

Familienname, Vorname (bitte durch eine Leerspalte trennen, ä = ae etc.)

[illegible]

Berufsnummer

Prüflingsnummer

5	5
---	---

1	1	9	6
---	---	---	---

--	--	--

Termin: Mittwoch, 26. November 2008

IHK

Abschlussprüfung Winter 2008/09

Fachinformatiker/Fachinformatikerin
Anwendungsentwicklung

1196



Ganzheitliche Aufgabe I Fachqualifikationen

5 Handlungsschritte
90 Minuten Prüfungszeit
100 Punkte

Bearbeitungshinweise

1. Der vorliegende Aufgabensatz besteht aus insgesamt 5 Handlungsschritten zu je 25 Punkten.

In der Prüfung zu bearbeiten sind 4 Handlungsschritte, die vom Prüfungsteilnehmer frei gewählt werden können.

Der nicht bearbeitete Handlungsschritt ist durch Streichung des Aufgabentextes im Aufgabensatz und unten mit dem Vermerk „Nicht bearbeiteter Handlungsschritt: Nr. ... „ an Stelle einer Lösungsniederschrift deutlich zu kennzeichnen. Erfolgt eine solche Kennzeichnung nicht oder nicht eindeutig, gilt der 5. Handlungsschritt als nicht bearbeitet

2. Füllen Sie zuerst die **Kopfzeile** aus. Tragen Sie Ihren Familiennamen, Ihren Vornamen und Ihre Prüflings-Nr. in die oben stehenden Felder ein.
3. Lesen Sie bitte den **Text** der Aufgaben ganz durch, bevor Sie mit der Bearbeitung beginnen.
4. Halten Sie sich bei der Bearbeitung der Aufgaben genau an die **Vorgaben der Aufgabenstellung** zum Umfang der Lösung. Wenn z. B. vier Angaben gefordert werden und Sie sechs Angaben anführen, werden nur die ersten vier Angaben bewertet.
5. Tragen Sie die frei zu formulierenden **Antworten dieser offenen Aufgabenstellungen** in die dafür lt. Aufgabenstellung vorgesehenen Bereiche (Lösungszeilen, Formulare, Tabellen u. a.) des Arbeitsbogens ein.
6. Sofern nicht ausdrücklich ein Brief oder eine Formulierung in ganzen Sätzen gefordert werden, ist eine **stichwortartige Beantwortung** zulässig.
7. Schreiben Sie deutlich und gut lesbar. Ein nicht eindeutig zuzuordnendes oder **unleserliches Ergebnis** wird als **falsch** gewertet.
8. Zur Lösung der Rechenaufgaben darf ein nicht programmierter, netzunabhängiger **Taschenrechner** ohne Kommunikationsmöglichkeit mit Dritten verwendet werden.
9. Wenn Sie ein **gerundetes Ergebnis** eintragen und damit weiterrechnen müssen, rechnen Sie (auch im Taschenrechner) nur mit diesem gerundeten Ergebnis weiter.
10. Ein **Tabellenbuch** oder ein **IT-Handbuch** oder eine **Formelsammlung** ist als Hilfsmittel zugelassen.
11. Für **Nebenrechnungen/Hilfsaufzeichnungen** können Sie das im Aufgabensatz enthaltene Konzeptpapier verwenden. Dieses muss vor Bearbeitung der Aufgaben herausgetrennt werden. Bewertet werden jedoch nur Ihre Eintragungen im Aufgabensatz.

Nicht bearbeiteter Handlungsschritt ist Nr.

Wird vom Korrektor ausgefüllt!

Bewertung

Für die Bewertung gilt die Vorgabe der Punkte in den Lösungshinweisen. Für den abgewählten Handlungsschritt ist anstatt der Punktzahl die Buchstabenkombination „AA“ in die Kästchen einzutragen.

Prüfungszeit

Die entsprechende Ziffer (1, 2 oder 3) finden Sie in der Abfrage nach der Prüfungszeit im Anschluss an die letzte Aufgabe

Unterschrift

Gemeinsame Prüfungsaufgaben der Industrie- und Handelskammern. Dieser Aufgabensatz wurde von einem überregionalen Ausschuss, der entsprechend § 40 Berufsbildungsgesetz zusammengesetzt ist, beschlossen.

Ausschluss, der entsprechend § 40 Berufsberatungsgesetz zusammengefasst ist, beschließen.
Die Vervielfältigung, Verbreitung und öffentliche Wiedergabe der Prüfungsaufgaben und Lösungen ist nicht gestattet. Zuwiderhandlungen werden zivil- und strafrechtlich (§§ 97 ff., 106 ff. UrhG) verfolgt. – © ZPA Nord-West 2008 – Alle Rechte vorbehalten!

Die Handlungsschritte 1 bis 5 beziehen sich auf die folgende Ausgangssituation:

Sie sind Mitarbeiter/-in der XSEC AG, einem Systementwickler für Gebäudemanagement und -sicherheit. Die XSEC AG wurde von der Schubert GmbH, einem Medikamentenhersteller, mit der Erstellung eines Mitarbeiterverwaltungssystems beauftragt.

Sie sollen im Rahmen dieses Auftrags

1. ein relationales Datenbankmodell entwickeln.
2. SQL-Abfragen zur Auswertung einer Zeiterfassungsdatenbank erstellen.
3. ein Use-Case-Diagramm erstellen und ein Programm entwickeln.
4. ein Klassendiagramm und ein Programm (objektorientiert) zur Zugangskontrolle erstellen.
5. eine Programmanalyse durchführen.

1. Handlungsschritt (25 Punkte)

Bislang verwaltet die Schubert GmbH die Mitarbeiterdaten in folgender Excel-Tabelle:

Neben den Stammdaten der Mitarbeiter werden auch deren Zugangsberechtigungen gespeichert. Die Mitarbeiter/-innen dürfen sich in einem oder mehreren Bereichen nur zu bestimmten Zeiten aufhalten.

Mitarbeiter_ID	Nachname	Vorname	PLZ	Ort	BLZ	Bank	Konto_Nr	Bereich	Zutritt_ab	Austritt_bis
0811	Müller	Max	50871	Köln	51234588	KBank	1234567	Labor	06:00	20:00
0811	Müller	Max	50871	Köln	51234588	KBank	1234567	Lager	06:00	20:00
0988	Schulz	Liese	50337	Brühl	76589923	LieBa	8917235	Küche	20:00	22:00
0988	Schulz	Liese	50337	Brühl	76589923	LieBa	8917235	Kantine	20:00	22:00
0988	Schulz	Liese	50337	Brühl	76589923	LieBa	8917235	Kasino	20:00	22:00
1004	Klein	Manni	53111	Bonn	51234588	KBank	2345678	Lager	06:00	20:00
1005	Groß	Otto	41460	Neuss	50070080	Rheiba	1001234	Labor	20:00	24:00
...										

Die XSEC AG schlägt der Schubert GmbH vor, zukünftig diese Daten in einer Datenbank zu verwalten, um die Änderungs- und Löschanomalien dieser Excel-Tabelle auszuschließen.

aa) Nennen Sie für den oben dargestellten Datenbestand zwei Änderungen, die zu Änderungsanomalien führen. (4 Punkte)

ab) Nennen Sie für den oben dargestellten Datenbestand zwei Löschanomalien, für den Fall, dass nach einer Kündigung alle Daten von Liese Schulz gelöscht werden. (6 Punkte)

b) Erstellen Sie auf der Folgeseite anhand der oben gegebenen Excel-Tabelle ein relationales Datenbankmodell. Kennzeichnen Sie die Primär- und Sekundärschlüssel. (15 Punkte)

2. Handlungsschritt (25 Punkte)

Korrekturrand

- a) Für den Mitarbeiter Jens Müller wurde in der Tabelle Fehlzeit ein Datensatz falsch erfasst:

Statt einer zweitägigen „Dienstreise“ vom 17. bis 18.11.2008 wurde versehentlich ein eintägiger „Urlaub“ für den 17.11.2008 eingetragen (siehe perforierte Anlage, Tabelle Fehlzeit).

Erstellen Sie eine SQL-Anweisung, mit der die Korrektur durchgeführt werden kann.

(5 Punkte)

- b) Erstellen Sie eine SQL-Abfrage, welche die Urlaubstage aller Mitarbeiter im Jahr 2008 ermittelt.

Beispielausgabe:

811	Müller	Jens	15
812	Scholz	Birgit	10
815	Schmidt	Ulrich	0
817	Storck	Hans	0
841	Ullmann	Franz	21

(10 Punkte)

- c) Die bestehende Datenbank soll wie im Folgenden beschrieben verändert werden.

Erstellen Sie dazu jeweils die SQL-Anweisung.

- ca) Die Tabelle *Fehlzeit* soll gelöscht werden.

(2 Punkte)

Fortsetzung 2. Handlungsschritt →

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

Anlage zum 2. und 5. Handlungsschritt

Mitarbeiter

MA_ID	Nachname	Vorname	Geb-Datum	Tagesarbeitszeit
811	Müller	Jens	14.04.1982	8
812	Scholz	Birgit	23.08.1964	4
815	Schmidt	Ulrich	02.11.1957	8
817	Storck	Hans	14.11.1990	6
841	Ullmann	Franz	21.12.1959	8
902	Sorge	Susanne	02.03.1952	8
...				

KommenGehenBuchung

KG_ID	MA_ID	Datum	Kommen_Zeit	Gehen_Zeit
1	811	17.04.2008	07:00	11:45
2	811	17.04.2008	12:15	16:00
3	811	18.04.2008	07:32	08:10
4	902	17.04.2008	07:21	12:06
...	...			

Fehlzeit

FZ_ID	MA_ID	Von_Datum	Bis_Datum	Grund	Fehltage
1	811	18.04.2008	23.04.2008	Urlaub	4
2	902	18.04.2008	08.05.2008	Krank	14
3	811	19.06.2008	20.06.2008	Krank	2
4	811	17.11.2008	17.11.2008	Urlaub	1
5	904	31.12.2008	31.12.2008	Urlaub	1
6	904	01.01.2009	09.01.2009	Urlaub	6

Hinweis: Jahresübergreifender Urlaub generiert zwei Datensätze (siehe Beispiel MA_ID 904).

Monat_Jahr_Arbeitstage

MJ_ID	Monat	Jahr	Arbeitstage
...	...		
123	10	2008	20
124	11	2008	20
125	12	2008	20

- cb) Es soll eine Tabelle *Fehlzeitgrund* mit folgenden Feldern erstellt werden.

Fehlzeitgrund

Grund_ID	Grund
1	Urlaub
2	Krank
3	Dienstreise

(3 Punkte)

- cc) Die Tabelle *Fehlzeit* soll in der dargestellten Form neu erstellt werden. In die Tabelle *Fehlzeit* sollen in der Spalte Grund_ID nur solche Werte eingetragen werden können, die in der Tabelle *Fehlzeitgrund* als Primärschlüssel vorkommen.

Fehlzeit

MA_ID	Von_Datum	Bis_Datum	Grund_ID	Fehltage
811	18.04.2008	23.04.2008	1	4
902	18.04.2008	09.05.2008	2	14

(5 Punkte)

3. Handlungsschritt (25 Punkte)

Korrekturrand

a) Das Mitarbeiterverwaltungssystem soll folgende Funktionalität bieten:

- Jeder Mitarbeiter/jede Mitarbeiterin kann sich den Status eines Lesegerätes anzeigen lassen.
- Jeder Mitarbeiter/jede Mitarbeiterin kann sich die Protokolldaten zu einem Lesegerät anzeigen lassen.
- Ein Administrator kann Lesegeräte anmelden und abmelden.
- Ein Administrator kann Zugangsberechtigungen an einem Lesegerät setzen.
- Ein Administrator kann einen Funktionscheck für ein Lesegerät durchführen. In diesem Fall wird stets der Status des Lesegerätes angezeigt.

Jeder Administrator ist ein Mitarbeiter/eine Mitarbeiterin.

Erstellen Sie ein Use-Case-Diagramm.

(8 Punkte)

b) Das Mitarbeiterverwaltungssystem speichert folgende Zugangsdaten chronologisch in einer Protokolldatei.

Datum;Zeit;Bereichs_ID;Mitarbeiter_ID;Erlaubnis;Zugang/Abgang

12.11.2008;07:45;B22;0798>true;Z

12.11.2008;08:11;B21;0811>true;Z

12.11.2008;08:15;B21;0019>true;Z

12.11.2008;09:46;B21;0902>false;Z

12.11.2008;09:47;B21;1221>true;Z

12.11.2008;11:17;B21;0811>true;A

Erstellen Sie die Prozedur *ErmittleMitarbeiterImBereich*(Bereich_ID: Integer), die anhand der Protokolldatei die IDs der Mitarbeiter/-innen in eine Liste ausgibt, die sich in einem bestimmten Bereich aufhalten. Die Bereichs_ID wird der Prozedur als Parameter übergeben.

Zur Lösung dieser Aufgabe können Sie Arrays beliebigen Typs verwenden, ohne eine Dimensionierung vorzugeben. Diese Arrays besitzen stets ausreichend Speicherplatz.

(17 Punkte)

Folgende Funktionen stehen Ihnen zur Verfügung:

Korrekturrand

Funktion	Beschreibung
<i>leseProtokollsatz</i>	Liest den nächsten Protokollsatz ein und speichert die durch Semikolon getrennten Informationen in einem Array (6 Stringelemente)
<i>schreibeInArray(Array,ArrayElement)</i>	Speichert das angegebene ArrayElement in das angegebene Array
<i>löscheAusArray(Array,ArrayElement)</i>	Löscht das angegebene ArrayElement aus dem angegebenen Array

4. Handlungsschritt (25 Punkte)

Korrekturrand

Die XSEC AG soll für die Schubert GmbH ein Zugangskontrollsystem erstellen.
Die dazu erforderliche Software soll objektorientiert programmiert werden.
Dazu sollen zunächst die Klassen *Lesegeraet* und *Datenauswerter* erstellt werden.

a) Die Objekte vom Typ *Lesegeraet* rufen Methoden der Objekte vom Typ *Datenauswerter* auf.

Die Klasse *Lesegeraet* soll so erstellt werden, dass die Klasse *Datenauswerter* durch eine andere Klasse mit erweiterter Funktionalität ersetzt werden kann, ohne dass die Klasse *Lesegeraet* verändert werden muss.

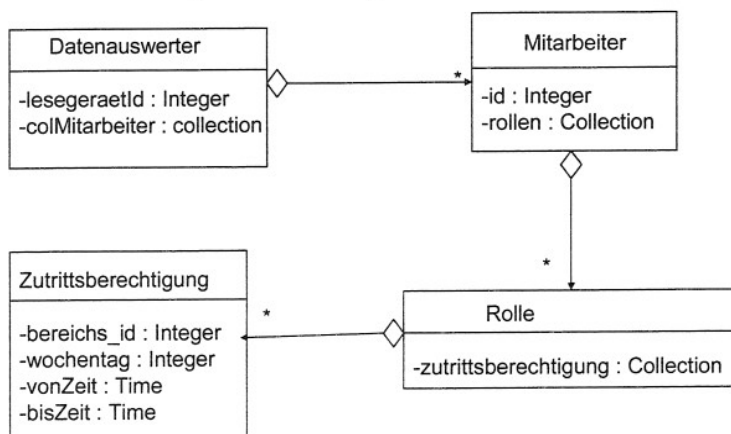
Erstellen Sie ein Klassendiagramm, das zeigt, wie dieser Anforderung unter Verwendung einer abstrakten Klasse oder eines Interface entsprochen werden kann. (6 Punkte)

b) Die XSEC AG hat die Zugangskontrolle wie folgt konzipiert:

Die Mitarbeiter/-innen der Schubert GmbH erhalten Zugangsberechtigungen über ein Rollenkonzept.

Ein Mitarbeiter hat eine oder mehrere Rollen. Für jede Rolle sind entsprechende Zugangsberechtigungen (Bereiche und Zutrittszeiten) festgelegt. Die Bereiche werden mit Lesegeräten versehen, an denen die Mitarbeiter/-innen ihre jeweilige ID eingeben.

Es wurde bereits folgendes Klassendiagramm erstellt.



Jede Collection besitzt u. a. die folgenden Methoden:

Methode	Beschreibung
<code>length()</code>	Liefert die Anzahl der Elemente der Collection
<code>get(index : Integer)</code>	Liefert die Objektreferenz des Elementes an der Position index

In jeder Klasse sind für jede Eigenschaft öffentliche get-Methoden vorhanden.

Erstellen Sie auf der Folgeseite für die Klasse *Datenauswerter* eine Methode *zutrittspruefung*, deren Rückgabewerte true oder false sind, je nachdem, ob der Zutritt gewährt wird oder nicht. Die Methode soll mit folgenden Parametern aufgerufen werden:

- MA_ID: Integer
- bereichs_ID: Integer
- wochentag: Integer
- uhrzeit: Time

(19 Punkte)

5. Handlungsschritt (25 Punkte)

Korrekturrand

Die XSEC AG hat für die Schubert GmbH ein System zur Arbeitsstundenverwaltung erstellt (zugrunde liegende Datenbank siehe Anlage zum 2. und 5. Handlungsschritt).

Die Prozedur *Soll_Ist_Vergleich* führt für alle Mitarbeiter für den übergebenen Monat eines Jahres einen Soll-/Ist-Abgleich der Arbeitszeit durch und gibt folgende Daten aus: MA_ID, Zeitdifferenz in Stunden und Minuten.

Beispiel der Ausgabe:

MA_ID	Std	Min
811	4	26
902	-2	45
...

Pseudocode:

```
Soll_Ist_Vergleich(Monat : Integer, Jahr : Integer)
```

```
  MonatsArbeitstage := Hole_ArbeitstageMonat(Monat, Jahr)
```

```
  MA_Array := Hole_Mitarbeiter()
```

```
  Für i := 0 bis Länge von MA_Array - 1
```

```
    // Fehltage ermitteln
```

```
    Fehltage := Hole_Fehltage(MA_Array[i].ID, Monat, Jahr)
```

```
    // SOLL-Arbeitszeit
```

```
    SollMinuten := (MonatsArbeitstage - Fehltage) * MA_Array[i].Tagesarbeitszeit * 60
```

```
    // IST-Arbeitszeit
```

```
    Stunden_Array := Hole_KGB(MA_Array[i].ID, Monat, Jahr)
```

```
    ISTMinuten := 0
```

```
    Für j := 0 bis Länge von Stunden_Array - 1
```

```
      ISTMinuten := ISTMinuten + _  
                      Zeitdifferenz(Stunden_Array[j].Kommen_Zeit, _  
                      Stunden_Array[j].Gehen_Zeit)
```

```
    Ende j
```

```
    // Differenz berechnen
```

```
    DiffMinuten := ISTMinuten - SOLLMinuten
```

```
    Std := DiffMinuten DIV 60
```

```
    Min := DiffMinuten MOD 60
```

```
    // Ausgabe
```

```
    Schreibe(MA_Array[i].ID, Std, Min)
```

```
  Ende i
```

Verwendete Funktionen:

Funktion	Beschreibung
<i>Hole_Arbeitstage</i> (Monat: Integer, Jahr: Integer): Integer	Liefert die Anzahl der Arbeitstage für den übergebenen Monat/Jahr
<i>Hole_Mitarbeiter</i> (): Array von Mitarbeiter	Liefert alle Mitarbeiter aus Tabelle Mitarbeiter in einem Array von gleicher Struktur
<i>Hole_Fehltage</i> (MA_ID: Integer, Monat: Integer, Jahr: Integer): Integer	Liefert die Anzahl der Fehltage für den angegebenen Mitarbeiter im übergebenen Monat/Jahr
<i>Hole_KGB</i> (MA_ID: Integer, Monat: Integer, Jahr: Integer): Array von KGB	Liefert alle Datensätze aus der Tabelle KommenGehenBuchung in einem Array gleicher Struktur
<i>Zeitdifferenz</i> (Kommen_Zeit: String, Gehen_Zeit: String): Integer	Liefert die Differenz zwischen den angegebenen Zeiten in Minuten
<i>Schreibe</i> (MA_ID: Integer, Std: Integer, Min: Integer)	Gibt die übergebenen Parameter aus

a) Stellen Sie das gegebene Programm in einem Struktogramm (DIN 66261) oder Programmablaufplan (DIN 66001) dar.

(19 Punkte)

Korrekturrand

b) Nach einem Testlauf wird bemängelt, dass zu den Fehltagen auch die Fehltage mit dem Fehlgrund Dienstreise addiert werden, was sachlich nicht richtig ist. Das Programm soll nun so abgeändert werden, dass Fehltage mit dem Fehlzeitgrund Dienstreise ausgeschlossen werden.

Beschreiben Sie kurz zwei Möglichkeiten zur Korrektur des Fehlers.

(6 Punkte)

bitte wenden!