

Sommersemester 2010		Blatt Nr. 1 von 12	
Fachbereich:	Informationstechnik	Semester:	IT3A/IT3B
Prüfungsfach:	Informatik 3	Fachnummer:	KTB/SWB/TIB 3011
Hilfsmittel:	Alle, bis auf Rechner	Zeit:	90 min
Name:	Vorname:	Matrikelnummer:	

Benutzen Sie in Ihren Lösungen ausschließlich die in der Aufgabenstellung angegebenen Namen für die Variablen, die Methoden, etc. Schreiben Sie wenn erforderlich kurze Erklärungen in Stichworten zu Ihren Lösungen. Verwenden Sie bitte keine roten Farbstifte. Benutzen Sie für Ihre Lösungen den hier vorgesehenen Platz. Schreiben Sie daher auf jedes Blatt Ihren Namen und Ihre Matrikelnummer.

Aufgabe 1 Wahr (w) oder falsch (f)?

15 min

Bitte kreuzen Sie die richtige Antwort an und begründen Sie in Stichworten:

	Aussage	w	f
1.	<i>Wird beim Aufruf einer Methode mit Defaultparameter ein Parameter weggelassen, dann dürfen keine weiteren Parameter weggelassen werden.</i> Begründung:		
2.	<i>Durch die public-Vererbung werden private-Attribute der Basisklasse in der abgeleiteten Klasse direkt benutzbar.</i> Begründung:		
3.	<i>Die Funktion funk(void) wird durch ihren Rückgabewert überladen, z.B. int funk(void) und bool funk(void). Korrekt?</i> Begründung:		
4.	<i>In nicht allen der drei Fällen wird const sinnvoll eingesetzt: int test (const int& r, const int* p, const int v);</i> Begründung:		

Sommersemester 2010		Blatt Nr. 2 von 12	
Fachbereich:	Informationstechnik	Semester:	IT3A/IT3B
Prüfungsfach:	Informatik 3	Fachnummer:	KTB/SWB/TIB 3011
Hilfsmittel:	Alle, bis auf Rechner	Zeit:	90 min
Name:	Vorname:	Matrikelnummer:	

5.	<p><i>Die Selbstzuweisung wird mit Hilfe des this-Zeigers festgestellt.</i></p> <p>Begründung:</p>		
6.	<p><i>Beim Ablauf eines Programms kann es folgende Sequenz geben:</i></p> <p>try .. try .. throw.. catch.. throw.. catch.</p> <p>Begründung:</p>		
7.	<p><i>Der unärer Operator ++ wird durch eine Methode einer Klasse überladen. In der Parameterliste wird kein Operand angegeben.</i></p> <p>Begründung:</p>		
8.	<p><i>In einer Klasse wird der Konstruktor und der Destruktor mit virtual gebildet.</i></p> <p>Begründung:</p>		

Sommersemester 2010		Blatt Nr. 3 von 12	
Fachbereich:	Informationstechnik	Semester:	IT3A/IT3B
Prüfungsfach:	Informatik 3	Fachnummer:	KTB/SWB/TIB 3011
Hilfsmittel:	Alle, bis auf Rechner	Zeit:	90 min
Name:	Vorname:	Matrikelnummer:	

9.	<p><i>Die Basisklasse enthält die Methode <code>virtual int funk(int a)=0;</code> Die abgeleitete Klasse enthält die Methode <code>int funk(int a, char b);</code> Von der abgeleiteten Klasse kann daher keine Instanz gebildet werden.</i></p> <p>Begründung:</p>		
10.	<p><i>Zur Initialisierung von <code>static</code>-Attributen in einer Klasse wird der Konstruktor verwendet wie zum Beispiel:</i></p> <pre>class A {static int x; A():x(1){ } };</pre> <p>Begründung:</p>		

Sommersemester 2010		Blatt Nr. 4 von 12	
Fachbereich:	Informationstechnik	Semester:	IT3A/IT3B
Prüfungsfach:	Informatik 3	Fachnummer:	KTB/SWB/TIB 3011
Hilfsmittel:	Alle, bis auf Rechner	Zeit:	90 min
Name:	Vorname:	Matrikelnummer:	

Aufgabe 2 Klassenmodellierung

35 min

Bei dieser Aufgabe geht es um eine Klasse **Lawamat**, mit der Waschautomaten beschrieben werden. Die Waschautomaten werden in Waschsalongen aufgestellt, somit gibt es auch die Situation, dass zwei oder mehr Waschmaschinen vom Typ **Lawamat** sich in einem Raum befinden (siehe Spezifikation 7.)

Zunächst wird eine **Spezifikation** vorgestellt, mit der die Funktionsweise der Automaten beschrieben wird. Anschließend sollen Sie die Attribute und Methoden der Klasse programmieren.

Spezifikation

1. Wenn ein Waschautomat (Instanz der Klasse **Lawamat**) erzeugt wird, wird er mit einer Seriennummer initialisiert. Die Seriennummer besteht aus 6 Dezimalziffern, wird als Stringkonstante (z.B. "654321") für die Initialisierung einer Instanz übergeben und in einem Feld innerhalb der Instanz gespeichert. Neben der Seriennummer wird das Herstellungsdatum in einer Waschmaschine gespeichert. Nehmen Sie für das Datum eine eigene Klasse, deren Instanzen Jahr, Monat und Tag aufnehmen können.
2. Über das Klasseninterface (Methode **werbistdu**) kann die Seriennummer abgefragt werden.
3. Über das Klasseninterface kann der Waschautomat eingeschaltet werden (Methode **einschalten**). Damit werden sämtliche Funktionen des Waschautomaten freigegeben. Freigeben bedeutet, dass sämtliche Methodenaufrufe Wirkung zeigen.
4. Über das Klasseninterface kann der Waschautomat ausgeschaltet werden (Methode **ausschalten**). Dadurch wird die weitere Bedienung des Waschautomaten gesperrt. Sperren bedeutet dabei, dass Methodenaufrufe (außer der Methode **einschalten**) keine Wirkung zeigen.
5. Über den Aufruf der Methode **auswahl** wird dem Objekt mitgeteilt, welche Art von Waschgang gewünscht wird. Mögliche Einstellungen sind: Buntwäsche, Wolle, Schleudern, Nichtschleudern, Menge der Wäsche. Stellen Sie die dazu erforderlichen Klassenattribute auf.
6. Über die Methode **memory** können die aktuellen Einstellungen des Waschautomaten im Heap gespeichert werden. Pro Instanz ist die Speicherung von nur einem Profil möglich (Profil = gespeicherte Einstellungen).
7. Über einen Datentransfer zwischen zwei bestehenden Waschautomaten können die aktuellen Einstellungen (nicht das Profil!) einer Waschmaschine in eine andere übergeben werden. Wie modellieren Sie diese Tatsache innerhalb der Klasse **Lawamat**?

Sommersemester 2010		Blatt Nr. 5 von 12	
Fachbereich:	Informationstechnik	Semester:	IT3A/IT3B
Prüfungsfach:	Informatik 3	Fachnummer:	KTB/SWB/TIB 3011
Hilfsmittel:	Alle, bis auf Rechner	Zeit:	90 min
Name:	Vorname:	Matrikelnummer:	

8. Es kommt vor, dass eine Waschmaschine verschrottet wird. D.h. es sind weniger Waschmaschinen im Einsatz als produziert wurden. Die Anzahl der aktuellen Waschmaschinen soll durch die Funktion `statistik()` ausgegeben werden können.

9. Integrieren Sie in die Klasse `Lawamat` auch einen geeigneten Destruktor.

Aufgabenstellung

a) Deklarieren Sie die Klasse `Lawamat` mit den erforderlichen Attributen sowie den Prototypen der Methoden einschließlich Konstruktoren und Destruktor. Geben Sie hier außerdem die Deklaration von eventuell erforderlichen eigenen Datentypen an.

b) Definieren Sie die geforderten Methoden außerhalb der Klasse.

Lösung a)

Sommersemester 2010		Blatt Nr. 6 von 12	
Fachbereich:	Informationstechnik	Semester:	IT3A/IT3B
Prüfungsfach:	Informatik 3	Fachnummer:	KTB/SWB/TIB 3011
Hilfsmittel:	Alle, bis auf Rechner	Zeit:	90 min
Name:	Vorname:	Matrikelnummer:	

Sommersemester 2010		Blatt Nr. 7 von 12	
Fachbereich:	Informationstechnik	Semester:	IT3A/IT3B
Prüfungsfach:	Informatik 3	Fachnummer:	KTB/SWB/TIB 3011
Hilfsmittel:	Alle, bis auf Rechner	Zeit:	90 min
Name:	Vorname:	Matrikelnummer:	

Lösung b

Sommersemester 2010	Blatt Nr. 9 von 12
Fachbereich: Informationstechnik	Semester: IT3A/IT3B
Prüfungsfach: Informatik 3	Fachnummer: KTB/SWB/TIB 3011
Hilfsmittel: Alle, bis auf Rechner	Zeit: 90 min
Name: Vorname:	Matrikelnummer:

```
};
```

```
void testeZeit() {  
    Zeit z1(5, 35);  
    Zeit z2(2, 45);  
  
    z1 += z2;  
  
    Zeit z3 = z1 + z2;  
    z3.print();  
}
```

Sommersemester 2010		Blatt Nr. 10 von 12	
Fachbereich:	Informationstechnik	Semester:	IT3A/IT3B
Prüfungsfach:	Informatik 3	Fachnummer:	KTB/SWB/TIB 3011
Hilfsmittel:	Alle, bis auf Rechner	Zeit:	90 min
Name:	Vorname:	Matrikelnummer:	

Aufgabe 4 Vererbung

20 min

Gegeben sei eine Klasse **AntriebAbstract**, die eine Methode **antreiben** bereitstellt, die aus dem aktuellen Gesamtgewicht eines Schiffes die jeweilige Geschwindigkeit (km/h) errechnet und zurückgibt.

```
class AntriebAbstract {
public:
    float antreiben();
};
```

Sowie zwei Funktionen:

```
float fahren(const AntriebAbstract& a) {
    return a.antreiben();
}

float segelnPlus(const Segel* s) {
    return s->antreiben() + s->hilfsmotor();
}
```

4.1 Abstrakte Basisklasse

Verändern Sie die gegebene Klasse **AntriebAbstract** so, dass eine abstrakte Basisklasse entsteht.

```
class AntriebAbstract {
public:
    float antreiben();
};
```

Sommersemester 2010		Blatt Nr. 11 von 12	
Fachbereich:	Informationstechnik	Semester:	IT3A/IT3B
Prüfungsfach:	Informatik 3	Fachnummer:	KTB/SWB/TIB 3011
Hilfsmittel:	Alle, bis auf Rechner	Zeit:	90 min
Name:	Vorname:	Matrikelnummer:	

4.2 Ableitung

Leiten Sie von der abstrakten Basisklasse **AntriebAbstract** die konkreten Klassen **segel** und **Motor** ab. Die Methode **antreiben** gibt nur eine konstante Geschwindigkeit zurück (Segel: 5.0f, Motor: 10.0f). Die Klasse **segel** erhält eine zusätzliche Methode **hilfsmotor**, die ebenfalls nur eine konstante Geschwindigkeit (2.0f) zurück gibt.

Sommersemester 2010		Blatt Nr. 12 von 12	
Fachbereich:	Informationstechnik	Semester:	IT3A/IT3B
Prüfungsfach:	Informatik 3	Fachnummer:	KTB/SWB/TIB 3011
Hilfsmittel:	Alle, bis auf Rechner	Zeit:	90 min
Name:	Vorname:	Matrikelnummer:	

4.3 Testprogramm

Folgendes – zunächst fehlerbehaftete - Programm soll verwendet werden, um die Klassen **AntriebAbstract**, **Segel** und **Motor** zu testen. Entfernen bzw. verbessern Sie die Fehler des Testprogramms so, dass kompiliert und gelinkt werden kann, das Programm ohne Absturz läuft und möglichst viele Anzeigen (**cout**) ausgegeben werden.

Die gegebenen Funktionen **fahren** und **segelnPlus** sollen dabei nicht verändert werden; passen Sie ggfs. Ihre Klassen an.

```

const int MAXANTRIEB = 2

void testeAntrieb() {
    AntriebAbstract* a[MAXANTRIEB];
    a[0] = new AntriebAbstract;
    cout <<"Der Antrieb fährt " << fahren(a[0]) <<"km." <<endl;

    Motor m;
    cout <<"Der Motor fährt " <<fahren(m) <<"km." <<endl;

    a[0] = new Motor;
    cout <<"Der Motor fährt " <<fahren(a[0]) <<"km." <<endl;

    a[1] = new Segel;
    cout <<"Das Segel segelt " <<segelnPlus(a[1])
        <<"km." <<endl;

}

```