

```
// Beispiel eines einfachen Datentyps
// Nur Konstruktion und simple Nutzung
```

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

```
class Bruch {
private:
    int zaehler;
    int nenner;
public:
    // Mehrere Konstruktoren
    Bruch ();
    Bruch (int);
    Bruch (int, int);

    // Eine Funktion
    void print ();
}; // Bruch
```

```
// Default-Konstruktor
```

```
Bruch::Bruch () {
    zaehler = 0;
    nenner = 1;
}
```

```
// Konstruktor aus ganzer Zahl
```

```
Bruch::Bruch (int z) {
    zaehler = z;
    nenner = 1;
}
```

```
// Konstruktor aus Zähler und Nenner
```

```
Bruch::Bruch (int z, int n) {
    if (n == 0) {
        // Programm mit Fehlermeldung beenden
        cerr << "Fehler: Nenner ist 0!" << endl; exit (1);
    }
    // Keine Normalisierung
    zaehler = z;    nenner = n;
}
```

```
void Bruch::print () {
    cout << zaehler << "/" << nenner << endl;
} // print
```

```
int main () {
    Bruch b;
    cout << "Bruch b = " ; b.print ();
    Bruch a (23, 41);
    Bruch c = Bruch (7, 9);
    cout << "Bruch a = " ; a.print ();
    cout << "Bruch c = " ; c.print ();
    b = a; // Zuweisungsoperator
    a = Bruch (17);
    cout << "Bruch b = " ; b.print ();
    cout << "Bruch a = " ; a.print ();
}
```

```
return 0;
```

```
} // main
```

```
/* Beispiellauf:
```

```
Bruch b = 0/1
Bruch a = 23/41
Bruch c = 7/9
Bruch b = 23/41
Bruch a = 17/1
```

```
*/
```