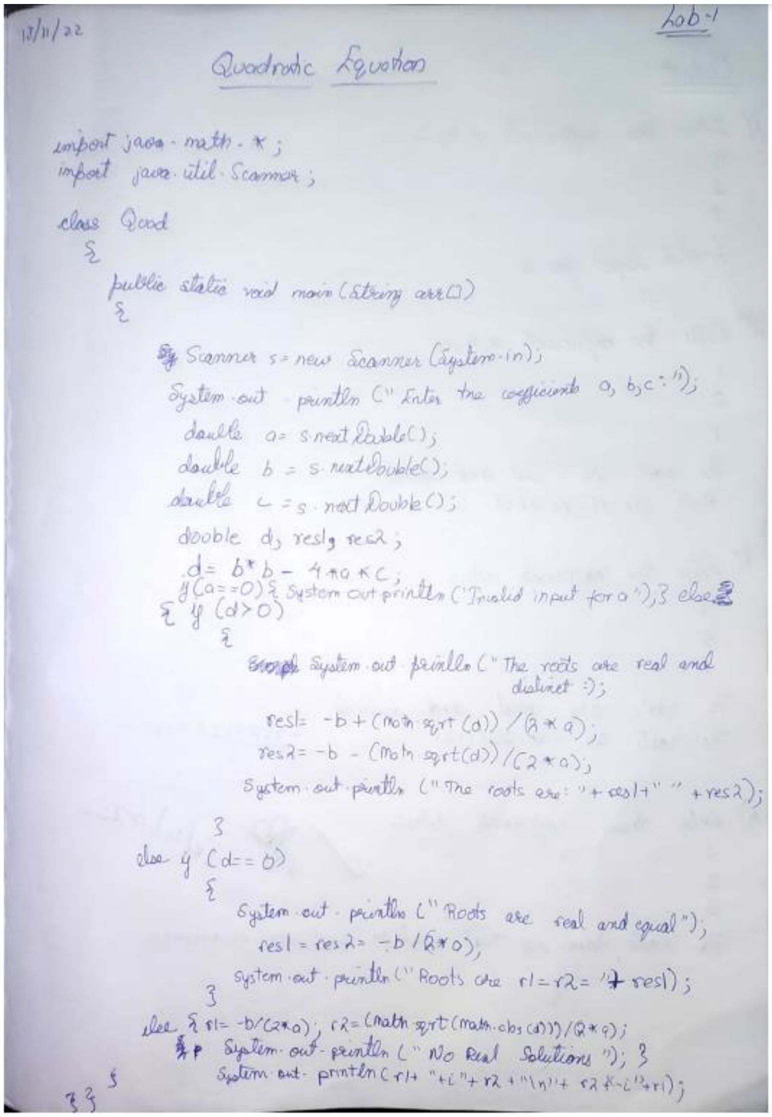
LAB PROGRAM-1

Develop a Java program that prints all real solutions to the quadratic equation ax2+bx+c = 0. Read in a, b, c and use the quadratic formula. If the discriminate b2 -4ac is negative, display a message stating that there are no real solutions.

PROGRAM

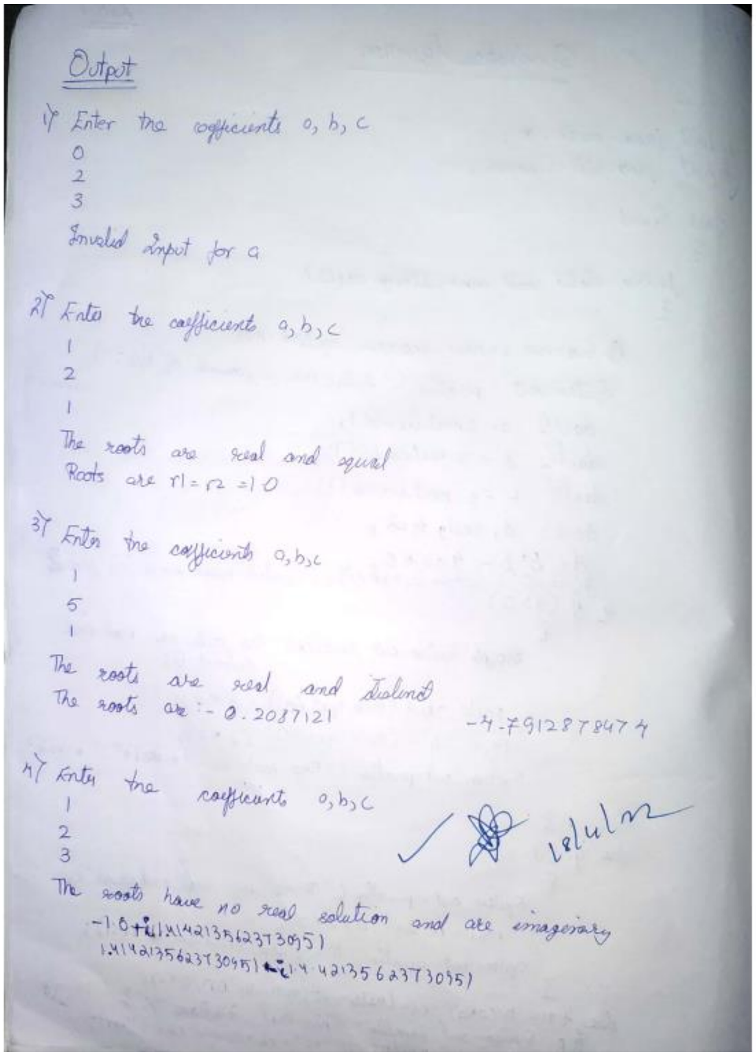


12/11/22

Lab-1

Quadratic Equation

```
import java.math.*;
import java.util.Scanner;

class Quad
 {
   public static void main(String arr[])
   {
       Scanner s= new Scanner(System.in);
       System.out.println("Enter the coefficients a, b,c:");
        double a= s.nextDouble();
        double b = s.nextDouble();
        double c =s.nextDouble();
        dooble d, res1, res2;
        d= b*b - 4*a*c;
        if(a==0){ System.out.println("Invalid input for a");} else
      { if (d>0)
          {
              System.out.println("The roots are real and
                                   distinct");
              res1= -b+(math.sqrt(d))/(2*a);
              res2= -b - (math.sqrt(d))/(2*a);
             System.out.println("The roots are: "+res1+" " +res2);

          }
     else if (d==0)
          {
             System.out.println("Roots are real and equal");
              res1 = res2= -b/(2*a);
             System.out.println("Roots are r1=r2= "+res1);
          }
     else { r1= -b/(2*a); r2=(math.sqrt(math.abs(d)))/(2*a);
          System.out.println("No Real Solutions"); }
            System.out.println(r1+ "+i"+r2 +"\n"+ r2+"-i"+r1);
   }
}}
```



Output

1) Enter the cofficients a, b, c

```
0
2
3
Invalid input for a
```

2) Enter the coefficients a,b,c

```
1
2
1
The roots are real and equal
Roots are r1=r2=1.0
```

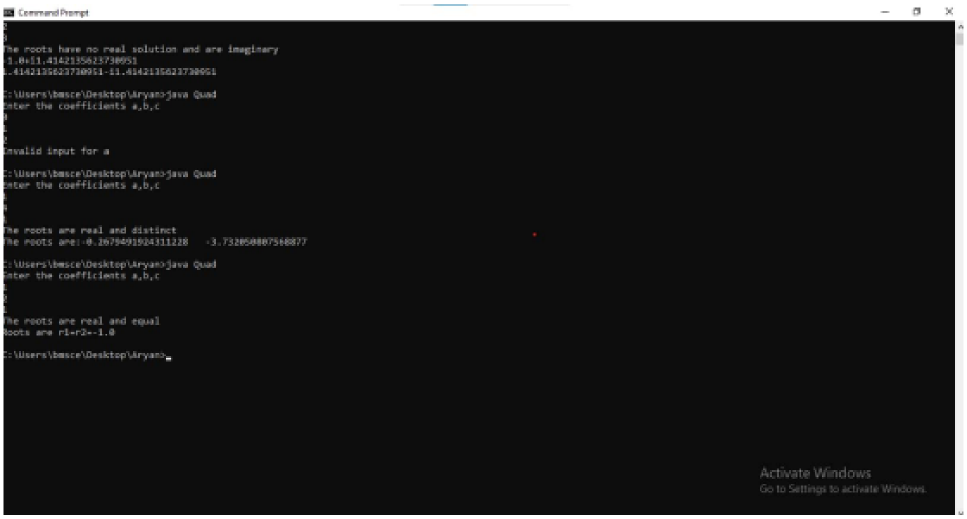
3) Enter the cofficients a,b,c

```
1
5
1
The roots are real and distinct
The roots are:-0.2087121          -4.7912878474
```

4) Enter the coefficients a,b,c

```
1
2
3
The roots have no real solution and are imaginary
-1.0+i1.4142135623730951
1.4142135623730951-i1.4142135623730951
```

✓ 18/4/22

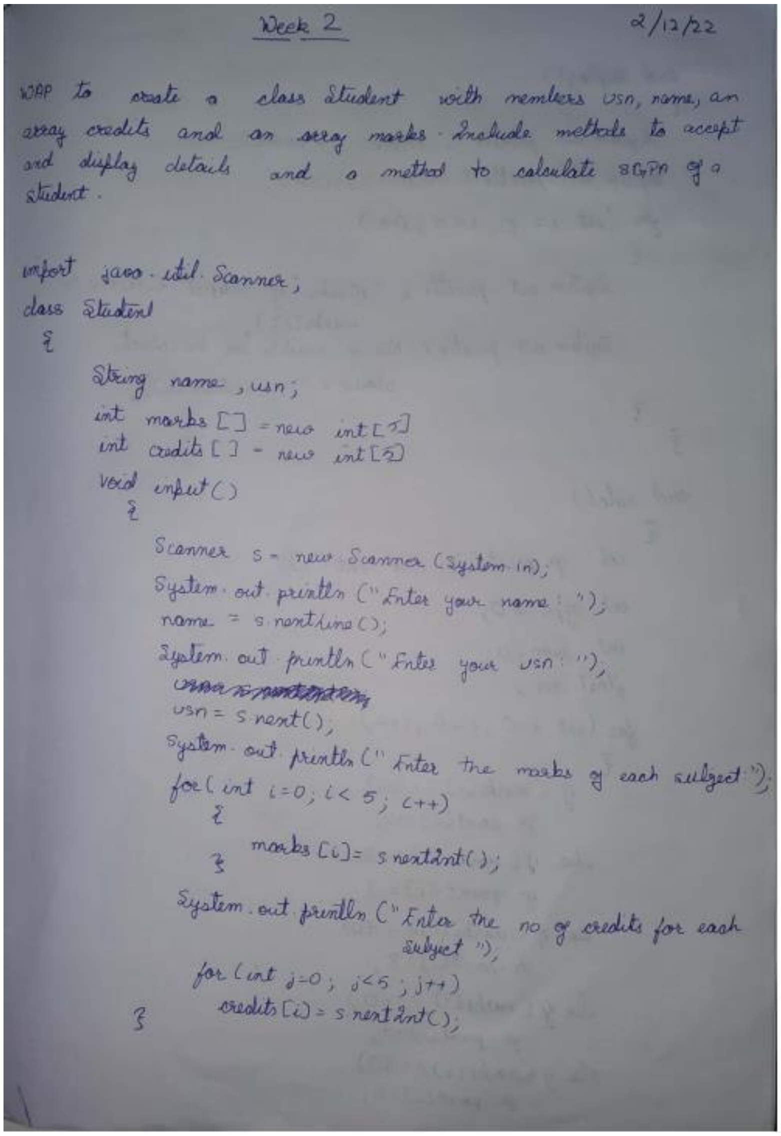


```
2
3
The roots have no real solution and are imaginary
-1.0+i1.4142135623730951
1.4142135623730951-i1.4142135623730951

C:\Users\bmsce\Desktop\Aryan>java Quad
Enter the coefficients a,b,c
0
1
2
Invalid input for a

C:\Users\bmsce\Desktop\Aryan>java Quad
Enter the coefficients a,b,c
1
4
1
The roots are real and distinct
The roots are:-0.2679491924311228   -3.732050807568877

C:\Users\bmsce\Desktop\Aryan>java Quad
Enter the coefficients a,b,c
1
2
1
The roots are real and equal
Roots are r1=r2=-1.0

C:\Users\bmsce\Desktop\Aryan>_
```

LAB PROGRAM -2

Develop a Java program to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.

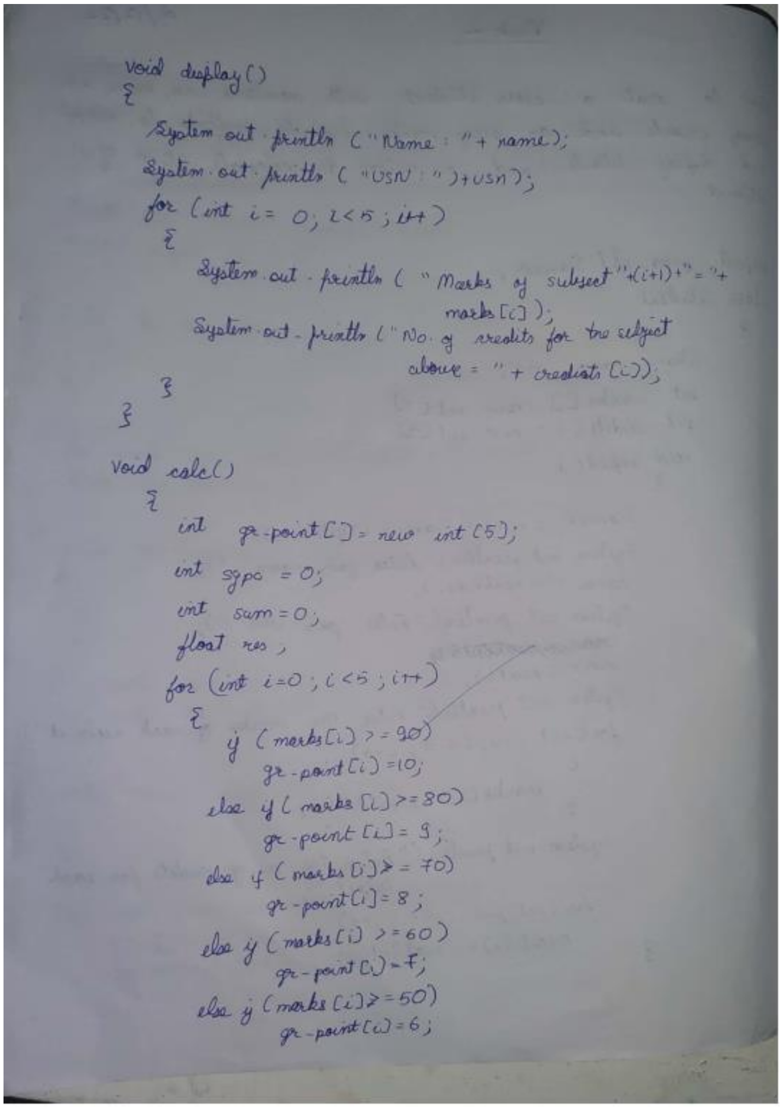
PROGRAM



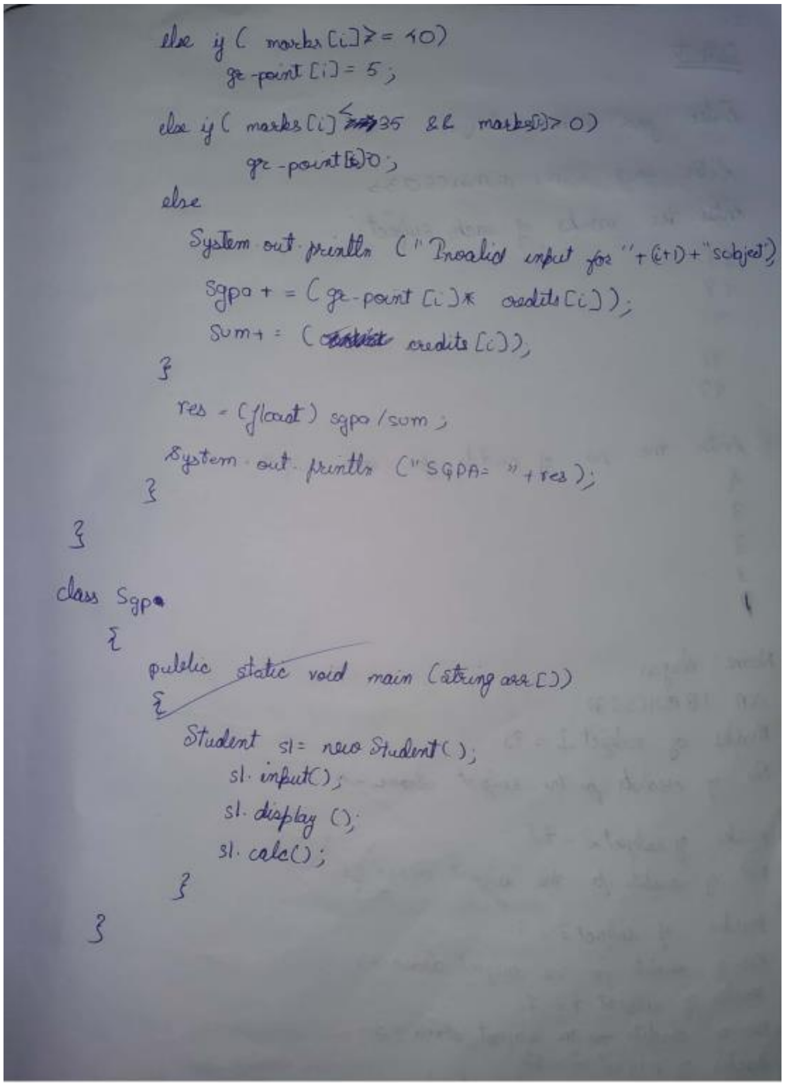
Week 2 2/12/22

WAP to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.

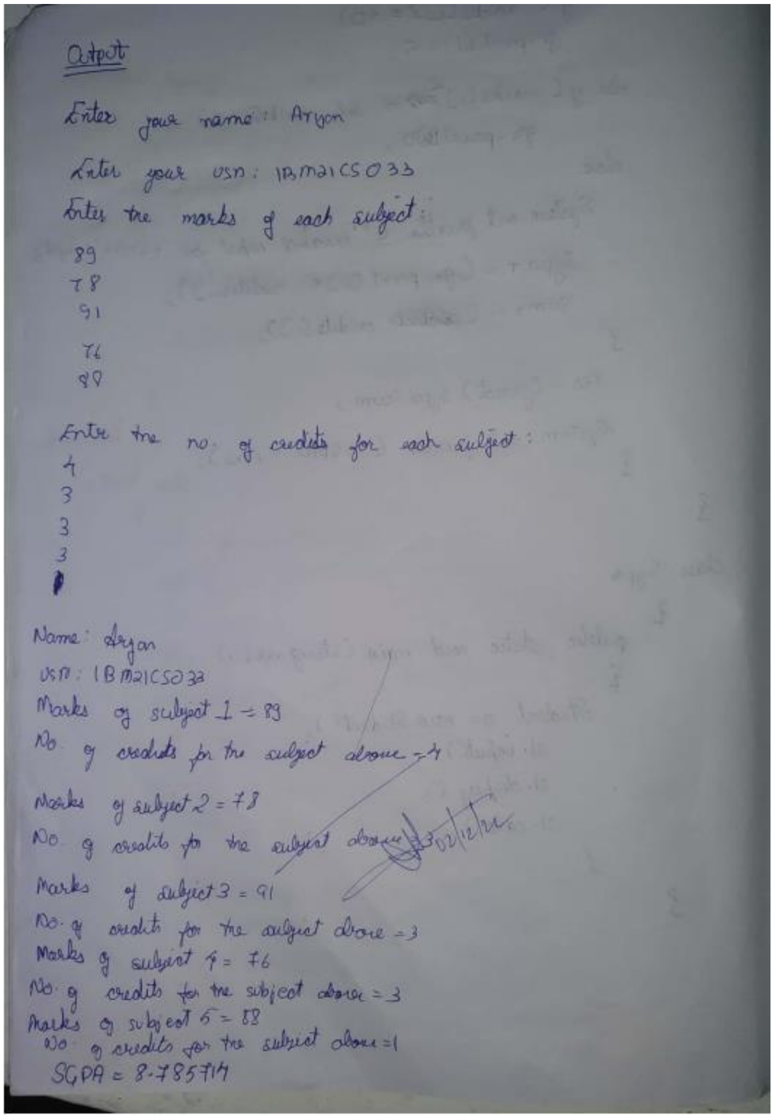
```
import java.util.Scanner;
class Student
 {
    String name , usn;
    int marks [] = new int[5]
    int credits [] - new int[5]
    void input()
     {
        Scanner s - new Scanner (System.in);
        System.out.println ("Enter your name : ");
        name = s.nextLine();
        System.out.println ("Enter your usn : ");
        usn = s.next();
        System.out.println (" Enter the marks of each subject ");
        for( int i=0; i< 5; i++)
         {
             marks [i]= s.nextInt();
         }
         System.out.println (" Enter the no of credits for each
                                 subject ");
         for (int j=0; j<5 ; j++)
           credits [i] = s.nextInt();
     }
```



```
void display()
{
  System.out.println ("Name : "+ name);
  System.out.println ( "USN : ")+usn);
  for (int i= 0; i<5 ; i++)
   {
      System.out.println ( "Marks of subject "+(i+1)+" = "+
                       marks[i] );
      System.out.println ("No. of credits for the subject
                          above = "+ credits [i]);
   }
}

void calc()
 {
    int  gr-point[] = new int [5];
    int  sgpa = 0;
    int  sum = 0;
    float res ,
    for (int i=0 ; i<5 ; i++)
     {
        if (marks[i] >= 90)
            gr-point[i] =10;
       else if ( marks [i] >= 80)
            gr-point [i] = 9 ;
       else if (marks [i] >= 70)
            gr-point[i] = 8 ;
       else if (marks[i] >= 60)
            gr-point [i] = 7;
       else if (marks [i] >= 50)
            gr-point [i] = 6 ;
```



```
        else if ( marks[i]>= 40)
             gr-point[i] = 5;

        else if ( marks[i] <35  && marks[i]> 0)
                gr-point[i]=0;
        else
          System.out.println ("Invalid input for "+(i+1)+"subject");
            sgpa + = (gr-point[i]* credits[i]);
            sum+ = ( credits[i]);
        }
         res = (float) sgpa /sum;
        System.out.println ("SGPA= "+res);
       }
  }
class Sgpa
   {
     public static void main (string arr[])
     {
        Student s1= new Student();
            s1.input();
            s1.display();
            s1.calc();
        }
   }
```



Output

Enter your name : Aryan

Enter your usn : 1BM21CS033

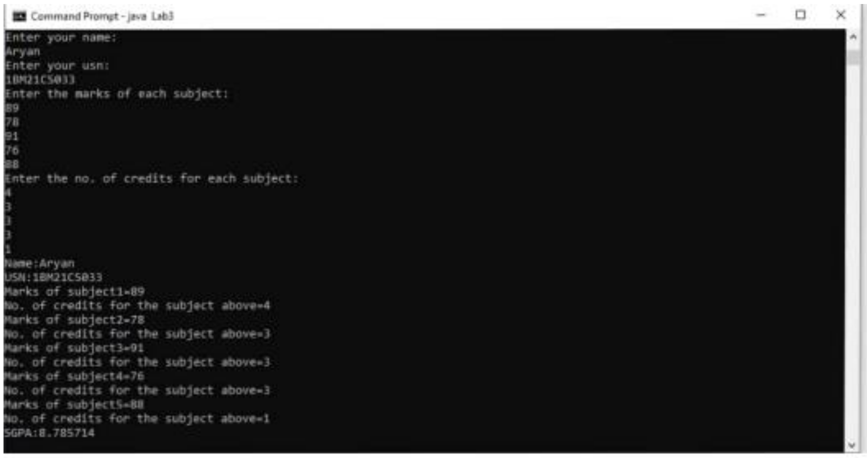
Enter the marks of each subject:

89
78
91
76
88

Enter the no. of credits for each subject:
4
3
3
3
[illegible]

Name: Aryan
USN: 1BM21CS033
Marks of subject 1 = 89
No. of credits for the subject above = 4
Marks of subject 2 = 78
No. of credits for the subject above = 3 02/12/21
Marks of subject 3 = 91
No. of credits for the subject above = 3
Marks of subject 4 = 76
No. of credits for the subject above = 3
Marks of subject 5 = 88
No. of credits for the subject above = 1
SGPA = 8.785714

OUTPUT

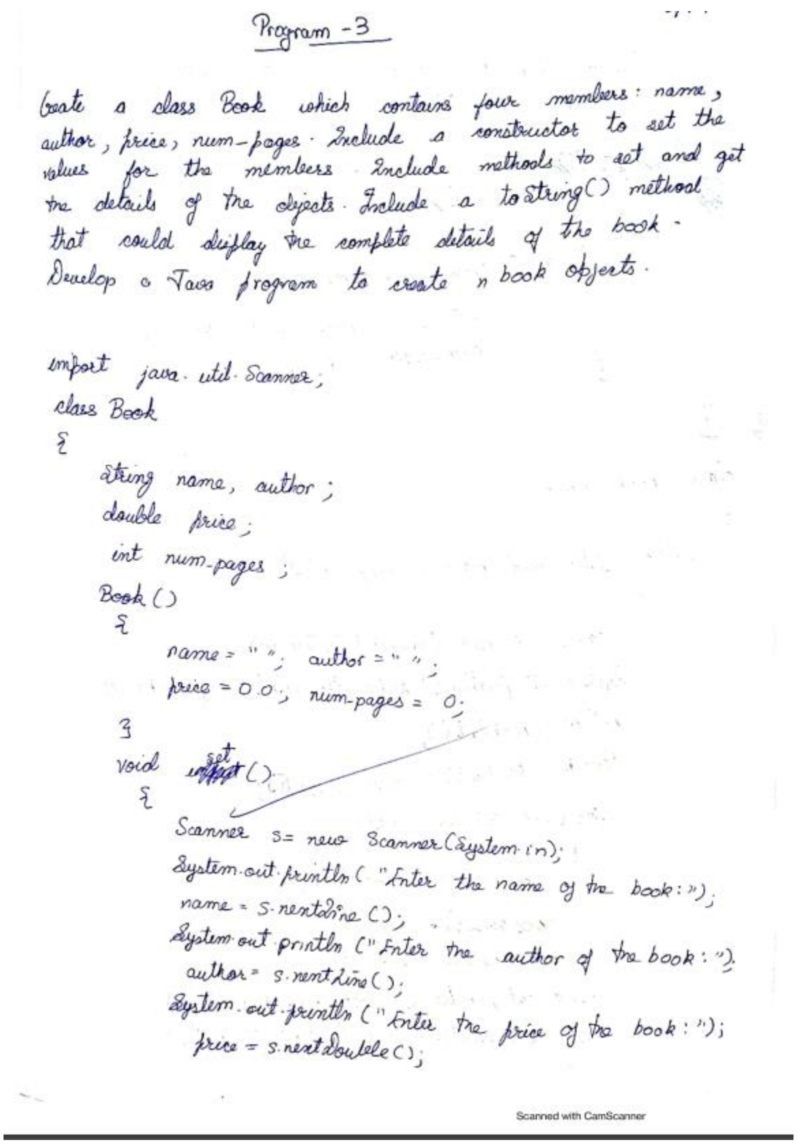


```
Enter your name:
Aryan
Enter your usn:
1BM21CS033
Enter the marks of each subject:
89
78
91
76
88
Enter the no. of credits for each subject:
4
3
3
3
1
Name:Aryan
USN:1BM21CS033
Marks of subject1=89
No. of credits for the subject above=4
Marks of subject2=78
No. of credits for the subject above=3
Marks of subject3=91
No. of credits for the subject above=3
Marks of subject4=76
No. of credits for the subject above=3
Marks of subject5=88
No. of credits for the subject above=1
SGPA:8.785714
```

LAB PROGRAM-3

Create a class Book which contains four members: name, author, price, num_pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a toString( ) method that could display the complete details of the book. Develop a Java program to create n book objects.

PROGRAM



Program -3

Create a class Book which contains four members: name, author, price, num-pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a toString() method that could display the complete details of the book. Develop a Java program to create n book objects.

```
import java.util.Scanner;
class Book
{
    String name, author;
    double price;
    int num-pages;
    Book ()
     {
         name = " "; author = " ";
         price = 0.0; num-pages = 0;
     }
     void ~~input~~ set ()
      {
         Scanner s= new Scanner(System.in);
         System.out.println( "Enter the name of the book:");
         name = s.nextLine ();
         System.out.println ("Enter the author of the book:");
          author= s.nextLine();
         System.out.println ("Enter the price of the book:");
          price = s.nextDouble();
```

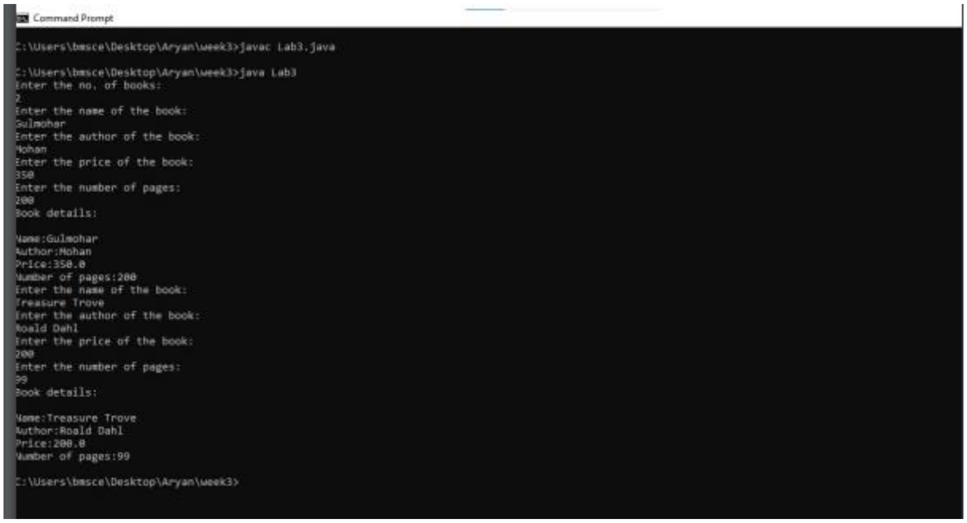
```
        System.out.println ("Enter the number of pages :");
        num_pages = s.nextInt();
    }
    public String toString ()
      {
        return ("Name : "+ name + " \n Author : "+ author +
                 "\n Price: "+ price f\n Number of pages :"+
                  num_pages + "\n");
      }
}
class Book_main
{
  public static void main (String x[])
    {
      Scanner s= new Scanner (System.in);
      System.out.println ("Enter the number of books");
      int n= s.nextInt();
      Book  books [] = new Book[n];
      for (int i=0; i<n ; i++)
        {
          books [i]= new Book();
          books [i]. set ();
      System.out.println ("Book Details :\n");
      System.out.println ( books [i] . get());
  }     }
}
```

Output

Enter the number of books : 2
Enter the name of the book: Gulmohar
Enter the author of the book: Mohan
Enter the price of the book: 350.0
Enter the number of pages : 200
Book Details:
Name : Gulmohar
Author : Mohan
Price : 350
Number of pages: 200
Enter the name of the book : Treasure Trove
Enter the author of the book : Roald Dahl
Enter the price of the book : 200.0
Enter the number of pages : 99
Book Details :
Name : Treasure Trove
Author: Roald Dahl
Price : 200.0
Number of pages: 99

02/12/22

OUTPUT

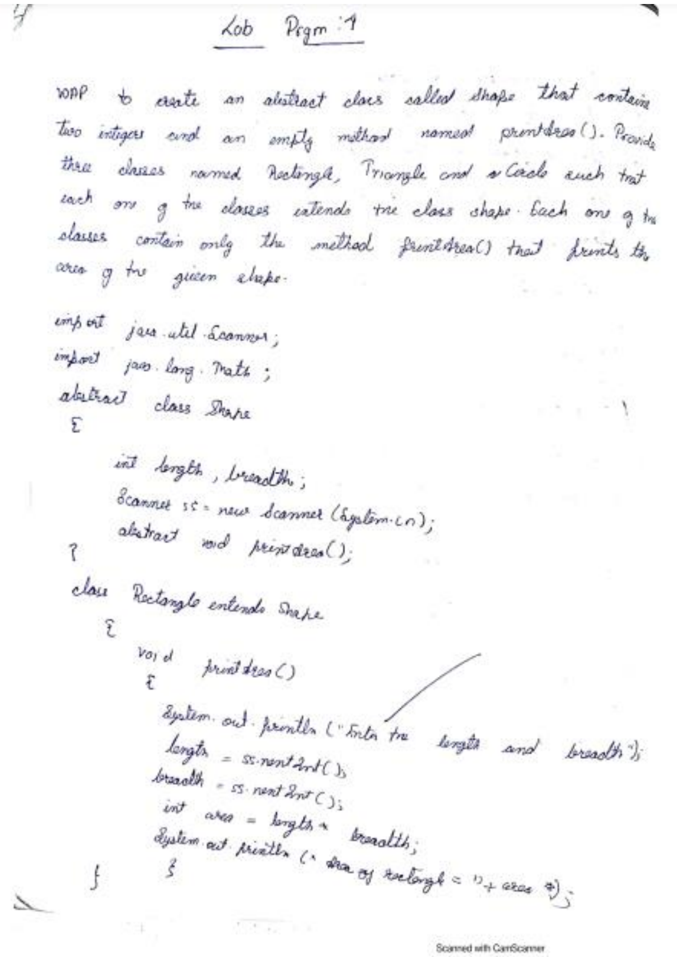


```
C:\Users\bmsce\Desktop\Aryan\week3>javac Lab3.java

C:\Users\bmsce\Desktop\Aryan\week3>java Lab3
Enter the no. of books:
2
Enter the name of the book:
Gulmohar
Enter the author of the book:
Mohan
Enter the price of the book:
350
Enter the number of pages:
200
Book details:

Name:Gulmohar
Author:Mohan
Price:350.0
Number of pages:200
Enter the name of the book:
Treasure Trove
Enter the author of the book:
Roald Dahl
Enter the price of the book:
200
Enter the number of pages:
99
Book details:

Name:Treasure Trove
Author:Roald Dahl
Price:200.0
Number of pages:99

C:\Users\bmsce\Desktop\Aryan\week3>
```

LAB PROGRAM-4

Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea( ). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea( ) that prints the area of the given shape.

PROGRAM



Lab Prgm :4

WAP to create an abstract class called Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and a Circle such that each one of the classes extends the class shape. Each one of the classes contain only the method printArea() that prints the area of the given shape.

```
import java.util.Scanner;
import java.lang.Math;
abstract class Shape
 {
      int length, breadth;
      Scanner ss = new Scanner(System.in);
      abstract void printArea();
 }
 class Rectangle extends Shape
    {
       void printArea()
        {
          System.out.println("Enter the length and breadth");
          length = ss.nextInt();
         breadth = ss.nextInt();
          int area = length * breadth;
         System.out.println("Area of rectangle = "+ area);
          }
    }
```

```
class Triangle extends Shape
 {
    void printArea()
    {
        System.out.println("Enter base , length and height");
        length = ss.nextInt();
        breadth = ss.nextInt();
        int area = (length * breadth)/2;
        System.out.println(" Area of the triangle = "+ area);
    }
 }
  class Circle extends Shape
     {
        void printArea()
        {
           ~~Scanner~~ System.out.println("Enter the radius");
             length=ss.nextInt();
               ~~int area~~
               double area = Math.PI *( length * length);
           System.out.println(" Area of the circle = "+area);
        }
   }
class Shape_Main
 {
  public static void main(String arr[])
  {
     int ch;
     Scanner ~~mam~~ s1=new Scanner (System.in);
    System.out.println("Select Shape → 1. Rectangle \n 2. Triangle \n
                                      3. Circle \n");
~~[illegible]~~ ch=sl.nextInt();
switch (ch)
 {
   case 1: Rectangle . r= new Rectangle();
               r.printArea();
               break;
   case 2: Triangle t= new Triangl();
               t.printArea();
               break;
   case 3: Circle c = new Circle ();
               c.printArea();
               break;
     default : System out. println ("Invalid Input");
 }
    }
}
```

<u>Output</u>

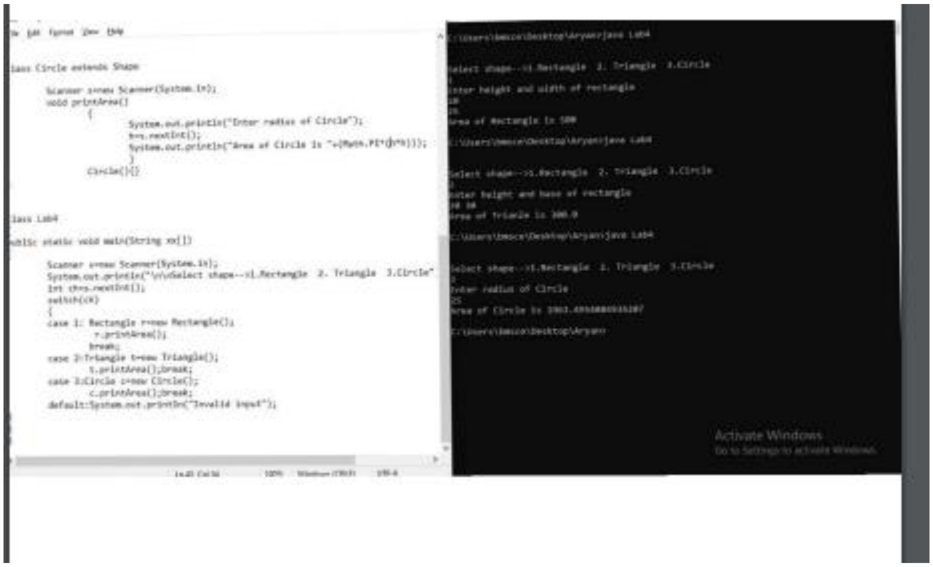
```
Select shape
1. Rectangle
2. Triangle
3. Circle
1
Enter length and breadth
  10 20
Area of Rectangle= 200
```

② Select Shape
1. Rectangle
2. Triangle
3. Circle
2
Enter base length and height
20 20
Area of triangle =
200

③ Select Shape
1. Rectangle
2. Triangle
3. Circle
3
Enter the radius
10
Area of the circle =
314

09/10

OUTPUT



```
class Circle extends Shape

        Scanner s=new Scanner(System.in);
        void printArea()
                {
                        System.out.println("Enter radius of Circle");
                        b=s.nextInt();
                        System.out.println("Area of Circle is "+(Math.PI*(b*b)));
                        }
                Circle(){}


class Lab4

public static void main(String xx[])

        Scanner s=new Scanner(System.in);
        System.out.println("\n\nSelect shape-->1.Rectangle  2. Triangle  3.Circle"
        int ch=s.nextInt();
        switch(ch)
        {
        case 1: Rectangle r=new Rectangle();
                 r.printArea();
                break;
        case 2:Triangle t=new Triangle();
                t.printArea();break;
        case 3:Circle c=new Circle();
                c.printArea();break;
        default:System.out.println("Invalid input");
```

```
C:\Users\bmsce\Desktop\Aryan>java Lab4


Select shape-->1.Rectangle  2. Triangle  3.Circle
1
Enter height and width of rectangle
20
25
Area of Rectangle is 500

C:\Users\bmsce\Desktop\Aryan>java Lab4


Select shape-->1.Rectangle  2. Triangle  3.Circle
2
Enter height and base of rectangle
20 30
Area of Trianle is 300.0

C:\Users\bmsce\Desktop\Aryan>java Lab4


Select shape-->1.Rectangle  2. Triangle  3.Circle
3
Enter radius of Circle
25
Area of Circle is 1963.4954084936207

C:\Users\bmsce\Desktop\Aryan>
```

LAB PROGRAM -5

Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, account number and type of account. From this derive the classes Cur-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks:

a) Accept deposit from customer and update the balance.

b) Display the balance.

c) Compute and deposit interest

d) Permit withdrawal and update the balance .

Check for the minimum balance, impose penalty if necessary and update the balance.

PROGRAM

Lab Pgm 5

WAP to create a class Bank that maintains two kinds of accoun for its customers, one called savings account and the other current account. The savings account provide compound interest and withd facilities but no cheque book facility. The current account provide cheque book facility but no ~~withou~~ ~~[illegible]~~ interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, acc number and type of account. From this deri the classes cur-acct and sav-acct to make them more specific to th requirements. Include the necessary methods in order to achieve the following tasks:

a) Accept deposit from ~~a~~ customer and update the balance
b) Display the balance
c) Compute and deposit interest
d) Permit withdrawal and update balace

Check for minimum balace, impose penalty and upade the balace

```
import java.util.Scanner;
import java.lang.Math;
class Account
  { ~~A~~
      String name, acc_type;
      int acc_no;
      double bal, dep;
      Scanner s = new Scanner (System.in);
```

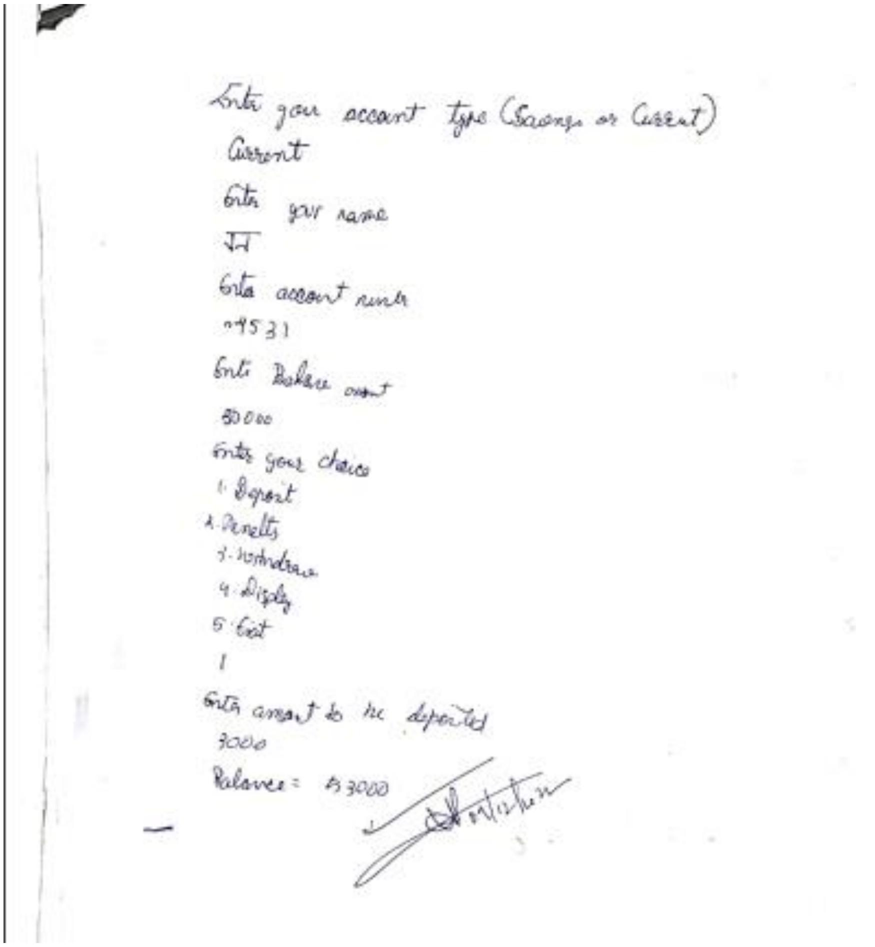
```
void setd()
  {
     System.out.println ("Enter your name :\n");
      name = ss.next();
     System.out.println ("Enter your account number");
      acc-no =ss.nextInt ();
      System.out.println ("Entr your account type : (Savings or Current)\n)
       acc_type =ss.nextInt();
      System.out.println ("Entr the bank balance :\n");
        bal=ss.nextDouble(),
  }
void dop()
  {
    System.out.println ("Name : "+name);
    System.out.println (" Account No : "+acc_no);
    System.out.println ( 'Account type : "+acc-type);
    System.out.println ("Current balance in: "+ bal);
  }
void deposit ()
    {
     System.out.println ("Entr the amt to be deposited ");
      dep= ss.nextDouble();
      bal+ = dep;
      System.out.println (" Balance account : +bal )
     }
   boolean acc (String acc_type)
    {
     if (acc_type = = "Savings")
        return true;
   else if (acc_type== "Current ")
      return false
      else
        return true ;
     }
  }
class Cur_acct extends Account
     {
       int penal()
        {  double min , pen;
           System.out.println ("\n Enter minimum balance & penalty amount
                                   if not followed :");

           min = ss.nextDouble();
           pen = ss.nextDouble();

         if (bal <min)
          {
             bal - = pen;
             System.out.println ("Penalty imposed for having insufficient
                                   Balance Balance");
          }
        else
          return 1;
        }
     void   withdrawal()
         {  double  amt ;
           System.out.println ("Enter the amount to be withdrawn :");
            amt = ss.nextDouble():
            int a = penal();
```

```
if(a==1)
  {
   if(bal>=amt)
   {
     bal-= amt
     System.out.println ("Account Balance after withdrawal is:"+bal);
   }
  }
  else
   System.out.println("\n The amount can't be withdrawn");
  }
 }

class Sav_acct extends Account
   {
      void calc_interest()
      {
      System.out.println ("Enter time and rate of interest");
       double t = ss.nextDouble();
       double r = ss.next Double();
       double CI = bal * Math.pow ((1+r)/100), t);
       System.out.println ("\n Compound Interest is :"+CI);
         bal+=CI;
       System.out.println ("Balance amount :"+bal);
    }
   void withdrawal()
    {
      double amt;
     System.out.println(" Enter amount to be withdrawn:");
      amt = ss.nextDouble();
```

```
        if (bal >= amt)
         {
           bal -= amt;
           System.out.println(" Account balance after withdrawal is:- " + bal);
         }
     }
     else
      System.out.println("\n The amount cannot be withdrawn ");
   }
}
class Bank
 {
   public static void main (String arr[])
   {
     Scanner ss = new Scanner (System.in);
     Account a1 = new Account();
     a1.setd();
   if (a1.acc (a1.acc_type) == true)
     {
      Sv_acct s1 = new Sv_acct();
          s1.name = a1.name;  s1.acc_no = a1.acc_no;
          s1.acc_type = a1.acc_type ; s1.bal = a1.bal;
        System.out.print(" Enter your choice : \n 1. Deposit \n 2. Calculate
                            3 Withdrawal \n 4. Display \n 5. Exit ");
```

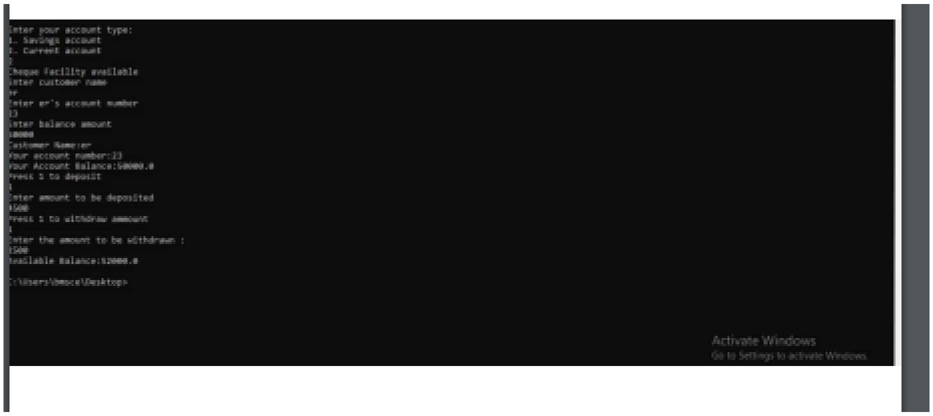
```
  int ch= s.nextInt();
 switch(ch)
 {
   case 1 : s1.deposit(); break;
   case 2 : s1.cal-interest(); break;
   case 3 : s1.witdrawal(); break;
   case 4 : s1.dip(); break;
   case 5 : exit(0); break;
   default : System.out.println("Invalid Input");
 }
}
else
{
  Cur-acct c1 = new Cur-acct();
  c1.name = o1.name; c1.accno = o1.acc_no;
  c1.acc-type = o1.acc-type; c1.bal = o1.bal;
  System.out.println("\n Enter your choice :\n 1.Deposit \n 2.Penalty
     check \n 3.Withdraw \n 4.Display \n 5.Exit");
  int ch = s.nextInt();
   switch(ch)
       {
        case 1 : c1.deposit(); break;
        case 2 : c1.penal(); break;
        case 3 : c1.withdrawal(); break;
        case 4 : exit(0); break;
        default : System.out.println("Invalid Input");
       }
}}
```

```
Enter your account type (1.Savings or 2.Current)
Current
Enter your name
John
Enter your account number
2222
Enter the Bank Balance
50000
Enter your choice
  1.Deposit
  2.Calculate Interest
  3.Withdraw
  4.Display
  5.Exit
1
Enter the amount to be ~~withdrawn~~ deposite
 5000
Account Balance after ~~withdrawals~~ deposits is 55000
Enter your choice
  1.Deposit
  2.Calculate interest
  3.Withdraw
  4.Display
 5.Exit
3
Enter amount to be withdrawn:
 5000
Account Balance after withdrawal is: 50000
```

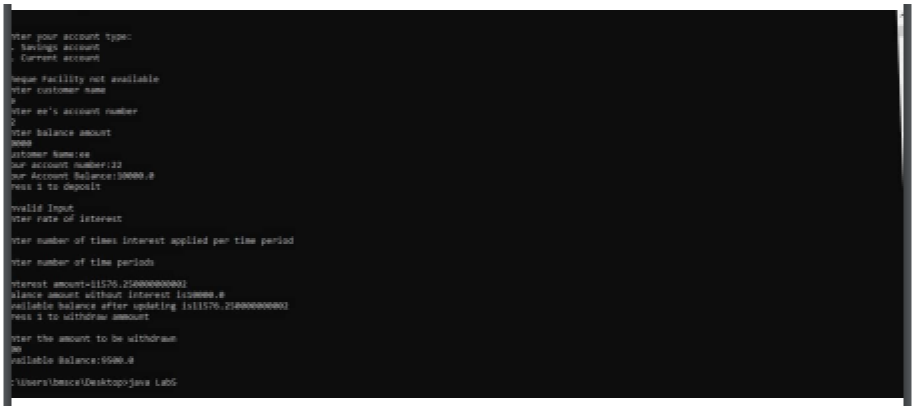


```
Enter your account type (Savings or Current)
Current
Enter your name
JJ
Enter account number
79531
Enter Balance amount
50000
Enter your choice
1. Deposit
2. Penalty
3. Withdraw
4. Display
5. Exit
1
Enter amount to be deposited
3000
Balance = 53000
```

OUTPUT



```
Enter your account type:
1. Savings account
2. Current account
2
Cheque facility available
Enter customer name
er
Enter er's account number
23
Enter balance amount
50000
Customer Name:er
Your account number:23
Your Account Balance:50000.0
Press 1 to deposit
1
Enter amount to be deposited
3500
Press 1 to withdraw ammount
1
Enter the amount to be withdrawn :
1500
Available Balance:52000.0

C:\Users\dmsce\Desktop>
```

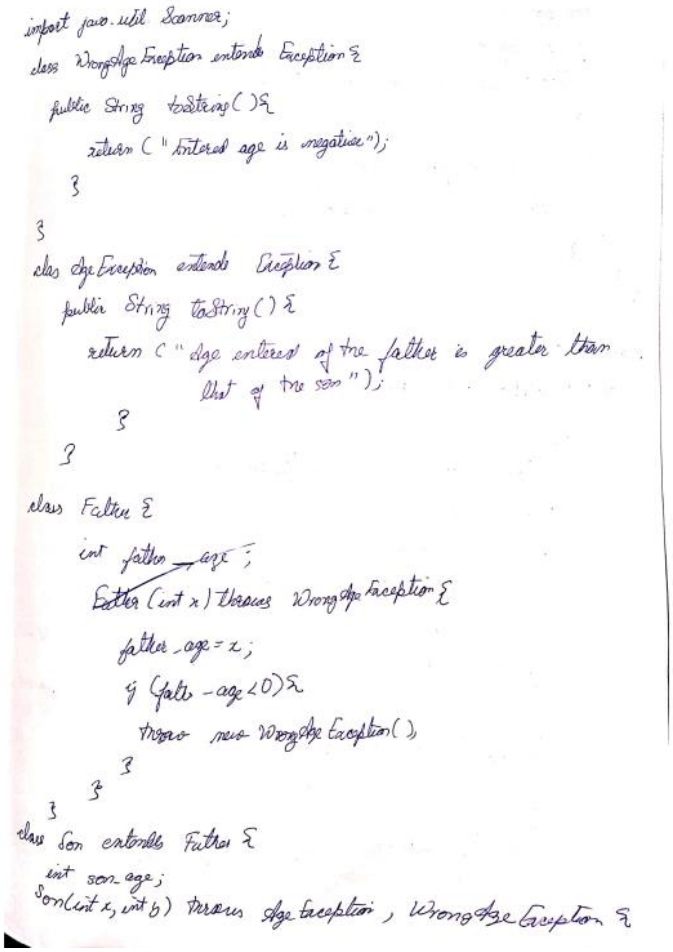


```
nter your account type:
. Savings account
. Current account

heque Facility not available
nter customer name
e
nter ee's account number
2
nter balance amount
0000
ustomer Name:ee
our account number:32
our Account Balance:10000.0
ress 1 to deposit

nvalid Input
nter rate of interest

nter number of times interest applied per time period

nter number of time periods

nterest amount=11576.250000000002
alance amount without interest is10000.0
vailable balance after updating is11576.250000000002
ress 1 to withdraw ammount

nter the amount to be withdrawn
00
vailable Balance:9500.0

:\Users\bmsce\Desktop>java Lab5
```

LAB PROGRAM-6

Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called "Father" and derived class called "Son" which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age<0. In Son class, implement a constructor that cases both father and son's age and throws an exception if son's age is >=father's age.

PROGRAM



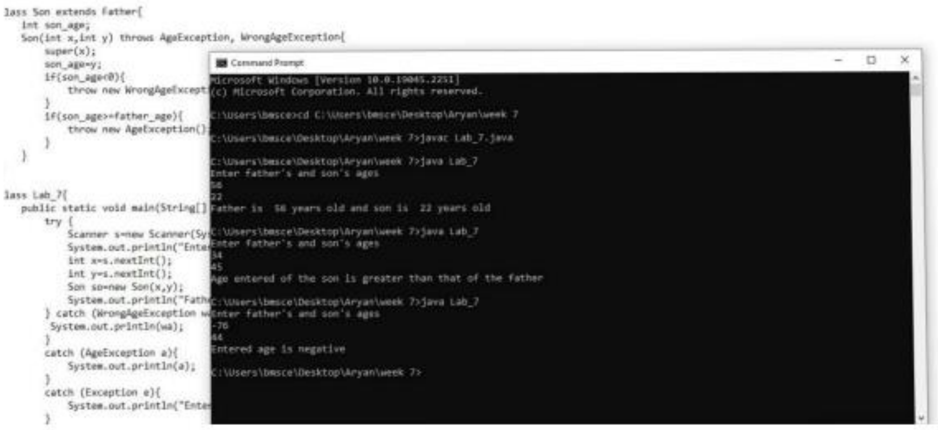
```
import java.util Scanner;
class WrongAgeException entends Exception {
  public String toString( ){
      return ("Entered age is negative");
   }
 }
class AgeException extends Exception {
   public String toString( ) {
      return ( "Age entered of the father is greater than
                  that of the son");
      }
   }
class Father {
     int father_age;
     Father (int x) throws WrongAgeException {
        father_age=x;
        if (fath -age<0){
          throw new WrongAgeException( ),
        }
      }
  }
class Son entends Father {
  int son_age;
Son(int x, int y) throws AgeException , WrongAgeException {
        super(x);
        son_age = y;
        if (son_age < 0) {
         Throw new WrongAgeException();
        }
      if (son_age >= father_age) {
          throw new AgeException();
        }
  }
}
class lab_7 {
  public static void main(String args[]) {
      try {
        Scanner s= new Scanner (System.in);
       System.out.println ("Enter tne age of father and son:");
       int  x= s.nextInt();
       int  y= s.nextInt();
        Son  so = new  Son(x,y);
       System.out.println ("Father is "+ so.father age +" years old and
                                           son is "+ so.sonage +" years old")
       } catch (AgeException a) {
              System.out.println(a);
       }
       catch (Exception e) {
          System.out.println ("Enter valid values");
       }
  }
}
```

Output

```
Enter father's and son's ages
  56
  22
 Father is 56 years old and son is 22 years old

Enter father's and son's ages
  34
  45
Age entered of the son is greater than that of the father

Enter father's and son's ages
  -76
  44
Entered age is negative.
```

30/12/2022

OUTPUT



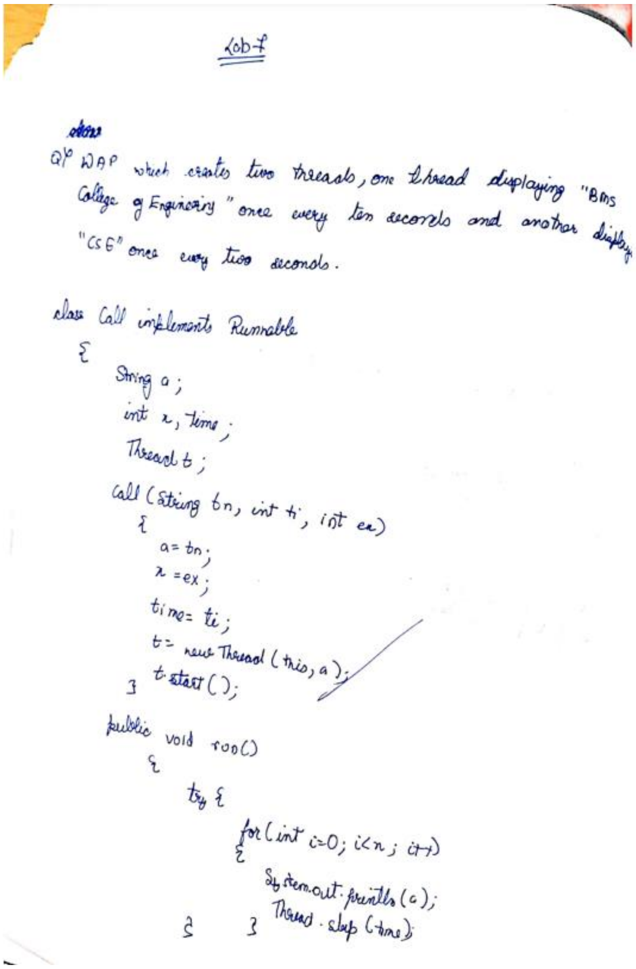
```
lass Son extends Father{
   int son_age;
   Son(int x,int y) throws AgeException, WrongAgeException{
       super(x);
       son_age=y;
       if(son_age<0){
           throw new WrongAgeExcept
       }
       if(son_age>=father_age){
           throw new AgeException();
       }
   }

lass Lab_7{
   public static void main(String[]
       try {
           Scanner s=new Scanner(Sy
           System.out.println("Ente
           int x=s.nextInt();
           int y=s.nextInt();
           Son so=new Son(x,y);
           System.out.println("Fath
       } catch (WrongAgeException w
        System.out.println(wa);
       }
       catch (AgeException a){
           System.out.println(a);
       }
       catch (Exception e){
           System.out.println("Ente
       }
```

```
Microsoft Windows [Version 10.0.19045.2251]
(c) Microsoft Corporation. All rights reserved.

C:\Users\bmsce>cd C:\Users\bmsce\Desktop\Aryan\week 7

C:\Users\bmsce\Desktop\Aryan\week 7>javac Lab_7.java

C:\Users\bmsce\Desktop\Aryan\week 7>java Lab_7
Enter father's and son's ages
56
22
Father is  56 years old and son is  22 years old

C:\Users\bmsce\Desktop\Aryan\week 7>java Lab_7
Enter father's and son's ages
34
45
Age entered of the son is greater than that of the father

C:\Users\bmsce\Desktop\Aryan\week 7>java Lab_7
Enter father's and son's ages
-76
44
Entered age is negative

C:\Users\bmsce\Desktop\Aryan\week 7>
```

LAB PROGRAM-7

Write a program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once every two seconds.

PROGRAM



Lab-7

Q) WAP which creates two threads, one thread displaying "BMS College of Engineering" once every ten seconds and another display "CSE" once every two seconds.

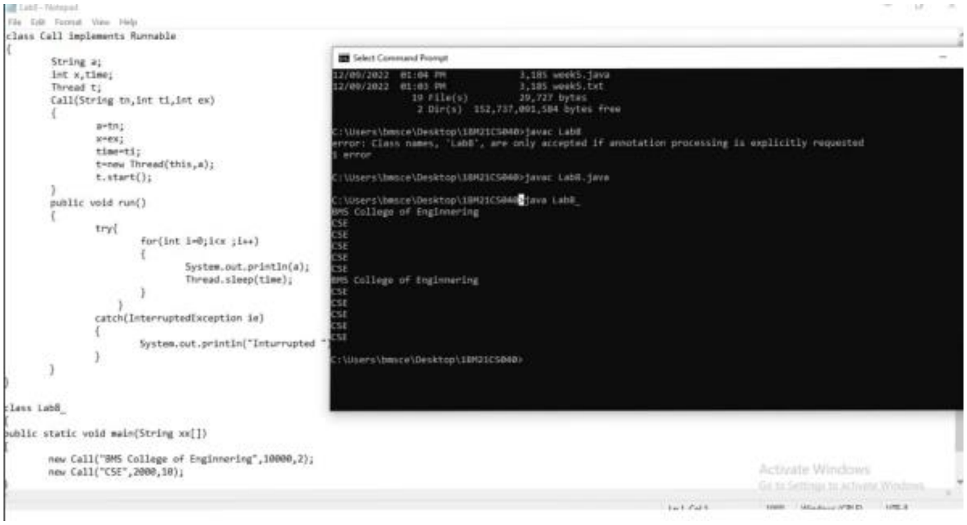
```
class Call implements Runnable
 {
     String a;
     int x, time;
     Thread t;
    Call (String tn, int ti, int ex)
      {
        a= tn;
        x =ex;
        time= ti;
        t= new Thread (this, a);
    }   t.start();
    public void run()
      {
          try {
                for(int i=0; i<n; i++)
                {
                    System.out.println(a);
          }     }   Thread.sleep(time);
catch (InterruptedException ie)
    {
        System.out.println ("Interrupted");
    }
  }
}
class Lab8_
{
    public static void main (String args[])
    {
        new Call ("BMS College of Engineering", 10000 , 2);
        new Call (" CSE ", 2000, 10);
    }
}
```

Output

```
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS college of Engineering
CSE
CSE
CSE
CSE
CSE
```

06/01/2023

OUTPUT


Lab8 - Notepad
File Edit Format View Help
Select Command Prompt
Activate Windows
Go to Settings to activate Windows.

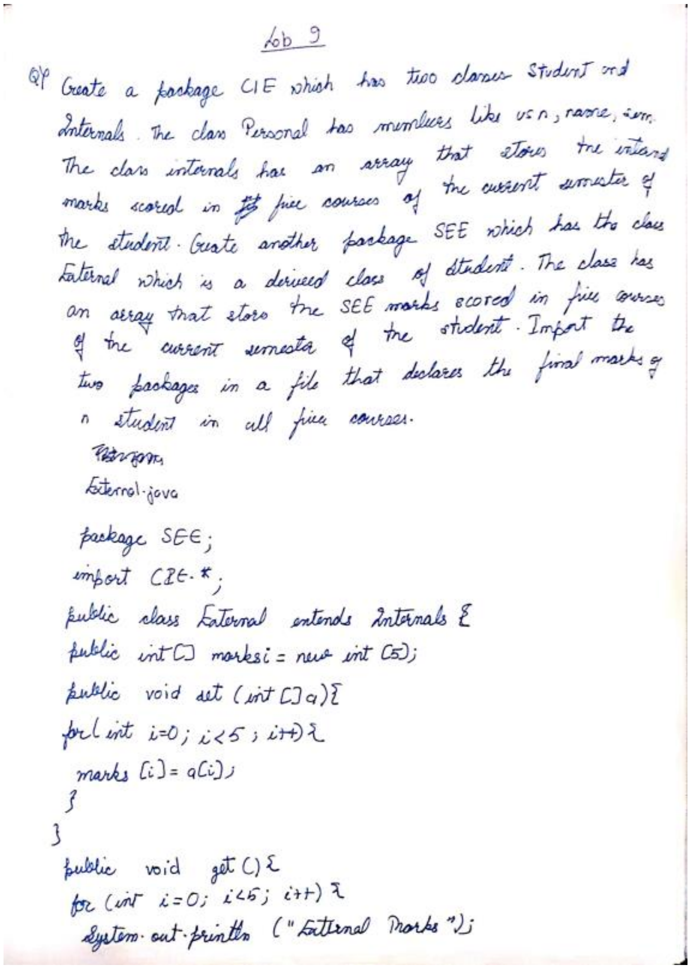
```
class Call implements Runnable
{
        String a;
        int x,time;
        Thread t;
        Call(String tn,int t1,int ex)
        {
                a=tn;
                x=ex;
                time=t1;
                t=new Thread(this,a);
                t.start();
        }
        public void run()
        {
                try{
                        for(int i=0;i<x ;i++)
                        {
                                System.out.println(a);
                                Thread.sleep(time);
                        }
                    }
                catch(InterruptedException ie)
                {
                        System.out.println("Inturrupted "
                }
        }
}

class Lab8_
{
public static void main(String xx[])
{
        new Call("BMS College of Enginnering",10000,2);
        new Call("CSE",2000,10);
}
```

```
12/09/2022  01:04 PM             3,185 week5.java
12/09/2022  01:03 PM             3,185 week5.txt
              19 File(s)         29,727 bytes
               2 Dir(s)  152,737,091,584 bytes free

C:\Users\bmsce\Desktop\1BM21CS040>javac Lab8
error: Class names, 'Lab8', are only accepted if annotation processing is explicitly requested
1 error

C:\Users\bmsce\Desktop\1BM21CS040>javac Lab8.java

C:\Users\bmsce\Desktop\1BM21CS040>java Lab8_
BMS College of Enginnering
CSE
CSE
CSE
CSE
CSE
BMS College of Enginnering
CSE
CSE
CSE
CSE
CSE

C:\Users\bmsce\Desktop\1BM21CS040>
```

LAB PROGRAM 8

Create a package CIE which has two classes- Student and Internals. The class Personal has members like usn, name, sem. The class internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which has the class External which is a derived class of Student. This class has an array that stores the SEE marks scored in five courses of the current semester of the student. Import the two packages in a file that declares the final marks of n students in all five courses.
Develop a Generic Class with Two Type Parameters.

PROGRAM



Lab 9

Q) Create a package CIE which has two classes Student and Internals. The class Personal has members like usn, name, sem. The class internals has an array that stores the internal marks scored in ~~ff~~ five courses of the current semester of the student. Create another package SEE which has the class External which is a derived class of student. The class has an array that stores the SEE marks scored in five courses of the current semester of the student. Import the two packages in a file that declares the final marks of n student in all five courses.

Program

External.java

```
package SEE;
import CIE.*;
public class External extends Internals {
public int[] marksi = new int [5];
public void set (int [] a){
for(int i=0; i<5; i++){
 marks [i] = a[i];
 }
}
public void get (){
 for (int i=0; i<5; i++) {
  System.out.println ("External Marks");
```

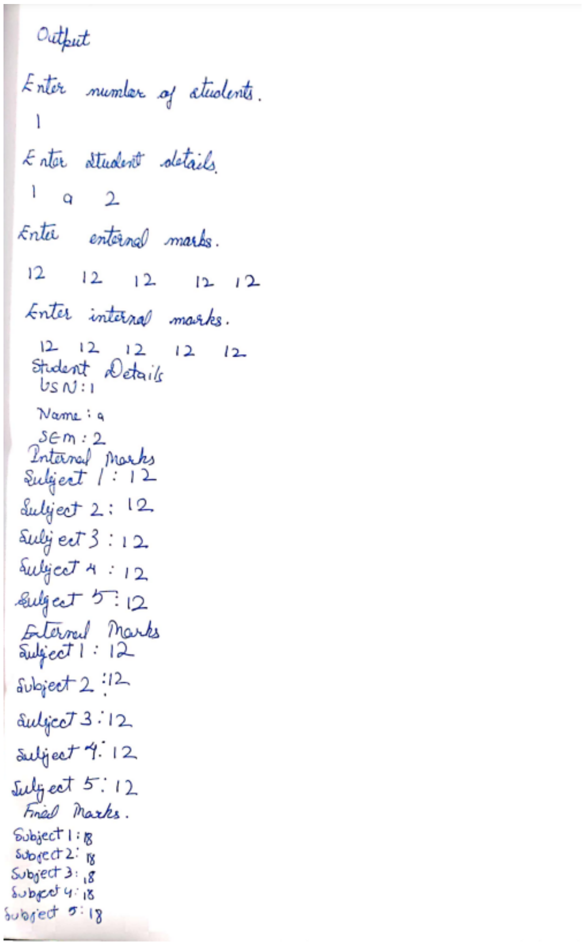
```
   System.out.println ("Subject" +i + ":" + marksi[i]);
   }
  }
 }
```

nternal.java

```
package CIE;
public class Internals extends Student {
public int[] marks = new int[5];
public void setint (int[]a){
for (int i=0; i<5; i++){
marks[i] = a[i];
 }
}
public void getint () {
System.out.println ("Internal marks");
 for(int i=0; i<5; i++){
   System.out.println ("Subject" + i + ":"+ marks[i]);
  }
 }
}
```

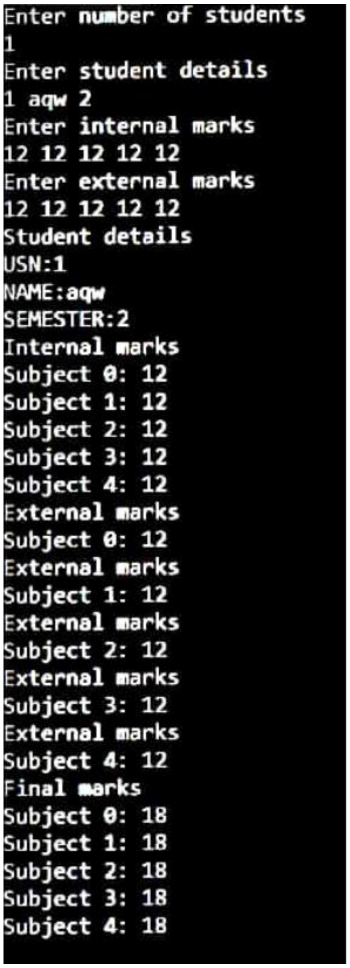
Student.java

```
package CIE;

public class Student {
public int USN;
public String name;
public int sem;

public void setdets (int u, String s ,int q) {
  USN=u;
  name =s;
   sem = q;
}
public void getdets () {
 System.out.println("Student details");
 System.out.println ( "USN:"+USN + "\n NAME:"+name +"\nSEMESTER:"+
                                                       sem);
 }
}
```



Output

```
Enter number of students.
 1
Enter student details.
 1   a   2
Enter external marks.
 12   12   12   12  12
Enter internal marks.
  12  12  12  12  12
 Student Details
  USN:1
  Name : a
  SEM : 2
 Internal Marks
 Subject 1 : 12
 Subject 2 : 12
 Subject 3 : 12
 Subject 4 : 12
 Subject 5 : 12
 External Marks
 Subject 1 : 12
 Subject 2 :12
 Subject 3 :12
 Subject 4: 12
Subject 5: 12
  Final Marks.
 Subject 1:18
 Subject 2: 18
 Subject 3: 18
 Subject 4: 18
Subject 5: 18
```



```
Enter number of students
1
Enter student details
1 aqw 2
Enter internal marks
12 12 12 12 12
Enter external marks
12 12 12 12 12
Student details
USN:1
NAME:aqw
SEMESTER:2
Internal marks
Subject 0: 12
Subject 1: 12
Subject 2: 12
Subject 3: 12
Subject 4: 12
External marks
Subject 0: 12
External marks
Subject 1: 12
External marks
Subject 2: 12
External marks
Subject 3: 12
External marks
Subject 4: 12
Final marks
Subject 0: 18
Subject 1: 18
Subject 2: 18
Subject 3: 18
Subject 4: 18
```