

Aufgabe 1

- a) Der Datentyp float hat eine Genauigkeit von 6 Stellen. Daher wird 10000.0001 auf 10000 gerundet. Verwendet man den Datentyp double, so erhält man eine Genauigkeit von 15 Stellen. Daher stimmt nun das Ergebnis.

Listing 1: determinante.cc

```
1 #include "fcpp.hh"
2
3 float determinante(float a, float b, float c, float d)
4 {
5     return a * d - b * c;
6 }
7
8 double determinante2(double a, double b, double c, double d)
9 {
10     return a * d - b * c;
11 }
12
13 int main ()
14 {
15     print(determinante2(100, 0.01, -0.01, 100));
16     return 0;
17 }
```

- b) Ab $n = 3$ treten Unterschiede innerhalb der ersten 6 Stellen der Zahl auf, Ungenauigkeiten bei $n \leq 2$ lassen sich direkt auf die Ungenauigkeit von floats zurückführen. Für $n = 3$ muss bei der Rechnung $a + (b + c)$ in $(b + c)$ die Zahl $10^3 + 10^{-3}$ gespeichert werden. Diese hat insgesamt 7 Stellen, ein float hat aber nur eine Genauigkeit von 6 Stellen.

Listing 2: addition.cc

```
1 #include "fcpp.hh"
2
3
4 float add(int n)
5 {
6     float a = pow(10, n);
7     float b = -pow(10, n);
8     float c = pow(10, -n);
9
10    return (a+b)+c;
11 }
12
13
14 float add2(int n)
15 {
```

```
16     float a = pow(10, n);
17     float b = -pow(10, n);
18     float c = pow(10, -n);
19
20     return a+(b+c);
21 }
22
23 int main()
24 {
25     int n;
26     std::cout << "Enter int n" << std::endl;
27     std::cin >> n;
28     print(add(n));
29     print(add2(n));
30     return 0;
31 }
```

Aufgabe 2

Welche Ergebnisse wären bei exakter Rechnung zu erwarten, und in welchen Punkten weichen die von Ihnen erhaltenen davon ab? Können Sie mögliche Ursachen eingeben? Bei exakter Rechnung müssten die Ergebnisse für float und double gleich sein. Das ist aber nicht der Fall, da ein float nur eine Genauigkeit von 6 Stellen hat, ein double hat stattdessen eine Genauigkeit von 15 Stellen.

Listing 3: zinssatz.cc

```
1 #include "fcpp.hh"
2
3 float zins(float z, int n)
4 {
5     return pow(1 + z * pow(n, -1), n) - 1;
6 }
7
8 double zins2(double z, int n)
9 {
10     return pow(1 + z/n, n) - 1;
11 }
12
13 void ausgabe(int n)
14 {
15     print(n);
16     print(zins(0.06, n));
17     print(exp(0.06) - 1 - zins(0.06, n));
18     print(zins2(0.06, n));
19     print(exp(0.06) - 1 - zins2(0.06, n));
20 }
```

```
21
22 int main()
23 {
24     ausgabe(1);
25     ausgabe(4);
26     ausgabe(12);
27     ausgabe(356);
28     ausgabe(365*24);
29     ausgabe(365*24*60);
30     ausgabe(365*24*60*2);
31     ausgabe(365*24*60*60);
32     return 0;
33 }
```

Aufgabe 3

Fallunterscheidung

1.) $a, b \geq 0$:

$$\begin{aligned} & s_n(d_n(a) \cdot d_n(b)) \\ &= s_n(a \cdot b) \\ &= a \cdot b \\ &= d_n(a \cdot b) \end{aligned}$$

2.) $a \geq 0, b < 0$

$$\begin{aligned} & s_n(d_n(a) \cdot d_n(b)) \\ &= s_n(a \cdot (2^n - |b|)) \\ &= s_n(a \cdot 2^n - a \cdot |b|) \\ &= -a \cdot |b| \\ &= d_n(-a \cdot |b|) \end{aligned}$$

3.) $a < 0, b < 0$

$$\begin{aligned} & s_n(d_n(a) \cdot d_n(b)) \\ &= s_n((2^n - |a|) \cdot (2^n - |b|)) \\ &= s_n(2^{2n} - 2^n \cdot |a| - 2^n \cdot |b| + |a| \cdot |b|) \\ &= |a| \cdot |b| \\ &= d_n(|a| \cdot |b|) \end{aligned}$$