

# Part C: Reflection & Short-Answer

## 1. Pros and Cons

List two benefits of hiding internal data behind methods, and one potential limitation or overhead introduced by this design approach.

Pros: Validation is needed before modifying data, Internal implementation can change without affecting external code.

Cons: Getter/setter methods introduce additional function calls that would not be otherwise needed.

## 2. Encapsulation vs. Other Concepts

How does encapsulation differ from abstraction?

Encapsulation hides internal data and restricts access. Abstraction hides complex implementation details and exposes only things that are necessary.

Why might we consider both “encapsulation” and “abstraction” to be forms of “information hiding”?

They can both be considered forms of information hiding because they hide data to make sure only necessary information is exposed.

### 3. Testing Encapsulated Classes

If the data is private, how can we still unit test the class thoroughly? Propose a short strategy that ensures valid testing without exposing private data.

You can test by running methods to make sure that they make sense. If the class was a bank account, you could test that deposit and withdraw work correctly.