COSC 1437 (DL)- Fall 2016
Quiz 3- Chapters 13-15 & Notes
Total Points:

Due: Sunday November 27th @ 11:59PM. Look at Syllabus/ICR about late work.

Directions: For Questions 1-25, clearly mark answers on a separate word (or notepad) document. See sample file/directions provided by your professor and submit to the appropriate location on the MyTCC (BlackBoard) site.

– Assume all variables are properly declared- unless otherwise mentioned.

Multiple Choice. Mark the one best answer for each question. **(2 pts. each)**

Short Answer. Clearly mark answers as directed. Partial Credit will be given. (**20 @ 2 each)**

1. Static functions ____ access non-static variables.
   A. can
   B. cannot
   C. should
   D. should not

Use the following UML class diagram to answer Questions 2-4

clockType
-hr: int -min: int -sec: int
+setTime(int, int, int): void +getTime(int&, int&, int&) const: void +printTime() const: void +incrementSeconds(): int +incrementMinutes(): int +incrementHours(): int +equalTime(const clockType&) const: bool

2. According to the UML class diagram above, how many private members are in the class?
   A. none
   B. zero
   C. two
   D. three

3. The word ____ at the end of the member functions in the accompanying class `clockType` above specifies that these functions cannot modify the member variables of a `clockType` object.
   A. `static`
   B. `const`
   C. `automatic`
   D. `private`

4. Consider the UML class diagram shown above. Which of the following is the name of the class?
   A. `clock`
   B. `clockType`
   C. `Type`
   D. `+clockType`

5. Polymorphism in C++ will not work unless _____.
    A. the `<polymorphic>` header file is included
    B. the members of the class are public
    C. pointers or references are being used
    D. All of the above

6. A member function of a class that only accesses the value(s) of the data member(s) is called a(n) ____ function.
    A. accessor
    B. mutator
    C. constructor
    D. destructor

7. C++ requires that a copy constructor's parameter be
    A. an integer data type.
    B. a reference to an object.
    C. a pointer variable.
    D. None of the above

8. Operators can be overloaded as
    A. non-friends, non-members of a class.
    B. members of a class.
    C. friends of a class.
    D. All of the above

9. Objects are referred to as ____ of a class.
    A. state
    B. members
    C. built-in types
    D. instances

10. The access modifier ____ means a class member cannot be accessed using any statements in any functions that are not also part of the class.
    A. `public`
    B. `private`
    C. `protected`
    D. `static`

11. A default constructor:
    A. Is a constructor that must receive no arguments.
    B. Is the constructor generated by the compiler when no constructor is provided by the programmer.
    C. Does not perform any initialization.
    D. Both A and B

12. When a member function is defined outside of the class declaration, the function name must be qualified with the class name, followed by _____
    A. the `public` access specifier
    B. the scope resolution operator (`::`)
    C. a tilde (~)
    D. a semicolon(`;`)

13. Run-time binding is also known as ____ binding.
    A. static
    B. shallow
    C. dynamic
    D. deep

14. How many destructors can a class have?
    A. no explicit destructors
    B. one
    C. two
    D. any number

15. If a base class has a non-virtual member function named `print`, and a pointer variable of that class is pointing to a derived object, then the code `ptr->print();`
    A. causes a run-time error.
    B. calls the derived `print` function.
    C. calls the base class `print` function.
    D. calls both the derived and base `print` functions.

16. If you overload the binary arithmetic operator `+` as a member function, how many objects must be passed as parameters?
    A. none
    B. one
    C. two
    D. three

17. Given the following class definition, how would you declare an object of the class, so that the object automatically called the default constructor?

```
class ItemClass
{
public:
     ItemClass();
     ItemClass(int newSize, double newCost);
     int getSize();
     double getCost();
     void setSize(int newSize);
     void setCost(double newCost);
private:
     int size;
     double cost;
};
```

    A. `ItemClass myItem();`
    B. `ItemClass() myItem;`
    C. `ItemClass myItem;`
    D. `ItemClass myItem(1, 0.0);`

18. When you instantiate a class object that has been derived from another class, ____.
    A. the constructor for the base class is called first, followed by the derived class constructor
    B. the constructor for the derived class is called first, followed by the base class constructor
    C. only the constructor for the derived class is called
    D. only the constructor for the base class is called

19. What is the correct signature for the `=` operator function?
    A. `Rational operator=(const Rational &secondRational);`
    B. `operator=(const Rational &secondRational);`
    C. `Rational operator==(const Rational &secondRational);`
    D. `operator==(const Rational &secondRational);`

20. A ____ sign in front of a member name on a UML diagram indicates that this member is a `protected` member.
    A. `+`
    B. `-`
    C. `#`
    D. `$`

Short Answer. Clearly mark answers as directed. Partial Credit will be given. (**10 @ 2 each)**

21. Consider the following class:

```
class Student
 {
     private:
         int idNum;
         string lastName;
         double gradePointAverage;
 };
```

Add a public function `void displayStudentData()`to the class definition and then write an implementation for the function.

22. Implement a default constructor for the following class:

```
class Employee
{
   private:
      int idNum;
      double hourlyRate;
   public:
      Employee();
      void setIdNum(const int);
      void setHourlyRate(const double);
      int getIdNum();
      double getHourlyRate();
};
```

23. Consider the following class declaration:

```
class Employee
{
   private:
     int idNum;
     double salary;
   public:
     Employee(int, double);
     double operator+(Employee);
};
```

Write an implementation for the `operator+()` function.

24. Write a destructor for the following class:

```
class Classroom
{
  private:
    string *student;
    int numStudents;
    int gradeLevel;
  public:
    Classroom();
    ~Classroom();
    void display();
};

Classroom::Classroom()
{
  int x;
  cout << "What grade level is this class? ";
  cin >> gradeLevel;
  cout << "How any students in this class? ";
  cin >> numStudents;
  student = new string[numStudents];
  for(x = 0; x < numStudents; ++x)
  {
      cout << "Please enter the student's name ";
      cin >> student[x];
  }
}
```

25. Consider the following code fragments:

```
class PetStoreItem
{
     protected:
          int stockNum;
          double price;
     public:
          PetStoreItem(int, double);
};

class PetStoreAnimal : public PetStoreItem
{
   protected:
      int petAge;
   public:
      PetStoreAnimal(int);
};

PetStoreAnimal::PetStoreAnimal(int age)
{
   petAge = age;
}
```

Change the `PetStoreAnimal` constructor to avoid getting an error.

Extra Credit: Implement the following program. Follows same program guidelines and graded on the same scale as program sets. Submit only your `.cpp` file- no test runs/folder required. Partial credit given. (**10 points)**

Create two classes. The first holds customer data-specifically, a customer number and zip code. The second, a class for cities, holds the city name, state, and zip code. Additionally, each class contains a constructor that takes parameters to set the field values. Create a `friend` function that displays a customer number and the customer's city, state, and zip code. Write a C++ program to test the classes and the `friend` function. Place all classes into one file. Output should look similar to below.

Sample Run:

```
1572 from Cary, Illinois, 60013
```

Name the program: `TestFriendship.cpp`, where `XX` are your initials.