Christoph Ott | Institut für Angewandte Informationsverarbeitung | 28. August 2008
Blatt 4

WiMa-Praktikum/ Programmier-Praktikum (SS 2008)
Submit-Übungsaufgaben

Submit-Frist bis: Freitag 29. August 2008 9:15 Uhr

8 Eine Klasse für die komplexen Zahlen

Schreibt ein Programm, das komplexe Zahlen addieren, multiplizieren und in schöner Form am Bildschirm ausgeben kann. Da Ihr mittlerweile den Begriff der Modularisierung kennt, sollt Ihr Euer Program in die Dateien *complex.h*, *complex.cpp* und *compute.cpp* aufsplitten. Folgende Anforderungen werden an Eure Klasse *Complex* gestellt:

- Eure Klasse soll Elementvariablen für den Realteil und den Imaginärteil einer komplexen Zahl besitzen.
- Eure Klasse soll zudem Elementfunktionen haben, die ein weiteres Element der Klasse *Complex* zu dem aktuellen hinzuaddieren (bzw. drauf-multiplizieren). Komplexe Zahlen werden nach folgenden Regeln addiert bzw. multipliziert:
  - Für die Addition zweier komplexer Zahlen $a + b \cdot i$ und $c + d \cdot i$ gilt:
  $$(a + b \cdot i) + (c + d \cdot i) = (a + c) + (b + d) \cdot i$$
  - Für die Multiplikation gilt entsprechend
  $$(a + b \cdot i) \cdot (c + d \cdot i) = (ac - bd) + (ad + bc) \cdot i$$
- Außerdem soll eine Elementfunktion *print* zur schönen Ausgabe einer Zahl in der Form $a + b \cdot i$ vorhanden sein. Ebenso wie in der Mathematik zeichnet sich ein schöne Ausgabe dadurch aus, dass:
  - ein Realteil von 0 nicht ausgegeben wird. In diesem Fall wird natürlich auch das nachfolgende + nicht ausgegeben.
  - im Falle eines negativen Imaginärteils das + nicht erscheint, sondern lediglich das $-$.
  - für einen Imaginärteil von 1 oder -1 die 1 vor dem i nicht ausgegeben wird.
  - für einen Imaginärteil von 0 außer dem Realteil gar nicht ausgegeben wird.
  - wenn Real- und Imaginärteil beide 0 betragen, lediglich die Zahl 0 ausgegeben wird.

Euer Hauptprogramm soll nun sowohl Realteil als auch Imaginärteil von drei komplexen Zahlen von der Standardeingabe einlesen und daraus drei Objekte der Klasse *Complex* erzeugen. Anschließend soll es die zweite komplexe Zahl zur ersten hinzuaddieren und am Bildschirm mittels *print* ausgeben. Auf das Ergebnis der Addition soll anschließend die dritte komplexe Zahl drauf-multipliziert werden. Auch dieses Ergebnis soll anschließend mittels *print* am Bildschirm ausgegeben werden. Am Bilschrim sieht das dann etwa wie folgt aus

```
1       ->Realteil 1.Zahl
1       ->Imaginärteil 1.Zahl
2       ->Realteil 2.Zahl
2       ->Imaginärteil 2.Zahl
3       ->Realteil 3.Zahl
3       ->Imaginärteil 3.Zahl
3+3i    ->Ergebnis nach Addition
18i      ->Ergebnis nach Multiplikation
```

Hinweis: In den Submit-Befehl müssen in diesem Fall alle einzureichenden Dateinamen aufgenommen werden, d. h.

```
submit wimaprak 8 complex.h complex.cpp compute.cpp
```

Viel Erfolg!