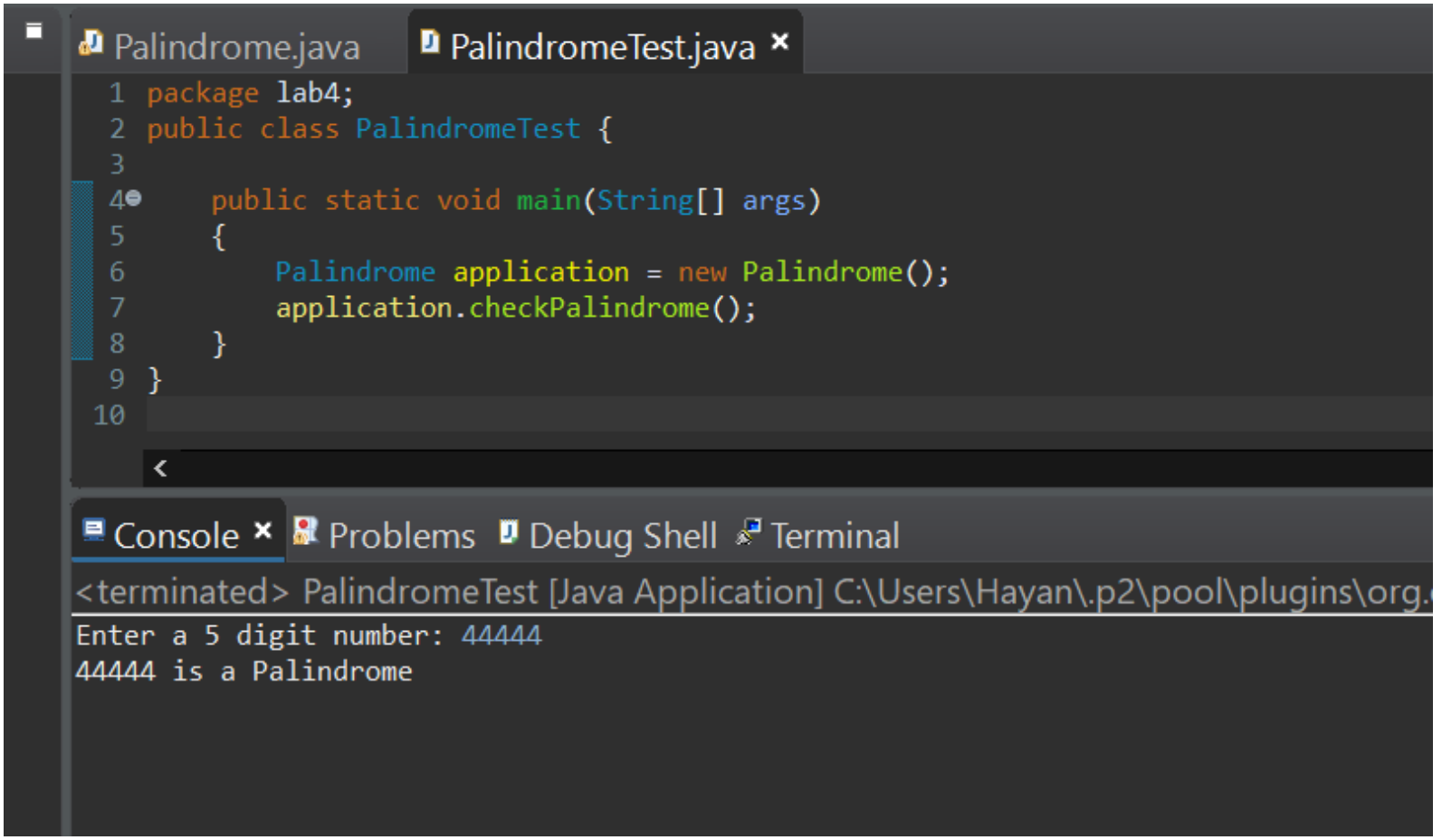
Part 1:

A palindrome is a sequence of characters that reads the same backward as forward. For example, each of the following five-digit integers is a palindrome: 12321, 55555, 45554 and 11611. Write an application that reads in a five-digit integer and determines whether it is a palindrome. If the number is not five digits long, display an error message and allow the user to enter a new value.

Sample Output

```
Enter a 5-digit number: 1234
Number must be 5 digits
Enter a 5-digit number: 123456
Number must be 5 digits
Enter a 5-digit number: 54345
54345 is a palindrome!!!
```



```java

public class Palindrome
{
    public void checkPalindrome()
    {
        Scanner input = new Scanner( System.in );
        int number;
        int digit1 = 0;
        int digit2 = 0;
        int digit4 = 0;
        int digit5 = 0;

        System.out.print("Enter a 5 digit number: ");
        number = input.nextInt();

        int digits = Integer.toString(number).length();

        while ( digits != 5)
        {
            System.out.println("Number must be 5 digits long, try again");
            System.out.print("Enter a 5 digit number: ");
            number = input.nextInt();
            digits = Integer.toString(number).length();
        }

        digit1 = number/10000;
        digit2 = (number%10000)/1000;
        digit4 = (((number%10000)%1000)%100)/10;
        digit5 = ((((number%10000)%1000)%100)%10);

        if (digit1 == digit5 && digit2 == digit4)
            System.out.printf("%d is a Palindrome", number);
        else
            System.out.printf("%d is not a palindrome", number);
    }
}
```

Part 2:

Write a Java application that inputs a series of 10 integers and determines and prints the largest integer. Your program should use at least the following three variables:

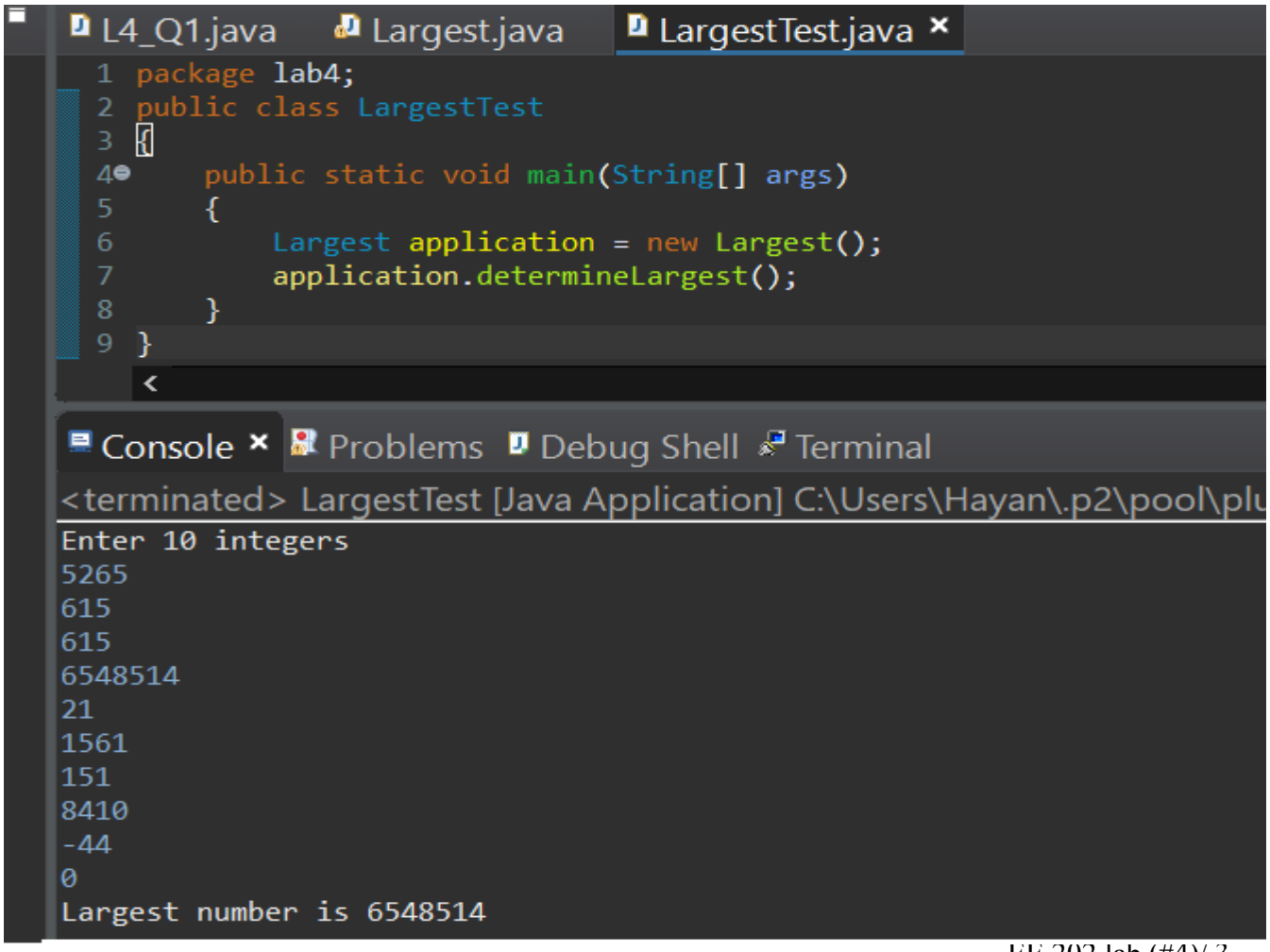
a) counter: A counter to count to 10 (i.e., to keep track of how many numbers have been input and to
determine when all 10 numbers have been processed).

b) number: The integer most recently input by the user.

c) largest: The largest number found so far.

Sample Output

```
Enter number: 56
Enter number: -10
Enter number: 200
Enter number: 25
Enter number: 8
Enter number: 500
Enter number: -20
Enter number: 678
Enter number: 345
Enter number: 45
Largest number is 678
```



```java
package lab4;
import java.util.Scanner;
public class Largest
{
    public void determineLargest()
    {
        Scanner input = new Scanner(System.in);
        int largest = 0;
        int number = 0;
        int counter = 0;

        System.out.println("Enter 10 integers");
        while (counter<10)
        {
            number = input.nextInt();
            if (number > largest)
                largest = number;
            counter++;
        }
        System.out.printf( "Largest number is %d\n", largest );
    }
}
```