

Hochschule Deggendorf Dr. Peter Jüttner	
Vorlesung: Objektorientierte Programmierung	SS 2019
Übung 2	Termin 28.3.19

Klassen

1. Aufgabe Klasse Complex

Schreiben Sie eine Klasse Complex, für komplexe Zahlen. Die Klasse soll zwei private Attribute für Real- und Imaginärteil vom Typ float haben.

Des weiteren soll die Klasse folgende Methoden besitzen:

- lesecomplex() die Real- und Imaginärteil eines Objekts einliest.
- ausgabe(), die das gerufene Objekt auf dem Bildschirm ausgibt.
- addiere (complex b), die zum Aufrufobjekt eine komplexe Zahl b addiert.
- subtrahiere (complex b), die vom Aufrufobjekt eine komplexe Zahl b subtrahiert.
- addiere_real(float f), die zum Realteil des gerufenen Objekts f addiert.

Probieren Sie die Klasse in einem Hauptprogramm aus.

2. Aufgabe Fuhrparkverwaltung

Erstellen Sie ein Objektorientiertes Programm zur Verwaltung eines Fuhrparks mit 10 Fahrzeugen.

Erstellen Sie dazu zunächst eine Klasse Fahrzeug mit folgenden Attributen:

- Hersteller und Typ als char Array der Länge 20
- Kennzeichen als char Array der Länge 11
- Kilometerstand vom Typ Long

Die Klasse Fahrzeug soll Methoden zum Schreiben der Attribute erhalten und eine Methode zur Ausgabe der Fahrzeugdaten auf Bildschirm.

Die Fuhrparkverwaltung soll als Klasse definiert werden mit folgenden Attributen:

- Fahrzeugbestand als Array von 10 Fahrzeugen
- aktuelle Anzahl der Fahrzeuge

Die Fuhrparkverwaltung soll folgende Methoden haben:

- Löschen eines Fahrzeugs aus dem Bestand (als Kennzeichen, dass das Fahrzeug gelöscht ist, soll der Kilometerstand des Fahrzeugs im Array auf -1 gesetzt werden):
loesche(int fzgnummer) /* das Fahrzeug mit der Nummer fzgnummer soll aus dem Bestand gelöscht werden */
Falls fzgnummer nicht im erlaubten Bereich ist oder das Fahrzeug schon gelöscht ist, sollen entsprechende Fehlermeldungen ausgegeben werden.

- Eintrag eines neuen Fahrzeugs auf die erste freie Position des Arrays. Falls keine Position mehr frei ist, soll eine Fehlernummer ausgegeben werden.
- Aktualisieren des Kilometerstands eines ausgewählten Fahrzeugs
`akt_km(int fzg_nummer, long km)`
- Ausgabe des gesamten Fahrzeugbestands inkl. aller Fahrzeugdaten
`printbestand()`

Ergänzen Sie bei Bedarf die Klasse Fahrzeug um weitere Methoden, die sie für die Klasse Fuhrparkverwaltung benötigen. Attribute der Klasse Fahrzeug sollen aber gekapselt bleiben.