

Assessments

The programming solutions for each chapters' questions can be found in our GitHub repository at the following URL: <https://github.com/PacktPublishing/Demystified-Object-Oriented-Programming-with-CPP/tree/master>. Each full program solution can be found in the GitHub under the appropriate chapter heading (subdirectory, such as `Chapter01`) in the subdirectory `Assessments`, in a file that corresponds to the chapter number, followed by a dash, followed by the solution number in the chapter at hand. For example, the solution for question 3 in chapter 1 can be found in the subdirectory `Chapter01/Assessments` in a file named `Chp1-Q3.cpp` under the aforementioned GitHub directory.

The written responses for non-programming questions can be found in the following sections. Should an exercise have a programming portion and a follow-up question, the answer to the follow-up question may be found both in the next sections and in a comment at the top of the programming solution on GitHub (as it may be appropriate to review the solution in order to fully understand the answer to the question).

Chapter 4 – Indirect Addressing: References

1. **a – c:** Please see `Chapter04/Assessments/Chp4-Q1.cpp` in the GitHub repository.
c: (follow-up question) Pointer variables need not only call the version of `ReadData(Student *)` that accepts a pointer to a `Student` and reference variables need not only call the version of `ReadData(Student &)` that accepts a reference to a `Student`. For example, a pointer variable may be dereferenced with `*` and then call the version that accepts a reference. Likewise, a reference variable may have its address taken using `&` and then call the version that accepts a pointer (though this is less commonly done). You simply need to make the data types match with respect to what you are passing and what the function expects.