

Parsons Problems — Week 7

Recursion · Problem 1 of 2

Problem 1 — Reverse a string

Rearrange the slips to form a complete Java program that reads a word from the command line and prints it reversed, using a recursive reverse function.

Example: \$ java ReverseString hello prints olleh.

2 of the slips are distractors — they look plausible but do not belong in the solution.

```
String word = args[0];

System.out.println(reverse(word));

if (s.isEmpty()) return s;

public class ReverseString {

    public static void main(String[] args) {

        return reverse(s) + s.charAt(0);

        return reverse(s.substring(1)) + s.charAt(0);

        return s.charAt(0) + reverse(s.substring(1));

        static String reverse(String s) {

        }

    }

}
```

Parsons Problems — Week 7

Recursion · Problem 2 of 2

Problem 2 — Check for a palindrome

Rearrange the slips to form a complete Java program that reads a word from the command line and prints whether it is a palindrome (reads the same forwards and backwards), using a recursive `isPalindrome` function.

Example: `$ java IsPalindrome racecar` prints `true`.

2 of the slips are distractors — they look plausible but do not belong in the solution.

```
String word = args[0];

System.out.println(isPalindrome(word));

if (s.charAt(0) != s.charAt(s.length() - 1)) return false;

if (s.length() <= 1) return true;

if (s.length() == 0) return true;

public class IsPalindrome {

    public static void main(String[] args) {

        return isPalindrome(s.substring(0, s.length() - 1));

        return isPalindrome(s.substring(1, s.length() - 1));

        static boolean isPalindrome(String s) {

    }

    }

}
```