

<p>1. What has to be true about a data set before you can use a binary search algorithm on with it?</p>	<p>2. What is wrong with the following recursive method?</p> <pre>public static int doit(int a) { return a + doit(a-1); }</pre>
<p>3. What is printed by the call review(1)?</p> <pre>public static void review(int a) { if(a<7) review(a+3); System.out.print(a+" "); }</pre>	<p>4. What is printed by the call review(8)?</p> <pre>public static void review(int a) { System.out.print(a+" "); if(a<10) review(a+1); }</pre>
<p>5. What is returned by the call review(10)?</p> <pre>public static int review(int a) { if(a>15) return 1; return a + review(a+3); }</pre>	<p>6. What is returned by the call review(10)?</p> <pre>public static int review(int a) { if(a==8) return 1; return a * review(a-1); }</pre>
<p>5. Write a recursive method changeOX which returns a given String with any occurrence of the letters "ox" replaced with an "oo". So "xbox" returns "xboo".</p> <pre>public static String changeOX(String str) { }</pre>	
<p>6. Write a recursive method addDigitsExcept7s which returns the sum of all the digits in a number of any length except that it skips any 7s. So sending 127 returns 3, and sending 1024 returns, ironically, 7.</p> <pre>public static int addDigitsExcept7(int n) { }</pre>	

7. The following questions deal with this array and a binary search algorithm.

```
int[] array = { 1, 3, 7, 10, 13, 16, 19, 22, 25 };
```

Which is the first value (value, not index) checked when a binary search algorithm is used to search this array for 12?

What is the second value checked?

What is the third value checked?

What is the fourth value checked?

What is the final result returned?

8. Write a recursive method that adds all the integers from 1 up to the integer sent in (assume the parameter is greater than 1). So sending in 3 returns 6 (1 + 2 + 3) and sending in 4 returns 10.

```
public static int addEmUp(int n)
{

}

}
```

9. Write a recursive method that adds all the integers divisible by 3 from 3 up to the integer sent in (assume the parameter is greater than 3). So sending in 6 returns 9 (3 + 6) and sending in 17 returns 45 (3 + 6 + 9 + 12 + 15)

```
public static int iHeart3s(int n)
{

}

}
```

10. How many times will this code print "Hi" if you call sayHi(5) ?

```
public static void sayHi(int n)
{
    if(n>3) sayHi(n-1);
    System.out.println("Hi");
}
```