Hochschule Furtwangen
Fakultät Informatik
Prof. Dr. Lothar Piepmeyer

Praktikum ‚Objektorientierte Programmierung'

Aufgabenblatt 7

In den folgenden Aufgaben werden Sie Operatoren überladen. Achten Sie bei Ihren Implementierungen auf eine möglichst *redundanzarme* Definition. Zu allen Teilaufgaben gehören umfassende Tests.

Aufgabe 1:
Machen Sie sich mit der folgenden Klasse vertraut:

```
class Vector3D {
private:
  float x, y, z;
public:
  Vector3D(float a, float b, float c) {
     x = a; y = b; z = c;
  }
};
```

a. Ändern Sie die Klasse so, dass ihre drei Attribute unveränderbar sind. Die Parameter des Konstruktors sollen in `x`, `y` und `z` umbenannt werden.

b. Ist es nötig den Kopierkonstruktor, den Destruktor, den Zuweisungsoperator oder einen Vergleichsoperator selbst zu definieren? Begründen Sie Ihre Meinung! Führen Sie gegebenenfalls die Implementierungen mit Tests durch.

c. Überladen Sie die Operatoren `+`, `-` und << in der Klasse `Vector3D` analog zum Beispiel der Klasse `Point` aus der Vorlesung. Testen Sie!

d. Definieren Sie die skalare Multiplikation zwischen einer Zahl vom Typ `float` und einem Objekt vom Typ `Vector3D`. Beachten Sie die in der Vorlesung vermittelten Prinzipien.

e. Definieren Sie die Multiplikation zweier Objekte vom Typ `Vector3D` sinnvoll. Hinweis: Das Ergebnis hat den Typ `float`.

Aufgabe 2:
In den vorherhergehenden Aufgabenblättern haben Sie die Klasse `Friends` implementiert.

a. Überladen Sie den Operator << sinnvoll. Das folgende Code-Fragment zeigt Ihnen, wie Sie die Ausgabe in ein Objekt vom Datentyp `std::string` umlenken können:

```
std::ostringstream out;
out << "Hello World" << std::endl;
std::string text = out.str();
```

Nutzen Sie diesen Hinweis, um Ihre Implementierung von << zu testen.

b. Warum ist es erforderlich den Zuweisungsoperator zu überladen? Überladen Sie den Operator. Beachten Sie die in der Vorlesung vermittelten Prinzipien. Testen Sie Ihre Implementierung des Zuweisungsoperators.

c. Überladen Sie den Indexoperator `[]` sinnvoll. Testen Sie Ihre Implementierung des Indexoperators.