Blatt 12

Vincent Kümmerle und Elvis Gnaglo

13. Januar 2026

1 Datentypen

```cpp
#include <iostream>
#include <string>
#include <cmath>
#include <typeinfo>

int main() {
    auto v1 = 3 + 5; // Ganzzahl-Addition
    std::cout << "3 + 5 = " << v1 << " | Typ: " << typeid(v1).name()
        << std::endl;

    auto v2 = 3 + 5.0; // Misch-Addition (Implicit Type Promotion)
    std::cout << "3 + 5.0 = " << v2 << " | Typ: " << typeid(v2).name
        () << std::endl;

    // "3" + "5" würde einen Compilerfehler verursachen (Zeiger-
        Addition)

    auto v4 = std::string("3") + "5"; // String-Konkatenation
    std::cout << "std::string(\"3\") + \"5\" = " << v4 << " | Typ: "
        << typeid(v4).name() << std::endl;

    auto v5 = 3 / 2; // Ganzzahl-Division
    std::cout << "3 / 2 = " << v5 << " | Typ: " << typeid(v5).name()
        << std::endl;

    auto v6 = 3.0 / 2; // Gleitkomma-Division
    std::cout << "3.0 / 2 = " << v6 << " | Typ: " << typeid(v6).name
        () << std::endl;

    auto v7 = int(2.71828); // Explizites Casting (Nachkommastellen
        fallen weg)
```

```
25      std::cout << "int(2.71828) = " << v7 << " | Typ: " << typeid(v7).
            name() << std::endl;
26
27      auto v8 = std::round(2.71828); // Mathematisches Runden
28      std::cout << "std::round(2.71828) = " << v8 << " | Typ: " <<
            typeid(v8).name() << std::endl;
29
30      return 0;
31  }
```

Output: int = Gazzahl, double = Gleitkommazahl

```
3 + 5 = 8 | Typ: int
3 + 5.0 = 8 | Typ: double
std::string("3") + "5" = 35 | Typ: class std::basic_string<char,
    struct std::char_traits<char>,class std::allocator<char> >
3 / 2 = 1 | Typ: int
3.0 / 2 = 1.5 | Typ: double
int(2.71828) = 2 | Typ: int
std::round(2.71828) = 3 | Typ: double
```

2