



BERGISCHE
UNIVERSITÄT
WUPPERTAL

Prof. Dr. Hans-Jürgen Buhl
Praktische Informatik/Numerik

Fakultät für
Mathematik und Naturwissenschaften,
Mathematik und Informatik

E-MAIL buhl@math.uni-wuppertal.de

WWW www.math.uni-wuppertal.de/~buhl

DATUM 8. November 2017

generische Programmierung

WS 2017/2018 – Übungsblatt 3

Ausgabe: 9. November 2017

Abgabe bis 16. November 2017 an: <mailto:gregor.hildebrand@uni-wuppertal.de>

Aufgabe 1. Ziele des modernen C++

Diskutieren Sie die Ziele des C++-Standards C++11ff.

<http://www.artima.com/cppsource/cpp0x.html>

insbesondere im Hinblick auf `templates`.

Wie passen die in C++14 eingeführten `variable templates` zu diesen Zielen?

Aufgabe 2. Typsichere generische Programmierung

Was versteht man unter typsicheren Programmiersprachen

(siehe <http://de.wikipedia.org/wiki/Typsicherheit> und

http://en.wikipedia.org/wiki/Type_safety)?

Sind C++-Templates typsicher? Welche Probleme existieren noch bei der Benutzung von C++-Templates (`interfaces`)?

Aufgabe 3. Generizität in C++ und in Java

Wie unterscheiden sich C++-Templates von Java Generics

(siehe http://de.wikipedia.org/wiki/Generische_Programmierung_in_Java)?

Aufgabe 4. Copy- und Move-Semantik

Was ist der Unterschied von Copy- und Move-Semantik etwa beim `operator=()` oder beim Kopierkonstruktor (<http://www.stroustrup.com/C++11FAQ.html#default2>)? Die Benutzung welcher Art von Operatoren erzeugt schneller ausführbare Binaries? Lesen Sie insbesondere auch "`To move or not to move`".

Aufgabe 5. Requirements an generische aktuelle Parameter

Welche Requirements muß der generische aktuelle Parametertyp erfüllen, um mittels `sort()` sortiert werden zu können?