

**Computer Information Systems  
Classes and Functions Review Lab**

Using the attached design, create a C++ console application that uses classes. This assignment should not take an extended amount of time as the code is very simple. You may use your books and any other code as samples. When you have finished your program, print out the source code and attach the source and the design specs to this document. Don't forget to put your name on this document. Then turn everything in.

Name : \_\_\_\_\_

**Scoring Guide**

<b>Item</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
Includes a header comment with: Name, creation date, program name, purpose and copyright statements	1 of 5	2 of 5	3 of 5	4 of 5	5 of 5
Program runs as designed					Yes
Syntax errors	4	3	2	1	0
Comment areas missed	4	3	2	1	0
Spelling Errors in printed output					No
Staying on task (instructor's choice)					
Items missed in design	4	3	2	1	0

Total points: \_\_\_\_\_ out of 35

A = 32 B = 28 C = 25 D = 21

## Classes and Functions Review Lab # 1

1. Add the class definition for a class named Media
  - a. In the public section
    - i. Add the prototype for the default constructor for the Media class
    - ii. Add the prototype for a setName function that returns void and takes one string parameter
    - iii. Add the prototype for a setYear function that returns void and takes one integer parameter
    - iv. Add the prototype for a setLocation function that returns void and takes one string parameter
    - v. Add the prototype for a getName function that returns a string and takes no parameters
    - vi. Add the prototype for a getYear function that returns an integer and takes no parameters
    - vii. Add the prototype for a getLocation function that returns a string and takes no parameters
  - b. In the private section
    - i. Add a string attribute named Name
    - ii. Add a string attribute named Location
    - iii. Add an integer attribute named Year
2. Create the default constructor for the Media class( remember that it is named the same as the class)
  - a. Assign a blank to Name
  - b. Assign a blank to Location
  - c. Assign a 0 to Year
3. Create the setName class function (remember that it returns void and takes one string parameter)
  - a. Assign Name = parameter that you create
4. Create the setYear class function (remember that it returns void and takes one integer parameter)
  - a. Assign Year = parameter that you create
5. Create the setLocation class function (remember that it returns void and takes one string parameter)
  - a. Assign Location = parameter that you create
6. Create the getLocation class function (remember that it returns a string and takes no parameters)
  - a. Return the Location
7. Create the getName class function (remember that it returns a string and takes no parameters)
  - a. Return the name
8. Create the getYear class function (remember that it returns a string and takes no parameters)
  - a. Return the Year
9. In the main function
  - a. Create two objects of type Media
  - b. Assign a different name to each object using the setName class function
  - c. Assign a year to each object using the setYear class function
  - d. Assign a different location to each object using the setLocation class function
  - e. Display the Name of each object using the getName function and cout

## Classes and Functions Review Lab # 2

1. Add the class definition for a class named Dog
  - a. In the public section
    - i. Add the prototype for the default constructor for the Dog class
    - ii. Add the prototype for a setBreed function that returns void and takes one string parameter
    - iii. Add the prototype for a setYear function that returns void and takes one integer parameter
    - iv. Add the prototype for a setColor function that returns void and takes one string parameter
    - v. Add the prototype for a getBreed function that returns a string and takes no parameters
    - vi. Add the prototype for a getAge function that returns an integer and takes no parameters
    - vii. Add the prototype for a getColor function that returns a string and takes no parameters
  - b. In the private section
    - i. Add a string attribute named Breed
    - ii. Add a string attribute named Color
    - iii. Add an integer attribute named Age
2. Create the default constructor for the Dog class( remember that it is named the same as the class)
  - a. Assign a blank to Breed
  - b. Assign a blank to Color
  - c. Assign a 0 to Age
3. Create the setBreed class function (remember that it returns void and takes one string parameter)
  - a. Assign Breed = parameter that you create
4. Create the setAge class function (remember that it returns void and takes one integer parameter)
  - a. Assign Age = parameter that you create
5. Create the setColor class function (remember that it returns void and takes one string parameter)
  - a. Assign Color = parameter that you create
6. Create the getColor class function (remember that it returns a string and takes no parameters)
  - a. Return the Color
7. Create the getBreed class function (remember that it returns a string and takes no parameters)
  - a. Return the Breed
8. Create the getAge class function (remember that it returns a string and takes no parameters)
  - a. Return the Age
9. In the main function
  - a. Create two objects of type Dog
  - b. Assign a different Breed to each object using the setBreed class function
  - c. Assign a Age to each object using the setAge class function
  - d. Assign a different Color to each object using the setColor class function
  - e. Display the Breed of each object using the getBreed function and cout

## Classes and Functions Review Lab # 3

1. Add the class definition for a class named Equipment
  - a. In the public section
    - i. Add the prototype for the default constructor for the Media class
    - ii. Add the prototype for a setName function that returns void and takes one string parameter
    - iii. Add the prototype for a setIDNum function that returns void and takes one integer parameter
    - iv. Add the prototype for a setEquipType function that returns void and takes one string parameter
    - v. Add the prototype for a getName function that returns a string and takes no parameters
    - vi. Add the prototype for a getIDNum function that returns an integer and takes no parameters
    - vii. Add the prototype for a getEquipType function that returns a string and takes no parameters
  - b. In the private section
    - i. Add a string attribute named Name
    - ii. Add a string attribute named EquipType
    - iii. Add an integer attribute named IDNum
2. Create the default constructor for the Equipment class( remember that it is named the same as the class)
  - a. Assign a blank to Name
  - b. Assign a blank to EquipType
  - c. Assign a 0 to IDNum
3. Create the setName class function (remember that it returns void and takes one string parameter)
  - a. Assign Name = parameter that you create
4. Create the setIDNum class function (remember that it returns void and takes one integer parameter)
  - a. Assign IDNum = parameter that you create
5. Create the setEquipType class function (remember that it returns void and takes one string parameter)
  - a. Assign EquipType = parameter that you create
6. Create the getEquipType class function (remember that it returns a string and takes no parameters)
  - a. Return the EquipType
7. Create the getName class function (remember that it returns a string and takes no parameters)
  - a. Return the name
8. Create the getIDNum class function (remember that it returns a string and takes no parameters)
  - a. Return the IDNum
9. In the main function
  - a. Create two objects of type Media
  - b. Assign a different name to each object using the setName class function
  - c. Assign a IDNum to each object using the setIDNum class function
  - d. Assign a different EquipType to each object using the setEquipType class function
  - e. Display the Name of each object using the getName function and cout

## Classes and Functions Review Lab # 4

1. Add the class definition for a class named Vacation
  - a. In the public section
    - i. Add the prototype for the default constructor for the Vacation class
    - ii. Add the prototype for a setName function that returns void and takes one string parameter
    - iii. Add the prototype for a setCost function that returns void and takes one integer parameter
    - iv. Add the prototype for a setLocation function that returns void and takes one string parameter
    - v. Add the prototype for a getName function that returns a string and takes no parameters
    - vi. Add the prototype for a getCost function that returns an integer and takes no parameters
    - vii. Add the prototype for a getLocation function that returns a string and takes no parameters
  - a. In the private section
    - i. Add a string attribute named Name
    - ii. Add a string attribute named Location
    - iii. Add an integer attribute named Cost
2. Create the default constructor for the Vacation class( remember that it is named the same as the class)
  - a. Assign a blank to Name
  - b. Assign a blank to Location
  - c. Assign a 0 to Cost
3. Create the setName class function (remember that it returns void and takes one string parameter)
  - a. Assign Name = parameter that you create
4. Create the setCost class function (remember that it returns void and takes one integer parameter)
  - a. Assign Cost = parameter that you create
5. Create the setLocation class function (remember that it returns void and takes one string parameter)
  - a. Assign Location = parameter that you create
6. Create the getLocation class function (remember that it returns a string and takes no parameters)
  - a. Return the Location
7. Create the getName class function (remember that it returns a string and takes no parameters)
  - a. Return the name
8. Create the getCost class function (remember that it returns a string and takes no parameters)
  - a. Return the Cost
9. In the main function
  - a. Create two objects of type Vacation
  - b. Assign a different name to each object using the setName class function
  - c. Assign a Cost to each object using the setCost class function
  - d. Assign a different location to each object using the setLocation class function
  - e. Display the Name of each object using the getName function and cout