Recursion Practice

1. Recursive Multiplication

Write a recursive function that accepts two arguments into the parameters `x` and `y` . The function should return the value of `x` times `y` . Remember, multiplication can be performed as repeated addition as follows:

> **7 * 4=4+4+4+4+4+4+47 * 4=4+4+4+4+4+4+4**

2. `isMember` Method

Write a recursive `boolean` method named `isMember` . The method should search an array for a specified value, and return `true` if the value is found in the array, or `false` if the value is not found in the array. Demonstrate the method in a program.

3. String Reverser

Write a recursive method that accepts a `string` as its argument and prints the `string` in reverse order. Demonstrate the method in a program.

4. `maxElement` Method

Write a method named `maxElement` , which returns the largest value in an array that is passed as an argument. The method should use recursion to find the largest element. Demonstrate the method in a program.

5. Palindrome Detector

> **A palindrome is any word, phrase, or sentence that reads the same forward and backward. Here are some well-known palindromes:**

- Able was I, ere I saw Elba
- A man, a plan, a canal, Panama
- Desserts, I stressed
- Kayak

Write a `boolean` method that uses recursion to determine whether a String argument is a palindrome. The method should return `true` if the argument reads the same forward and backward. Demonstrate the method in a program.

6. Character Counter

Write a method that uses recursion to count the number of times a specific character occurs in an array of characters. Demonstrate the method in a program.

7. Recursive Power Method

Write a method that uses recursion to raise a number to a power. The method should accept two arguments: the number to be raised and the exponent. Assume that the exponent is a nonnegative integer. Demonstrate the method in a program.

Shape1 *(missing image)*

The Recursive Power Problem

8. Sum of Numbers

Write a method that accepts an integer argument and returns the sum of all the integers from `1` up to the number passed as an argument. For example, if `50` is passed as an argument, the method will return the sum of `1, 2, 3, 4, . . . 50.`

> **Use recursion to calculate the sum. Demonstrate the method in a program.**