



**BERGISCHE  
UNIVERSITÄT  
WUPPERTAL**

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

Fachbereich C  
Mathematik und Naturwissenschaften,  
Mathematik und Informatik

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

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

DATUM 13. November 2014

## **generische Programmierung**

**WS 2014/2015 – Übungsblatt 4**

**Ausgabe: 17. November 2014**

**Abgabe bis 24. November 2014 an: <mailto:125319@uni-wuppertal.de>**

### **Aufgabe 1. *TypeTraits***

Wozu werden in C++11 in Abschnitt 20.9 `UnaryTypeTraits` benutzt?

Lesen Sie dazu insbesondere

[http://publib.boulder.ibm.com/infocenter/comphelp/v9v111/index.jsp?topic=/com.ibm.xlcpp9.aix.doc/stdlib/header\\_type\\_traits.htm](http://publib.boulder.ibm.com/infocenter/comphelp/v9v111/index.jsp?topic=/com.ibm.xlcpp9.aix.doc/stdlib/header_type_traits.htm)

und <http://dotnet-snippets.de/dns/einfaches-beispiel-fuer-traits-SID877.aspx> sowie Seite 12ff.  
in

<http://artins.org/ben/programming/mactechgrp-artin-cpp-type-traits.pdf>.

### **Aufgabe 2. *Linguistic Support for Generic Programming in C++***

Lesen Sie in Bjarne Stroustrups Artikel

<http://www.stroustrup.com/oopsla06.pdf>

alles sich auf den Einsatz generischer Konstrukte Beziehende und stellen Sie es in eigenen Worten kurz dar. Was sollte in C++11 typischer werden als in der Vorgängerversion?

### **Aufgabe 3. *eingeschränkte Generizität***

Lesen Sie

<http://bartoszmilewski.com/2010/06/24/c-concepts-a-postmortem/>

und referieren Sie Alternativen, die eine Verbesserung der bei C++-Templates erzeugten Fehlermeldungen auch ohne Concepts bringen könnten.

### **Aufgabe 4. *STL reverse() und sort()-Requirement-Verletzung***

Schreiben Sie ein Testprogramm, das die vielseitige Anwendbarkeit des STL-Algorithmus `reverse()` für möglichst viele Datentypen demonstriert und testet.

Schreiben Sie ein Testprogramm, das die `sort`-Requirements beim Instanzierungsversuch mit einem geeigneten Containerexemplar als Parameter verletzt. Interpretieren Sie die einzelnen Fehlermeldungsanteile.

### **Aufgabe 5.** *Boost Filesystem*

Informieren Sie sich in

[http://www.boost.org/doc/libs/1\\_57\\_0/libs/filesystem/doc/index.htm](http://www.boost.org/doc/libs/1_57_0/libs/filesystem/doc/index.htm)  
über die im nächsten C++-Standard nach

**C++12DIS** File System TS  
zu erwartenden Neuerungen bezüglich Dateien/Inhaltsverzeichnissen/...

Welche neuen Klassen mit welchen Methoden sind zu erwarten  
([http://www.boost.org/doc/libs/1\\_57\\_0/libs/filesystem/doc/reference.html](http://www.boost.org/doc/libs/1_57_0/libs/filesystem/doc/reference.html))?