

## Programmieren II für Studierende der Mathematik

### Blatt 9

**Aufgabe 10** Vereinbaren Sie ein Template für eine Funktion `pNorm`. Templateparameter sollen ein Gleitpunktzahlen-Typ  $T$  (diese Eigenschaft muss hier und im Folgenden nicht überprüft werden) und eine positive ganze Zahl  $p$  sein. `pNorm` soll für einen Parameter vom Datentyp `complex` aus der Standardlibrary (für den gegebenen Gleitpunktzahlen-Typ) die  $p$ -Norm dieser komplexen Zahl berechnen und als Wert des gegebenen Gleitpunktzahlen-Typs zurückgeben.

*Hinweis.* Die  $p$ -Norm  $\|\cdot\|_p$  einer komplexen Zahl ist definiert als:

$$\|a + ib\|_p := (|a|^p + |b|^p)^{\frac{1}{p}}$$

Vereinbaren und implementieren Sie eine von `complex` abgeleitete Klasse `complex_norm`. Die Definition soll ein Template mit Parametern für den zu verwendeten Gleitpunktzahlentyp und eine Norm-Funktion sein. Es soll sich eine Instanz Ihres templates `pNorm` für den zweiten Parameter einsetzen lassen.

Die Klasse `complex_norm` soll eine konstante Methode `norm` haben, die die als template Parameter übergebenen Norm-Funktion aufruft.

Vereinbaren Sie unter Verwendung von `complex_norm` und `pNorm` einen parametrisierten Typalias `complex_pnorm`. Parameter sollen ein Gleitpunktzahlentyp  $T$  und eine positive ganze Zahl  $p$  sein.

Die  $p$ -Normen sind *äquivalent*. D.h. für jedes Paar  $1 < p_1 \leq p_2 \in \mathbb{N} \cup \{\infty\}$  existiert eine Konstante  $c \in \mathbb{R}$  sodass für alle  $x \in \mathbb{C}$  gilt:

$$\|x\|_{p_1} \leq c \|x\|_{p_2}$$

Es gilt:

$$c = \frac{\|1 + i\|_{p_1}}{\|1 + i\|_{p_2}}$$

Implementieren Sie ein Hauptprogramm, das zunächst  $c$  für  $p_1 = 2$  und  $p_2 = 3$  berechnet und mit einer gut an die Genauigkeit angepassten Anzahl von Nachkommastellen ausgibt. Danach sollen solange komplexe Zahlen  $z_k$  von der Standardeingabe eingelesen werden, solange dies nicht fehlschlägt und jeweils das Verhältnis  $c_k = \frac{\|z\|_2}{\|z\|_3}$  berechnet und ebenfalls ausgegeben werden. Es soll zudem ausgegeben werden ob  $c_k \leq c$  gilt.

*Beispiel* (Kontrollergebnis). Ihr Programm könnte sich verhalten, wie folgt:

```
1.122462048309373
real, imag = 1.5 2.0
1.111619628853093 true
real, imag = 7.0 7.0
1.122462048309373 true
real, imag = 3.7 42.1
1.003627499490993 true
real, imag =
```