1. Memberwise assignment
   A. safely copies one class to another
   B. occurs when a default copy constructor is called
   C. copies all members except pointers.
   D. copies everything except member variables.

2. Memberwise assignment should not be used
   A. ever
   B. when working with structs
   C. when memory has been allocated to a member
   D. with arrays that are declared statically

3. When is a copy constructor called?
   A. When a program begins.
   B. When a class goes out of scope.
   C. Just before the destructor is called.
   D. When a class is initialized.

4. Aggregation occurs
   A. when a class contains an instance of another class.
   B. when an array of classes is declared.
   C. for every class by default.
   D. when a class is copied to another class.

True/false 1-point each.

5. T  F Private members must be declared before public members.

6. T  F Class members are private by default.

7. T  F Members of a struct are private by default.

8. I  F Classes and structures in C++ are very similar.

9. T  F All private members of a class must be declared together.

10. T  F All public members of a class must be declared together.

11. I  F It is legal to define a pointer to a class object.

12. I  F You can dynamically allocate an instance of a class.

13. I  F Constructors cannot have a return type.

14. T  F Destructors can be overloaded.

15. T  F The [] operator cannot be overloaded in a class.
16. What are the four major features of object oriented design and programming?

___________________________ abstraction
___________________________ encapsulation
___________________________ inheritance
___________________________ polymorphism

17. What is the exact output of the following program code segment? Write your answer in the box to the right.

```c
int a = 5, b = 3;
a -= b++;
b += ++a;
a += --b;
b *= a++;
cout << a++ << ++b << endl;
```

18. Write a for-loop that will fill the following array with unique values that fall within the range 0 to 1.

```c
double arr[25];
for (int i=0; i<25; i++)
    arr[i] = (double)i * (1.0 / 24.0);
```

19. Given a class named Turtle, write the function header for its copy constructor.

```
Turtle(const Turtle &t)
(const is optional, & is required)
```

20. Given a class named Yen, write the header for a member function that overloads the < operator for the class.

```
bool operator<(const Yen &y)
(const is optional, bool return type is required)
```
Coding portion

Write a complete program that is described below.
Name your program /222/exam2/circle.cpp.

Write a class declaration named Circle with a private member variable named radius. Write set and get functions to access the radius variable, and a function named getArea that returns the area of the circle. The area is calculated as 3.14159 * radius * radius.

Add a default constructor to the Circle class. The constructor should initialize the radius member to 1.0.

Add an overloaded constructor to the Circle class. The constructor should accept an argument and assign its value to the radius member variable.

Write a main function that demonstrates your class.

Write a statement that defines an array of five objects of the Circle class. Let the default constructor execute for each element of the array.

Write a statement that defines an array of five objects of the Circle class. Pass the following arguments to the elements’ constructor: 12, 7, 9, 14, and 8.

In your class, overload the += operator in order to increase a circle's radius by the value of another circle's radius.

Use a for-loop to increase the radius values of your first array by calling your overloaded += operator.

Use for-loops to display the radius and area of the circles represented by the arrays you defined.