



# Generische Programmierung (Spezielle Kapitel der praktischen Informatik)

WS 2010/2011 – Übungsblatt 7

6. Dezember 2010

Abgabe: bis 13. Dezember 2010 an [c.seepold@uni-wuppertal.de](mailto:c.seepold@uni-wuppertal.de)

## Aufgabe 1. *Draft Proposal: Dynamic Libraries in C++*

Wie sollten gemäß

<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2003/n1496.html>

shared Objekte (so's) in C++ systemunabhängig standardisiert werden?

Warum wird dieser Erweiterungsvorschlag nach

<http://herbsutter.wordpress.com/2007/02/07/iso-c0x-complete-public-review-draft-in-october-2007/>

nicht in C++0x realisiert werden?

## Aufgabe 2. *Draft Proposal: Modules in C++*

Welche Vorteile wären mit Modulen in C++ verbunden:

<http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2006/n2073.pdf>

Wie soll die Syntax aussehen? Welche Vorteile brächten Module gegenüber der momentanen Praxis (welche ist das)?

Aus welchen Gründen wurden Module noch nicht in C++0x aufgenommen (vgl.

<http://herbsutter.wordpress.com/2007/02/07/iso-c0x-complete-public-review-draft-in-october-2007/>)?

## Aufgabe 3. *Compilezeit Fakultät*

Die Template-Funktion `fact` „berechnet“ zur Compilezeit die Fakultät:

```
template<int i>
class fact {
public:
    static const long long result = i * fact<i-1>::result;
```

```
};  
  
template<> class fact<1> {  
public:  
    static const long long result = 1;  
};
```

Ergänzen Sie diesen Quellcode um ein Test-Hauptprogramm. Wie wird `fact` aufgerufen? Wozu kann es benutzt werden, da es eine Compilezeit-Konstante produziert? Wie überprüft man diese Compilezeit-Evaluierung?

Modifizieren Sie das Codestück zu einer Klasse mit `enum-result` beziehungsweise zu einer normalen `template-Funktion` ohne `enum` oder `static const`. Welche Unterschiede stellen Sie fest?

#### **Aufgabe 4.** *Compilezeit Ganzzahlpotenz*

Schreiben Sie eine ähnliche `Template-Klasse` zur Berechnung von  $n^m$  für ganzzahlige `n` und `m`. Beschreiben Sie eine Nutzenanwendung für dieses `Template`, die die `Compilezeit-Evaluation` deutlich in den Vordergrund stellt.

Wo setzt die `C++-Einschränkung` an `Template nontype-Parameter` der Einsatzfähigkeit dieser Technik Grenzen?