**Chapter 1 Review Answers**

**1. What is a module?**

A module is an independent piece of code that provides specific and tightly coupled functionality. Examples in C++ are functions, classes and methods. See Section 1.1.2 of your textbook.

**2. What is an object?**

An object is a variable whose data type is defined by a class. That is, if we declare variables of a class type, we call them objects. Objects are also called instances of a class.

**3. What is encapsulation?**

See Section 1.1.2 in your textbook.

**4. What is inheritance?**

See Section 1.1.2 in your textbook.

**5. What is polymorphism?**

See Section 1.1.2 in your textbook.

**6. When talking about modules, what is cohesion?**

See Section 1.2.1 in your textbook.

**7. When talking about modules, what is coupling?**

See Section 1.2.2 in your textbook.

**8. What is a precondition? What is a postcondition? How are they used?**

See section 1.3.1 of your textbook.

**9. What is a client programmer?**

A client programmer is the programmer who uses a class (or class template) to create an application program. See page 8 of your textbook.

**10. Describe two good ways to handle unusual (usually invalid) conditions in a class.**

See section 1.3.2 in your textbook. Hint: Ignoring invalid conditions or trying to guess the client's intentions are not good ways to handle them.

**11. What does abstraction mean? What is data abstraction?**

See section 1.3.3 of your textbook.

**12. What is information hiding?**

See section 1.3.4 of your textbook.

**13. What is an ADT?**

An ADT is is a logical description of a group of data values and the operations that are allowed on them. This specification should not indicate how the data will be stored or how the operations will be implemented.

14. **According to the textbook, what is the final step in designing an ADT (assuming that you plan to implement the ADT in C++)?**

Write an abstract base class for the ADT that documents the ADT specifications. See Section 1.5.3 of your textbook.

**15.What do we call the place where a class meets and interacts with other code modules?**

An interface - see Section 1.3.5 of your textbook.

**16. Should a client programmer be able to write an application based on the information contained in an ADT interface file?**

Yes. According to the textbook, the interface file should contain comments that completely specify the methods of an ADT. For each method, the interface file should contain:

* the purpose of the method
* a description of all parameters
* a description of the return value
* any preconditions and postconditions

See the two notes in section 1.5.3 of the textbook.