



Generische Programmierung (Spezielle Kapitel der praktischen Informatik)

WS 2008 /2009 – Übungsblatt 7

10. Dezember 2008

Abgabe: bis 17. Dezember 2008 an
gcaltavu@studs.math.uni-wuppertal.de

Aufgabe 1. *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 2. *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?

Aufgabe 3. *Template binary*

Testen Sie:

```
template <unsigned long N>
struct binary
{
    static unsigned const value
        = binary<N/10>::value * 2    // prepend higher bits
        + N%10;                      // to lowest bit
};

template <>                                // specialization
struct binary<0>                          // terminates recursion
{
    static unsigned const value = 0;
};
```

Was wird hier berechnet? Lassen Sie eine Tabelle berechneter Werte ausdrucken. Welche ähnliche Anwendungen von Templates erscheinen Ihnen nützlich?

Aufgabe 4. *Metaprogrammierung*

Lesen Sie

<http://divykapoor.blogspot.com/2008/07/walking-through-your-first-template.html>

und testen Sie das Fibonacci-Beispiel.

Welche Einsatzgebiete sieht der Autor für Metaprogrammierung?