**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:

\*

\*\*

\*\*\*

\*\*\*\*

import java.util.Scanner;

class Printpattern{

public static void main(String[] args){

Scanner sc = new Scanner(System.in);

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

int n = sc.nextInt();

for(int i = 1;i <= n;i++)

{

for (int j = 1;j <= i;j++)

{

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

}

System.out.println( );

}

}

}

A computer screen shot of a black screen

Description automatically generated

2. Remove ArrayDuplicates

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

public class RemoveDuplicates {

public static int removeDuplicates(int[] nums) {

if (nums == null || nums.length == 0) {

return 0;

}

int uniqueCount = 1; // Start counting from the first element

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

if (nums[i] != nums[uniqueCount - 1]) {

nums[uniqueCount] = nums[i];

uniqueCount++;

}

}

return uniqueCount; // Return the new length

}

public static void main(String[] args) {

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

int newLength = removeDuplicates(nums);

System.out.println("New length: " + newLength);

System.out.print("Unique elements: ");

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

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

}

}

}

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"

import java.util.Scanner;

class RemovewhitespacesS{

public static void main(String[] args){

Scanner sc = new Scanner(System.in);

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

String str= sc.nextLine();

str = str.replaceAll("\\s", "");

System.out.println(str);

}

}

**OUTPUT**

A screen shot of a computer program

Description automatically generated

4. Reverse a String

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

Test Cases:

Input: "hello"

Output: "olleh"

Input: "Java"

Output: "avaJ"

import java.util.Scanner;

class Reversestring{

public static void main(String[] args){

Scanner sc = new Scanner(System.in);

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

String str= sc.nextLine();

char temp;

String str2=" ";

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

temp=str.charAt(i);

str2=temp+str2;

}

System.out.print("Reverse string is " + str2);

}

}

**OUTPUT**

A screenshot of a computer program

Description automatically generated

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]

import java.util.Scanner;

public class Reversearray{

public static void main(String args[]) {

Scanner sc = new Scanner(System.in)

System.out.print("Enter the number of elements in array: ");

int n = sc.nextInt(); // Define n

int[] arr = new int[n];

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

System.out.print("Enter array element " );

arr[i] = sc.nextInt();

}

System.out.print("Array in reverse order: ");

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

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

}

}

}

A computer screen with white text

Description automatically generated

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"

import java.util.Scanner;

class Reversewords

{

public static void main(String[] args)

{

Scanner sc= new Scanner(System.in); //System.in is a standard input stream

System.out.print("Enter a string: ");

String str= sc.nextLine(); //reads string

char temp;

String str2=" ";

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

temp=str.charAt(i);

str2=temp+str2;

}

System.out.print("Reverse string is " + str2);

}

}

OUTPUT

A screenshot of a computer program

Description automatically generated

7. Reverse a Number

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

Test Cases:

Input: 12345

Output: 54321

Input: -9876

Output: -6789

import java.util.Scanner;

class Reversenumber{

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

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

int num= sc.nextInt();

int reversed = 0;

System.out.println("Original Number: " + num);

// run loop until num becomes 0

while(num != 0) {

// get last digit from num

int digit = num % 10;

reversed = reversed \* 10 + digit;

// remove the last digit from num

num = num/10;

}

System.out.println("Reversed Number: " + reversed);

}

}

**OUTPUT**

A screenshot of a computer program

Description automatically generated

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:

import java.io.\*;

import java.util.Scanner;

public class Palindrome

{

public static void checkPalindrome(String s)

{

// reverse the given String

String reverse = new StringBuffer(s).reverse().toString();

// check whether the string is palindrome or not

if (s.equals(reverse))

System.out.println("True");

else

System.out.println("False");

}

public static void main (String[] args)

throws java.lang.Exception

{

Scanner sc= new Scanner(System.in); //System.in is a standard input stream

System.out.print("Enter a string: ");

String str= sc.nextLine();

checkPalindrome(str);

}

}

OUTPUT

A screen shot of a computer program

Description automatically generated