Experiment – 3.1

Student Name: Rohan Jaiswal
UID: 21BCS2856
Branch: BE-CSE
Section: KRG_CC_1-B
Semester: 06
Date of Performance: 19-03-2024
Subject Name: Project-Based Learning in Java with Lab
Subject Code: 21CSH-319

1. **Aim:** Create a palindrome creator application for making a longest possible palindrome out of given input string.

2. **Objective:** To learn about the concept of Strring in Java.

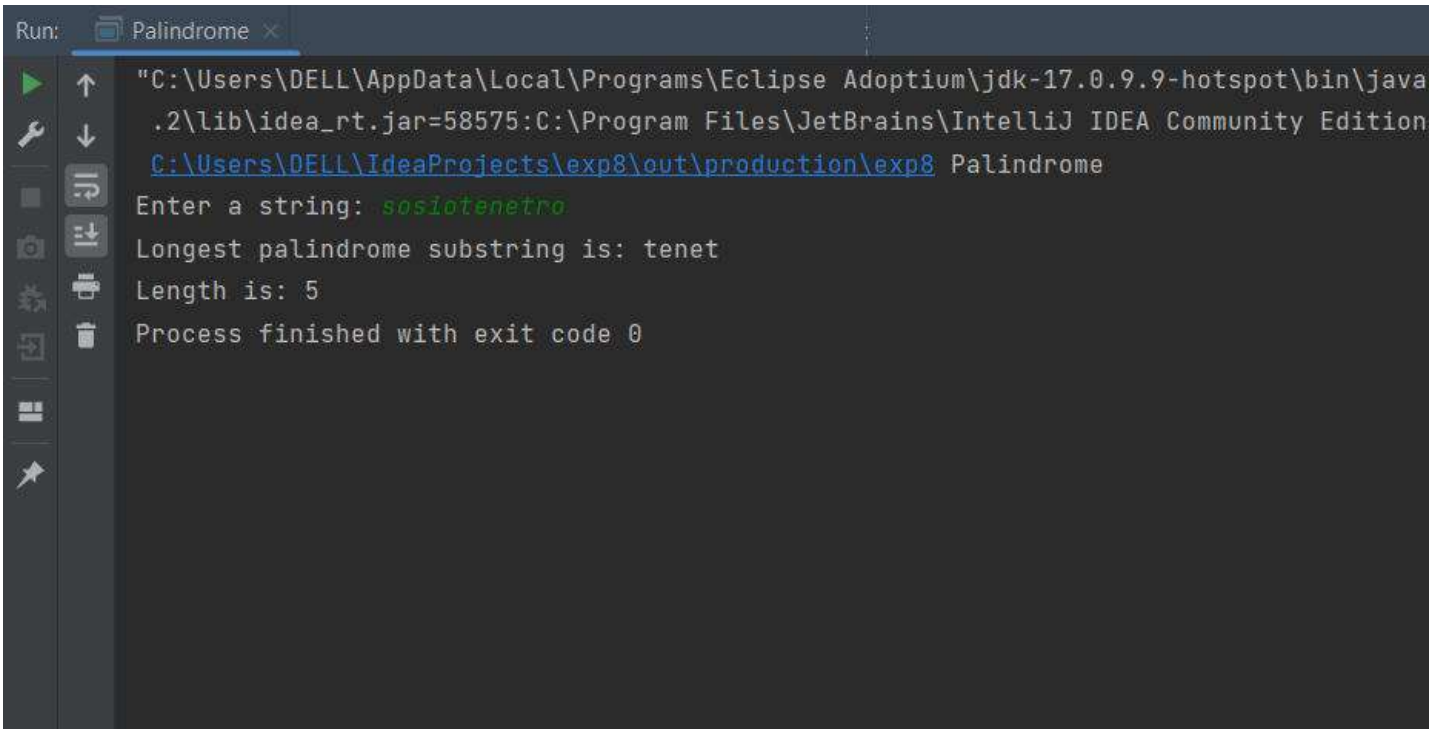
3. **Input/Apparatus Used:** Hardware Requirements: - Minimum 384MB RAM, 100 GB hard Disk, processor with 2.1 MHz Software Requirements: - Eclipse, NetBeans, IntelliJ, etc.

4. **Procedure/Algorithm/Pseudocode:**

1. Start
2. Define a function printSubStr(str, low, high) to print a substring of a string str from index low to high (inclusive).
3. Define a function longestPalSubstr(str) to find the longest palindrome substring in the input string str and return its length.
4. Initialize variables maxLength = 1 and start = 0 to keep track of the length and starting index of the longest palindrome substring respectively.
5. Iterate over each character in the string str using a nested loop:
a. Outer loop: i from 0 to the length of str.
b. Inner loop: j from i to the length of str.
c. For each combination of i and j, check if the substring from i to j is
- If flag is still 1 after the inner loop and the length of the substring (j- i + 1) is greater than maxLength:
i. Update start = i and maxLength = j - i + 1.
6. Print "Longest palindrome substring is: " followed by the substring from start to s

5. Script and Output:

```
import java.util.Scanner;

public class Palindrome {
   static void printSubStr(String str, int low, int high) {
      for (int i = low; i <= high; ++i)
         System.out.print(str.charAt(i));
   }
   static int longestPalSubstr(String str) {
      int n = str.length();

      int maxLength = 1, start = 0;

      for (int i = 0; i < str.length(); i++) {
         for (int j = i; j < str.length(); j++) {
            int flag = 1;

            for (int k = 0; k < (j - i + 1) / 2; k++)
               if (str.charAt(i + k) != str.charAt(j - k))
                  flag = 0;

            if (flag != 0 && (j - i + 1) > maxLength) {
               start = i;
               maxLength = j - i + 1;
            }
         }
      }
      System.out.print("Longest palindrome substring is: ");
      printSubStr(str, start, start + maxLength - 1);
      return maxLength;
   }

   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter a string: ");
      String str = scanner.nextLine();
      System.out.print("\nLength is: " + longestPalSubstr(str));
      scanner.close();
   }
}
```



6. **Learning Outcome:** In this experiment, we learned about
   a. concept of Strings in Java
   b. Learned about Palindrome Checking.