

Name _____ Period _____

1. Refer to the code below,

```
//Assume nextLine() and nextInt() are static methods in  
//a class named Scanner that reads a String and an integer  
//from the keyboard.
```

```
Scanner rdr = new Scanner(System.in);  
String str = rdr.nextLine();  
int j = rdr.nextInt();  
  
try {  
    System.out.print( str.charAt(j) );  
} catch(StringIndexOutOfBoundsException e) {  
    System.out.print("Error: " + j);  
}
```

(a) What is the output of the code above, given the input below?

big mama
2

/1

(b) What is the output of the code above, given the input below?

big mama
22

/1

2. Refer to the code below,

```
//Returns the product of two integers represented as  
//strings. If either string cannot be parsed,  
//a message indicating why is printed  
//If both are not numbers, returns 1.  
  
public static int product(String str1, String str2) {  
    int prd = 1;  
    try {  
        prd*=Integer.parseInt(str1);  
    } catch(NumberFormatException e) {  
        <*1>  
    }  
    try {  
        prd*=Integer.parseInt(str2);  
    } catch(NumberFormatException e) {  
        <*1>  
    }  
    return prd;  
}
```

(a) What should replace <code><*1></code> in the code above to make it do what the remarks suggest?	/1
(b) What is returned by <code>product ("two", "5")</code> ?	/1

<p>3. What is output by the code to the right if the static method called <code>test()</code> encounters the following line of code? Assume the test signature includes <code>throws NumberFormatException</code>.</p> <pre>int j = Integer.parseInt("Two Thousand");</pre>	<pre>try{ test(); } catch(NumberFormatException e) { System.out.println("Error with number format"); } catch(RuntimeException e) { System.out.println("Error"); }</pre>
	/1

4. If the code designated by <*1> to the right does not throw any exceptions, which of the remaining code sections will execute?	<pre>try { <*1> } catch(RuntimeException e) { <*2> } finally { <*3> }</pre>	
		/1

5. The following code blocks could cause errors. Fix the code to prevent an error from occurring	
<pre>class InfiniteLoop { public static void test(int j) { for(int i = 1; i > 0; i++){ System.out.println(j); } } }</pre>	<pre>class StackOverflow { public static void test(int i) { //Not correct as base condition //leads to non-stop recursion if //if i is positive if (i == 0) return; else test(i++); } }</pre>

	/2

6. For each of the following code segments identify the unchecked error that would occur	
<p>(a) <code>int[] myArray = {1, 2, 3}</code> <code>System.out.println(myArray[3]);</code></p> <p>(b) <code>System.out.println(10/0);</code></p> <p>(c) <code>String pointer = null;</code> <code>if(pointer.equals("this"))</code> <code>//do something</code></p> <p>(d) <code>Object x = new Integer(0);</code> <code>System.out.println((String)x);</code></p> <p>(e) <code>String s = "Hello";</code> <code>System.out.println(s.charAt(5));</code></p>	
	/5

7. For each of the following unchecked errors, write a try-catch block to catch the error.	
<code>int[] myArray = {1, 2, 3}</code> <code>System.out.println(myArray[3]);</code>	

```
System.out.println(10/0);
```

```
String pointer = null;  
if(pointer.equals("this")  
    //do something
```

```
Object x = new Integer(0);  
System.out.println((String)x);
```

```
String s = "Hello";  
System.out.println(s.charAt(5));
```

/10