CSYE 6200 Exam + Fa 2019

1) Complete the following sentences: [6]
a) The three main principles of object-oriented programming are
___Inheritance___, ___Encapsulation____, and _____Polymorphism_____.
b) ___javac_____ compiles Java files into _____ Bytecode ____.
d) Java programs start in a method called ____main___.

2) Fill in the table: [6]

Primitive	Bits
byte	8
char	16
float	32
double	64
short	16
int	32
long	64

3) Write a for loop that counts from 5 to 30 stepping by 5: Answer(For):

```
for (int i = 5; i < 30 (i <= 30); i = i + 5 (i += 5)) { }
```
(+2 initialization, +2 comparison, +2 iteration)

4) Complete the following sentences. [4]
The ____implements___ keyword is used when inheriting from an interface.
The ___extends____ keyword is used when inheriting from a regular class.

5) After the following statements: [2]

```
double turnLength = 3.0;
float ropeLength = 33.5f;
int pCnt = (int) (ropeLength / turnLength);
System.out.println("prodCnt = " + pCnt);
```

What will print on the console? ___ prodCnt = 11___

6) After the showCount() method is executed: [2]

```
public class CTest {
    private int count = 5;
    public void showCount(int count) {
        if (true) count = 8;
        System.out.println("count = " + this.count);
    }}
```

What will print on the console if the method showCount() is called?
___count = 5____

7) Based on the code snippet below, what will display on the console? [5]

```
int choice = 5;
int val = 0;
choice++;
switch (choice) {
    case 5: val = 50; break;
    case 6: val = 60; break;
    case 7: val = 70;
    default:
    case 8: val = 80; break;
}
System.out.println("val = " + val);
```

Answer: _____val = 80 (80[3])_____

8) Overriding methods is an example of which Object-Oriented Programming tenant? [2]
Best answer: ____ inheritance _____

9) Overloading methods is an example of which Object-Oriented Programming tenant? [2]
Best answer: ____ polymorphism _____

10) Complete the following sentence: [2]
In Java, to prevent a method from being overridden, use the keyword _____
final/static/private _____

11) After the following statements: [5]

```
int counter = 0;
boolean done = false;
while (!done) {
    if (counter == 3) continue;
    char outC = (counter < 3) ? 'X' : 'O';
    System.out.print(outC + "-");
    counter++;
    if (counter == 5) break;
    if (counter > 5) done = true;
}
```

What will print on the console? _____X-X-X- (X-X-X [3])_____

12) For the following method, what should the last line be? [2]
(Note: write your answer in the space shown below.)

```
public double findArea(double width, double length) {
    double area = length * width;
    if (area < 0.0) area *= -1.0;
    ______return area____ ;

}
```

13) Complete the following sentences: [4]
Java defines one special superclass called ___Object_____ that is the parent of all other classes.
The _____this_____ keyword is an implicit argument that is passed to each method call, providing a reference to the calling object.

14) "One interface, multiple methods" is a key tenant of Java. What feature best exempli? [2]
Answer: ___ Polymorphism _____

15) Write a public method to calculate the total area. Finish the method shown below: [10]

```
public class DNAList {
    private String fragmentString = "ACGTGACAGT";
    public void printReverse(String input) {
        int stringLen = input.length();
        if(stringLen == 0) return;
        System.out.print(input.charAt(stringLen - 1));
        printReverse(input.subString(0, stringLen - 1));
    }
}
```

16) Complete the following sentences [4]

In Java, the name of a method plus its parameter list is call a _____signature____ .

A class with one or more abstract methods is said to be a ___abstract ___ class.

17) For the following code, fill in the blanks so that the ProductRegistry class exhibits a Singleton design pattern: [10]

```
public class ProductRegistry {
    private String dataFileName = "dataFile.txt";
    __ private ____ __ static__ ProductRegistry instance = null ;
    ___ private ____ ProductRegistry() {
    }
    public ____ static ____ ProductRegistry instance() {
    if (instance == null )
        instance = new ProductRegistry();
    return __ instance; __
    }
    public File getDataFile() throws IOException {
        File file = new File(dataFileName);
        return file;
    }
}
```

18) To create an instance of the ProductRegistry class (from Q. 17), and open the data file, what code should you write (fill in blanks below)? [6]

```
public class Z { …
        private ProductRegistry registry =
                __ProductRegistry.instance();___
        public BufferedReader openDataFileReader() {
            try {
                File file = registry.getDataFile();
                return new BufferedReader(file);
            } catch (_____ Exception _____ ex) { … }
        } … (any reasonable exception is allowed)
```

19) Fill in the blanks [10]

a) To make a member variable visible by all instances, use the __static __ keyword .

b) When passing a primitive type(i.e. **int** or **double**) to a method, Java uses pass-by-___value _____.

c) A static block is called when a class is ___loaded _____.

d) The ___private/static/final ____ keyword is used to prevent a method from being overridden by an inheriting classes.

e) When a method calls itself, this is an example of ____recursion _____ .

20) Use the following code, draw the UML class association diagram. [10]

```
public abstract class DriverRegistry extends Registry {
    protected String regID;
    private ArrayList<Driver> list = new ArrayList<Driver>();
    public void add(Driver driver) { list.add(driver); }
    public Driver getDriver(index) {
        int return (list.get(index));
    }
    public abstract void setReqId( String id );
… // rest of class omitted
}

public class CustomRegistry extends DriverRegistry implements
RegistryIoI {
    private VehicleRegistry vReg = VehicleRegistry.instance();
    @Override
    public Vehicle getVehicle( int index ) {
        Driver driver = getDriver(index);
        return (vReg.getVehicle(driver.getAltDriver());
    }
… // rest of class omitted
}
```

Answer - Draw the UML Class association diagram below:
(Note: interior class details may be omitted. Only draw associations between classes).

[+1 DriverRegistry]
[+1 CustomRegistry]
[+1 RegistryIoI]
[+2 inheritance association arrows]
[+1 Abstract/Interface notation]
[+1 Vehicle class w/weak association]
[+1 Driver class w/weak association]
[+2 Collection association for Drivers]