

Die Aufgaben 1 bis 4 beziehen sich auf die folgende Ausgangssituation:

Die NOVA Digital GmbH hat sich als Anbieter innovativer Softwareanwendungen für die Systemgastronomie einen guten Ruf erarbeitet. Aufgrund des anhaltenden Wachstums und der kontinuierlich steigenden Nachfrage nach digitalen Lösungen in der Restaurantbranche beabsichtigt die NOVA Digital GmbH, ihr bestehendes Produktportfolio gezielt zu erweitern.

1. Aufgabe (24 Punkte)

a) Sie werden gebeten, für das Vorhaben „Digitales Tischreservierungssystem“ an einer Machbarkeitsuntersuchung mitzuwirken. Ihre Aufgabe besteht darin, die Realisierbarkeit des Projekts anhand von zwei selbst gewählten Kriterien zu bewerten. **8 Punkte**

Benennen Sie die von Ihnen ausgewählten Kriterien und erläutern Sie diese jeweils.

technische Machbarkeit

Ist die Idee mit den gegebenen technischen Mitteln umsetzbar?

personelle Machbarkeit

Habe ich das Personal was nötig ist um die Idee umzusetzen ?

2. Aufgabe (27 Punkte)

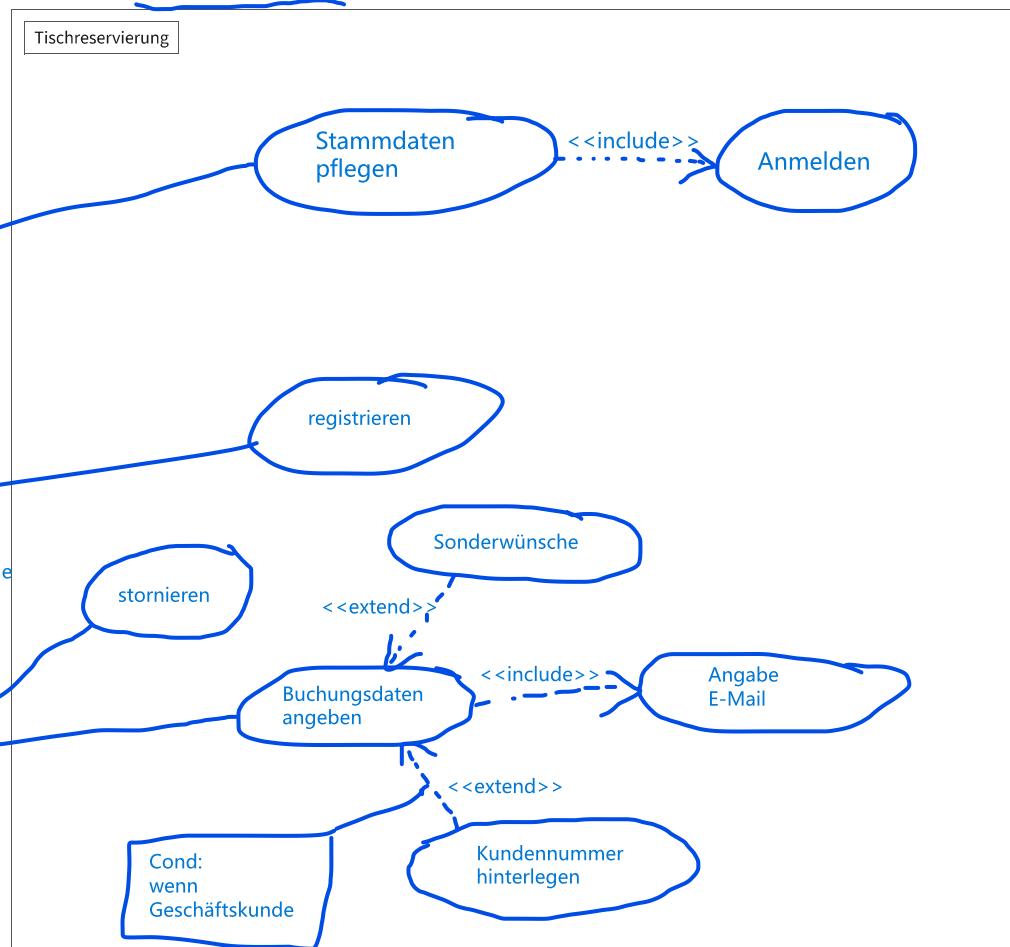
a) Die bestehende Anwendung soll um ein Modul zur Tischreservierung ergänzt werden. Geschäftskunden benötigen eine Registrierung, um per Rechnung bezahlen zu können. Privatgäste können eine Reservierung auch ohne Registrierung durchführen. **18 Punkte**

Korrekturband

Für das Tischreservierungsmodul gelten folgende Anforderungen:

- ✓ Ein Mitarbeiter darf reservierungsrelevante Stammdaten (z. B. Öffnungszeiten, Tischanzahl) pflegen. Voraussetzung ist eine Anmeldung mit gültigen Zugangsdaten.
- ✓ Geschäftskunden müssen sich einmalig registrieren, um eine Kundennummer zu erhalten.
- ✓ Bei einer Tischreservierung gibt der Gast die Buchungsdaten (Datum, Uhrzeit und Personenanzahl) an. Zusätzlich können optionale Sonderwünsche (z. B. ~~Barrierefrei~~ separater Raum, barrierefreier Zugang) angegeben werden.
- ✓ Jede Reservierung erfordert die Angabe einer gültigen E-Mail-Adresse. Bei Geschäftskunden ist zusätzlich die Kundennummer zu hinterlegen.
- ✓ Eine bestehende Reservierung kann von allen Gästegruppen storniert werden.

Erstellen Sie ein Anwendungsfalldiagramm für das Tischreservierungsmodul.



Fortsetzung 2. Aufgabe →

Fortsetzung 3. Aufgabe

bc) Ein Kollege weist Sie darauf hin, dass für das einzusetzende Protokoll XSD-Dateien vorliegen. XSD-Dateien (XML-Schemadefinition) beschreiben den Aufbau eines XML-Dokuments und ermöglichen damit eine grundlegende Strukturvalidierung.

6 Punkte

Korrekturand

Begründen Sie anhand der nachfolgenden XML-Datei, warum eine ausschließliche Validierung gegen das XML-Schema nicht ausreicht. Die Fehlerstelle im Dokument ist markiert.

```
<?xml version="1.0" encoding="UTF-8" ?>
<Invoice>
  <InvoiceLines>
    <InvoiceLine>
      <LineNumber>1</LineNumber>
      <ArticleCode>ART-4471</ArticleCode>
      <UnitPrice>12.00</UnitPrice>
      <Quantity>8</Quantity>
      <NetAmount>96.00</NetAmount>
      <TaxAmount>18.24</TaxAmount>
      <GrossAmount>114.24</GrossAmount>
    </InvoiceLine>
    <InvoiceLine>
      <LineNumber>2</LineNumber>
      <ArticleCode>ART-8823</ArticleCode>
      <UnitPrice>25.00</UnitPrice>
      <Quantity>4</Quantity>
      <NetAmount>100.00</NetAmount>
      <TaxAmount>19.00</TaxAmount>
      <GrossAmount>119.00</GrossAmount>
    </InvoiceLine>
  </InvoiceLines>
  <InvoiceSummary>
    <TotalNet>238.00</TotalNet> <!-- Fehler -->
  </InvoiceSummary>
</Invoice>
```

Eine XML Validierung kann den Inhalt nicht auf Richtigkeit prüfen sondern nur die Struktur des XML.

c) Um den Entwicklungsaufwand unterschiedlicher technischer Umsetzungen der eRechnung besser einschätzen zu können, werden verschiedene Ansätze mithilfe von **Prototyping** erprobt. 6 Punkte

Erläutern Sie, was unter **Prototyping** in diesem Kontext zu verstehen ist und unter welchen Umständen der Einsatz von Prototyping in der Konzeptionsphase sinnvoll ist.

Prototyping ist eine Methode der Software- und Produktentwicklung

um frühzeitig zu erkennen ob ein Idee so realisierbar ist wie man es sich vorstellt.

Also eine möglichst einfache Vorabversion des geplanten Produktes.

4. Aufgabe (24 Punkte)

Zur Qualitätssicherung können Gäste einzelne Restaurantleistungen von Servicepersonal und Küchenteam nach jedem Besuch online bewerten. Die Datenspeicherung erfolgt zunächst in einer nicht normalisierten Tabelle.

Ausschnitt aus der Tabelle Bewertung

Gast_ID	Gast_Kontakt	Besuch_ZS	Leist_ID	Leistung_Bez	Bewertung	MA_ID	MA_Name	MA_Tel
21	Martin Bauer 01637821456 m.bauer@gmx.de	2025-09-15- 18-30	101, 402, 611	Service, Rinderbraten, Panna Cotta	5, 4, 3	12, 47, 47	Altmann, Reuter, Reuter	12, 08, 08
35	Sabine Kramer 01752341890 s.kramer@web.de	2025-10-02- 19-00	101, 723	Service, Caprese Salat	2, 4	12, 63	Altmann, Vogel	12, 22
21	Martin Bauer 01637821456 m.bauer@gmx.de	2025-10-11- 20-00	723	Caprese Salat	5	<u>47</u>	Reuter	08
44	Frank Lehmann 01584629037 f.lehmann@t-online.de	2025-09-20- 17-45	402, 723	Rinderbraten, Caprese Salat	3, 5	63, 63	Vogel, Vogel	22, 22

a) Bei dieser Art der Datenhaltung werden zahlreiche Informationen mehrfach gespeichert.

2 Punkte

Erklären Sie den Begriff Redundanz und belegen Sie ihn mit einem konkreten Beispiel aus der Tabelle Bewertung.

z.B. wenn Daten in eine DB mehrfach vorkommen. In diesem Fall kommt der Name des Gastes Martin Bauer mehrfach vor.

b) Redundante Datenhaltung kann zu unerwarteten Anomalien führen.

6 Punkte

Ergänzen Sie die nachfolgende Tabelle: Beschreiben Sie jede Anomalieart und veranschaulichen Sie sie mit einem konkreten Beispiel aus der Tabelle Bewertung.

Anomalieart	Beschreibung mit Beispiel
Änderungsanomalie	Nach einer Änderung gibt es mehrere Daten (Namen) z.B. zu einer Person und es kann nicht festgestellt werden welches der richtige/aktuelle Name ist. Wenn der Name von Hr. Bauer sich an einer Stelle ändert...
Einfügeanomalie	Einfügen einer Mitarbeiter_Id zu der es keinen Mitarbeiter in der DB gibt.
Löschanomalie	Ich lösche einen Datensatz auf den noch von anderer Stelle Bezug genommen wird. z.B. wir löschen einen Mitarbeiter der noch in Buchungsdaten enthalten ist mit seiner ID.

Fortsetzung 4. Aufgabe →

Fortsetzung 4. Aufgabe

c) Überführen Sie die Tabelle Bewertung in die 3. Normalform und stellen Sie das Ergebnis als relationales Datenbankmodell dar.

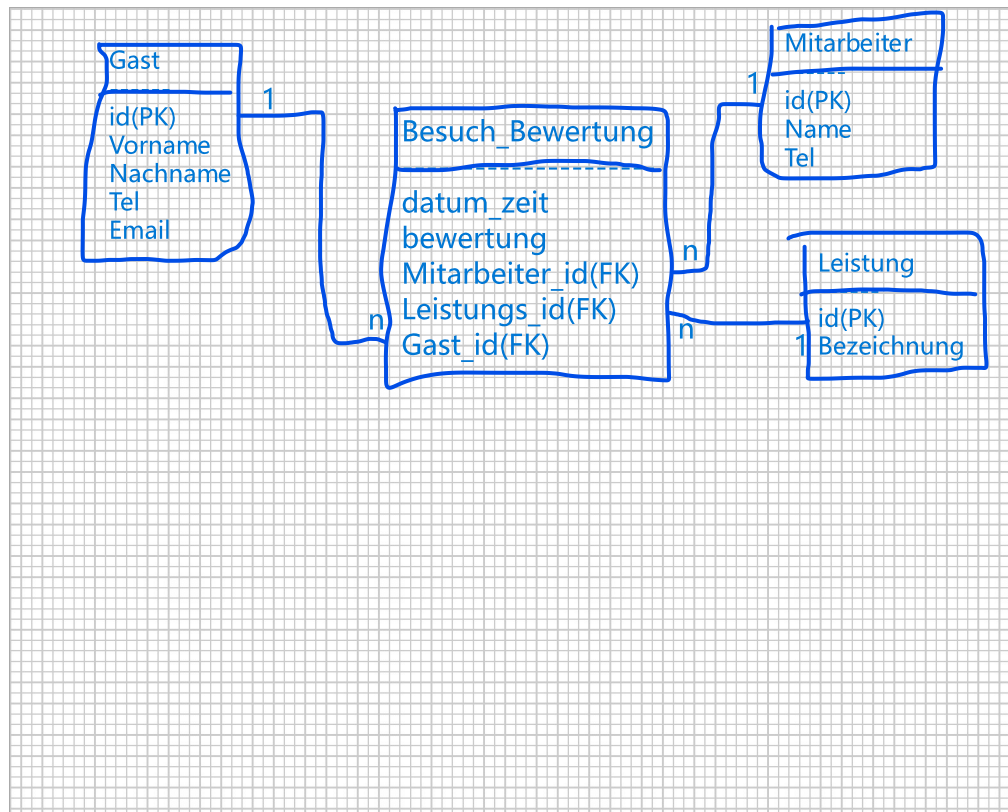
16 Punkte

Korrekturband

Gast_ID	Gast_Kontakt	Besuch_ZS	Leist_ID	Leistung_Bez	Bewertung	MA_ID	MA_Name	MA_Tel
21	Martin Bauer 01637821456 m.bauer@gmx.de	2025-09-15-18-30	101, 402, 611	Service, Rinderbraten, Panna Cotta	5, 4, 3	12, 47, 47	Altmann, Reuter, Reuter	12, 08, 08
35	Sabine Kramer 01752341890 s.kramer@web.de	2025-10-02-19-00	101, 723	Service, Caprese Salat	2, 4	12, 63	Altmann, Vogel	12, 22
21	Martin Bauer 01637821456 m.bauer@gmx.de	2025-10-11-20-00	723	Caprese Salat	5	47	Reuter	08
44	Frank Lehmann 01584629037 f.lehmann@t-online.de	2025-09-20-17-45	402, 723	Rinderbraten, Caprese Salat	3, 5	63, 63	Vogel, Vogel	22, 22

Hinweise:

- Sämtliche Primär- und Fremdschlüssel sind ausschließlich aus den vorhandenen Datenfeldern abzuleiten.
- Kennzeichnen Sie Primärschlüssel mit (PK) und Fremdschlüssel mit (FK). Tragen Sie alle Beziehungen zwischen den Tabellen ein.
- Gemäß Spezifikation wird eine bestimmte Leistung pro Besuch nur einmalig erbracht.



PRÜFUNGSZEIT – NICHT BESTANDTEIL DER PRÜFUNG!

Wie beurteilen Sie im Rückblick auf die Bearbeitung die Ihnen zur Verfügung gestandene Prüfungszeit?

- 1 Sie hätte kürzer bemessen sein können.
- 2 Sie war angemessen.
- 3 Sie hätte länger sein sollen.

4. Aufgabe (30 Punkte)

a) Für die Buchung von Räumen und zugehörigen Ressourcen ist der nachfolgend beschriebene Geschäftsprozess vorgesehen. Abbruchmöglichkeiten brauchen nicht berücksichtigt zu werden.

- Der Interessent übermittelt eine Anfrage mit dem gewünschten Buchungszeitraum. Daraufhin prüft der Anbieter die Verfügbarkeit dieses Zeitraums. Ist der Zeitraum nicht frei, gibt der Interessent einen alternativen Zeitraum an.
- Ist der Zeitraum verfügbar, teilt der Interessent seine Anforderungen hinsichtlich der benötigten Ressourcen und der Raumgröße mit. Beide Anforderungen werden daraufhin von den zuständigen Fachabteilungen gleichzeitig geprüft.
- Können beide Anforderungen erfüllt werden, erstellt der Anbieter ein Angebot, das der Interessent anschließend prüft.
- Kann der Anbieter die Anforderungen nicht erfüllen oder möchte der Interessent das Angebot nicht annehmen, passt der Interessent seine Anforderungen an.
- Nimmt der Interessent das Angebot an, bucht er die Ressourcen und den Raum verbindlich.

Erstellen Sie auf der nebenstehenden Seite ein UML-Aktivitätsdiagramm, das diesen Prozess vollständig abbildet.

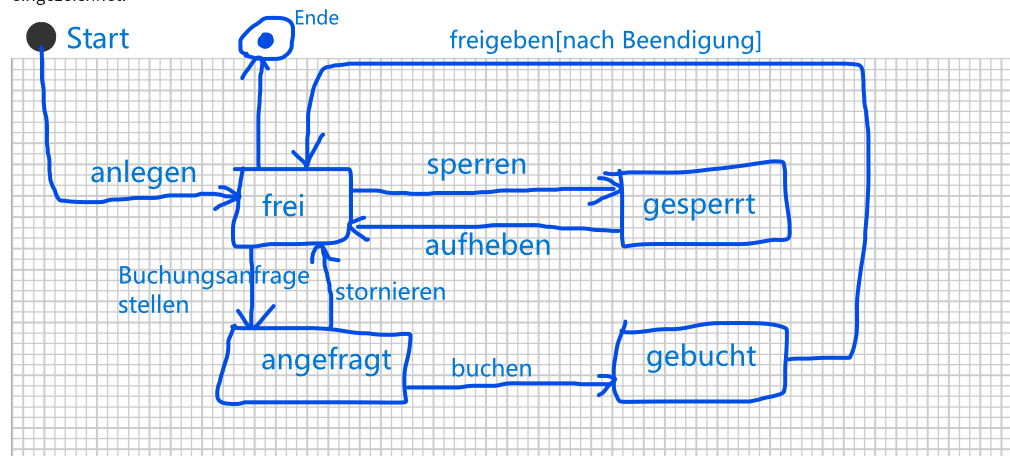
14 Punkte

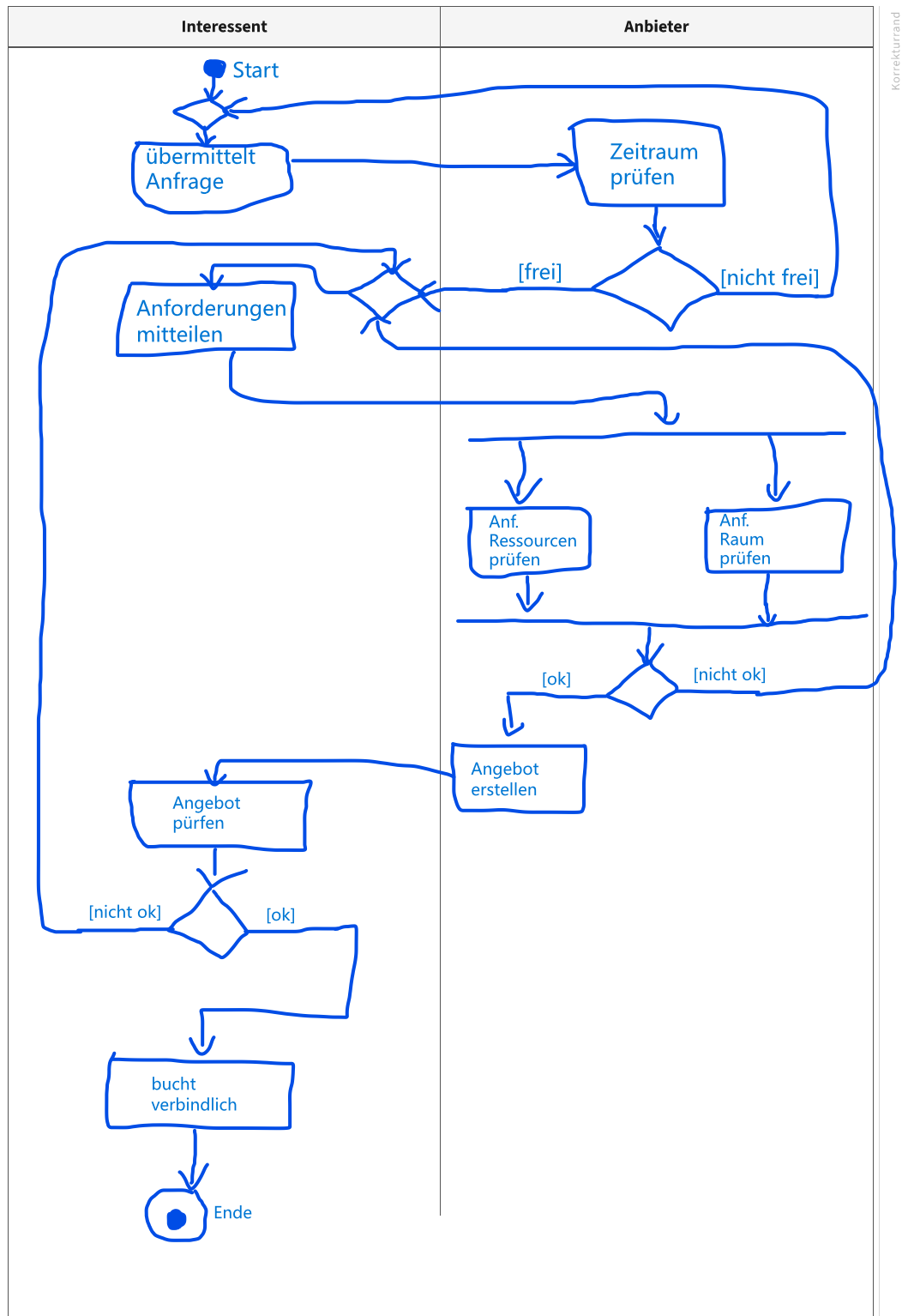
b) Räume können verschiedene Zustände einnehmen. Die folgenden Regeln gelten:

- Nach der Anlage eines Raums ist dieser zunächst frei.
- Freie Räume können gesperrt werden. Eine Sperrung kann wieder aufgehoben werden, woraufhin der Raum erneut den Zustand „frei“ erhält.
- Für einen freien Raum kann eine Buchungsanfrage gestellt werden; der Raum wechselt dann in den Zustand „angefragt“.
- Angefragte Räume können storniert oder gebucht werden. Bei Stornierung wechselt der Raum wieder in den Zustand „frei“.
- Gebuchte Räume kehren nach Abschluss der Veranstaltung in den Zustand „frei“ zurück.

Erstellen Sie ein UML-Zustandsdiagramm für die möglichen Raumzustände. Der Startzustand ist bereits eingezeichnet.

11 Punkte





Die Aufgaben 1 bis 4 beziehen sich auf die folgende Ausgangssituation:

Ein überregional tätiger Weiterbildungsanbieter betreibt an sieben Standorten Seminare, Schulungen und Fachkongresse. An den Standorten stehen zusammen 38 Seminarräume mit insgesamt über 520 IT-Arbeitsplätzen bereit. Alle Veranstaltungen können auch online gebucht werden.

Die gesamte IT-Infrastruktur wird von zwei zentralen Rechenzentren betrieben.

Da ein Teil der Schulungs- und Büroräume nicht ausgelastet ist, sollen diese künftig flexibel an externe Nutzer vermietet werden.

1. Aufgabe (25 Punkte)

Im Zuge der Entwicklung von Lernsoftware soll ein Sortieralgorithmus erstellt werden, der ein Array von Raum-Objekten aufsteigend nach deren Auslastungsgrad sortiert.

Raum
...
+ <u>getAuslastung()</u> : Integer
...

Der Algorithmus soll nach folgendem Verfahren vorgehen:

- Erstes und zweites Element vergleichen, bei Bedarf tauschen
- Zweites und drittes Element vergleichen, bei Bedarf tauschen
- Vorgang für alle aufeinanderfolgenden Elementpaare bis zum Arrayende wiederholen
- Den gesamten Durchlauf so oft wiederholen, bis das Array vollständig sortiert ist

a) Entwickeln Sie einen Algorithmus, der ein Array von Raum-Objekten nach dem beschriebenen oder einem gleichwertigen alternativen Verfahren sortiert. Stellen Sie den Algorithmus in Pseudocode, als Struktogramm oder als Programmablaufplan dar.

10 Punkte

void sortieren (raeume: Array von Raum)

```

für x von 0 , bis x kleiner raeume.laenge() - 1, x++
  für y von 0, bis y kleiner raeume.laenge() - 1 - x , y++
    wenn raeume[y].getAuslastung() > raeume[y+1].getAuslastung()
      tmp      = raeume[y]
      raeume[y] = raeume[y+1]
      raeume[y+1] = tmp
    ende wenn
  ende für
ende für

```

b) Zur Flexibilisierung des Algorithmus wird eine Vergleichsfunktion benötigt, die:

- den Wert 0 zurückgibt, wenn beide übergebenen Objekte als gleich eingestuft werden,
- einen positiven Wert zurückgibt, wenn das erste Objekt größer als das zweite ist, und
- andernfalls einen negativen Wert liefert.

Entwickeln Sie die Funktion **vergleicheMitarbeiter(Mitarbeiter m1, Mitarbeiter m2)**. Der Vergleich soll auf Basis des jeweiligen Monatsgehalts durchgeführt werden.

Stellen Sie die Funktion in Pseudocode, als Struktogramm oder als Programmablaufplan dar.

5 Punkte

Mitarbeiter
...
+ getMonatsgehalt() : Double
...

vergleicheMitarbeiter (Mitarbeiter m1, Mitarbeiter m2) : Integer

```

wenn m1.getMonatsgehalt() == m2.getMonatsgehalt()
    rückgabe 0
sonst wenn m1.getMonatsgehalt() grösser m2.getMonatsgehalt()
    rückgabe 1
sonst
    rückgabe -1
ende wenn
ende funktion
  
```

c) Da der Sortieralgorithmus künftig auch für andere Objekttypen verwendet werden soll, wird die Nutzung generischer Klassen empfohlen.

3 Punkte

Erklären Sie, was unter einer generischen Klasse zu verstehen ist.

Eine Klasse die abhängig von einem gegebenen Datentyp zur Compilezeit erstellt wird.

Dabei wird für einen Platzhalter <T> der gewünschte Datentyp eingesetzt .

Fortsetzung 1. Aufgabe →

```
List<Auto> meine_autos = new List<Auto>();
```

Fortsetzung 1. Aufgabe

d) Für eine verbesserte Version des Sortieralgorithmus soll die vorhandene generische Klasse **Comparator<T>** eingesetzt werden. Ein Comparator-Objekt kann eine Vergleichsfunktion speichern, die zwei Parameter vom Typ T entgegennimmt und einen Integer-Rückgabewert liefert.

Verwendung eines Comparator<T>-Objekts (Beispiel)	
Speichern einer Funktion in einem Comparator-Objekt:	Comparator<Mitarbeiter> c = vergleicheMitarbeiter;
Aufruf der gespeicherten Funktion:	c(mitarbeiter1, mitarbeiter2)

Zusätzlich soll die generische Klasse **Liste<T>** genutzt werden, um die zu sortierenden Objekte vom Typ T aufzunehmen.

Liste<T>
...
+ lesen(position: Integer) : T
+ schreiben(position: Integer, element: T) : void
+ laenge() : Integer
...

Aktualisieren Sie Ihren Sortieralgorithmus so, dass er ein Liste<T>-Objekt sowie ein Comparator<T>-Objekt als Parameter entgegennimmt. Für den Vergleich im Algorithmus soll die im Comparator-Objekt gespeicherte Funktion verwendet werden.

7 Punkte

sortieren(Liste<T> liste, Comparator<T> c): void

```
für x von 0 , bis x kleiner liste.laenge() - 1, x++
  für y von 0, bis y kleiner liste.laenge() - 1 - x , y++
    wenn c( liste.lesen( y ) , liste.lesen( y + 1) ) > 0
      tmp = liste.lesen( y )
      liste.schreiben( y , liste.lesen( y + 1 ) )
      liste.schreiben( y + 1 , tmp )
    ende wenn
  ende für
ende für
```

2. Aufgabe (25 Punkte)

Für die Lernplattform sollen interaktive Lehrmaterialien entwickelt werden. Ihre Aufgabe besteht darin, einen vorhandenen rekursiven Algorithmus zur Umrechnung von Dezimalzahlen in Dualzahlen mithilfe eines Schreibtischtests auf Korrektheit zu prüfen und eventuelle Fehler zu beheben.

a) Der folgende Pseudocode-Algorithmus ist gegeben:

```
void Konvertierung(dezZahl: int)
BEGINN
    rest: int;
    rest = dezZahl modulo 2;
    AUSGABE(rest);
    dezZahl = dezZahl div 2; //div steht für ganzzahlige Division
    WENN dezZahl > 0 DANN
        Konvertierung(dezZahl);
    ENDE
ENDE
```

aa) Führen Sie den Schreibtischtest für beide Testfälle durch und tragen Sie das ermittelte Ergebnis in die rechte Spalte ein. **4 Punkte**

Testfall	Eingabe (dezimal)	Erwartete Ausgabe (dual)	Tatsächliche Ausgabe (Schreibtischtest)
1	9	1001	
2	13	1101	

ab) Beschreiben Sie den im Algorithmus enthaltenen Fehler und schlagen Sie eine Möglichkeit zur Korrektur vor. **4 Punkte**

ac) Der korrigierte Algorithmus soll im Rahmen eines White-Box-Tests validiert werden.

Erklären Sie den Begriff Anweisungsüberdeckung und geben Sie an, wie viele Testfälle mindestens benötigt werden, um für den vorliegenden Algorithmus eine vollständige Anweisungsüberdeckung zu erzielen. **3 Punkte**

b) Vergleichen Sie die rekursive mit einer iterativen Lösung: Nennen Sie jeweils einen Vorteil und einen Nachteil der rekursiven Implementierung. **4 Punkte**

Vorteil:

Nachteil:

Fortsetzung 2. Aufgabe →

Fortsetzung 2. Aufgabe

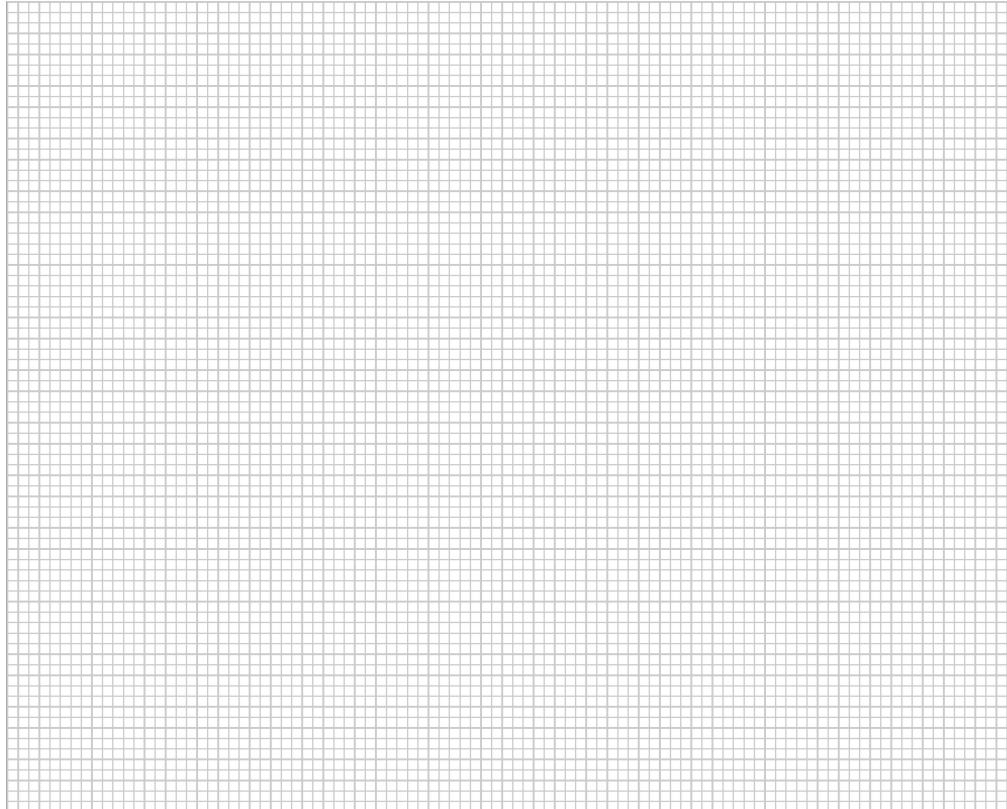
c) Der rekursive Algorithmus soll durch eine iterative Variante ersetzt werden. Dafür steht die Klasse **Stapel** zur Verfügung, deren Methoden in der folgenden Tabelle beschrieben sind:

<code>ablegen(wert:int):void</code>	Legt den übergebenen Wert oben auf dem Stapel ab
<code>tiefe():int</code>	Gibt die aktuelle Anzahl der Elemente auf dem Stapel zurück
<code>zugreifen(pos:int):int</code>	Liefert das Element an Position pos; Position 0 entspricht dem obersten Element
<code>Stapel()</code>	Konstruktor; erzeugt einen leeren Stapel

Entwickeln Sie den iterativen Algorithmus, der die Dezimal-zu-Dual-Konvertierung korrekt ausführt.

10 Punkte

void Konvertierung(dezZahl: int)



Korrekturand

Dieses Blatt kann an der Perforation aus dem Aufgabensatz herausgetrennt werden!

Abbildung zur 3. Aufgabe

Buchungsbestätigung

Vorgangsnummer: 207531

Datum der Buchung: 12. Oktober 2022

Peter Hoffmann

Bergweg 8

04109 Leipzig

Tel: 0341987654

E-Mail: p.hoffmann@mailbox.de

Kundennummer: 215

Ansprechpartnerin: Monika Stern

Durchwahl: 089 765-3412

E-Mail: m.stern@bildung-xyz.de

Sehr geehrter Herr Hoffmann,

herzlichen Dank für Ihre Buchung. Nachstehend finden Sie alle relevanten Informationen zu Ihrer Reservierung:

Standort: Hamburg Mietzeitraum: 14. November 2022 bis 17. November 2022

Bildungszentrum XYZ

Hafenweg 3

20457 Hamburg

Gebuchte Räume am Standort:

1. Raum: 102 Art: Tagungsraum
2. Raum: 104 Art: Tagungsraum
3. Raum: 213 Art: Schulungsraum

Gebuchte Ausstattung:

- 2 x Laserdrucker
- 2 x Projektor
- 3 x Flipchart
- 1 x Moderationskoffer

Bei Rückfragen zur Buchung wenden Sie sich bitte direkt an Frau Stern, die Ihre Buchung betreut.

Mit freundlichen Grüßen

Ihr Service-Team

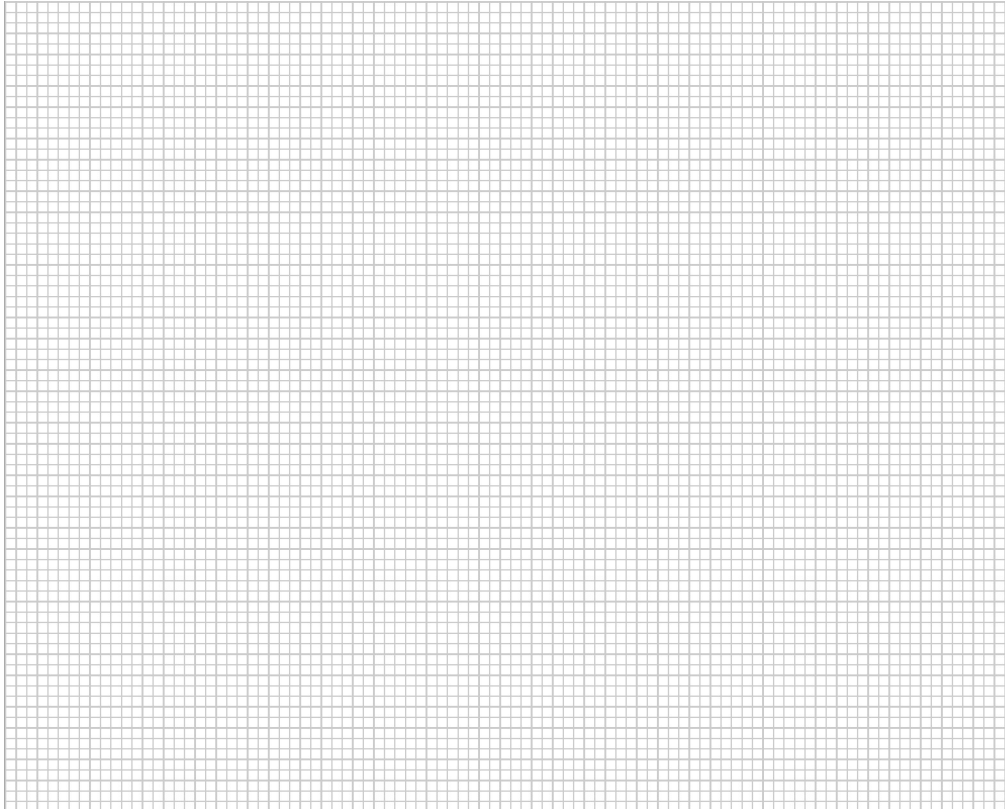
Zusätzliche Anforderungen:

- Jede Buchung soll dem zuständigen Betreuungsmitarbeiter zugeordnet sein.
- Der geografische Standort der gebuchten Räume soll ebenfalls erfasst werden.

3. Aufgabe (25 Punkte)

Unternehmensdaten sollen in einem relationalen Datenbanksystem strukturiert gespeichert werden. Ein erstes ER-Modell, das bereits die Entitätstypen Mitarbeiter und Serviceleistung enthält, liegt vor. Dieses Modell soll um Raumbuchungen durch Kunden ergänzt werden. Als Informationsgrundlage dienen die oben abgebildete Buchungsbestätigung sowie die nachfolgend genannten Anforderungen.

- a) Ergänzen Sie das nachstehende ER-Modell ausschließlich um die zusätzlich notwendigen Entitätstypen und erfassen Sie die Beziehungen zwischen ihnen einschließlich der Kardinalitäten. (Hinweis: Attribute müssen nicht eingetragen werden.)

18 Punkte

Fortsetzung 3. Aufgabe →

Fortsetzung 3. Aufgabe

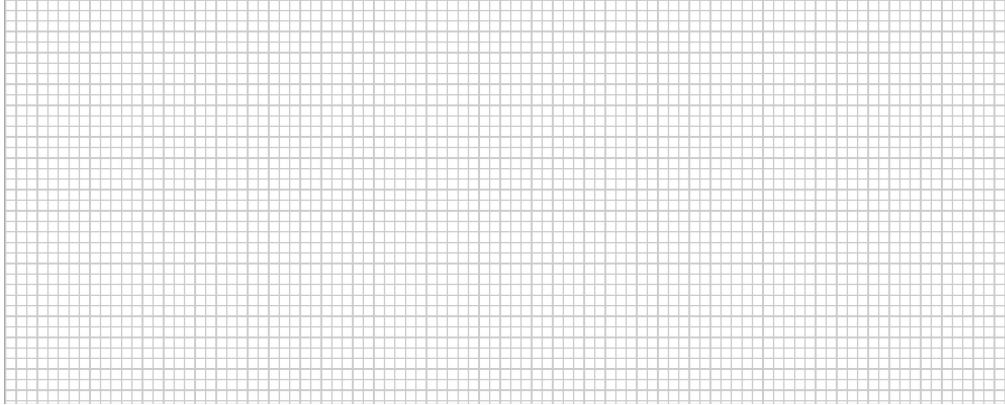
b) Ausgehend vom ER-Modell soll nun ein relationales Datenmodell erstellt werden.

Überführen Sie dafür **ausschließlich** die Entitätstypen **Mitarbeiter und Serviceleistung** sowie deren Beziehung und alle Attribute in ein relationales Datenmodell, das der 3. Normalform genügt.

Kennzeichnen Sie Primärschlüssel mit PK und Fremdschlüssel mit FK und tragen Sie alle Beziehungen mit Kardinalitäten ein.

Hinweis: Die Anschrift muss nicht in eine eigene Tabelle ausgelagert werden.

10 Punkte

A large rectangular area filled with a fine grid of small squares, resembling graph paper, intended for the student to draw their relational data model.

