**ANAGHA ACHARYA**

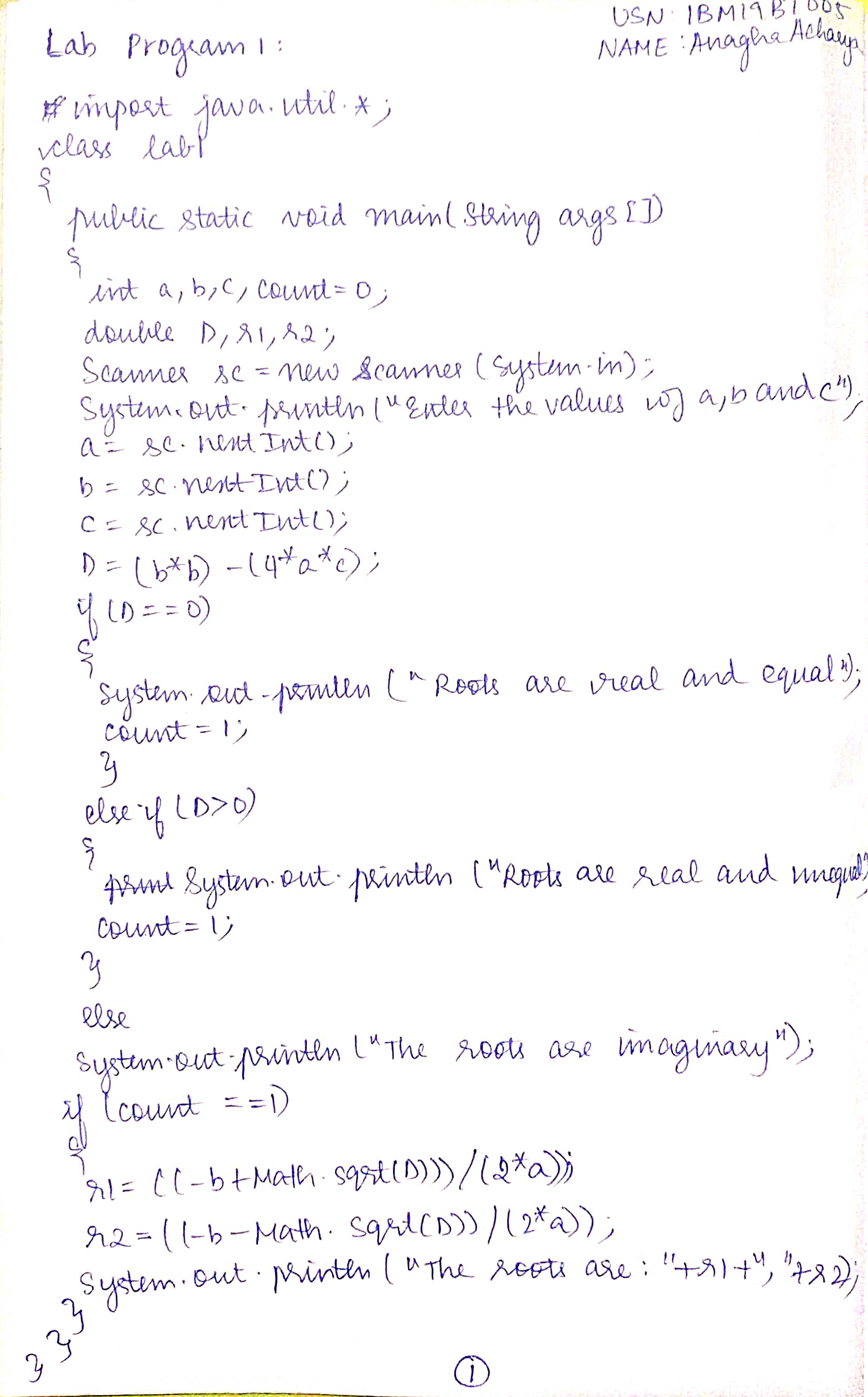
**1BM19BT005**

**OOJ Lab Record**

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.

OBSERVATION:



PROGRAM:

import java.util.\*;

class lab1

{

public static void main(String args[])

{

int a,b,c,count=0;

double D,r1,r2;

Scanner sc=new Scanner(System.in);

System.out.println("Enter the values of a,b and c");

a=sc.nextInt();

b=sc.nextInt();

c=sc.nextInt();

D=(b\*b)-(4\*a\*c);

if(D==0)

{

System.out.println("The roots are real and equal");

count=1;

}

else if(D>0)

{

System.out.println("The roots are real and unequal");

count =1;

}

else

System.out.println("The roots are imaginary");

if(count==1)

{

r1=((-b+Math.sqrt(D))/(2\*a));

r2=((-b-Math.sqrt(D))/(2\*a));

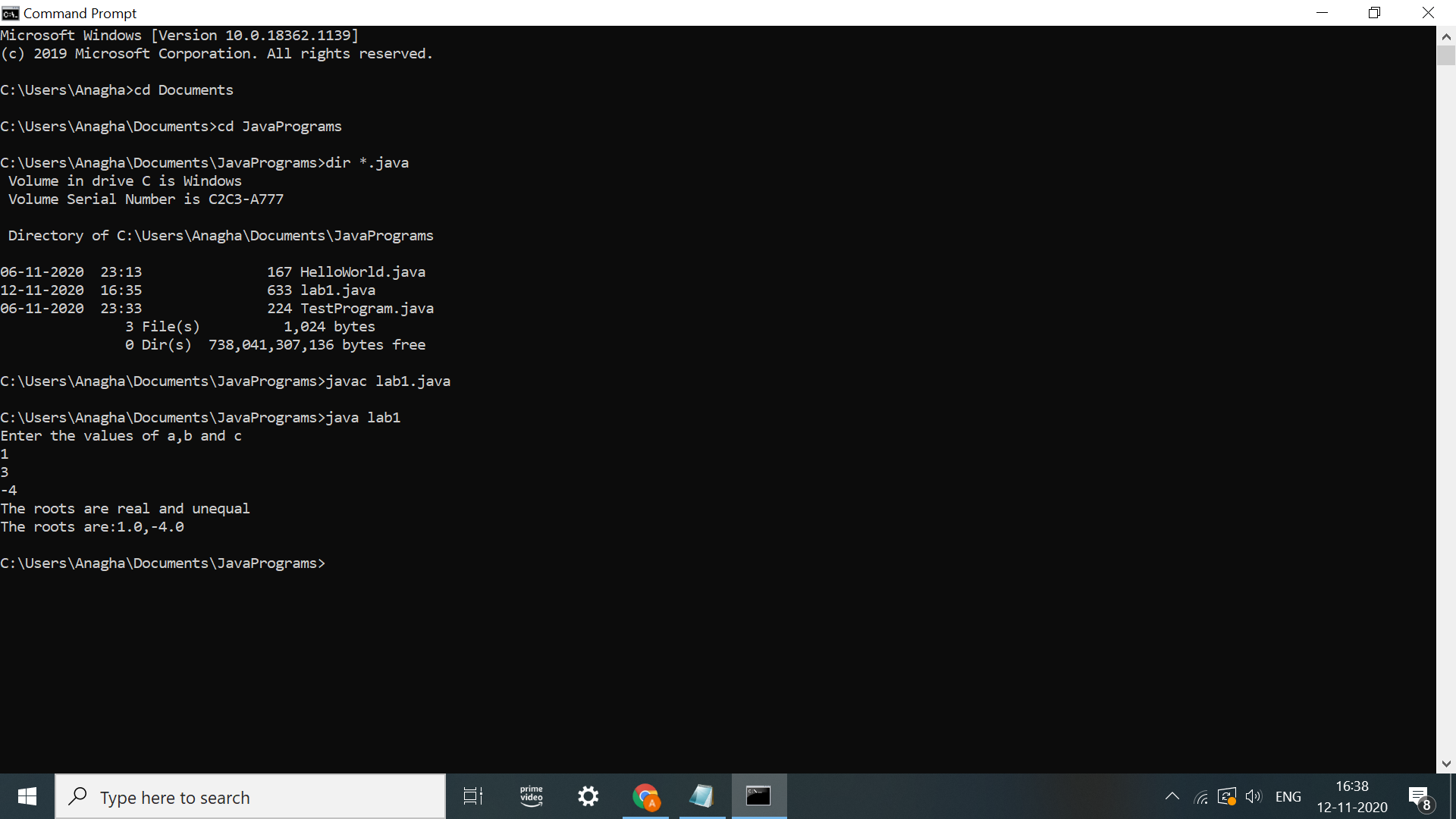
System.out.println("The roots are:"+r1+","+r2);

}

}

}

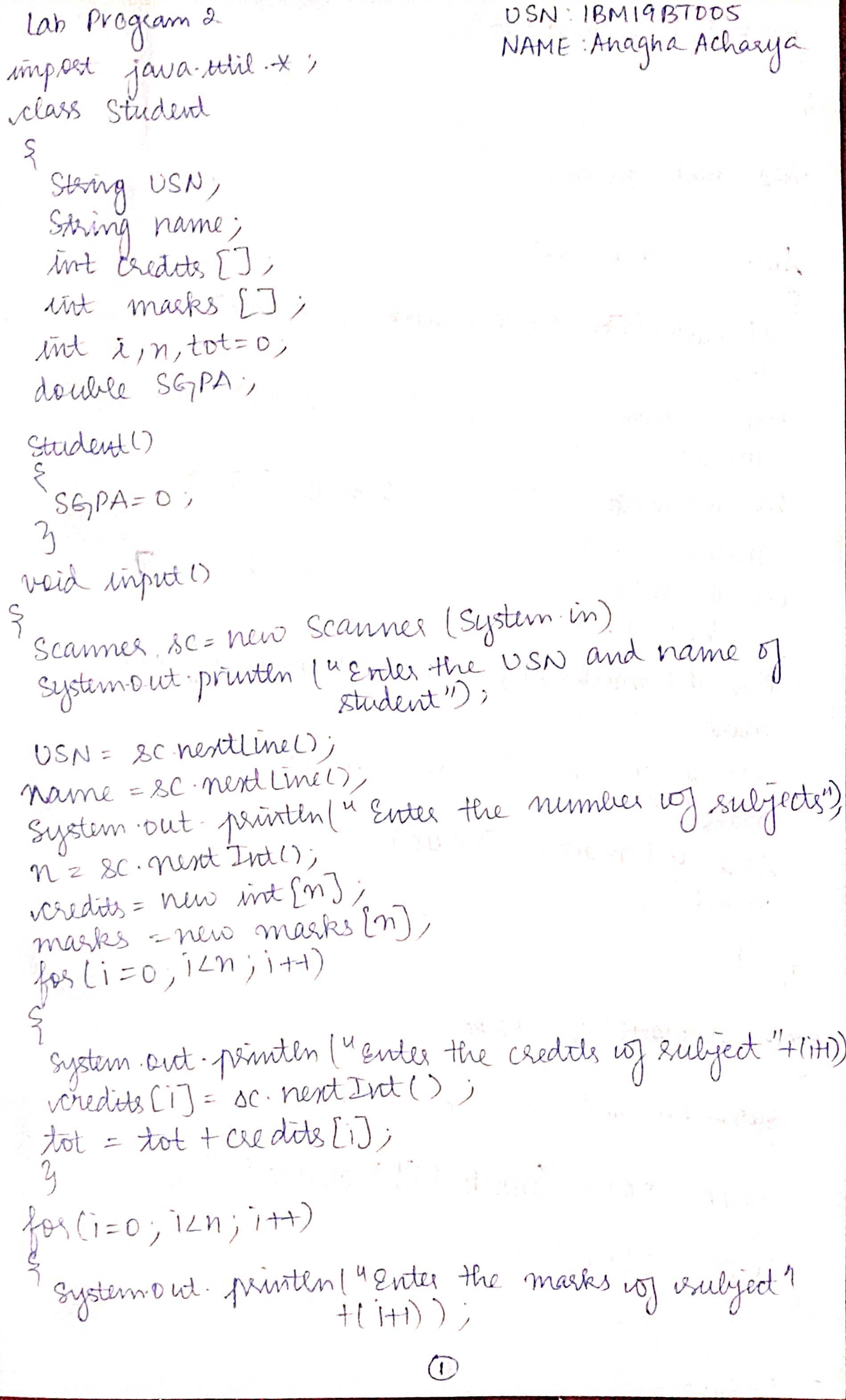
OUTPUT:

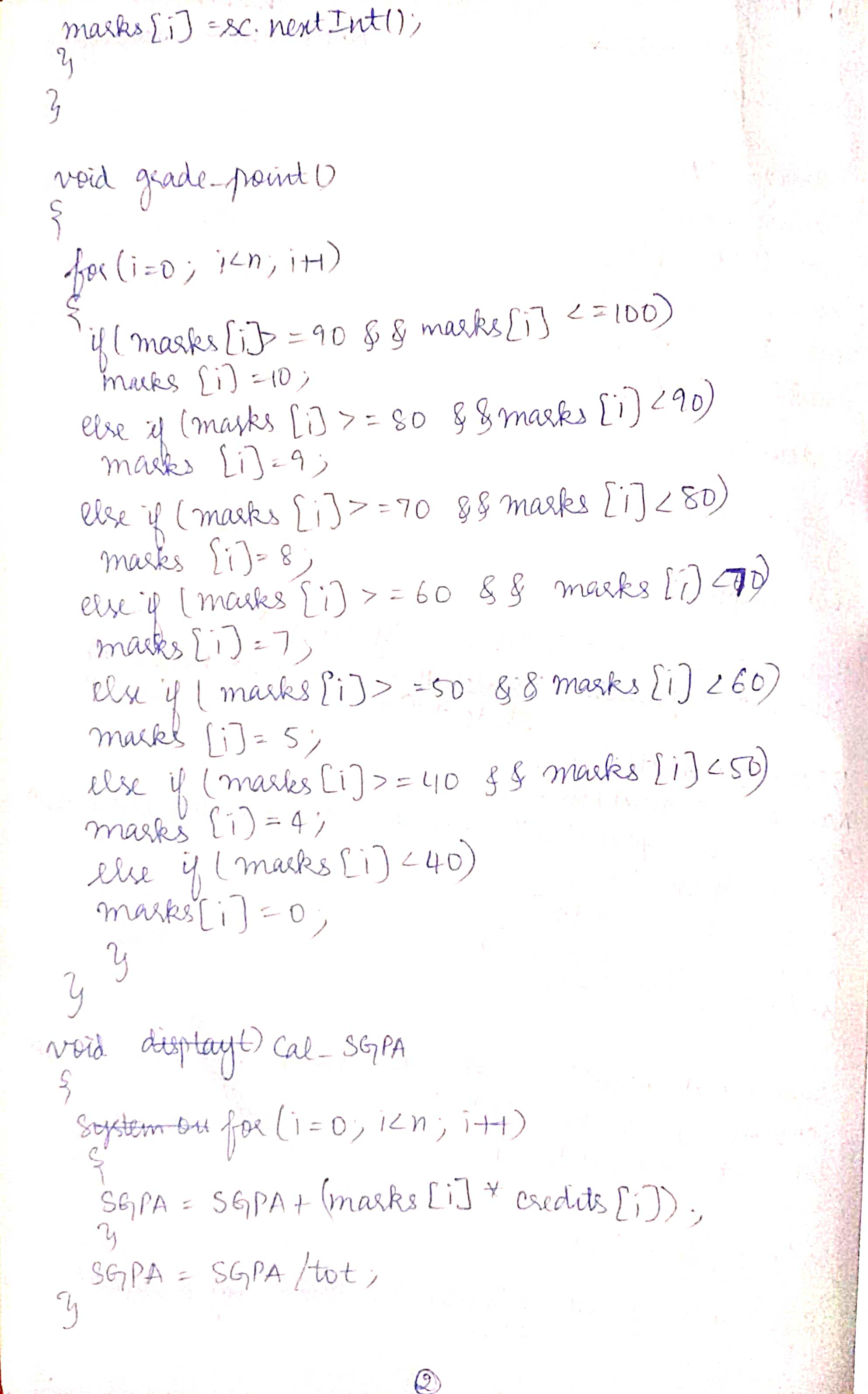


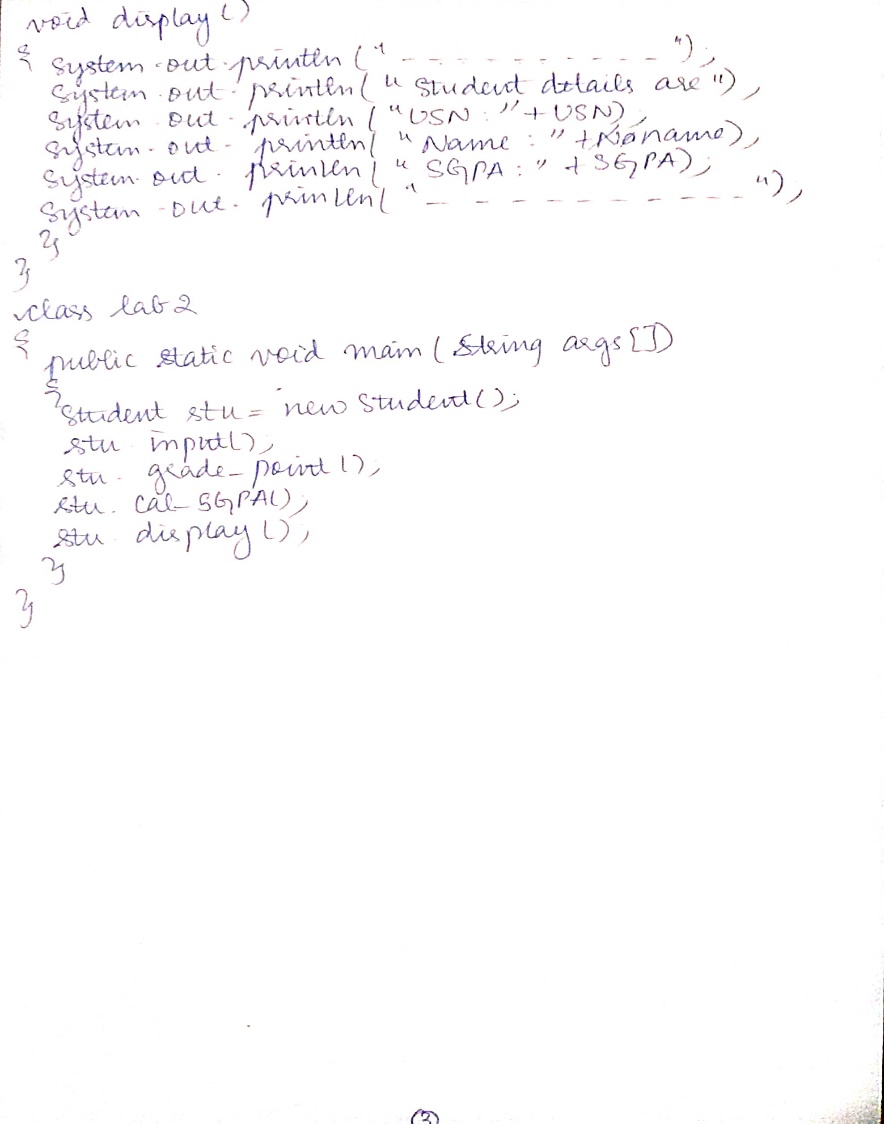
Lab Program 2:

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.

OBSERVATION:







PROGRAM:

import java.util.\*;

class Student

{

String USN;

String name;

int credits[];

int marks[];

int i,n,tot=0;

double SGPA;

Student()

{

SGPA=0;

}

void input()

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter the USN and name of student");

USN=sc.nextLine();

name=sc.nextLine();

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

n=sc.nextInt();

credits=new int[n];

marks=new int[n];

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

{

System.out.println("Enter the credits of subject "+(i+1));

credits[i]=sc.nextInt();

tot=tot+credits[i];

}

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

{

System.out.println("Enter the marks of subject "+(i+1));

marks[i]=sc.nextInt();

}

}

void grade\_point()

{

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

{

if(marks[i]>=90 && marks[i]<=100)

marks[i]=10;

else if(marks[i]>=80 && marks[i]<90)

marks[i]=9;

else if(marks[i]>=70 && marks[i]<80)

marks[i]=8;

else if(marks[i]>=60 && marks[i]<70)

marks[i]=7;

else if(marks[i]>=50 && marks[i]<60)

marks[i]=5;

else if(marks[i]>=40 && marks[i]<50)

marks[i]=4;

else if(marks[i]<40)

marks[i]=0;

}

}

void cal\_SGPA()

{

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

{

SGPA=SGPA+(credits[i]\*marks[i]);

}

SGPA=SGPA/tot;

}

void display()

{

System.out.println("--------------------------------------------------------");

System.out.println("Student details are:");

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

System.out.println("Name:"+name);

System.out.println("SGPA:"+SGPA);

System.out.println("--------------------------------------------------------");

}

}

class lab2

{

public static void main(String args[])

{

Student stu=new Student();

stu.input();

stu.grade\_point();

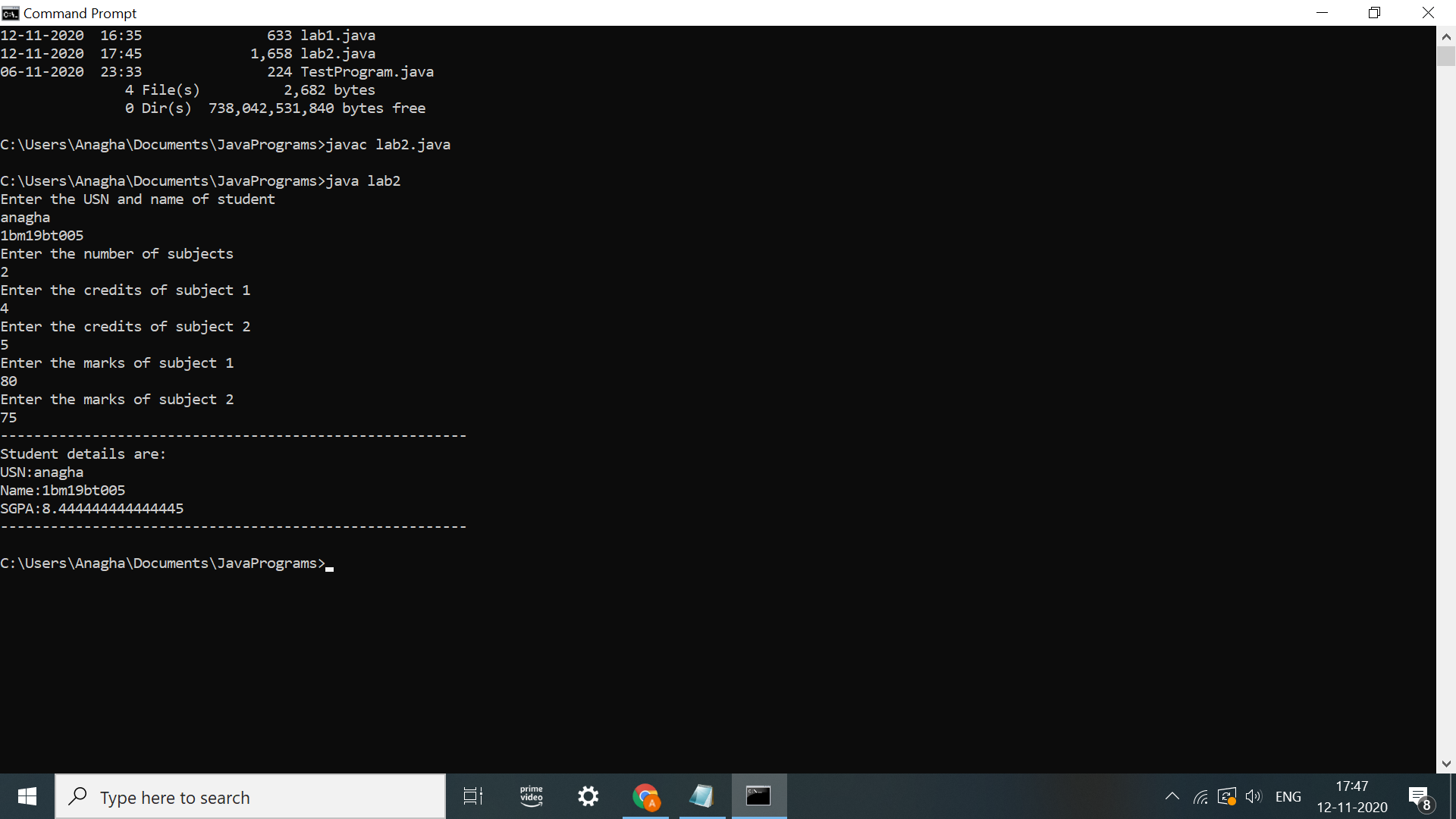
stu.cal\_SGPA();

stu.display();

}

}

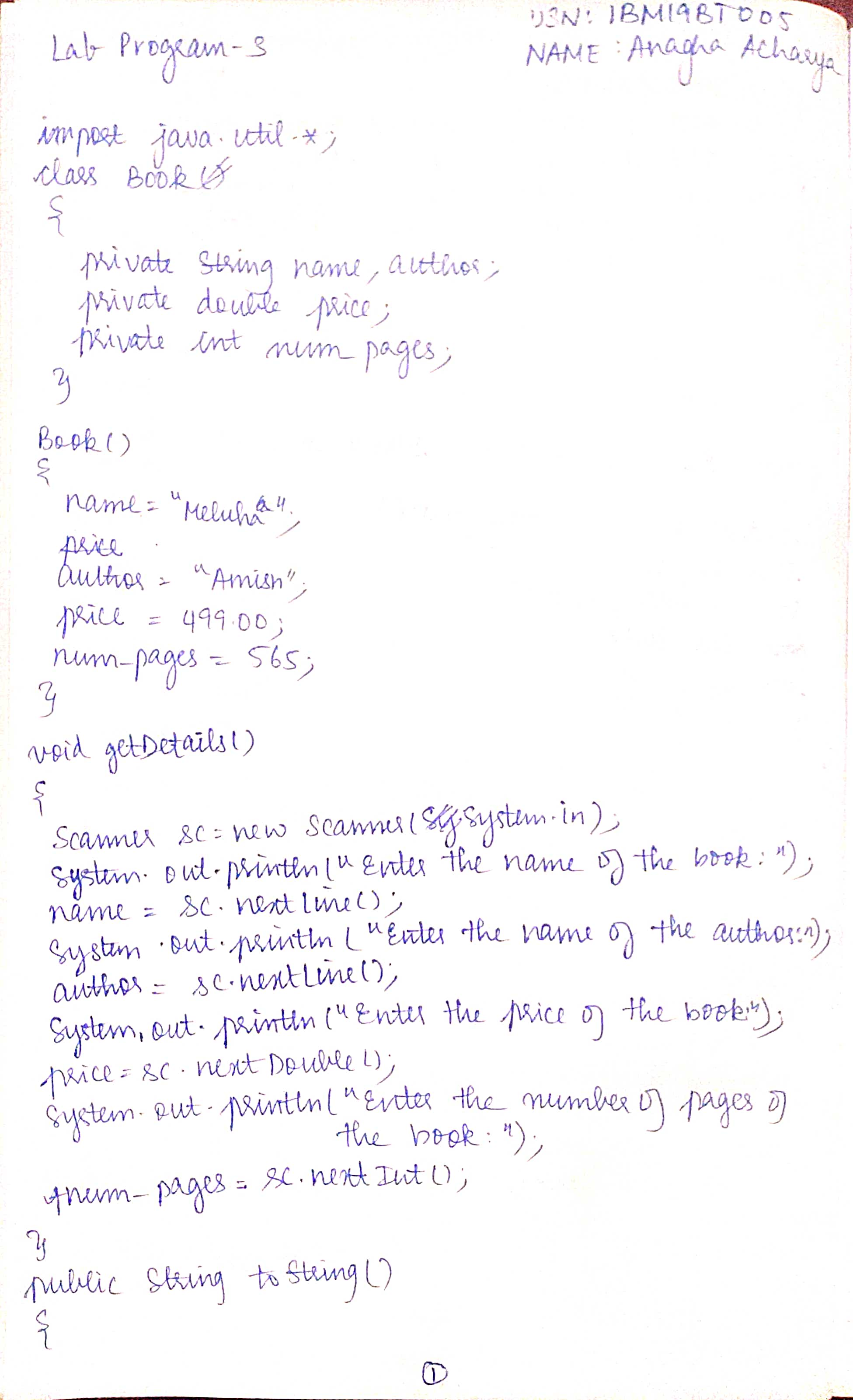
OUTPUT:

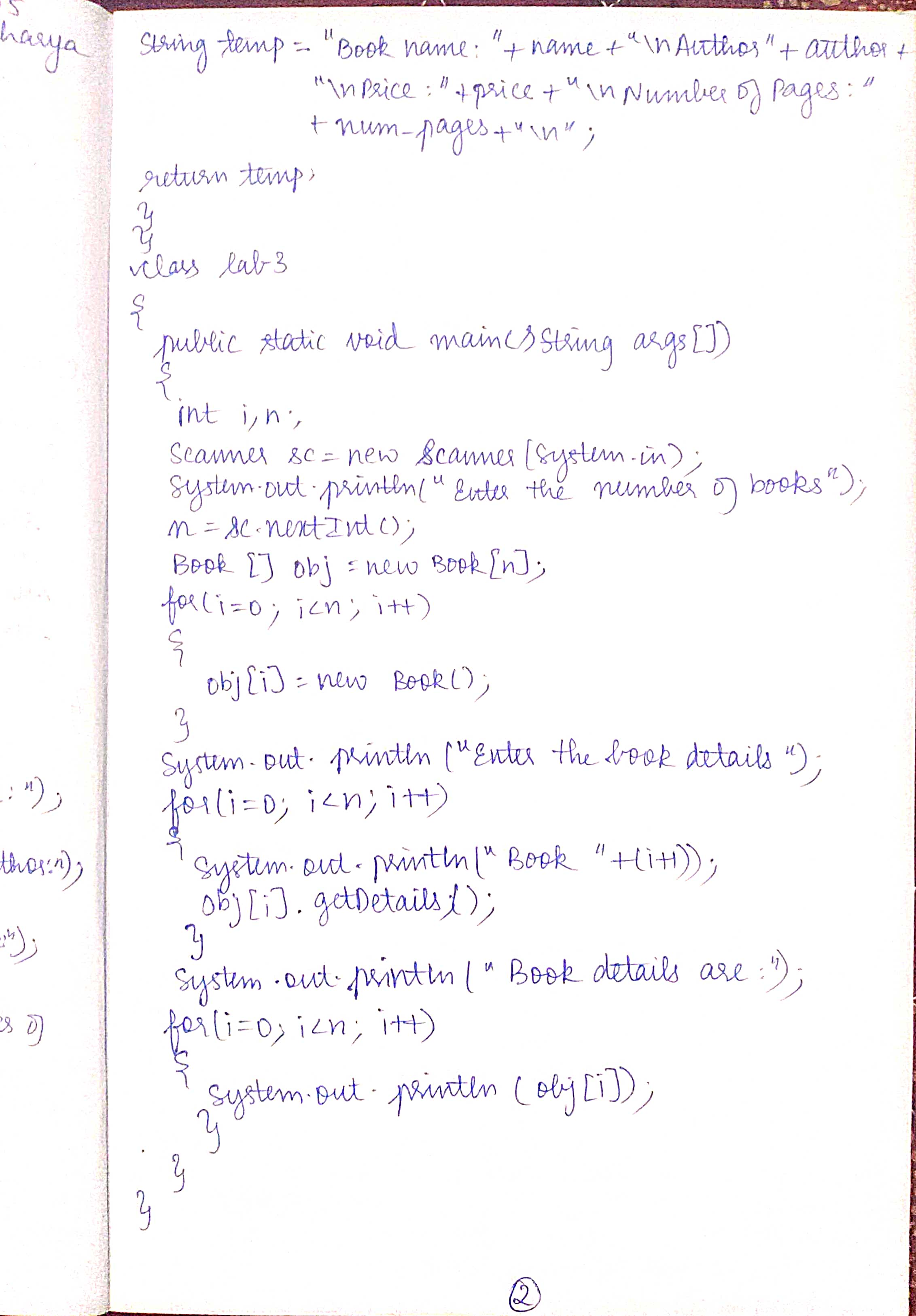


Lab program 3

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.

OBSERVATION:





PROGRAM:

import java.util.\*;

class Book

{

private String name,author;

private double price;

private int num\_pages;

Book()

{

name="Meluha";

author="Amish";

price=499.00;

num\_pages=565;

}

void getDetails()

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter the name of the book:");

name=sc.nextLine();

System.out.println("Enter the author of the book:");

author=sc.nextLine();

System.out.println("Enter the price of the book:");

price=sc.nextDouble();

System.out.println("Enter the number of pages of the book:");

num\_pages=sc.nextInt();

}

public String toString()

{

String temp="Book name:"+name+"\nAuthor:"+author+"\nPrice:"+price+"\nNumber of pages:"+num\_pages+"\n";

return (temp);

}

}

class lab3

{

public static void main(String args[])

{

int i,n;

Scanner sc=new Scanner(System.in);

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

n=sc.nextInt();

Book[] obj=new Book[n];

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

{

obj[i]=new Book();

}

System.out.println("Enter the book details");

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

{

System.out.println("Book"+(i+1)+":");

obj[i].getDetails();

}

System.out.println("Book details are");

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

{

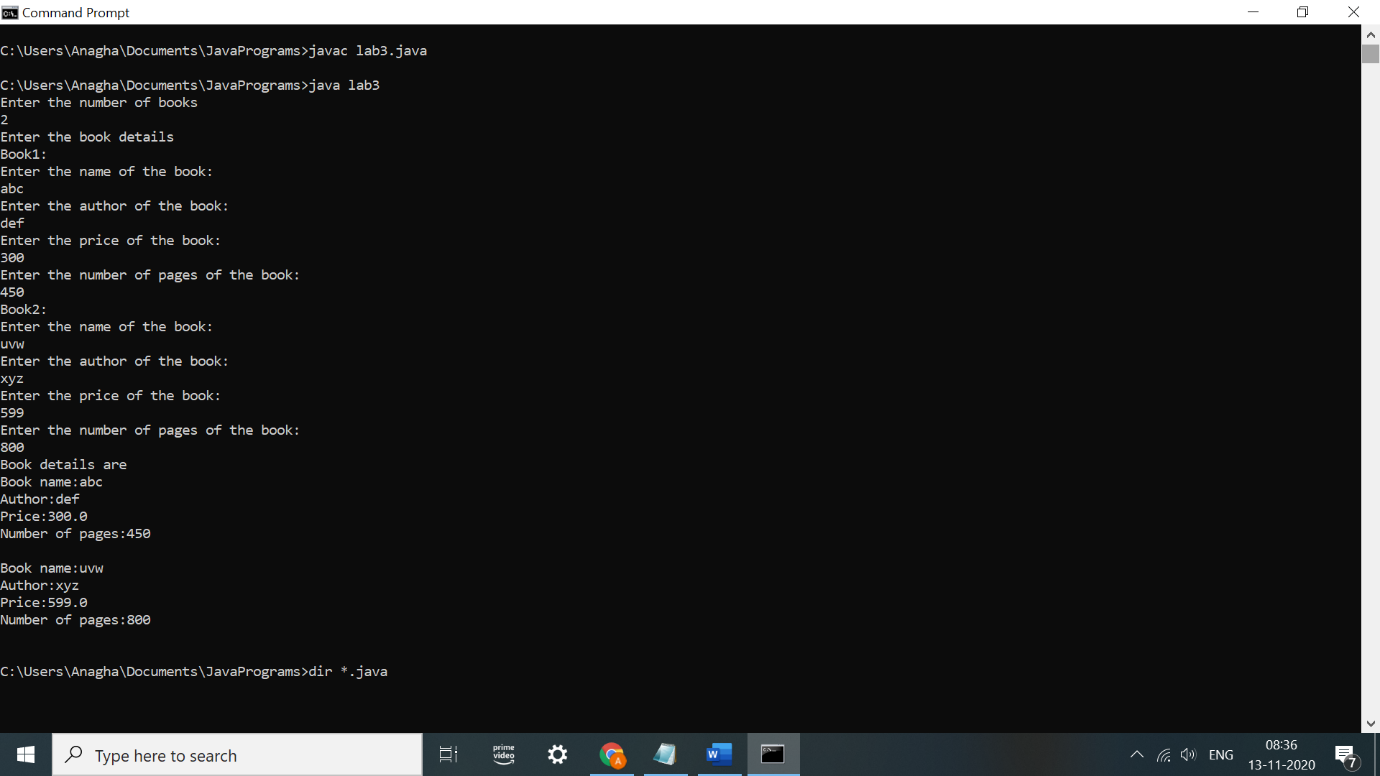
System.out.println(obj[i]);

}

}

}

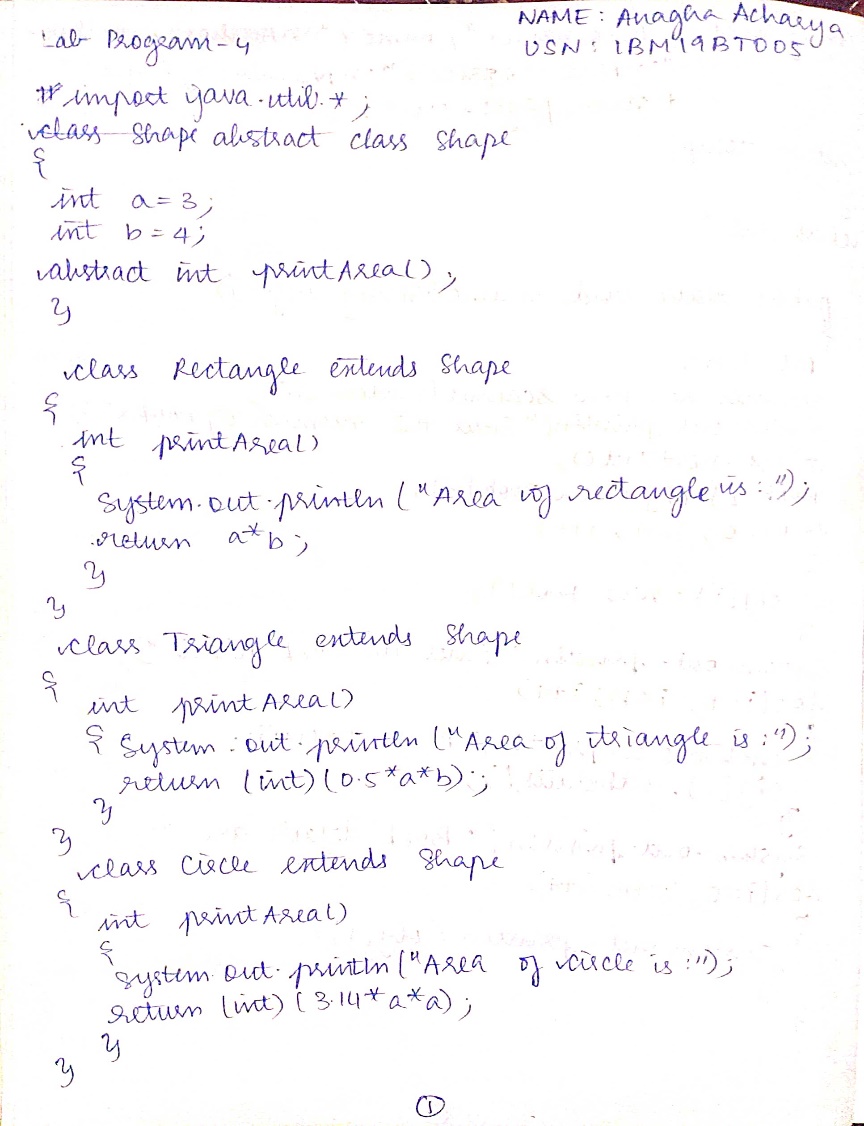
OUTPUT

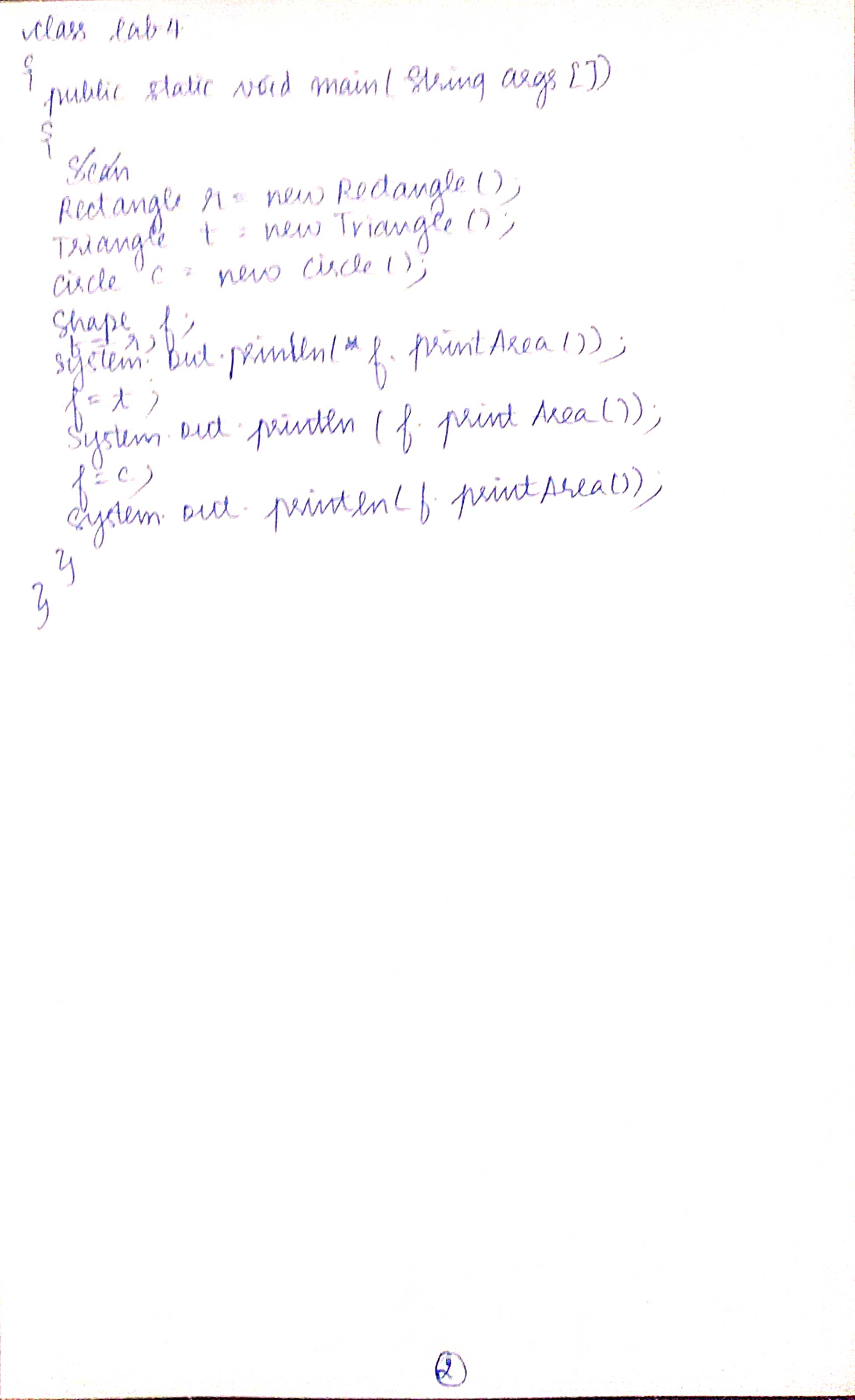


Lab program 4

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.

OBSERVATION:





PROGRAM:

import java.util.\*;

abstract class Shape

{

int a=3;

int b=4;

abstract int printArea();

}

class Rectangle extends Shape

{

int printArea()

{

System.out.println("Area of rectangle is:");

return a\*b;

}

}

class Triangle extends Shape

{

int printArea()

{

System.out.println("Area of triangle is:");

return (int)(0.5\*a\*b);

}

}

class Circle extends Shape

{

int printArea()

{

System.out.println("Area of circle is:");

return (int)(3.14\*a\*a);

}

}

class lab4

{

public static void main(String args[])

{

Rectangle r=new Rectangle();

Triangle t=new Triangle();

Circle c=new Circle();

Shape f;

f=r;

System.out.println(f.printArea());

f=t;

System.out.println(f.printArea());

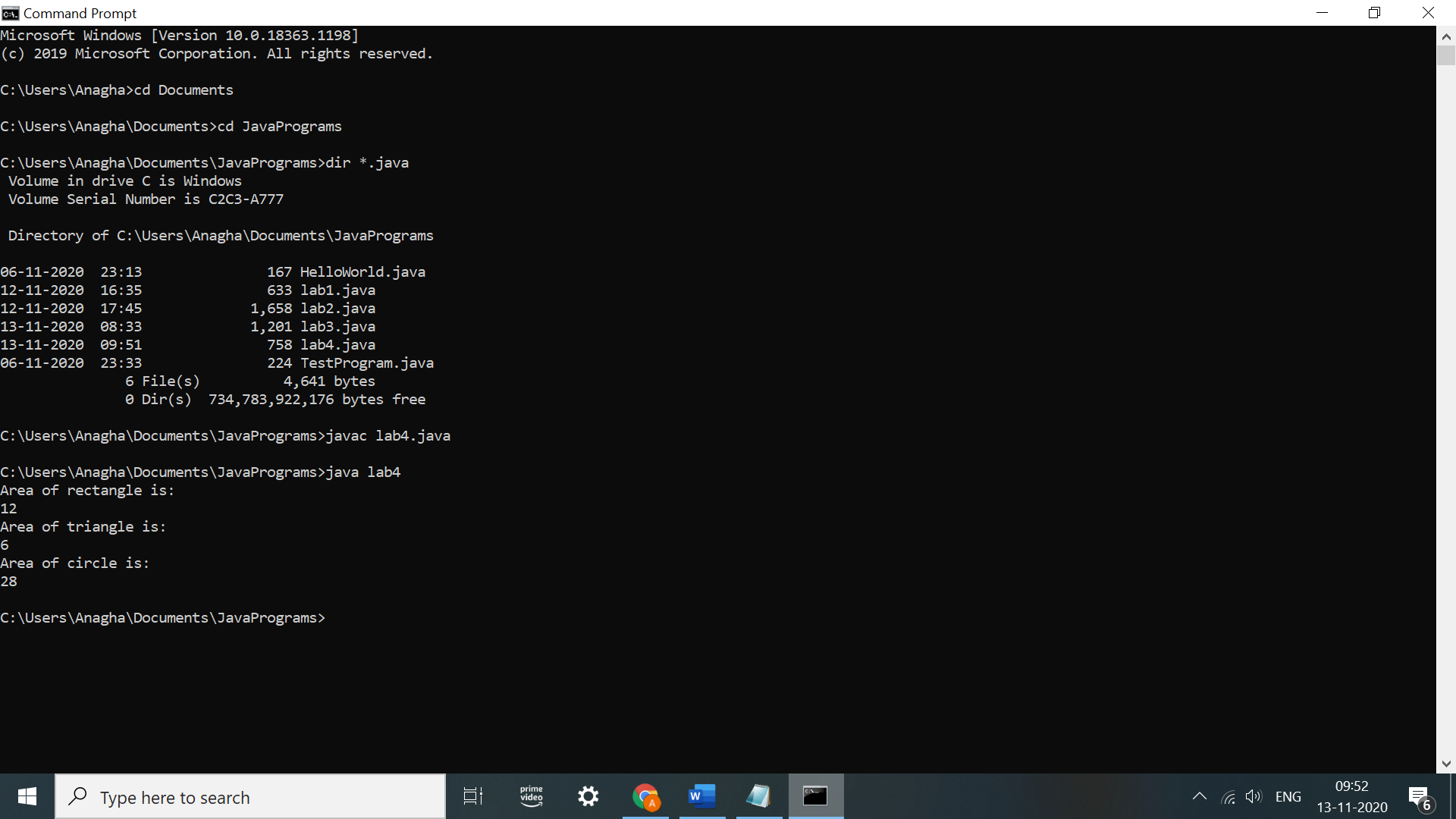
f=c;

System.out.println(f.printArea());

}

}

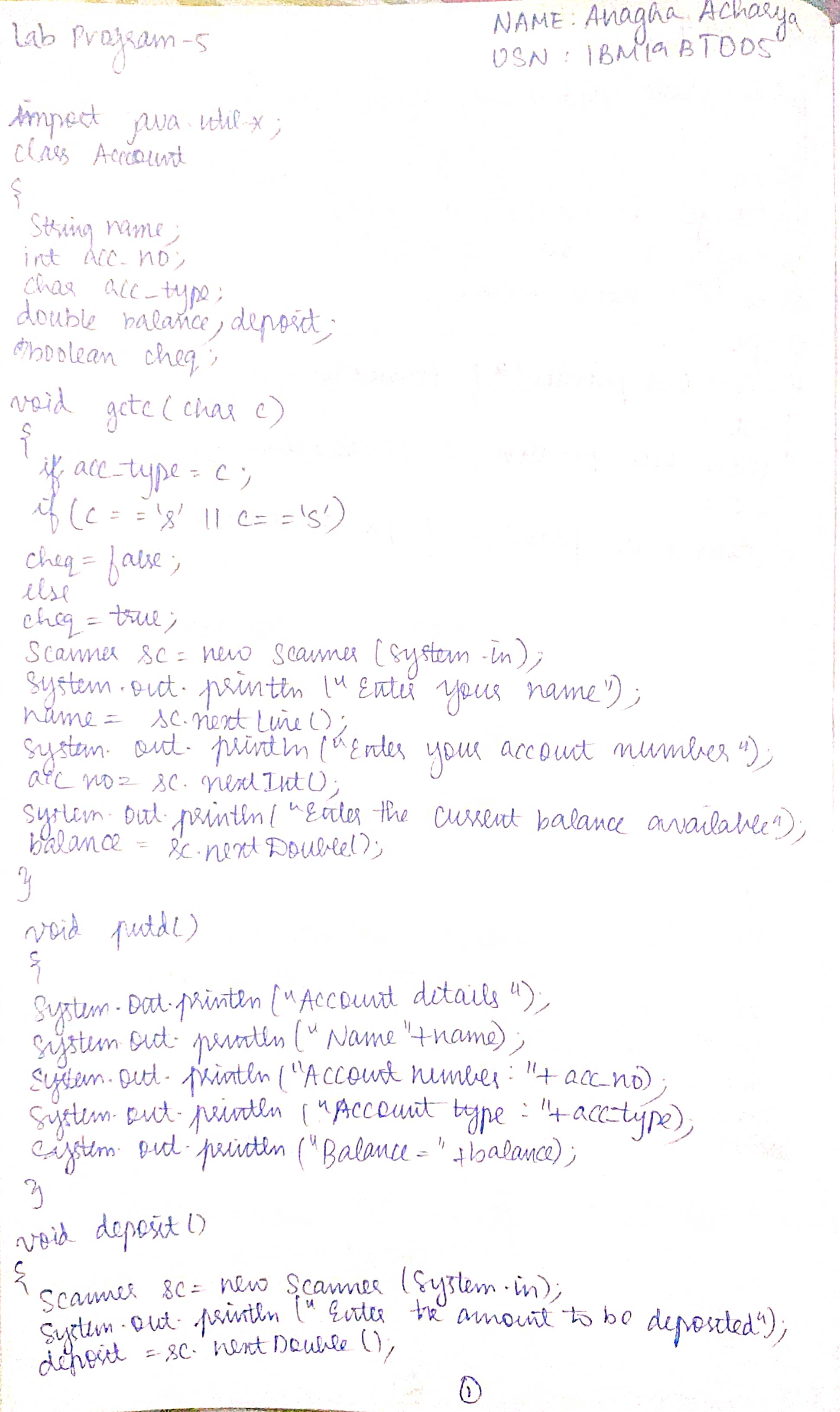
OUTPUT:

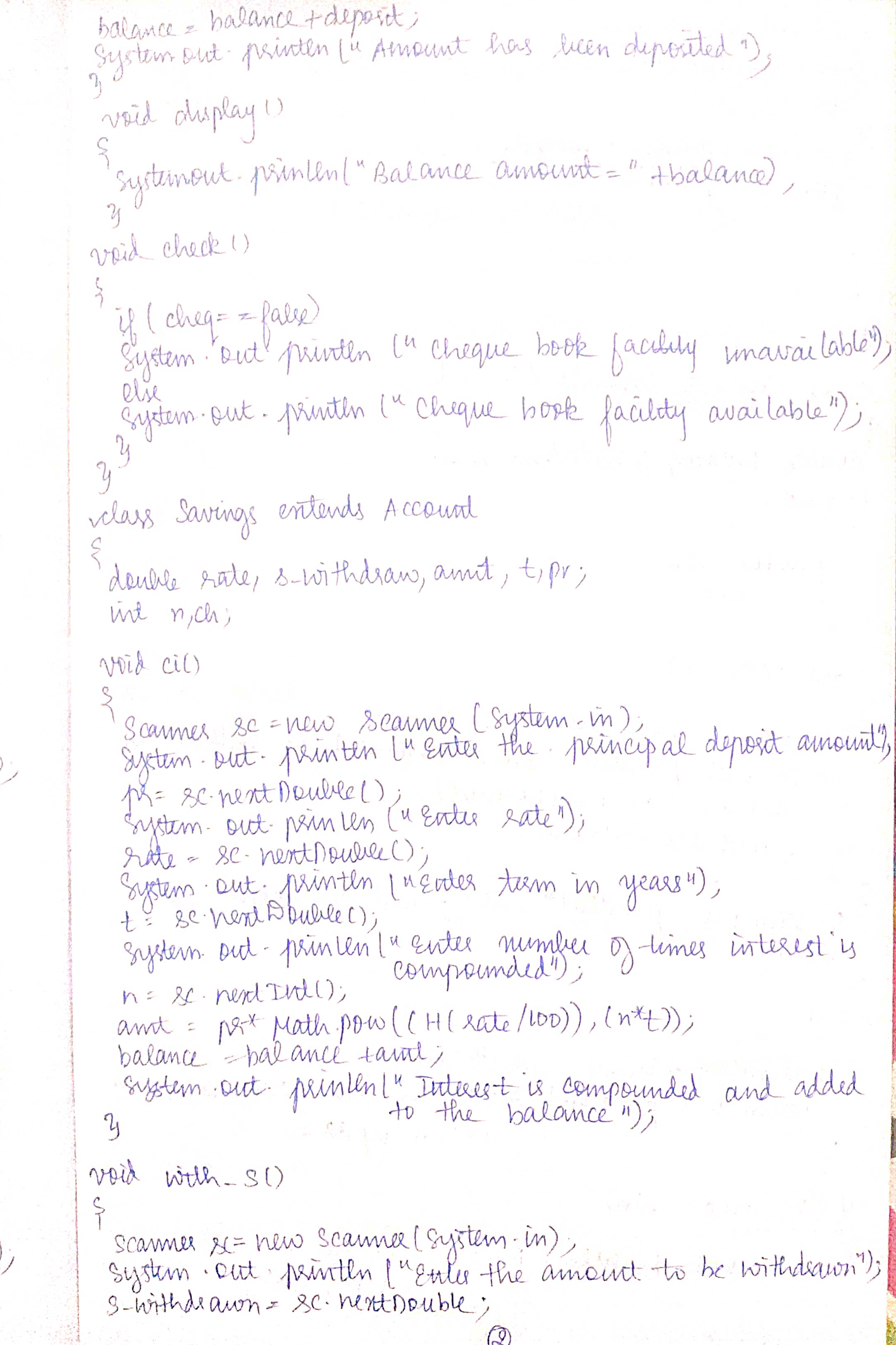


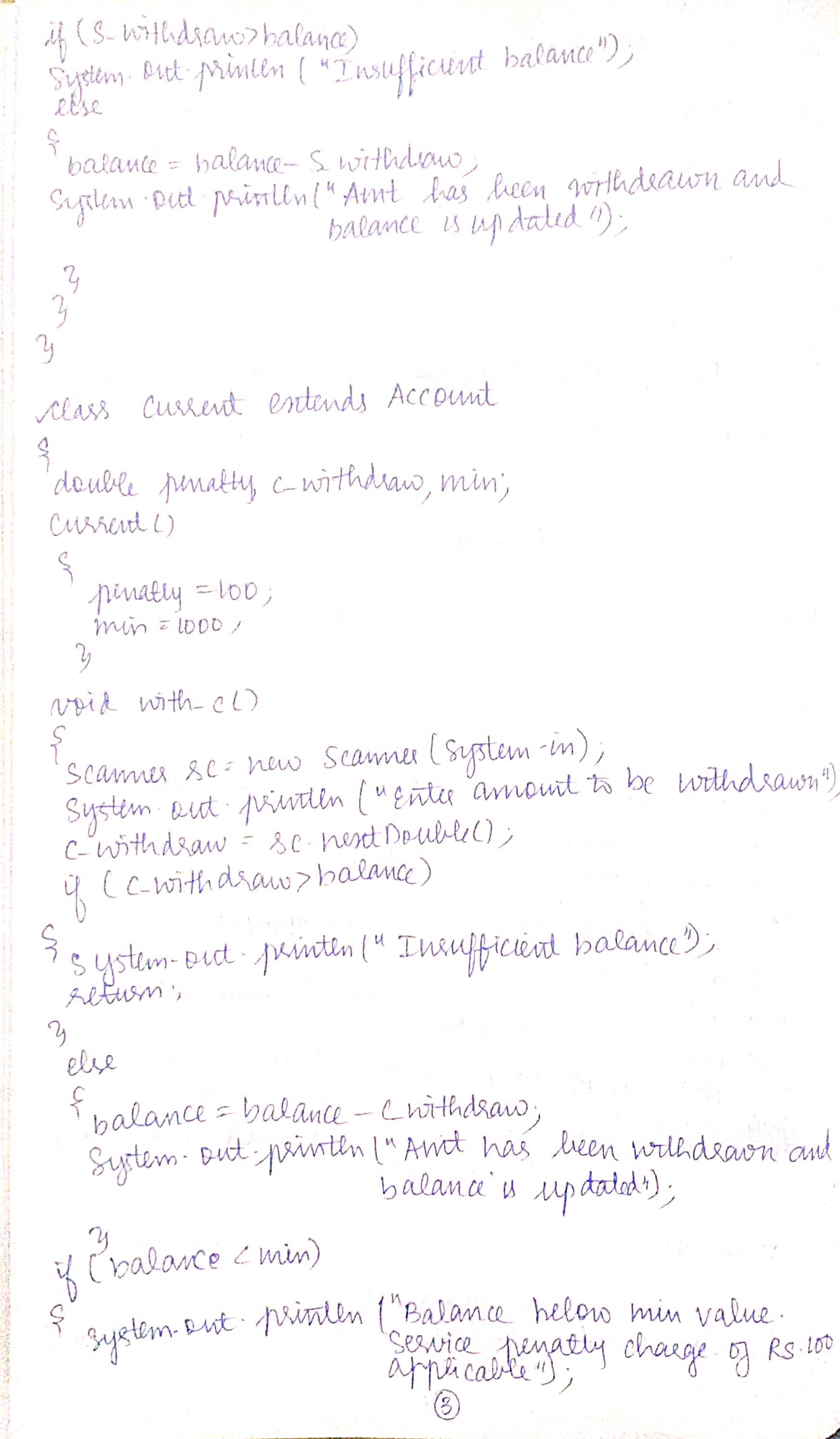
Lab program 5

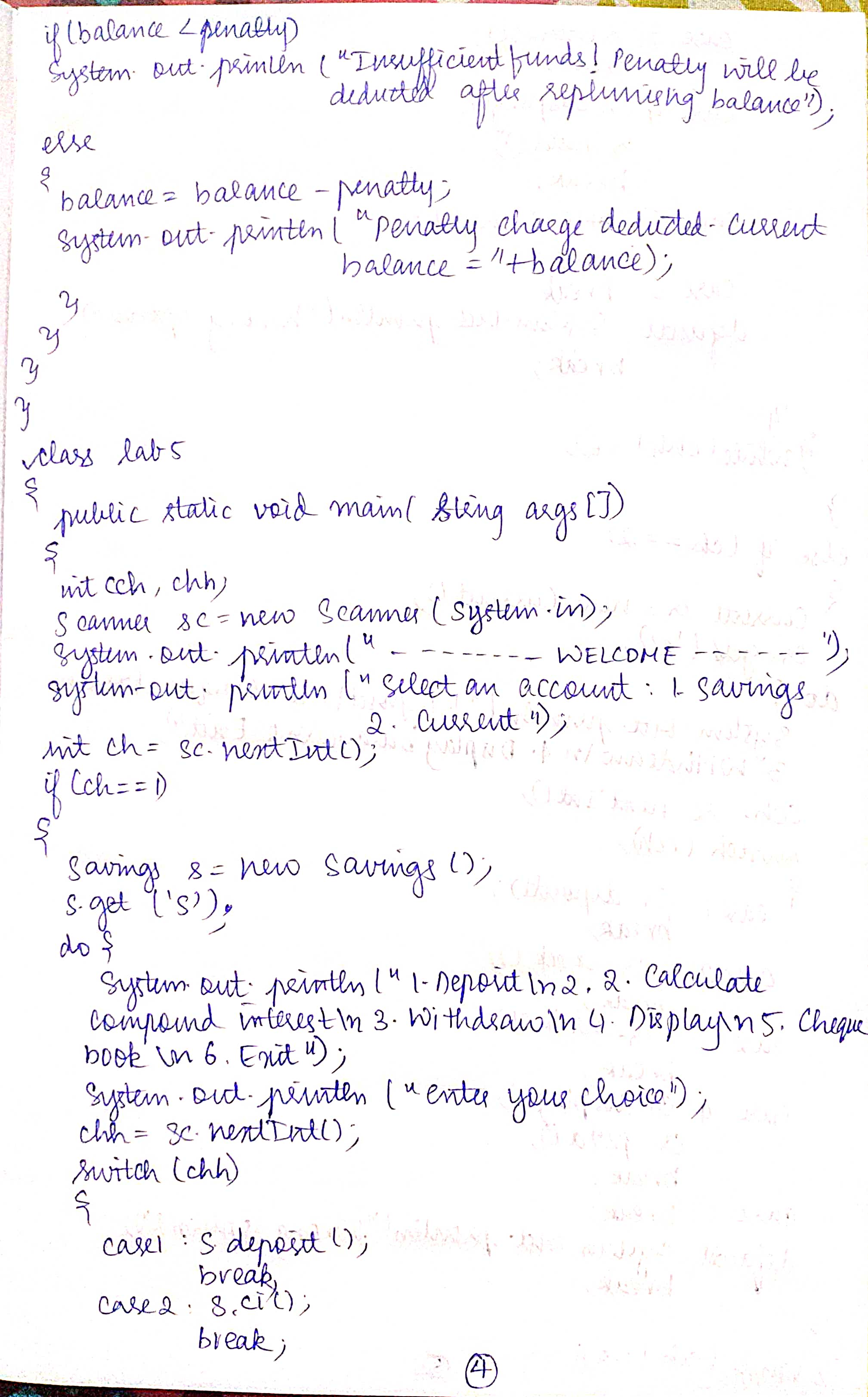
Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book 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: 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.

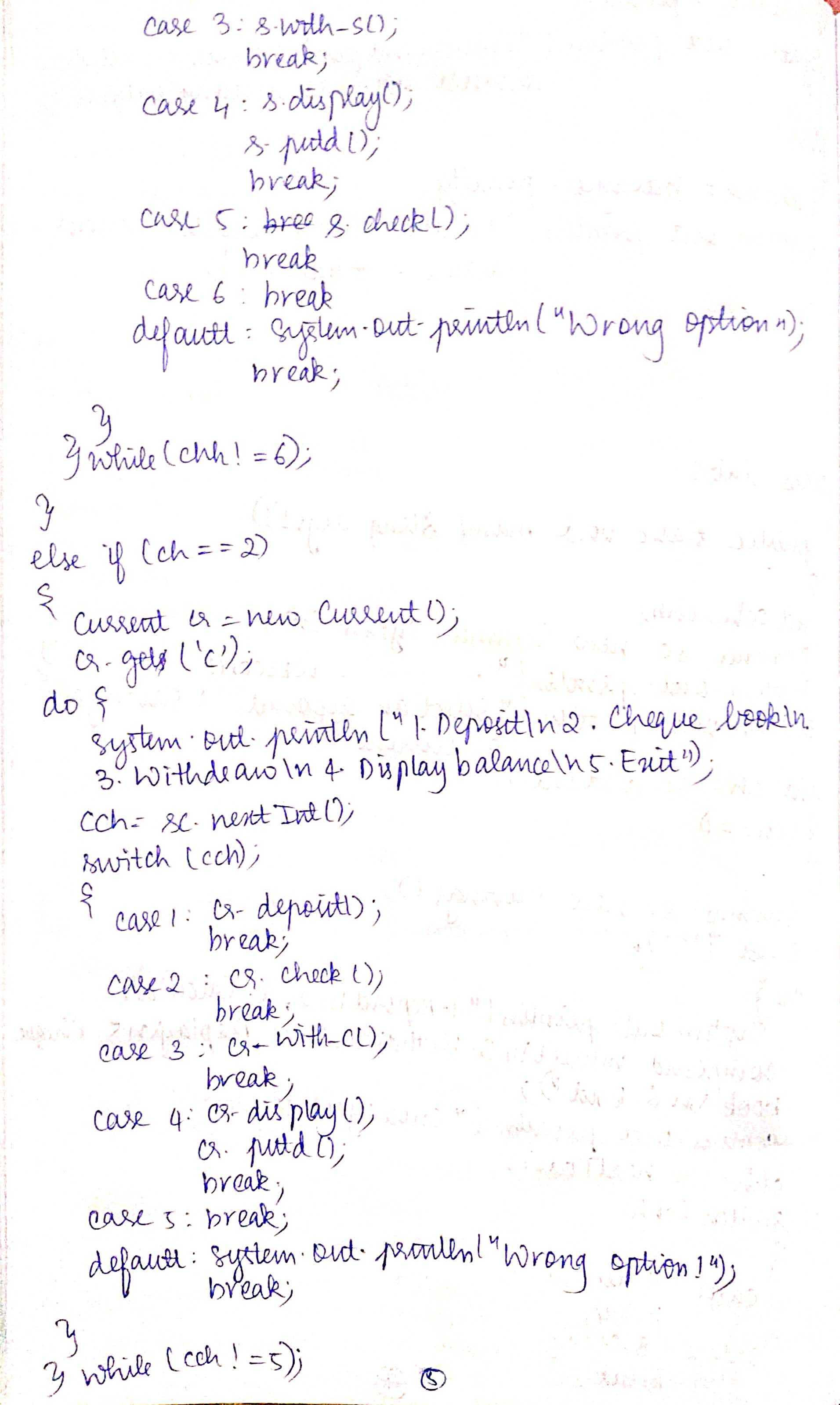
OBSERVATION:













PROGRAM:

import java.util.\*;

class Account

{

String name;

int acc\_no;

char acc\_type;

double balance;

double deposit;

boolean cheq;

void get(char c)

{

acc\_type=c;

if(c=='s' || c=='S')

cheq=false;

else

cheq=true;

Scanner sc=new Scanner(System.in);

System.out.println("Enter your name");

name=sc.nextLine();

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

acc\_no=sc.nextInt();

System.out.println("Enter the current balance available");

balance=sc.nextDouble();

}

void putd()

{

System.out.println("Account details");

System.out.println("Name:"+name);

System.out.println("Account number:"+acc\_no);

System.out.println("Account type:"+acc\_type);

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

}

void deposit()

{

Scanner sc=new Scanner(System.in);

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

deposit=sc.nextDouble();

balance=balance+deposit;

System.out.println("Amount has been deposited");

}

void display()

{

System.out.println("Balance amount="+balance);

}

void check()

{

if(cheq==false)

System.out.println("Cheque book facility is unavailable");

else

System.out.println("Cheque book facility available");

}

}

class Savings extends Account

{

double rate,s\_withdraw,amt,t,pr;

int n,ch;

void ci()

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter the principal deposit amount");

pr=sc.nextDouble();

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

rate=sc.nextDouble();

System.out.println("Enter the term in years");

t=sc.nextDouble();

System.out.println("Enter the number of times the intest is compounded");

n=sc.nextInt();

amt=pr\*Math.pow((1+(rate/100)),(n\*t));

balance=balance+amt;

System.out.println("Interest is compounded and added to the balance");

}

void with\_s()

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter the amount to be withdrawn");

s\_withdraw=sc.nextDouble();

if(s\_withdraw>balance)

System.out.println("Insufficient balance");

else

{

balance=balance-s\_withdraw;

System.out.println("Amount has been withdrawn and balance is updated");

}

}

}

class Current extends Account

{

double penalty,c\_withdraw,min;

Current()

{

penalty=100;

min=1000;

}

void with\_c()

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter the amount to be withdrawn");

c\_withdraw=sc.nextDouble();

if(c\_withdraw>balance)

{

System.out.println("Insufficient balance");

return;

}

else

{

balance=balance-c\_withdraw;

System.out.println("Amount has been withdrawn and balance is updated");

}

if(balance<min)

{

System.out.println("Balance below the minimum value. Service penalty charge of Rs.100 applicable");

if(balance<penalty)

System.out.println("Insufficient funds!Penalty will be deducted after replenishing balance");

else

{

balance=balance-penalty;

System.out.println("Penalty charge has been deducted. Current balance="+balance);

}

}

}

}

class lab5

{

public static void main(String args[])

{

int cch,chh;

Scanner sc=new Scanner(System.in);

System.out.println("----------WELCOME----------");

System.out.println("Select an account: 1.Savings 2.Current");

int ch=sc.nextInt();

if(ch==1)

{

Savings s=new Savings();

s.get('S');

do{

System.out.println("1.Deposit\n2.Calculate compound interest\n3.Withdraw\n4.Display balance\n5.Cheque book facility\n6.Exit");

System.out.println("enter your choice");

chh=sc.nextInt();

switch(chh)

{

case 1:s.deposit();

break;

case 2:s.ci();

break;

case 3:s.with\_s();

break;

case 4:s.display();

s.putd();

break;

case 5:s.check();

break;

case 6:break;

default:System.out.println("Wrong option!");

break;

}

}while(chh!=6);

}

else if(ch==2)

{

Current cr=new Current();

cr.get('C');

do{

System.out.println("1.Deposit\n2.Cheque book facility\n3.Withdraw\n4.Display balance\n5.Exit");

cch=sc.nextInt();

switch(cch)

{

case 1:cr.deposit();

break;

case 2:cr.check();

break;

case 3:cr.with\_c();

break;

case 4:cr.display();

cr.putd();

break;

case 5:break;

default:System.out.println("Wrong option!");

break;

}

}while(cch!=5);

}

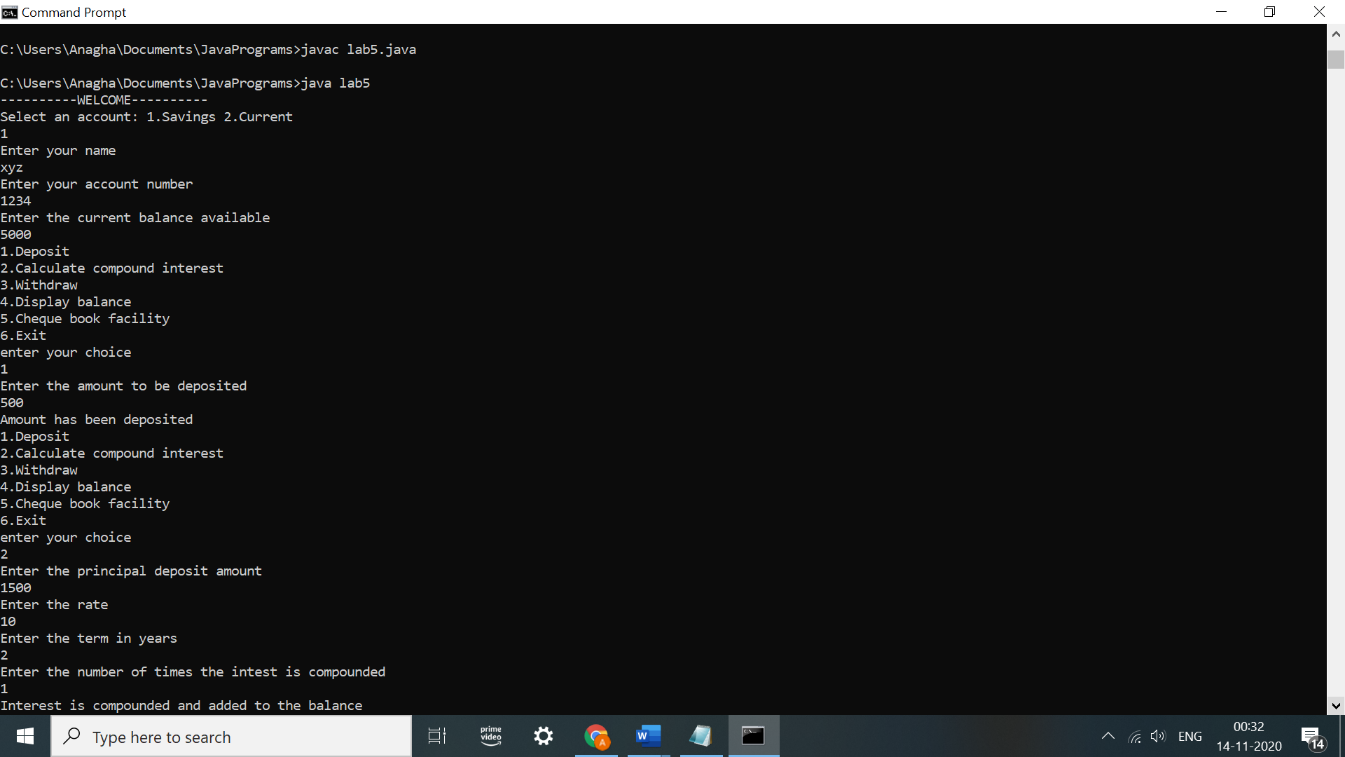
else

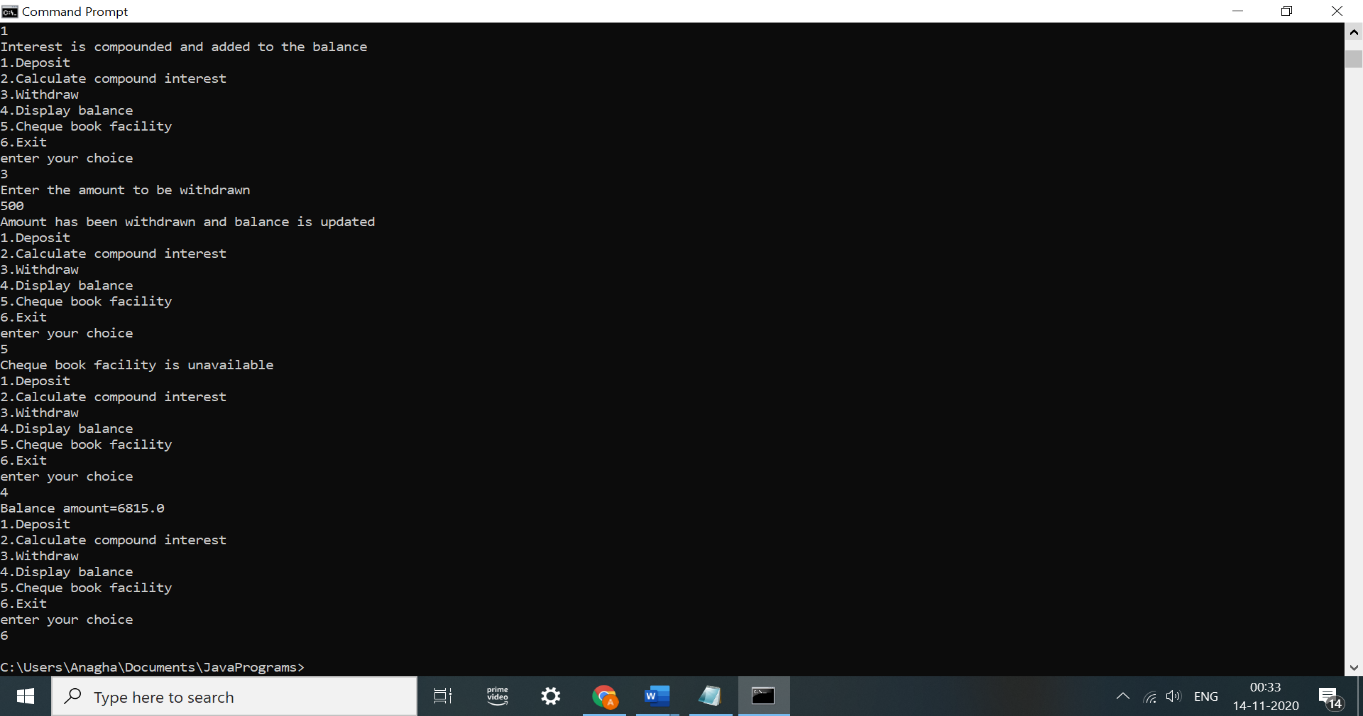
System.out.println("Wrong!");

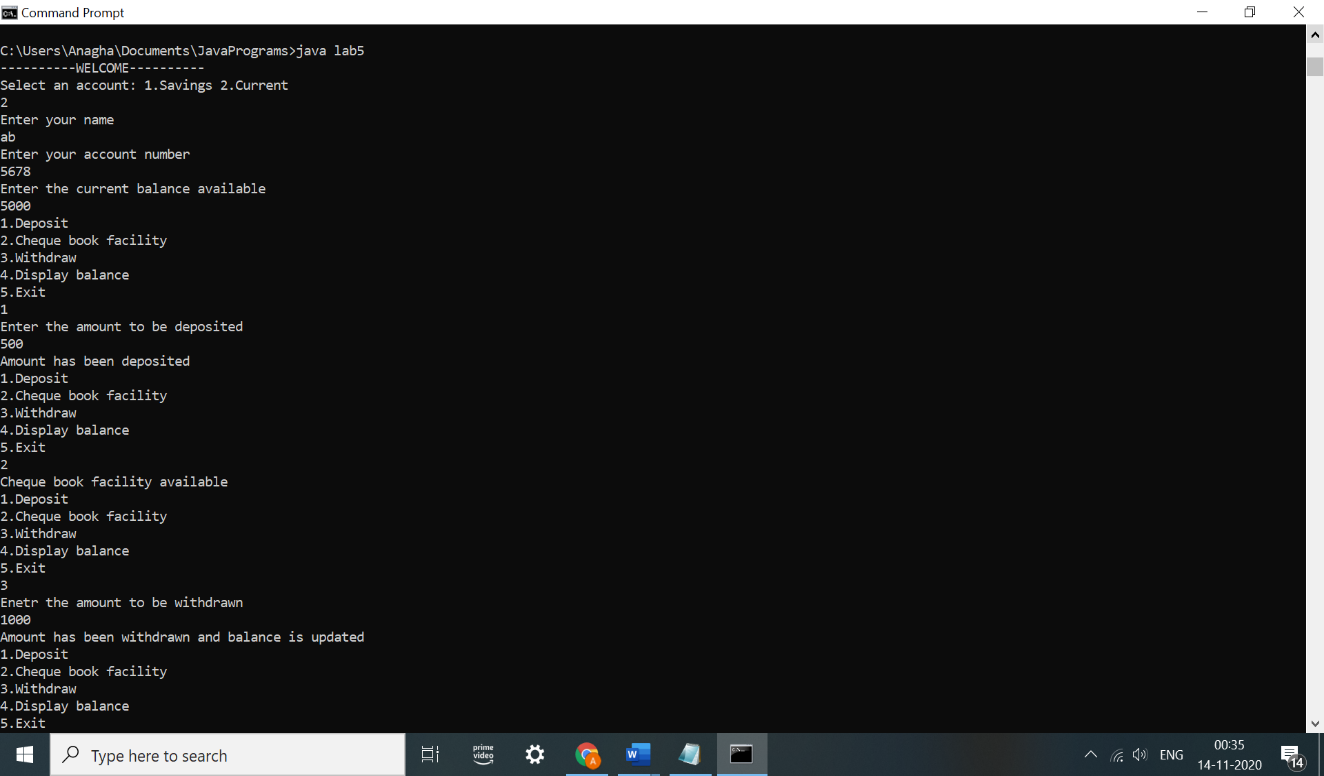
}

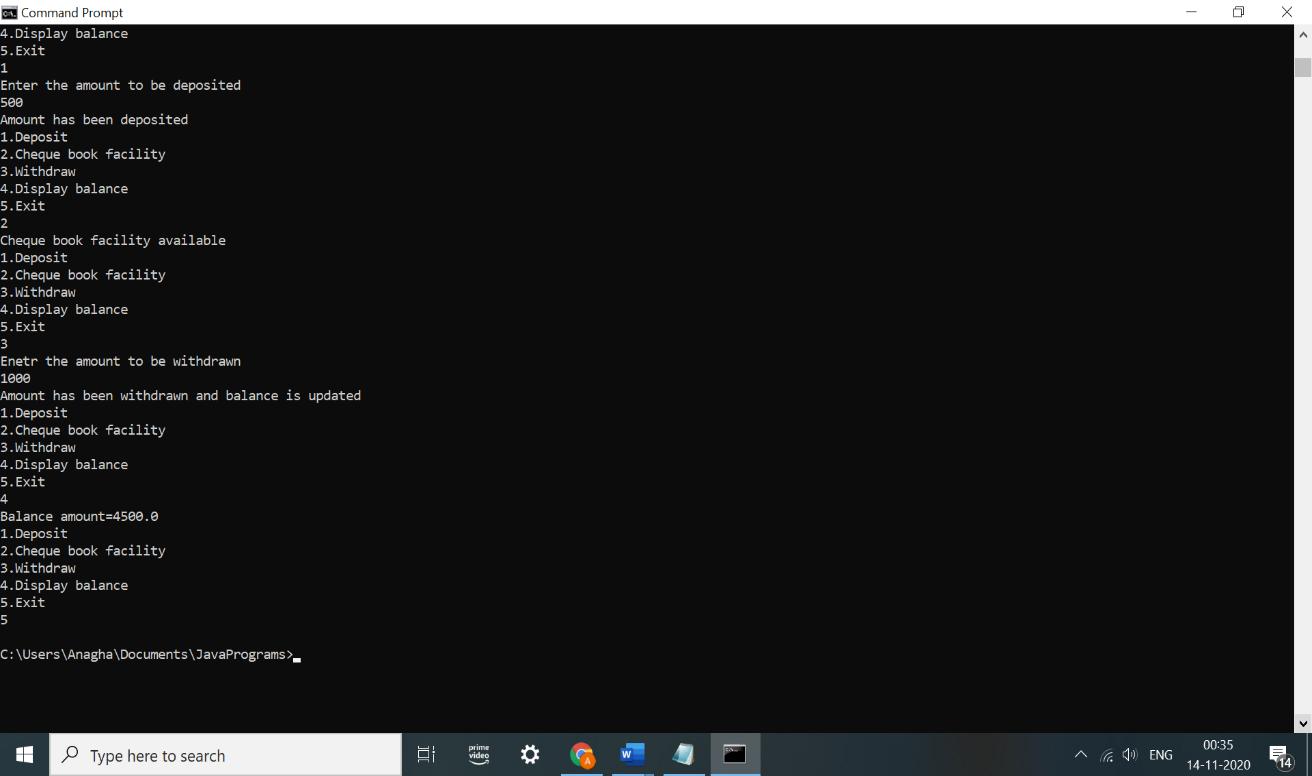
}

OUTPUT:





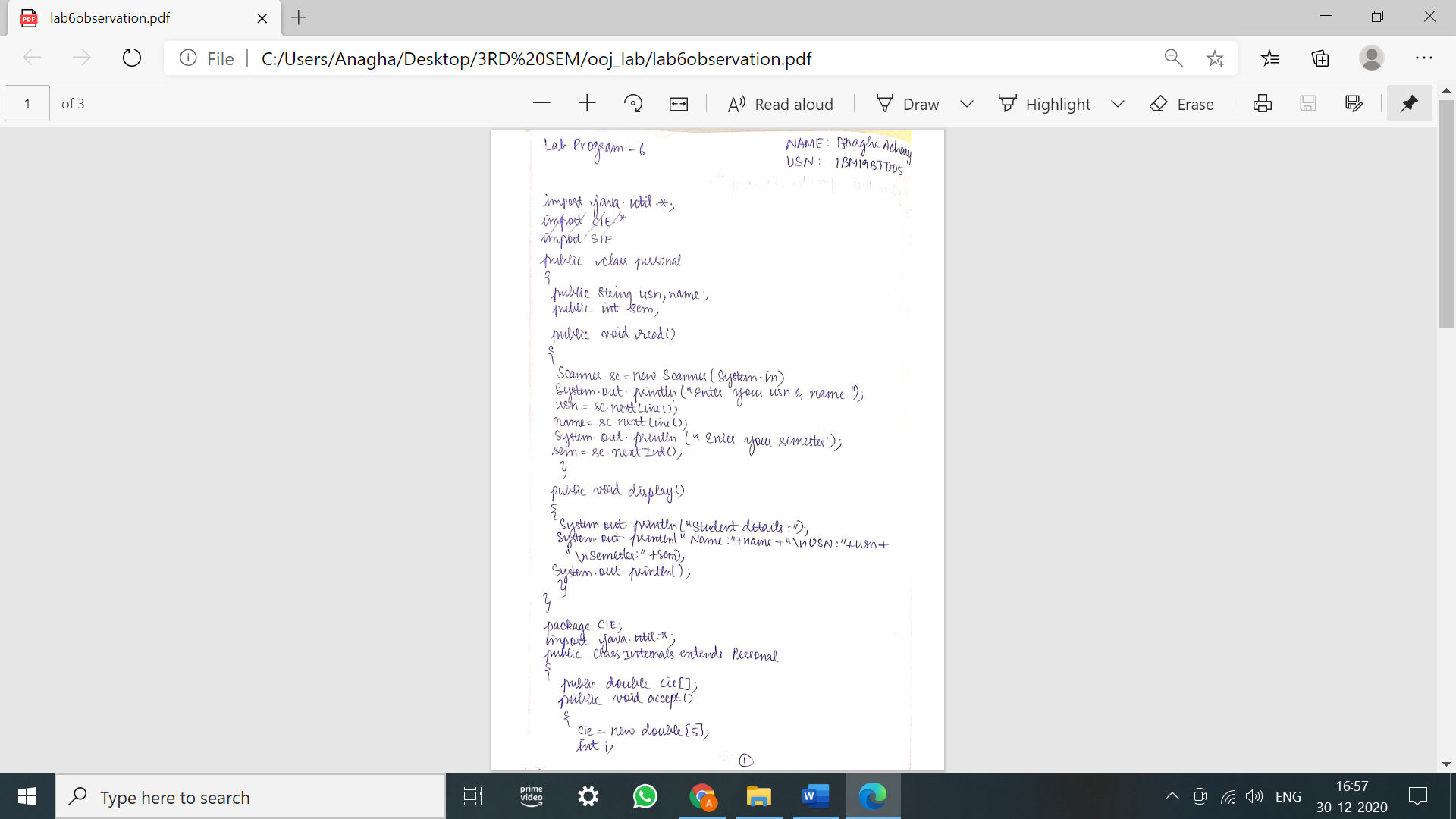




Lab Program-6

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.

OBSERVATION:



PROGRAM:

import java.util.\*;

public class Personal

{

public String usn,name;

public int sem;

public void read()

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter your usn and name");

usn=sc.nextLine();

name=sc.nextLine();

System.out.println("Enter your semester ");

sem=sc.nextInt();

}

public void display()

{

System.out.println("Student details:");

System.out.println("Name:"+name+"\nUSN:"+usn+"\nSemester:"+sem);

System.out.println();

}

}

package CIE;

import java.util.\*;

public class Internals extends Personal

{

public double cie[];

public void accept()

{

cie=new double[5];

int i;

Scanner sc=new Scanner(System.in);

for(i=0;i<5;i++)

{

System.out.println("Enter the cie marks for subject "+(i+1)+":");

cie[i]=sc.nextDouble();

}

}

}

package SEE;

import CIE.\*;

import java.util.\*;

public class External extends Personal

{

public double see[];

public void get()

{

see=new double[5];

int i;

Scanner sc=new Scanner(System.in);

for(i=0;i<5;i++)

{

System.out.println("Enter the see marks for subject "+(i+1)+":");

see[i]=sc.nextDouble();

}

}

}

import java.util.\*;

import CIE.\*;

import SEE.\*;

class lab6

{

public static void main(String args[])

{

Scanner sc=new Scanner(System.in);

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

int n=sc.nextInt();

CIE.Internals in[]=new CIE.Internals[n];

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

int i,j;

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

{

System.out.println("Student "+(i+1));

in[i]=new CIE.Internals();

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

in[i].read();

System.out.println("CIE marks:");

in[i].accept();

System.out.println("SEE marks:");

en[i].get();

System.out.println();

in[i].display();

for(j=0;j<5;j++)

System.out.println("Final total marks for subject "+(j+1)+": "+(in[i].cie[j] + (en[i].see[j]/2)));

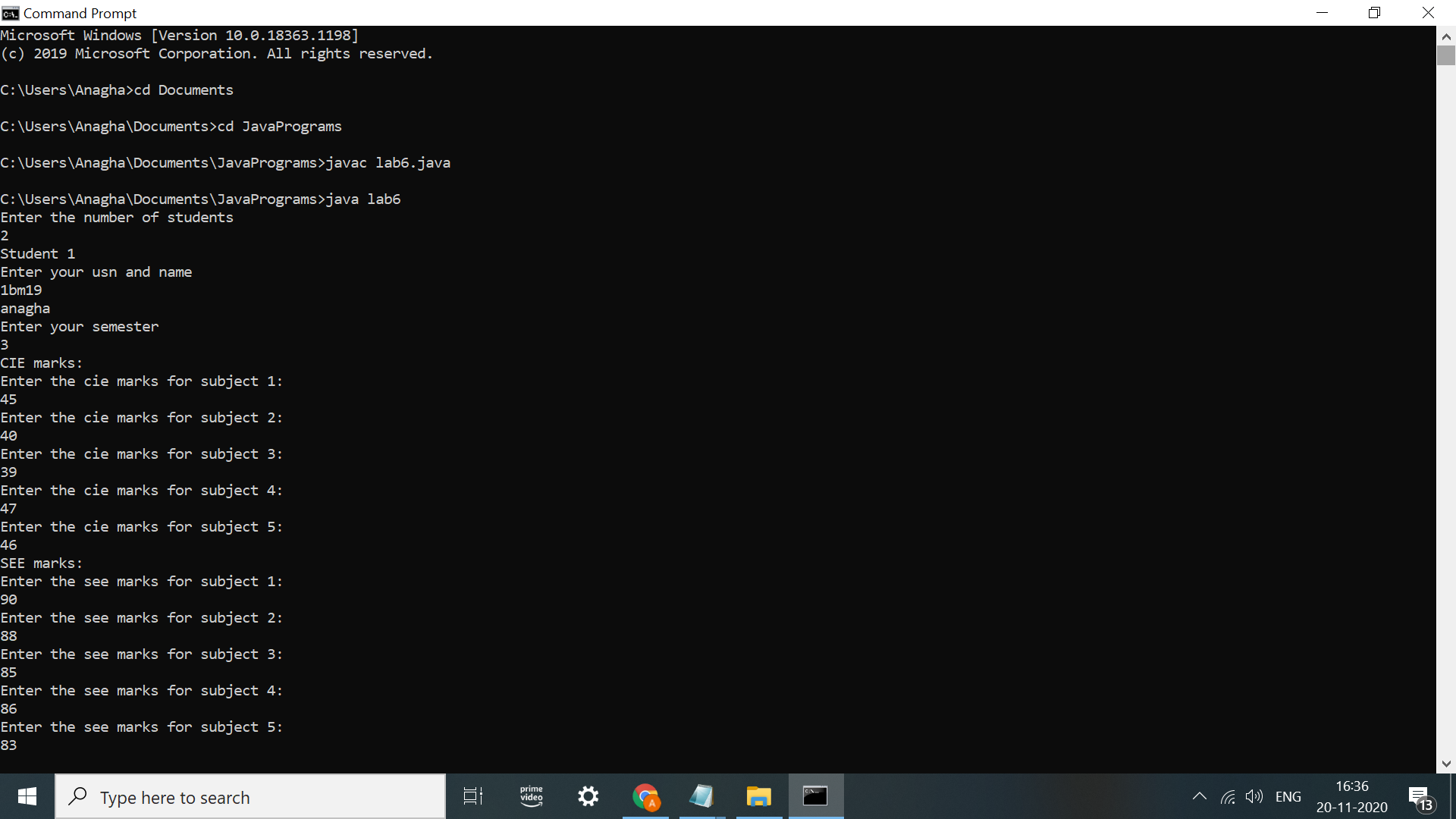
System.out.println();

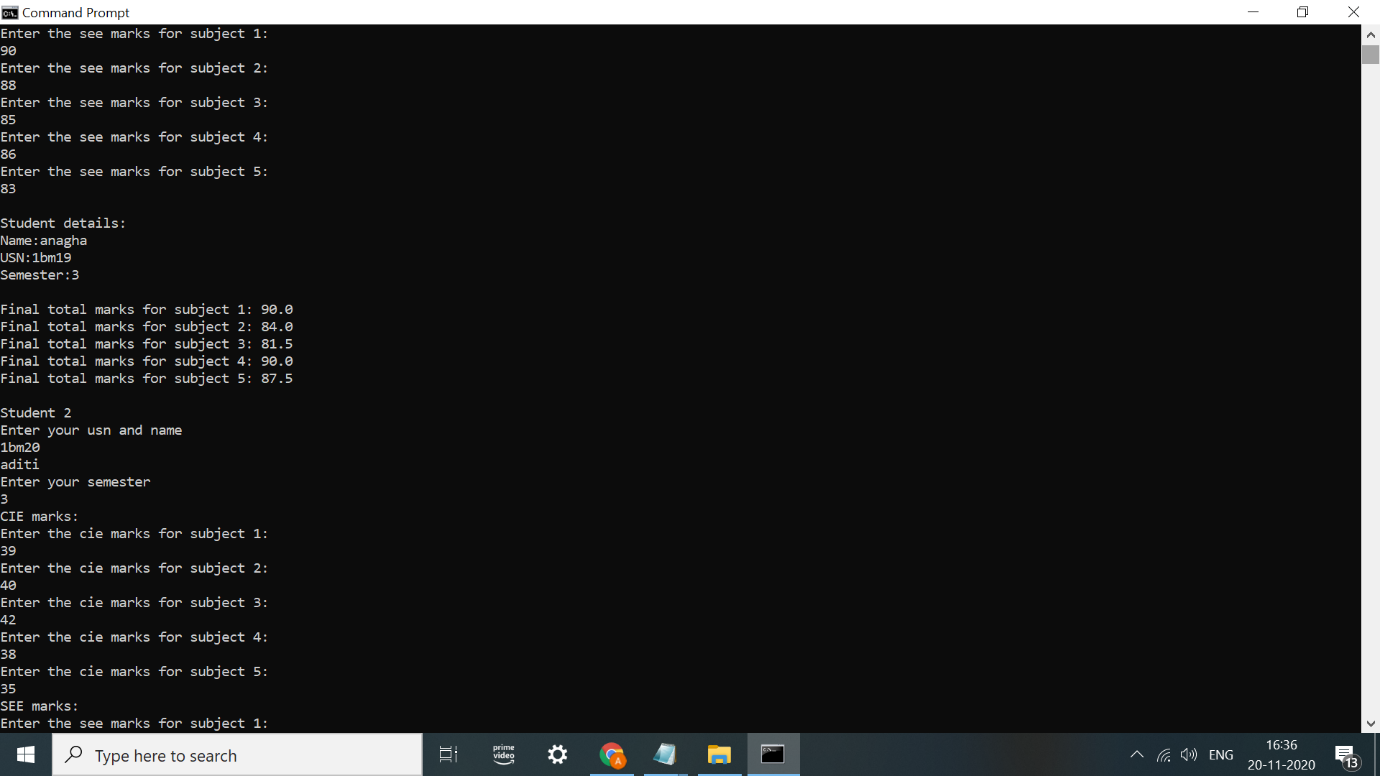
}

}

}

OUTPUT:



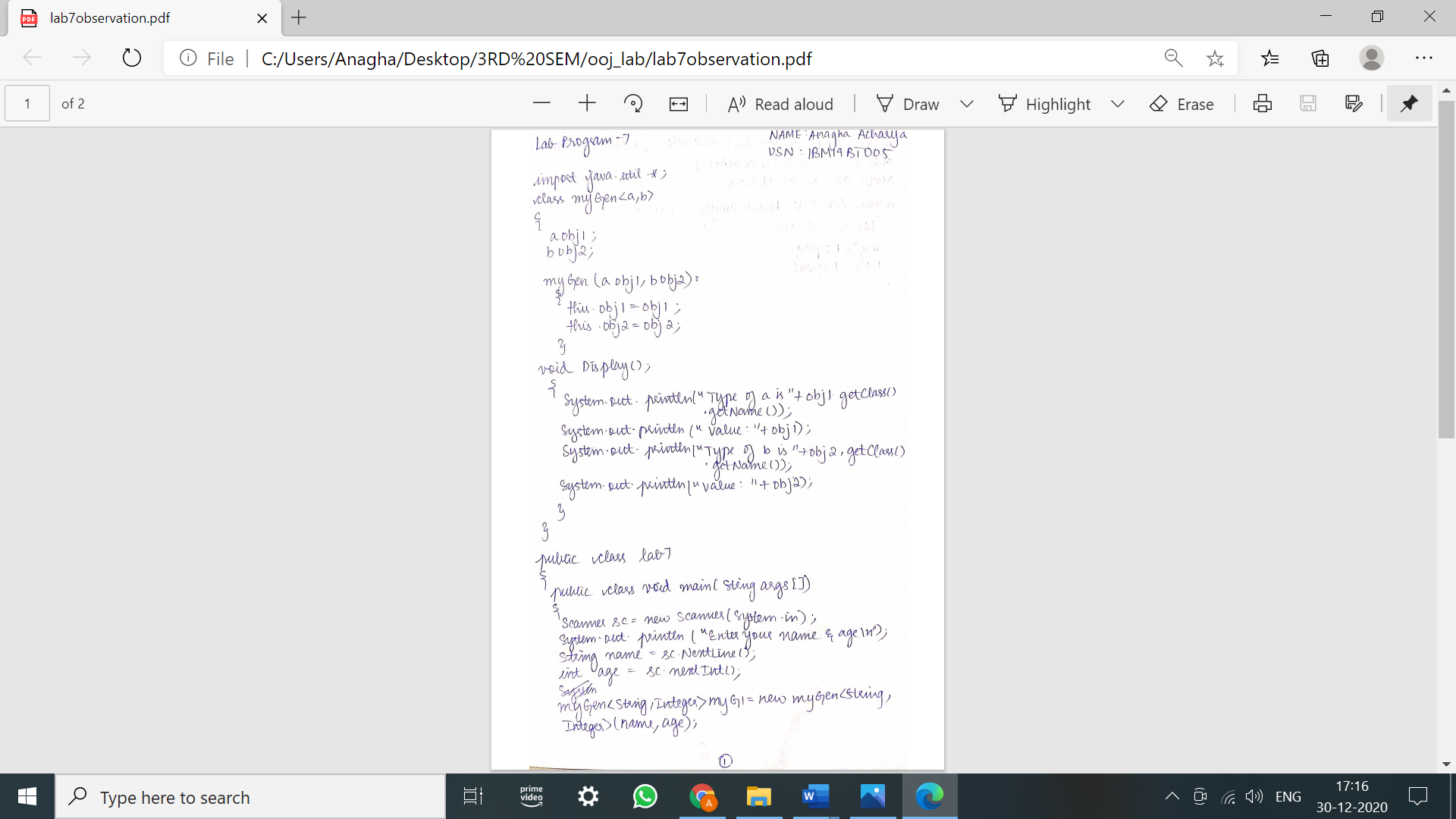


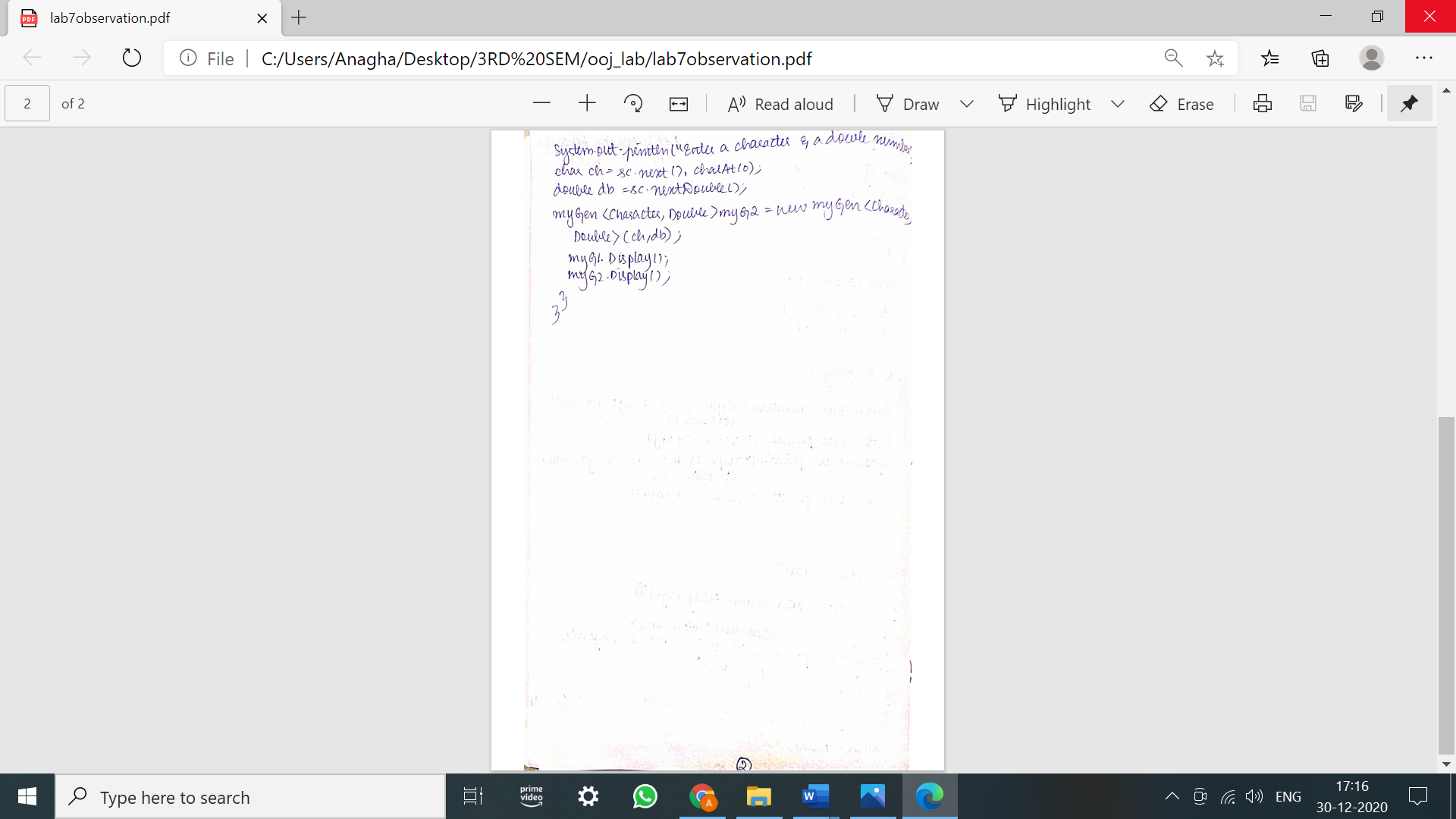


Lab program 7

Write a program to demonstrate generics with multiple object parameters.

OBSERVATION:





PROGRAM:

import java.util.\*;

class myGen<a,b>{

a obj1;

b obj2;

myGen(a obj1, b obj2)

{

this.obj1 = obj1;

this.obj2 = obj2;

}

void Display()

{

System.out.println("Type of a is " +obj1.getClass().getName());

System.out.println("Value: "+obj1);

System.out.println("Type of b is " +obj2.getClass().getName());

System.out.println("Value: "+obj2);

}

}

public class lab7{

public static void main(String args[])

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter your name and age\n");

String name=sc.nextLine();

int age=sc.nextInt();

myGen<String,Integer>myG1 = new myGen<String,Integer>(name,age);

System.out.println("Enter a character and a double number\n");

char ch=sc.next().charAt(0);

double db=sc.nextDouble();

myGen<Character,Double>myG2 = new myGen<Character,Double>(ch,db);

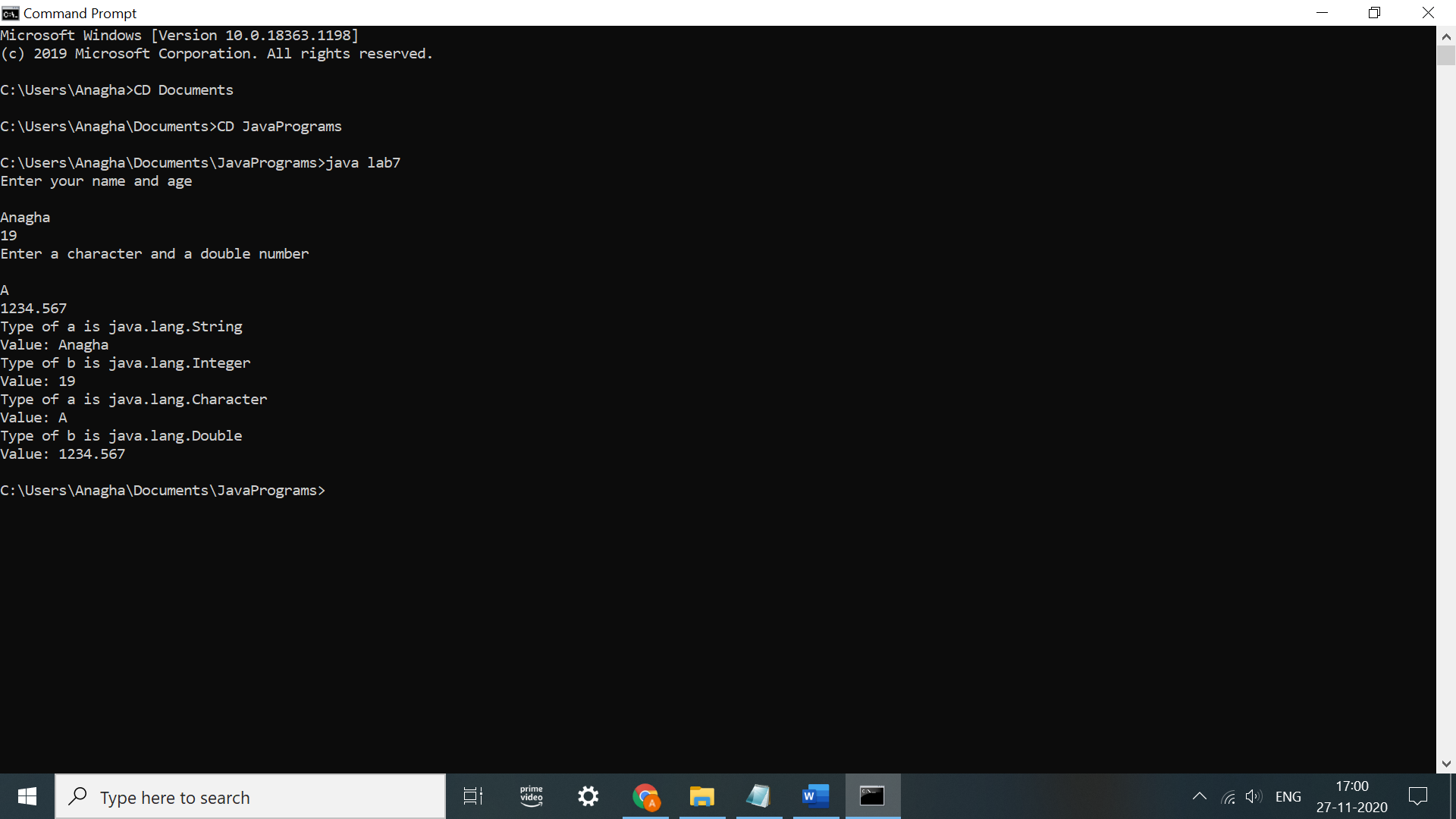
myG1.Display();

myG2.Display();

}

}

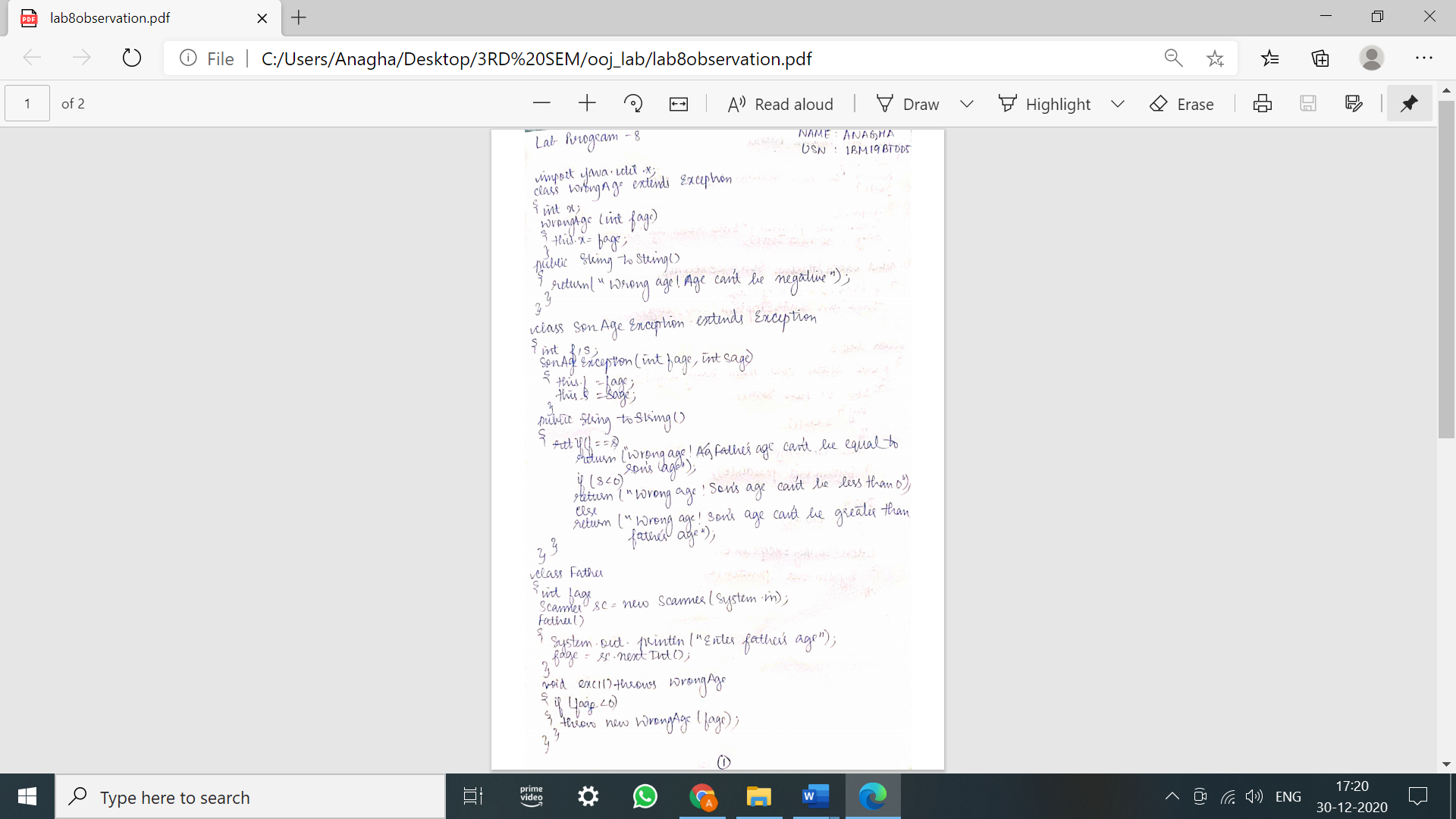
OUTPUT:

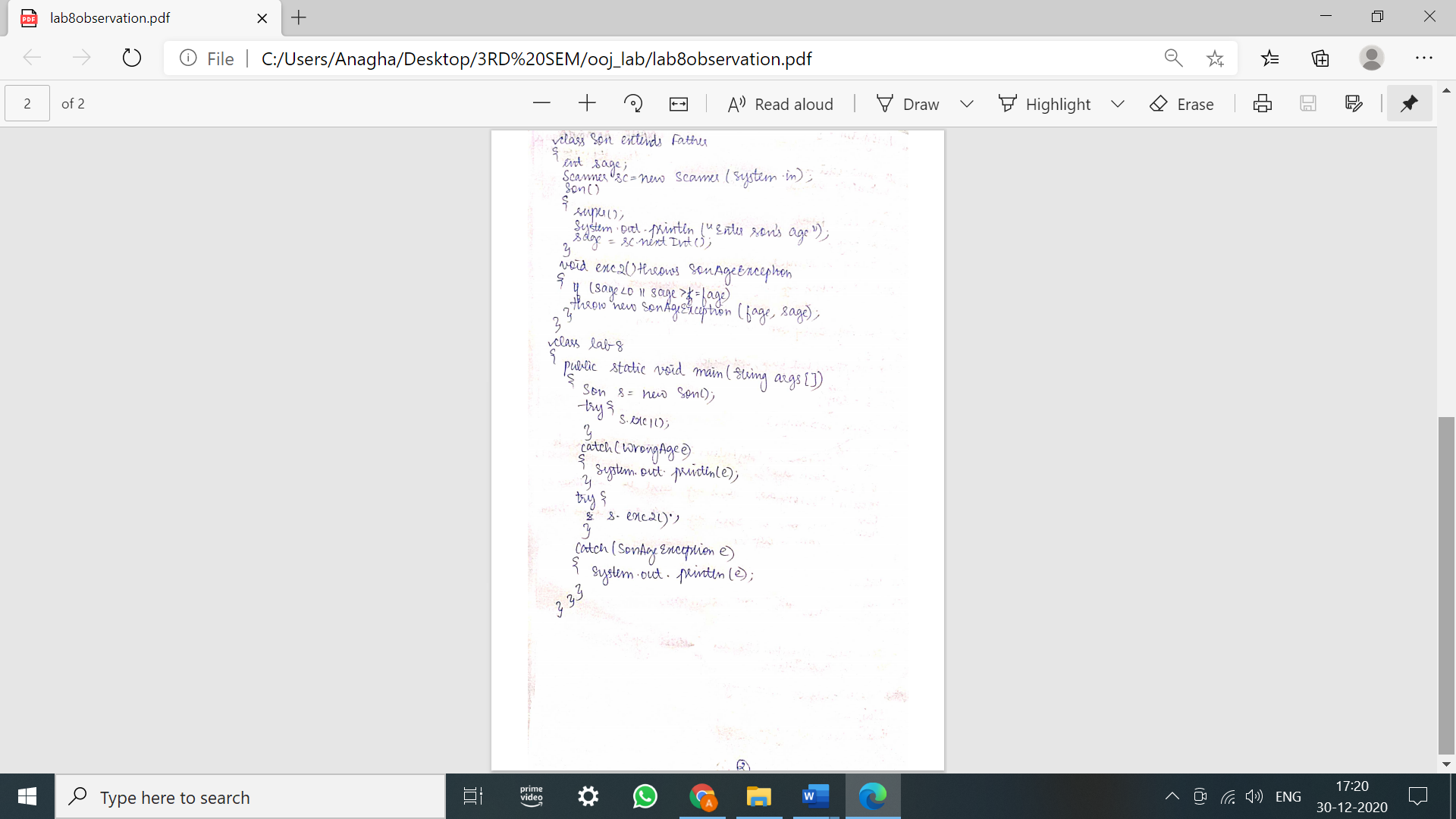


LAB PROGRAM 8

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 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.

OBSERVATION:





PROGRAM:

import java.util.\*;

class WrongAge extends Exception

{

int x;

WrongAge(int fage)

{

this.x=fage;

}

public String toString()

{

return("Wrong age!Father's age can't be negative");

}

}

class SonAgeException extends Exception

{

int f,s;

SonAgeException(int fage,int sage)

{

this.f=fage;

this.s=sage;

}

public String toString()

{

if(f==s)

return("Wrong age!Father's age can't be equal to son's age");

if(s<0)

return("Wrong age!Son's age can't be less than 0");

else

return("Wrong age!Son's age can't be greater than father's age");

}

}

class Father

{

int fage;

Scanner sc=new Scanner(System.in);

Father()

{

System.out.println("Enter the father's age");

fage=sc.nextInt();

}

void exc1() throws WrongAge

{

if(fage<0)

throw new WrongAge(fage);

}

}

class Son extends Father

{

int sage;

Scanner sc=new Scanner(System.in);

Son()

{

super();

System.out.println("Enter the son's age");

sage=sc.nextInt();

}

void exc2()throws SonAgeException

{

if(sage<0 || sage>=fage)

throw new SonAgeException(fage,sage);

}

}

class lab8

{

public static void main(String args[])

{

Son s=new Son();

try{

s.exc1();

}

catch(WrongAge e)

{

System.out.println(e);

}

try{

s.exc2();

}

catch(SonAgeException e)

{

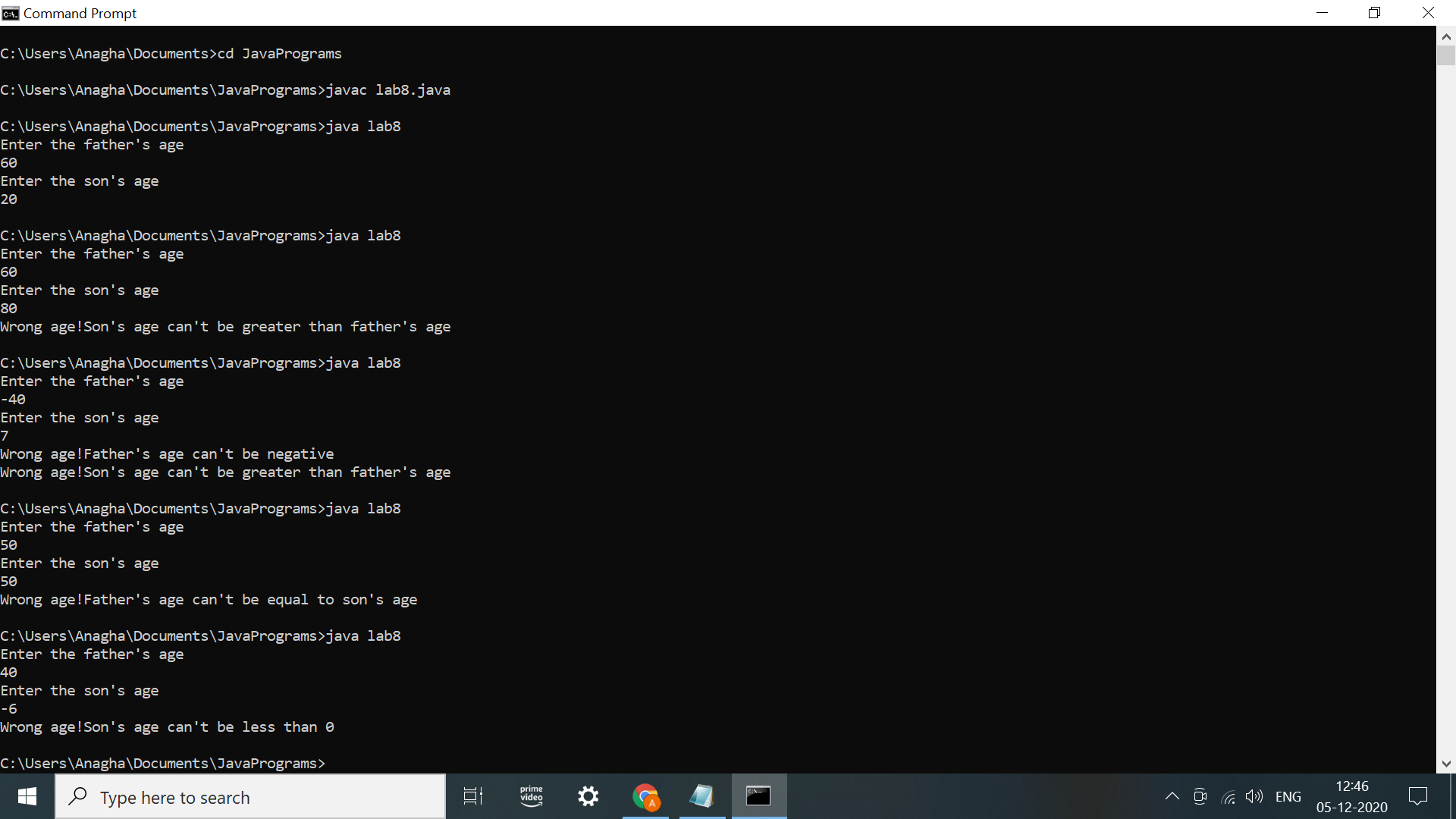
System.out.println(e);

}

}

}

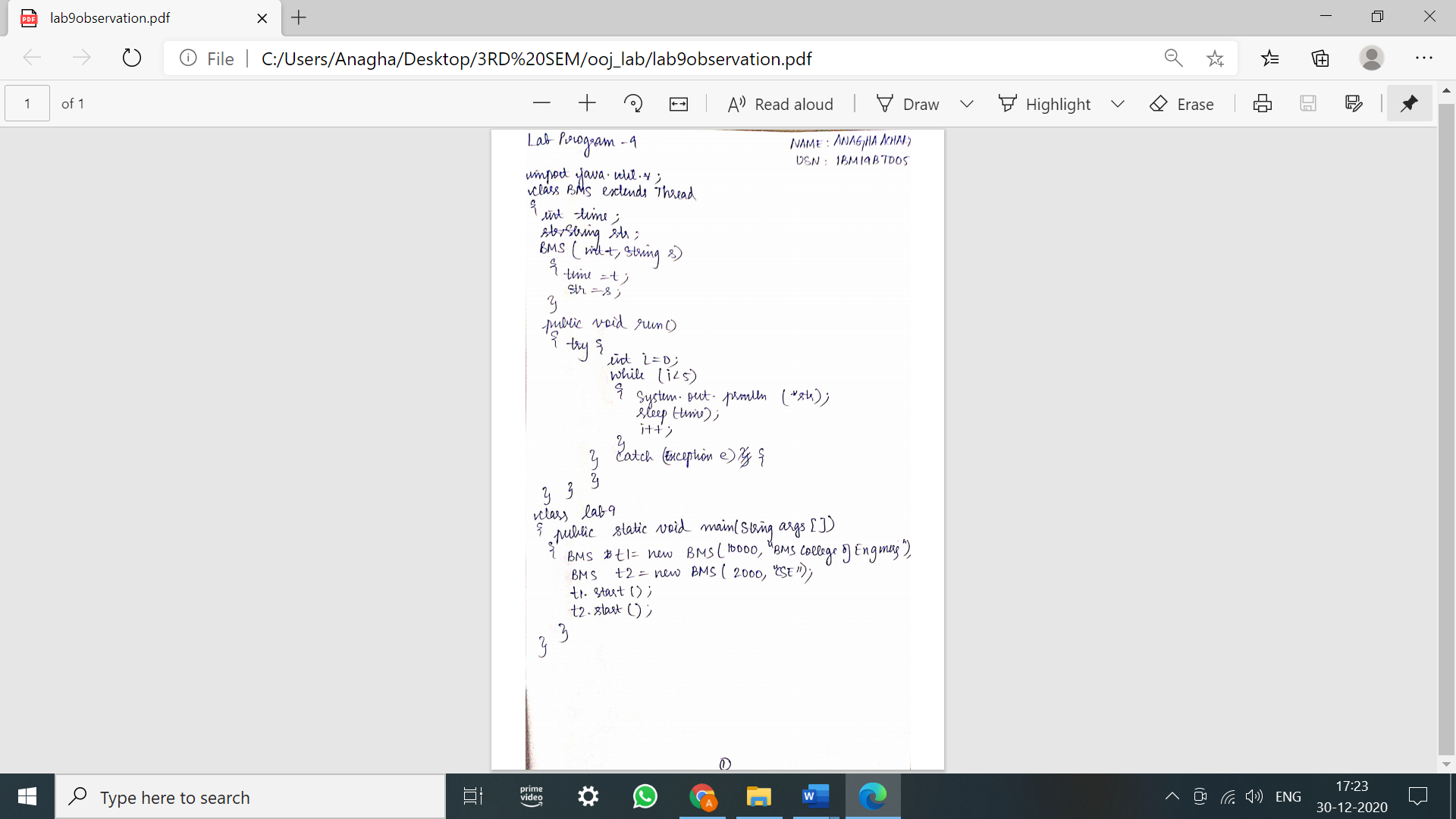
OUTPUT



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.

OBSERVATION:



PROGRAM:

import java.util.Scanner;

class BMS extends Thread

{

int time;

String str;

BMS(int t,String s)

{

time = t;

str = s;

}

public void run()

{

try{

int i=0;

while (i<5)

{

System.out.println(str);

sleep(time);

i++;

}

} catch (Exception e){

}

}

}

class lab9

{

public static void main(String args[])

{

BMS t1 = new BMS(10000,"BMS College of Engineering");

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

t1.start();

t2.start();

}

}

OUTPUT

