

```

1  #include <iostream>
2  #include <iomanip>
3  using namespace std;
4
5  int Fehlercode=0;
6  // Aufgabe 16
7  double berechneSumme(double & Mittelwert, double Liste[], int Anzahl) {
8      double Summe=0;
9      for (int i=0; i < Anzahl; i=i+1) {
10         Summe=Summe+Liste[i];
11     }
12     Mittelwert=Summe/Anzahl;
13     return Summe;
14 }
15 // Aufgabe 17
16 int berechneQuersumme(int Zahl) {
17     int Summe=0;
18     while (Zahl != 0) {
19         Summe=Summe+Zahl%10;
20         Zahl=Zahl/10;
21     }
22     return Summe;
23 }
24
25 int main() {
26     bool Bedingung=false;
27     char Weiter='J';
28     double Liste[10]={1.1,1.4,2.3};
29     int Anzahl=3;
30     double Mittelwert, Summe;
31     int Quersumme;
32
33     do {/*
34         cout << "Anzahl Element? ";
35         cin >> Anzahl;
36         for (int i=0; i < Anzahl; i=i+1) {
37             cin >> Liste [i];
38         }*/
39         Summe=berechneSumme(Mittelwert, Liste, Anzahl);
40
41         cout << "Mittelwert ist " << Mittelwert << endl;
42         cout << "Summe der Liste ist " << Summe << endl;
43
44         Quersumme=berechneQuersumme(123);
45         cout << "Quersumme von 123 ist " << Quersumme << endl;
46         Quersumme=berechneQuersumme(3234);
47         cout << "Quersumme von 3234 ist " << Quersumme << endl;
48         Quersumme=berechneQuersumme(1234567432);
49         cout << "Quersumme von 1234567432 ist " << Quersumme << endl;
50
51         cout << "Weiter ("<< Weiter<< "=Ja,sonst=Nein) ";
52         cin >> Weiter;
53         cin.sync();
54         Bedingung=(Weiter=='J');
55     } while(Bedingung);
56
57     return Fehlercode;
58 }
59 // Aufgabe 18
60 #include <iostream>
61 #include <iomanip>
62 using namespace std;
63
64 void sortiere1(int Liste[], int Anzahl) {
65     bool nichtSortiert=true;
66     int Temp;
67     while (nichtSortiert) {
68         nichtSortiert=false;
69         for (int i=0; i < Anzahl-1; i=i+1) {
70             if (Liste[i] > Liste[i+1]) {
71                 Temp=Liste[i];
72                 Liste[i]=Liste[i+1];
73                 Liste[i+1]=Temp;

```

```

74         nichtSortiert=true;
75     }
76 }
77 }
78 return;
79 }
80 // Alternative Lösung 11 Durchläufe
81 void sortiere(int Liste[], int Anzahl) {
82     int Temp;
83     int i=0;
84     while (i < Anzahl-1) {
85         if (Liste[i] > Liste[i+1]) {
86             Temp=Liste[i];
87             Liste[i]=Liste[i+1];
88             Liste[i+1]=Temp;
89             i=0;
90         } else {
91             i=i+1;
92         }
93     }
94     return;
95 }
96
97 int Fehlercode=0;
98
99 int main() {
100     int Liste[6]={1,3,5,0,9,15};
101     sortiere(Liste, 6);
102     for (int i: Liste) {
103         cout << i << ' ';
104     }
105     return Fehlercode;
106 }
107

```