

## 2. Test aus Datenverarbeitung für Physiker

### Dienstag, 19.1.2010, 09:00-10:30

- Name:
- Matrikelnummer:
- Gruppe:

Vorbereitung: Loggen Sie sich bitte unter `edv1dip##a` ein, wobei `##` die Gruppennummer bedeutet. Das Passwort steht an der Tafel. Damit können die bisher durchgeführten Übungen im Home-Verzeichnis der Gruppe gesehen werden, aber nicht die Verzeichnisse der anderen Testteilnehmer.

#### 1. Ein-/Ausgabe

a) Wie öffnet man in C++ ein Datenfile mit Namen `out.dat` zur Datenausgabe?  
(2 Punkte)

b) Wie schreibt man eine Dezimalzahl in dieses File?  
(1 Punkt)

c) Wie wird das File geschlossen?  
(1 Punkt)

2. Was bedeutet `using namespace std;`?  
(2 Punkte)

3. Gegeben seien die Klasse namens `Klasse`, zwei Instanzen (`a` und `b`) dieser Klasse, sowie die Funktionen `int func1(Klasse)` und `double func2(Klasse &)`.

a) Woran erkennt man den (die) Konstruktor(en) dieser Klasse? (2 Punkte)

b) In welcher/welchen der folgenden Zeilen wird der Kopierkonstruktor aufgerufen? (2 Punkte)

- `a=b;`
- `Klasse c=a;`
- `Klasse d(b);`
- `Klasse e;`
- `func1(a);`
- `func2(b);`

c) Für welche Klassen muss man den Kopierkonstruktor selbst schreiben?(2 Punkte)

4. Überladen von Funktionen:

a) Welche der folgenden Deklarationen sind im selben Namensraum zueinander nicht kompatibel?

(1 Punkt)

(1) `double func(int i);`

(2) `double func(int* i);`

(3) `double func(int &j);`

b) Was (Gleitkommazahl, ganze Zahl, nichts, Adresse, etc.) wird bei den zu den Deklarationen (1-3) gehörigen Funktionsaufrufen in `i` bzw. `j` kopiert? (2 Punkte)

5. Gegeben sei eine Klasse `Klasse1`.

Eine weitere Klassendefinition beginnt mit:

```
class Klasse2 : public Klasse1
```

In welcher Beziehung stehen die Klassen zueinander? Wie wird die Klasse1 in diesem Zusammenhang bezeichnet? (2 Punkte)

6. Entwickeln Sie eine Klasse, welche Zeitintervalle (auf Millisekunden genau) darstellt. Sowohl positive als auch negative Zeitintervalle (Maximum  $\pm 500$  h  $\hat{=}$  Datentyp integer) sollen in der üblichen Notation dargestellt werden können.

Implementieren Sie folgende Methoden:

a) Zwei Konstruktoren:

- Mit 3 Argumenten: `double sec, int min, int std`
- Mit 1 Argument: `int t` in [ms]

b) Den Operator `<<`

```
ostream& operator<<(ostream& out, const Zeit z);
```

Ausgabe im Format hh:mm:ss.sss (ms)

c) Die Operatoren `+` und `-`: Addition und Subtraktion von Zeitintervallen

Testen Sie die Klasse mit einigen Beispielen:

12:01:1.000 - 13:59:59.999

00:00:30.500 + 00:00:58.999

Programmname:

(18 Punkte)