1) write reverse() function. Should be efficient..

List<Character>  list = {‘a’,’b’,’c’,’d’,’e’};

traverse(list); {‘a’,’b’,’c’,’d’,’e’};

reverse(list);

traverse(list); {‘e’,’d’,’c’,’b’,’a’};

1)      Which three are methods of the Object class?

1. notify();
2. notifyAll();
3. isInterrupted();
4. synchronized();
5. interrupt();
6. wait(long msecs);
7. sleep(long msecs);
8. yield();

2)  What will be the output of the program?

public class ArrayTest

{

    public static void main(String[ ] args)

    {

        float f1[ ], f2[ ];

        f1 = new float[10];

        f2 = f1;

        System.out.println("f2[0] = " + f2[0]);

    }

}

3)  public class A

{

    void A() /\* Line 3 \*/

    {

        System.out.println("Class A");

    }

    public static void main(String[] args)

    {

        new A();

    }

}

4) Which two are equal?

1. 32/4
2. (8 >> 2) << 4
3. 2^5
4. 128 >>> 2
5. 2 >> 5

5) What will be the output of the program?

public class ThreadTest extends Thread

{

    public void run()

    {

        System.out.println("In run");

        yield();

        System.out.println("Leaving run");

    }

    public static void main(String []argv)

    {

        (new ThreadTest()).start();

    }

}

6)  What will be the output of the program?

class s1 extends Thread

{

    public void run()

    {

        for(int i = 0; i < 3; i++)

        {

            System.out.println("A");

            System.out.println("B");

        }

    }

}

class Test120 extends Thread

{

    public void run()

    {

        for(int i = 0; i < 3; i++)

        {

            System.out.println("C");

            System.out.println("D");

        }

    }

    public static void main(String args[])

        {

        s1 t1 = new s1();

        Test120 t2 = new Test120();

        t1.start();

        t2.start();

    }

}

7) What will be the output of the program?

public class X

{

    public static void main(String [] args)

    {

        try

        {

            badMethod();

            System.out.print("A");

        }

        catch (Exception ex)

        {

            System.out.print("B");

        }

        finally

        {

            System.out.print("C");

        }

        System.out.print("D");

    }

    public static void badMethod()

    {

        throw new Error(); /\* Line 22 \*/

    }

}

8) Which four can be thrown using the throw statement?

1. Error
2. Event
3. Object
4. Throwable
5. Exception
6. RuntimeException

9)  What will be the output of the program?

class A

{

    public A(int x){}

}

class B extends A { }

public class test

{

    public static void main (String args [])

    {

        A a = new B();

        System.out.println("complete");

    }

}

10)  What will be the output of the program?

public class ExamQuestion7

{

    static int j;

    static void methodA(int i)

    {

        boolean b;

        do

        {

            b = i<10 | methodB(4); /\* Line 9 \*/

            b = i<10 || methodB(8);  /\* Line 10 \*/

        }while (!b);

    }

    static boolean methodB(int i)

    {

        j += i;

        return true;

    }

    public static void main(String[] args)

    {

        methodA(0);

        System.out.println( "j = " + j );

    }

}

11)  What will be the output of the program?

try

{

    Float f1 = new Float("3.0");

    int x = f1.intValue();

    byte b = f1.byteValue();

    double d = f1.doubleValue();

    System.out.println(x + b + d);

}

catch (NumberFormatException e) /\* Line 9 \*/

{

    System.out.println("bad number"); /\* Line 11 \*/

}

12)  What will be the output of the program?

public class Test

{

    public static void main(String[] args)

    {

        final StringBuffer a = new StringBuffer();

        final StringBuffer b = new StringBuffer();

        new Thread()

        {

            public void run()

            {

                System.out.print(a.append("A"));

                synchronized(b)

                {

                    System.out.print(b.append("B"));

                }

            }

        }.start();

        new Thread()

        {

            public void run()

            {

                System.out.print(b.append("C"));

                synchronized(a)

                {

                    System.out.print(a.append("D"));

                }

            }

        }.start();

    }

}

13) What will be the output of the program?

class SC2

{

    public static void main(String [] args)

    {

        SC2 s = new SC2();

        s.start();

    }

    void start()

    {

        int a = 3;

        int b = 4;

        System.out.print(" " + 7 + 2 + " ");

        System.out.print(a + b);

        System.out.print(" " + a + b + " ");

        System.out.print(foo() + a + b + " ");

        System.out.println(a + b + foo());

    }

    String foo()

    {

        return "foo";

    }

}

a)  9 7 7 foo 7 7foo

b)  72 34 34 foo34 34foo

c)  9 7 7 foo34 34foo

d)     72 7 34 foo34 7foo

14) What will be the output of the program?

class BoolArray

{

    boolean [] b = new boolean[3];

    int count = 0;

    void set(boolean [] x, int i)

    {

        x[i] = true;

        ++count;

    }

    public static void main(String [] args)

    {

        BoolArray ba = new BoolArray();

        ba.set(ba.b, 0);

        ba.set(ba.b, 2);

        ba.test();

    }

    void test()

    {

        if ( b[0] && b[1] | b[2] )

            count++;

        if ( b[1] && b[(++count - 2)] )

            count += 7;

        System.out.println("count = " + count);

    }

}

15) Which two statements are equivalent?

1. 3/2
2. 3<2
3. 3\*4
4. 3<<2

16) What will be the output of the program?

Float f = new Float("12");

switch (f)

{

    case 12: System.out.println("Twelve");

    case 0: System.out.println("Zero");

    default: System.out.println("Default");

}

17) What will be the output of the program?

public class Test

{

    public static void aMethod() throws Exception

    {

        try /\* Line 5 \*/

        {

            throw new Exception(); /\* Line 7 \*/

        }

        finally /\* Line 9 \*/

        {

            System.out.print("finally "); /\* Line 11 \*/

        }

    }

    public static void main(String args[])

    {

        try

        {

            aMethod();

        }

        catch (Exception e) /\* Line 20 \*/

        {

            System.out.print("exception ");

        }

        System.out.print("finished"); /\* Line 24 \*/

    }

}

18) What will be the output of the program?

class MyThread extends Thread

{

    public static void main(String [] args)

    {

        MyThread t = new MyThread();

        t.start();

        System.out.print("one. ");

        t.start();

        System.out.print("two. ");

    }

    public void run()

    {

        System.out.print("Thread ");

    }

}

19) class X implements Runnable

{

    public static void main(String args[])

    {

        /\* Missing code? \*/

    }

    public void run() {}

}

20)  What will be the output of the program?

class MyThread extends Thread

{

    MyThread() {}

    MyThread(Runnable r) {super(r); }

    public void run()

    {

        System.out.print("Inside Thread ");

    }

}

class MyRunnable implements Runnable

{

    public void run()

    {

        System.out.print(" Inside Runnable");

    }

}

class Test

{

    public static void main(String[] args)

    {

        new MyThread().start();

        new MyThread(new MyRunnable()).start();

    }

}

21) What will be the output of the program?

class s implements Runnable

{

    int x, y;

    public void run()

    {

        for(int i = 0; i < 1000; i++)

            synchronized(this)

            {

                x = 12;

                y = 12;

            }

        System.out.print(x + " " + y + " ");

    }

    public static void main(String args[])

    {

        s run = new s();

        Thread t1 = new Thread(run);

        Thread t2 = new Thread(run);

        t1.start();

        t2.start();

    }

}

22)  What will be the output of the program?

public class Test138

{

    public static void stringReplace (String text)

    {

        text = text.replace ('j' , 'c'); /\* Line 5 \*/

    }

    public static void bufferReplace (StringBuffer text)

    {

        text = text.append ("c");  /\* Line 9 \*/

    }

    public static void main (String args[])

    {

        String textString = new String ("java");

        StringBuffer textBuffer = new StringBuffer ("java"); /\* Line 14 \*/

        stringReplace(textString);

        bufferReplace(textBuffer);

        System.out.println (textString + textBuffer);

    }

}

23)  Is there any error?

public static void main(String[] args)

    {

        byte a = 0;

        short b = 0;

        int c = 0;

        long d = 0;

        float e = 0;

        double f = 0;

        boolean g = false;

        char h = 'a';

        Byte a = 0;

        Short b = 0;

        Integer c = 0;

        Long d = 0;

        Float e = 0;

        Double f = 0;

        Boolean g = true;

        Character h = 'a';

}

24) Which is the Parent class of Wrapper Classes?

25) Difference between String/StringBuffer/StringBuilder?

26) What will be the output of the program (in jdk1.6 or above)?

public class BoolTest

{

    public static void main(String [] args)

    {

        Boolean b1 = new Boolean("false");

        boolean b2;

        b2 = b1.booleanValue();

        if (!b2)

        {

            b2 = true;

            System.out.print("x ");

        }

        if (b1 & b2) /\* Line 13 \*/

        {

            System.out.print("y ");

        }

        System.out.println("z");

    }

}

28)  What will be the output of the program?

Class

{

abc(Obj o)

{

        sout(1)

}

abc(String s)

{

        sout(2);

}

public static void main(String []args)

{

        abc(null);

}

 }

29)  What will be the output of the program?

Class

{

abc(Obj o,String s)

{

        sout(1)

}

abc(Obj o1,String s1)

{

        sout(1);

}

public static void main(String []args)

{

        abc(null,null);

}

 }

30)  What will be the output of the program?

Class

{

abc(Obj o)

{

        sout(1)

}

abc(String s)

{

        sout(2);

}

abc(StringBuffer stbuf)

{

        sout(3);

}

public static void main(String []args)

{

        abc(null);

}

 }