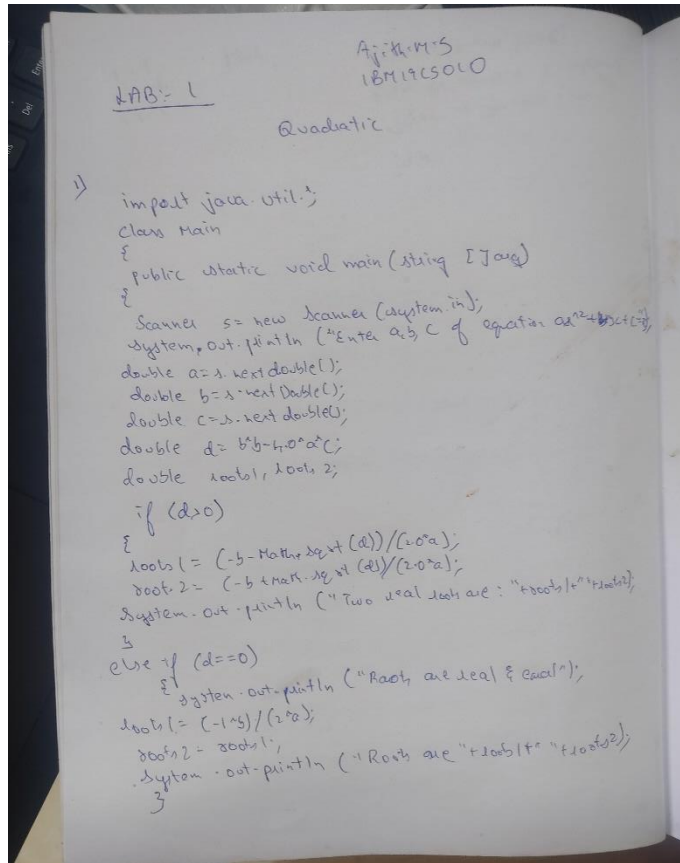


LAB-1

1. Develop a Java program that prints all real solutions to the quadratic equation $ax^2 + bx + c = 0$.

Read in a , b , c and use the quadratic formula. If the discriminate $b^2 - 4ac$ is negative, display a message stating that there are no real solutions.

OBSERVATION:



OUTPUT:

```
Enter a,b,c of the equation ax^2+bx+c=0:
1 2 1
0.0
Roots are real and equal
root1: -1.0root2:-1.0

C:\Users\Hima\Desktop\java>javac Quadratic.java

C:\Users\Hima\Desktop\java>java Quadratic
Enter a,b,c of the equation ax^2+bx+c=0:
1 1 1
-3.0
roots are imaginary

C:\Users\Hima\Desktop\java>javac Quadratic.java

C:\Users\Hima\Desktop\java>java Quadratic
Enter a,b,c of the equation ax^2+bx+c=0:
1 -1 -6
25.0
roots are real and unequal
root1 :-2.0 root2:3.0
```

LAB-2

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.

OBSERVATION:

LAB-2
SGPA

```
import java.util.*;
class Student
{
    private int n, credit [], marks [], gp [],
    private String name, usn;
    double sum, sgpa;

    Student()
    {
        name = " ";
        usn = 0;
    }

    void accept()
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter name of subject:");
        n = sc.nextInt();
        credit = new int[n+1];
        marks = new int[n+1];
        System.out.println("Enter usn & name of student:");
        usn = sc.next();
        name = sc.next();
        System.out.println("Enter credit & marks in each subject");
        for (int i = 1; i <= n; i++)
        {
            System.out.println("Enter subject " + i + " credit & marks");
            credit[i] = sc.nextInt();
            marks[i] = sc.nextInt();
        }
    }
}
```

```
else
{
    System.out.println("There is no real root");
}
}
```

OUTPUT:

```
Enter the usn and name
1bm1234 serena
Enter number of subjects
3
Enter the credits and marks in each subject
Enter the credits and marks in subject 1
3 56
Enter the credits and marks in subject 2
5 79
Enter the credits and marks in subject 3
4 99
usn:1bm1234 name:serena
marks:56 grade points:5
marks:79 grade points:8
marks:99 grade points:10
sgpa:7.916666666666667
```

LAB-3

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.

OBSERVATION:

Week-5

Ajith M.S

```
import java.util.*;
class Book
{
    String name;
    String author;
    double price;
    int num-pages;
    Book()
    {
        name = " ";
        author = " ";
        price = 0.0;
        num-pages = 0;
    }
    void get()
    {
        System.out.println("Enter the details of book(s):");
        Scanner xx = new Scanner(System.in);
        name = xx.next();
        author = xx.next();
        price = xx.nextDouble();
        num-pages = xx.nextInt();
    }
}
```

```

public String toString()
{
    return ("Book: " + name + " Author: " + author + " Price: " + price + " Number of Pages: " + numPages);
}

class BookMain
{
    public static void main(String args[])
    {
        int n;
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number of objects");
        n = sc.nextInt();
        Book xc[] = new Book[n];
        for (int i = 0; i < n; i++)
        {
            xc[i] = new Book();
            xc[i].get();
        }
        for (int i = 0; i < n; i++)
        {
            System.out.println("Details of Book " + (i+1) + ": ");
            System.out.println(xc[i]);
        }
    }
}

```

OUTPUT:

Enter the nubmer of objects:
3

Enter the details of the book

twincke dan 450 1200

Enter the details of the book

Dwen Rossie 345 560

Enter the details of the book

life Ambrose 100 234

details of book1:

Book:twincke
Author:dan
Price:450.0
Number of pages:1200

details of book2:

Book:Dwen
Author:Rossie
Price:345.0
Number of pages:560

details of book3:

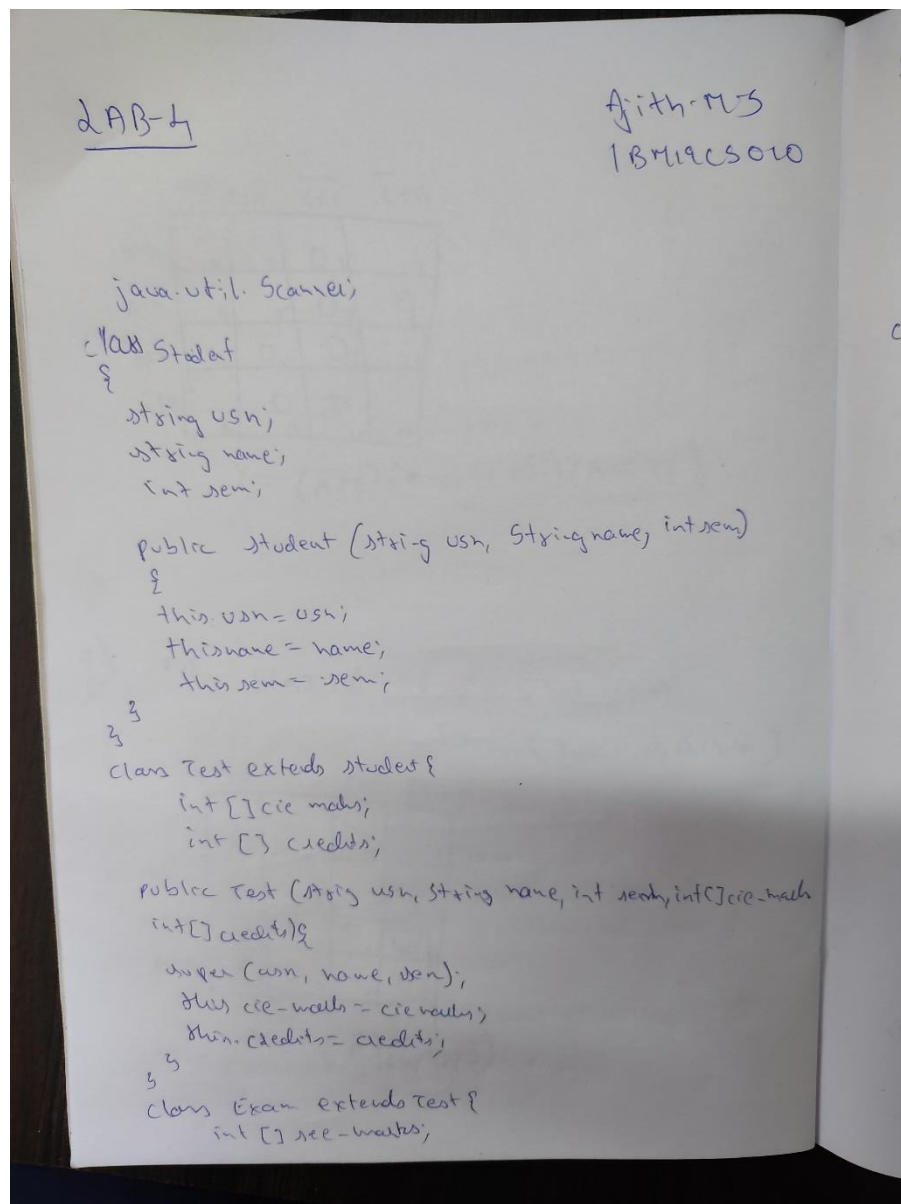
Book:life
Author:Ambrose
Price:100.0
Number of pages:234

LAB-4

4. Develop a Java program to create a class Student whose variables are usn, name and sem.

Derive a class Test from Student to include an array of cie marks of each course and their corresponding credits in another array. Derive a class Exam from Test which includes an array of see marks. Derive a class Result which calculates the grade for each course and the SGPA. Create n student objects and displays all the above details.

OBSERVATION




```

public Exam (String user, String name, int sem, int[] credits,
int[] see marks) {

```

```

    super(user, name, sem, cr-credits, credits);

```

```

    this.see-marks = see-marks;
}

```

```

class results extends Exam {

```

```

    public result (String user, String name, int sem, int[] credits,
int[] see marks)

```

```

    {

```

```


```

```

        double sgpa;

```

```

        char[] grade = new char[credits.length];

```

```

        public void cal-grad = sgpa(int tota-credits);

```

```

        int sum = 0;

```

```

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

```

```

            double marks = cr-credits[i] +

```

```

            (see-marks[i]/2.0);

```

```

            if (marks >= 90) {

```

```

                sum += credits[i] * 10;

```

```

                grade[i] = 'S';

```

```

            } else if (marks >= 80 && marks < 90) {

```

```

                sum += credits[i] * 9;

```

```

                grade[i] = 'A';

```

```

            } else if (marks >= 70 && marks < 80) {

```

```

                sum += credits[i] * 8;

```

```

                grade[i] = 'B';

```

```

            } else if (marks >= 60 && marks < 70) {

```


class Main {

public static void main(String args[])

{

Scanner sc = new Scanner(System.in);

System.out.print("Enter number of students:");

int n = sc.nextInt();

Result[] = new Result[n];

int[] totCredits = new int[n];

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

{

System.out.print("Enter student " + (i + 1) + " details:");

~~@@@~~

int[] credits = new int[n];

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

{

int marks = sc.nextInt();

credits[j] = marks;

}

System.out.print("Enter credits:");

int[] credits = new int[n];

int sum = 0

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

int marks = sc.nextInt();

credits[j] = marks;

sum = credits[j];

}

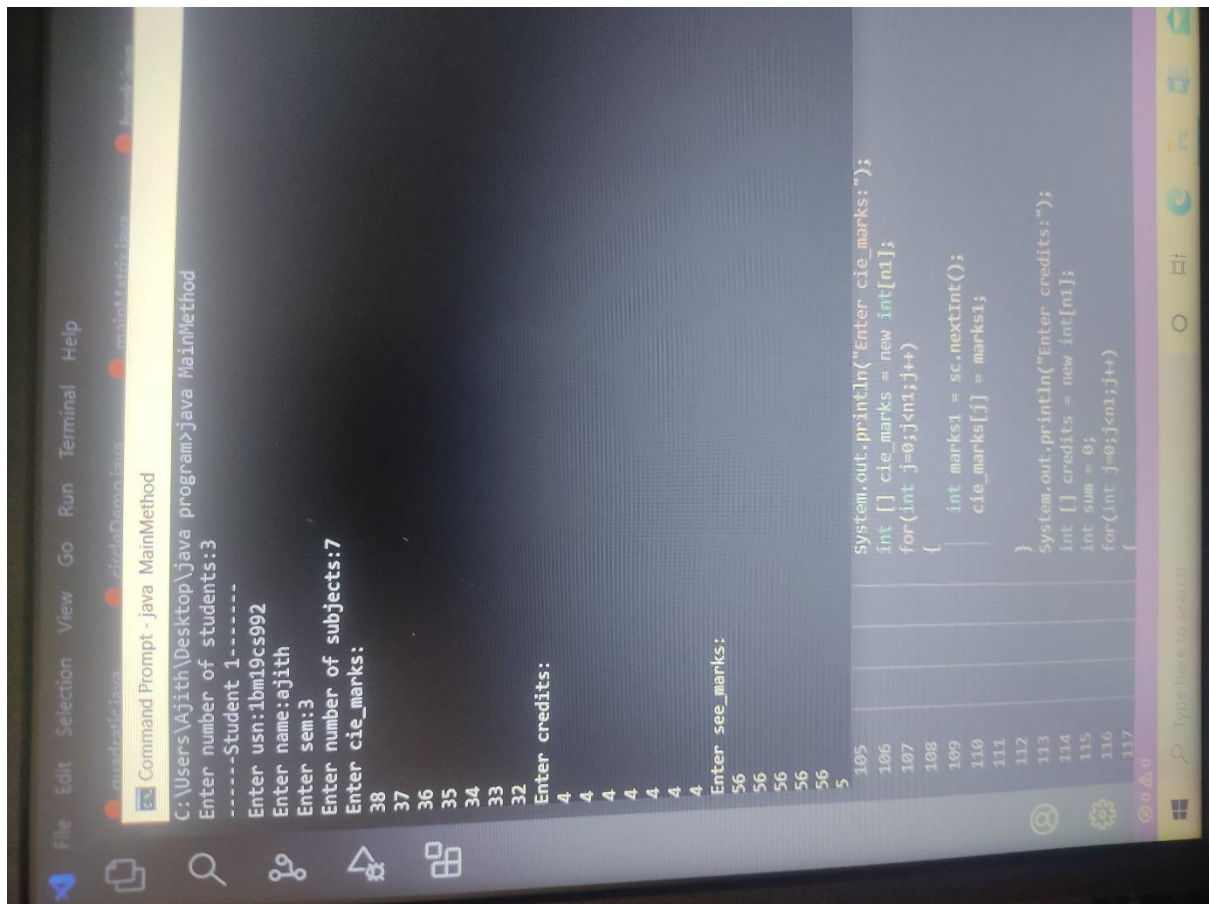
```

totCredits[i] = sum;
system.out.println("Enter see-marks:");
int[] see-marks = new int[n];
for (int j = 0; j < n; j++)
{
    int marks = sc.nextInt();
    see-marks[j] = marks;
}

s[i] = new
Result(cusn, name, sem, credits, credit, see-marks);
}
for (int k = 0; k < n; k++)
{
    system.out.println("Details of student " + (k+1) + ":");
    s[k].cal-grade-sgpa (tot-credits [k]);
    s[k].display();
}
}
}

```

OUTPUT:



```
Command Prompt - java MainMethod
C:\Users\Ajith\Desktop\java program>java MainMethod
Enter number of students:3
-----Student 1-----
Enter usn:1bm19cs992
Enter name:ajith
Enter sem:3
Enter number of subjects:7
Enter cie_marks:
38
37
36
35
34
33
32
Enter credits:
4
4
4
4
4
4
4
Enter see_marks:
56
56
56
56
56
56
5
System.out.println("Enter cie_marks:");
int [] cie_marks = new int[n1];
for(int j=0;j<n1;j++)
{
    int marks1 = sc.nextInt();
    cie_marks[j] = marks1;
}
System.out.println("Enter credits:");
int [] credits = new int[n1];
int sum = 0;
for(int j=0;j<n1;j++)
{
```

LAB-5

5. 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.

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 Curr-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks: • Accept deposit from customer and update the balance. • Display the balance. • Compute and deposit interest • Permit withdrawal and update the balance • Check for the minimum balance, impose penalty if necessary and update the balance

OBSERVATION:

LAB 5:-

Q5

abstract class shape

int x, y;

abstract void printArea();
}

class rectangle extends shape

{
rectangle (inta, intb)

{
x = a;
y = b;
}

void printArea()

{
system.out.println ("Area of rectangle is " + (x*y));
}

class triangle extends shape

{
triangle (inta, intb)

{
x = a;
y = b;
}

void printArea()

{
system.out.println ("Area of triangle is " + (x*y));
}

class circle extends shape

```
{  
    circle (inta) {  
        x=a; }  
}
```

```
void printArea() {  
    System.out.println("Area of circle is "+(3.142*x*x));  
}
```

public class main

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

```
{  
    rectangle rect = new rectangle(5,5);  
    triangle tri = new triangle(6,3);  
    circle cir = new circle(5);
```

```
    rect.printArea();
```

```
    tri.printArea();
```

```
    cir.printArea();
```

```
    }  
}
```

area();

LAB:- 5

Ajith M S
18M19CS010

```
java.util.Scanner;  
class Account  
{  
    String name, account, acctype;  
    double balance;  
    Account() {}  
    void input()  
    {  
        Scanner sc = new Scanner(System.in);  
        System.out.println("Enter name, ac number");  
        name = sc.next();  
        account = sc.next();  
        System.out.println("Enter balance");  
        balance = sc.nextDouble();  
    }  
    void displayDetails()  
    {  
        System.out.println("name = " + name + " number = " + account);  
        balance = " + balance + " account type = " + acctype;  
    }  
    void displayBalance()  
    {  
        System.out.println("balance = " + balance);  
    }  
}
```

LAB-2

```
class curacct extends Account
```

```
{  
    curacct()
```

```
{
```

```
    acctype = "current";
```

```
    double minbal = 5000;
```

```
void check()
```

```
{  
    double penalty = 100;
```

```
    if (balance < minbal)
```

```
{
```

```
        balance = balance - penalty;
```

```
        System.out.println("penalty is imposed");
```

```
        System.out.println("balance = " + balance);
```

```
    }
```

```
    else
```

```
{
```

```
        System.out.println("penalty not imposed");
```

```
    }
```

```
void deposit()
```

```
{  
    Scanner sc = new Scanner(System.in);
```

```
    System.out.println("Enter amount to deposit");
```

```
    double amt = sc.nextDouble();
```

```
    balance = balance + amt;
```

```
}
```

```
}
```

```
class savacc extends Account
```

```
{
```

```
    savacc()
```

```
{  
    acctype = "savings";
```

```

double ci;
void calcompound(int n, int t)
{
    ci = balance * (Math.pow((1 + (0.2/n)), t));
    balance = balance + ci;
    System.out.println("Compound Interest: %.2f", ci);
    System.out.printf("Balance: %.2f, balance);
}

```

```

void withdrawal(double amt)
{
    double min bal = 5000;
    if (balance < 5000)
        System.out.println("Amount can't be withdrawn as  
min balance (5000)  
constraint will be violated");
    else
        balance = balance - amt;
}

```

```

class account main
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter 1 for savings 2 for current account");
        int acctype = sc.nextInt();
        savacc s = new savacc();
        curacc c = new curacc();
        if (acctype == 1)
    }
}

```

```

{
    system.out.println("Enter your details:");
    u.input();
    S.displaydetail();
    system.out.println("no. of time it has to be
    compounded over t(u), for n years");

    int n = sc.nextInt();
    int t = sc.nextInt();
    S.calcompd(u, t);

    int n1 = 1;
    while(n1 != 1)
    {
        system.out.println("Enter 1. deposit 2. withdraw 3. exit");
        int w = sc.nextInt();
        if (w == 1)
        {
            S.deposit();
        }
        else if (w == 2)
        {
            system.out.println("enter the amount:");
            double amt = sc.nextDouble();
            S.withdrawal(amt);
            S.display();
        }
        else {
            system.exit(0);
        }
    }
}

```

```

else if (accType == 2)
{
    system.out.println("Enter your details");
    C.input();
    C.displayDetails();
    C.check();
    C.deposit();
    C.withdraw();
}
}
}

```

OUTPUT:

