JAVA Programming Language Homework VI: Threads & I/O

ID: Name:

1. When comparing java.io.BufferedWriter to java.io.FileWriter, which capability exists as a method in only one of the two?

A. Closing the stream
B. Flushing the stream
C. Writing to the stream
D. Marking a location in the stream
E. Writing a line separator to the stream

ANS:

2. Which of the following is true?

A. A program will terminate only when all user threads stop running.
B. A program will terminate only when all daemon stop running.
C. A daemon thread always runs at Thread.MIN_PRIORITY.
D. None of the above.

ANS:

3. Given the following Java code:

```
1.  class B implements Runnable{
2.      public void run() {}
3.  }
4.  class A {
5.      public static void main(String[] args) {
6.          Thread my1 = new Thread() ;
7.          Thread my2 = new Thread("B") ;
8.          Thread my3 = new Thread(new B()) ;
9.          Thread my4 = new Thread("B", new B()) ;
10.     }
11. }
```

What is the result?

A. A compile-time error is generated at line 6

B. A compile-time error is generated at line 7

C. A compile-time error is generated at line 8

D. A compile-time error is generated at line 9

E. None of the above

ANS:

4. Given the following Java code: [5 points]

```
1.      class B extends Thread {
2.          public String x;
3.          B(String in) {
4.              x = in;
5.          }
6.          public void run() {
7.              for(int i=1; i<5; i++) {
8.                  System.out.println(x+"-"+i);
9.              }
10.         }
11.     }
12.
13.     class A {
14.         public static void main(String[] args) {
15.             B obj1 = new B("o");
16.             B obj2 = new B("x");
17.             obj1.setPriority(1);
18.             obj2.setPriority(10);
19.             obj1.start();
20.             obj2.start();
21.         }
22.     }
```

Which of the following is true?

A. This program will go exception when compiling.

B. Obj1 runs at Thread.MIN_PRIORITY.

C. Obj2 runs at Thread.MIN_PRIORITY.
D. The Thread.setDaemon method can change Thread.MIN_PRIORITY.
E. None of the above.

ANS:

5. Given the following Java code:

```
1.      class A extends Thread {
2.          private String i;
3.          public void run() {
4.              i = “A”;
5.          }
6.          public static void main(String[ ] args) {
7.          A a = new A();
8.          a.start();
9.          System.out.print(a.i);
10.         }
11.     }
```

Which of the following are possible results of attempting to compile and run the program?
A. prints: A
B. prints: 0
C. prints: null
D. prints: i
E. Compile-time error

ANS:

6. Given the following Java code:

```
1.      public B extends Thread {
2.          public void run() {
3.              System.out.print(“A”);
4.          }
5.      }
6.      class A {
```

```
7.          public static void main (String[] args) {
8.              B obj = new B() ;
9.              obj.start() ;
10.             obj.start() ;
11.         }
12.     }
```

What is the result of attempting to compile and run the program?

A. The program compiles and runs fine but prints nothing.

B. prints: A

C. Compiler error

D. An IllegalThreadStateException is thrown at run-time

E. None of the above

ANS:

7. Given the following Java code:

```
1.      public class Hello implements Runnable {
2.          public void run () {
3.              System.out.print ( "running" );
4.          }
5.          public static void main ( String[] args ) {
6.              Thread t = new Thread ( new Hello());
7.              t.run ();
8.              t.run ();
9.              t.start ();
10.         }
11.     }
```

What is the result?

A. Compilation fails

B. An exception is thrown at runtime

C. The code executes and prints "running"

D. The code executes and prints "runningrunning"

E. The code executes and prints "runningrunningrunning"

ANS:

8. Chain these constructors to create objects to read from a file named "in" and to write to a file named "out".

```
1.      Reader = [1. place here] [2. place here] "in" ) );
2.      Writer = [3. place here] [4. place here] [5. place here] "out" ) );
```

Constructors:

A. new FileReader (	B. new PrintReader (	C. new BufferedReader (
D. new BufferedWriter (	E. new FileWriter (	F. new PrintWriter (

Which sequence is correct?

A. CAFDE

B. ACDFE

C. CAEDF

D. CBFDE

E. BCDFE

ANS:

9. Place the code fragments into position to use a BufferedReader to read in an entire text file.

```
1.      class PrintFile {
2.          public static void main (String[] args) {
3.              BufferedReader buffReader = null;
4.              // more code here to initialize buffReader
5.              try {
6.                  String temp;
7.                  while( [1. place here] [2. place here] ) {
8.                      System.out.println(temp);
9.                  }
10.             } catch [3. place here]
```

```
11.                 e.printStackTrace();
12.             }
13.         }
14.     }
```

Code Fragments:

A. (temp = buffReader.readLine ())	B. && buffReader.hasNext ()
C. (temp = buffReader.nextLine ())	D. (IOException e) {
E. ! = null	F. (FileNotFoundException e) {

Which sequence is correct?

A. AED

B. AEF

C. ABD

D. CBF

E. CED

ANS:

10. Place the Fragments into program, so that the program will get lines from a text file, display them, and then close the resources.

```
1.  import java.io.*;
2.  public class ReadFile {
3.      public static void main (String[] args) {
4.      try {
5.          File x1 = new File("MyText.txt");
6.          [1. Place here] x2 = new [2. Place here](x1);
7.          [3. Place here] x4 = new [4. Place here](x2);
8.          String x3 = null;
9.          while(( x3 = x4.[5. place here]()) != null ) {
10.             System.out.println(x3);
11.         }
12.         x4.close();
13.     }
14.     catch (Exception ex) {
```

```
15.          ex.printStackTrace ();
16.      }
17.      }
18.  }
```

Code Fragments:

A. BufferedReader	B. StreamReader	C. FileReader	D. readLine
E. readLn	F. read	G. closeFile	F. close

Try to fill them:

1._____ 2._____ 3._____ 4._____ 5._____

ANS: