CSE18R272-LAB MANUAL

KALASALINGAM ACADEMY OF RESEARCH AND EDUCATION COMPUTER SCIENCE AND EDUCATION

Date: 19-09-2020

Name: BITRA PURNA KOTESWARA RAO Regno: 9919004035

Course name: java programming Course Code: CSE18R272 Section:A5

1. Find mean and standard deviation.

Source Code:

Import java.lang.\*;

Import java.util.Scanner;

Class Main StandardDeviation{

Public static void main (String [] args)

{

System.out.println(“Enter 5 numbers to find standard deviation”);

Scanner in = new Scanner (System.in);

Double arr[]= new double [5];

Double sum=0, mean=0,dist;

For (int i=0; i<5; i++)

{

System.out.print(“Enter a number : “);

Arr[i]=in.nextDouble();

Sum+=arr[i];

}

Mean=sum/5;

Sum=0;

System.out.println(“Mean : “+mean);

For (int i=0; i<5; i++)

{

Dist=Math.pow((arr[i]-mean),2);

Sum+=dist;

}

Mean=sum/5;

Double deviation=Math.sqrt(mean);

System.out.println(“Deviation : “+ deviation);

}

}

1. Find the nCr and nPr. Source Code:

Import java.util.Scanner;

Class Main NcrNpr

{

Public static void main(String args[])

{

Int n, r;

Scanner s = new Scanner(System.in);

System.out.print(“Enter Value of n : “); N = s.nextInt(); System.out.print(“Enter Value of r : “); R = s.nextInt();

System.out.print(“NCR = “ +(fact(n)/(fact(n-r)\*fact(r))));

System.out.print(“\nNPR = “ +(fact(n)/(fact(n-r))));

}

Public static int fact(int num)

{

Int fact=1, i;

For(i=1; i<=num; i++) Fact = fact\*i;

Return fact;

}

}

1. Print all prime numbers in the given range. Source Code:

Import java.util.Scanner;

Class Main CheckPrime

{

Public static void main(String args[])

{

Int start, end;

Scanner s = new Scanner(System.in); Start =s.nextInt();

End =s.nextInt();

For (int i=start;i<=end;i++) If(prime(i)) System.out.println(i);

}

Public static boolean prime(int n)

{

For (int i=2;i<=n/2;i++) If(n%i==0)

Return false;

Return true;

}

}

1. Find sum of the digits.

Source Code:

Import java.util.Scanner; Class Main SoD

{

Public static void main(String args[])

{

Int num,sum=0,r;

Scanner s = new Scanner(System.in); System.out.println(“Enter a number:”); Num =s.nextInt();

While (num >0)

{

R=num%10; Sum+=r; Num=num/10;

}

System.out.println(“sum of the digits :”+sum);

}

}

1. Check whether a given number is palindrome or not. Source Code:

Import java.util.Scanner; Class Main CheckPalindrome

{

Public static void main(String args[])

{

Int num,n;

Scanner s = new Scanner(System.in); System.out.print(“Input a number: “);

Num =s.nextInt(); Int sum =0,r; N=num;

While (num >0)

{

R=num%10;

Sum=(sum\*10)+r; Num=num/10;

}

If (n== sum)

System.out.println( n + “ is a palindrome”); Else

System.out.println( n + “ is not palindrome”);

}

}

1. Check whether a given number is prime factor or not. Source Code:

import java.util.Scanner;

public class CheckPrimeFactors { public static void main(String args[]){

int number;

Scanner sc = new Scanner(System.in);

System.out.println("Enter a number :”); number = sc.nextInt();

for(int i = 2; i< number; i++) { while(number%i == 0) {

System.out.println(i+" "); number = number/i;

}

}

if(number >2) { System.out.println(number);

}

}

}

}

}

1. Check whether a given number is perfect number or not. Source Code:

import java.util.Scanner; public class CheckPerfect {

public static void main(String[] args){ int n,sum =0;

Scanner s = new Scanner(System.in); System.out.print("Enter any integer:");

n = s.nextInt();

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

{

if(n % i == 0)

{

sum = sum + i;

}

}

if(sum == n)

{

System.out.println("Given number is Perfect");

}

else

{

System.out.println("Given number is not Perfect");

}

}

}

1. Check whether a given number is deficient number or not. Source Code:

import java.util.Scanner; public class CheckDeficient {

public static void main(String[] args){ int n,sum =0;

Scanner s = new Scanner(System.in); System.out.print("Enter any integer you want to check:"); n = s.nextInt();

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

{

if(n % i == 0)

{

sum = sum + i;

}

}

if(sum < n)

{

System.out.println("Given number is Deficient");

}

else

{

System.out.println("Given number is not Deficient");

}

}

}

1. Apply any one of the sorting algorithm. Source Code:

Import java.util.Scanner;

Class Main Sorting{

Public static void main(String []args) { Int n, i, j, temp;

Scanner s = new Scanner(System.in);

System.out.println(“Enter the number of integers to sort:”); N = s.nextInt();

Int arr[] = new int[n];

System.out.println(“Enter “ + n + “ integers: “);

For (i = 0; i < n; i++) Arr[i] = s.nextInt();

For (i = 0; i < ( n – 1 ); i++) {

For (j = 0; j < n – i – 1; j++) {

If (arr[j] > arr[j+1])

{

Temp = arr[j]; Arr[j] = arr[j+1]; Arr[j+1] = temp;

}

}

}

System.out.println(“Sorted list of integers:”);

For (i = 0; i < n; i++) System.out.println(arr[i]);

}

}

1. Number conversion from decimal to binary. Program:

import java.util.Scanner;

public class BinaryToDecimal {

public static void main(String[] args){ int Array[];

Array= new int[100];

Scanner sc=new Scanner(System.in); System.out.println("Enter the decimal value: "); int n = sc.nextInt();

int i=0; while(n>0){

Array[i]=n%2; n=n/2;

i++;

}

for(int j=i-1;j>=0;j--){ System.out.print(Array[j]);

}

}

}

1. Write a program to complete and exp(e) exp(e)= 1 + x/1! + x^2/2! + x^3/3! + ......

Program:

import java.util.Scanner; public class Exponential{

static float exponential(int n, float x)

{

float sum = 1;

for (int i = n - 1; i > 0; --i ) sum = 1 + x \* sum / i;

return sum;

}

public static void main (String[] args)

{

System.out.print("Enter Starting Number : "); Scanner sc=new Scanner(System.in);

int n = sc.nextInt(); float x = 1;

System.out.println("e^x = "+exponential(n,x));

}

}

1. Program to compute row sum, column sum and trace of a matrix Program:

**import** jjavautil.Scanner;

**class** Matrix {

**static int** m = 4;

**static int** n = 4;

**static void** row\_sum(**int** arr[][])

{

**int** i,j,sum = 0;

System.out.print( "\nFinding Sum of each row:\n\n");

**for** (i = 0; i < 4; ++i) {

**for** (j = 0; j < 4; ++j) {

sum = sum + arr[i][j];

}

System.out.println( "Sum of the row "

+ i + " = " + sum);

sum = 0;

}

}

**static void** column\_sum(**int** arr[][])

{

**int** i,j,sum = 0;

System.out.print( "Finding Sum of each column:\n\n");

**for** (i = 0; i < 4; ++i) {

**for** (j = 0; j < 4; ++j) {

sum = sum + arr[j][i];

}

System.out.println( "Sum of the column "

+ i + " = " + sum);

sum = 0;

}

}

**public static void** main (String[] args) {

**int** i,j;

**int** [][]arr = **new int**[m][n];

**int** x = 1;

**for** (i = 0; i < m; i++)

**for** (j = 0; j < n; j++) arr[i][j] = x++;

row\_sum(arr); column\_sum(arr);

}

}

1. Write a program to encrypt a code in Caesar's code Program:

class caesarCipher{ String plain;

int key;

public caesar(String text,int k){ plain=text;

key=k;

}

String encrypt(){

String out="";char ch;

for (int i=0; i<plain.length(); i++){ if(Character.isUpperCase(plain.charAt(i)))

ch=(char)(((int)plain.charAt(i)-65+key)%26+65); else

ch=(char)(((int)plain.charAt(i)-97+key)%26+97); out+=ch;

}

return out;

}

public static void main(String[] args) { caesar c = new caesar("Koteswarao", 18);

String out = c.encrypt(); System.out.println(out);

}

}

1. Write a program Java Program to implement the Mono alphabetic Cipher. program:

class MACipher

{

static String encoder(char[] key)

{

String encoded = "";

boolean[] arr = new boolean[26];

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

{

if (key[i] >= 'A' && key[i] <= 'Z')

{

if (arr[key[i] - 65] == false)

{

encoded += (char) key[i]; arr[key[i] - 65] = true;

}

}

else if (key[i] >= 'a' && key[i] <= 'z')

{

if (arr[key[i] - 97] == false)

{

encoded += (char) (key[i] - 32);

arr[key[i] - 97] = true;

}

}

}

for (int i = 0; i < 26; i++)

{

if (arr[i] == false)

{

arr[i] = true;

encoded += (char) (i + 65);

}

}

return encoded;

}

static String cipheredIt(String msg, String encoded)

{

String cipher = "";

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

{

if (msg.charAt(i) >= 'a' && msg.charAt(i) <= 'z')

{

int pos = msg.charAt(i) - 97; cipher += encoded.charAt(pos);

}

else if (msg.charAt(i) >= 'A' && msg.charAt(i) <= 'Z')

{

int pos = msg.charAt(i) - 65; cipher += encoded.charAt(pos);

}

else

{

cipher += msg.charAt(i);

}

}

return cipher;

}

public static void main(String[] args)

{

String key; key = "koti";

System.out.println("Keyword : " + key); String encoded = encoder(key.toCharArray()); String message = "Koteswarao";

System.out.println("Message before Ciphering : " + message); System.out.println("Ciphered Text : " + cipheredIt(message, encoded));

}

}

1. Write a program Java Program to implement simple Encryption Decryption with Modulo 26

Program:

Class Modulo {

String plain;

public Substitution(String text){

plain = text;

}

String encrypt() { String out="";

String alpha="abcdefghijklmnopqrstuvwxyz"; String sub="kdgfnslvbwahexjmqcpzrtyiuo";

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

{

int key = 0;

char ch = (char) (( (int) plain.charAt(1) - 97 +key )% 26 +97); out = out + ch;

} return out;

}

public static void main(String[] args)

{

Substitution c = new Substitution("Koteswarao");

String out = c.encrypt(); System.out.println(out);

}

}

1. Write a program Java Program for XOR Cipher. Program:

class xor{ String plain; char key;

public xor(String text,char k){ plain=text;

key=k;

}

String encrypt(){

String out="";char ch;

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

ch=(char)((int)plain.charAt(i)^(int)key); out+=ch;

}

return out;

}

public static void main(String[] args) { xor c = new xor("Koti", 'H');

String out = c.encrypt(); System.out.println(out);

}

}

1. Write a program Java Program for Latin alphabet cipher. Program:

class Latincipher{ String plain; Latincipher (String text)

{

plain =text;

}

String encrypt()

{

String alpha="abcdefghijklmnopqrstuvwxyz";

String out="";

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

{

char ch=plain.charAt(i);

int p=alpha.indexOf(ch)+1; out =out+p+" ";

}

return out;

}

public static void main(String[] args) { Latincipher c=new Latincipher("Koteswarao"); String out=c.encrypt();

System.out.println(" encryption : "+out);

}

}

1. Write a program called Harmonic Sum to compute the sum of a harmonic series, as

shown below, where n=50000. The program shall compute the sum from left-to- right as

well as from the right-to-left. Are the two sums the same? Obtain the absolute difference

between these two sums and explain the difference. Which sum is more accurate?

public class MyClass {

public static void main(String args[]){ int harmonic = 50000;

double L2R=0, R2L=0;

for(int i=1; i<=harmonic; i++){ L2R += (double)(1)/i;

R2L += (double)(1)/(harmonic-i+1);

}

double difference = R2L-L2R;

System.out.println("left to right = " + L2R); System.out.println("right to left = " + R2L); System.out.println("difference is " + difference);

}

}

1. Write a program which prompts user for the number of students in a class (a non-

negative integer), and saves it in an int variable called num Students. It then prompts user

for the grade of each of the students (integer between 0 to 100) and saves them in an int

array called grades. The program shall then compute and print the average (in double

rounded to 2 decimal places) and minimum/maximum (in int). Program:

import java.util.Scanner; public class GradesAverage {

private final int LOWEST\_GRADE = 0;

private final int HIGHEST\_GRADE = 100; private int[] grades;

private Scanner in;

public static void main(String[] args)

{

GradesAverage aGradesAverage = new GradesAverage(); aGradesAverage.in = new Scanner(System.in);

System.out.print("Enter the number of students: "); int numStudents = aGradesAverage.in.nextInt();

aGradesAverage.run(numStudents);

}

private void run(int numStudents)

{

if (numStudents <= 0) {

System.out.println("Invalid number of students."); return;

}

grades = new int[numStudents];

double sum = 0; int i = 0;

while (i < numStudents)

{

System.out.printf("Enter the grade for student %1$d: ", (i+1)); int grade = in.nextInt();

// chek if grade is between 0 and 100

if ((grade >= LOWEST\_GRADE) && (grade <= HIGHEST\_GRADE))

{

grades[i] = grade; sum += grade; i++;

continue;

}

System.out.println("Invalid grade, try again...");

}

System.out.printf("The average is %1$.2f\n", (sum / numStudents));

}

}

1. Write a Java program to separate 0s on left side and 1s on right side of an array of 0s and

1s in random order. Program:

import java.util.Arrays; import java.util.Scanner;

public class MyClass {

public static void main(String[] args)

{

int arr[] = new int[8]; int result[];

System.out.println("Original Array "); System.out.println(Arrays.toString(arr)); Scanner sc = new Scanner(System.in);

System.out.println("Enter the elements of the array: "); for(int i=0; i<8; i++)

{

arr[i]=sc.nextInt();

}

int n = arr.length;

result = separate\_arr(arr, n); System.out.println("New Array "); System.out.println(Arrays.toString(result));

}

static int [] separate\_arr(int arr[], int n)

{

int count = 0;

for (int i = 0; i < n; i++) { if (arr[i] == 0)

count++;

}

for (int i = 0; i < count; i++) arr[i] = 0;

for (int i = count; i < n; i++) arr[i] = 1;

return arr;

}

}

1. Write a Java program to add and remove a specific element from an array. Program:

import java.util.Scanner; public class Remove

{

public static void main(String[] args)

{

int n, x, flag = 1, loc = 0;

Scanner s = new Scanner(System.in); System.out.print("Enter no. of elements you want in array:"); n = s.nextInt();

int a[] = new int[n]; System.out.println("Enter all the elements:"); for (int i = 0; i < n; i++)

{

a[i] = s.nextInt();

}

System.out.print("Enter the element you want to delete:"); x = s.nextInt();

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

{

if(a[i] == x)

{

flag =1; loc = i; break;

}

else

{

flag = 0;

}

}

if(flag == 1)

{

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

{

a[i-1] = a[i];

}

System.out.print("After Deleting:"); for (int i = 0; i < n-2; i++)

{

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

}

System.out.print(a[n-2]);

}

else

{

System.out.println("Element not found");

}

}

}

1. Write a program called CozaLozaWoza which prints the numbers 1 to 110, 11 numbers

per line. The program shall print "Coza" in place of the numbers which are multiples of 3,

"Loza" for multiples of 5, "Woza" for multiples of 7, "CozaLoza" for multiples of 3 and 5,

and so on. The output shall looklike:

1 2 Coza 4 Loza Coza Woza 8 Coza Loza 11

Coza 13 Woza CozaLoza 16 17 Coza 19 Loza CozaWoza 22

23 Coza Loza 26 Coza Woza 29 CozaLoza 31 32 Coza Program:

public class MyClass {

public static void main(String args[]) { int i = 1;

while (i <= 110) { boolean test = false; if (i % 3 == 0) {

System.out.print("coza"); test = true;

}

if (i % 5 == 0) {

System.out.print("loza"); test = true;

}

if (i % 7 == 0) {

System.out.print("woza"); test = true;

}

if (!test) {

System.out.print(i);

}

System.out.print(" "); if (i % 11 == 0) {

System.out.println();

} i++;

}

}

}