LAB 1

→ Java program to print real solutions of quadratic eqn $ax^2+bx+c=0$

$D = b^2 - 4ac$

```
import java.util.Scanner;
class Quadratic {
public static void main(System.in):
Scanner in = new Scanner (System.in);
double a, b, c, d, x1, x2;
System.out.println("enter the value of variable a");
  a = in.nextDouble();
System.out.println("enter the value of variable b");
  b = in.nextDouble();
System.out.println("enter the value of variable c");
  c = in.nextDouble();
  d = (b*b) - 4*a*c;
  if (d > 0){
  x1 = (-b + Math.sqrt(d))/(2*a);
  x2 = (-b - Math.sqrt(d))/(2*a);
  System.out.println("the roots are real and distinct");
  System.out.println("the roots are "+x1+"and"+x2);
  }
  else if (d==0){
  x1 = x2 = -b/(2*a);
  System.out.println("the roots are real and equal");
  System.out.println("the roots are "+x1+"and"+x2);
  }
  else if (d<0)
  {
  System.out.println("there are no real roots");
  }
  }
  }
```

LAB - 2

```
import java.util.Scanner;
class Student{
 String usn, name;
 static int credits[];
 static double marks[];
void input(int n)
{
Scanner sc = new Scanner(System.in);
System.out.println("enter usn and name");
 usn = sc.nextline();
 name = sc.nextline();
System.out.println("enter the marks with credits
      of that subject");
 for(int i=0; i<n; i++)
   {
    marks[i] = sc.nextDouble();
    credits[i] = sc.nextInt();
    System.out.println();
   }
}
```

```
double calculate (int n)
{
  int c, cred=0;
  double tot, total = 0.0;
  for (int i=0; i<n; i++)
  {
    tot = mark[i];
    if (tot>90)
        c=10;
      else if (tot > 80)
        c = 9;
      else if (tot> 70)
        c = 8;
      else if (tot > 60)
        c=7;
      else if (tot > 50)
        c = 6
      else if (tot > 40)
        c = 5;
      else
      c = 0;
  total = total + (c*credits[i]);
    cred = cred+ credits[i];
  }
  total = total/cred;
    return (total);
  }
```

```
void display( int n, double total)
{
System.out.println("name of student : "+name);
System.out.println("usn of student : "+usn);
System.out.println("marks of student along with cours");
for (int i =0; i<n; i++)
{
System.out.println(marks[i]+" "+credits[i]);
}
System.out.println("sgpa of Student :"+total);
}
public static void main(String args[])
{
Scanner sc= new Scanner(System.in);
Student obj = new student();
System.out.println("enter the number of courss");
int n = sc.nextInt();
credits = new int[n];
marks = new double[n];
obj.input(n);
double total = obj.calculate(n);
obj.display(n, total);
}
}
```

LAB 3

```
import java.util.Scanner;
class Book{
String name;
String author;
int price;
int num_pages;
Scanner p = new Scanner(System.in);
void details(){
System.out.println("Enter the name of the book");
name = p.nextLine();
System.out.println("Enter the name of author");
author = p.nextLine();
System.out.println("Enter the price of book");
price = p.nextInt();
System.out.println("Enter the number of pages in the bo
num_pages = p.nextInt();
}
public String toStrig(){
return ("name :"+name+"\n author :"+author+"\n price
+price+"\n num_pages :"+num_pages);
}
}
class Bk{
public static void main(String args[]){
Scanner q = new Scanner(System.in);
Book obj[] = new Book[10];
int n;
System.out.println("Enter the number of objects required
n = q.nextInt();
```

```
for (int i=0 ; i<n ; i++)
{
  obj[i] = new Book();
  obj[i].details();
  System.out.println(" "+obj[i].toString());
  }
  }
  }
```

⇒ Using Abstract to override methods [LAB-4]

```
import java.util.Scanner;
abstract class Shape{
  double dim1, dim2;
  Shape (double d1, double d2){
      dim1 = d1;
      dim2 = d2;
    }
  abstract double print_area();
  }
  class Rectangle extends Shape{
   Rectangle (double d1, double d2){
     super (d1, d2);
    }
   double print_area(){
    return dim1 * dim2;
   }
  }
  class Triangle extends Shape{
   Triangle (double d1, double d2){
       super(d1, d2);
     }
   double print_area (){
    return (dim1 * dim2)/2;
   }
  }
```

```
class Circle extends Shape {
  Circle (double d1, double d2){
   super (d1, d2);
  }
  double print_area() {
   return 3.14 * dim1 * dim2;
  }
  }
  class Demo4
   public static void main (String args[]) {
        Rectangle r = new Rectangle (5,10);
        Triangle t = new Triangle (5,10);
        Circle   c = new Circle (5,5);
         Shape s;
          s = r;
        System.out.println ("The area of Rectangle:" + s.print_area
                                                        ());
  s=t;
  System.out.println("The area of triangle:" + s.print_area());
  s=c;
  System.out.println("The area of Circle:" + s.print_area());
  }
}
```

LAB - 5

⇒ Using A Bank class to derive Savings and current account

```
import java.util.Scanner;
class Bank {
    public static void main(String args[]){
        boolean nxt=true;
    Scanner sc = new Scanner(System.in);
    while (nxt) {
    System.out.println("Enter 1 for 'Saving Account'");
    System.out.println("Enter 2 for 'current Account'");
System.out.println("Enter the type of account");
int n = sc.nextInt();
String s = sc.nextLine();
if(n==1){
    Sav_acct ob = new Sav_acct();
    System.out.println("Enter name");
    ob.name = sc.nextLine();
    System.out.println("Enter account number");
    ob.accno = sc.nextInt();
    ob.acceptBalance();
    ob.display();
    ob.Compute();
    ob.withdraw();
    ob.chqbk();
}
else {
    Cur_acct ob = new Cur_acct();
    System.out.println("Enter Name");
ob.name = sc.nextLine();
System.out.println("Enter acc number");
    ob.accno = sc.nextInt();
```

```
ob.acceptBalance();
ob.checkmin();
ob.display();
ob.withdraw();
ob.chqbk();
}
System.out.println("Enter 1 for next customer, Enter 2 to end");
int c = sc.nextInt();
if (c==1)
continue;
else
nxt=false;
}
}
}
class Account{
String name;
int accno;
String acctype;
}
class Curr_acct extends Account{
double balance;
void acceptBalance(){
Scanner sc = new Scanner(System.in);
System.out.println("Enter deposit amount");
double d = sc.nextDouble();
balance+=d;
}
void display(){
System.out.println("Balance :+balance");
}
void withdraw(){
Scanner sc = new Scanner(System.in);
System.out.println("Enter the amount to withdraw");
int w = sc.nextInt();
balance -=w;
System.out.println("Balance :+balance);}
```

```
}
 Void check min(){
      if (balance < 500){
           balance -=50;
System.out.println("Servic charge of Rs. 50/- has been imposed");
System.out.println("Balance after deductica :"+balance);
  }
  else
   return;
}
 void chqbk(){
     System.out.println(" Name :"+super.name);
  System.out.println("Account Number :"+super.accno);
  System.out.println("Balance :"+balance);
System.out.println("Account type : current account");
  }
}
class Sav_acct extends Account{
     double balance;
      void acceptBalance(){
        Scanner sc = new Scanner (System.in);
System.out.println("Enter deposit amount");
    double d = sc.nextDouble();
     balance+=d;
 }
 void display(){
  System.out.println("Balance :"+balance);
 }
 void compute(){
    Scanner sc=new.Scanner (System.in);
 System.out.println("Enter duration in months");
    int n = sc.nextInt();
      balance +=(0.025*n);
   }
```

```
void withdrawal(){
    Scanner sc = new Scanner (System.in);
 System.out.println ("Enter amount to withdraw");
    int    wd = sc.nextInt();
       balance = balance - wd;
     System.out.println("Balance :"+balance);
  }
  void chqbk(){
        System.out.println("Name :"+super.name);
        System.out.println("Account Number :"+super.accno);
      System.out.println("Balance :"+balance);
      System.out.println("Account type : Savings account");
       }
}
```

LAB 6

Package CIE program

file Internals.java:-

```
package CIE;

import java.util.*;

public class Internals {

    public int internals[] = new int[5];

    Scanner input = new Scanner (System.in);

    public void GetDetails () {

        System.out.println("Enter marks of CIE :\n");

        for (int j=1; j<=5; j++){

            System.out.println("Enter marks of course : "+j);

            internals[j-1] = input.nextInt();

        }

    }

}
```

Student.java:-

```
package CIE;

public class Student {

    public int usn;

    public String name;

    public int sem;

}
```

externals.java :-

```
package sec;
import java.util.Scanner;
Import CIE.*;
public class externals extends Student {
    public int Semester[] = new int[5];
    Scanner in = new Scanner(System.in);
public void EnterDetails(int i){
    System.out.println("Student "+i);
    System.out.println("Enter the usn of the Student:\n");
    usn = in.nextInt();
    System.out.println("Enter the name of the student:\n");
    name = in.next();
    System.out.println("Enter the sem of the student:\n");
    sem = in.nextInt();
    System.out.println("Enter the marks of SEE :\n");
        for(int j=0; j<5; j++){
            System.out.println("Enter the marks of course"+(j+1));
            Semester[j] = in.nextInt();
        }
    }
}
```

marks.java

```
package marks;
import CIE.*;
import see.*;
import java.util.*;

class marks {

    public static void main (String args[]) {
        int n;
        Scanner in = new Scanner (System.in);
        System.out.println ("Enter the number of students :\n");
        n=in.nextInt();
        Internal [] cie = new Internals [n];
        externals[] sem = new externals[n];

    for (int i=0; i<n; i++) {
        sem[i] = new externals();
        sem[i].EnterDetails(i+1);
        cie[i] = new Internals();
        cie[i].GetDetails();
    }
    System.out.println ("The Marks and Details of Students Registered are:\n");
for (int j=0; j<n; j++){
        System.out.println (" Student " + (j+1));
        System.out.println ("USN: " +sem[j].usn);
        System.out.println( "Name : "+sem[j].name);
        System.out.println (" Semester : "+sem[j].sem);
    System.out.println ("& Final marks :\n");
for (int i=0; i<5; i++) {
    System.out.println ("Course " + (i+1) + "&"\n" + ((sem[j].semester[i])/2
                        + cie[j].internals[i]));
        System.out.println();
    }
}
}
```

Test.java:-

```
package test;
import java.util.*;
class std
{
String usn, name;
 static int credits[];
 static double marks[];
void input(int n)
{
Scanner sc = new Scanner(System.in);
System.out.println("enter usn and name:");
 usn = sc.nextLine();
 name = sc.nextLine();
System.out.println("enter marks along with credit");
for(int i=0; i<n; i++)
 {marks[i] = sc.nextDouble();
  credits[i]= sc.nextInt();
  System.out.println();
  }
 }
double calculate(int n)
{
int c, cred=0;
double tot; total=0.0;
for(int i=0; i<n; i++)
 {
 tot = marks[i];
 if (tot>=90)
        c=10;
        else if(tot>=80)
        c=9;
        else if(tot>=70)
        c=8;
        else if(tot>=60)
```

```
c=7;
else if (tot >= 50)
c=6;
else if (tot >=40)
c=4;
else
c=0;
total = total + (credits[i]);
cred = cred + credits[i];
}
total = total / cred;
return (total);
}
void display ( int n, double Total)
{
System.out.println("name of student : "+name);
System.out.println("usn of student : "+usn);
System.out.println("marks of student along with credits of
course");
for (int i = 0 ; i<n ; i++)
{
System.out.println(marks[i]+" ,"+ credits[i]);
}
System.out.println("sgpa of student : "+total);
}
class Test{
public void main (String args[])
{
Scanner sc = new Scanner (System.in);
Std obj = new Std();
System.out.println("enter no of course");
int n = sc.nextInt();
```

```
        obj.credith = new int[n];
        obj.marks = new double[n];
        obj.input(n);
        double total = obj.calculate(n);
        obj.display(n, total);
    }
}
}
```

LAB 7

```
class TwoGen<T,V>{
    T ob1;
    V ob2;
    TwoGen(T o1, V o2){
    ob1 = o1;
    ob2 = o2;
    }
void ShowTypes(){
System.out.println("Type of T is " + ob1.getclass().getName());
System.out.println("Type of V is " + ob2.getclass().getName());
}
T.getob1(){
  return ob1;
  }
V getob2(){
  return ob2;
 }
}
class SimpGen{
  public static void main(String args[]){
TwoGen<Integer, String> tgobj =
  new TwoGen<Integer, String>(88, "Generics");
tgobj.ShowTypes();
int v = tgObj.getob1();
System.out.println("value: "+v);
String str = tgObj.getob2();
System.out.println("value: "+str);
}}
```

LAB8

```
package interfaces_exceptionhandling;
    class Father extends Exception{
        private int Fage;
        Father(){}
        Father(int a){
        Fage=a;
            }
    public String toString(){
            return "Father age cannot be negetive: ";
            }
    }
class Son extends Father{
        int sage;
        int fage;
        Son(int s, int f){
            sage = s;
            fage = f;
        }
        public String toString(){
            return "Father age cannot be less than or equal to son age
            }
    }
    class Exception{
        static void ageCompare(int fage, int sage) throws Son{
                if(fage <= sage)
                throw new Son(sage, fage);
        }
```

```
static void NegativeAgeCheck (int age) throws Father {
        if (age < 0)
            throw new Father (age);
    }
public static void main (String args[]){
    try {
        NegativeAgeCheck (-30);
    } catch (Father e) {
        System.out.println(e);
    }
    try { AgeCompare (16,26); }
    catch (Son e) {
        System.out.println (e);
    }
}
```

LAB 9

```
class NewThread extends Thread {

  NewThread () {
   super ("Demo thread");
  System.out.println ("child thread : "+this);
   start();
  }
  public void run () {
   try {
   for (int i =0; i<15; i++){
   System.out.println (" child thread : CSE ");
     Thread . sleep (2000);
   }
   } catch (InterruptedExeption e) {
     System.out.println (" child interrupted:");
    }
  System.out.println ("Exiting child thread");
  }
  }
  class Thread2 {
  public static void main (String args[]){
   new NewThread ();
   try {
    for (int i=0; i<3 ; i++) {
   System.out.println ("Main Thread : BMS college of Engineering");
   Thread.sleep (10000);
  }
  }
```

```
Catch (InterruptedException e) {
 System.out.println("Main thread interrupted.");
}
System.out.println("Main thread exiting.");
}
}
```

LAB 10:-

```
import java.awt.*;
import java.awt.event.*;
public class Lab10 extends Frame implements ActionListener{
    TextField num1, num2;
    Label ob;
    Button n;
    Lab10(){
    num1 = new TextField();
    num1.setBounds(50, 50, 200, 25);

    num2 = new TextField();
    num2.setBounds(50, 100, 200, 25);

    ob = new Label();
    ob.setBounds(50, 150, 300, 50);

    n = new Button("Divide");
    n.setBounds(50, 200, 100, 50);
    n.addActionListner(this);

    add(n);
    add(num1);
    add(num2);
    add(ob);
    setsize(800, 800);
    setLayout(null);
    setVisible(true);
    }
```

```
public void actionPerformed (ActionEvent e) {

    try {
        String n1 = num1.getText();
        String n2 = num2.getText();
        ob.setText("Quotient:" + (Integer.parseInt(n1)/Integer.parseInt
                                  (n2)));
    }
    catch (NumberFoundExeption ze) {
        ob.setText("Cannot divide non-inter values");
    }
    catch (ArithematicExeption ze) {
        ob.setText("CannotDivide");
    }
    catch (Exeptin ex) {
        System.out.println(ex);
    }
}
public static void main (String[] args) {
    new lab10();
}
```