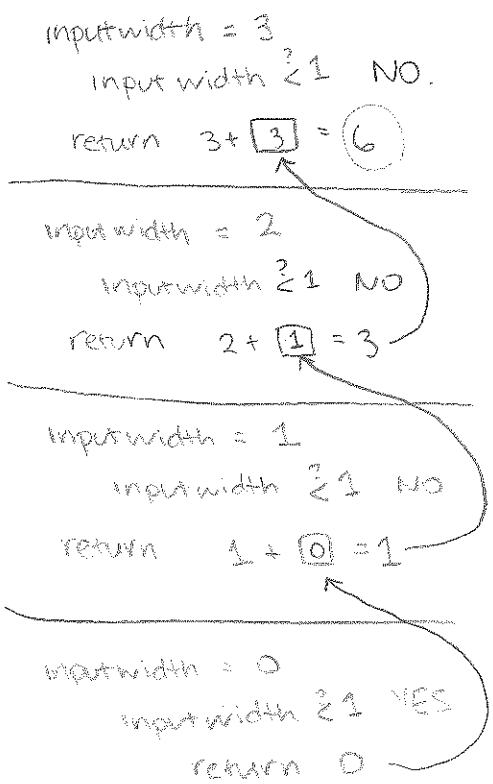
Name: Answer Key **CM:** _____ **Sect:** ____ **Grade:** _____ **of 10**

1. For the following functions, show a trace of the values of each variable for each recursive step:

//We do this one together

```
public int rightTriangleArea(int inputWidth) {
      if (inputWidth < 1)
            return 0;
      return inputWidth + rightTriangleArea(inputWidth - 1);
}
```

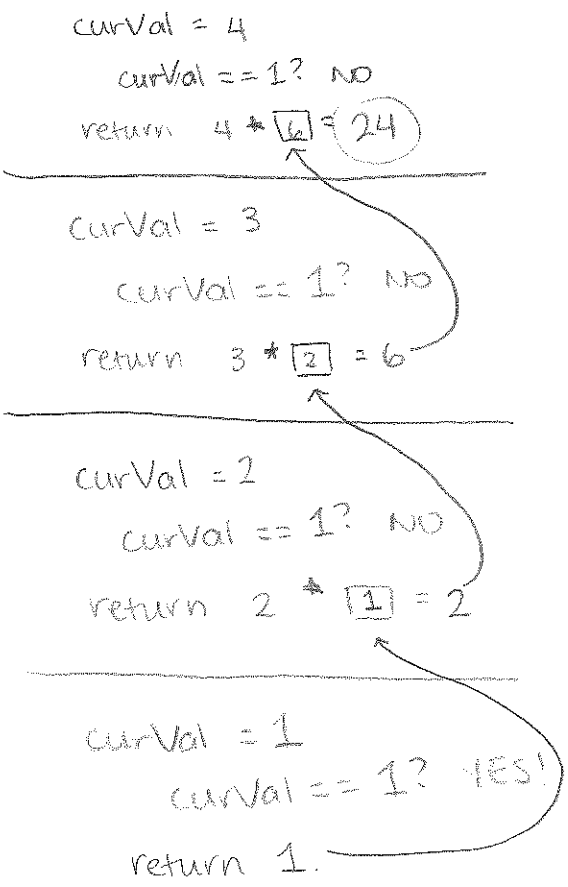
//call from main:
`rightTriangleArea(3);`



//Try this on your own

```
2. public int recursiveFactorial(int curVal) {
      if(curVal == 1)
            return 1;
      return curVal * recursiveFactorial(curVal - 1);
}
```

//call from main:
`recursiveFactorial(4);`

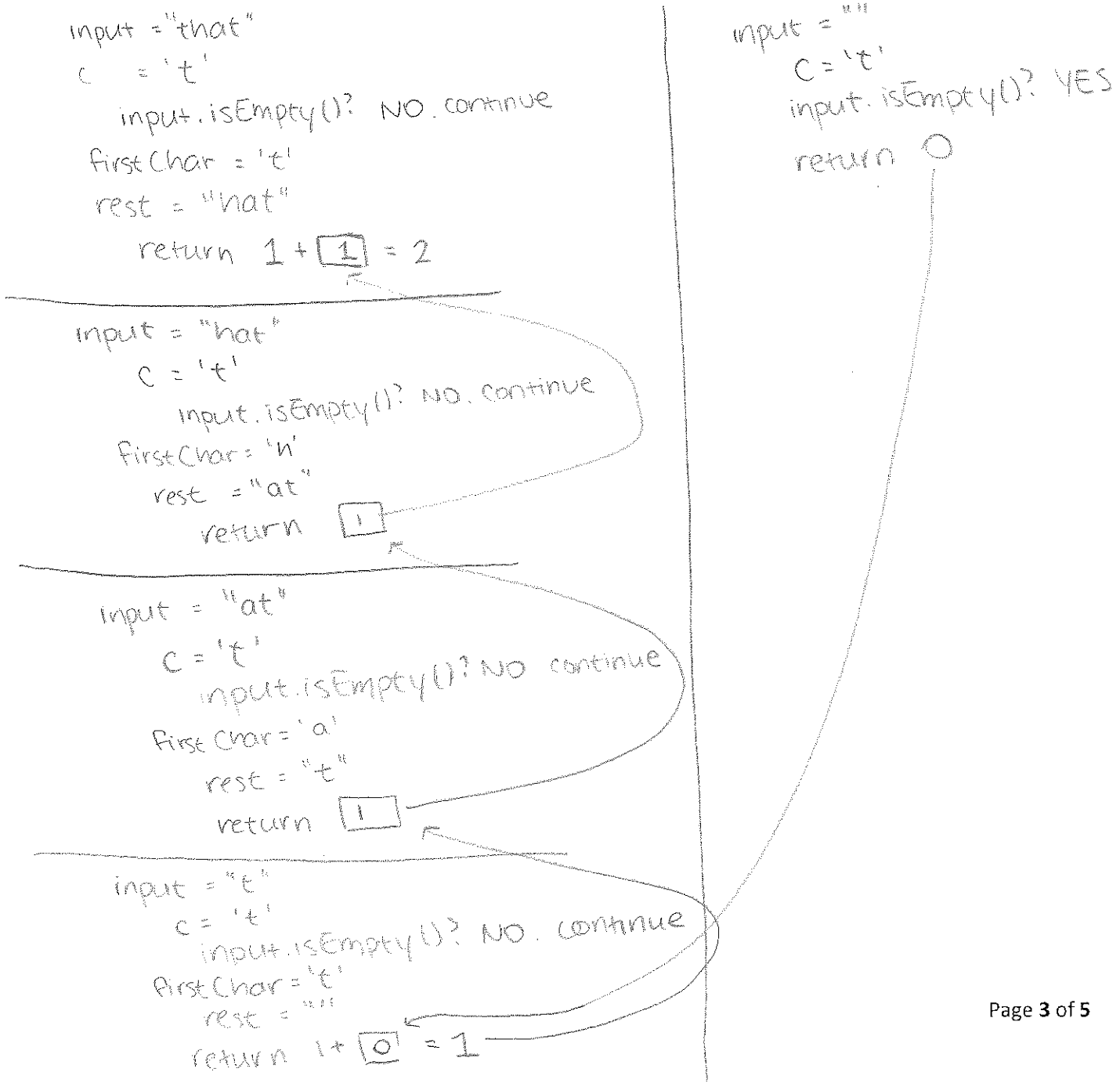


//We do this one together

```
3     public static int letterCount(String input, char c) {
            if(input.isEmpty()) {
                  return 0;
            }
            char firstChar = input.charAt(0);
            String rest = input.substring(1);
            if(c == firstChar) {
                  return 1 + letterCount(rest, c);
            } else {
                  return letterCount(rest, c);
            }
      }
```

//call from main:

```
letterCount("~~Count this letter~~ that", 't');
```



```
//Try this on your own

4. public static int findMax(ArrayList<Integer> listOfInts) {
      if(listOfInts.size() == 0)
            return Integer.MIN_VALUE;
      if(listOfInts.size() == 1)
            return listOfInts.get(0);

      int first = listOfInts.get(0);
      int size = listOfInts.size();
      int second = findMax(new ArrayList<Integer>(listOfInts.subList(1, size)));

      return (first > second) ? first : second;
}

// hint: that last line is equivalent to

// if(first > second) { return first; } else {return second; }

//in main this method is called with an array list of [1, 55, 97, 43074, 3]
```

listOfInts = [1, 55, 97, 43074, 3]
listOfInts.size == 0? No. continue
listOfInts.size == 1? NO. continue
first = 1
size = 5
second = 43074
return 43074

listOfInts = [55, 97, 43074, 3]
listOfInts.size == 0? No. continue
listOfInts.size == 1? NO continue
first = 55
size = 4
second = 43074
return 43074

listOfInts = [97, 43074, 3]
listOfInts.size == 0? NO.
listOfInts.size == 1? NO.
first = 97
size = 3
second = 43074
return 43074

listOfInts = [43074, 3]
listOfInts.size == 0? No. continue
listOfInts.size == 1? NO. continue
first = 43074
size = 2
second = 3
return 43074

listOfInts = [3]
listOfInts.size == 0? NO. continue
listOfInts.size == 1? YES
return 3

```
//Try this on your own

// bonus (not for credit)

5.  public static String findAnagrams(String prefix, String word) {
      if(word.length() <= 1)
            return (prefix + word + );
      else {
            String localStr = "";
            for(int i = 0; i < word.length(); i++) {
                  String cur = word.substring(i, i + 1);
                  String before = word.substring(0, i); // letters before cur
                  String after = word.substring(i + 1); // letters after cur
                  localStr += findAnagrams(prefix + cur, before + after);
            }
            return localStr;
      }
}
```

//call from main:
`findAnagrams("", "arts");`

prefix = ""
word = "arts"
word.length <= 1? NO
localStr = ""
i = 0
cur = "a"
before = ""
after = "rts"
localStr = ⓪ "arts, arst, atrs, atsr, asrt, astr"

i = 1
cur = "r"
before = "a"
after = "ts"
localStr = ① "rats, rast, rtas, rtsa, rsat, rsta"

i = 2
cur = "t"
before = "ar"
after = "s"
localStr = ②

i = 3
cur = "s"
before = "art"
after = ""
localStr = ③

return

prefix = "a"
word = "rts"
word.length <= 1? NO
localStr = ""
i = 0
cur = "r"
before = ""
after = "ts"
localStr = ⓪⓪ "arts, arst"

i = 1
cur = "t"
before = "r"
after = "s"
localStr = ⓪① "arts, arst, atrs, atsr"

i = 2
cur = "s"
before = "rt"
after = ""
localStr = ⓪② "arts, arst, atrs, atsr, asrt, astr"

return ⓪

THIS QUESTION IS MORE COMPLEX THAN YOU'D SEE ON AN EXAM

and so on and so forth...
(This solution is not complete!)