**089\_Sagar Bhanuse\_KH C-DAC Mumbai Date 26/09/2024**

**Subject: Algorithm and Data Structure**

**Assignment 1**

**Solve the assignment with following thing to be added in each question.**

-Program

-Flow chart

-Explanation

-Output

-Time and Space complexity

1. Printing Patterns

Problem: Write a Java program to print patterns such as a right triangle of stars.

Test Cases:

Input: n = 3

Output:

\*

\*\*

\*\*\*

Input: n = 5

Output:

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

Ans: class Star{

public static void main(String[] args){

int n=3;

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

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

System.out.print("\*");

}

System.out.println();

}

}

}

2. Remove Array Duplicates

Problem: Write a Java program to remove duplicates from a sorted array and return the new length of the array.

Test Cases:

Input: arr = [1, 1, 2]

Output: 2

Input: arr = [0, 0, 1, 1, 2, 2, 3, 3]

Output: 4

Ans: class Duplicate{

public static void main(String[] args){

int arr[]={0,0,1,1,2,2,3,3};

int count=arr.length;

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

if(arr[i]==arr[i+1]){

count-=1;

}

}

System.out.print(count);

}

}

3. Remove White Spaces from String

Problem: Write a Java program to remove all white spaces from a given string.

Test Cases:

Input: "Hello World"

Output: "HelloWorld"

Input: " Java Programming "

Output: "JavaProgramming"

Ans: import java.io.\*;

import java.util.\*;

class Space{

public static void main(String[] args){

String str="Hello Java kaise hai";

String str1="";

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

char ch=str.charAt(i);

if (!Character.isWhitespace(ch)) {

str1 += ch;

}

}

System.out.println(str1);

}

}

4. Reverse a String

Problem: Write a Java program to reverse a given string.

Test Cases:

Input: "hello"

Output: "olleh"

Input: "Java"

Output: "avaJ"

Ans: class Reverse{

public static void main(String[] args){

String str="Hello";

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

char ch=str.charAt(i);

System.out.print(ch);

}

}

}

5. Reverse Array in Place

Problem: Write a Java program to reverse an array in place.

Test Cases:

Input: arr = [1, 2, 3, 4]

Output: [4, 3, 2, 1]

Input: arr = [7, 8, 9]

Output: [9, 8, 7]

Ans: class RevArr{

public static void main(String[] args){

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

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

System.out.print(arr[i]+" ");

}

}

}

6. Reverse Words in a String

Problem: Write a Java program to reverse the words in a given sentence.

Test Cases:

Input: "Hello World"

Output: "World Hello"

Input: "Java Programming"

Output: "Programming Java"

public class WordRev{

public static void main(String[] args){

String str = "Hello World kasie ho";

String[] strArray = str.split(" ");

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

System.out.print(strArray[i]+" ");

}

}

}

7. Reverse a Number

Problem: Write a Java program to reverse a given number.

Test Cases:

Input: 12345

Output: 54321

Input: -9876

Output: -6789

class Reverse {

public static void main(String[] args) {

int num = 12345;

String strNum = Integer.toString(num);

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

System.out.print(strNum.charAt(i));

}

}

}

8. Array Manipulation

Problem: Perform a series of operations to manipulate an array based on range update queries. Each query adds a value to a range of indices.

Test Cases:

Input: n = 5, queries = [[1, 2, 100], [2, 5, 100], [3, 4, 100]]

Output: 200

Input: n = 4, queries = [[1, 3, 50], [2, 4, 70]]

Output: 120

9. String Palindrome

Problem: Write a Java program to check if a given string is a palindrome.

Test Cases:

Input: "madam"

Output: true

Input: "hello"

Output: false

Here’s a continuation of the list of assignment questions starting from question 21, with two test cases for each:

class Palindrome{

static boolean pal(String str){

for(int i=0;i<str.length()/2;i++){

int n=str.length();

if(str.charAt(i)!=str.charAt(n-i-1)){

return false;

}

}

return true;

}

public static void main(String[] args){

String str = "Madam";

System.out.println(pal(str));

}

}

10. Array Left Rotation

Problem: Write a Java program to rotate an array to the left by d positions.

Test Cases:

Input: arr = [1, 2, 3, 4, 5], d = 2

Output: [3, 4, 5, 1, 2]

Input: arr = [10, 20, 30, 40], d = 1

Output: [20, 30, 40, 10]

Ans: class Rotate{

public static void main(String[] args){

int[] arr={10,20,30,40};

int d=2;

int size=arr.length;

int copyArr[]=new int[size];

for(int i=0;i<size;i++)

{

copyArr[i]=arr[(i+d)%size];

System.out.print(copyArr[i]+" ");

}

}

}