**3. Aufgabenblatt**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Vorname** | **Nachname** | **Matrikelnummer** | | **Fachrichtung** |
| Marius | Grumer | 63284 | | MABB |
| Philipp | Huber | 63326 | | MABB |
| **Punkte** | | | xx/yy | | |
| **Bestanden** | | | ja/nein | | |

*1. Aufgabe:*

Um den kontrollierten Zugriff auf die Datenelemente einer Klasse (Attribute) sicherzustellen, werden die Datenelemente normalerweise als private deklariert. Der lesende und schreibende Zugriff auf die Datenelemente erfolgt ausschließlich durch ............................

**Lösung**: Methoden

*2. Aufgabe:*

Speicher für die Datenelemente wird angelegt, wenn

1. die Klasse definiert wird
2. die Header-Datei mit der Klassendefinition inkludiert wird
3. ein Objekt vom Typ der Klasse angelegt wird

**Lösung**: c

***Erklärung:*** *Erst bei einer Initialisierung eines Objektes wird Speicher angelegt*

*3. Aufgabe:*

Gegeben sei die folgende Klassendefinition:

**class Test**

**{**

**private:**

**int count;**

**public:**

**void setCount(int n);** // n an count zuweisen

};

Für ein Objekt myTest der Klasse Test ist folgende Anweisung zulässig:

1. myTest.count = 7;
2. myTest::count = 7;
3. myTest.setCount(7);

**Lösung**: c

***Erklärung****: In der Main.cpp wird das Objekt „myTest“ initialisiert. Auf count kann auf Grund des Private Status in der main nicht zugegriffen werden. Die einzig zulässige Möglichkeit für ein Ändern von count stellt Lösung c dar.*

*4. Aufgabe:*

Gegeben sei die folgende Definition:

**struct Time**

**{**

**unsigned short hour, minute, second;**

**};**

Ein Objekt ***alarm*** vom Typ ***Time***, das mit 9:30:00 Uhr initialisiert wird, kann wie folgt definiert werden: ..........................................................

**Lösung**:

Time alarm;

alarm.hour = 9;

alarm.minute = 30;

alarm.second =0;

*5. Aufgabe:*

Die Größe eines Objektes vom Typ Time entspricht:

1. der Größe des Datenelementes unsigned short
2. der Summe der 3 Datenelemente vom Typ unsigned short

**Lösung: *b***

***Erklärung****: Das Struct enthält 3 Datenelemente vom Typ short. Deshalb ist die Größe des Objektes 3\*Datenelemente vom Typ unsigned short.*

***3\*2bit=6bit***

Cout<<sizeof(alarm);

*6. Aufgabe:*

Die Voreinstellung für den Zugriff auf Elemente einer mit ***struct*** definierten Klasse (oder benutzerdefinierten Datentyps) ist .........................

**Lösung**: Zugriff per Punkt

***Erklärung****: Mittels eines Punktes kann auf die Elemente zugegriffen werden (vgl. Aufgabe 4)*

*7. Aufgabe:*

Welche Fehler können Sie in den folgenden Klassendefinitionen erkennen?

**class A**

**{**

**numerator, denominator;**

**private:**

**void convert(double);**

**long gcd(void);**

**}**

**class B**

**{**

**private:**

**long limit = 1000L;**

**double x = 0.0;**

**public:**

**void save();**

**};**

**Lösung*:***



**class A**

**{**

**Private:**

**int numerator, denominator;**

**public:**

**void convert(double n);**

**void gcd(long x);**

**};**

Methoden in public gesetzt, da diese in private keinen Sinn machen.



**class B**

**{**

**private:**

**long limit = 1000 ;**

**double x = 0.0;**

**public:**

**void save();**

**};**

Keine Doppeldeklaration des longs nötig.

*8. Aufgabe:*

Bei der Zuweisung von Objekten einer Klasse

1. wird die Adresse des Quellobjektes in das Zielobjekt kopiert
2. werden die Datenelemente des Quellobjektes in die entsprechenden Datenelemente des Zielobjektes kopiert
3. wird der Speicherbereich des Quellobjektes Byte für Byte in den Speicherbereich des Zielobjektes kopiert

**Lösung: b**

***Erklärung***: Bei einer Zuweisung wird ein Copy Ctor ausgeführt und die einzelnen Attribute kopiert