

## Week 4

### Exercise 25

../25/insertable/insertable.h

```
1  #ifndef INCLUDED_INSERTABLE_
2  #define INCLUDED_INSERTABLE_
3
4  #define HDR_    template <typename Data, \
5                  template <typename, typename> class Container, \
6                  template <typename> class AllocationPolicy>
7  #define CONT_  Container<Data, AllocationPolicy<Data>>
8  #define INS_   Insertable<Data, Container, AllocationPolicy>
9
10 #include <vector>
11 #include <memory>
12 #include <iterator>
13
14 template <typename Data,
15 template <typename, typename> class Container = std::vector,
16 template <typename> class AllocationPolicy = std::allocator>
17 class Insertable: public Container<Data, AllocationPolicy<Data>>
18 {
19     public:
20
21     Insertable();
22     Insertable(const CONT_ &RHS);
23     Insertable(const Insertable &RHS);
24     Insertable(Data RHS);
25
26     friend std::ostream &operator<<(std::ostream &out, const Insertable &ins)
27     {
28         std::copy (ins.begin(), ins.end(), std::ostream_iterator<Data>(out, "\n"));
29         return out;
30     };
31 };
32
33 HDR_
34 INS_::Insertable()
35 : CONT_()
36 {};
37 HDR_
38 INS_::Insertable(const CONT_ &RHS)
39 : CONT_(RHS)
40 {};
41 HDR_
42 INS_::Insertable(const Insertable &RHS)
43 : CONT_(RHS)
44 {};
45 HDR_
46 INS_::Insertable(Data RHS)
47 : CONT_(RHS)
48 {};
49
50 #undef HDR_
51 #undef CONT_
52 #undef INS_
53 #endif
```