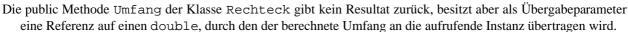
Übungen zu Klassen

1 Klasse Rechteck

Gegeben ist das untenstehende Programm, bestehend aus einem vordefinierten Hauptprogramm und einer vordefinierten Klasse Punkt. Das Programm soll, wie aus der main-Routine ersichtlich ist, den Umfang und die Fläche eines Rechtecks unter Anwendung der Klasse Rechteck berechnen. Schreiben Sie die hierzu erforderliche Klasse Rechteck unter folgenden Annahmen:

Die Klasse Rechteck enthält zwei private Elemente P1 und P2 vom Typ Punkt (= Eckpunkte des Rechtecks). Der Konstruktor der Klasse hat als Übergabe zwei Parameter vom Typ Punkt.

Die public Methode Flaeche der Klasse Rechteck besitzt keine Übergabeparameter und gibt einen double-Wert, die Fläche des Rechteckes, zurück.





```
P2(x,y)
#include <iostream>
#include <cmath>
class Punkt {
 private:
                                        P1(x,y)
           double x, y;
 public:
           Punkt(double X=0, double Y=0)
                                                  x=X; y=Y;
                                                              //Konstruktor
                 set (double X , double Y)
                                                  x=X; y=Y;
           double getX ()
                                                  return x;
                                                            } //
           double getY ()
                                                 return y; } //
};
class Rechteck {
```

```
void main() {
    // Zwei Variablen vom Typ "Punkt" zur Eingabe der Rechteckkoordinanten
    Punkt A, B;

    // Eingabe der Koordinaten der Punkte A und B
    double x,y;
    cout << "Koordinaten x und y Punkt A: "; cin >> x >> y; A.set(x,y);
    cout << "Koordinaten x und y Punkt B: "; cin >> x >> y; B.set(x,y);

    // Erzeugen eines Objekts der Klasse "Rechteck" und
    // Berechnung des Umfangs und der Flaeche des Rechteckes
    Rechteck RE( A, B);
    cout << "Die Rechteckflaeche betraegt " << RE.Flaeche() << endl;

    double Umfang;
    RE.Umfang(Umfang);
    cout << "Der Umfang des Rechteckes betraegt " << Umfang << endl;
}</pre>
```

30.01.2007 / AV 1 / 4

2 Klasse Semester

};

Eine Klasse Semester soll angeben, welche Studenten die Prüfungen bestanden haben, und den Notendurchschnitt aller Studenten berechnen.

Die Klasse Semester mit folgenden Eigenschaften ist zu schreiben:

- Die Studenten sind in einer privaten Variable studenten vom Typ vector<Student> zu speichern. Der Typ Student ist unten angegeben.
- Eine öffentliche Methode studentHinzufuegen(Student student) fügt einen Studenten als neues Element zum privaten Vektor studenten hinzu.
- Die öffentliche Methode bestanden() gibt die Vornamen und Namen aller Studenten auf den Bildschirm aus, die einen Notendurchschnitt grösser oder gleich 4 haben.
- Die öffentliche Methode gesamtDurchschnitt(), gibt den Gesamtnotendurchschnitt aller Studenten als double zurück.



```
const int AnzPruefungen = 10;
                                           //Anzahl der Prüfungen
struct Student{
                                           //Klasse für einen Studenten
    string name, vorname;
                                           //Name und Vorname
    double noten[AnzPruefungen];
                                           //In den Prüfungen erhaltene Noten
    double durchschnitt(){
                                           //Berechnet den Notendurchschnitt
        double durchschnitt=0.;
        for(int i=0;i< AnzPruefungen;i++) durchschnitt+=noten[i];</pre>
        return durchschnitt/AnzPruefungen;
    }
};
class Semester{
```

30.01.2007 / AV 2 / 4

3 SIMcards für Mobiltelefone

Schreiben Sie die Klasse SIMcard mit den nachfolgend beschriebenen Eigenschaften. Verwenden Sie dabei den unten gegebenen Typ Eintrag, in dem jeweils ein Name mit einer Telefonnummer gespeichert wird.

- Die Klasse hat als private Elemente ein Telefonbuch vom Typ vector<Eintrag> und die PIN (Sperrcode) als int-Zahl.
- Der Konstruktor besitzt einen Referenzparameter vom Datentyp int, in dem die PIN der SIMcard gespeichert wird.
- Die öffentliche Methode trageEin vom Typ bool trägt in das Telefonbuch einen neuen Eintrag ein. Übergabeparameter sind ein Name, eine Telefonnummer sowie die PIN. Bei Angabe eine falschen PIN wird kein Eintrag vorgenommen und die Methode gibt den Wert false zurück, ansonsten true.



• Die öffentliche Methode sucheNummer vom Typ int sucht die zu einem Namen gehörige Telefonnummer im Telefonbuch. Die Suche wird nur durchgeführt, wenn die übergebene PIN derjenigen der SIMcard entspricht. Die Methode gibt den Wert 0 zurück, wenn kein Eintrag vorhanden ist und -1, wenn die übergebene PIN falsch ist.

```
struct Eintrag{
   string name;
   int nummer;
}
```

30.01.2007 / AV 3 / 4

4 Verpackungsanlage

In einem Versandbetrieb soll zum Verpacken einer Ware vom Typ Quader geprüft werden, ob diese in einen Karton passt, ebenfalls vom Typ Quader. Ergänzen Sie die Klasse Quader wie folgt: Die Methode SortiereAbmessungen sortiert die Quaderabmessungen d[0],d[1] und d[2] der Grösse nach. Die Methode PasstIn gibt den Wert true zurück, wenn die Ware in den Karton hineinpasst, andernfalls false.

```
WSL
```

```
};
```

30.01.2007 / AV 4 / 4