University College Dublin
An Coláiste Ollscoile, Baile Átha Cliath

Professional Java Programming Part 2 (COMP41620) Exam, 09 November 2012

Prof. Liam Murphy and Prof. John Murphy

NAME: ____

STUDENT NUMBER: ____

Instructions:

Answer ALL 30 questions. Clearly mark your choice(s) for each question on this exam paper. If you want to change your answer, please ensure that your final choice(s) is/are clearly marked.

Do NOT detach pages from this exam, and do NOT add anything - only your indicated choices will be marked, there is no need to provide any explanation.

This is a closed-book exam. You may bring some blank sheets into the exam (for rough work) but you should NOT submit them with your exam answers.

Time allowed: 90 minutes.

1. Consider the following code fragment:

```
public class Q1{
  public static void main(String[] args) {
      int[] x = {6, 0, 4, 9, 0};
      try{
        System.out.println("x[6]: " + x[6]);
        } catch (ArrayIndexOutOfBoundsException oe) {
          System.out.println("Array index out of bound!" );
        }
          catch (IndexOutOfBoundsException ie) {
          System.out.println("Some kind of index out of bound!");
        }
        finally {
          System.out.println("finally block must be executed!");
        }
   }
}
```

Which *one* of the following is the output of this code?

A. Array index out of bound!
finally block must be executed!

B. Array index out of bound!
Some kind of index out of bound!
finally block must be executed!

C. Some kind of index out of bound!
finally block must be executed!

D. No output – a compiler error occurs

2. **Suppose a method called `methodTest()` consists of a `try` block, followed by a `catch` block, followed by a `finally` block. Assuming the JVM does not crash and the code does not execute a `System.exit()` call, under what circumstances will the `finally` block *not* begin to execute? (Choose all that apply)**

A. The `try` block throws an exception, and the `catch` block also throws an exception.

B. The `try` block throws an exception that is not handled by the `catch` block.

C. Given the above conditions, the `finally` block will **always** execute.

3. True or False: if a method `m()` throws an exception, it is legal for a method which overrides `m()` to not throw any exception.

A. True

B. False

4. Consider the following code fragment:

```
1. public class FinallyTest{
2. public static void main(String[] args) {
3. try{
4. System.out.println ("I was in try");
5. }
6. finally {
7. System.out.println("I was in finally");
8. } } }
```

What is the result of executing this code? (Select the correct answer)

A. I was in try

B. I was in finally

C. I was in try
I was in finally

D. A compiler error occurs at line 6.

E. The program compiles, but throws an exception during execution.

5. Assume that the variable `x` is already properly declared, has some value, and is in scope. Which *one* of the following code fragments is the most appropriate way of throwing an exception?

A.

```
if ( x > 10) {
  throw new IndexOutOfBoundsException("The value of index x=" + x + " is out of bounds!");
}
```

B.

```
if ( x > 10) {
  throws new IndexOutOfBoundsException("The value of index x=" + x + " is out of bounds!" );
}
```

C.

```
IndexOutOfBoundsException iob = new IndexOutOfBoundsException("Index out of bound!");
if ( x > 10) {
  throw iob;
}
```

D.

```
if ( x > 10) {
  throw ("Index is out of bound!");
}
```

6. When is it appropriate to write Java code that constructs and throws an error? (Choose all that apply)

A. When a public method’s preconditions are violated

B. When a public method’s postconditions are violated

C. When a static method’s preconditions are violated

D. When a static method’s postconditions are violated

E. It is never appropriate for application programmers to construct and throw errors

7. Which of the following is true about assertions in Java? (Choose all that apply)

A. Assertions are mostly used during testing to uncover internal program errors.

B. An AssertionError is thrown if the ***condition*** specified in ***assert: <condition>*** is true.

C. An AssertionError is thrown if the ***condition*** specified in ***assert: <condition>*** is false.

D. When you enable or disable assertions, you have to re-compile your Java program.

8. Which of the following is true about file handling in Java? (Choose all that apply)

A. When you construct an instance of `File`, if you do not use the file-naming semantics of your platform, the constructor will throw an `IOException`.

B. When you construct an instance of `File`, if the corresponding file does not exist on your file system, one will be created.

C. When you construct an instance of `File`, it can be used to represent a file or a directory.

D. None of the above.

9. Which *one* of the following methods of `java.io.File` will delete a file from the hard drive?

A. **`delete()`**

B. **`deleteFile()`**

C. **`remove()`**

D. **`removeFile()`**

E. Java's **`File`** class does not allow a method to delete files from the hard drive.

10. How many 8-bit bytes does the following Java code fragment write to file *tester*?

```
try {
  FileOutputStream fos = new FileOutputStream(“tester”);
  DataOutputStream dos = new DataOutputStream(fos);
  dos.writeByte(-3);
  dos.writeFloat(1.0001f);
  dos.close();
  fos.close();
}
catch (IOException e) { }
```

A. 2

B. 5

C. 9

D. Compiler error, so no bytes written to the file.

11. What is output when this code is compiled & run? Select the <u>two</u> correct answers.

```
public class Q11 {
public static void main(String args[]) {
    String s1 = "abc"; String s2 = new String (" abc");
    if(s1 == s2) System.out.println(1);
    else System.out.println(2);
    if(s1.equals(s2)) System.out.println(3);
    else System.out.println(4);
    }
}
```

A. 1

B. 2

C. 3

D. 4

12. Consider the following code:

```
1. import java.io.*;
2. public class Q12 {
3.   public static void main(String[] args) throws IOException {
4.      File inputFile = new File("scjp.txt");
5.      File outputFile = new File("scjpcopy.txt");
6.      BufferedReader in = new BufferedReader(inputFile);
7.      BufferedWriter out = new BufferedWriter(new FileWriter(outputFile));
8.      String lineOut;
9.      while ((lineOut = in.readLine()) != null){
10.       out.write(lineOut);
11.       out.newLine();
12.     }
13.   in.close();
14.   out.close();
15.   }
16. }
```

What is the output of this code? (Choose the correct answer)

A. A compiler error occurs at line 6.

B. The code compiles fine but throws an exception during execution at line 6.

C. A compiler error occurs at line 7.

D. The code compiles fine but throws an exception during execution at line 7.

E. The code compiles and executes without any error or exception.

13. Consider the line of code: `String str = new String("Hi");`

Which of the following modify the String to which `str` refers? (Choose all that apply)

A. `str.concat("there");`

B. `str.replace('H', 'M');`

C. `str.toLowerCase();`

D. None of the above

14. Consider the following code fragment:

```
public class MyStringClass {
  public static void main(String[] args) {
    String str = "Me" + " too";
    System.out.println(str);
  }
}
```

Which *one* of the following is true about this code fragment?

A. The code compiles and executes fine, and generates the output “Me too”.

B. No output – a compiler error occurs.

C. Compiles fine, but there is no output and an exception is thrown at runtime.

15. Which *one* of these method calls would return the default locale for the JVM being used?

A. `Locale.getLocale()`

B. `Locale.getDefaultLocale()`

C. `Locale.getDefault()`

16. Which *one* of the following tells a Scanner called `sc` to use a single digit as a delimiter?

A. `sc.useDelimiter(“d”);`

B. `sc.useDelimiter(“\d”);`

C. `sc.useDelimiter(“\\d”);`

D. `sc.useDelimiter(“d+”);`

17. What is the output of this code?

```
public class WrapperTest {
        public static void main(String[] args) {
              Boolean bool1 = new Boolean(true);
              Boolean bool2 = new Boolean(false);
              Boolean bool3 = new Boolean("false");
              Boolean bool4 = new Boolean(bool1);
              System.out.print(bool1.equals(bool4));
              System.out.print(" " + (bool2 == bool3));
              System.out.print(" " + (bool1 == bool4));
        }
   }
```

A. The program will not compile because the creation of `bool4` will cause a compilation error.

B. The program compiles and produces the output ***true false false***

C. The program compiles and produces the output ***true true true***

D. The program compiles and produces the output ***true false true***

18. What happens when you try to compile and run the following application?

```
1. import java.util.*;
2.
3. public class Q18 {
4.   public static void main(String[] args) {
5.     Set<Q18> set = new TreeSet<Q18>();
6.     set.add(new Q18());
7.     set.add(new Q18());
8.   }
9. }
```

A. Compiler error.

B. An exception is thrown at line 5.

C. An exception is thrown at line 6.

D. An exception is thrown at line 7.

E. Compiles and runs fine – no exception is thrown.

19. Which of the following statements about the `hashCode()` method in Java are *true*? (Choose all that apply)

A. The **`hashCode()`** method is implemented in the **`Object`** class.

B. If two objects are equal according to the **`equals()`** method, then invoking **`hashCode()`** on those objects must return the same hashcode value in each case.

C. If two objects are unequal according to the **`equals()`** method, then invoking **`hashCode()`** on those objects may return a different hashcode value in each case.

20. Given the following code:

```
class ThreadBoth extends Thread implements Runnable {
  public void run(){ System.out.print("hi "); }
  public static void main(String [] args){
    Thread t1 = new ThreadBoth();
    Thread t2 = new Thread(t1);
    t1.run();
    t2.run();
    t1.run();
  }
}
```

What is the result?

A. Prints 'hi hi '

B. Prints 'hi hi hi '

C. Compiler error.

D. Compiles ok, but an exception is thrown at runtime.

21. True or False: A `LinkedList` provides constant-time access to a specific element in the list, but insertions and deletions are linear in time.

A. True

B. False

22. Which of the following are methods of the `Object` class? (Choose all that apply)

A. `sleep()`

B. `run()`

C. `wait()`

D. `notify()`

23. Which *one* of the following statements is false about the `wait()` method?

A. The **`wait()`** method can be invoked with an argument representing a time duration.

B. When a thread executes a call to the **`wait()`** method, it itself temporarily stops executing.

C. A call to **`wait()`** stops the execution of the application.

24. Which of the following statements about the `wait()` and `notify()` methods are true? (Choose all that apply)

A. The **`wait()`** and **`notify()`** methods can be called outside synchronized code.

B. The programmer can specify which thread should be notified in a **`notify()`** call.

C. The thread that calls **`wait()`** goes into the monitor's pool of waiting threads.

D. The thread that calls **`notify()`** gives up the lock.

25. When does an exception's stack trace get recorded in the exception object? (Choose all that apply)

A. When the exception is thrown.

B. When the exception is caught.

C. When the exception's **`printStackTrace()`** method is called.

D. When the exception is constructed.

26. Consider the following code:

```
1. import java.util.*;
2. public class Q26{
3.   public static void main(String[] args) {
4.     Integer x = 1;
5.     x++;
6.     Integer y = 2;
7.     if(x==y){
8.       System.out.println("Area: " + areaOfASquare(4.0d));
9.     }
10.  }
11.  public static Double areaOfASquare(Double side){
12.    return side*side;
13.  }
14. }
```

What is the result when you attempt to compile this code?

A. compiler error at line 4

B. compiler error at line 7

C. compiler error at line 8

D. compiles fine (no errors)

27. When is it appropriate to pass a cause to an exception's constructor? (Choose all that apply)

A. Always

B. When the exception is being thrown in response to the catching of a different exception type

C. When the exception is being thrown from a public method

D. When the exception is being thrown from a private method

28. Consider the following code fragment:

```
1. ArrayList<Integer> list = new ArrayList<Integer>();

2. list.add(new Integer(1));

3. list.add(new Integer(2));

4. Iterator<Integer> itr = list.iterator();

5. for(Integer i:list){

6.    System.out.println("number: " + i);

7. }
```

What is the output of this code fragment?

A. ***number: 1***

number: 2

B. Compiler error at line 3

C. Compiler error at line 4

D. Runtime error

29. True or false: `Thread(String s)` is a legal `Thread` constructor in Java.

A. True

B. False

30. True or false: Java's `String` class cannot be subclassed in your code using `extends`.

A. True

B. False