***Question 1 (chapter 21)***

**Write a program to store at least 20 pairs of a state and its capital in a map. Your program should prompt the user to**

**Choose 1 if they want to checkByState**

**Choose 2 if they want to checkByCapital**

**if the user chooses 1, the program will prompt the user to enter a state and should display the capital for the state.**

**if the user chooses 2, the program will prompt the user to enter a capital and should display the state for the capital.**

***Code***

import java.util.HashMap;

import java.util.Scanner;

public class StateNCapital {

public static void main(String [] args) {

Scanner input = new Scanner (System.in);

System.out.println("From A to M States, please select following option");

System.out.println("Option 1 if they want to checkByState - display the capital for the state you desired");

System.out.println("Option 2 if they want to checkByCapital - display the state for the capital you desired");

int opt = input.nextInt();

switch (opt) {

case 1: System.out.println("Enter the State:");

CheckByState(input.next());

break;

case 2: System.out.println("Enter the Capital");

CheckByCapital(input.next());

break;

}

}

private static void CheckByCapital(String nextLine) {

HashMap<String, String> value = ArrayOfStates();

String state = null ;

for (String e :value.keySet()) {

if(value.get(e).equals(nextLine)) {

state = e;

}

}

if (state!= null) {

System.out.println( nextLine + " is the capital of " + state );

}

else {

System.out.println("Invalid input");

}

}

private static void CheckByState(String nextLine) {

HashMap<String, String> value = ArrayOfStates();

if(value.get(nextLine) != null) {

System.out.println("Capital of " + nextLine + " is " + value.get(nextLine) );

}

else {

System.out.println("Invalid input");

}

}

private static HashMap<String, String> ArrayOfStates() {

HashMap <String, String> state = new HashMap<>();

state.put ( "Alabama" , "Montgomery" ) ;

state.put ( "Alaska" , "Juneau" ) ;

state.put ( "Arizona" , "Phoenix" ) ;

state.put ( "Arkansas" , "Little Rock" ) ;

state.put ( "California" ,"Sacramento" ) ;

state.put ( "Colorado" , "Denver" ) ;

state.put ( "Connecticut" , "Hartford" ) ;

state.put ( "Delaware" , "Dover" ) ;

state.put ( "Florida" , "Tallahassee" ) ;

state.put ( "Georgia" , "Atlanta" ) ;

state.put ( "Hawaii" , "Honolulu" ) ;

state.put ( "Idaho" , "Boise" ) ;

state.put ( "Illinois" , "Springfield" ) ;

state.put ( "Indiana" , "Indianapolis" ) ;

state.put ( "Iowa" , "Des Moines" ) ;

state.put ( "Kansas" , "Topeka" ) ;

state.put ( "Kentucky" , "Frankfort" ) ;

state.put ( "Louisiana" , "BatonRouge" ) ;

state.put ( "Maine" , "Augusta" ) ;

state.put ( "Maryland" , "Annapolis" ) ;

state.put ( "Massachusetts" , "Boston" ) ;

state.put ( "Michigan" , "Lansing" ) ;

state.put ( "Minnesota" , "Saint Paul" ) ;

state.put ( "Mississippi" , "Jackson" ) ;

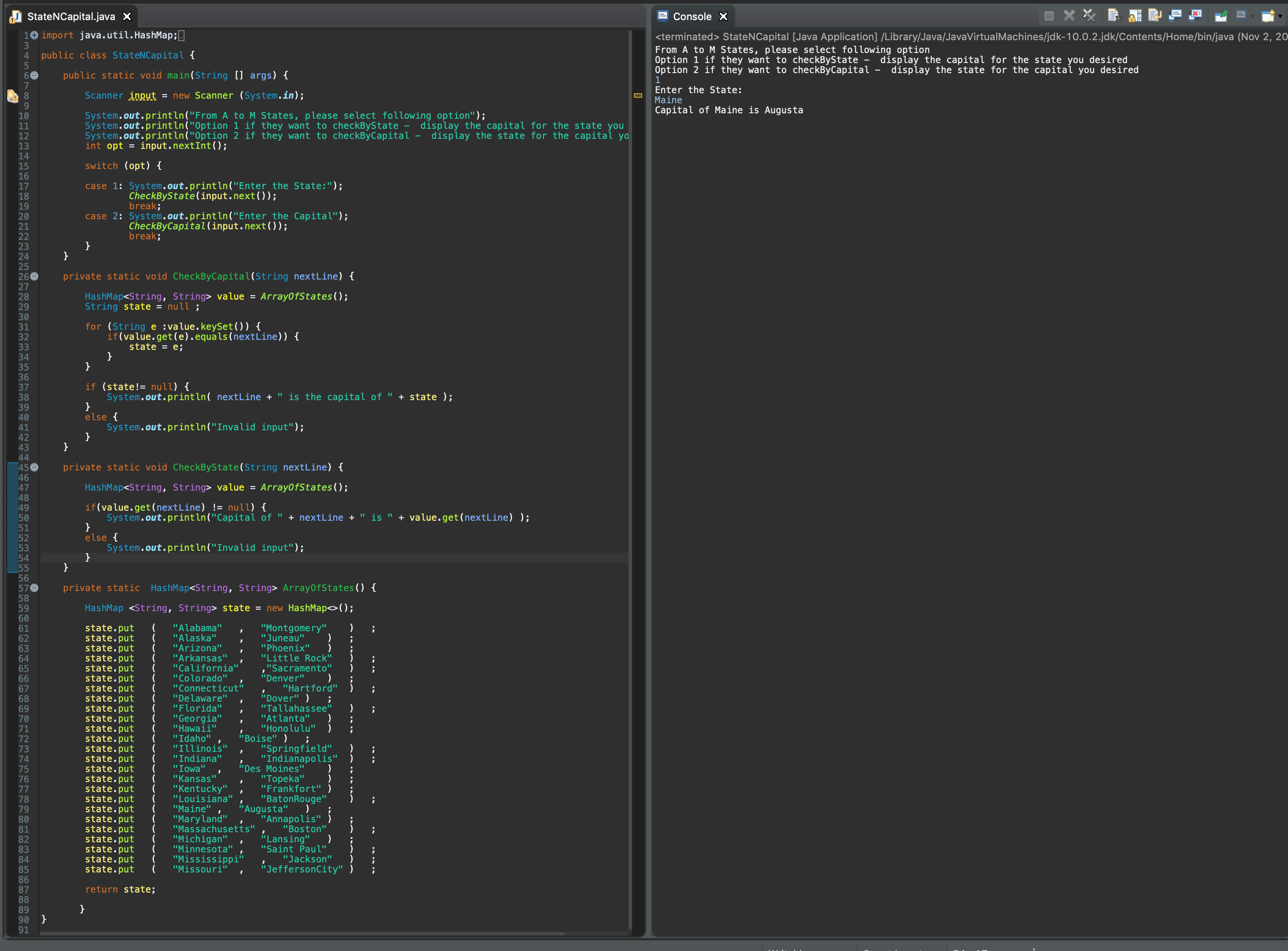
state.put ( "Missouri" , "JeffersonCity" ) ;

return state;

}

}

***Screenshot***

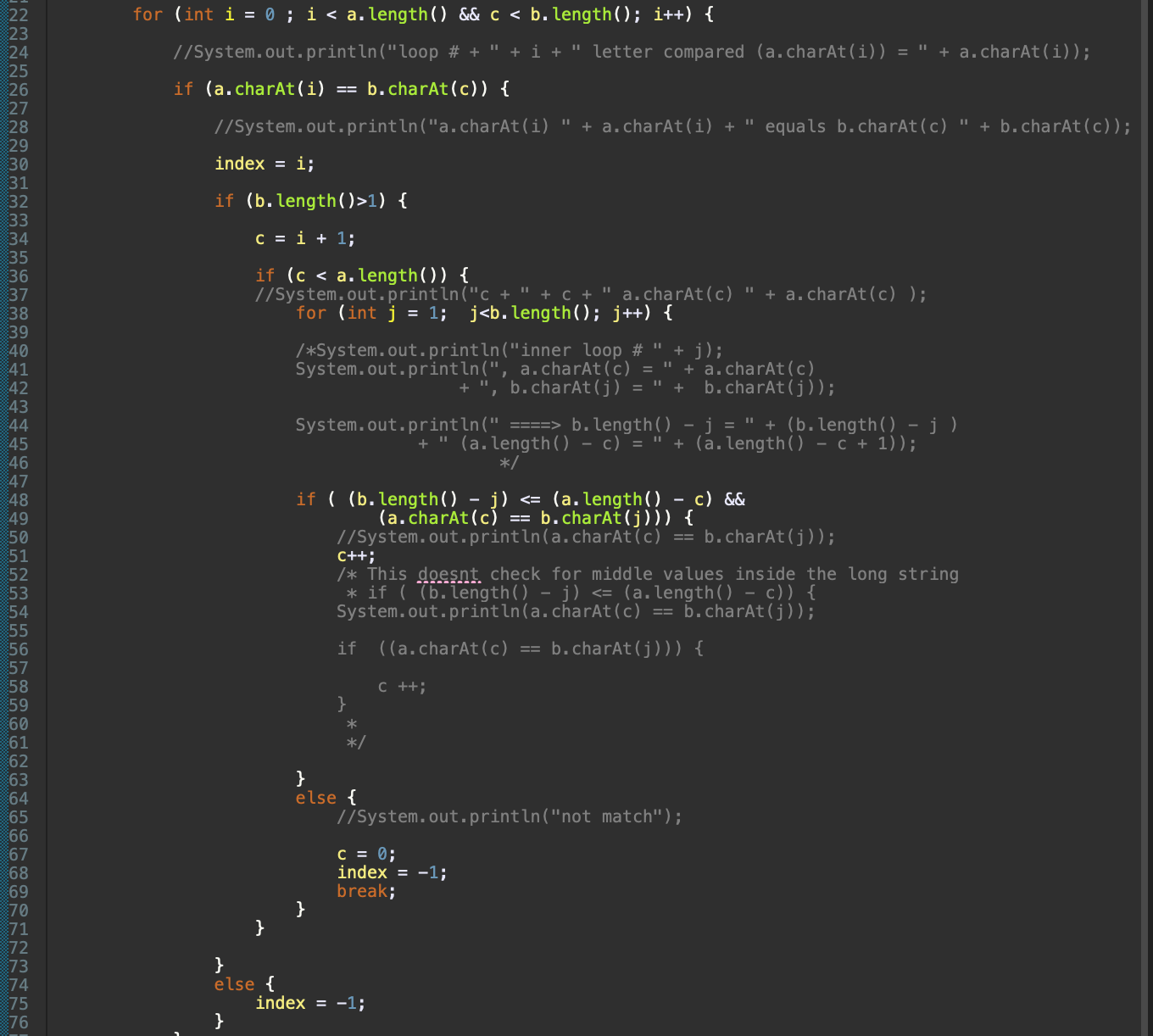
******

***Question 2 (chapter 22)***

**Write a program that prompts the user to enter two strings and test whether the second string is a substring of the first string (don't use the indexOf method in the String class.) Analyze the time complexity of your algorithm. For example, if you enter Mississippi and sip the program will print that they were matched at index 6. Now write an O(n) time program that will perform the same task. Provide the code of both programs and highlight the changes made.**

First Code - O()

Based on first program it runs at O() - (Quadratic Time). Since it has two nested for-loops for execution at same time (used for matching adjourns word in the 1st String). Other factors are carries as constants (simple if statement...etc.) in this program.

****

***Code***

import java.util.Scanner;

public class StringNSubString {

public static void main(String[] args) {

Scanner input = new Scanner (System.in);

System.out.println("Enter the 1st String: ");

String a = input.next();

System.out.println("Enter the 2nd String ");

String b = input.next();

//String a = "mississippi";

//String b = "m";

int index = -1;

int c = 0;

for (int i = 0 ; i < a.length() && c < b.length(); i++) {

//System.out.println("loop # + " + i + " letter compared (a.charAt(i)) = " + a.charAt(i));

if (a.charAt(i) == b.charAt(c)) {

//System.out.println("a.charAt(i) " + a.charAt(i) + " equals b.charAt(c) " + b.charAt(c));

index = i;

if (b.length()>1) {

c = i + 1;

if (c < a.length()) {

//System.out.println("c + " + c + " a.charAt(c) " + a.charAt(c) );

for (int j = 1; j<b.length(); j++) {

/\*System.out.println("inner loop # " + j);

System.out.println(", a.charAt(c) = " + a.charAt(c)

+ ", b.charAt(j) = " + b.charAt(j));

System.out.println(" ====> b.length() - j = " + (b.length() - j )

+ " (a.length() - c) = " + (a.length() - c + 1));

\*/

if ( (b.length() - j) <= (a.length() - c) &&

(a.charAt(c) == b.charAt(j))) {

//System.out.println(a.charAt(c) == b.charAt(j));

c++;

/\* This doesnt check for middle values inside the long string

\* if ( (b.length() - j) <= (a.length() - c)) {

System.out.println(a.charAt(c) == b.charAt(j));

if ((a.charAt(c) == b.charAt(j))) {

c ++;

}

\*

\*/

}

else {

//System.out.println("not match");

c = 0;

index = -1;

break;

}

}

}

else {

index = -1;

}

}

}

}

if (index == -1 ) {

System.out.println("No match");

}

else {

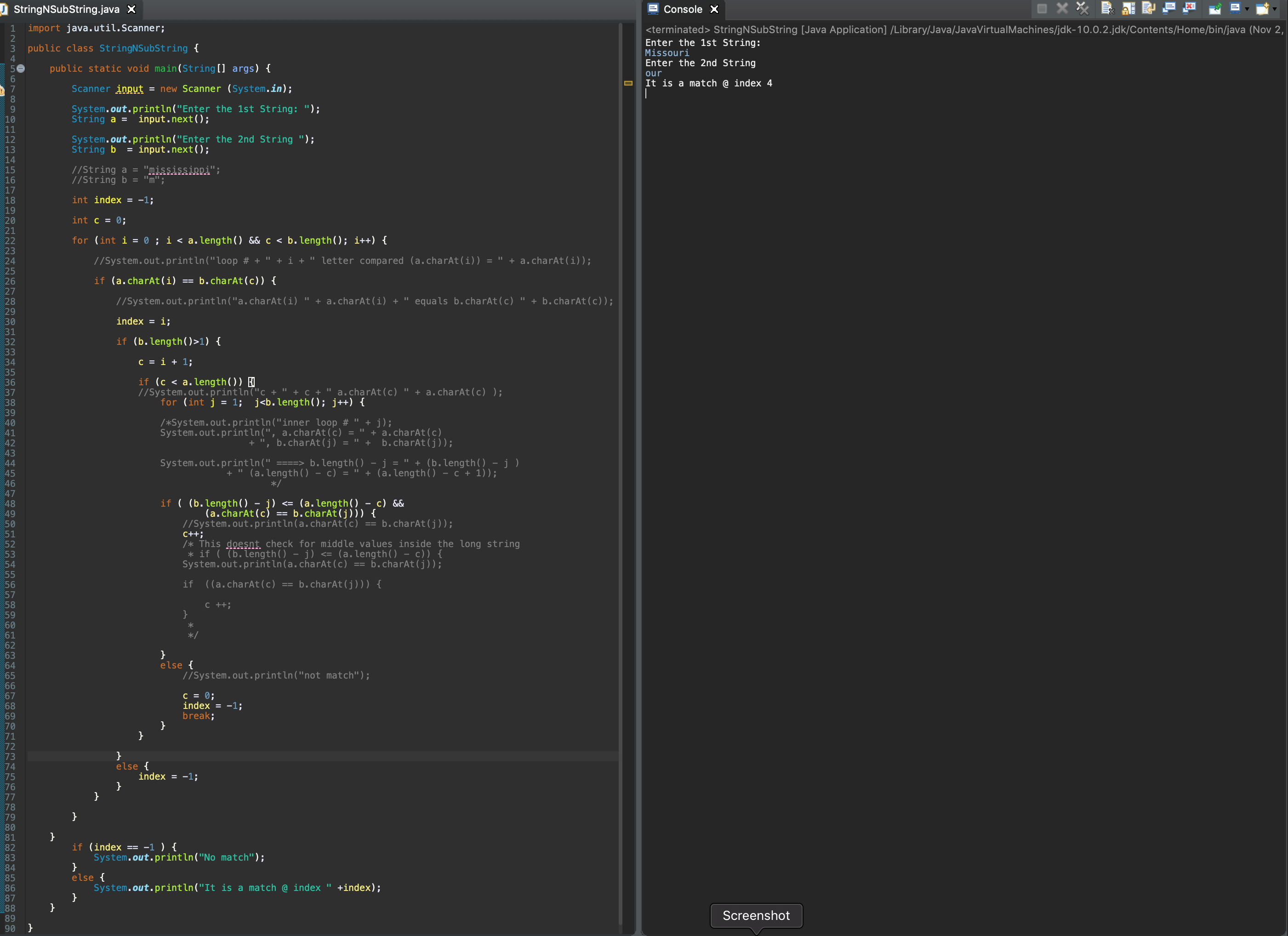
System.out.println("It is a match @ index " +index);

}

}

}

***Screenshot***

**

Second Code - O(n)

Based on second program,it runs at O(n) - (Linear Time). Since it has single for-loops for execution. In order to accommodate for inner for-loop, I have used a counter variable. (used for matching adjourns word in the 1st String) Other factors are consider as Constants. (simple if statement... etc.) Also used an array of integers for storing the indexes. I believe it is constant time, since it is adding particular value to particular index (not searching).

**

***Code***

import java.util.Scanner;

public class StringNSub2 {

public static void main(String[] args) {

Scanner input = new Scanner (System.in);

System.out.println("Enter the 1st String: ");

String a = input.next();

System.out.println("Enter the 2nd String ");

String b = input.next();

//String a = "abcde";

//String b = "cd";

int count =0;

int index [] = new int [b.length()];

for (int i = 0; i< a.length()+1; i++){

//helps to find extra unwanted char from 2nd string

if (i == a.length()) {

index [0] = -1;

break;

}

// 1

if (a.charAt(i) != b.charAt(count)) {

index [0] = -1;

count = 0;

// System.out.println(i+ " 1" + "a.charAt(i) " + a.charAt(i) + " b.charAt(count)" + b.charAt(count) );

}

//2

if ( a.charAt(i) == b.charAt(count) ) {

index [count] = i;

//System.out.println(i+ " " +a.charAt(i) + " " + b.charAt(count) );

count ++;

//System.out.println(i+" 2" + "a.charAt(i) " + a.charAt(i) + " b.charAt(count)" + b.charAt(count) );

}

if (count == b.length() ) {

break;

}

}

if (index[0] == -1 ) {

System.out.println("No match, try again");

}

else {

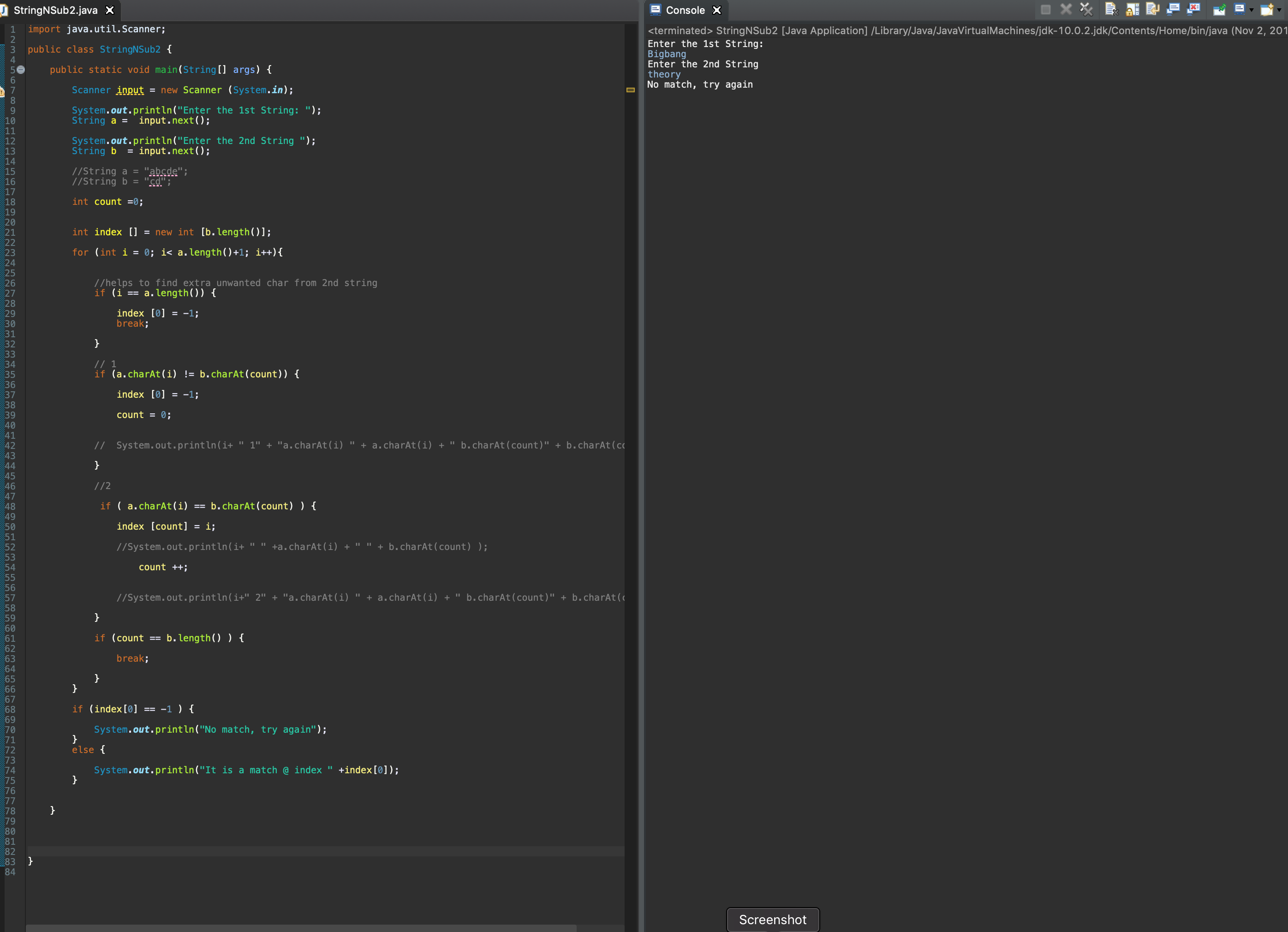
System.out.println("It is a match @ index " +index[0]);

}

}

}

***Screenshot***

******