Lab-01

Q1) Array Problems

1.Linear Search (return an index of element if found otherwise return -1)

2. Reverse the array

3. Find maximum absolute difference.

Code:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Name : Naman Lambat ; Roll No: 202151096;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

import java.util.Scanner;

public class Main

{

public static int linearsearch(int [] a, int k)

{

int index=-1;

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

{

if(a[i]== k)

index=i;

}

return index;

}

public static void reversearray(int [] a)

{

int [] b = new int[a.length];

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

{

b[i]= a[(a.length)-(i+1)];

}

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

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

}

public static int maxabsdiff(int [] a)

{

int max=a[0]; int min = a[0];

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

{

if(a[i]>max)

max=a[i];

if(a[i]<min)

min=a[i];

}

int sum = max-min;

return sum;

}

public static void main(String[] args) {

Scanner sc= new Scanner(System.in);

int [] x= new int[]{1,2,3,4,5,6,7,8,9,99};

int y = sc.nextInt();

int i =linearsearch(x,y);

System.out.println(i);

int l= maxabsdiff(x);

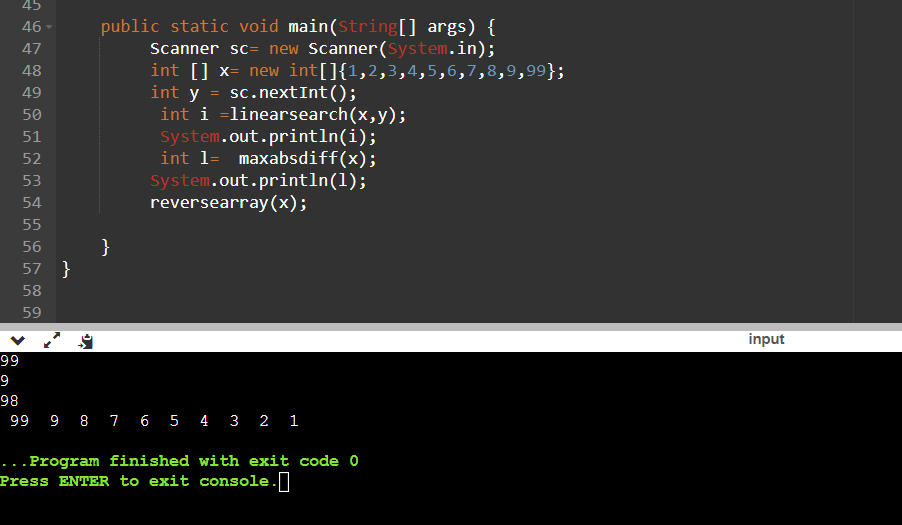
System.out.println(l);

reversearray(x);

}

}

Result:



Q2) Pattern

Code:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Name : Naman Lambat ; Roll No : 202151096

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

import java.util.Scanner;

public class Main

{

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter the number of Rows");

int n = sc.nextInt();

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

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

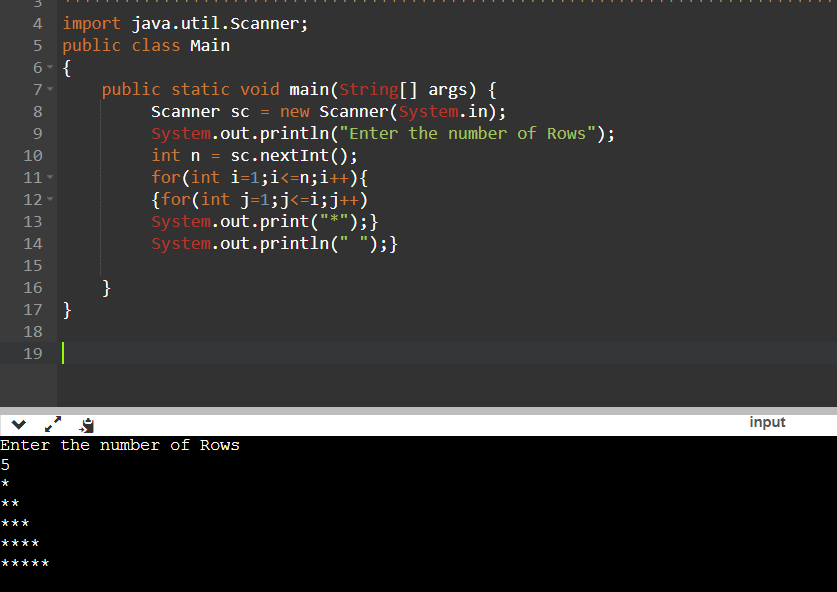
System.out.print("\*");}

System.out.println(" ");}

}

}

Result:



Q3) Mathematical Concepts

Code:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Name: Naman Lambat ; Roll no : 202151096

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

import java.util.Scanner;

public class Main

{

public static int division (int a,int b)

{

int c = a/b;

return c;

}

public static int lcm(int a,int b)

{

int k=gcd(a,b); int ans= (a\*b)/k;

return ans;

}

public static int power(int a,int b)

{

int product =1;

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

product\*=a;

return product;

}

public static int max(int [] a)

{

int max = 0;

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

if(a[i]>max) {

max = a[i];

}

}

return max;

}

public static int min(int [] a) {

int min = a[0];

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

if(a[i]<min) {

min = a[i];

}

}

return min;

}

public static int abs(int a)

{

int k=0;

if(a>=0)

k=a;

if(a<0)

k=k+a;

return k;

}

public static int factorial(int a)

{

int prod=1;

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

prod\*=i;

return prod;

}

public static int sum(int [] a)

{

int sum=0;

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

sum+=a[i];

return sum;

}

public static int sumOfDigits(int a)

{

int sum=0;

while (a!=0)

{

int rem=a%10;

sum+= rem;

a/=10;

}

return sum;

}

public static int isPrime(int a )

{

int c=0;

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

if(a%i!=0)

c++;

if(c==2)

return 1;

else

return 0;

}

public static int isLeapYear( int a)

{

int k=0;

if(a%4==0 && a%100!=0)

k=1;

if(a%4==0 && a%400==0)

k=1;

return k;

}

public static int isPalindrome(int a)

{

int b=a; int rev=0;

while (a!=0)

{

int rem =a%10;

rev= rev\*10 + rem;

a=a/10;

}

if(rev==b)

return 1;

else

return 0;

}

public static int isArmstrong(int a)

{

int nodig=0; int b=a; int c=a;

while (a!=0)

{

nodig+=1;

a/=10;

}

int sum=0;

while(b!=0)

{

int rem = b%10;

sum+= power(rem,nodig);

b/=10;

}

if(sum==c)

return 1;

else

return 0;

}

public static int ArithmeticSequenceSum(int a, int d, int n)

{

int sum=0;

sum= a + (n-1)\*d;

return sum;

}

public static double GeometricSequenceSum(int a, int r)

{

double sum= a/(1-r);

return sum;

}

public static int gcd (int a , int b)

{ int flag=1;

for(int i=1;i<=a||i<=b;i++)

if(a%i==0 && b%i==0)

flag =i;

return flag;

}

public static void main(String args[])

{

Scanner sc = new Scanner (System.in);

int x=sc.nextInt();

int y= sc.nextInt();

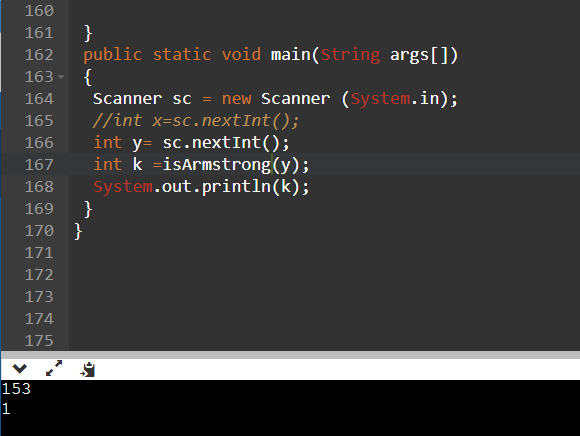
int k =lcm(x,y);

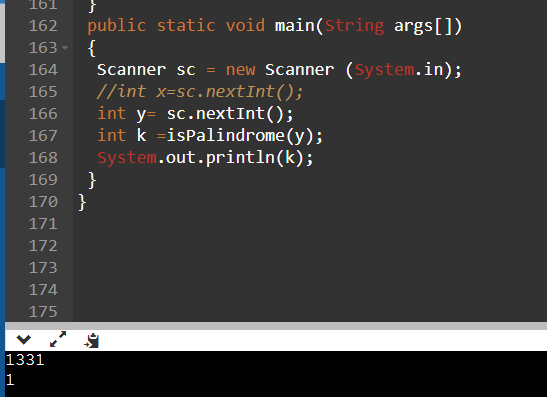
System.out.println(k);

}

}

Result:







Note: Here the output 1 represents true or correct.

Q4) Matrix Problems

Code:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Name : Naman Lambat ; ROll no : 202151096

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

import java.util.Scanner;

public class Main

{

public static void addmatrix()

{

int row, col,i,j;

Scanner in = new Scanner(System.in);

System.out.println("Enter the inputs below for addition");

System.out.println("Enter the number of rows");

row = in.nextInt();

System.out.println("Enter the number columns");

col = in.nextInt();

int mat1[][] = new int[row][col];

int mat2[][] = new int[row][col];

int res[][] = new int[row][col];

System.out.println("Enter the elements of matrix 1");

for ( i= 0 ; i < row ; i++ )

{

for ( j= 0 ; j < col ;j++ )

mat1[i][j] = in.nextInt();

System.out.println();

}

System.out.println("Enter the elements of matrix 2");

for ( i= 0 ; i < row ; i++ )

{

for ( j= 0 ; j < col ;j++ )

mat2[i][j] = in.nextInt();

System.out.println();

}

for ( i= 0 ; i < row ; i++ )

for ( j= 0 ; j < col ;j++ )

res[i][j] = mat1[i][j] + mat2[i][j] ;

System.out.println("Sum of matrices:-");

for ( i= 0 ; i < row ; i++ )

{

for ( j= 0 ; j < col ;j++ )

System.out.print(res[i][j]+"\t");

System.out.println();

}

}

public static void multiplymatrix()

{

int m, n, p, q, sum = 0, c, d, k;

Scanner sc = new Scanner(System.in);

System.out.println("Enter the inputs below for multiplication");

System.out.println("Enter the number of rows and columns of first matrix");

m = sc.nextInt();

n = sc.nextInt();

int first[][] = new int[m][n];

System.out.println("Enter elements of first matrix");

for (c = 0; c < m; c++)

for (d = 0; d < n; d++)

first[c][d] = sc.nextInt();

System.out.println("Enter the number of rows and columns of second matrix");

p = sc.nextInt();

q = sc.nextInt();

if (n != p)

System.out.println("The matrices can't be multiplied with each other.");

else

{

int second[][] = new int[p][q];

int multiply[][] = new int[m][q];

System.out.println("Enter elements of second matrix");

for (c = 0; c < p; c++)

for (d = 0; d < q; d++)

second[c][d] = sc.nextInt();

for (c = 0; c < m; c++)

{

for (d = 0; d < q; d++)

{

for (k = 0; k < p; k++)

{

sum = sum + first[c][k]\*second[k][d];

}

multiply[c][d] = sum;

sum = 0;

}

}

System.out.println("Product of the matrices:");

for (c = 0; c < m; c++)

{

for (d = 0; d < q; d++)

System.out.print(multiply[c][d]+"\t");

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

}

}

}

public static void main(String args[])

{

multiplymatrix();

addmatrix();

}

}

Results:

