VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“JnanaSangama”, Belgaum -590014, Karnataka.

LAB REPORT
on

Object Oriented Java Programming

(23CS3PCOOJ)

Submitted by

THANU GEORGE (1WA23CS019)

in partial fulfillment for the award of the degree of
BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING

B.M.S. COLLEGE OF ENGINEERING
(Autonomous Institution under VTU)

BENGALURU-560019
Sep-2024 to Jan-2025

B.M.S. College of Engineering,
Bull Temple Road, Bangalore 560019
(Affiliated To Visvesvaraya Technological University, Belgaum)
Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the Lab work entitled “Object Oriented Java Programming (23CS3PCOOJ)” carried out by **THANU GEORGE (1WA23CS019),** who is bonafide student of **B.M.S. College of Engineering.** It is in partial fulfillment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum. The Lab report has been approved as it satisfies the academic requirements in respect of an Object Oriented Java Programming (23CS3PCOOJ) work prescribed for the said degree.

Prasad G R Assistant Professor Department of CSE, BMSCE	Dr. Kavitha Sooda Professor & HOD Department of CSE, BMSCE

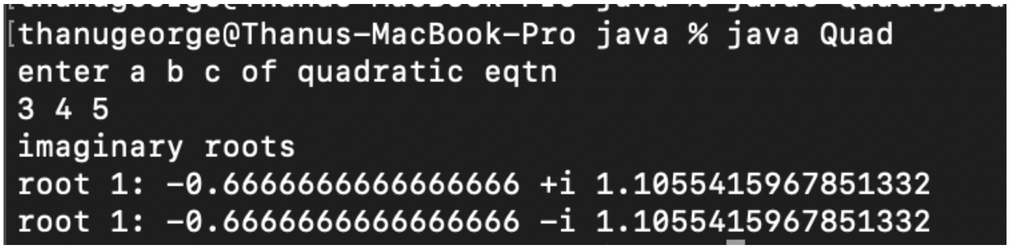
Index

Program 1

Implement Quadratic Equation

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

class Quad{
        public static void main(String args[]){
                double a,b,c,s1,s2;
                Scanner sc = new Scanner(System.in);
                System.out.println("enter a b c of quadratic eqtn");
                a = sc.nextDouble();
                b = sc.nextDouble();
                c = sc.nextDouble();
                if(a==0){
                        System.out.print("INVALID EQTN");
                        return;
                }
                double d= b*b-4*a*c;
                if (d>0){
                        System.out.println("real and distinct roots");
                        s1=(-b + Math.sqrt(d))/(2*a);
                        s2=(-b - Math.sqrt(d))/(2*a);
                        System.out.print("roots are "+s1+" "+s2);
                }else if(d==0){
                        System.out.println("real and equal roots");
                        s1= -b/(2*a);
                        System.out.print("root: "+s1);
                }else{
                        s1= -b/(2*a);
                        s2= Math.sqrt(-d)/(2*a);
                        System.out.println("imaginary roots");
                        System.out.println("root 1: "+s1+" +i "+s2);
                        System.out.println("root 1: "+s1+" -i "+s2);
                }
        }
}
```

OUTPUT:

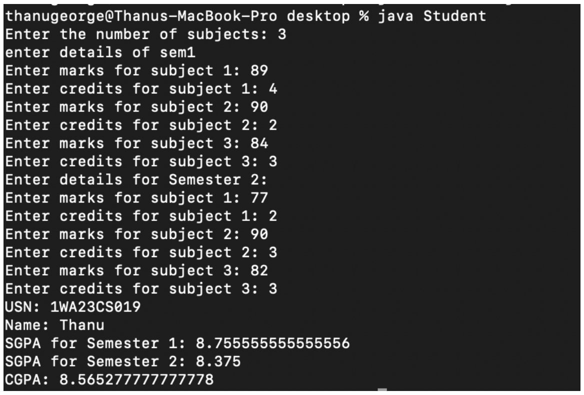


```
[thanugeorge@Thanus-MacBook-Pro java % java Quad
enter a b c of quadratic eqtn
3 4 5
imaginary roots
root 1: -0.6666666666666666 +i 1.1055415967851332
root 1: -0.6666666666666666 -i 1.1055415967851332
```

Program 2
Calculate SGPA

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

class StudentD{
String usn, name;
int[] m1=new int[10],m2=new int[10];
int[] c1 = new int[10], c2 = new int[10];
int sub;

void accept(String n,String u, int s){
this.usn=u;
this.name=n;
this.sub=s;
Scanner sc = new Scanner(System.in);
System.out.println("enter details of sem1");
for(int i=0;i<s;i++){
        System.out.print("Enter marks for subject " + (i + 1) + ": ");
        m1[i] = sc.nextInt();
        System.out.print("Enter credits for subject " + (i + 1) + ": ");
        c1[i] = sc.nextInt();
}
System.out.println("Enter details for Semester 2:");
for (int i = 0; i < sub; i++) {
        System.out.print("Enter marks for subject " + (i + 1) + ": ");
        m2[i] = sc.nextInt();
        System.out.print("Enter credits for subject " + (i + 1) + ": ");
        c2[i] = sc.nextInt();
}

}

void display() {
System.out.println("USN: " + usn);
System.out.println("Name: " + name);
}
double sgpa(int sem){
double totalm=0,totalc=0;
if (sem==1){
for(int i=0;i<sub;i++){
totalm+=m1[i]*c1[i];
totalc+=c1[i];
}
}
else if (sem == 2) {
for (int i = 0; i < sub; i++) {
totalm += m2[i] * c2[i];
totalc += c2[i];
}
}
if (totalc == 0) return 0;
return totalm / (totalc*10);

}

double calcCgpa() {
double sgpaSem1 = sgpa(1);
double sgpaSem2 = sgpa(2);
return (sgpaSem1 + sgpaSem2) / 2;
}
}

public class Student {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.print("Enter the number of subjects: ");
int sub = sc.nextInt();
StudentD student = new StudentD();
student.accept("Thanu","1WA23CS019", sub);
student.display();
double sgpaSem1 = student.sgpa(1);
double sgpaSem2 = student.sgpa(2);
System.out.println("SGPA for Semester 1: " + sgpaSem1);
System.out.println("SGPA for Semester 2: " + sgpaSem2);
double cgpa = student.calcCgpa();
System.out.println("CGPA: " + cgpa);
sc.close();
}
}
```

OUTPUT:



```
thanugeorge@Thanus-MacBook-Pro desktop % java Student
Enter the number of subjects: 3
enter details of sem1
Enter marks for subject 1: 89
Enter credits for subject 1: 4
Enter marks for subject 2: 90
Enter credits for subject 2: 2
Enter marks for subject 3: 84
Enter credits for subject 3: 3
Enter details for Semester 2:
Enter marks for subject 1: 77
Enter credits for subject 1: 2
Enter marks for subject 2: 90
Enter credits for subject 2: 3
Enter marks for subject 3: 82
Enter credits for subject 3: 3
USN: 1WA23CS019
Name: Thanu
SGPA for Semester 1: 8.755555555555556
SGPA for Semester 2: 8.375
CGPA: 8.565277777777778
```

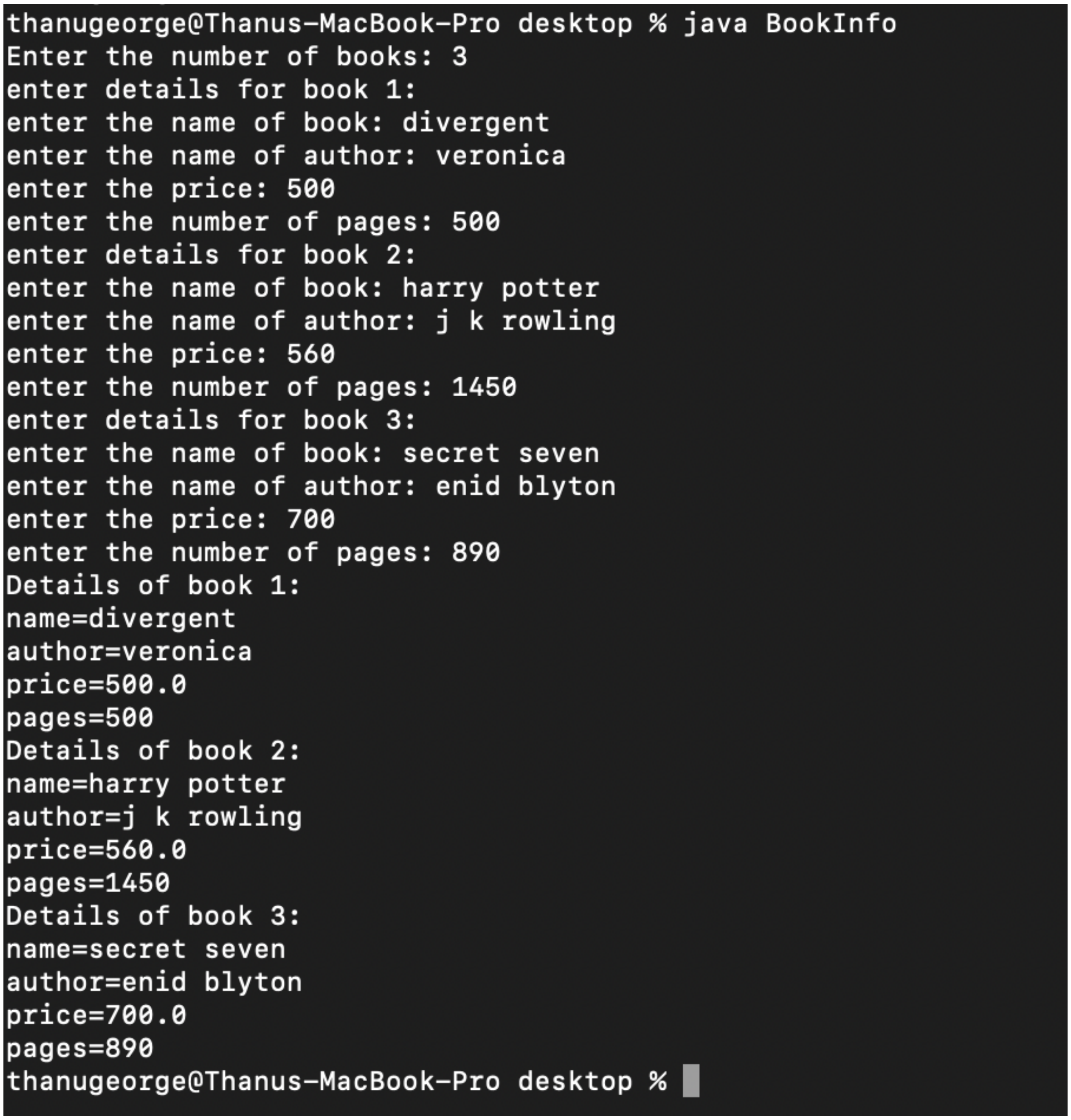
Program 3

N book objects

```
import java.util.Scanner;
class Book {
   String name;
   String author;
   double price;
   int pages;
public Book(String name, String author, double price, int pages) {
      this.name = name;
      this.author = author;
      this.price = price;
      this.pages = pages;
   }
public String toString(){
return "name="+name+"\n"+"author="+author+"\n"+"price="+price+"\n"+"pages="+pages;
}
}
public class BookInfo {
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.print("Enter the number of books: ");
      int n = sc.nextInt();
      sc.nextLine();
      Book[] books = new Book[n];
      for (int i = 0; i < n; i++) {
         System.out.println("enter details for book " + (i + 1) + ":");
         System.out.print("enter the name of book: ");
         String name = sc.nextLine();
         System.out.print("enter the name of author: ");
         String author = sc.nextLine();
         System.out.print("enter the price: ");
         double price = sc.nextDouble();
         sc.nextLine();
         System.out.print("enter the number of pages: ");
         int pages = sc.nextInt();
         sc.nextLine();
         books[i] = new Book(name, author, price, pages);
      }

      for (int i = 0; i < n; i++) {
         System.out.println("Details of book " + (i + 1) + ":");
         System.out.println(books[i].toString());

      }
```

```
        sc.close();
    }
}
```

OUTPUT:

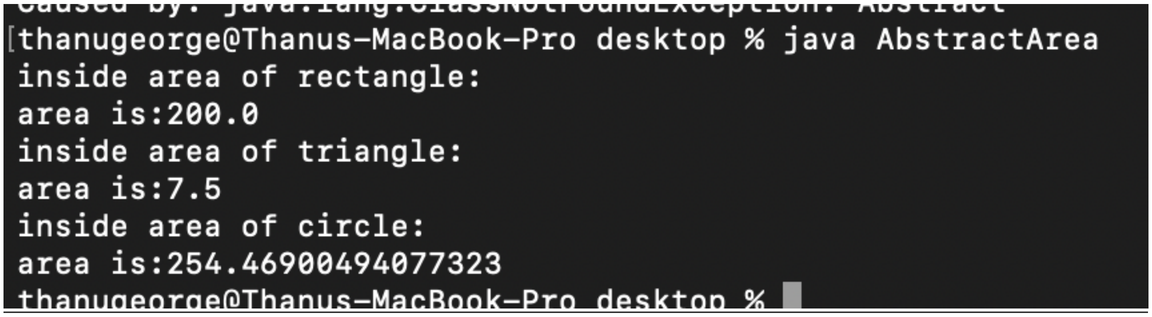


```
thanugeorge@Thanus-MacBook-Pro desktop % java BookInfo
Enter the number of books: 3
enter details for book 1:
enter the name of book: divergent
enter the name of author: veronica
enter the price: 500
enter the number of pages: 500
enter details for book 2:
enter the name of book: harry potter
enter the name of author: j k rowling
enter the price: 560
enter the number of pages: 1450
enter details for book 3:
enter the name of book: secret seven
enter the name of author: enid blyton
enter the price: 700
enter the number of pages: 890
Details of book 1:
name=divergent
author=veronica
price=500.0
pages=500
Details of book 2:
name=harry potter
author=j k rowling
price=560.0
pages=1450
Details of book 3:
name=secret seven
author=enid blyton
price=700.0
pages=890
thanugeorge@Thanus-MacBook-Pro desktop % 
```

program 4

```
import java.util.*;
abstract class Figure{
double dim1;
double dim2;
Figure(double a,double b)
{
dim1=a;
dim2=b;
}
abstract double area();
}
class rectangle extends Figure{
rectangle(double a,double b){super(a,b);}
double area()
{
System.out.println("inside area of rectangle:");
return dim1*dim2;
}
}
class triangle extends Figure{
triangle(double a,double b){super(a,b);}
double area()
{
System.out.println("inside area of triangle:");
return (dim1*dim2)/2;
}
}
class circle extends Figure{
circle(double a){super(a,a);}
double area()
{
System.out.println("inside area of circle:");
return Math.PI*(dim1*dim1);
}
}
class AbstractArea{
public static void main(String [] args){
rectangle r=new rectangle(10,20);
triangle t=new triangle(3,5);
circle c=new circle(9);
Figure figref;
figref=r;
System.out.println("area is:"+figref.area());
figref=t;
System.out.println("area is:"+figref.area());
figref=c;
System.out.println("area is:"+figref.area());
}
}
```

OUTPUT:



```
Caused by: java.lang.ClassNotFoundException: Abstract
[thanugeorge@Thanus-MacBook-Pro desktop % java AbstractArea
inside area of rectangle:
area is:200.0
inside area of triangle:
area is:7.5
inside area of circle:
area is:254.46900494077323
thanugeorge@Thanus-MacBook-Pro desktop %
```

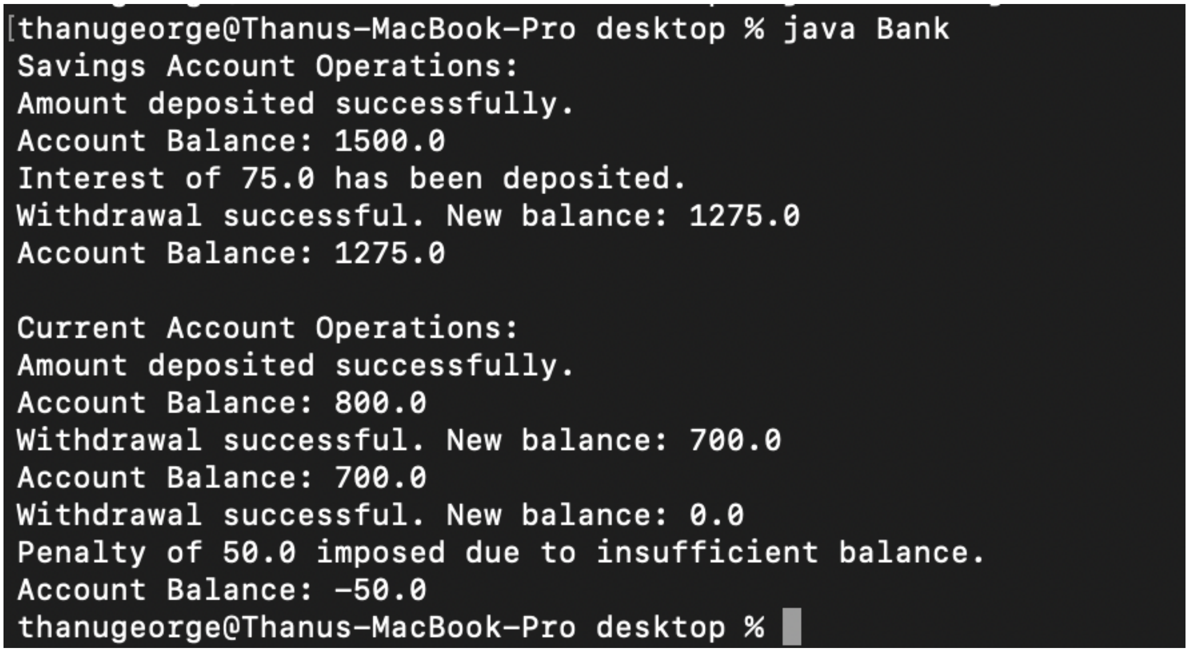
program5

bank account

```
import java.util.Scanner;
class Account {
   protected String customerName;
   protected String accountNumber;
   protected String accountType;
   protected double balance;

   public Account(String customerName, String accountNumber, String accountType, double
initialBalance) {
      this.customerName = customerName;
      this.accountNumber = accountNumber;
      this.accountType = accountType;
      this.balance = initialBalance;
   }

   public void deposit(double amount) {
      if (amount > 0) {
         balance += amount;
         System.out.println("Amount deposited successfully.");
      } else {
         System.out.println("Invalid deposit amount.");
      }
   }

   public void displayBalance() {
      System.out.println("Account Balance: " + balance);
   }
}

class SavAcct extends Account {
   private double interestRate;

   public SavAcct(String customerName, String accountNumber, double initialBalance, double
interestRate) {
      super(customerName, accountNumber, "Savings", initialBalance);
      this.interestRate = interestRate;
   }

   public void computeAndDepositInterest() {
      double interest = balance * interestRate / 100;
      balance += interest;
      System.out.println("Interest of " + interest + " has been deposited.");
   }
```

```
    public void withdraw(double amount) {
        if (amount > 0 && amount <= balance) {
            balance -= amount;
            System.out.println("Withdrawal successful. New balance: " + balance);
        } else {
            System.out.println("Invalid withdrawal amount or insufficient balance.");
        }
    }
}

class CurAcct extends Account {
    private static final double MINIMUM_BALANCE = 500.0;
    private static final double PENALTY = 50.0;

    public CurAcct(String customerName, String accountNumber, double initialBalance) {
        super(customerName, accountNumber, "Current", initialBalance);
    }

    public void checkAndImposePenalty() {
        if (balance < MINIMUM_BALANCE) {
            balance -= PENALTY;
            System.out.println("Penalty of " + PENALTY + " imposed due to insufficient balance.");
        }
    }

    public void withdraw(double amount) {
        if (amount > 0 && amount <= balance) {
            balance -= amount;
            System.out.println("Withdrawal successful. New balance: " + balance);
            checkAndImposePenalty();
        } else {
            System.out.println("Invalid withdrawal amount or insufficient balance.");
        }
    }
}

public class Bank {
    public static void main(String[] args) {
        SavAcct savingsAccount = new SavAcct("Alice", "SA12345", 1000.0, 5.0);
        CurAcct currentAccount = new CurAcct("Bob", "CA54321", 600.0);

        // Savings account operations
        System.out.println("Savings Account Operations:");
        savingsAccount.deposit(500);
        savingsAccount.displayBalance();
        savingsAccount.computeAndDepositInterest();
        savingsAccount.withdraw(300);
        savingsAccount.displayBalance();

        // Current account operations
        System.out.println("\nCurrent Account Operations:");
        currentAccount.deposit(200);
        currentAccount.displayBalance();
        currentAccount.withdraw(100);
        currentAccount.displayBalance();
        currentAccount.withdraw(700);
        currentAccount.displayBalance();
    }
}
```

OUTPUT



```
[thanugeorge@Thanus-MacBook-Pro desktop % java Bank
Savings Account Operations:
Amount deposited successfully.
Account Balance: 1500.0
Interest of 75.0 has been deposited.
Withdrawal successful. New balance: 1275.0
Account Balance: 1275.0

Current Account Operations:
Amount deposited successfully.
Account Balance: 800.0
Withdrawal successful. New balance: 700.0
Account Balance: 700.0
Withdrawal successful. New balance: 0.0
Penalty of 50.0 imposed due to insufficient balance.
Account Balance: -50.0
thanugeorge@Thanus-MacBook-Pro desktop %
```

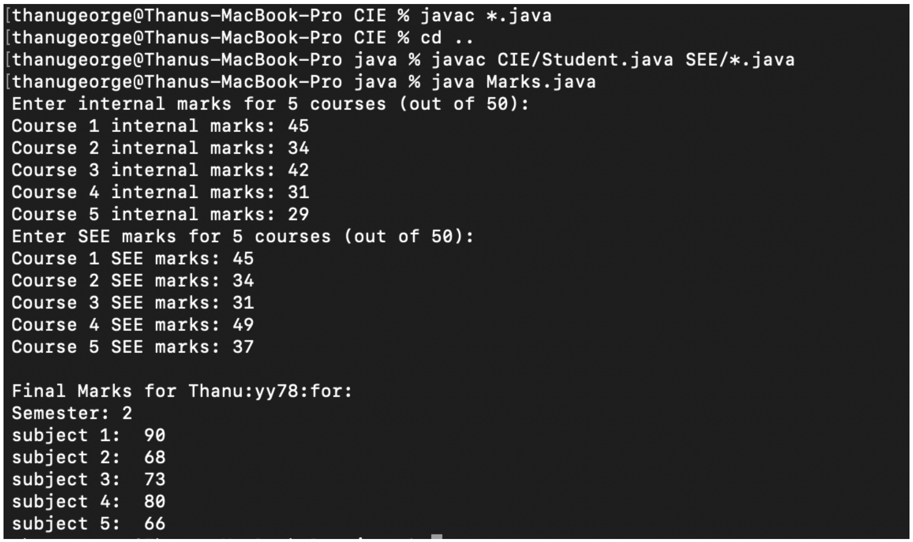
program6

packages

```
import CIE.*;
import SEE.*;
import java.util.Scanner;

class Marks{
public static void main(String args[]){
Scanner sc=new Scanner(System.in);

System.out.println("Enter internal marks for 5 courses (out of 50): ");
int internalMarks[] = new int[5];
for (int j=0;j<5;j++)
{
System.out.print("Course " + (j + 1) + " internal marks: ");
internalMarks[j] = sc.nextInt();
}
Internals im = new Internals(internalMarks);
System.out.println("Enter SEE marks for 5 courses (out of 50): ");
int seeMarks[] = new int[5];
for (int j=0;j<5;j++)
{
System.out.print("Course "+(j + 1)+" SEE marks: ");
seeMarks[j] = sc.nextInt();
}
Externals ext=new Externals("Thanu","yy78",2,seeMarks);
System.out.println("\nFinal Marks for " + ext.name + ":" + ext.usn + ":for:");
System.out.println("Semester: " + ext.sem);
for(int x=0;x<5;x++){
int total=0;
total=im.m[x]+ext.m[x];
System.out.println("subject "+(x+1)+":  "+total);
}
}
}

package CIE;

public class Internals{
        public int[] m = new int[5];
        public Internals(int[] marks) {
        for (int i = 0; i < 5; i++) {
                m[i] = marks[i];        }
        }
}
```

```
package CIE;

public class Student{
        public String usn;
        public String name;
        public int sem;
        public Student(String n, String u,int s){
        usn=u;
        name=n;
        sem=s;
        }
}

package SEE;
import CIE.Student;

public class Externals extends Student{
public int[] m = new int[5];
public Externals(String name, String usn, int sem,int seeMarks[]){
super(name,usn,sem);
for (int i = 0; i < 5; i++)
{
this.m[i] = seeMarks[i];
}
}
}
```

OUTPUT:



```
[thanugeorge@Thanus-MacBook-Pro CIE % javac *.java
[thanugeorge@Thanus-MacBook-Pro CIE % cd ..
[thanugeorge@Thanus-MacBook-Pro java % javac CIE/Student.java SEE/*.java
[thanugeorge@Thanus-MacBook-Pro java % java Marks.java
 Enter internal marks for 5 courses (out of 50):
 Course 1 internal marks: 45
 Course 2 internal marks: 34
 Course 3 internal marks: 42
 Course 4 internal marks: 31
 Course 5 internal marks: 29
 Enter SEE marks for 5 courses (out of 50):
 Course 1 SEE marks: 45
 Course 2 SEE marks: 34
 Course 3 SEE marks: 31
 Course 4 SEE marks: 49
 Course 5 SEE marks: 37

 Final Marks for Thanu:yy78:for:
 Semester: 2
 subject 1:  90
 subject 2:  68
 subject 3:  73
 subject 4:  80
 subject 5:  66
```

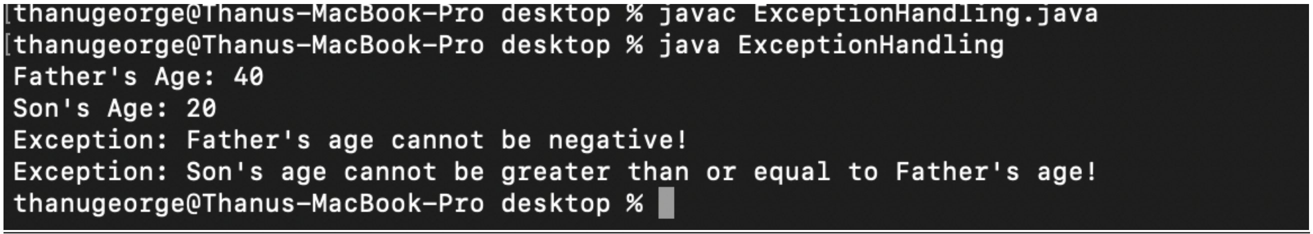
program7

exceptions

```
class WrongAgeException extends Exception {
   public WrongAgeException(String message) {
      super(message);
   }
}

class Father {
   protected int age;

   public Father(int age) throws WrongAgeException {
      if (age < 0) {
         throw new WrongAgeException("Father's age cannot be negative!");
      }
      this.age = age;
   }
}

class Son extends Father {
   private int sonAge;

   public Son(int fatherAge, int sonAge) throws WrongAgeException {
      super(fatherAge);
      if (sonAge < 0) {
         throw new WrongAgeException("Son's age cannot be negative!");
      }
      if (sonAge >= fatherAge) {
         throw new WrongAgeException("Son's age cannot be greater than or equal to Father's age!");
      }
      this.sonAge = sonAge;
   }

   public void displayAges() {
      System.out.println("Father's Age: " + age);
      System.out.println("Son's Age: " + sonAge);
   }
}

public class ExceptionHandling {
   public static void main(String[] args) {
      try {
         Father father = new Father(40);
         Son son = new Son(40, 20);
         son.displayAges();
```

```
            Father invalidFather = new Father(-5);
        } catch (WrongAgeException e) {
            System.out.println("Exception: " + e.getMessage());
        }
        try {
            Son invalidSon = new Son(30, 35);
        } catch (WrongAgeException e) {
            System.out.println("Exception: " + e.getMessage());
        }
    }
}
```

OUTPUT:



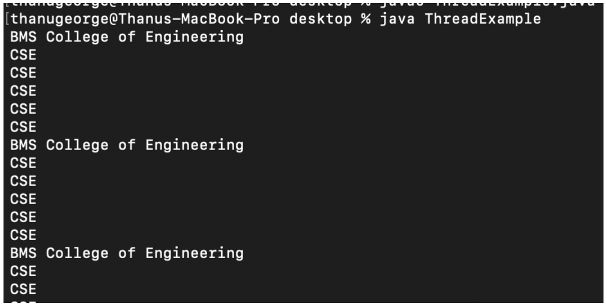
```
[thanugeorge@Thanus-MacBook-Pro desktop % javac ExceptionHandling.java
[thanugeorge@Thanus-MacBook-Pro desktop % java ExceptionHandling
Father's Age: 40
Son's Age: 20
Exception: Father's age cannot be negative!
Exception: Son's age cannot be greater than or equal to Father's age!
thanugeorge@Thanus-MacBook-Pro desktop %
```

program8

threads

```
class CollegeThread extends Thread {
    public void run() {
        try {
            for (int i = 0; i < 5; i++) {
                System.out.println("BMS College of Engineering");
                Thread.sleep(10000);
            }
        } catch (InterruptedException e) {
            System.out.println("CollegeThread interrupted.");
        }
    }
}
class CSEThread extends Thread {
    public void run() {
        try {
            for (int i = 0; i < 25; i++) {
                System.out.println("CSE");
                Thread.sleep(2000);
            }
        } catch (InterruptedException e) {
            System.out.println("CSEThread interrupted.");
        }
    }
}
public class ThreadExample {
    public static void main(String[] args) {
        CollegeThread collegeThread = new CollegeThread();
        CSEThread cseThread = new CSEThread();

        collegeThread.start();
        cseThread.start();
    }
}
```

OUTPUT:



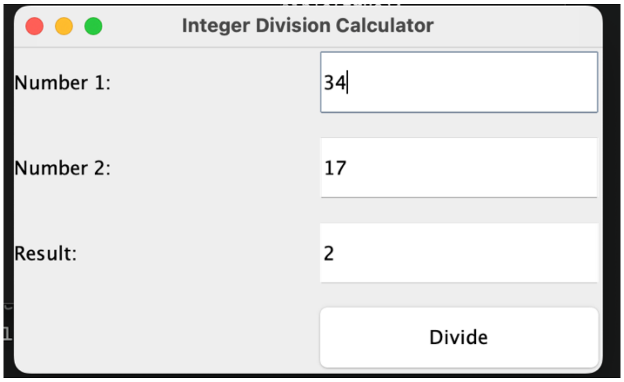
```
thanugeorge@Thanus-MacBook-Pro desktop % javac ThreadExample.java
thanugeorge@Thanus-MacBook-Pro desktop % java ThreadExample
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
```

program9

Interface to perform integer division

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class IntDiv extends JFrame {
   private JTextField txtNum1, txtNum2, txtResult;
   private JButton btnDivide;

   public IntDiv() {
      setTitle("Integer Division Calculator");
      setSize(400, 250);
      setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      setLayout(new GridLayout(4, 2, 10, 10));

      initializeComponents();

      addComponentsToFrame();

      setLocationRelativeTo(null);
   }

   private void initializeComponents() {

      JLabel lblNum1 = new JLabel("Number 1:");
      JLabel lblNum2 = new JLabel("Number 2:");
      JLabel lblResult = new JLabel("Result:");

      txtNum1 = new JTextField();
      txtNum2 = new JTextField();
      txtResult = new JTextField();
      txtResult.setEditable(false);

      btnDivide = new JButton("Divide");
      btnDivide.addActionListener(new ActionListener() {
         @Override
         public void actionPerformed(ActionEvent e) {
            performDivision();
         }
      });
   }

   private void addComponentsToFrame() {
      add(new JLabel("Number 1:"));
      add(txtNum1);
      add(new JLabel("Number 2:"));
      add(txtNum2);
      add(new JLabel("Result:"));
```

```
        add(txtResult);
        add(new JLabel());
        add(btnDivide);
    }

    private void performDivision() {
        try {
            int num1 = Integer.parseInt(txtNum1.getText());
            int num2 = Integer.parseInt(txtNum2.getText());
            int result = num1 / num2;
            txtResult.setText(String.valueOf(result));

        } catch (NumberFormatException nfe) {
            JOptionPane.showMessageDialog(
                this,
                "Please enter valid integers!",
                "Input Error",
                JOptionPane.ERROR_MESSAGE
            );
        } catch (ArithmeticException ae) {
            JOptionPane.showMessageDialog(
                this,
                "Cannot divide by zero!",
                "Arithmetic Error",
                JOptionPane.ERROR_MESSAGE
            );
        }
    }

    public static void main(String[] args) {
        SwingUtilities.invokeLater(new Runnable() {
            public void run() {
                new IntDiv().setVisible(true);
            }
        });
    }
}
```


Integer Division Calculator

Number 1: 34

Number 2: 17

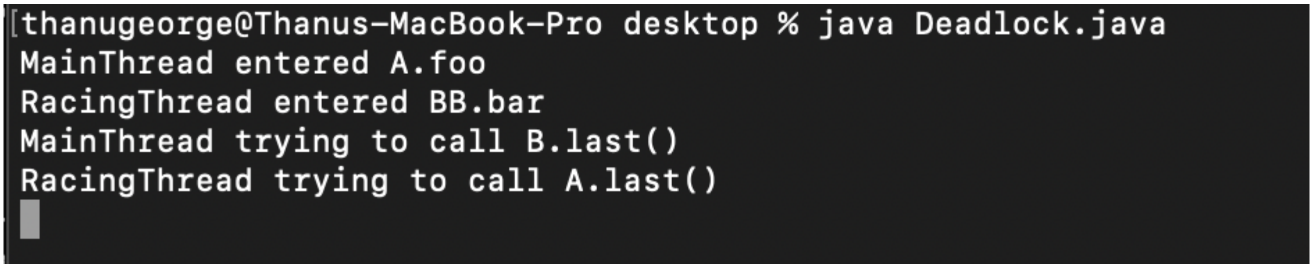
Result: 2

program10

Interprocess Communication and Deadlock

```
class AA {
  synchronized void foo(BB b) {
            String name = Thread.currentThread().getName();
    System.out.println(name + " entered A.foo");
    try {
            Thread.sleep(1000);
    } catch(Exception e) {
            System.out.println("A Interrupted");
    }
    System.out.println(name + " trying to call B.last()");
    b.last();
  }
  synchronized void last() {
    System.out.println("Inside A.last");
  }
}

class BB {
  synchronized void bar(AA a) {
    String name = Thread.currentThread().getName();
    System.out.println(name + " entered BB.bar");
    try {
            Thread.sleep(1000);
    } catch(Exception e) {
            System.out.println("B Interrupted");
    }
    System.out.println(name + " trying to call A.last()");
    a.last();
   }
   synchronized void last() {
    System.out.println("Inside A.last");
   }
}

class Deadlock implements Runnable {
    AA a = new AA();
    BB b = new BB();
    Deadlock() {
            Thread.currentThread().setName("MainThread");
            Thread t = new Thread(this, "RacingThread");
            t.start();
            a.foo(b); // get lock on a in this thread.
            System.out.println("Back in main thread");
    }
    public void run() {
            b.bar(a); // get lock on b in other thread.
```

```
                System.out.println("Back in other thread");
        }
        public static void main(String args[]) {
                new Deadlock();
        }
}
```

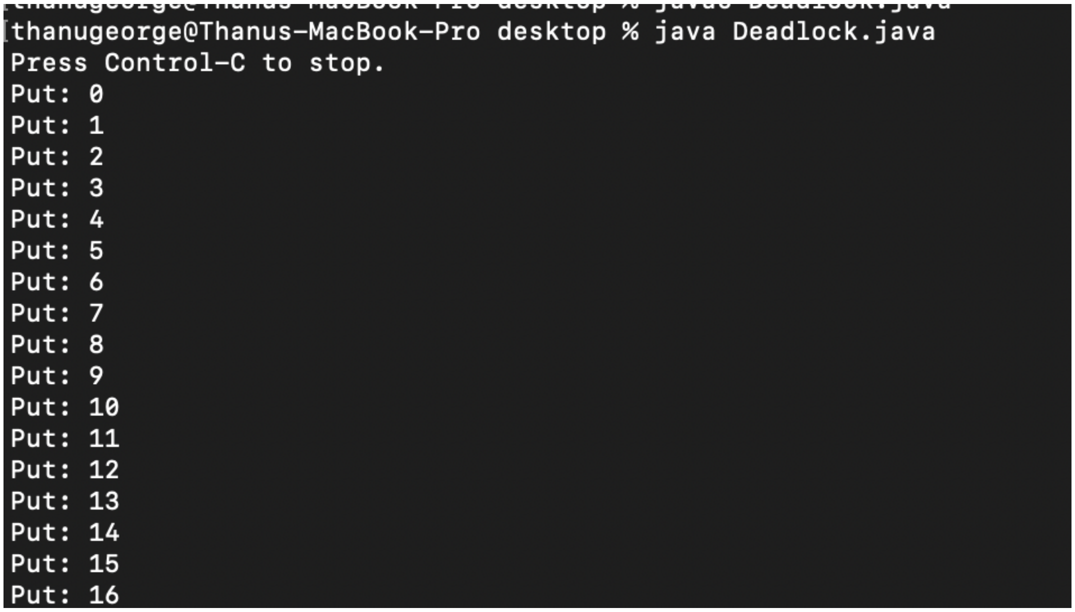


```
[thanugeorge@Thanus-MacBook-Pro desktop % java Deadlock.java
MainThread entered A.foo
RacingThread entered BB.bar
MainThread trying to call B.last()
RacingThread trying to call A.last()
```

INTERPROCESS COMMUNICATION

```
class Q {
        int n;
        synchronized int get() {
                System.out.println("Got: " + n);
                return n;
        }
        synchronized void put(int n) {
                this.n = n;
                System.out.println("Put: " + n);
        }
}

class Producer implements Runnable {
        Q q;
        Producer(Q q) {
                this.q = q;
                new Thread(this, "Producer").start();
        }
        public void run() {
                int i = 0;
                while(true) {
                        q.put(i++);
                }
        }
}

class Consumer implements Runnable {
        Q q;
```

```
    Consumer(Q q) {
        this.q = q;
        new Thread(this, "Consumer").start();
    }
    public void run() {
        while(true) {
            q.get();
        }
    }
}

class Deadlock {
    public static void main(String args[]) {
        Q q = new Q();
        new Producer(q);
        new Consumer(q);
        System.out.println("Press Control-C to stop.");
    }
}
```



```
thanugeorge@Thanus-MacBook-Pro desktop % javac Deadlock.java
[thanugeorge@Thanus-MacBook-Pro desktop % java Deadlock.java
Press Control-C to stop.
Put: 0
Put: 1
Put: 2
Put: 3
Put: 4
Put: 5
Put: 6
Put: 7
Put: 8
Put: 9
Put: 10
Put: 11
Put: 12
Put: 13
Put: 14
Put: 15
Put: 16
```