Answer for Homework 4:

2.

The instantiation of Set<Bottle>::insert calls Set<Bottle>::contains, which calls Set<Bottle>::find, which contains the expression p->m\_value != value, where both operands are Bottle. The operator!= for Bottle operands has never been defined in the function.

3.B

If we only overload once, then there’s no way that the path can be traced from the root to leaves. So every time we call the recursive function, the path will not be updated correctly.

4.A

O(N^3): cause there’s three for loops, and each of them will compile N times.

4.B

O(N^3): cause there’s three for loops, the first two will compile N\*(N-1)/2 times, and the most inner one will still compile N times, so the maximum will be N^3 times. But actually, B compiles only approximately half of A’s times.

5.A

Both the insert function (which calls contains, and then calls find, in which there’s a for loop) and the get function will compile N times, and the for loop will also compile N times, so this is a O(N^2). Other steps will compile N or constant times, so we can ignore them.

5.B

The sort function will take O(NlogN) times, other steps are all O(N), which are dominated by O(NlogN), so this is in all a O(NlogN).