Assignment

1. JAVA

A. Create an array with the values (1, 2, 3, 4, 5, 6, 7) and shuffle it.

Code- The below code defines a shuffleArray function that takes an array of integers and shuffles it using the Fisher-Yates algorithm. Then, in the main method, we create an array with the values (1, 2, 3, 4, 5, 6, 7) and call the shuffleArray function to shuffle it. Finally, it prints the shuffled array.

import java.util.Random;

public class ShuffleArray {

public static void main(String[] args) {

int[] array = {1, 2, 3, 4, 5, 6, 7};

shuffleArray(array);

for (int num : array) {

System.out.print(num + " ");

}

}

public static void shuffleArray(int[] array) {

int n = array.length;

Random rand = new Random();

for (int i = n - 1; i > 0; i--) {

int j = rand.nextInt(i + 1);

// Swap array[i] and array[j]

int temp = array[i];

array[i] = array[j];

array[j] = temp;

}

}

}

B. Enter a Roman Number as input and convert it to an integer. (ex IX = 9)

Code- In the below code, we use a HashMap to map Roman numerals to their corresponding integer values. We iterate through the Roman numeral string from right to left, and if the current numeral is smaller than the previous one, we subtract its value from the result; otherwise, we add its value to the result.Just change the romanNumeral variable to the Roman numeral you want to convert, and the program will output its integer equivalent.

import java.util.HashMap;

import java.util.Map;

public class RomanToInteger {

public static void main(String[] args) {

String romanNumeral = "IX"; // Change this to the Roman numeral you want to convert

int result = romanToInteger(romanNumeral);

System.out.println("The integer equivalent of " + romanNumeral + " is " + result);

}

public static int romanToInteger(String s) {

Map<Character, Integer> romanMap = new HashMap<>();

romanMap.put('I', 1);

romanMap.put('V', 5);

romanMap.put('X', 10);

romanMap.put('L', 50);

romanMap.put('C', 100);

romanMap.put('D', 500);

romanMap.put('M', 1000);

int result = 0;

int prevValue = 0;

for (int i = s.length() - 1; i >= 0; i--) {

int currentValue = romanMap.get(s.charAt(i));

if (currentValue < prevValue) {

result -= currentValue;

} else {

result += currentValue;

}

prevValue = currentValue;

}

return result;

}

}

C. Check if the input is pangram or not. (Pangram is a sentence that contains all the alphabet

from a-z)

Code- In this code, we create a boolean array to keep track of whether each letter from 'a' to 'z' has been seen in the input. We iterate through the input, convert all characters to lowercase for case insensitivity, and mark the corresponding letter in the array as seen. Finally, we check if all letters have been seen, and if so, the input is a pangram.

public class Pangram{

public static void main(String[] args) {

String input = "Five or six big jet planes zoomed quickly by the tower.";

boolean isPangram = isPangram(input);

if (isPangram) {

System.out.println("The input is a pangram.");

} else {

System.out.println("The input is not a pangram.");

}

}

public static boolean isPangram(String input) {

// Create a boolean array to represent the 26 letters (a-z)

boolean[] letters = new boolean[26];

// Convert the input to lowercase for case-insensitivity

input = input.toLowerCase();

// Iterate through the characters in the input

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

char ch = input.charAt(i);

// Check if the character is a lowercase letter

if (ch >= 'a' && ch <= 'z') {

// Mark the letter as seen

letters[ch - 'a'] = true;

}

}

// Check if all letters have been seen

for (boolean letter : letters) {

if (!letter) {

return false; // Not a pangram

}

}

return true; // It's a pangram

}

}

2 JavaScript

A. Take a sentence as an input and reverse every word in that sentence.

a. Example - This is a sunny day > shiT si a ynnus yad.

Code- To reverse every word in a sentence in JavaScript without using built-in methods, we can split the sentence into words, reverse each word, and then join them back into a sentence. Here's a function that accomplishes this:

function reverseWords(sentence) {

// Split the sentence into words

const words = sentence.split(' ');

// Create an array to store the reversed words

const reversedWords = [];

// Iterate through each word and reverse it

for (const word of words) {

let reversedWord = '';

for (let i = word.length - 1; i >= 0; i--) {

reversedWord += word[i];

}

reversedWords.push(reversedWord);

}

// Join the reversed words back into a sentence

const reversedSentence = reversedWords.join(' ');

return reversedSentence;

}

// Example usage:

const inputSentence = "This is a test round";

const reversedSentence = reverseWords(inputSentence);

console.log(reversedSentence); // Output: "sihT si a tset dnuor"

B. Perform sorting of an array in descending order.

Code- To sort an array in descending order without using built-in sorting methods, we can implement a custom sorting algorithm such as the bubble sort. Here's an example of sorting an array in descending order using the bubble sort algorithm:

function sortDescending(arr) {

const n = arr.length;

for (let i = 0; i < n - 1; i++) {

for (let j = 0; j < n - i - 1; j++) {

if (arr[j] < arr[j + 1]) {

// Swap the elements if they are in the wrong order

const temp = arr[j];

arr[j] = arr[j + 1];

arr[j + 1] = temp;

}

}

}

}

// Example usage:

const arr = [5, 2, 9, 1, 5, 6];

sortDescending(arr);

console.log(arr); // Output: [9, 6, 5, 5, 2, 1]

3. HTML

A. Create a basic calculator using HTML, CSS, and JavaScript with the functionality of add,

subtract, multiply and divide. Use the following picture for reference.

B. Create a survey form with Fields; First Name, Last Name, Date of Birth, Country (dropdown),

Gender (checkbox), Profession, email, and mobile number. All the input fields are

necessary to submit the form. Create two buttons Submit and Reset. Reset will reset the

form while clicking on submit, first, it will check all the fields and necessary validations and

then a popup will appear displaying all the selected values with the label in front of it. On

closing the popup, the form should reset all the values. Use the following for reference