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 mamma          g
2
```

/1

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

```
big mamma          Error: 22
22
```

/1

2. Refer to the code below,

```
//Returns the product of two integers represented as
//strings. If either string is not a number, returns the other
//number. 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)  {
      <*1>
}  try  {
      prd*=Integer.parseInt(str2);
}  catch(NumberFormatException)  {
      <*1>
}
      return prd;
}
```

(a) What should replace <*1> in the code above to make it do what the remarks suggest?

Nothing

/1

(b) Assume <*1> has been filled in correctly. What is returned by product ("two", "5")?

5

/1

3. What is output by the code to the right if the static method called `test( )` encounters the following line of code? Assume the test signature includes throws NumberFormatException.

```
int j =
Integer.parseInt("Two
Thousand");
```

Error with number format

```
try{
      test( );

}
catch(NumberFormatException e) {
      System.out.println("Error with number
format");
}  catch(RuntimeException e) {
     System.out.println("Error");

}
```

/1

4. If the code designated by <*1> to the right does not throw any exceptions, which of the remaining code sections will execute?

<*3>

```
try  {
      <*1>

}  catch(RunTimeException e)   {
      <*2>

}  finally  {
      <*3>

}
```

/1

5. Indicate the stack and the output for the code below.

Code	Stack	Output

`public static void showMe(int arg)`
`{`
`  if (arg < 10)`
`  {`
`  showMe(arg + 1);`
`  }`
`  else`
`  {`
`  System.out.print(arg + " ");`
`  }`
`}`
`public static void main(String args[]){`
`  showMe(0)`
`}`
	ShowMe(9)
	showMe(8)
	showMe(7)
	showMe(6)
	showMe(5)
	showMe(4)
	showMe(3)
	showMe(2)
	showMe(1)
	showMe(0)
		10

6. Indicate the stack and the output for the code below

Code	Stack	Output

`public static void whatsItDo(String str)`
`{`
`  int len = str.length();`
`  if (len > 1)`
`  {`
`  String temp = str.substring(0, len – 1);`
`  System.out.println(temp);`
`  whatsItDo(temp);`
`  }`
`}`
`public static void main(String args[]){`
`  whatsItDo("WATCH")`
`}`
	whatsItDo("W")
	whatsItDo("WA")
	whatsItDo("WAT")
	whatsItDo("WATC")
	whatsItDo("WATCH")
		WATC
		WAT
		WA
		W

Score _____/18

7. Indicate the stack and the output for the code below

Code	Stack	Output

`public static void puf(int n)`
`{`
`  if(n == 1)`
`  {`
`  System.out.print("x");`
`  }`
`  else if( n%2 = = 0) //n is even`
`  {`
`  System.out.print("{");`
`  puf(n-1);`
`  System.out.print("}");`
`  }`
`  else //n is odd`
`  {`
`  System.out.print("<");`
`  puf(n-1);`
`  System.out.print(">");`
`  }`
`}`
`public static void main(String args[]){`
`  puf(5);`
`}`
	puf(1)
	puf(2)
	puf(3)
	puf(4)
	puf(5)
		<{<{x}>}>

8. Indicate the stack and the output for the code below

Code	Stack	Output

`public static void sort(int[] data)`
`{`
`for (int j = 0; j < data.length - 1; j++)`
`{`
`  int m = j;`
`  for (int k = j + 1; k < data.length; k++)`
`  {`
`  if (data[k] < data[m])`
`  /* Compare values */`
`  {`
`  m = k;`
`  }`
`  }`
`int temp = data[m];`
`/* Assign to temp */`
`data[m] = data[j];`
`data[j] = temp;`
`/* End of outer loop */`
`}`
`}`
`public static void main(String args[]){`
`  int[] iArr = {1, 5, 3};`
`  sort(iArr);`
`}`		

Score _____/18