Assignments:

# 1.Find Factorial using data structure.

import java.util.Scanner;

public class Factorial {

public static void main(String[] args) {

// Create a Scanner object to read input from the user.

Scanner read = new Scanner(System.in);

// Print a prompt to the user to enter the number.

System.out.println("Enter a number: ");

// Read the number from the user.

int number = read.nextInt();

// Initialize the factorial variable to 1.

long factorial = 1;

// Loop from 1 to the number and multiply the factorial by each number.

for (int i = 1; i <= number; i++) {

factorial \*= i;

}

// Print the factorial of the number.

System.out.println("The factorial of " + number + " is " + factorial);

}

}2.Write a java program to make Fibonacci series and sum it.

import java.util.ArrayList;

import java.util.List;

public class Fibonacci {

public static List<Integer> getFibonacciSeries(int n) {

List<Integer> fibonacciSeries = new ArrayList<>();

fibonacciSeries.add(0);

fibonacciSeries.add(1);

for (int i = 2; i <= n; i++) {

fibonacciSeries.add(fibonacciSeries.get(i - 1) + fibonacciSeries.get(i - 2));

}

return fibonacciSeries;

}

public static int getSumOfFibonacciSeries(int n) {

List<Integer> fibonacciSeries = getFibonacciSeries(n);

int sum = 0;

for (int i = 0; i < fibonacciSeries.size(); i++) {

sum += fibonacciSeries.get(i);

}

return sum;

}

public static void main(String[] args) {

int n = 10;

System.out.println("The Fibonacci series up to " + n + " is:");

for (int i = 0; i < n; i++) {

System.out.print(getFibonacciSeries(i) + " ");

}

System.out.println();

System.out.println("The sum of the Fibonacci series up to " + n + " is " + getSumOfFibonacciSeries(n));

}

}

3. WAP to test if an Array contains a specific value using:

A.Linear/Sequencial Search

public class LinearSearch {

public static boolean contains(int[] arr, int item) {

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

if (arr[i] == item) {

return true;

}

}

return false;

}

public static void main(String[] args) {

int[] arr = {1, 2, 3, 4, 5};

int item = 3;

if (contains(arr, item)) {

System.out.println("The array contains the value " + item);

} else {

System.out.println("The array does not contain the value " + item);

}

}

}

B. Binary Search

import java.util.Arrays;

public class BinarySearch {

public static boolean contains(int[] array, int value) {

int low = 0;

int high = array.length - 1;

while (low <= high) {

int mid = (low + high) / 2;

if (array[mid] == value) {

return true;

} else if (array[mid] < value) {

low = mid + 1;

} else {

high = mid - 1;

}

}

return false;

}

public static void main(String[] args) {

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

int value = 3;

System.out.println("Does the array contain the value " + value + "? " + contains(array, value));

}

}

4.WAP to find the maximum and minimum value in an Array.

import java.util.Arrays;

public class MaxMin {

public static void main(String[] args) {

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

int max = Integer.MIN\_VALUE;

int min = Integer.MAX\_VALUE;

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

if (array[i] > max) {

max = array[i];

}

if (array[i] < min) {

min = array[i];

}

}

System.out.println("The maximum value in the array is " + max);

System.out.println("The minimum value in the array is " + min);

}

}

5.WAP to find duplicate values of an array

import java.util.Arrays;

import java.util.HashSet;

public class FindDuplicates {

public static void main(String[] args) {

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

HashSet<Integer> set = new HashSet<>();

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

if (!set.contains(array[i])) {

set.add(array[i]);

} else {

System.out.println("The duplicate value is " + array[i]);

}

}

}

}

6. What is an Algorithm?

algorithm is a step-by-step procedure for solving a problem. It is a set of instructions that a computer can follow to complete a task. Algorithms are used in all aspects of computer programming, from simple tasks like adding two numbers to complex tasks like searching for information on the internet.

7. How to work with HashMap?

8. What is Recursion?

Recursion is a programming technique where a function calls itself

9. Write a program in Java using the functions append(), prepend().

import java.util.ArrayList;

public class DataStructure {

public static void main(String[] args) {

ArrayList<String> list = new ArrayList<>();

list.add("1");

list.add("2");

list.add("3");

System.out.println("The original list is: " + list);

append(list, "4");

prepend(list, "0");

System.out.println("The new list is: " + list);

}

public static void append(ArrayList<String> list, String value) {

list.add(value);

}

public static void prepend(ArrayList<String> list, String value) {

list.add(0, value);

}

}