

8. Aufgabenblatt – C++

Ausgabe: 16. Juni 2009 Abgabe: 23. Juni 2009 Revision: 3318

1 C++

Die Sprache C++ ist eine objektorientierte Erweiterung von C. Sie zeichnet sich dadurch aus, dass sie größtenteils kompatibel zu C ist und man also genauso low-level wie in C programmieren kann. Darüber hinaus setzt sie zahlreiche modernere Features um, wie Klassen, Vererbung (auch Mehrfachvererbung), oder generische Programmierung. C++ ist eine wahrhaft “universelle” Sprache, deren Universalität aber eher durch die Summe ihrer Features erreicht wird. Im Vergleich zu C++ ist Java eine weit elegantere und schlankere Sprache; es gibt sehr gute Tutorials, die C++ aus Java-Perspektive beschreiben.¹²

2 Einfache Templates

Implementieren Sie eine Klasse *Vector*, die analog zur gleichnamigen Java-Klasse *Elemente* eines beliebigen Typs *T* aufnehmen kann. Der Einfachheit halber soll Ihre Klasse die folgenden Operationen unterstützen:

- Konstruktor unter Angabe der Größe: `Vector<T> v(int size)`
- Lesen eines beliebigen Elements: `T get(int i)`
- Schreiben eines beliebigen Elements: `void set(int i, T x)`

Die Lese- und Schreib-Operationen sollen (im Gegensatz zu den eingebauten C++-Feldern) eine Fehlermeldung ausgeben, falls mit einem ungültigen Index zugegriffen wird.

Hier ein Beispiel für die Anwendung:

¹<http://pages.cs.wisc.edu/hasti/cs368/CppTutorial/index.html>

²<http://www.java2s.com/Tutorial/Cpp/CatalogCpp.htm>

```
const int SIZE = 100;
Vector<int> v(SIZE);
for (int i = 0; i < SIZE; i++) {
    v.set(i, i * 2);
}
for (int i = 0; i < SIZE; i++) {
    assert(v.get(i) == i * 2);
}
int x = v.get(SIZE); // Fehlermeldung
```

Zur Implementierung der Vector-Klasse benutzen Sie den `new`-Operator, um eingebaute C++-Felder anzulegen.

```
template class<T>
class Vector {
    T *pData;
    int size;
public:
    ...
    pData = new T[size]; // Feld anlegen
    ...
    return pData[i]; // Zugriff
    ...
}
```

Wenn Sie C++-Experte sind (oder werden wollen), dürfen Sie auch Copy-Konstruktor, Destruktor, Zuweisung, und den `[]`-Operator definieren – und Werte durch Referenzen ersetzen.

3 Spezialisierte Templates

Implementieren Sie eine spezielle Variante `Vector<bool>`, die für Boolesche Werte optimiert ist. Hier sollen Sie die einzelnen Bits nicht in einem `bool`-Feld ablegen, sondern die Bits von `unsigned int`-Variablen (vorzeichenlose ganze Zahlen) einzeln ansprechen. Benutzen Sie hierfür die Bit-Operationen `|` (oder), `&` (und), und `~` (Negation).

4 Metaprogrammierung

Templates ermöglichen in C++ *Metaprogrammierung* – und damit Berechnungen zur Übersetzungszeit. Tatsächlich sind die C++-Templates berechnungsuniversell.

Betrachten Sie das folgende Programm `Foo.cxx`. Was tut es, und wie?

```
#include <iostream>
using namespace std;

template <int N>
struct Foo
{
    enum { value = N * Foo<N - 1>::value };
};

template <>
struct Foo<0>
{
    enum { value = 1 };
};

int main()
{
    int x = Foo<4>::value;
    int y = Foo<0>::value;
    cout << "x = " << x << ", y = " << y << endl;
}
```

5 Diskussion

Diskutieren Sie folgende Fragestellungen auf etwa einer DIN A4 Seite:

1. Welche Vorteile hat eine Allround-Sprache wie C++? Welche Nachteile?
2. Wie erleichtert Java im Vergleich zu C++ die Programmierung?
3. Welche Anwendungen gibt es für Metaprogrammierung?

6 Abgabe Ihrer Lösung

Drucken Sie Ihre Lösung aus und werfen Sie den Ausdruck bis zum **23. Juni 2009 um 9:00 Uhr** in den Briefkasten des Lehrstuhls für Softwaretechnik³. Vergessen Sie dabei nicht, Ihren Ausdruck mit Ihrem Namen und Ihrer Matrikelnummer zu versehen.

³Gebäude E1 1, neben dem InfoPoint des Rechenzentrums.

Bringen Sie bitte zum Proseminar ebenfalls einen Ausdruck mit, so dass Sie ihn bei etwaigen Diskussionen vorliegen haben.

7 Links und Hinweise

Zum Erstellen und Bearbeiten Ihres Programms können Sie einen gewöhnlichen Texteditor benutzen. Der GNU C++-Compiler ist auf so gut wie jeder Unix-Installation vorhanden. Er wird aufgerufen mit

```
$ g++ -Wall -g -o meinProgramm meinProgramm.cxx
```

Das entstehende Programm können Sie dann so ausführen:

```
$ ./meinProgramm
```

Mit Hilfe der Standard-Ausgabe cout können Sie den Ablauf Ihres Programms verfolgen.