

## Aufgabe 1

Listing 1: aufgabe1.cc

---

```
1 #include<iostream>
2 #include<fstream>
3 #include<string>
4
5 int main(int argc, char** argv)
6 {
7     std::ifstream input;
8     std::ofstream output;
9     input.open("test.txt");
10    output.open("test-a.txt");
11    int counter = 0;
12    bool firstline = true;
13    while (input.good())
14    {
15        if (firstline)
16        {
17            std::string line;
18            std::getline(input, line);
19            ++counter;
20            output << counter << ": " << line;
21            firstline = false;
22        }
23        else
24        {
25            std::string line;
26            std::getline(input, line);
27            ++counter;
28            output << std::endl << counter << ": " << line;
29            firstline = false;
30        }
31    }
32    input.close();
33    output.close();
34 }
```

---

## Aufgabe 2

- In Zeile 41 `aq = ap`; ist ein Fehler, da `aq` private ist.
- In Zeile 52 `aX()`; ist ein Fehler, da `aX()` private ist.
- In Zeile 60 `b.bY()`; ist ein Fehler, da `bY()` private ist.

- In Zeile 64 `c.aq = 5;` ist ein Fehler, da `aq` private ist.

## Aufgabe 3

- `A a;` ist ein Fehler, da `A` eine abstrakte Klasse ist.
- `pa->c();` ist ein Fehler, da `A::c` private ist.
- `pa->b();` ist ein Fehler, da `b()` eine Methode in `B` ist.
- `pa->vb();` ist ein Fehler, da `vb()` eine Methode in `B` ist.
- `pa->a(x);` wird für `float` überladen und liefert daher keinen Fehler, obwohl keine Methode explizit dafür definiert ist
- `pb->a();` ist ein Fehler, da in `B` die Methode `void a()` durch `void a(int/double a)` überschrieben wird
- `pb->c();` ist ein Fehler, da `c` nicht in `B` implementiert ist
- `pb->a(x);` ist kein Fehler, da in `B` einfach `double` überladen wird
- `pa->c();` ist ein Fehler, da `A::c` private ist, darf es nicht benutzt werden
- `pa->b();` ist ein Fehler, da `b` nur in `B` und `C` sichtbar ist
- `pa->vb();` ist ein Fehler, da `vb()` nur in `B` und `C` sichtbar ist