Name:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. What this program does? Write down each recursive call, its parameter and return for number = 3.

import java.util.Scanner;  
public class Recursion {  
 public static void recursiveCall(int number, int lowVal, int highVal) {  
 int midVal;  
 midVal = (highVal + lowVal) / 2;  
 System.*out*.print(number);  
 System.*out*.print(" ");  
 System.*out*.print(midVal);  
 if (number == midVal) {  
 System.*out*.println(" f");  
 }  
 else {  
 if (number < midVal) {  
 System.*out*.println(" l");  
 *recursiveCall*(number, lowVal, midVal);  
 }  
 else {  
 System.*out*.println(" h");  
 *recursiveCall*(number, midVal + 1, highVal);  
 }  
 }  
 }

public static void main(String[] args) {  
 Scanner scnr = new Scanner(System.*in*);  
 int number;  
 number = scnr.nextInt();  
 *recursiveCall*(number, 0, 10);  
 }  
}

3 5 l

3 2 h

3 4 l

3 3 f

1. Write a method that calculates the Fibonacci of n. Knowing that Fibonnacci of 0 or negative number returns 0, Fibonacci of 2 or 1 returns 1, and Fibonacci of a number greater than 2 is the sum of the two previous Fibonacci values.

public static int fibonacci(int n){  
 if(n <= 0) return 0;  
 if(n == 1 || n == 2) return 1;  
 return *fibonacci*(n-1) + *fibonacci*(n-2);  
}

1. How can we change the stringManipulator method to reverse the string backwards, meaning that we start from the end instead of the begin?

public static String stringManipulator(String str) {  
 if (str.isEmpty()) return str;  
 return stringManipulator(str.substring(1)) + str.charAt(0);  
}

public static String stringManipulatorBackwards(String str) {  
 if (str.isEmpty()) return str;  
 return str.charAt(str.length()-1) + stringManipulatorBackwards(str.substring(0, str.length()-1));  
}

1. Write a recursive method that verifies if a String is a palindrome or not. A word is a [**palindrome**](https://en.wikipedia.org/wiki/Palindrome) if the letters in the word are symmetric.

public static boolean palindrome(String str){  
 if(str.isEmpty() || str.length() == 1) return true;  
 if(str.charAt(0) != str.charAt(str.length()-1)) return false;  
 return palindrome(str.substring(1, str.length()-1));  
}

1. Write exactly what is going to be printed in the program below. Draw the recursive calls to make sure that your answer is correct.

public class Recursion {  
 public static void function1(int n)  
 {  
 if (n > 0) {  
 *function1*(n - 1);  
 System.*out*.print(" "+ n);  
 }  
 }  
  
 public static void function2(int n)  
 {  
 if (n > 0)  
 {  
 System.*out*.print(n + " ");  
 *function2*(n - 1);  
 }  
 }  
 public static void main(String[] args)  
 {  
 int x = 3;  
 *function1*(x);  
 System.*out*.println();  
 *function2*(x);  
 }  
}

1 2 3

3 2 1