CIS 121 - Exam 2 - Friday, July 16, 2004

Name: ________________________________  J Number: ________________________________

1. **Fill in the Blank** (30 pts)

1. We can use the method __________________ to add an image to a MediaTracker.

2. We can visualize a __________________ as an array of bytes stored in a file.

3. We can use a __________________ to open an external file in order to read bytes from it.

4. In order for a thread to obtain a lock on an object, it can enter a __________________ method or block of code in the object.

5. A method that is __________________ is not guaranteed to be supported in future releases of a language.

6. The subclasses of __________________ are meant for reading from character files.

7. When using the readLine() method to read from a BufferedReader, we know we are done reading when that method returns __________________.

8. A running thread leaves the running state and enters the waiting state by calling the method __________________.

9. The unchecked exceptions are __________________, __________________, and their subclasses.

10. In order to get a reference to an applet’s code, we use the method __________________.

11. A system is said to be __________________ if it can provide either full or reduced functionality even after an error or failure has occurred.

12. __________________ is a dynamic binding that occurs at runtime between a method call and its correct implementation.

13. If we have an instance of a Calendar called c, then to obtain the current minute, we use the code c.get(______________________).
14. When an exception occurs we say that an exception has been _________________.

15. When we call the method start() on a Thread, it enters the ________________ state.

16. In order to make the currently executing thread pause for one minute, we would use the code Thread.sleep(____________________).

17. The method _________________ when called in an applet is a request to the context of the applet to call it’s paint method as soon as possible.

18. The head of the branch of the Java tree which describes exceptional behaviors is _________________.

19. If a method might cause a checked exception to occur, we can sometimes declare that by using a _________________ clause in the method header.

20. The method ________________ informs only one waiting thread to leave the waiting state and go back to the ready state.

21. _________________ and _________________ are two classes in the javax.swing package that are not lightweight components.

22. Given an ObjectOutputStream object, we can write the bytes of an object created from a Serializable class definition to a file with the method _________________.

23. Serializable is a _________________ interface since it doesn’t contain any methods or constants.

24. Subclasses of _________________ are usually errors during programming that we can recover from.

25. When calling the method readObject on an ObjectInputStream, we need to deal with a potential _________________ in addition to a possible IOException from just opening a connection to a file.

26. We use the method _________________ to get a reference to the currently executing thread.

27. If we create a FileWriter with a constructor that only accepts one String parameter, then if the file exists already, it will be _________________.

28. Java supports a _________________ model of exception handling.
II. Matching (10 pts) Please choose the most appropriate choice for each term from the candidates on the right.

29. -1 ______  A. A superclass of exceptions that we normally can recover from.
30. Unchecked exceptions ______  B. A superclass of exceptions that we normally can’t recover from.
31. Runnable ______  C. The package that contains the AudioClip class and the Calendar class.
32. java.applet ______  D. The method used within an applet to retrieve information sent in through a param tag.
33. Error ______  E. A piece of code that executes once an exception occurs.
34. Exception Handler ______  F. An abstract superclass in the java.io package whose subclasses are meant to read from character files.
35. throw ______  G. A piece of code that causes the program to exit anytime a RuntimeError occurs whether we have a catch block or not.
36. getParameter ______  H. All classes except Error, RuntimeException, and their subclasses.
37. Writer ______  I. The value returned from the read method on a BufferedReader when there are no more characters to read.
38. finally block ______  J. A block of code attached to a try or try-catch that always executes no matter what.
K. An abstract superclass in the java.io package whose subclasses are meant to write to character files.
L. All subclasses of Exception.
M. The first valid position in a String.
N. Causes an exception to occur.
O. The package that contains the Calendar class.
P. Error, RuntimeException, and their subclasses.
Q. The interface a class definition implements in order to become a deprecated class definition.
R. The interface a class definition implements in order to run in its own thread.
S. A method in a JFrame used to load a parameter from HTML Code that didn’t load the JFrame.
T. Precedes a list of checked exceptions a method might cause.
U. The package that contains the AudioClip class.
V. A block of code attached to a try or try-catch that may or may not execute.
III. Short Answer (30 pts)

39. (5 pts) What are the elements of the data hierarchy? If there are counterparts for the various elements in Java, list those.

40. (3 pts) What are three properties of a well-defined recursive method?

41. (4 pts) In what situations is a finally block optional? When is it not optional?

42. (5 pts) Suppose we have a class definition called MyClass which is abstract and contains an abstract method called draw. Is it possible to have an instance of MyClass called myClass and call the draw method on it? If so, how is this possible? What is this an example of?

43. (3 pts) What are three ways for a thread in the waiting state to go back to the ready state?
44. (4 pts) Three applet instance methods walk into a bar. The bartender calls them Larry, Moe, and Curly. Larry says that he is the first method called by the applet’s context when the applet is executed. Curly says he’s sick of getting called everytime there is a request to repaint the applet. Moe says that he’s also tired of being called whenever the applet’s page is returned to. What methods are represented by Larry, Moe, and Curly?

45. (4 pts) Suppose we have a class definition called MyClass which does not contain an extends clause nor a constructor in its class definition. Assuming that the Object class has a no-argument constructor, which of the following assignment statements is correct syntax? Explain.

a) MyClass myClass = new Object();

b) Object o = new MyClass();

46. (1 pt) Given that String is a subclass of Object, does the following make any sense? Explain.

throw “Hello”;

47. (1 pt) Given an instance of an Image called image, how would we create an image whose width and height are 1/2 the width and height of the original?
IV. Problem Solving and Coding (30 pts)

48. (5 pts) Is there any syntax error with the following code? If so, what is the problem?

```java
try {
    int num = Integer.parseInt("Hello");
} catch (Throwable t) {
    System.out.println(t);
} catch (Exception e) {
    System.out.println(e);
}
```

49. (5 pts) In the following code we have a BufferedInputStream called input. We want to read all of the bytes from the BufferedInputStream and eventually have them in an array referred to by bytes. What is wrong with the following code? Assume this is called from within a try block that has a catch block that catches IOException.

```java
byte[] bytes = new byte[0];
byte[] buffer = new byte[4000*1024];
int length = 0;
while ((length = input.read(buffer,0,buffer.length)) != -1) {
    byte[] temp = new byte[bytes.length+buffer.length];
    System.arraycopy(buffer,0,temp,0,buffer.length);
    System.arraycopy(buffer,0,temp,buffer.length,buffer.length);
    bytes = temp;
}
input.close();
```
50. (5 pts) Suppose that we have a class definition, MyClass, which contains the method `getObject` and suppose this method potentially causes a `ClassNotFoundException` which is a subclass of `Exception` but not of `RuntimeException`. If we have a reference to a `MyClass` object called `myClass`, is there a syntax error when a class definition containing the following method is compiled? Explain.

```java
public String toString() {
    String output = myClass.getObject().toString();
    return(output);
}
```

51. (5 pts) Is there any syntax error in the following code? If not, how many times is the word Hello printed when the code is interpreted? Explain.

```java
import java.io.*;
public class Question51 {
    public static void main(String[] args) {
        try {
            FileReader file = new FileReader("review.txt");
        } catch (IOException ie) {
        } finally
            System.out.println("Hello");
        if (true)
            throw new RuntimeException();
        System.out.println("Hello");
    }
}
```
52. (5 pts) Write a class definition for a JFrame which runs in its own thread. The activity of the thread is to print out integers beginning at 0. Assume that you have at your disposal a method to create a JButton which accepts a label, width, height, xlocation, ylocation, and an ActionListener. Add two JButtons, suspend and resume, to the JFrame's definition so that suspend will cause the thread to go into the waiting state, and resume will move the thread from the waiting state to the ready state. Recall that the ActionListener interface contains the method public void actionPerformed(ActionEvent e).

53. (5 pts) Write a class definition called MyClass which has two instance variables, an int and a String. In this class definition you should place an instance method which will write the current instance of the class to a file called “output”. Make sure you include all necessary code.