

## Objekte / Programmieren 2

### Aufgabe p02c (Bewertungs-Praktikum)

Lernziel: Klassendefinition, Aggregation, Attribute und Methode und Codewiederverwendung

- Gehen Sie systematisch vor, lesen Sie die Aufgabe zunächst vollständig und genau durch, machen Sie nur das was gefordert wird.
- Achten Sie darauf, dass sich das Programm übersetzen lässt in MS Visual Studio.
- Laden Sie Ihre Lösung RECHTZEITIG hoch, laden Sie Zwischenversionen hoch.



Quelle: [www.bhu.de](http://www.bhu.de)

Für ein Baum-Kataster wird ein Verwaltungsprogramm gebraucht:

Aufgabe:

Das Programm soll die Daten von max. 1000 Bäumen verwalten. Pro Baum werden **Baumart, Höhe und geographische Position** verwaltet. Die geographische Position ist durch die geographische Breiten- und Längenangabe als Gleitkommazahlen festgehalten.

Eine Besonderheit ist, dass das Programm bereits bei der Datenerfassung prüfen muss, ob eine Überschreitung der Höhe des Baumes vorliegt. Diese richtet sich nach dem Breitengrad, an dem der Baum steht. Bäume, die oberhalb oder unterhalb des 60. Breitengrads stehen, erreichen eine maximale Höhe von 6,5 Metern. Bäume, die oberhalb oder unterhalb des 30. Breitengrads stehen, erreichen eine maximale Höhe von 36,5 Metern. Bäume, die zwischen dem Äquator und oberhalb oder unterhalb des 30. Breitengrads stehen, erreichen eine maximale Höhe von 66,5 Metern.

Definieren Sie eine Klasse „cGeoPos“ für die geografische Position

- Mit den privaten Attributen für die Längen- und Breitenangaben (Longitude, Latitude) als Gleitkommazahlen.
- Erstellen Sie einen universellen Konstruktor, der die geografische Position über Vorgabewerte auf 0,0 (Grad Breite) und 0,0 (Grad Länge) initialisiert.
- Erstellen Sie eine öffentliche Funktion setGeoPos(), mit der die Werte gesetzt werden. Beachten Sie dabei, dass die Längen- und Breitenangaben nie größer als 180,0 Grad sein können und nie kleiner als -180,0 sein können. Rechnen Sie nicht passende Werte um. Ein Kreis hat 360,0 Grad.
- Erstellen Sie eine öffentliche Methode printGeoPos(), mit der die Werte ausgegeben werden.

Definieren Sie eine Klasse „cBaum“ mit den privaten Datenelementen

- String für die Angabe der Baumart
- Gleitkommazahl für die Höhe des Baums
- Daten für die geografische Position, aggregieren Sie dazu die Klasse „cGeoPos“, indem Sie ein Objekt vom Typ cGeoPos als Attribut aufnehmen.
- Programmieren Sie einen universellen Konstruktor für diese Klasse:
  - Vorgabewert für die Baumart ist die Zeichenkette “-“
  - Vorgabewert für die Höhe ist 10,0
  - Vorgabewerte für die geografische Position sind 48,79 (Grad Breite) und 8,17 (Grad Länge)
  - Der Konstruktor muss die Höhenkorrektur durchführen!
  - Erstellen Sie die innere Konstruktorenkaskade für das Datenelement der Geoposition
- Die Klasse cBaum besitzt außerdem folgende öffentliche Mitgliedsfunktionen:
  - „eingabe() zur Eingabe der Werte der Attribute von der Tastatur
  - „ausgabe()“ zur Ausgabe aller Attribute am Bildschirm
- Die Klasse cBaum besitzt außerdem folgende **private** Mitgliedsfunktion:
  - „korrHoehe()“ zur Überprüfung und Korrektur der Höhe des Baums wie in der Aufgabenstellung beschrieben. Überlegen Sie sich zunächst passende Typen für Übergabeparameter und Returncode, setzen Sie dann die Funktionalität um.
  - Überlegen Sie sich, an welchen Stellen Sie die Methode verwenden können.

Erstellen Sie in den beiden Klassen weitere Methoden, wenn Sie diese brauchen.

Schreiben Sie folgendes Hauptprogramm:

a) Definieren Sie 3 Objekte vom Typ cBaum wie folgt:

```
cBaum baum1("Birke", 111.8, 48.3, 8.72),  
      baum2("Baobap", 47.3, 62.76, 5.83), baum3;
```

und geben Sie die Werte dieser 3 Bäume aus.

b) Definieren Sie ein Array aus 1000 Objekten vom Typ cBaum.

Geben Sie Werte für die Objekte ein:

- Beginnend ab dem 1. Objekt wird in einer Schleife die eingabe()-Methode aufgerufen und es werden die Werte eingegeben.
- Sind die Werte eines Objektes fertig eingegeben, werden die eingegebenen Daten zur Kontrolle mit der Funktion ausgabe() angezeigt.
- Dann wird mit dem nächsten Objekt weiter gemacht.
- Sobald der Anwender als Gerätetyp ein Minus-Zeichen "-" eingibt, bricht die Eingabe-Schleife ab.

Das Programm soll dann eine Liste der Bäume ausgeben, in Form einer Tabelle wie folgt:

- Zunächst wird einmalig eine passende Überschrift ausgegeben.
- Dann so viele Zeilen, wie es Bäume gibt, für die Daten eingegeben wurden.

### **Bewertungsschema:**

**Programm hat Übersetzungsfehler in MS Visual Studio ==> 0 P**

sonst:

Kommentare und Einrücken gut	4 P
Klassen / Konstruktoren	4 P
Aggregation	4 P
Methodenimplementierungen	4 P
Hauptprogramm / Ein- / Ausgabeschleifen	4 P
	-----
Summe	20 P