1. Object-oriented programming primarily focuses on
   (a) classes.
   (b) functions.
   (c) variables.
   (d) constants.

2. Classes do not have the property of
   (a) encapsulating data.
   (b) information hiding.
   (c) containing both data and functions.
   (d) usually knowing how other classes are implemented.

3. The proper format for a struct is
   (a) `struct Time
       int hour
       int minute;`
   (b) `struct Time {
       int hour,
       int minute,
   }
   (c) `struct Time {
       int hour;
       int minute;
   }
   (d) `struct Time {
       int hour;
       int minute;
   };`

4. Which of the following is not a property of structs?
   (a) structs reserve space in memory when they are defined.
   (b) structs are built using elements of other data types.
   (c) Members of a struct must have unique names.
   (d) Structure variables are declared like other variables, except the structure name is used as the type.
5. **structs** are not allowed to contain
   (a) pointers to **structs** of different types.
   (b) **struct** variables of the same type.
   (c) pointers to themselves.
   (d) both **floats** and **ints**.

6. **timePtr** is a pointer to object **timeObject** with data member **hour**. Which of the following is not equivalent to **hour**?
   (a) **timeObject.hour**
   (b) **timePtr.hour**
   (c) **timePtr->hour**
   (d) (**timePtr**).**hour**

7. By default, structures are passed
   (a) call-by-value
   (b) call-by-reference
   (c) one member at a time
   (d) structures cannot be passed between functions

8. Which of the following is not true?
   (a) **classes** contain both data members and member functions
   (b) a class definition must be terminated with a semicolon
   (c) all classes can be represented as **structs**
   (d) the body of a class definition is delineated with left and right braces

9. Member access specifiers (**public** and **private**) can appear
   (a) in any order and multiple times.
   (b) in any order (**public** first or **private** first) but not multiple times.
   (c) in any order and multiple times, if they have brackets separating each type.
   (d) outside a class definition.

10. Object-oriented programming generally does not focus on
    (a) separating the interface and implementation of a program.
    (b) ease of program modifiability.
    (c) information hiding.
    (d) client-side access to implementation details.