

Name: Harshita M Jaka

USN: 1BM19CS058

Section: 3B

OOJ LAB OBSERVATION

LAB PROGRAM 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 discriminant $b^2 - 4ac$ is negative, display a message stating that there are no real solutions.

store
67

Week 3 Lab program 1

Lab program

```
import java.util.Scanner;
class equation
{
    public static void main(String args[])
    {
        double x1=0, x2=0;
        Scanner root = new Scanner(System.in);
        System.out.print("Enter the value of ");
        System.out.print("a in ax^2+bx+c=0:");
        double a = root.nextDouble();
        System.out.print("Enter the value of ");
        System.out.print("b in ax^2+bx+c=0:");
        double b = root.nextDouble();
        System.out.print("Enter the value of ");
        System.out.print("c in ax^2+bx+c=0:");
        double C = root.nextDouble();
        double D = 2*a;
        double D = (b*b) - 4*a*c;
        if (D>0)
        {
            System.out.println("Solutions real and distinct");
        }
    }
}
```

$x_1 = ((-b + \sqrt{D})/n);$
 $x_2 = ((-b - \sqrt{D})/n);$
System.out.println("solutions are");
System.out.println(x_1);
System.out.println(x_2);

{

else if ($D == 0$)

{

System.out.println("solution real and equal");
 $x_1 = x_2 = -b/n.$

System.out.println("solutions are");

System.out.println(x_1);

System.out.println(x_2);

{

else

{

System.out.println("No real solutions");

{

{

{}



A screenshot of a terminal window from a code editor. The tabs at the top are PROBLEMS (66), OUTPUT, DEBUG CONSOLE, and TERMINAL (underlined). The terminal output shows the following:

```
PS C:\Users\Harshita\Desktop\java> javac quad.java
> java equation
Enter the value of b of ax^2+bx+c=0:2
Enter the value of c ax^2+bx+c=0:1
solution real and equal
solutions are
-1.0
-1.0
PS C:\Users\Harshita\Desktop\java> java equation
=0:2
Enter the value of b of ax^2+bx+c=0:3
Enter the value of c ax^2+bx+c=0:4
NO real solutions
PS C:\Users\Harshita\Desktop\java> []
```

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.

Week 4

Lab program 2

```
import java.util.Scanner;  
class Student:  
{  
    private String usn;  
    private String sname;  
    private double sgpa;  
    private int []c;  
    private int []marks;  
    private int size=5;
```

```
void acceptDetails()  
{
```

```
Scanner s1 = new Scanner (System.in);  
System.out.println ("Enter the USN");  
usn = s1.next();  
System.out.println ("Enter the name");  
sname = s1.next();  
c = new int [size];  
marks = new int [size];  
for (int i=0; i<size; i++)
```

```
System.out.println ("Enter the credits  
and marks of subject "+i);
```

store
67

$c[i] = s1.nextInt();$

$\text{marks}[i] = s1.nextInt();$

}

}

void calculate()

{

int num=0, deno=0, gradeP=0;

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

{

if ($\text{marks}[i] \geq 90$)

gradeP = 10;

else if ($\text{marks}[i] \geq 80$)

gradeP = 9;

else if ($\text{marks}[i] \geq 70$)

gradeP = 8;

else if ($\text{marks}[i] \geq 60$)

gradeP = 7;

else if ($\text{marks}[i] \geq 50$)

gradeP = 5;

else if ($\text{marks}[i] \geq 40$)

gradeP = 4;

else if ($\text{marks}[i] \geq 40$)

gradeP = 0;

num += gradeP * c[i];

*

deno += CL[i];

}

Sgpa = num/deno;

System.out.println("Student
sgpa is :" + sgpa);

}

void printDetails()

{

System.out.println("Student Details");

System.out.println("Student USN:" + usn);

System.out.println("Student Name:" + sname);

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

System.out.println("Credit is :" + (CL[i]));

System.out.println("Marks of
subject" + i + " " + marks[i]);

}
}

Class Student Main

{

store
67

public static void main (String args[])

{
 Student s1 = new Student();
 s1.acceptDetails();
 s1.printDetails();
 s1.calculate();

}

}

ALGORITHM

Step 1: Create a ~~class~~ with name Student under class containing : usn, sname, sgpa , array of credits and marks .

Step 2: A function acceptDetails , accepts the usn, name , credits and marks of a student.

Step 3: A function calculate , takes gradepoints according to the marks by if else if statements .

Then $Sgpa = \text{num} / \text{den}$.

The screenshot shows a Windows desktop environment with a Java application running in a terminal window. The terminal window is titled 'TERMINAL' and displays the following output:

```
PS C:\Users\Harshtita\Desktop\java> javac week4.java
PS C:\Users\Harshtita\Desktop\java> java StudentMain
Enter the USN
IBH19C5058
Enter the name
Harshtita
enter the credits and marks of subject0
2 40
enter the credits and marks of subject1
3 39
enter the credits and marks of subject2
3 44
enter the credits and marks of subject3
4 38
enter the credits and marks of subject4
2 40
**Student Details**
Student USN:IBH19C5058
Student name:Harshtita
Credit is:2
Marks of subject0 40
Credit is:3
Marks of subject1 39
Credit is:3
Marks of subject2 44
Credit is:2
Marks of subject3 38
Credit is:2
Marks of subject4 46
Student sgra is:4.0
PS C:\Users\Harshtita\Desktop\java>
```

A small modal dialog box is visible in the center of the screen, asking 'Build failed, do you want to continue?'. It has three buttons: 'Proceed', 'Fix...', and 'Cancel'. The desktop taskbar at the bottom shows various icons for common applications like File Explorer, Edge, Mail, and Task View.

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 a object book.

store
67

Week 5: Lab program 3

```
import java.util.*;  
import java.util.*;  
class Book  
{  
    String name;  
    String author;  
    double price;  
    int num_pages;  
    public void getdetails()  
    {
```

```
        Scanner s = new Scanner (System.in);  
        System.out.println ("Enter Book Details");  
        System.out.print ("Enter Book Name: ");  
        name = s.next();  
        System.out.print ("Enter the Author");  
        author = s.next();  
        System.out.print ("Enter the Price of the  
book");  
        price = s.nextDouble();  
        System.out.print ("Enter the number of  
pages in the book");
```

```
    num_pages = s.nextInt();
```

}

```
public String toString()
```

{

```
    return "*** BOOK DETAILS ***" + "\n"
```

The name of the book : " + name + "\n" The
author of the book : " + author + "\n" The
price of the book : " + price + "\n" Number
of pages in book : " + num_pages);

}

}

Class Book Main

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

{

```
    int i, n;
```

```
    Scanner s = new Scanner(System.in);
```

System.out.print("Enter the number
of books");

```
    n = s.nextInt();
```

```
    Book b[] = new Book[n];
```

store
67

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

```
{
```

```
    b[i] = new Book();
```

```
    b[i] = getdetails();
```

```
}
```

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

```
{
```

```
    System.out.println(b[i]);
```

```
}
```

```
}
```

```
.
```

The screenshot shows a dark-themed terminal window within a code editor interface. The terminal bar at the top has tabs for PROBLEMS (27), OUTPUT, DEBUG CONSOLE, and TERMINAL, with TERMINAL being the active tab. The left sidebar contains icons for file operations like Open, Save, Find, Copy/Paste, and others. The main terminal area displays the following Java application output:

```
PS C:\Users\Harshita\Desktop> java week_4_Lab_book.java
PS C:\Users\Harshita\Desktop> java BookMain
Enter the number of books
2
Enter Book Details
Enter Book Name:
xyz
Enter the Author
abc
Enter the Price of the book
140
Enter the number of pages in the book
150
Enter Book Details
Enter Book Name:
lmn
Enter the Author
rst
Enter the Price of the book
400
Enter the number of pages in the book
300
*BOOK DETAILS*
The name of the book : xyz
The author of the book :abc
The price of the book: 140.0
Number of pages in book :150
*BOOK DETAILS*
The name of the book : lmn
The author of the book :rst
The price of the book: 400.0
Number of pages in book :300
PS C:\Users\Harshita\Desktop>
```

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.

store
67

Week 8

Lab program. 4

```
import java.util.Scanner;  
abstract class Shape
```

{

```
    int d1;
```

```
    int d2;
```

```
    Shape (int a, int b)
```

{

```
    d1 = a;
```

```
    d2 = b;
```

}

```
    abstract void printArea();
```

{

```
    class Rectangle extends Shape
```

{

```
    Rectangle (int a, int b)
```

{

```
    super(a, b);
```

}

```
    void printArea()
```

{

float area = (float)d1 * d2;
System.out.println ("Area of the rectangle:
+ area);

class Triangle extends Shape

{ Triangle (int a, int b)

{ super(a, b);

} void printarea()

{ float area = (float)d1 * d2 / 2;

System.out.println ("Area of the triangle:
+ area);

class Circle extends Shape

{ Circle (int a, int b)

{ super(a, b);

store
67

Void printarea()

{

float area=(float) 3.14 * d1 * d1;
System.out.println("Area of the circle:
+ area);

}

}

class Main

{

public static void main(String args[])

{

int ch, flag=0;

Scanner ss = new Scanner(System.in);

while(flag == 0)

{

System.out.println("Enter the choice
whose area has to be calculated");

System.out.println("1. RECTANGLE\n2. TRIANGLE\n3. CIRCLE");

ch = ss.nextInt();

Switch (ch)

{

case 1 :

System.out.println("Enter the
dimensions of rectangle");

int x = ss.nextInt();

int y = ss.nextInt()

Rectangle r = new Rectangle(x, y);

r.printarea();

break;

(case 2):

System.out.println("Enter the dimensions of triangle");

int s = ss.nextInt();

int w = ss.nextInt();

Triangle t = new Triangle(s, w);

t.printarea();

break;

(case 3):

System.out.println("Enter the radius of circle");

int f = ss.nextInt();

Circle c = new Circle(f, b);

c.printarea();

break;

default;

flag=1;

3
3
3
3

PROBLEMS 46 OUTPUT DEBUG CONSOLE TERMINAL

Try the new cross-platform PowerShell <https://aka.ms/pscore6>

```
PS C:\Users\Harshita\Desktop\java> javac abstract.java
PS C:\Users\Harshita\Desktop\java> java Main
Enter the choice whose area has to be calculated
1.RECTANGLE
2.TRIANGLE
3.CIRCLE
1
Enter the dimensions of rectangle
30 45
Area of the rectangle :1350.0
Enter the choice whose area has to be calculated
1.RECTANGLE
2.TRIANGLE
3.CIRCLE
2
Enter the dimensions of triangle
44 55
Area of the triangle :1210.0
Enter the choice whose area has to be calculated
1.RECTANGLE
2.TRIANGLE
3.CIRCLE
3
Enter the radius of circle
7
Area of the circle :153.86002
Enter the choice whose area has to be calculated
1.RECTANGLE
2.TRIANGLE
3.CIRCLE
6
PS C:\Users\Harshita\Desktop\java>
```

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 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.

Lab program 5

```
import java.util.Scanner;  
abstract class Account  
{  
    String cust_name;  
    long acc_no;  
    String acc_type;  
    double balance;  
    double min_bal = 1000.0;  
    Account (String cust_name, long acc_no,  
             String acc_type, double balance)  
    {  
        this.cust_name = cust_name;  
        this.acc_no = acc_no;  
        this.acc_type = acc_type;  
        this.balance = balance;  
    }
```

```
    abstract void deposit (double amount);  
    abstract void display ();  
    abstract void withdrawal (double amount);  
}
```

class Curr_acct extends Account

{

 double penalty = 100.0;

 Curr_acct (String cust_name, long acc_no,
 String acc_type, double balance)

{

 super (cust_name, acc_no, acc_type,
 balance);

 System.out.println ("Name of the customer:
 + cust_name);

 System.out.println ("Account Number acc_no:
 + acc_no);

 System.out.println ("Account type: " +
 acc_type);

 System.out.println ("Balance: " + balance);

}

 void deposit (double amount)

{

 this.balance = this.balance + amount;

}

 void withdrawal (double amount)

{

store
67

this.balance = this.balance - amount;
imposepenalty();

System.out.println("The current balance
is " + balance);

}
void imposepenalty()

{ if (this.balance < min_bal)

}
this.balance = this.balance - penalty;
System.out.println("The balance amount
is insufficient, the penalty
imposed = 100Rs.");

}
void display()

System.out.println("Balance is: " + this.
balance);

class Sav-act extends Account

10 x 1000

Sav acct (String cust_name, long acc_no,
String acc_type, double balance)

{
super (cust_name, acc_no, acc_type, balance);
System.out.println ("Name of the customer :"
+ cust_name);
System.out.println ("Account Number accno :"
+ acc_no);

System.out.println ("Account type : " + acc_type);
System.out.println ("Balance : " + balance);
}

Void desposite (double amount)

{
this.balance = this.balance + amount;
interest();

}
void interest()

{
int rate = 10, time = 1;
float ci = (float)(this.balance * Math.
pow (1 + rate / 100.0, time)
- this.balance);

System.out.println ("The interest amount
added to balance is " + ci);

this.balance = this.balance + (i);

}

void withdrawal (double amount)

{

this.balance = this.balance - amount;
System.out.println ("The current balance
is " + balance);

}

void display ()

{

System.out.println ("Balance is : " +
this.balance);

}

}

class AccountMain

{

public static void main (String [] args)

{

Scanner xx = new Scanner (System.in);

Double amount;

```
int flag=0;  
while (flag == 0)
```

{

System.out.println("Enter the type of
account: 1: current account
2: savings account");

```
int choice = xx.nextInt();  
switch (choice)
```

{

case 1: System.out.println ("Current
account")

System.out.println ("Enter the name
of account holder");

```
String f = xx.next();
```

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

```
long g = xx.nextLong();
```

System.out.println ("Enter the
balance amount");

```
double h = xx.nextDouble();
```

Curr-acct c = new Curr-acct(f, g,
"Current", h);

```
int flag1 = 0;
```

```
while (flag1 == 0)
```

store
67

{ System.out.println ("Enter your choice\n 1:
Deposit amount\n 2: Display Balance\n 3: Withdraw");

int choice = xx.nextInt();

switch (choice) {

{

case 1:

System.out.println ("Enter amount to be
deposited:");

amount = xx.nextDouble();

c.deposit(amount);

break;

case 2:

c.display();

break;

case 3:

System.out.println ("Enter amount you
want to withdraw:");

amount = xx.nextDouble();

c.withdrawal(amount);

break;

default;
flag1 = 1;

}
}

break;

case 2 : System.out.println("\n Savings account");
System.out.println("Enter the name of
account holder").

String p = xx.next();

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

long q = xx.nextLong();

System.out.println("Enter the balance
amount");

double r = xx.nextDouble();

Sav-acct s = new Sav-acct(p, q, "Savings");

int flag2 = 0;

while (flag2 == 0)

{

System.out.println("Enter your choice")
1: Deposit amount \n 2: Display Balance
\n 3: Withdraw"),

int choice2 = xx.nextInt();

Switch(choice2)

store
67

{
case 1: System.out.println("Enter
amount to be deposited:");
amount = xx.nextDouble();
s.deposite(amount);
break;

case 2:
s.display();
break;

case 3:

System.out.println("Enter amount you
want to withdraw:");
amount = xx.nextDouble();
s.withdraw(amount);
break;
default:

flag ≥ 1 ;

3

3

break;

default: flag ≥ 1 ;

3

3

3

```
PS C:\Users\Harshita\Desktop> java Bank.java
PS C:\Users\Harshita\Desktop> java AccountMain
Enter the type of Account:
1:Current account
2:Savings account
1

Current account:

Enter the name of account holder
ABC
Enter the account number
1234567
Enter the balance amount
200000
Name of the customer: ABC
Account Number accno: 1234567
Account type: current
Balance: 200000.0
Enter your choice
1:Deposit amount
2:DisplayBalance
3:Withdraw
1
Enter amount to be deposited:
500
Enter your choice
1:Deposit amount
2:DisplayBalance
3:Withdraw
3
Enter amount you want to withdraw:
200
The current balance is 200300.0
Enter your choice
1:Deposit amount
2:DisplayBalance
3:Withdraw
2
Balance is: 200300.0
Enter your choice
1:Deposit amount
2:DisplayBalance
3:Withdraw
7
Enter the type of Account:
1:Current account
2:Savings account
1

Current account:
```

```
1
Current account:

Enter the name of account holder
JKL
Enter the account number
745678
Enter the balance amount
800
Name of the customer: JKL
Account Number accno: 745678
Account type: current
Balance: 800.0
Enter your choice
1:Deposit amount
2:DisplayBalance
3:Withdraw
3
Enter amount you want to withdraw:
100
The balance amount is insufficient, the penalty imposted = 100Rs
The current balance is 600.0
Enter your choice
1:Deposit amount
2:DisplayBalance
3:Withdraw
7
Enter the type of Account:
1:Current account
2:Savings account
2

Savings account:

Enter the name of account holder
XYZ
Enter the account number
987654
Enter the balance amount
700000
Name of the customer: XYZ
Account Number accno: 987654
Account type: Savings
Balance: 700000.0
Enter your choice
1:Deposit amount
2:DisplayBalance
3:Withdraw
1
Enter amount to be deposited:
```

```
1:Deposit amount
2:DisplayBalance
3:Withdraw
7
Enter the type of Account:
1:Current account
2:Savings account
2

Savings account:
Enter the name of account holder
XYZ
Enter the account number
987654
Enter the balance amount
700000
Name of the customer: XYZ
Account Number accno: 987654
Account type: Savings
Balance: 700000.0
Enter your choice
1:Deposit amount
2:DisplayBalance
3:Withdraw
1
Enter amount to be deposited:
300
The interest amount added to balance is 70030.0
Enter your choice
1:Deposit amount
2:DisplayBalance
3:Withdraw
2
Balance is: 770330.0
Enter your choice
1:Deposit amount
2:DisplayBalance
3:Withdraw
3
Enter amount you want to withdraw:
400
The current balance is 769930.0
Enter your choice
1:Deposit amount
2:DisplayBalance
3:Withdraw
```

LAB PROGRAM 6:

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.

Week 9

Lab Program 6

- ⇒ Explorer
- ⇒ Student.java

```
package CIE;
import java.util.Scanner;

public class Student
{
    String name, usn;
    int sem;
    Scanner xx = new Scanner (System.in);
    public void accept()
    {
        System.out.println ("Enter name:");
        name = xx.nextLine();
        System.out.println ("Enter usn:");
        usn = xx.next();
        System.out.println ("Enter sem:");
        sem = xx.nextInt();
    }
}
```

public void display()

{
 System.out.println("Name:" + name);
 System.out.println("USN:" + usn);
 System.out.println("Sem:" + sem);
}

⇒ Internals.java

```
package CIE;  
import java.util.Scanner;  
public class Internals extends CIE.Student  
{  
    public int ciem[] = new int[5];  
    Scanner xx = new Scanner(System.in);  
    public void accept()  
    {  
        for (int i = 0; i < 5; i++)  
        {  
            System.out.println("Enter the CIE marks  
of subject " + (i + 1));  
            ciem[i] = xx.nextInt();  
        }  
    }  
}
```

⇒ Externals.java

```
package SEE;  
import java.io.*;  
import java.util.Scanner;
```

```
public class Externals extends Student
```

```
{  
    public int sem[] = new int[5];  
    Scanner xx = new Scanner(System.in);
```

```
    public void accept()
```

```
{  
    for (int i = 0; i < 5; i++)
```

```
        System.out.print("Enter the marks of subject " + (i + 1));  
    }
```

```
    sem[i] = xx.nextInt();
```

 } } }

⇒ TotalMarks.java

store
67

public static void main (String sss[])

{ int i, j, n;

int total [] = new int [5];

Scanner xx = new Scanner (System.in);

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

n = xx.nextInt();

CIE.Student s [] = new CIE.Student[n];

CIE.Internal i [] = new CIE.Internal[n];

SEE.External se [] = new SEE.External[n];

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

{

System.out.println ("ENTER STUDENT"
+ (i+1) + " DETAILS");

s[i] = new CIE.Student();

s[i].accept();

c[i] = new CIE.Internal();

c[i].accept();

se[i] = new SEE.External();

se[i].accept();

}

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

{

System.out.println ("DETAILS OF
STUDENT" + (i+1));

```
public static void main (String sss[])
{
    int i, j, n;
    int total [] = new int [5];
    Scanner xx = new Scanner (System.in);
    System.out.println ("Enter the number
                        of students");
    n = xx.nextInt();
    CIE.Student s [] = new CIE.Student[n];
    CIE.Internals ci [] = new CIE.Internals[n];
    SEE.Externals se [] = new SEE.Externals[n];
    for (i=0; i<n; i++)
    {
        System.out.print ("ENTER STUDENT"
                         + (i+1) + " DETAILS");
        s[i] = new CIE.Student();
        s[i].accept();
        ci[i] = new CIE.Internals();
        ci[i].accept();
        se[i] = new SEE.Externals();
        se[i].accept();
    }
    for (i=0; i<n; i++)
    {
        System.out.println ("DETAILS OF
                            STUDENT" + (i+1));
    }
}
```

```
s[i].display();
for (j=0; j<5; j++)
{
    total[j] = c[i][j], itemP[j] + (se[i],
                                    sum[i][j]);
    System.out.println ("Total marks in
                        subject " + (j+1) + " is " + total[j]);
```



Procedure to execute this program :-

1. Created two folders ~~at~~ name and named one folder as CIE and other as SEE.
2. Created one Super class file as Student.java and two sub class files (Internals.java and Externals.java) and one driver class named TotalMarks.java.
3. First, compiled Student.java and placed it in CIE folder. Student.class file in CIE folder.
4. Second, compiled Internals.java and placed the Internals.class again in CIE folder.
5. Third, compiled Externals.java and placed Externals.class file inside SEE folder.
6. Last, Compiled TotalMarks.java (~~main~~ driver class) and then executed the program.

```
C:\Users\akki\Desktop\java files\package>java TotalMarks
Enter the number of students
1
ENTER STUDENT1 DETAILS
Enter name:
Rashmi Singh

C:\Users\akki\Desktop\java files\package>javac Student.java

C:\Users\akki\Desktop\java files\package>javac Internals.java

C:\Users\akki\Desktop\java files\package>javac Externals.java

C:\Users\akki\Desktop\java files\package>javac TotalMarks.java

C:\Users\akki\Desktop\java files\package>java TotalMarks
Enter the number of students
1
ENTER STUDENT1 DETAILS
Enter name:
Rashmi Singh
Enter usn:
1BM19CS293
Enter sem:
2
Enter the cie marks of subject1 out of 50
34
Enter the cie marks of subject2 out of 50
45
Enter the cie marks of subject3 out of 50
42
Enter the cie marks of subject4 out of 50
38
Enter the cie marks of subject5 out of 50
39
Enter the see marks of subject1 out of 100
78
Enter the see marks of subject2 out of 100
76
Enter the see marks of subject3 out of 100
88
Enter the see marks of subject4 out of 100
80
```

```
C:\Users\akki\Desktop\java files\package>javac Externals.java
C:\Users\akki\Desktop\java files\package>javac TotalMarks.java
C:\Users\akki\Desktop\java files\package>java TotalMarks
Enter the number of students
1
ENTER STUDENT1 DETAILS
Enter name:
Rashmi Simgh
Enter usn:
1BM19CS293
Enter sem:
2
Enter the cie marks of subject1 out of 50
34
Enter the cie marks of subject2 out of 50
45
Enter the cie marks of subject3 out of 50
42
Enter the cie marks of subject4 out of 50
38
Enter the cie marks of subject5 out of 50
39
Enter the see marks of subject1 out of 100
78
Enter the see marks of subject2 out of 100
76
Enter the see marks of subject3 out of 100
88
Enter the see marks of subject4 out of 100
80
Enter the see marks of subject5 out of 100
72
DETAILS OF STUDENT 1
Name :Rashmi Simgh
Usn :1BM19CS293
Sem :2
Total marks in subject1 is 73
Total marks in subject2 is 83
Total marks in subject3 is 86
Total marks in subject4 is 78
Total marks in subject5 is 75
```

LAB PROGRAM 7 : Write a program to demonstrate generics with multiple object parameters.

store
67

Week 10

Lab program 7

```
import java.util.*;  
import java.lang.String;  
class generic<DT1, DT2, DT3>  
{  
    DT1 obj;  
    DT2 obj1;  
    DT3 obj2;  
    generic(DT1 a, DT2 b, DT3 c)  
    {  
        obj = a;  
        obj1 = b;  
        obj2 = c;  
    }  
    DT1 get1()  
    {  
        return obj;  
    }  
    DT2 get2()  
    {  
        return obj1;  
    }
```

DT3 get 3 ()

{ return obj2;

} void showdatatype()

System.out.println ("The type of datatype
used is = " + obj1.getClass().getName());

System.out.println ("The types of datatype
used is = " + obj1.getClass().getName());

System.out.println ("The types of datatype
used is = " + obj2.getClass().getName());

3
3
class genericmain

{ public static void main (String args[])

{ Scanner s = new Scanner (System.in);

System.out.println ("Enter the values");

int x = s.nextInt();

String str = s.next();

double xx = s.nextDouble();

generic < Integer, String, Double >

store
67

a = new generic < Integer, String, Double>
(67, str, 88);

a.showdatatype();

System.out.println ("The integer entered
is = " + a.get1());

System.out.println ("The string Entered
is = " + a.get2());

System.out.println ("The Integer entered
is = " + a.get3());

} }

OUTPUT DEBUG CONSOLE TERMINAL

```
C:\Users\akki\Desktop\java files>javac GenMain.java
```

```
C:\Users\akki\Desktop\java files>java GenMain
Type used is:java.lang.Integer
Type used is:java.lang.String
Type used is:java.lang.Double
integer value=77
string value=FRUITS
double value=55.63
```

```
C:\Users\akki\Desktop\java files>
```

LAB PROGRAM 8: 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 Wrong Age() 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.

Lab programs

```
import java.util.*;
import java.lang.String;
class WrongAge extends Exception {
    int age;
    WrongAge (int x)
    {
        age = x;
    }
    public String toString()
    {
        return "AGE of son = " + age + " is entered
                incorrectly";
    }
}
```

```
class father
{
    int a;
    father (int x)
    {
        a = x;
    }
}
```

```
class son extends father {
```

store
67

int age;

son (int fage, int sage)
{

super (fage);

} age = sage;

void compute () throws WrongAge
{

if (age >= a)

{

throw new WrongAge (age);

}
else {

System.out.println ("The ages
are entered correctly").

System.out.println ("Father's age = "
+ a + " \t " + "Son's age = " + age);

}

} }

class expmain

{ public static void main (String args [])

{ Scanner s = new Scanner (System.in);

```
System.out.println("Enter father's age")  
int f = s.nextInt();
```

```
System.out.println("Enter son's age")  
int so = s.nextInt();
```

```
son ss = new son(f, so);
```

```
try  
{
```

```
ss.compute();
```

```
}
```

```
catch (WrongAge e)
```

```
{
```

```
System.out.println(e);
```

```
}
```

```
}"
```

```
"
```

```
C:\Users\akki\Desktop\java files>javac exception_age.java
C:\Users\akki\Desktop\java files>java AgeMain
Enter Son's age:
78
Enter Father's age:
20
Wrong Age Exception

C:\Users\akki\Desktop\java files>java AgeMain
Enter Son's age:
23
Enter Father's age:
-65
Wrong Age Exception

C:\Users\akki\Desktop\java files>
```

LAB PROGRAM 9: 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.

Week 11Lab 9

```
import java.util.*;
import java.lang.String;

class NewThread implements Runnable
{
    String name;
    Thread t;
    int n;
    NewThread (String threadname, int x)
    {
        name = threadname;
        n = x;
        t = new Thread (this, name);
        System.out.println("New thread;" + t);
        t.start();
    }

    public void run()
    {
        try
        {
            for (int i=5; i>0; i--)
                System.out.println(name + " " + i);
        }
        catch (Exception e)
        {
            System.out.println("Exception caught");
        }
    }
}
```

```
{  
    System.out.println(name);  
    Thread.sleep(n);  
}  
}  
catch (InterruptedException e)  
{  
    System.out.println("Interrupted");  
    System.out.println(name + " exiting");  
}  
}
```

class Multithreaddemo

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

NewThread t1 = new NewThread ("BMS COLLEGE
OF ENGINEERING", 10000);

NewThread t2 = new NewThread ("CSE", 2000);

System.out.println ("Main thread Existing")

```
C:\Users\akki\Desktop\java files>javac thread2.java

C:\Users\akki\Desktop\java files>java Multithreaddemo
New thread:Thread[BMS COLLEGE OF ENGINEERING,5,main]
New thread:Thread[CSE,5,main]
BMS COLLEGE OF ENGINEERING
CSE
CSE
CSE
CSE
CSE
BMS COLLEGE OF ENGINEERING
CSE
CSE
CSE
CSE
CSE
BMS COLLEGE OF ENGINEERING

C:\Users\akki\Desktop\java files>[]
```

LAB PROGRAM 10: Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an ArithmeticException. Display the exception in a message dialog box.

Week 12

lab program 10.

```
import java.util.*;
import java.lang.String;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class Division extends Frame implements
    ActionListener
```

{

```
TextField n1, n2, res;
```

```
Label l1, l2, lres;
```

```
Button b;
```

```
public Division()
```

{

```
setLayout(new FlowLayout());
```

```
lable l1 = new Label("NUM1", Label.RIGHT);
```

```
lable l2 = new Label("NUM2", Label.RIGHT);
```

```
lable lres = new Label("RESULT", Label.RIGHT);
```

```
n1 = new TextField(12);
```

```
n2 = new TextField(8);
```

```
res = new TextField(10);
```

```
b = new Button("DIVIDE")
```

```
add(n1);
add(n2);
add(b);
add(res);
add(bres);
b.addActionListener(this);
addWindowListener(new MyWindowAdapter());
}
public void actionPerformed(ActionEvent ae)
{
    if(ae.getSource() == b)
    {
        try
        {
            int num1 = Integer.parseInt(n1.getText());
            int num2 = Integer.parseInt(n2.getText());
            int num3 = num1 / num2;
            res.setText(String.valueOf(num3));
        }
        catch(NumberFormatException ne)
        {
            JOptionPane.showMessageDialog(this, ne, "ERROR",
                JOptionPane.ERROR_MESSAGE);
        }
    }
}
```

store
67

catch (ArithmeticException a)

{

JOptionPane.showMessageDialog(this, a, "ERROR",
JOptionPane.ERROR_MESSAGE);

}

}

public static void main (String args[])

{

Division i = new Division (),
i.setSize (new Dimension (400, 400)),
i.setTitle ("INTEGER DIVISION OF TWO
NUMBERS");

i.setVisible (true);

}

class MyWindowAdapter extends WindowAdapter

{

public void windowClosing (WindowEvent we)

{

System.exit (0);

}

}

}

INTEGER DIVISION OF TWO NUMBERS

NUM1 NUM2 DIVIDE RESULT

INTEGER DIVISION OF TWO NUMBERS

NUM1 NUM2 DIVIDE RESULT



