CIS 36A :: LAB 05 - Strings

Student Name:

Task 2: Understanding Programming

Instructions: Answer each question below. Try to understand and explain the code. **Do not put an IDE code screenshot.**

Task 3: Programming Exercises

Instructions: Use any text editor to write and execute below exercises from the book chapter 5. Attach Snipping photos of your source code and execution of the code in the console. Make sure to create separate files for each exercise.

Chapter Exercises: Do the following chapter exercises.

- Exercise 07: SimpleCipher

```
class SimpleCipher {
 Run | Debug
 public static void main(String[] args) {
 String msg = "This is a test";
 String encMsg = "";
 String decMsg = "";
 String key = "12345678";
 System.out.print(s: "Original message: ");
 System.out.println(msg);
 // encode the message
 for(int i=0; i < msg.length(); i++)
 encMsg = encMsg + (char) (msg.charAt(i) ^ Integer.parseInt(key));
 System.out.print(s: "Encoded message: ");
 System.out.println(encMsg);
 // decode the message
 for(int i=0; i < msg.length(); i++)
 decMsg = decMsg + (char) (encMsg.charAt(i) ^ Integer.parseInt(key));
 System.out.print(s: "Decoded message: ");
 System.out.println(decMsg);
 }
}
```

- Exercise 20:

```java
/*
 Try This 5-1
 Demonstrate the Bubble sort.
*/
class Bubble {
    Run | Debug
    public static void main(String[] args) {
        String[] nums = { "99", "-10", "100123", "18", "-978", "5623", "463", "-9", "287", "49" };
        int a, b;
        String t;
        int size;
        size = 10; // number of elements to sort
        // display original array
        System.out.print(s: "Original array is:");
        for(int i=0; i < size; i++)
        System.out.print(" " + nums[i]);
        System.out.println();
        // This is the Bubble sort.
        for(a=1; a < size; a++)
        for(b=size-1; b >= a; b--) {
            if(nums[b-1].length() > nums[b].length()) { // if out of order
                // exchange elements
                t = nums[b-1];
                nums[b-1] = nums[b];
                nums[b] = t;
            }
        }
        // display sorted array
        System.out.print(s: "Sorted array is:");
        for(int i=0; i < size; i++)
        System.out.print(" " + nums[i]);
        System.out.println();
    }
}
```

```
PS G:\Java 1\CIS36A>  & 'C:\Program Files\Java\jdk-18.0.2.1\bin\java.exe'
Obukhov\AppData\Roaming\Code\User\workspaceStorage\17253e159982c637975172
Original array is: 99 -10 100123 18 -978 5623 463 -9 287 49
Sorted array is: 99 18 -9 49 -10 463 287 -978 5623 100123
PS G:\Java 1\CIS36A>
```

- Exercise 27: Sorter or Not Sorted

```java
public class SortCheck {
    Run | Debug
    public static void main(String[] args) {
        String[] nums = { "99", "-10", "100123", "18", "-978", "5623", "463", "-9", "287", "49" };
        int a, b;
        int size;
        int c = 0;
        size = 10; // number of elements to sort
        System.out.println();
        // This is the Bubble sort.
        for (a = 1; a < size; a++)
            for (b = size - 1; b >= a; b--) {
                if (nums[b - 1].length() > nums[b].length()) { // if out of order
                    c = 1;
                    break;
                }
            }
        // display sorted array
        if (c == 0) {System.out.println(x: "Sorted");}
        else{System.out.println(x: "Not sorted");}
    }
}

```

```
s' '-cp' 'C:\Users\Theodor Obukhov\AppData\Roaming\Co

Not sorted
PS G:\Java 1\CIS36A>
```

- Exercise 28: Palindrome String

```java
public class Palindrome {
    Run | Debug
    public static void main(String[] args){
        String str = "racecar";
        System.out.println("String: " + str);
        Boolean a = true;
        int left = 0, right = str.length() - 1;
        while (left < right) {
            if (str.charAt(left) != str.charAt(right)) {
                a = false;
                System.out.println(x: "Is not a palindrome");
            }
            left++;
            right--;
        }
        if (a==true){System.out.println(x: "Is a palindrome");}
    }
}
// "Inspiration" has been taken from this link
// https://www.educative.io/answers/how-to-check-if-a-string-is-a-palindrome-in-java
```

```
s' '-cp' 'C:\Users\Theodor Obukhov\App
String: raceacar
Is not a palindrome
PS G:\Java 1\CIS36A>  g:; cd 'g:\Java
s' '-cp' 'C:\Users\Theodor Obukhov\App
String: racecar
Is a palindrome
PS G:\Java 1\CIS36A>
```

- Exercise 29:

```java
import java.util.*;;
public class StringsAndSub {

    Run | Debug
    public static void main(String[] args){
        // For loop iterates through the string
        // Adds all the characters into another string until it hits a comma
        // After hitting comma, adds that string as an element into array, wipes temp string and continues until end of string

        String baseString = "abc,def";
        String tempString = "";
        List<String> SubArray = new ArrayList<String>();
        char current;

        for (int i=0; i<baseString.length()+1; i++){
            if (i == baseString.length()){
                SubArray.add(tempString);
                break;
            }

            current = baseString.charAt(i);
            if (current == ','){
                SubArray.add(tempString);
                tempString = "";
            }
            else{
                tempString = tempString + current;
            }
        }
        System.out.println(SubArray);
    }
}
```

```
s' '-cp' 'C:\Users\Theodor Obukhov\
[abc, def]
PS G:\Java 1\CIS36A>
```

Task 4: Programming Application

Instructions: Use any IDE to write and execute the program below. Attach Snipping photos of your source code and execution of the code in the console.

For this Task only, submit your .java file as well.

Hangman Game: Using Strings and strings methods, design a hangman game.

1. You should have a list of words to choose from (at least 20 words between lengths of 5 and 7). When each game starts it should choose a word from the list randomly.
2. Create a blank string with the exact size of your chosen word and fill it with underscores.
3. After each correct guess places all of the occurences of the correct letter in the blank string and displays it. (Hint: Use substring method to reconstruct the blank string)
4. If the user guesses a wrong letter, show a console-based hangman illustrating the progress. Show this illustration for correct guesses as well.

   Please see the sample display below.

Start Screen	**Progress Screen**	**Lost Screen**
H A N G M A N +---+ | | | ====== Missed letters:	+---+ 0 | | | | === Missed letters: o r _ a t	+---+ 0 | /|\ | / \ | ========

_ _ _ _ _ Guess a letter:	Guess a letter:	

Note: You may create a method to display this illustration.

5. Keep all of the user guesses in an array of strings (or chars) and warn the user if they entered the same letter again.
6. The game should repeat until the user guesses the word or loses the game. Allow users to miss up to four or five letters.
7. Add one more feature of your own.
8. Good Luck and Have Fun!!

```java
import java.util.ArrayList;
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
public class Hangman {
    //list of words to choose from (20)
    //Random word chosen every time
    //Display the word as underscores and change to correct character when the letter is guessed
    //Have hanging man be made every mistake

    //Function to choose random word from array and returns word
    //Function that takes input of guessed characters and the correct word and returns a string of the hidden word with characters replaced accordingly
    //Function that displays the hanging man with input of wrong characters and /r prints the hanging dude

    static String WordGuess(String[] wordList){
        int length = wordList.length;
        int chosenIndex = (int) ((Math.random() * (length - 0)) + 0);
        return wordList[chosenIndex];
    }

    static String[] GuessedLogic(ArrayList<Character> charList, String correctWord){
        int wrongCount=0;
        StringBuilder outputString = new StringBuilder(str: "");
        outputString.setLength(correctWord.length());
        for (int i = 0; i < correctWord.length(); i++){
            outputString.setCharAt(i, ch: '_');
        }

        for (int i = 0; i < correctWord.length(); i++) {
            char ch = correctWord.charAt(i);
            for (int j = 0; j < charList.size(); j++) {
                if (charList.get(j) == ch) {
                    outputString.setCharAt(i, ch);
                }
            }
        }
        for (int i=0; i < charList.size(); i++){
            if (correctWord.indexOf(charList.get(i))==-1){
                wrongCount = wrongCount+ 1;
            }
        }
        String[] returnStringArr = {String.valueOf(wrongCount), outputString.toString()};
        return returnStringArr;
    }
    static void displayManAndWord(int wrongCount, String correctWord, String guessedLogicOutput, ArrayList<Character> charList){
        String head = " ";
        String body = " ";
        String leftArm = " "; //When activated it needs to be " /"
        String rightArm = " ";
        String leftLeg = "  "; //When activated it needs to be " /"
        String rightLeg = " ";
        if (wrongCount > 0){head = "0";}
        if (wrongCount > 1){ body = "|";}
        if (wrongCount > 2){leftArm = " /";}
        if (wrongCount > 3){rightArm = "\\";}
        if (wrongCount > 4){leftLeg = " /";}
        if (wrongCount > 5){rightLeg = "\\";}
        String lineOne = "  +---+"; //7,7,7
        String LineTwo = "  "+ head + "    |"; //8,8,8
        String LineThree = leftArm + body + rightArm + "   |"; //8,8,9
        String LineFour = leftLeg + " " + rightLeg + "   |"; //8,9,9
        String LineFive = "========";


        //Print out
        System.out.println("\r" + lineOne);
        System.out.println("\r" + LineTwo);
        System.out.println("\r" + LineThree);
        System.out.println("\r" + LineFour);
        System.out.println("\r" + LineFive);
```

```java
        System.out.println("\r" + LineFive);
        System.out.println("\r" + guessedLogicOutput);
        System.out.println("\r" + charList.toString());
    }

    Run | Debug
    public static void main(String[] args) throws IOException{
        String[] wordList = {"test", "abraham", "cactus", "life", "death", "daredevil", "joker", "rifle", "plane", "helicopter", "misspell", "java", "programming"};
        ArrayList<Character> charList = new ArrayList<Character>();
        String correctWord = WordGuess(wordList);
        System.out.println(correctWord);
        System.out.println(x: "Hangman the game:");
        BufferedReader scanner = new BufferedReader(new InputStreamReader(System.in));
        String userInputString;
        char userInputChar = ' ';
        int wrongCount;
        String wrongCountStr;
        String guessedOutput;
        while (true){
            System.out.println(x: "Guess a letter:");
            userInputString = scanner.readLine();
            try{
                userInputChar = userInputString.charAt(index: 0);
            }
            catch (Exception e){
                System.out.println(x: "You have inputted an invalid character. Please try again.");
                continue;
            }
            charList.add(userInputChar);
            String[] guessed = GuessedLogic(charList, correctWord);
            wrongCountStr = guessed[0];
            guessedOutput = guessed[1];
            wrongCount = Integer.parseInt(wrongCountStr);
            System.out.println(wrongCount);
            displayManAndWord(wrongCount, correctWord, guessedOutput, charList);
```

```
102                 System.out.println(wrongCount);
103                 displayManAndWord(wrongCount, correctWord, guessedOutput, charList);
104                 for (int i = 0; i < guessedOutput.length(); i++){
105                     if (guessedOutput.indexOf(ch: '_')==-1){
106                         System.out.println(x: "You have guessed correctly. Goodbye.");
107                         System.exit(status: 0);
108                     }
109                 }
110                 if (wrongCount==6){
111                     System.out.println(x: "You have guessed incorrectly. Goodbye.");
112                     System.exit(status: 0);
113                 }
114             }
115         }
116     }
117
```

```
Hangman the game:
Guess a letter:
p
0
  +---+
      |
      |
      |
========
p____
[p]
Guess a letter:
l
0
  +---+
      |
      |
      |
========
pl___
[p, l]
Guess a letter:
a
0
  +---+
      |
      |
      |
========
pla__
[p, l, a]
Guess a letter:
n
0
  +---+
      |
      |
      |
========
plan_
[p, l, a, n]
Guess a letter:
e
0
  +---+
      |
      |
      |
========
plane
[p, l, a, n, e]
You have guessed correctly. Goodbye.
```