



**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 3. November 2014

## **generische Programmierung**

**WS 2014/2015 – Übungsblatt 2**

**Ausgabe: 3. November 2014**

**Abgabe bis 10. November 2014 an: <mailto:125319@uni-wuppertal.de>**

### **Aufgabe 1. TR1 und TR2**

Was ist im Zusammenhang mit C++ unter TR1 und was unter TR2 zu verstehen?

Was liefert das folgende Programmsegment Neues?

```
#include <iostream>
#include <typeinfo>
#include <tr1/utility>
using std::pair; using std::make_pair;
using std::tr1::get; using std::tr1::tuple_element;
using std::tr1::tuple_size;
using std::cout;

template <class Ty>
void show(const Ty& pr)
{
    cout << "size:  " << tuple_size<Ty>::value << '\n';
    cout << "first type:  "
        << typeid(typename tuple_element<0, Ty>::type).name() << '\n';
    cout << "second type:  "
        << typeid(typename tuple_element<1, Ty>::type).name() << '\n';
    cout << "first:  " << get<0>(pr) << '\n';
    cout << "second:  " << get<1>(pr) << '\n' << '\n';
}

int main()
{
    show(make_pair(1, 2));
    show(make_pair(3.0, 1.1f));
    return 0;
}
```

Wie werden also die TR1-Ergänzungen nutzbar? Welche weiteren Neuerungen im Container-Umfeld sind in der TR1 enthalten ([http://en.wikipedia.org/wiki/C++\\_Technical\\_Report\\_1](http://en.wikipedia.org/wiki/C++_Technical_Report_1))?

Beachten Sie auch

[Erweiterung der Programmbibliothek](#) und [C++ TR 24733](#).

**Aufgabe 2.** *C++11*

Welche Veränderungen bezüglich der template-Nutzung sind im neuen C++-Standard

<http://en.wikipedia.org/wiki/C%2B%2B11>

(siehe auch <http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2011/n3242.pdf>) durchgeführt?

**Aufgabe 3.** *STL-Iteratoren*

Wie unterscheiden sich die verschiedenen STL-Iterator-Kategorien (welche sind das?) voneinander (vergleiche <http://www.sgi.com/tech/stl/>)?

**Aufgabe 4.** *generische Programmierung*

Was versteht man nach

<http://theory.gsi.de/~vanhees/faq/gravitation/node70.html>

unter generischer Programmierung? Welche Vor-, welche Nachteile sind mit ihr verbunden?

**Aufgabe 5.** *C++11 in gcc 4.x*

Welche g++-Version benutzen Sie? Welche C++11-Eigenschaften enthält Ihr Compiler nach <http://gcc.gnu.org/projects/cxx0x.html> also noch nicht?