```cpp
//
// Created by Jan Duchscherer on 18.11.22.
//

#include <iostream>
#include <complex>

class IZweipol {
  public:
    virtual std::complex<double> impedanz(double w) const = 0;
    virtual ~IZweipol() = default;
};

class R : public IZweipol {
  private:
    double r;
  public:
    explicit R(double r) : r{r} {}
    std::complex<double> impedanz(double) const override {
        //return static_cast<std::complex<double>>(r);
        return {r, 0.0};
    }
};

class L : public IZweipol {
  private:
    double l;
  public:
    explicit L(double l) : l{l} {}
    std::complex<double> impedanz(double w) const override {
        return {0, w * l};
    }
};

```

```
35 class C : public IZweipol {
36   private:
37     double c;
38   public:
39     explicit C(double c) : c{c} {}
40     std::complex<double> impedanz(double w) const
   override {
41         return {0, -1 / (w * c)};
42     }
43 };
44
45 class Schaltung : public IZweipol {
46   private:
47     IZweipol* elemente[2];
48   public:
49     Schaltung(IZweipol* s1, IZweipol* s2)
50         : elemente{s1, s2} {}
51
52     void add(IZweipol* e, size_t idx) {
53         elemente[idx] = e;
54     }
55     void remove(size_t idx) {
56         if (elemente[idx]) {
57             elemente[idx] = nullptr;
58         }
59     }
60     IZweipol* getChild(size_t idx) const {
61         return elemente[idx];
62     }
63 };
64
65 class SerSchaltung : public Schaltung {
66   public:
67     explicit SerSchaltung(IZweipol* s1, IZweipol*
   s2) : Schaltung(s1, s2) {}
68     std::complex<double> impedanz(double w) const
   override {
69         return getChild(0)->impedanz(w) + getChild(
```

```
69 1)->impedanz(w);
70     }
71 };
72
73 class ParSchaltung : public Schaltung {
74   public:
75     explicit ParSchaltung(IZweipol* s1, IZweipol*
   s2) : Schaltung(s1, s2) {}
76     std::complex<double> impedanz(double w) const
   override {
77         return std::complex<double>(1, 0) / (std::
   complex<double>(1, 0) / getChild(0)->impedanz(w)
78             + std::complex<double>(1, 0) /
   getChild(1)->impedanz(w));
79
80     }
81 };
82
83 int main() {
84     R r1{50};
85     R r2{300};
86     R r3{20};
87     L l1{1};
88     L l2{1.5};
89     C c1{10e-6};
90     double w = 300;
91     SerSchaltung s1{&r3, &l2};
92     SerSchaltung s2{&r2, &c1};
93     SerSchaltung s3{&r1, &l1};
94     ParSchaltung p1{&s1, &s2};
95     SerSchaltung s4{&s3, &p1};
96     std::cout << "impedanz of s4: " << s4.impedanz
   (w) << std::endl;
97     return 0;
98 }
99
```