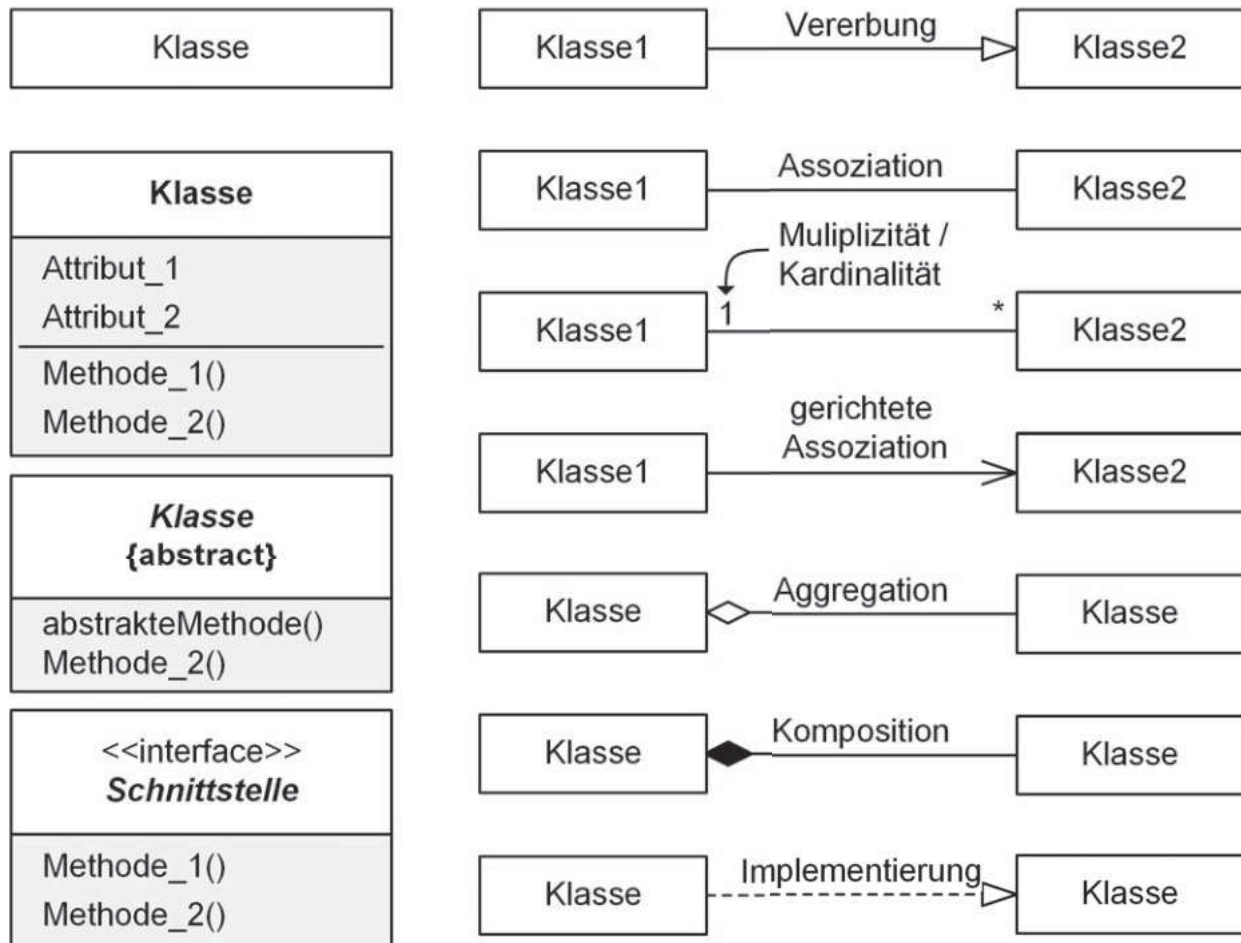
Belegsatz zur Abschlussprüfung

SQL-Syntax (Begrenzte Auswahl)

Syntax	Beschreibung
<i>Befehle, Klauseln, Attribute</i>	
SELECT <i>spaltenname1, ...</i> FROM <i>tabellenname</i>	Ausgabe von Inhalten, der angegebenen Tabellen und Spalten
SELECT * FROM <i>tabellenname</i>	Ausgabe aller Spalten der angegebenen Tabellen
SELECT ... FROM ... (SELECT ... FROM ... WHERE ...) WHERE ...	Unterabfrage / subquery, Ergebnis der Unterabfrage wird wie eine Tabelle behandelt
SELECT <i>spaltenname AS sp</i> FROM <i>tabellenname AS tbl</i>	Alias, um einen temporären Namen für Tabellen oder Spalten zu vergeben
SELECT DISTINCT	Ergebnisabgabe ohne doppelte Zeilen
JOIN / INNER JOIN ... ON ...	Datensätze aus zwei Tabellen mit übereinstimmenden Daten
LEFT JOIN / LEFT OUTER JOIN ... ON ...	Alle Daten aus der linken Tabelle (erste Angabe) und alle übereinstimmenden Daten von der zweiten Tabelle
WHERE	Filterung von Einträgen
WHERE EXISTS	EXISTS prüft, ob eine Unterabfrage Daten liefert. EXISTS liefert TRUE, wenn einer oder mehrere Datensätze geliefert werden.
WHERE NOT EXISTS	NOT EXISTS negiert die Bedingung.
WHERE ... IN (unterabfrage/liste)	Angabe mehrerer Werte oder einer Unterabfrage innerhalb einer WHERE-Klausel
WHERE NOT ... IN (unterabfrage/ liste)	Negierte Angabe mehrerer Werte oder einer Unterabfrage innerhalb einer WHERE-Klausel
GROUP BY <i>spaltenname1 ...</i> HAVING <i>Bedingung</i>	Gruppierung von Zeilen mit selben Werten innerhalb einer Summen-Zeile WHERE kann bei GROUP BY nicht verwendet werden, hier ist HAVING zu verwenden
ORDER BY <i>spaltenname1 ...</i> ASC DESC	Sortierung nach Inhalt der genannten Spalten ASC: aufsteigend DESC: absteigend

→ Fortsetzung auf nächster Seite

UML-Klassendiagramm



Syntax für Attribute:

Sichtbarkeit Attributname : Typ {Eigenschaften}

Syntax für Methoden:

Sichtbarkeit Methodenname (parameter1 : Typ, ...) :
Rückgabotyp {Eigenschaften}

Sichtbarkeiten:

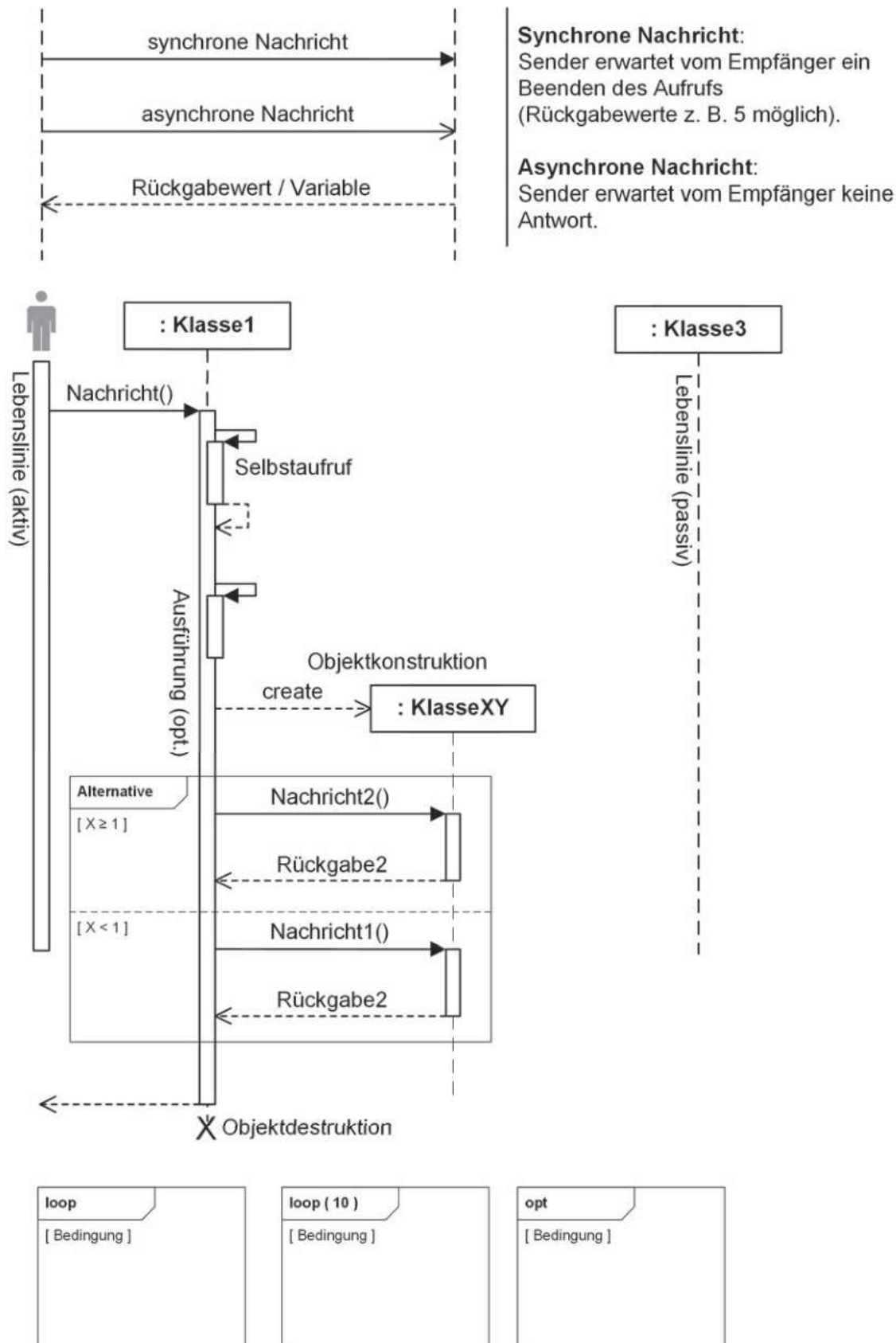
+ public
protected
- private
~ package

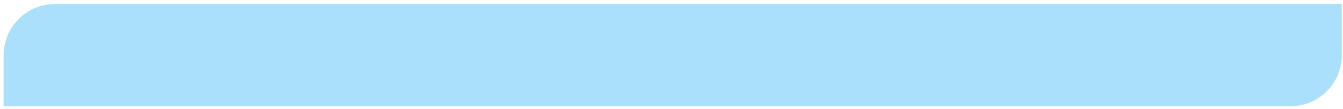
Eigenschaften:

{static, final, ...}

Notiz

UML-Sequenzdiagramm





Lösung Prüfung 1

Prüfungsbereich 1: Planen eines Software-Produktes

1. Aufgabe (29 Punkte)

a)

aa) Konsequenzen bei Nichtbeachtung der Eindeutigkeit:

Durch uneindeutige Anforderungen können **Missverständnisse entstehen**. Hierdurch entsteht ein höherer Kommunikationsbedarf.

Konsequenzen bei Nichtbeachtung der Testbarkeit:

Wenn Anforderungen nicht überprüft werden können, wird die **Qualitätssicherung erschwert**. Die Qualität der Softwarepakete wird sich voraussichtlich verschlechtern.

☑ Ähnliche Lösungen sind möglich.

☑ Zwei Punkte je korrekter Beschreibung, für teilweise korrekte Beschreibungen wäre ein Punkt zu vergeben.

Weiterführende Informationen

Konsequenzen bei Nichtbeachtung der Vollständigkeit: Wenn wichtige Anforderungen fehlen, können Fehlfunktionen oder unzureichende Implementierungen entstehen. Das kann zu zusätzlichen Kosten oder Verzögerungen führen.

ab)

Kriterien	Bewertung mit kurzer Begründung
Vollständigkeit	Nicht vollständig Unklar, ob auch mehrere Lokomotiven eingesetzt werden können
Eindeutigkeit	Uneindeutige Aussage „nicht zu kurz“
Testbarkeit	Testbarkeit ist gegeben Maximale Länge einer Lokomotive ist angegeben

☑ Andere Bewertungen sind je nach Begründung möglich.

☑ Zwei Punkte je korrekter Begründung.

Prüfung 1

b)

ba) Allgemeingültige Kriterien zur Auswahl einer Programmiersprache:

- Performance
- Speicherverbrauch
- Portabilität
- Verfügbare Frameworks / Bibliotheken
- Bereits vorhandenes Know-how / Fachwissen
- Skalierbarkeit
- Einsatz von integrierten Entwicklungsumgebungen
- ...

✓ Ein Punkt je korrekter Nennung.

bb) Möglichkeiten, um ein Programm effizient zu gestalten:

- Schleifen abbrechen, sobald das Ziel (z. B. bei einer Suche) erreicht ist
- Optimierte Bibliotheksfunktionen verwenden
- Iterative statt rekursiver Funktionen verwenden (je nach Anwendung)
- Unnötige Tests vermeiden, indem bedingte Anweisungen gemäß ihrer Zutreffenswahrscheinlichkeit verschachtelt werden
- ...

✓ Ein Punkt je korrekter Nennung.

c)

ca)

	Quadratische Sortierverfahren	Logarithmische Sortierverfahren
Vorteil	- Einfache Implementierung - Gut für kleine Datenmengen geeignet - Leicht verständlich - Einfaches Debugging möglich	- Effizientere und schnellere Sortierung - Gute Skalierbarkeit - Häufiger Standard in Bibliotheken
Nachteil	- Je nach Datenmenge sehr ineffizient - Viele unnötige Vertauschungen notwendig - Nicht optimal für moderne Hardware - Nicht für verteilte Systeme geeignet	- Aufwändige Implementierung - Benötigt ggf. zusätzlichen Speicherplatz - Ggf. höherer Verwaltungsaufwand
Beispiel	- Bubble Sort - Selection Sort - Insertion Sort	- Merge Sort - Quick Sort - Heap Sort

✓ Es genügt einen Vor- bzw. Nachteil sowie ein Beispiel anzugeben.

✓ Ähnliche Lösungen sind möglich.

✓ Ein Punkt je korrektem Vorteil, Nachteil und Beispiel.

Prüfungsbereich 2:

Entwicklung und Umsetzung von Algorithmen

1. Aufgabe (25 Punkte)

```
1 public ArrayList<Waggon> zugreihung (ArrayList<Waggon> waggons, String
   zielbahnhof, int gefahrgutklasse, int maxAnzahl) {
2     ArrayList<Waggon> zug = new ArrayList<Waggon>();
3     int i = 0;
4     // Auswahl aller Waggons der passenden Gefahrgutklasse:
5     solange (i < waggons.laenge() && zug.laenge() < maxAnzahl)
6     {
7         Waggon w = waggons.get(i);
8         falls (w.zielbahnhof == zielbahnhof &&
9             w.gefahrgutklasse == gefahrgutklasse)
10        {
11            zug.hinzufuegen(w);
12            waggons.loeschen(i);
13        }
14        sonst
15        {
16            i++;
17        }
18        // nach dem Entfernen muss dieselbe Listenposition neu betrachtet
        werden, daher "sonst"
19    }
20    i = 0;
21    // Auffüllen mit Waggons einer niedrigeren Gefahrgutklasse:
22    solange (i < waggons.laenge() && zug.laenge() < maxAnzahl)
23    {
24        Waggon w = waggons.get(i);
25        falls (w.zielbahnhof == zielbahnhof &&
26            w.gefahrgutklasse < gefahrgutklasse &&
27            (w.prioritaet == 0 || w.gefahrgutklasse==1 && w.prioritaet==1))
28        {
29            zug.hinzufuegen(w);
30            waggons.loeschen(i);
31        }
32        sonst
33        {
34            i++;
35        }
36    }
37    Rueckgabe zug;
38 }
```

→ Fortsetzung auf nächster Seite

Prüfung 1

☑ *Jeweils fünf Punkte für jede erfüllte Aufgabe gemäß Angabe: 25 Punkte*

- Zunächst werden bis zu maxAnzahl Waggons der als Parameter angegebenen Gefahrgutklasse mit dem angegebenen Zielbahnhof ausgewählt, d. h. aus der Liste „waggons“ gelöscht und an die Ergebnisliste angehängt.
- Sollte danach die maximale Anzahl von Waggons noch nicht erreicht sein, wird der Zug mit Waggons niedriger Gefahrgutklassen und dem korrekten Zielbahnhof aufgefüllt. Da Gefahrgutzüge langsamer sind, dürfen diese Waggons allerdings keine zu hohe Priorität haben. Es gilt:
 - An einen Zug der Gefahrgutklasse 2 dürfen auch Waggons der Klasse 1 mit Priorität 0 oder 1 und Waggons der Klasse 0 mit Priorität 0.
 - An einen Zug der Gefahrgutklasse 1 dürfen auch Waggons der Klasse 0 mit Priorität 0.
- Stehen nicht genug geeignete Waggons zur Verfügung, fährt der Zug mit weniger als maxAnzahl Waggons.

☑ *Punktabzug bei logischen, Syntax- oder Einrückungs-Fehlern*

2. Aufgabe (25 Punkte)

a)

```
1  int gesperrt() {
2      int zaehler=0;
3      for (int i=0; i<5; i++) {
4          for (int j=0; j<5; j++){
5              if (verbindung[i][j]==-1){
6                  zaehler++; }
7          }
8      }
9      return zaehler; }
```

☑ *Punktabzug bei Berechnungs-, Syntax- oder Einrückungs-Fehlern.*

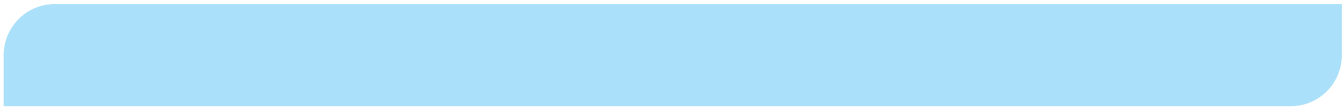
☑ *Ähnliche Lösungen sind möglich.*

b)

ba)

Start	Ziel	Ergebnis
0	1	0 1
3	0	3 1 0
0	3	Endlosrekursion (0 1 0 1 0 1 ...)

☑ *Zwei Punkte je korrekter Ergebniszeile.*



Zusätzliche übungen Lösung

1. Umfeldanalyse (10 Punkte)

a) Beispielhafte Bereiche einer Umfeldanalyse:

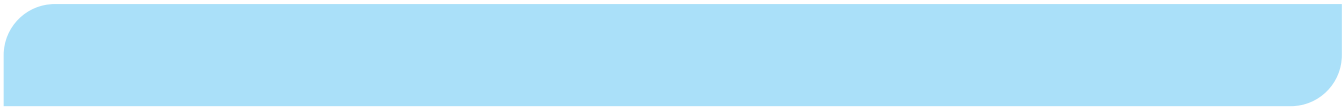
- Technisches Umfeld
- Organisatorisches Umfeld
- Rechtliches Umfeld
- Ethisches Umfeld
- Wirtschaftliches Umfeld
- ...

☑ Ein Punkt je korrekter Nennung.

b)

Bereich der Umfeldanalyse	Erläuterung eines Aspektes
Technisches Umfeld	Die bestehende IT-Infrastruktur muss erfasst und dokumentiert werden. Anschließend kann abgeglichen werden, ob die bereits bestehenden Systeme kompatibel mit den neuen Anforderungen sind.
Organisatorisches Umfeld	Es muss geprüft werden, wie bestehende Abläufe, Zuständigkeiten und Rollen in das neue ERP-System integriert werden können, um reibungslose Prozesse sicherzustellen.
Rechtliches Umfeld	Die Anforderungen der DSGVO müssen beachtet werden. Demnach sollte z. B. ermittelt werden, ob die Einwilligung der Beschäftigten zur Verarbeitung und Verwendung personenbezogener Daten vorliegt.
Ethisches Umfeld	Die Einführung des ERP-Systems sollte fair und transparent erfolgen. Beschäftigte sollten frühzeitig eingebunden und mögliche Auswirkungen auf ihre Arbeit wertschätzend kommuniziert werden.
Wirtschaftliches Umfeld	Die Kosten für Software, Schulung und Technik müssen dem erwarteten Nutzen gegenübergestellt werden, um die Wirtschaftlichkeit der ERP-Einführung zu prüfen.
...	

☑ Drei Punkte je korrekter Erläuterung eines Aspektes zu einem passenden Bereich der Umfeldanalyse.
Für nur teilweise korrekte Erläuterungen wären einer oder mehrere Punkte abzuziehen.



WiSo

Wirtschafts- und Sozialkunde

Lösung

1. Aufgabe

- ☒ können bei Streitigkeiten zwischen Arbeitgebern und Arbeitnehmern den organisierten Arbeitnehmern Rechtsbeistand leisten.

Ausschluss der übrigen Antwortmöglichkeiten:

- „haben bei Tarifverhandlungen für die Belange aller Arbeitnehmer einzutreten.“
→ Falsch. Gewerkschaften vertreten grundsätzlich die Interessen ihrer Mitglieder, nicht automatisch aller Arbeitnehmer.
- „sind in den Betrieben für die Schlichtung von Auseinandersetzungen zwischen Arbeitnehmern und Arbeitgebern zuständig.“
→ Falsch. Das ist eher Aufgabe des Betriebsrates oder der Einigungsstelle, nicht der Gewerkschaften direkt im Betrieb.
- „haben die Aufgabe, die Betriebsratswahlen zu kontrollieren.“
→ Falsch. Die Betriebsratswahlen werden von einem Wahlvorstand durchgeführt. Die Gewerkschaft kann unterstützen, hat aber keine Kontrollfunktion.
- „handeln mit den Arbeitgeberverbänden auch Betriebsvereinbarungen aus.“
→ Falsch. Betriebsvereinbarungen werden zwischen dem Betriebsrat und dem Arbeitgeber geschlossen, nicht von der Gewerkschaft.

2. Aufgabe

- ☒ Eine JAV kann nur gewählt werden, wenn bereits ein Betriebsrat besteht.

Erläuterung: Diese Anforderung ergibt sich indirekt aus den Bestimmungen zur Wahl und den Aufgaben eines Betriebsrates sowie einer JAV im BetrVG. Ein Betriebsrat ist notwendig, um die Bedingungen für die Wahl einer JAV zu schaffen

3. Aufgabe

- ☒ 8 Wochen

Erläuterung: Die Mutterschutzfrist beginnt 6 Wochen vor dem errechneten Geburtstermin und endet normalerweise 8 Wochen nach der Geburt. Bei Früh-, Mehrlingsgeburten oder wenn eine Behinderung ärztlich festgestellt wird, verlängert sich der Mutterschutz nach der Geburt auf 12 Wochen. Siehe Mutterschutzgesetz, § 3 Schutzfristen vor und nach der Entbindung.

4. Aufgabe

- ☒ Sie dürfen nicht Mitglied des Betriebsrats sein.

Erläuterung: Siehe § 61 BetrVG: „Mitglieder des Betriebsrats können nicht zu Jugend- und Auszubildendenvertretern gewählt werden.“