NAME: Soham Saha

SECTION: CSE2A

CLASS ROLL: 61

ENROLLMENT NUMBER: 12019009001389

ASSINGEMENT DATE: 23rd APRIL,2021

SUBJECT: OOP’S JAVA

1.i)

class q1

{

public static void main(String as[])

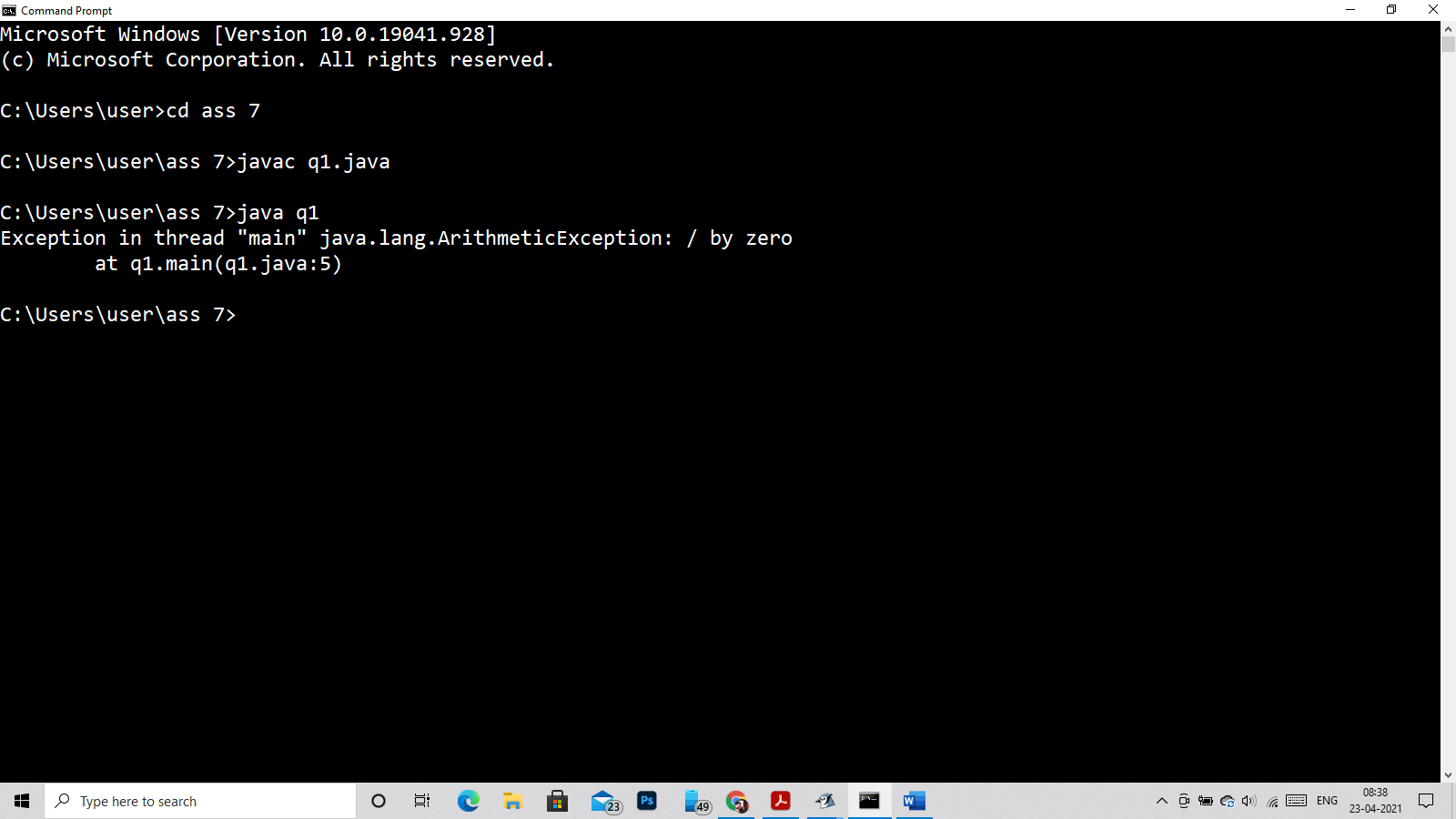
{

int i=10/0;

System.out.println(i);

}

}



ii)

class q1

{

public static void main(String as[])

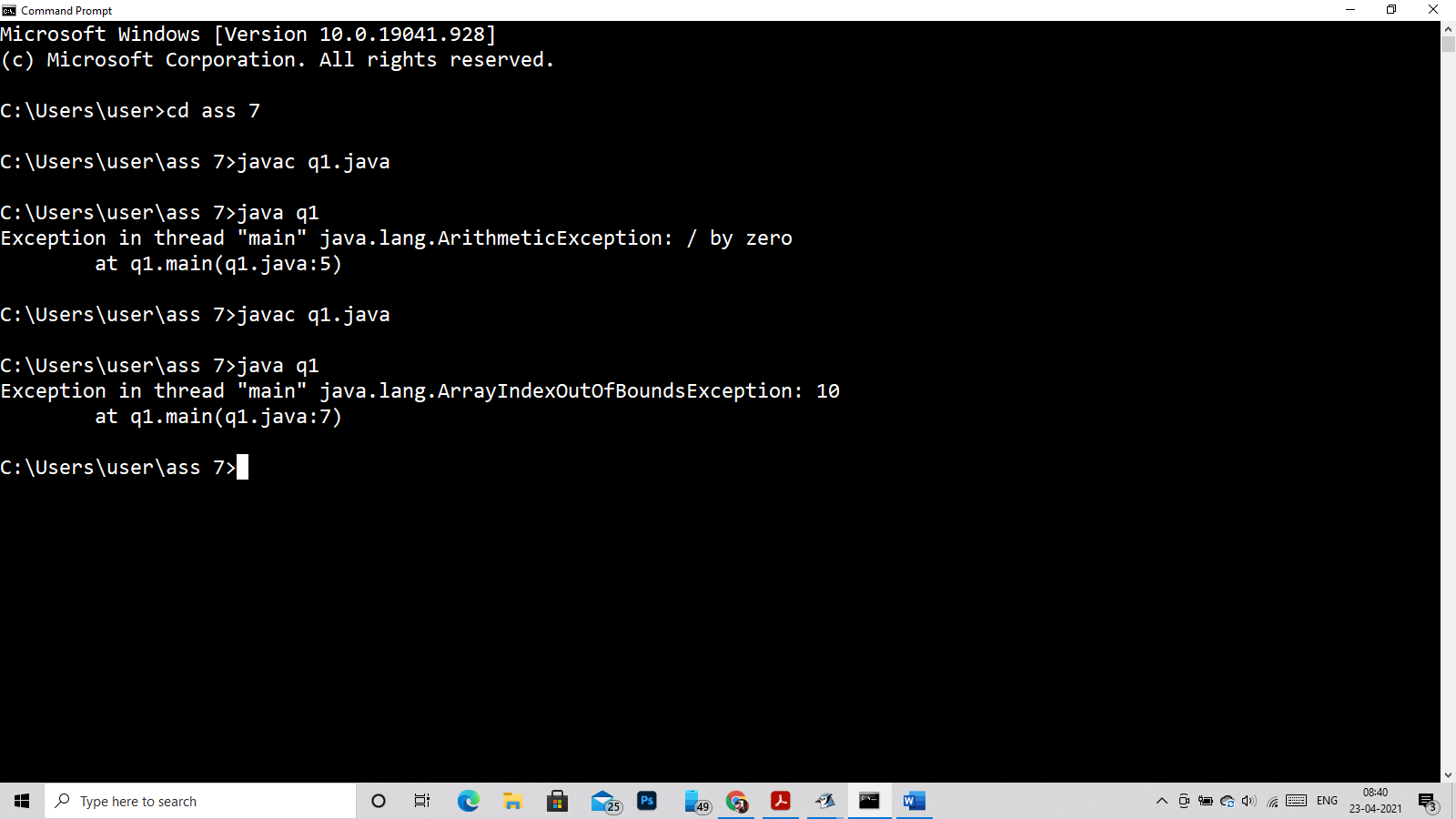
{

int i[]={10,34,23,44};

System.out.println(i[10]);

}

}



iii)

import java.util.\*;

class q1

{

public static void main(String as[])

{

Scanner in = new Scanner(System.in);

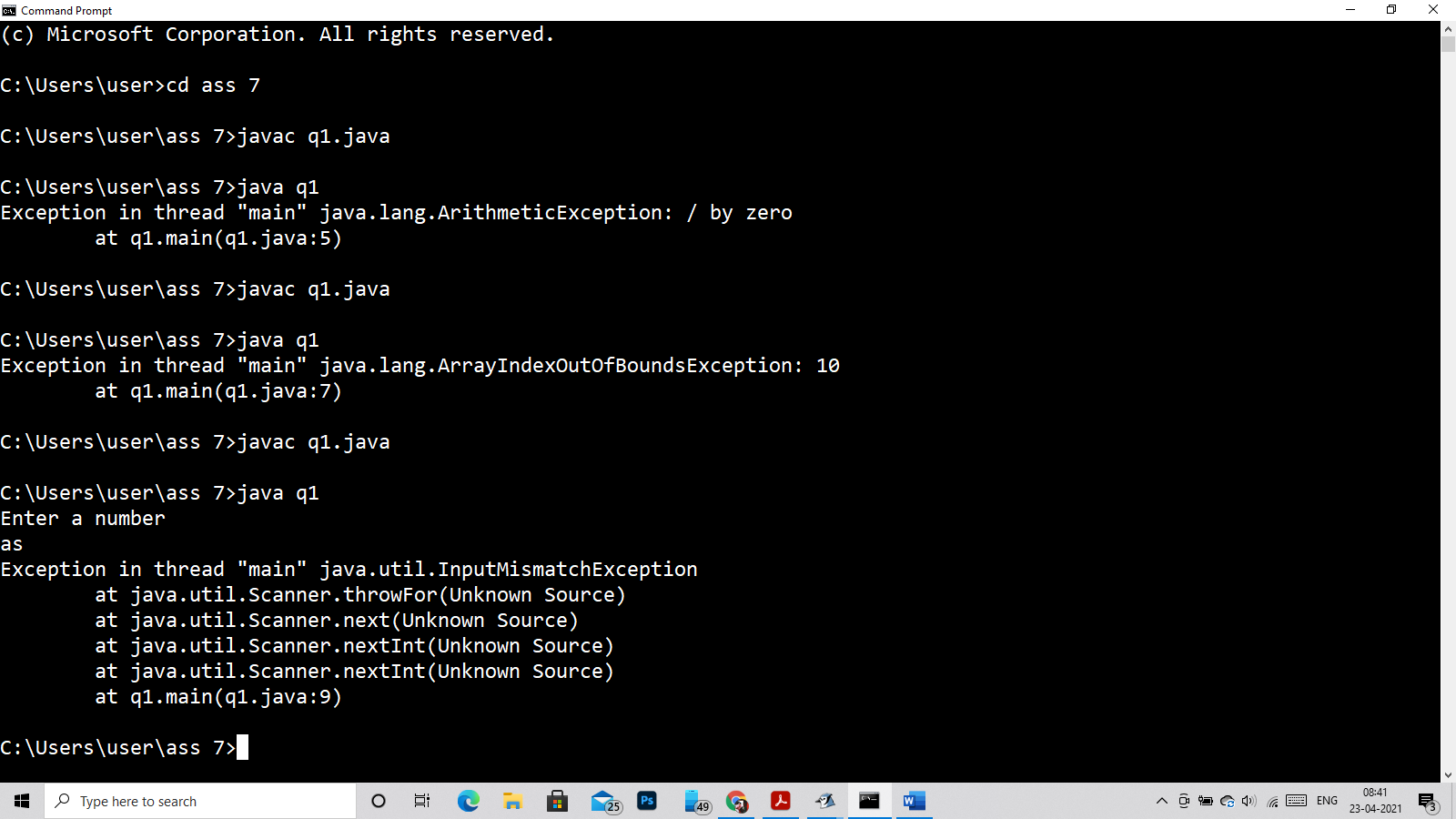
int i;

System.out.println("Enter a number");

i = in.nextInt();

}

}



vi)

class q1

{

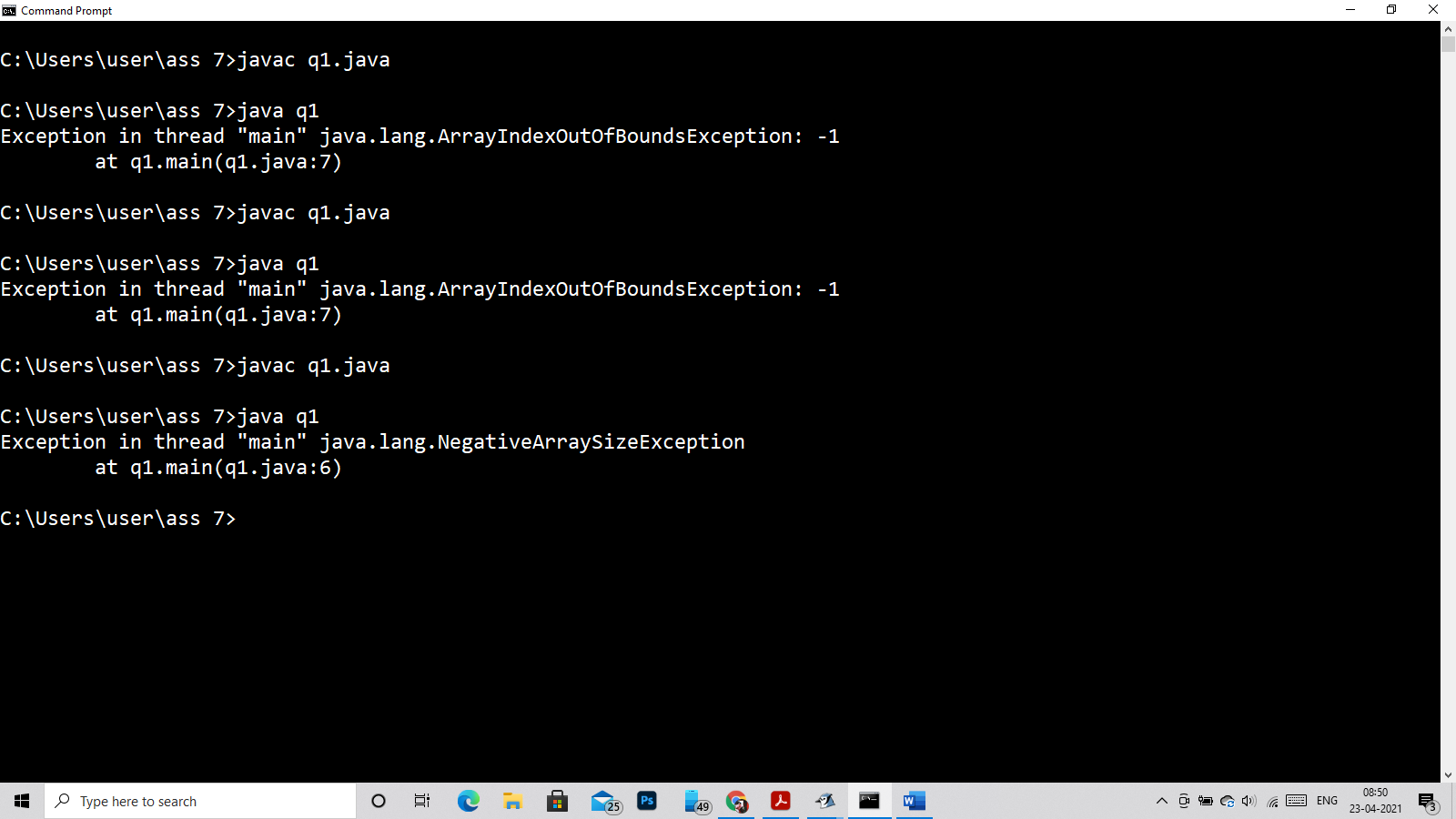
public static void main(String as[])

{

int i[]=new int[-1];

}

}



2)

class q2

{

public static void main(String as[])

{

try

{

int i[]=new int[-1];

}catch(Exception e)

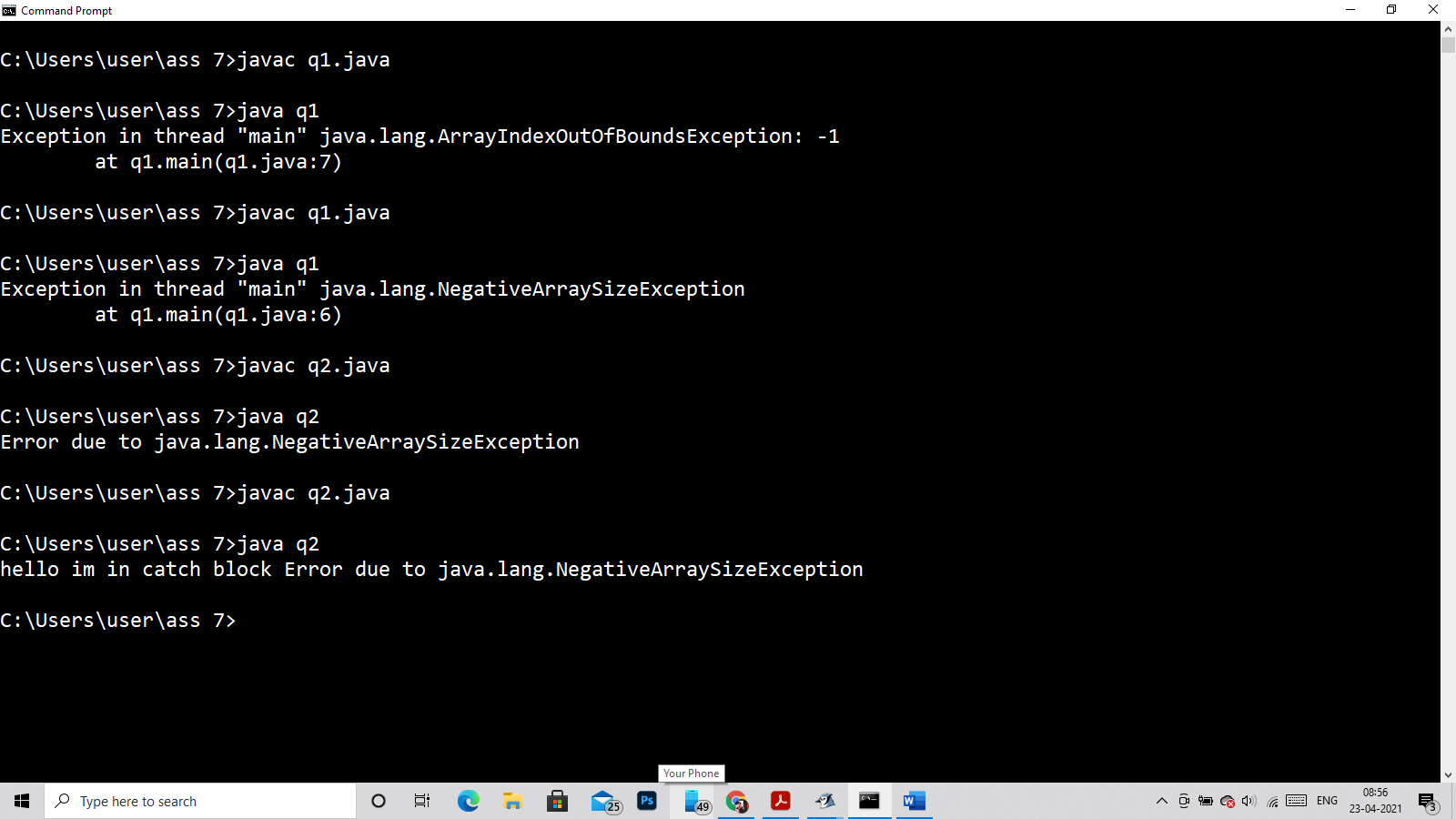
{

System.out.println("hello im in catch block Error due to "+e);

}

}

}



3)

class NoMatchException extends Exception {

public NoMatchException(String message){

super(message);

}

}

class error {

private String s;

error(String s) throws NoMatchException {

this.s = s;

if (s.equals("India")) {

System.out.print("Matched!\n");

} else {

throw new NoMatchException("Not Matched!\n");

}

}

}

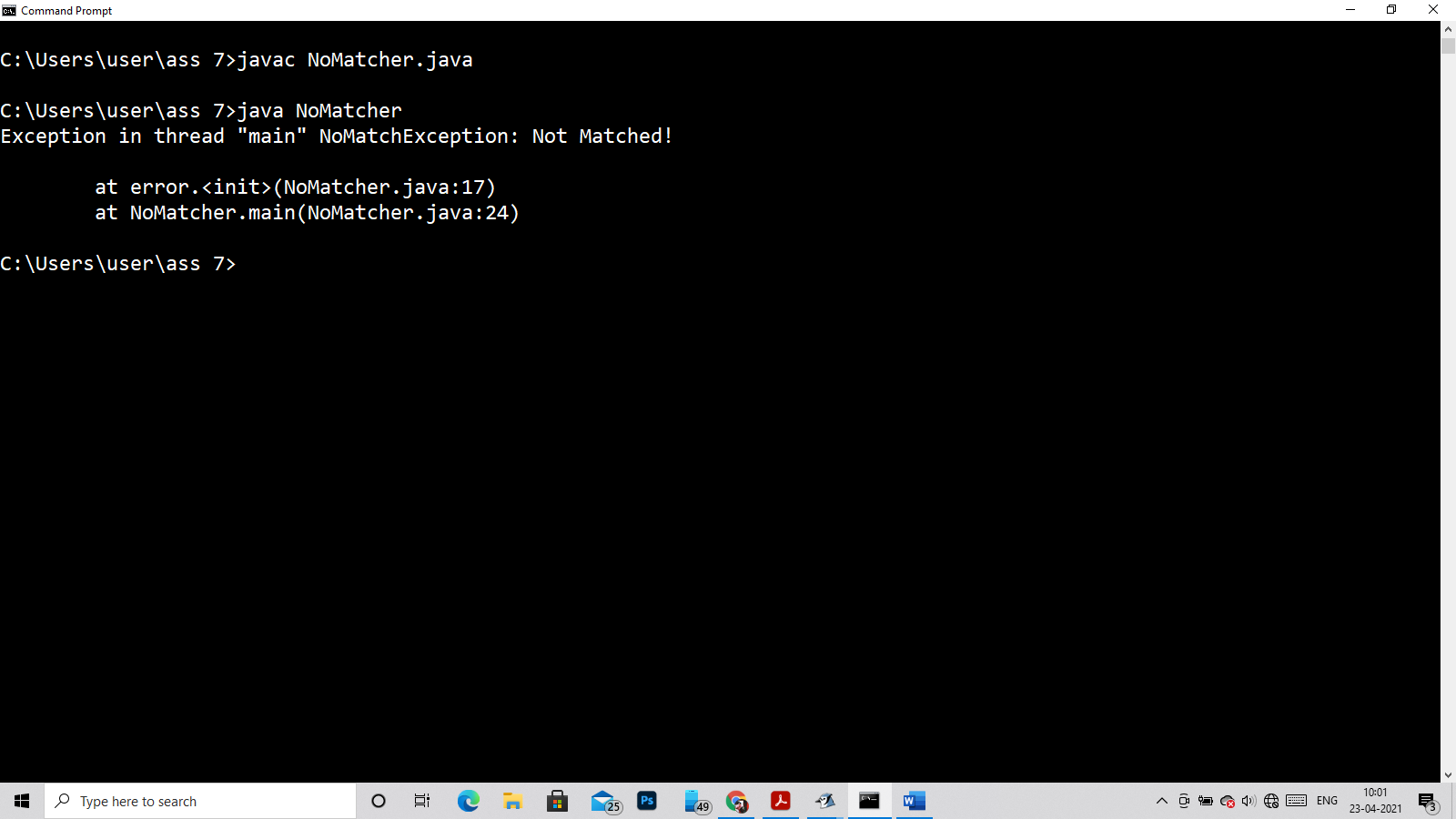
public class NoMatcher {

public static void main(String[] a) throws NoMatchException {

error v = new error("America");

}

}



4)

class error

{

error(char c)throws Exception

{

if((c>='a' && c<='z' )||(c>='A' && c<='Z' ))

{

System.out.println(c);

}

else

{

throw new Exception("not a charecter");

}

}

}

public class q4

{

public static void main(String args[])throws Exception

{

String str ="Binod12kumar";

char ch;

int len=str.length();

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

{

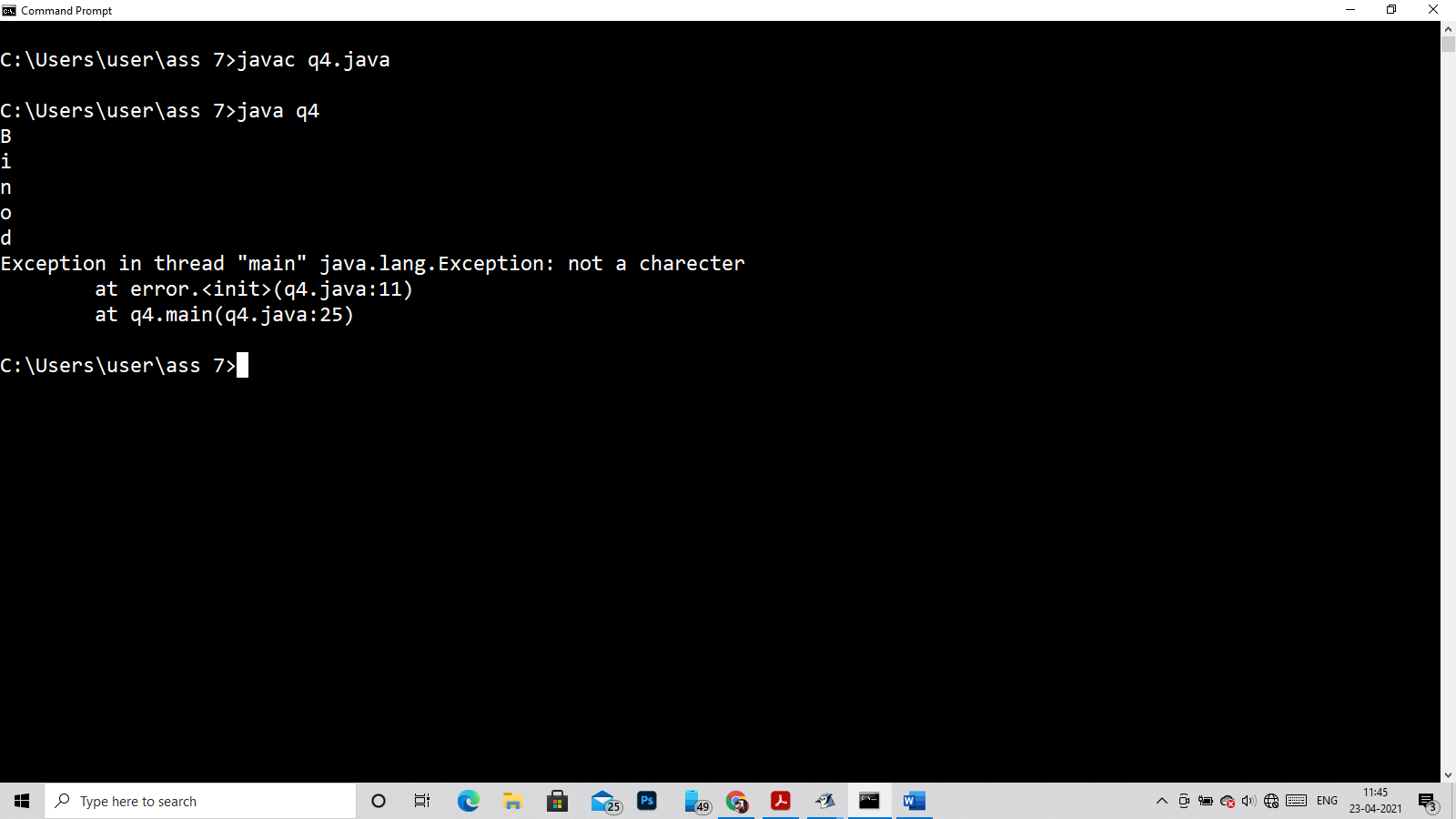
ch = str.charAt(i);

error obj = new error(ch);

}

}

}



5)

class IllegalArgumentException extends Exception

{

public IllegalArgumentException(String msg)

{

super(msg);

}

}

public class Factroial

{

static long range=20;

static long check(int i)throws IllegalArgumentException

{

if(i<0)

{

throw new IllegalArgumentException("Value of x must be positive");

}else

if(i>range)

{

throw new IllegalArgumentException("Result will overflow");

}else

{

int k;long fu=1;

for(k=1;k<=i;k++)

{

fu=fu\*k;

}

return fu;

}

}

public static void main(String args[])

{

long arr[]=new long[5];

int k=0;

try

{

for (int i=3;i<=40;i=i+5)

{

arr[k]=check(i);

k++;

}

}

catch(Exception e){

System.out.println("Error:: "+e);

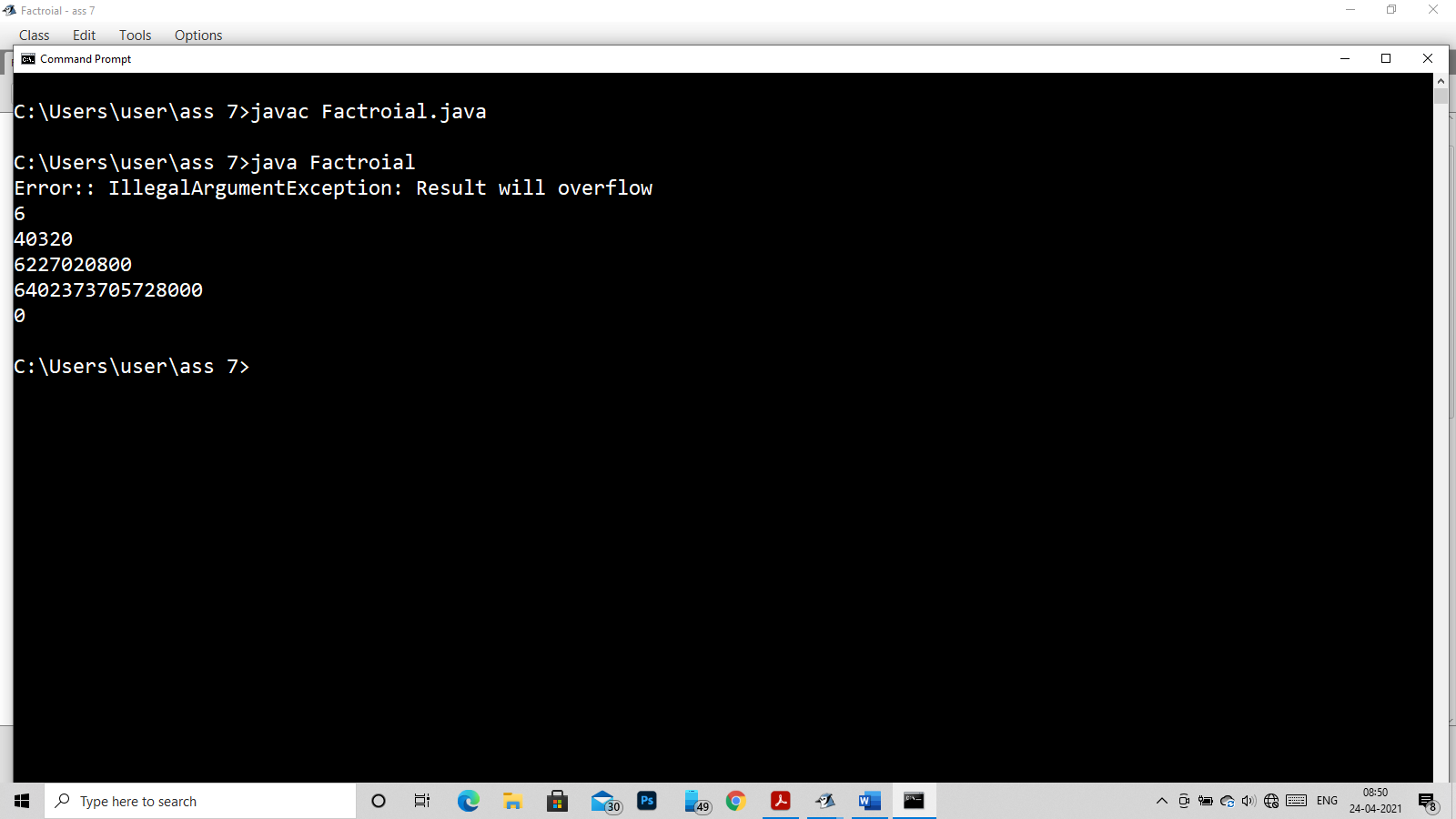
}

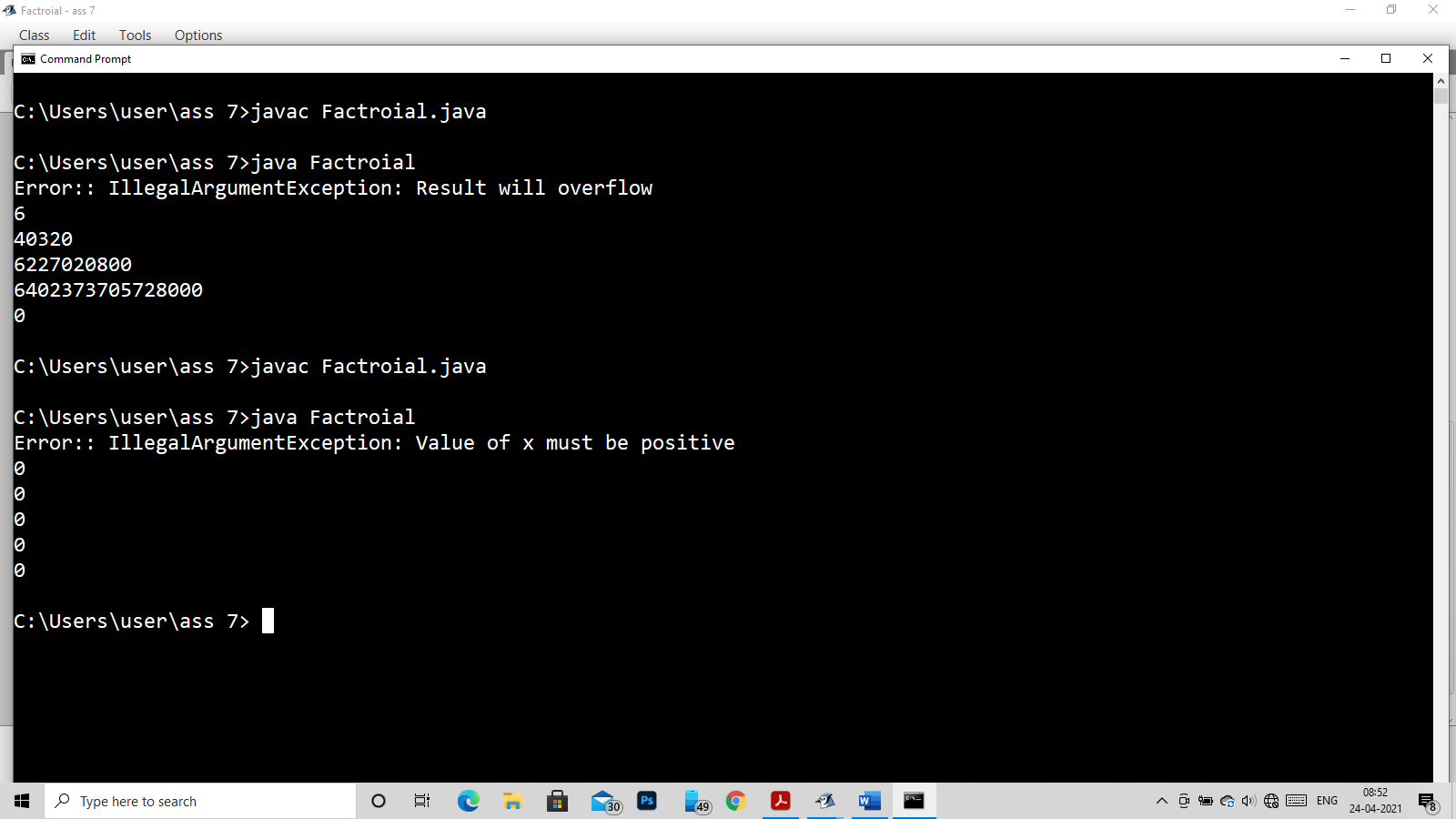
for(int i=0;i<5;i++)

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

}

}





Q6) is same as q4

Q7)

import java.util.\*;

class NoMatchFoundException extends Exception

{

public NoMatchFoundException(String msg)

{

super(msg);

}

}

public class q7

{

static void check(String str)throws NoMatchFoundException

{

if(str.equals("India"))

{

System.out.println("Delhi");

}else

if(str.equals("Srilanka"))

{

System.out.println("Colombo");

}else if(str.equals("Pakistan"))

{

System.out.println("Ishlamadab");

}else

{

throw new NoMatchFoundException("No capital found");

}

}

public static void main(String asd[])throws NoMatchFoundException

{

Scanner in = new Scanner(System.in);

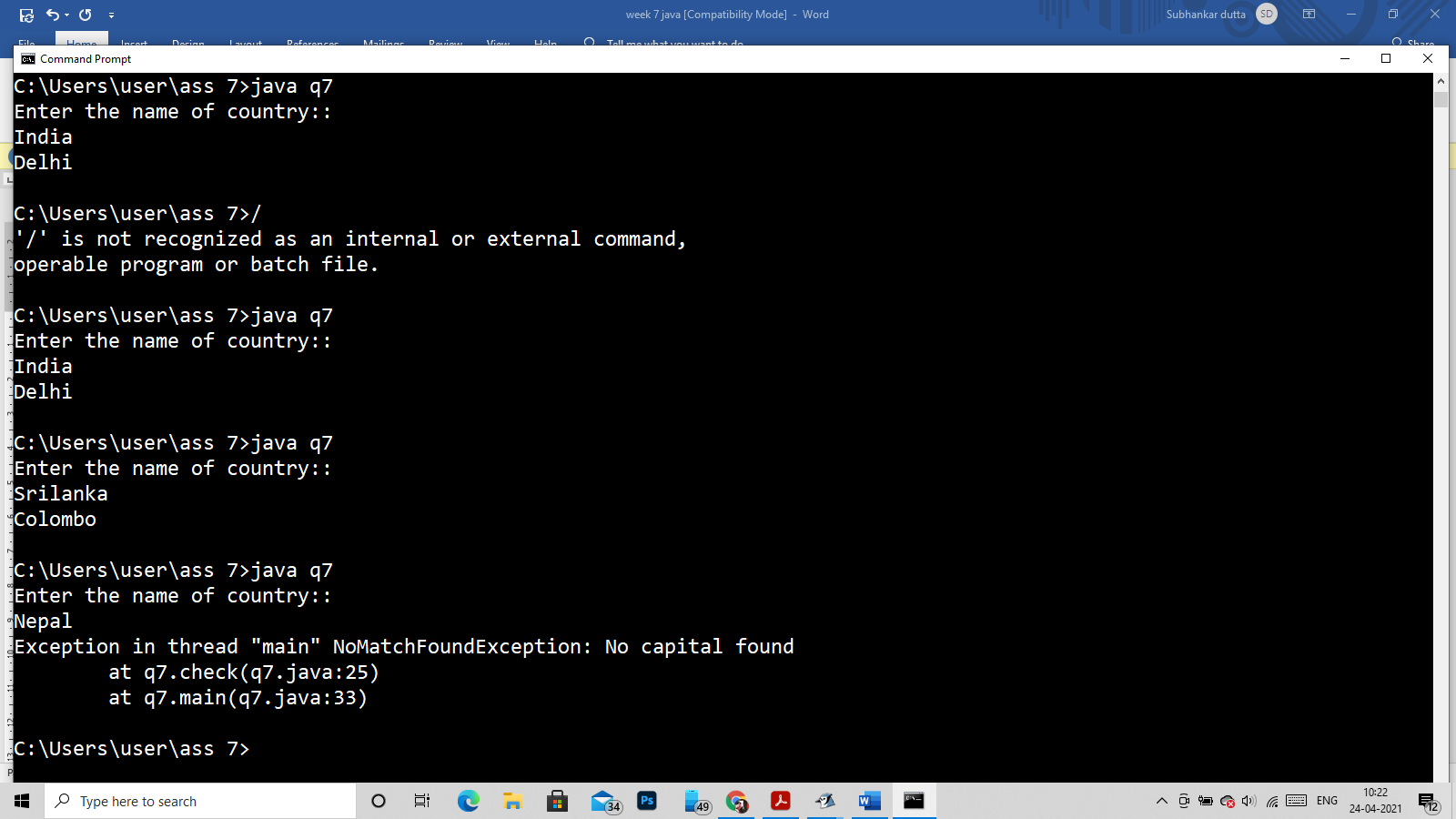
System.out.println("Enter the name of country::");

String str=in.nextLine();

check(str);

}

}



8)

class IllegalArgumentException extends Exception

{

public IllegalArgumentException(String msg)

{

super(msg);

}

}

public class q8

{

static int fact(int n)throws IllegalArgumentException

{

if(n<0)

throw new IllegalArgumentException("Error");

else

if(n==0)

return 1;

else

return(n\*fact(n-1));

}

public static void main(String a[])

{

try

{

int x = Integer.parseInt(a[0]);

System.out.println(x +"! = "+fact(x));

}catch (ArrayIndexOutOfBoundsException e) {

System.out.println("You must specify an argument");

}

catch (NumberFormatException e) {

System.out.println("The argument you specify must be an integer");

}

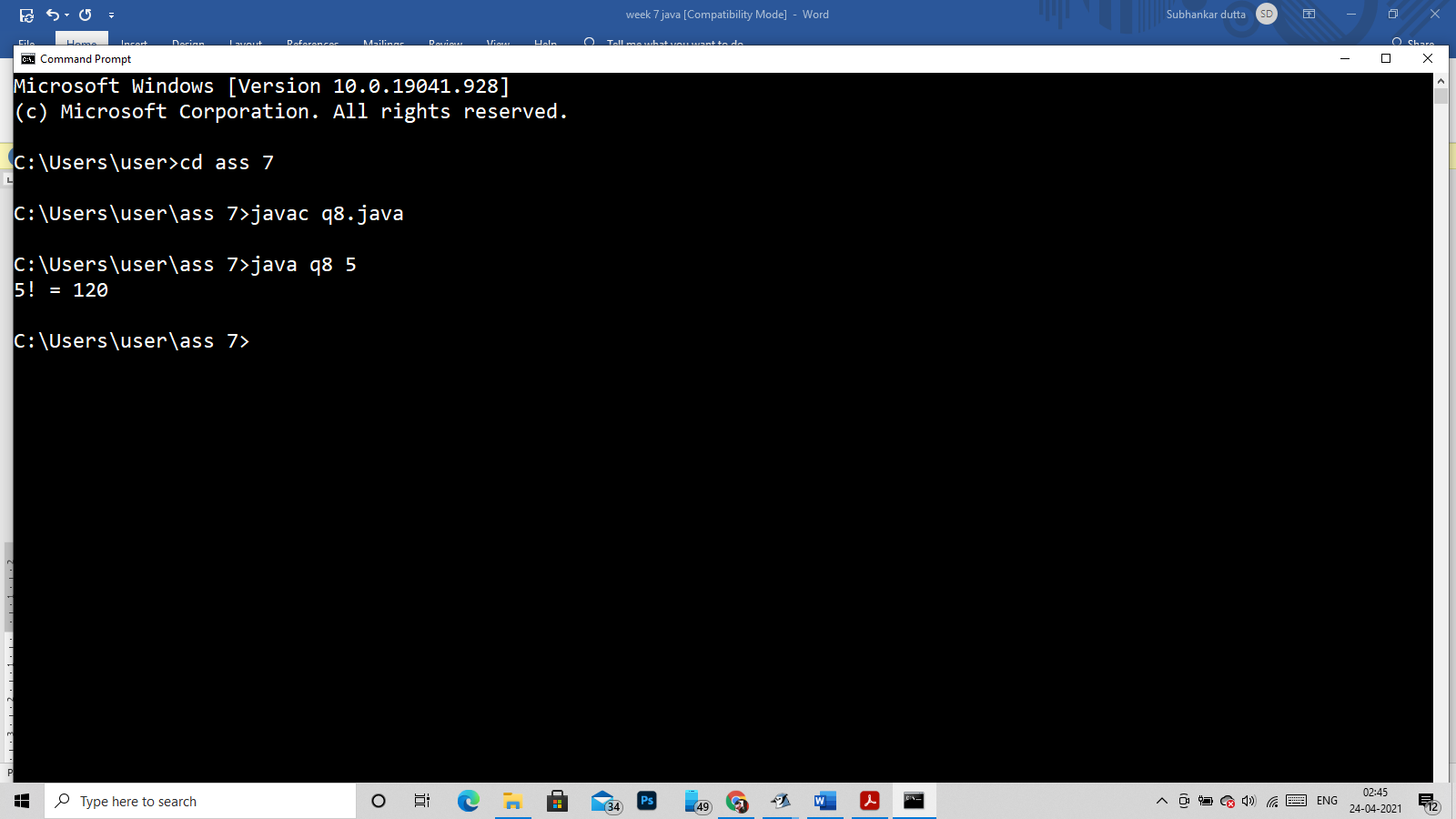
catch (IllegalArgumentException e) {

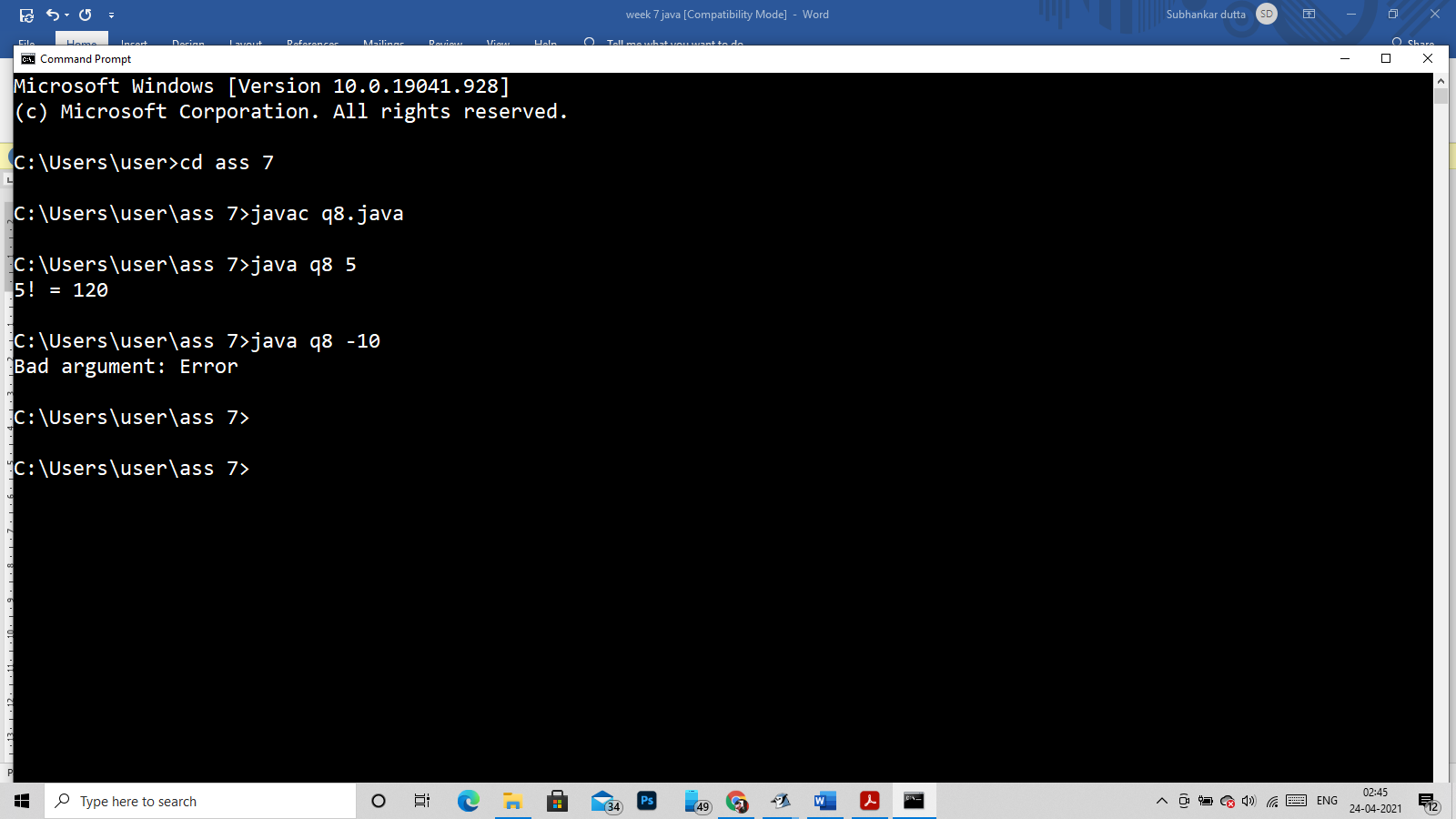
System.out.println("Bad argument: " + e.getMessage( ));

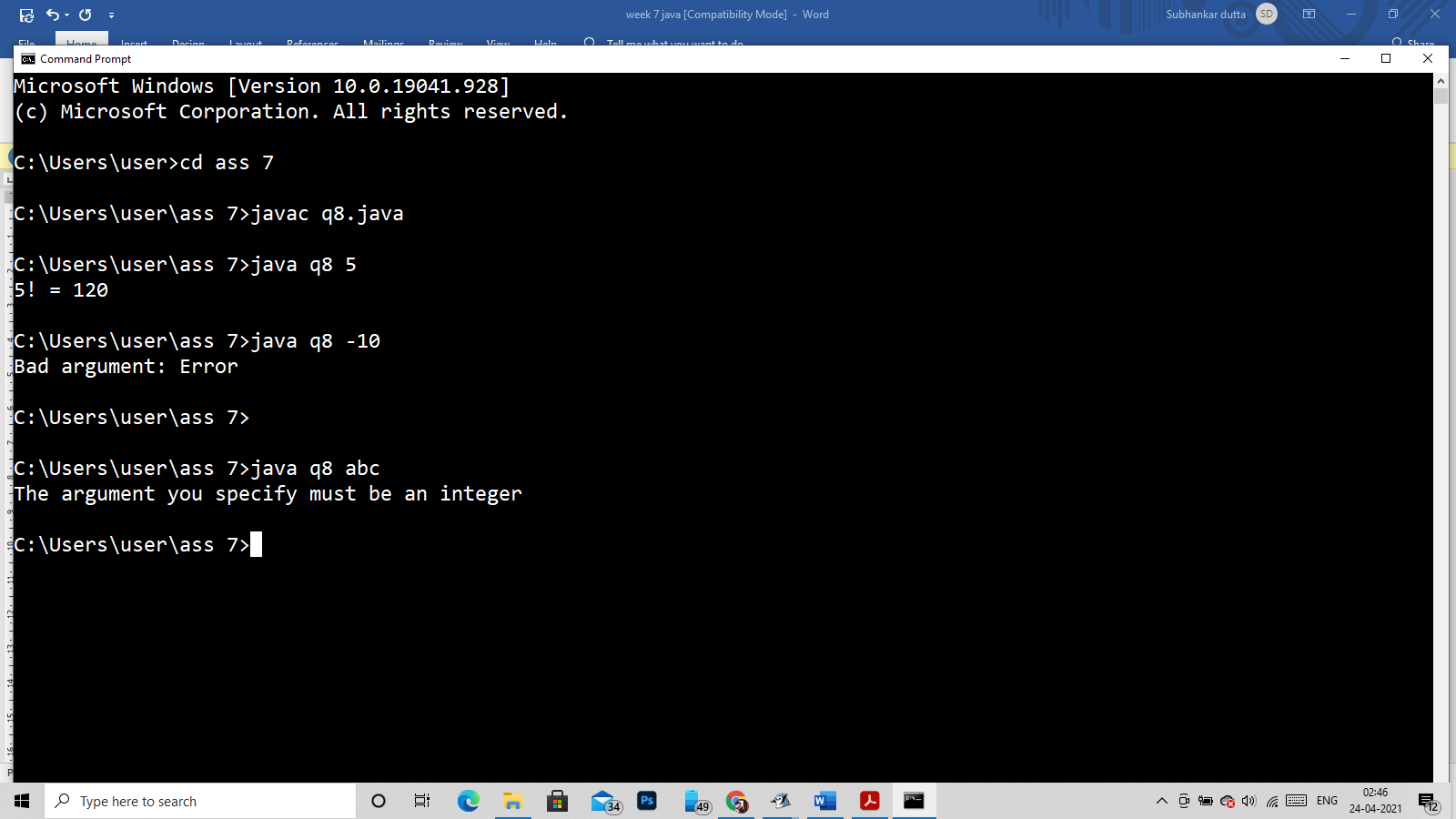
}

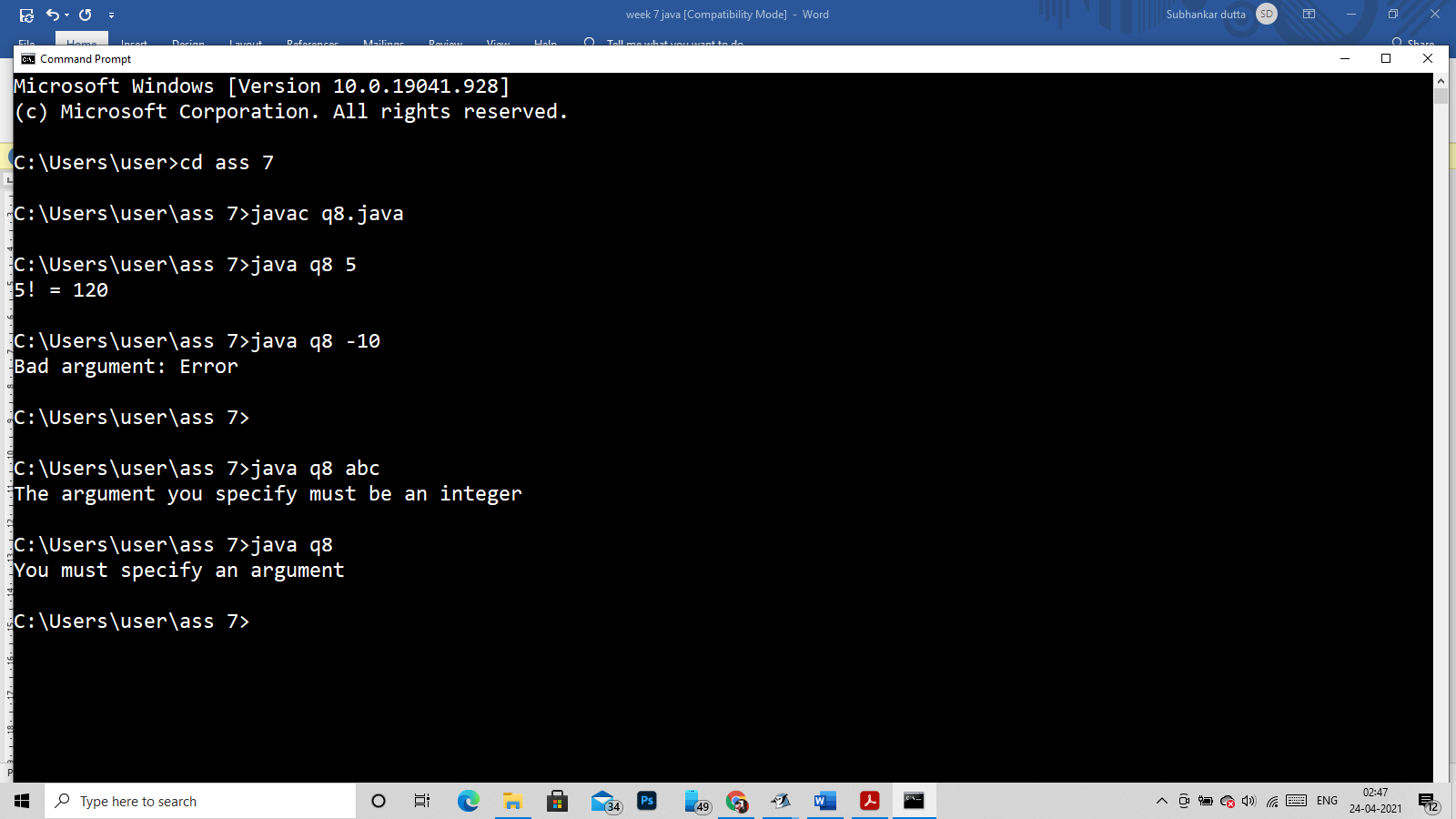
}

}









Q9)

class CheckArgumentexception extends Exception

{

public CheckArgumentexception(String msg)

{

super(msg);

}

}

public class q9

{

static void check(String arr[])throws CheckArgumentexception

{

if(arr.length!=5)

{

throw new CheckArgumentexception("CheckArgumentexception error not =5"+arr.length);

}

else

{

int i,sum=0;

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

{

sum+=Integer.parseInt(arr[i]) ;

}

System.out.println(sum);

}

}

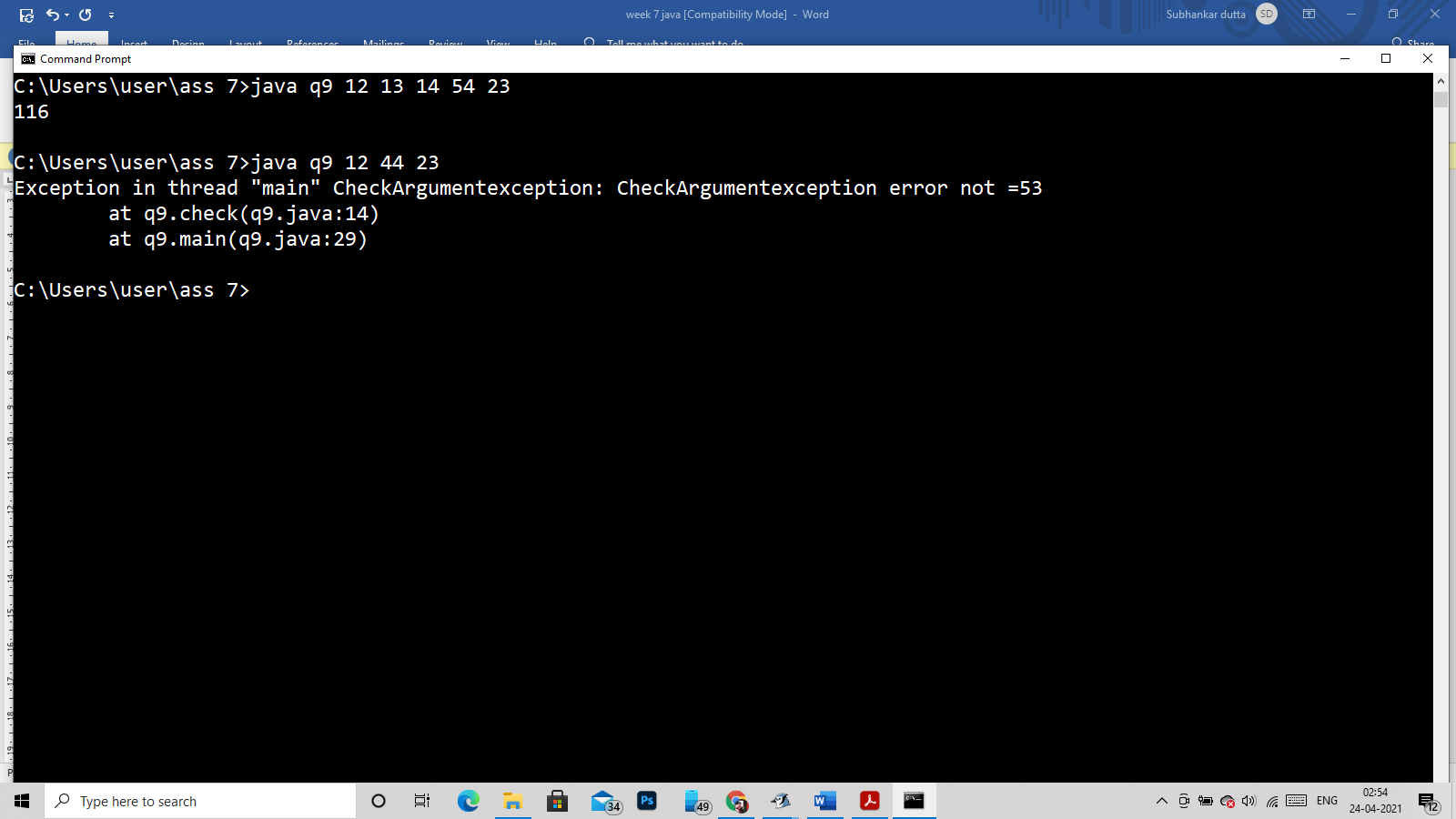
public static void main(String args[])throws CheckArgumentexception

{

check(args);

}

}



Q10)

class RangeException extends Exception

{

public RangeException(String msg)

{

super(msg);

}

}

public class Student

{

static void marks(String str[])throws RangeException

{

int i=0,sum=0;

String name="";

for(String st : str)

{

if(i>=1)

{

int I = Integer.parseInt(st);

if(I>50||I<0)

{

throw new RangeException("Error in range <50 and >0");

}

else

{

sum+=I;

}

}

else

{

name=st;

}

i++;

}

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

System.out.println("Your full marks is:: "+sum);

float per = sum\*100/300;

System.out.println("The %age is:: "+per);

}

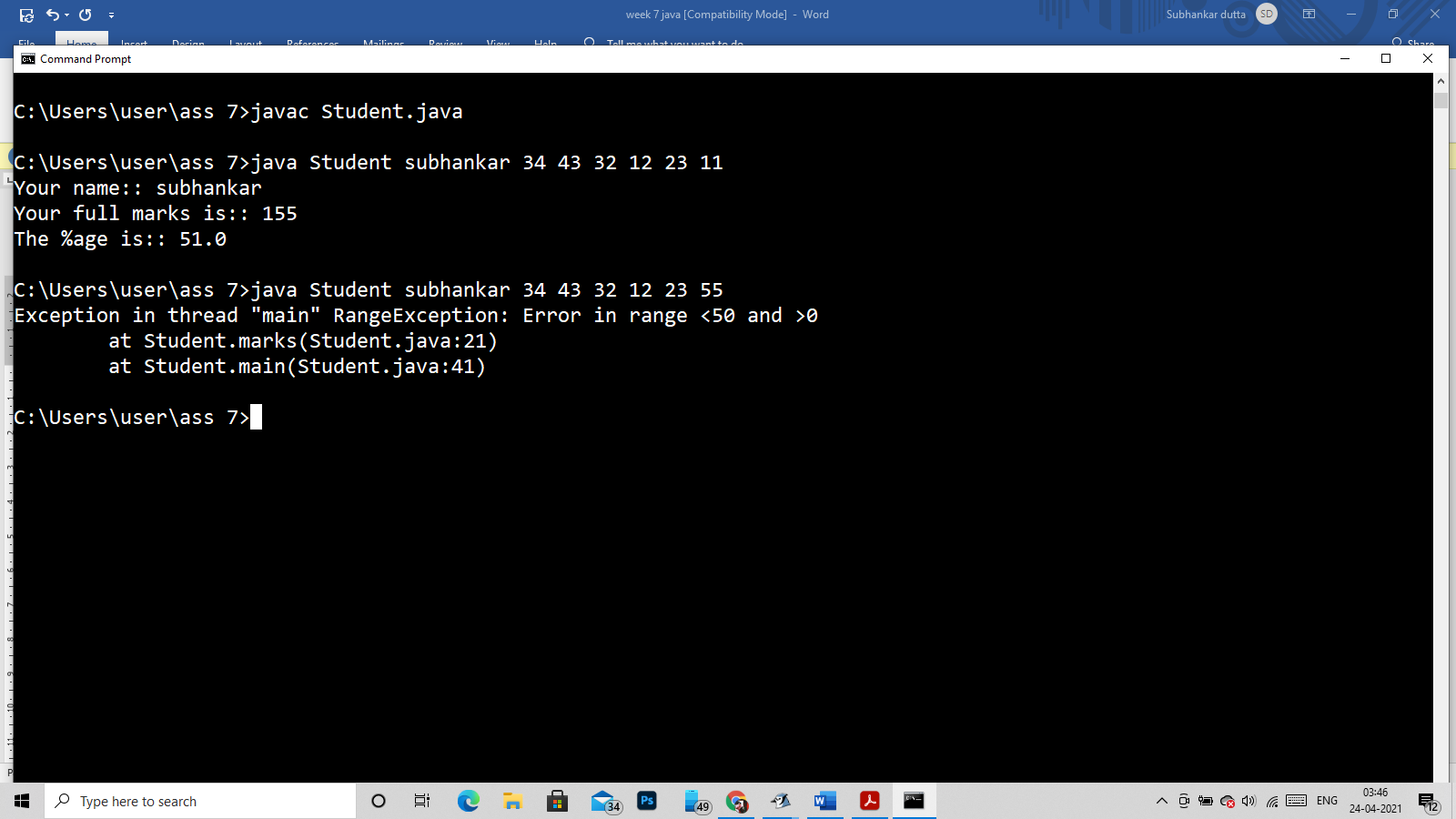
public static void main(String args[])throws RangeException

{

marks(args);

}

}



Q12)

class TempException extends Exception

{

public TempException(String msg)

{

super(msg);

}

}

public class q11

{

static void Temp(int temp)throws TempException

{

if(temp<5)

{

throw new TempException("Too Cold");

}else if(temp>35)

{

throw new TempException("Too Hot");

}else

{

System.out.println("Normal");

double f = (temp\*9/5)+32;

System.out.println("Fahrenhite:: > "+f);

}

}

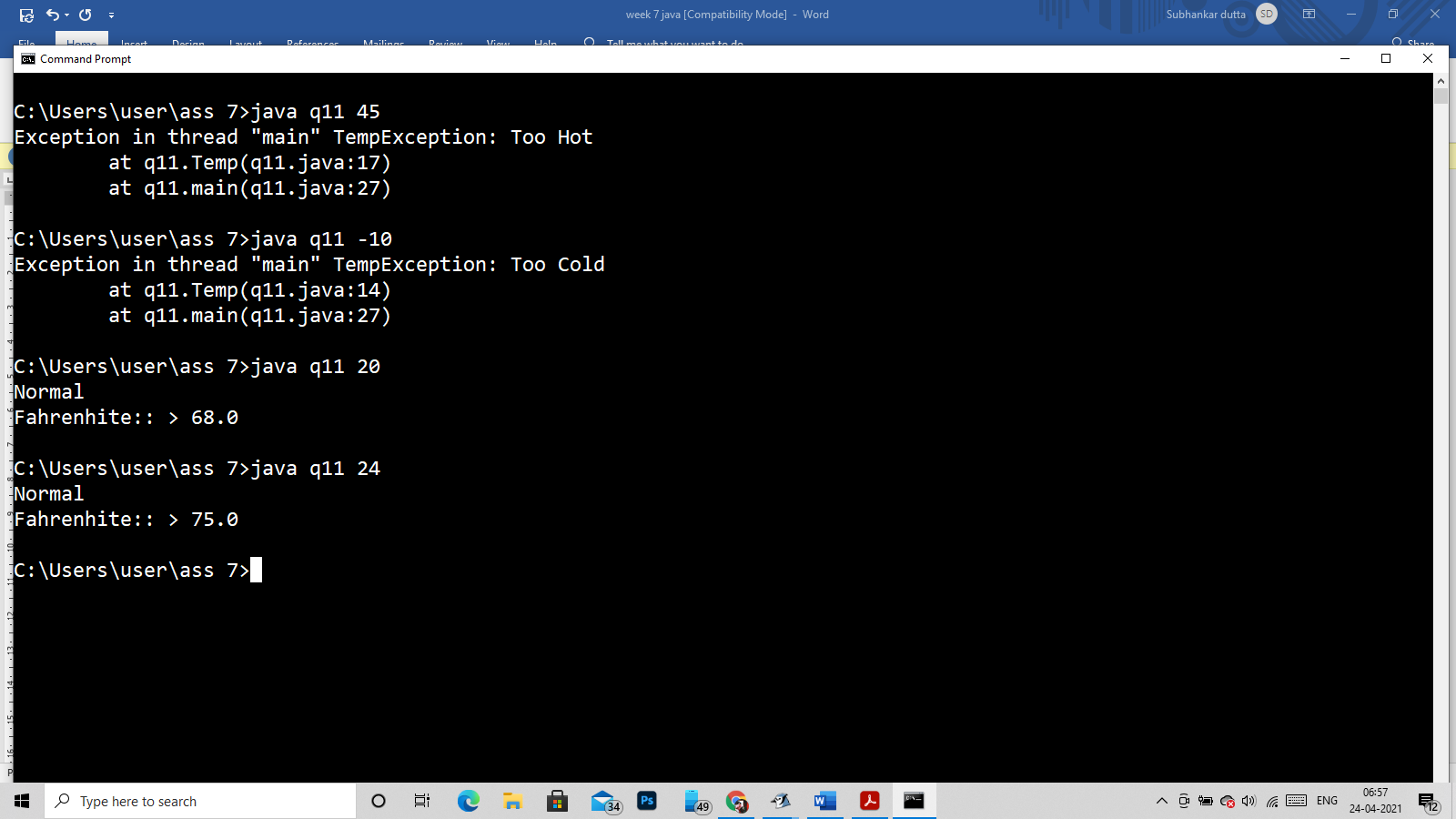
public static void main(String args[])throws TempException

{

Temp(Integer.parseInt(args[0]));

}

}



Q13)

class AgeException extends Exception

{

public AgeException(String msg)

{

super(msg);

}

}

public class q11

{

static void Age(String nm,int age)throws AgeException

{

if(age<20)

{

throw new AgeException("Too Young");

}else if(age>45)

{

throw new AgeException("Too Older");

}else

{

System.out.println("Eligible");

System.out.println("Name:: > "+nm);

}

}

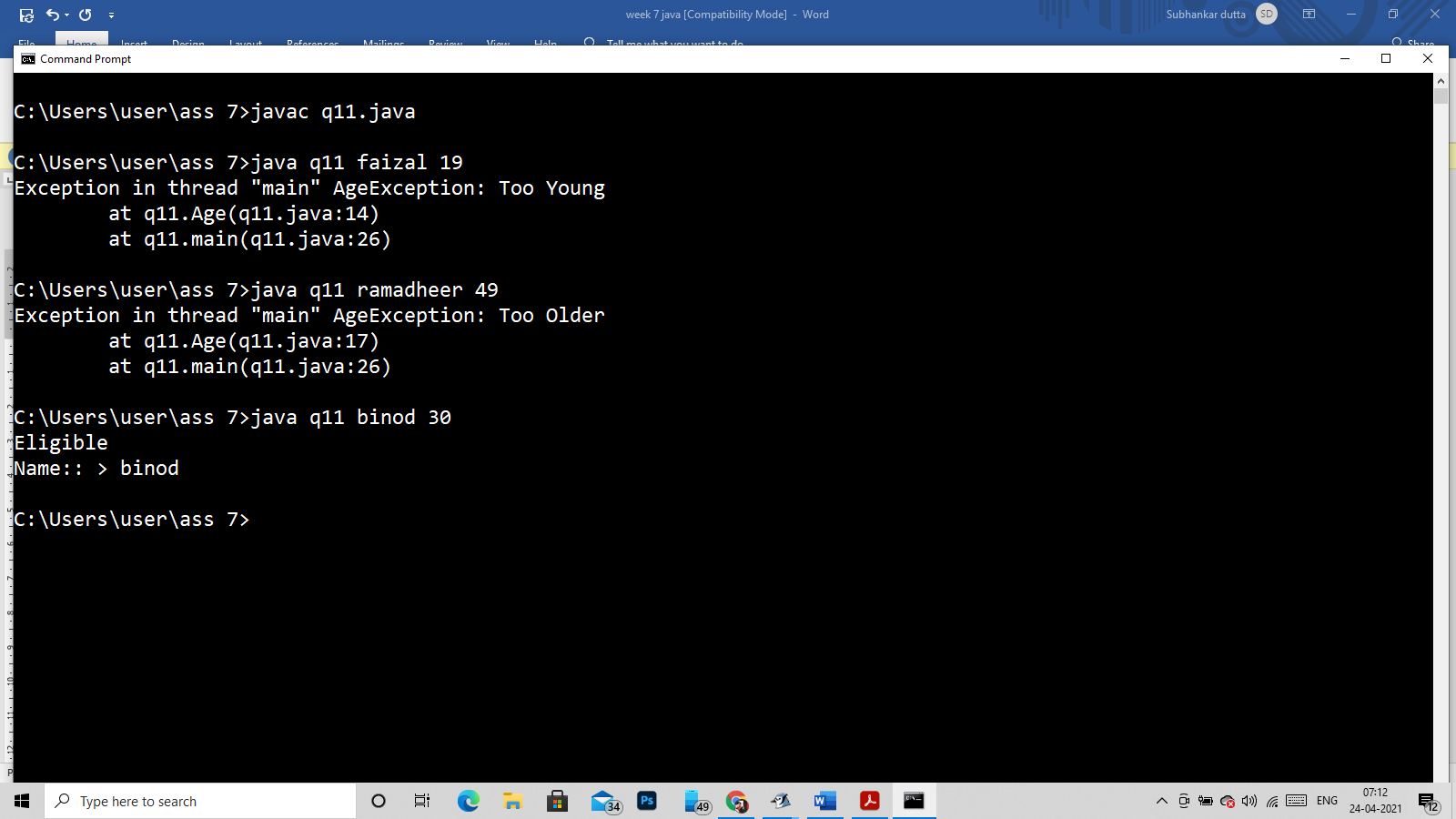
public static void main(String args[])throws AgeException

{

Age(args[0],Integer.parseInt(args[1]));

}

}



Q14)

//bin to dece

class WrongNumberFormat extends Exception

{

public WrongNumberFormat(String msg)

{

super(msg);

}

}

public class q12

{

static void convert(long exp)throws WrongNumberFormat

{

long d=1,sum=0,i=0;

while(exp != 0)

{

d=exp%10;

if(d==0||d==1)

{

sum+= d\*Math.pow(2,i);

i++;

}else

{

throw new WrongNumberFormat("wrong input");

}

exp=exp/10;

}

System.out.println("The Decimal is:> "+sum);

}

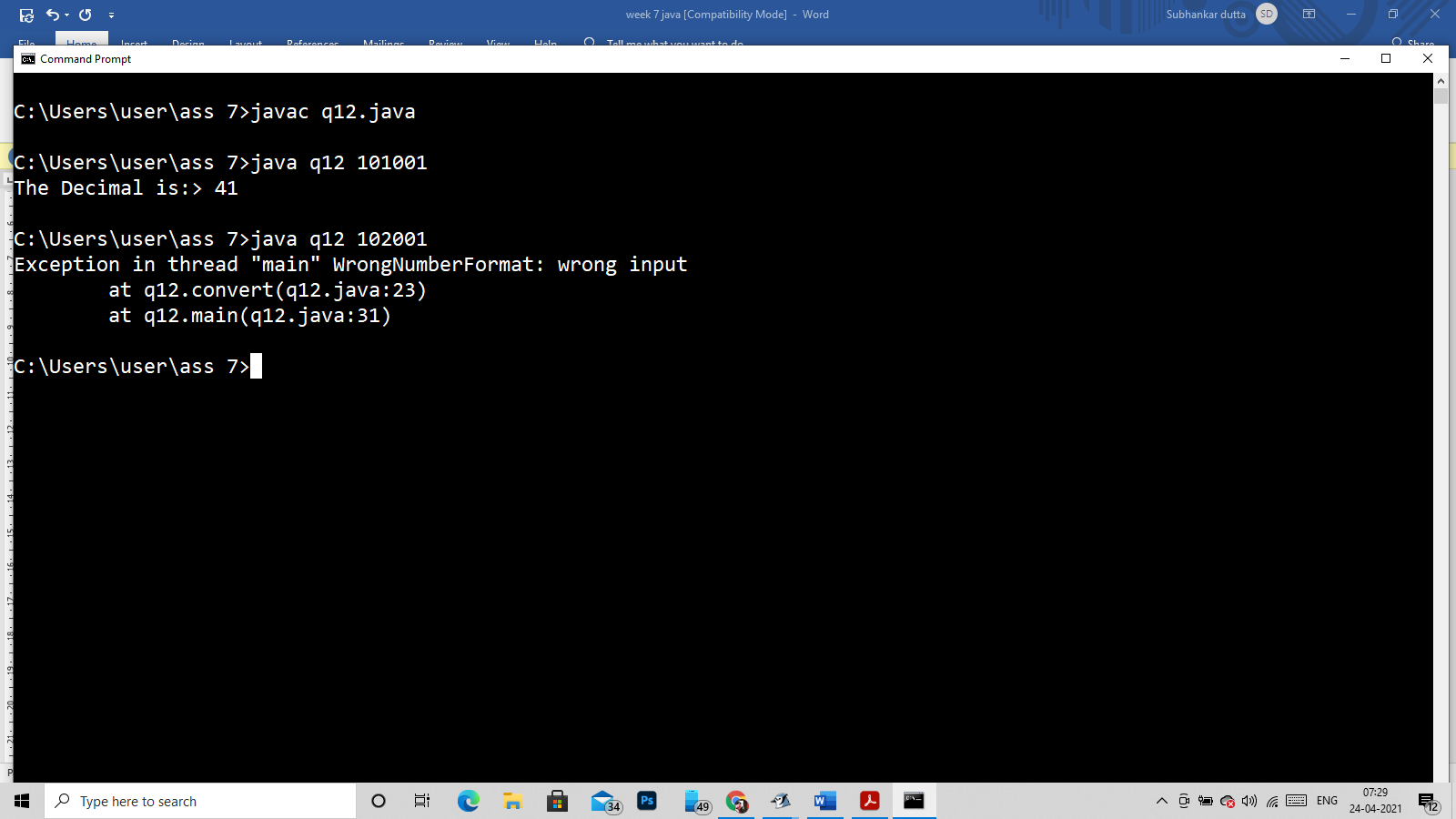
public static void main(String args[])throws WrongNumberFormat

{

convert(Long.parseLong(args[0]));

}

}



Q15)

class NestTry{

public static void main(String args[]){

try{

try{

System.out.println("going to divide");

int b =39/0;

}catch(ArithmeticException e){System.out.println(e);}

try{

int a[]=new int[5];

a[5]=4;

}catch(ArrayIndexOutOfBoundsException e){System.out.println(e);}

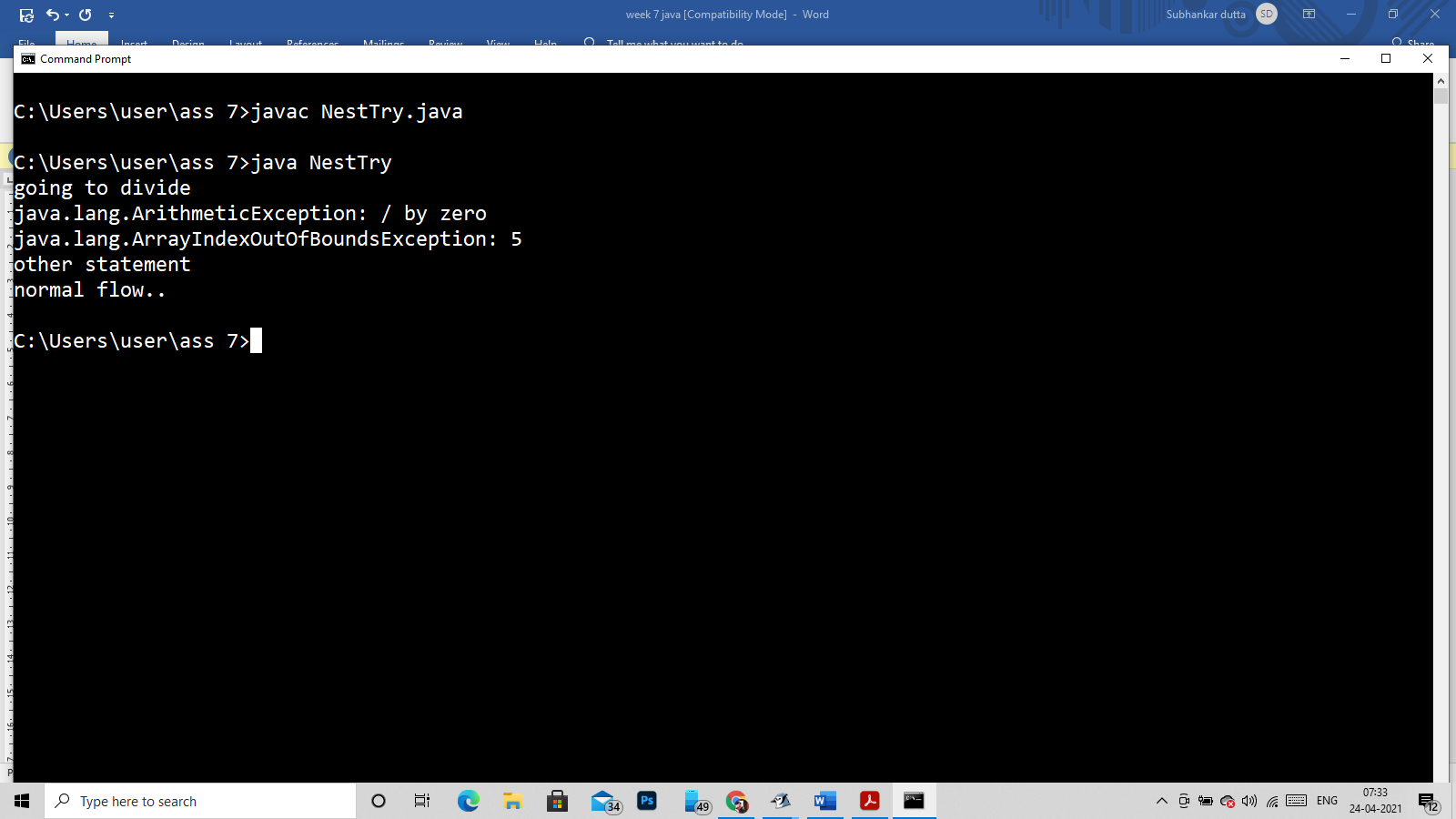
System.out.println("other statement");

}catch(Exception e){System.out.println("handeled");}

System.out.println("normal flow..");

}

}



Q16)

import java.util.\*;

class LessBalanceException extends Exception

{

public LessBalanceException(double bal)

{

System.out.println("Withdrawing "+bal+" is invalid");

}

}

class Account

{

int accno;

double bal;

String name;

static int count=0;

Account(int accno,String name,double bal)

{

System.out.println("\*\*\*New Account Created\*\*\*");

System.out.println("Accno :: "+accno);

this.accno = accno;

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

this.name = name;

System.out.println("Accno :: "+bal);

this.bal = bal;

count++;

System.out.println("Total Account is:: "+count);

}

void diposit(double amt)

{

System.out.println("Accno:: "+accno);

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

System.out.println("Old Bal:: "+bal);

bal=bal+amt;

System.out.println("New Bal:: "+bal);

System.out.println("\*\*\*\*\* Tranx Compleated \*\*\*\*\*");

}

void withdraw(double amt)throws LessBalanceException

{

if(bal-amt<500)

{

throw new LessBalanceException(amt);

}

System.out.println("Accno:: "+accno);

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

System.out.println("Old Bal:: "+bal);

bal=bal-amt;

System.out.println("New Bal:: "+bal);

System.out.println("\*\*\*\*\* Tranx Compleated \*\*\*\*\*");

}

void balance()

{

System.out.println("=============== Account Info ===============");

System.out.println("Accno:: "+accno);

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

System.out.println("Acc Bal:: "+bal);

System.out.println("\*\*\*\*\* Tranx Compleated \*\*\*\*\*");

}

}

public class Bank

{

static int i=0;

public static void main(String args[])

{

Scanner in = new Scanner(System.in);

Account obj[] = new Account[10];

int accno;

String name="";

int ch;

int flag=0;

double bal;

int I;

while(true)

{

flag=0;

System.out.println("Enter option:: \n1)Create Acc \n2)Deposit \n3)Withdraw \n4)Balance \n5)Exit");

ch=in.nextInt();

switch(ch)

{

case 1:

System.out.print("Enter the Accno::> ");

accno = in.nextInt();

System.out.print("Enter the Accno Holder Name:> ");

name=in.next();

System.out.print("Enter the Opening balance:> ");

bal=in.nextDouble();

if(bal<500)

{

System.out.println("Cannot create acc less the 500/-");

}

else

{

obj[i]=new Account(accno,name,bal);

i++;

}

break;

case 2:

System.out.println("Enter the accno:: ");

accno = in.nextInt();

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

{

if(accno==obj[I].accno)

{

break;

}

}

if(I<i)

{

System.out.println("Enter the Deposit Amt:: ");

bal = in.nextDouble();

obj[I].diposit(bal);

}else

{

System.out.println("Invalid Account!!");

}

break;

case 3:

System.out.println("Enter the accno:: ");

accno = in.nextInt();

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

{

if(accno==obj[I].accno)

{

break;

}

}

if(I<i)

{

System.out.println("Enter the Withdraw Amt:: ");

bal = in.nextDouble();

try

{

obj[I].withdraw(bal);

}catch(LessBalanceException e)

{

}

}else

{

System.out.println("Invalid Account!!");

}

break;

case 4:

System.out.println("Enter the accno:: ");

accno = in.nextInt();

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

{

if(accno==obj[I].accno)

{

break;

}

}

if(I<i)

{

obj[I].balance();

}else

{

System.out.println("Invalid Account!!");

}

break;

case 5:

System.exit(1);

default:

System.out.println("Error in Choice!!!");

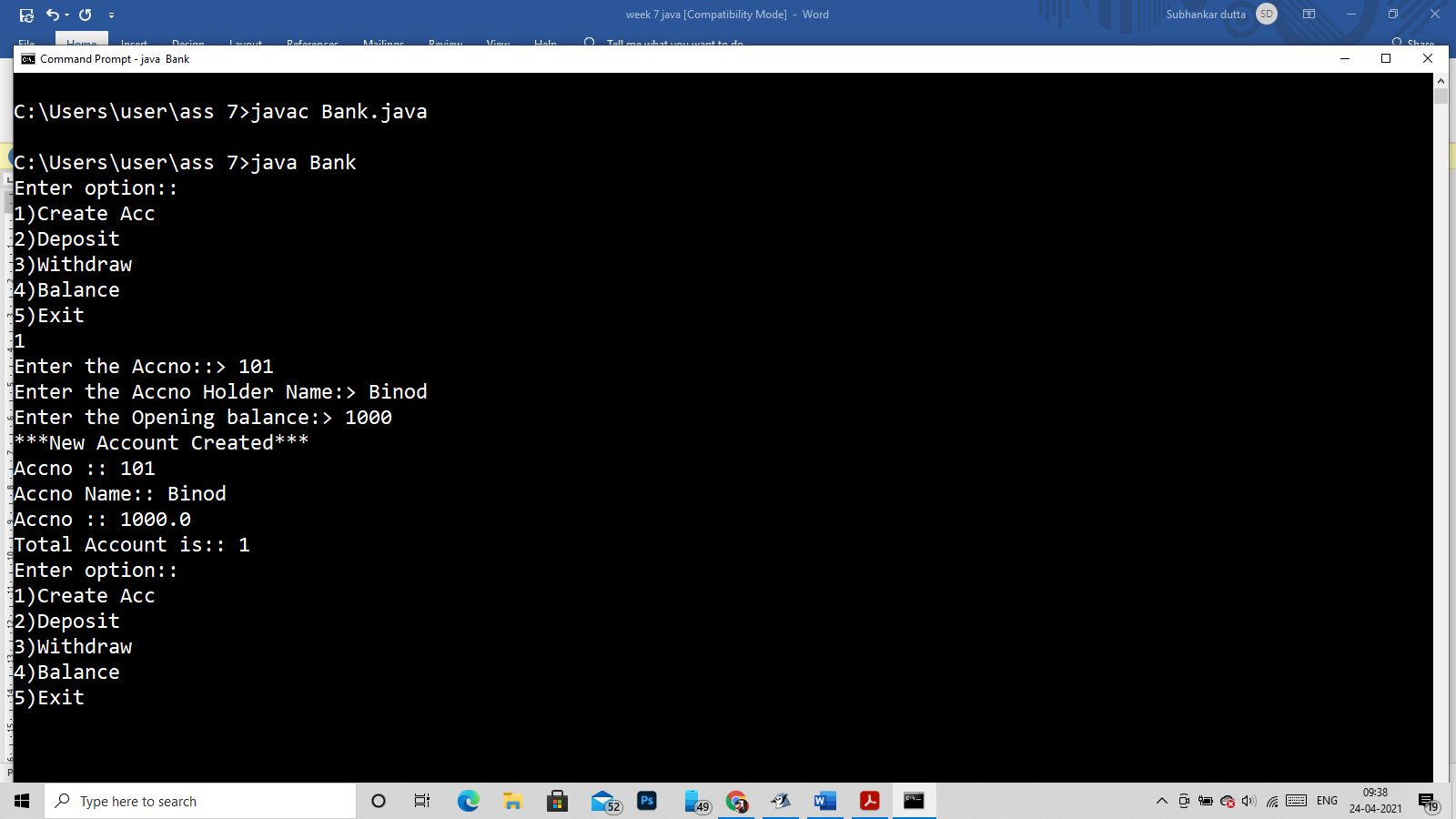
}

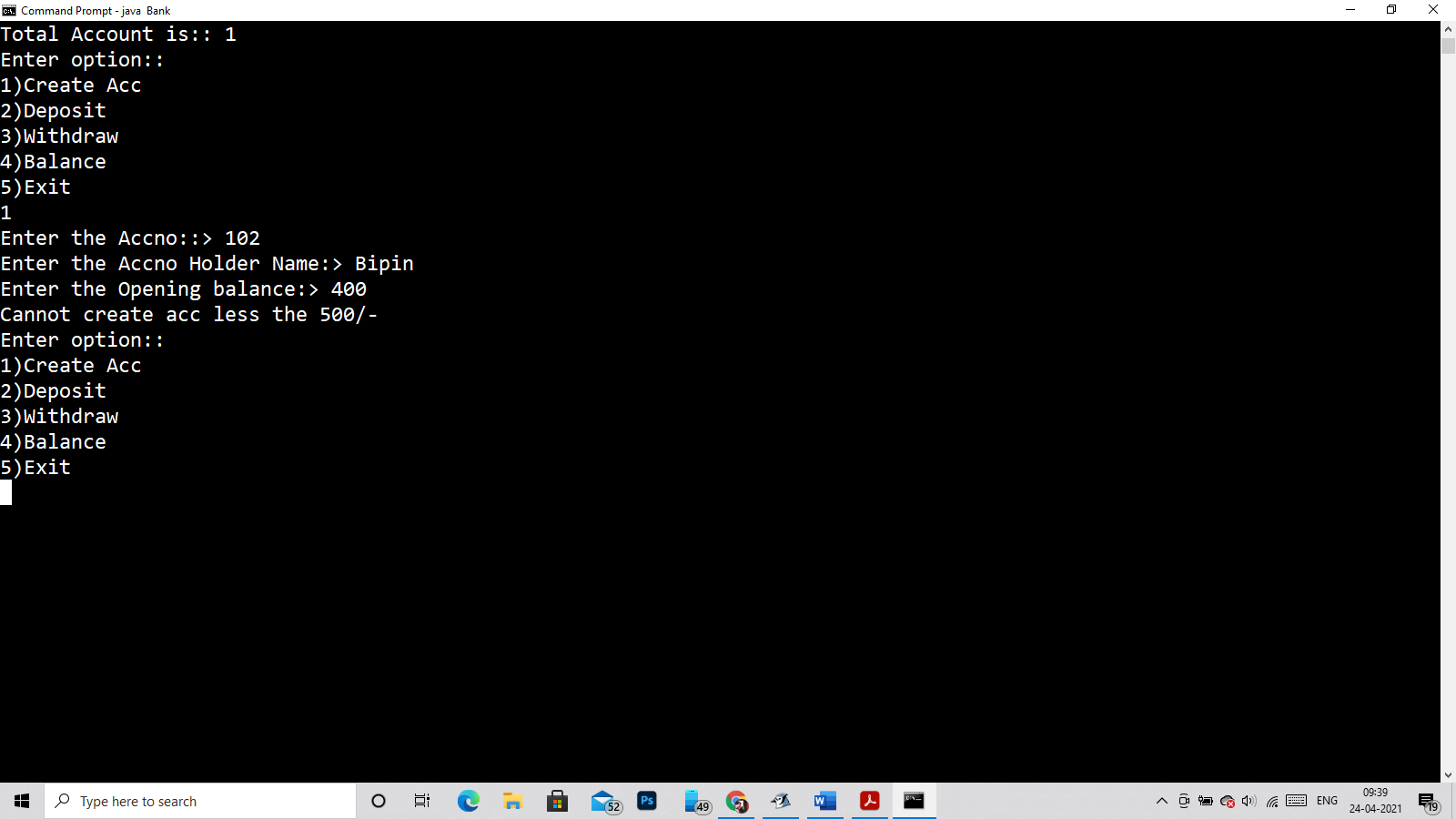
}

}

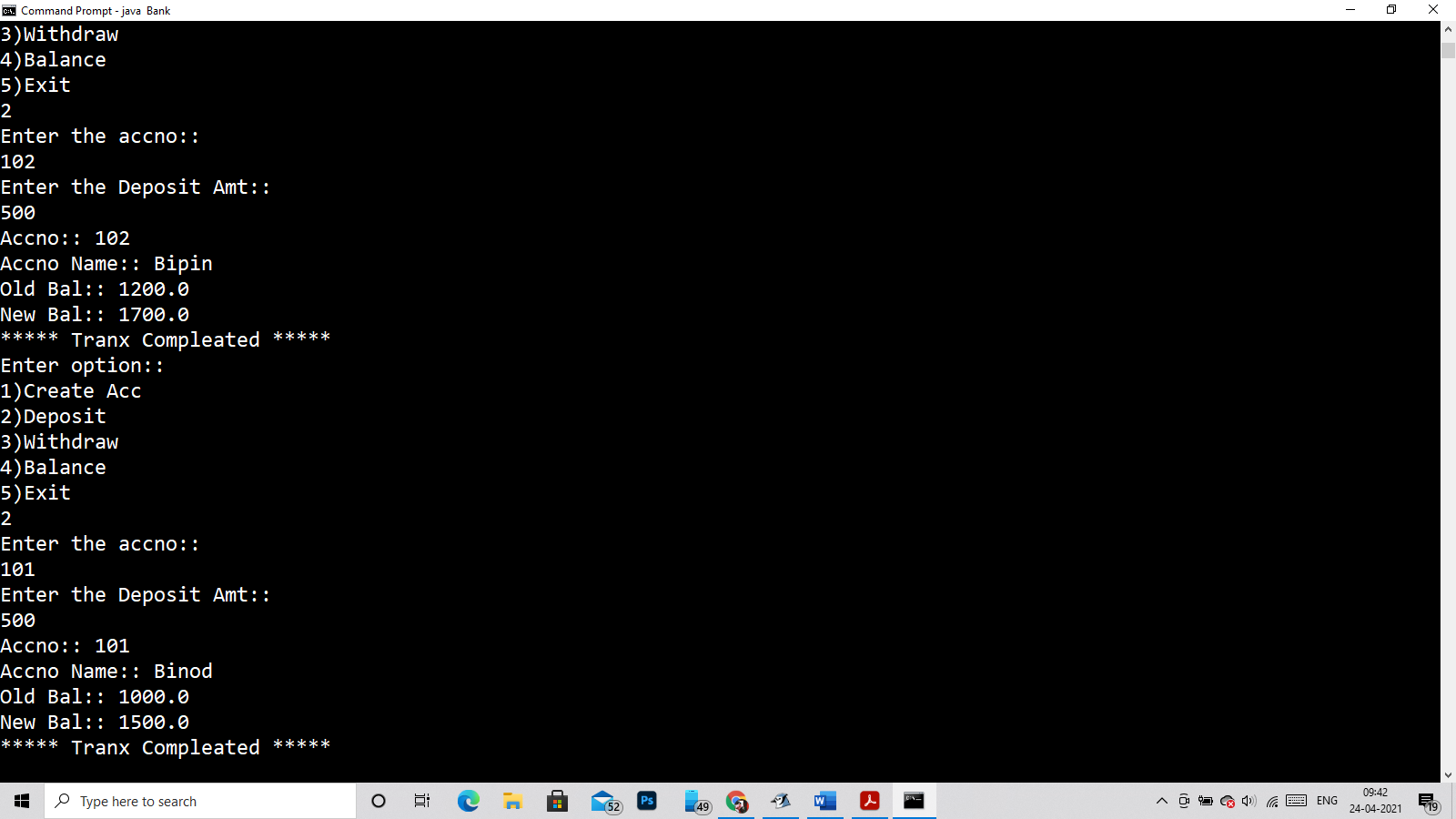
}

Account Creating

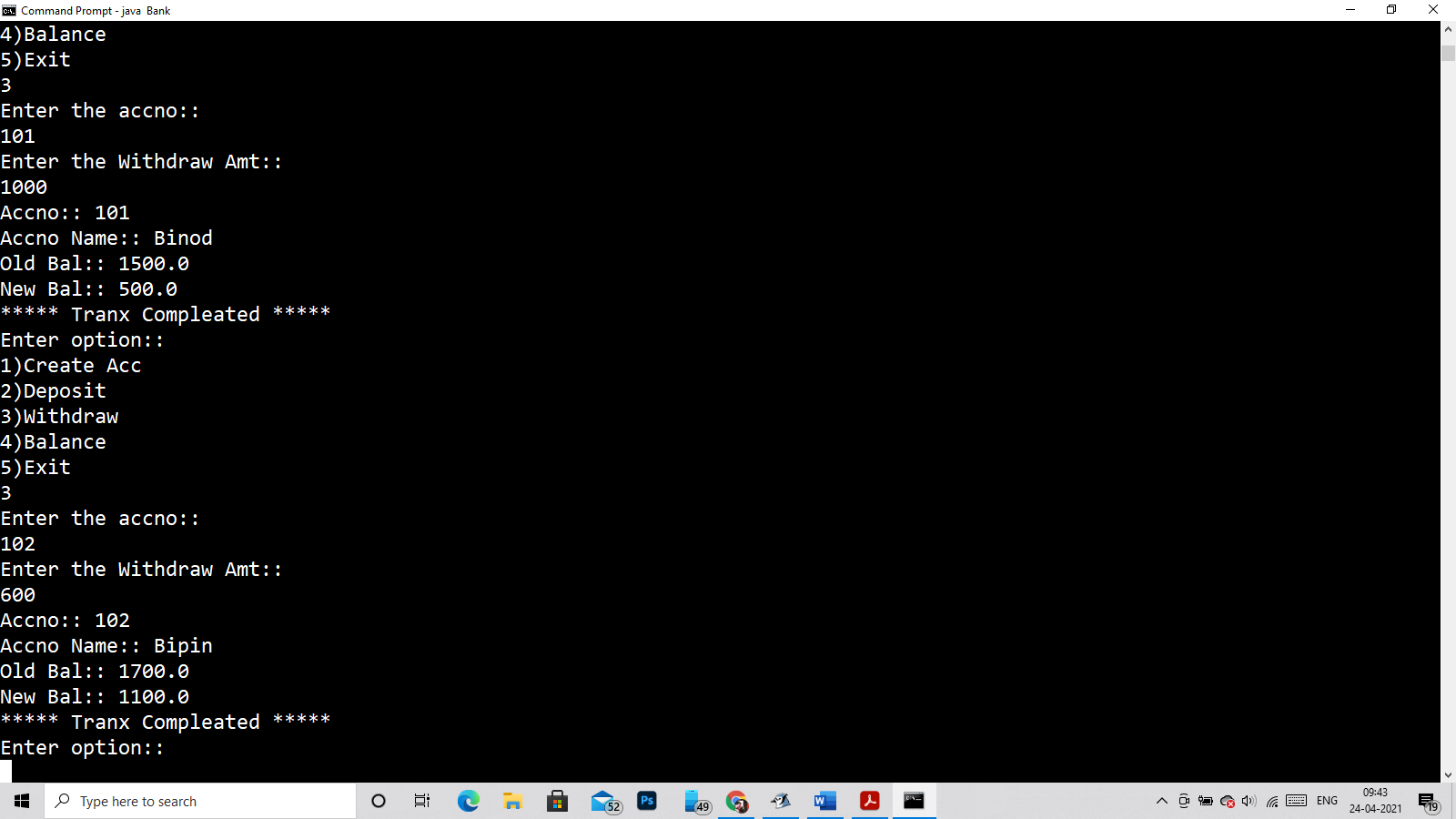




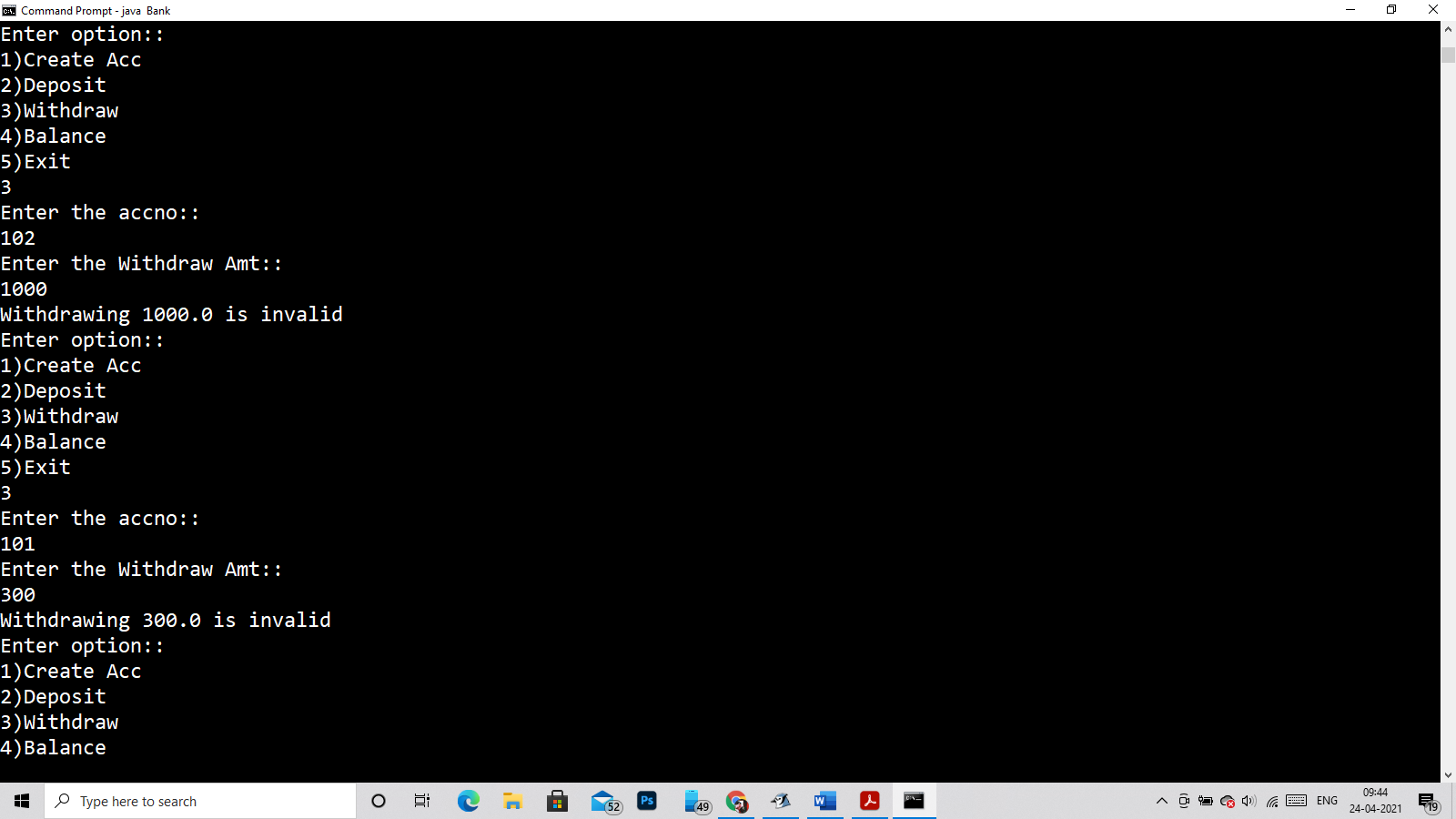
Deposit



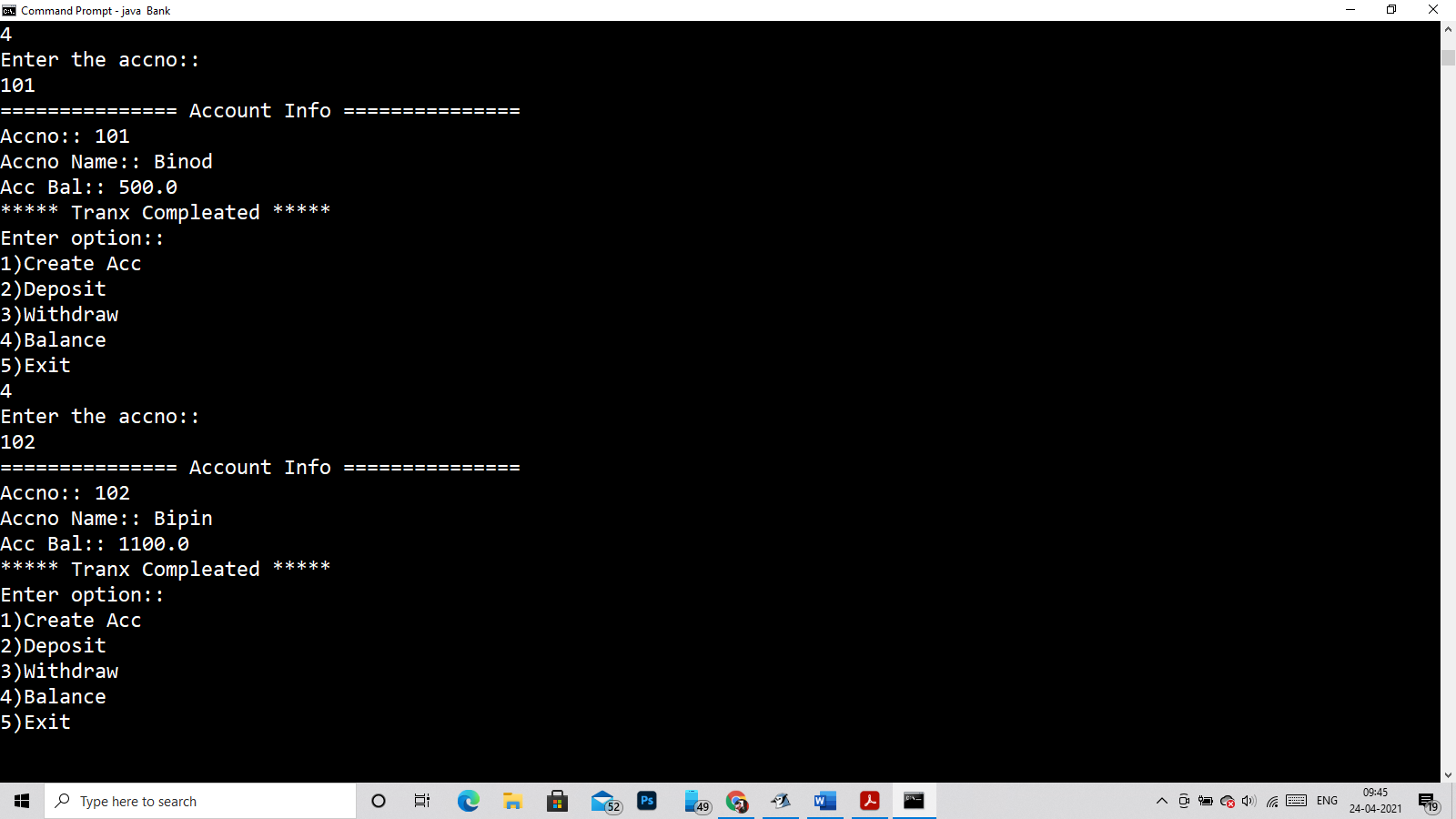
Withdraw (Successful)



Withdraw (Unsuccessful)



Account Info



Q17)

import java.util.\*;

class BookNotFoundException extends Exception

{

public BookNotFoundException(String msg)

{

super(msg);

}

}

public class Libary

{

static void check(String str)throws BookNotFoundException

{

int i;

String Book[]={"Three Man In A Boat","Concept of Physics","IIT Mathematics","Who am i","Gita"};

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

{

if(Book[i].equalsIgnoreCase(str))

{

break;

}

}

if(i<5)

{

System.out.println("Book found!!");

}else

{

throw new BookNotFoundException("Book not found!!!");

}

}

public static void main(String asd[])throws BookNotFoundException

{

Scanner in = new Scanner(System.in);

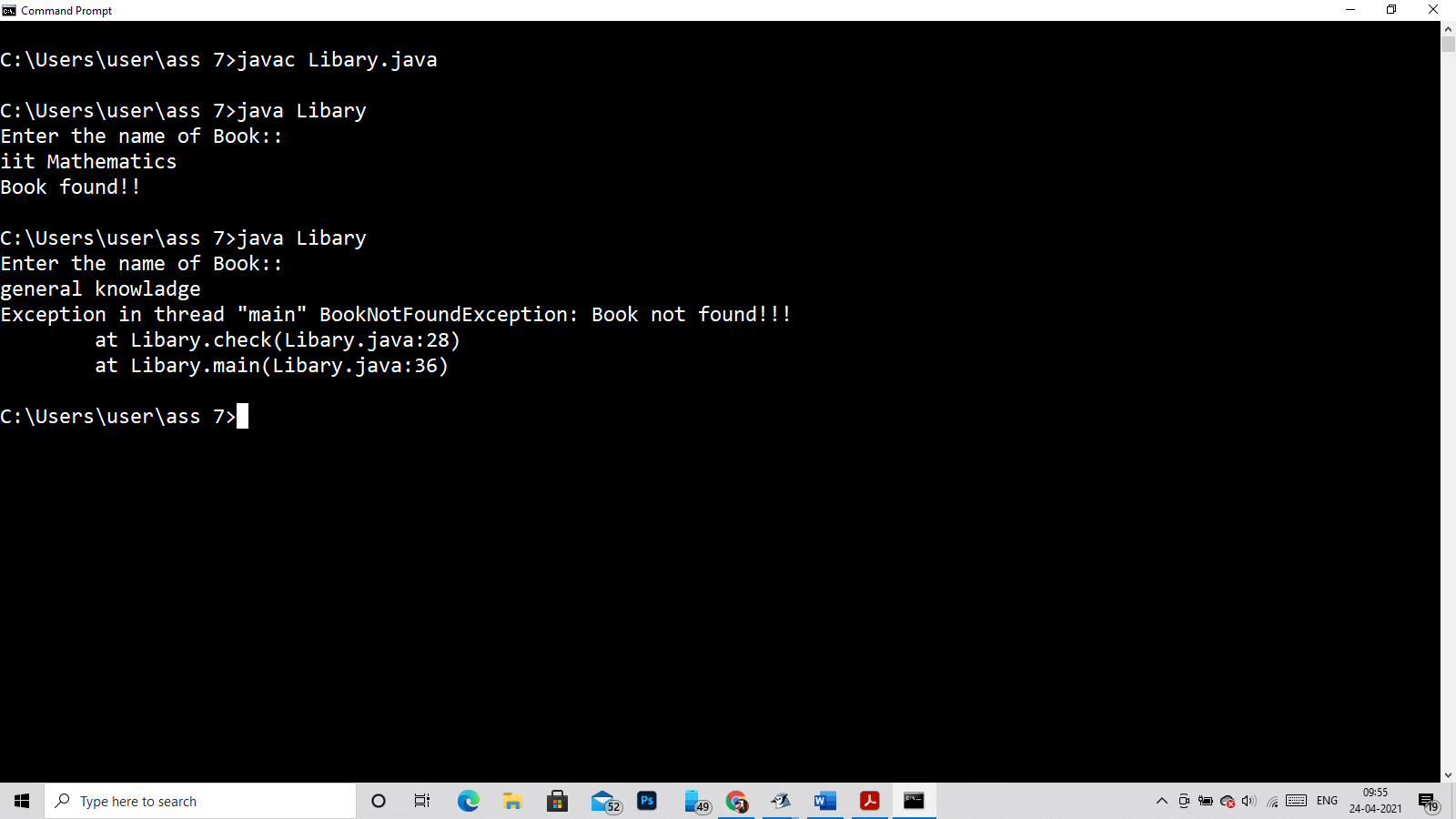
System.out.println("Enter the name of Book::");

String str=in.nextLine();

check(str);

}

}



Q18)

import java.util.\*;

class NotCorrectException extends Exception

{

public NotCorrectException(String msg)

{

super(msg);

}

}

public class Quiz

{

public static void main(String asd[])throws NotCorrectException

{

Scanner in = new Scanner(System.in);

String ans;

System.out.println("Q1. The Capital of India:: ");

ans=in.next();

if(ans.equalsIgnoreCase("Delhi"))

{

System.out.println("Q2. The National Animal:: ");

ans=in.next();

if(ans.equalsIgnoreCase("Tiger"))

{

System.out.println("Q3. The National Flower:: ");

ans=in.next();

if(ans.equalsIgnoreCase("Lotus"))

{

System.out.println("Q4. The National Bird:: ");

ans=in.next();

if(ans.equalsIgnoreCase("Peacock"))

{

System.out.println("Q5. The National Fruit:: ");

ans=in.next();

if(ans.equalsIgnoreCase("Mango"))

{

System.out.println("good! The answer is correct");

}

else

{

throw new NotCorrectException("Worng answer");

}

}

else

{

throw new NotCorrectException("Worng answer");

}

}

else

{

throw new NotCorrectException("Worng answer");

}

}

else

{

throw new NotCorrectException("Worng answer");

}

}

else

{

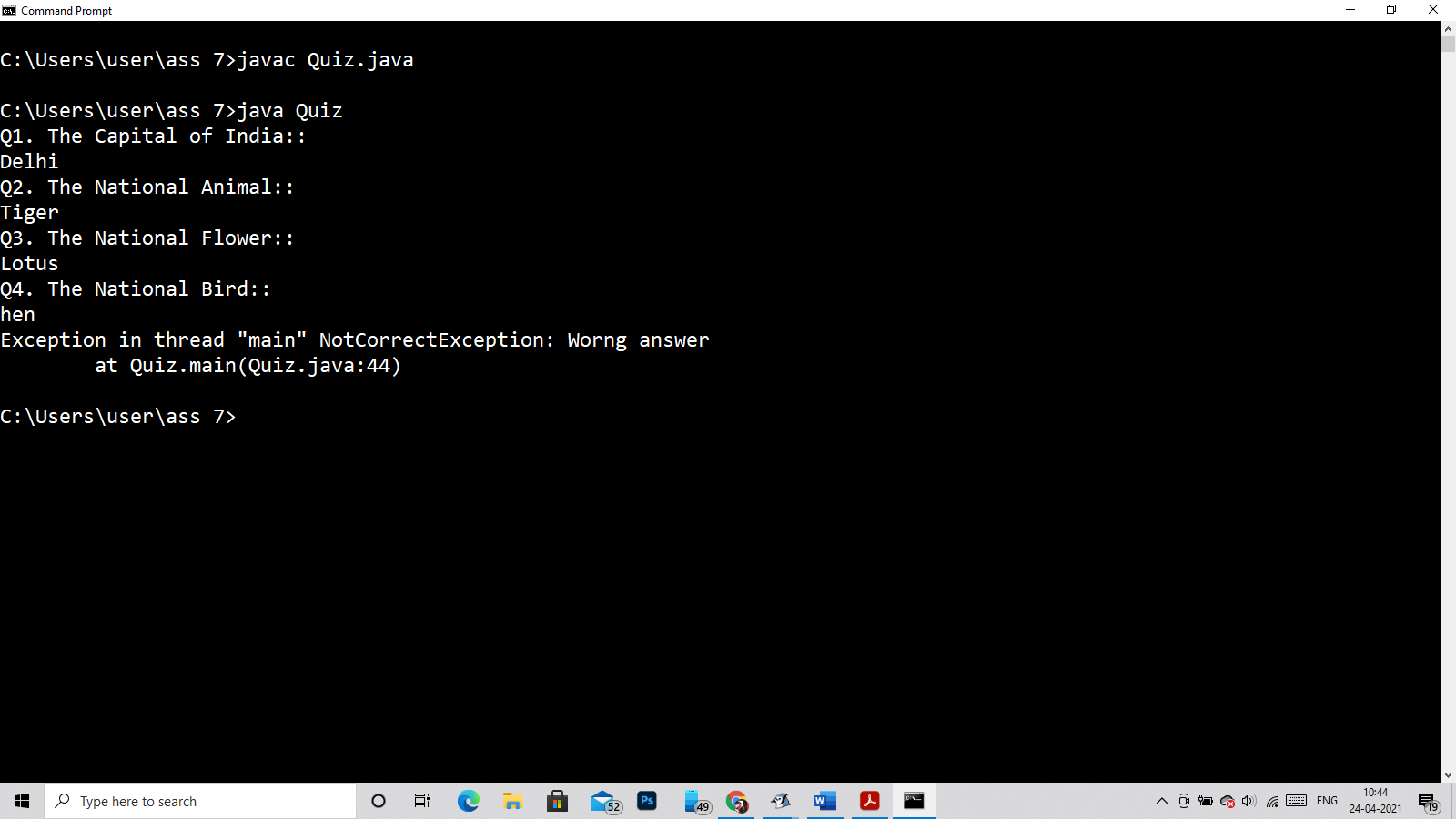
throw new NotCorrectException("Worng answer");

}

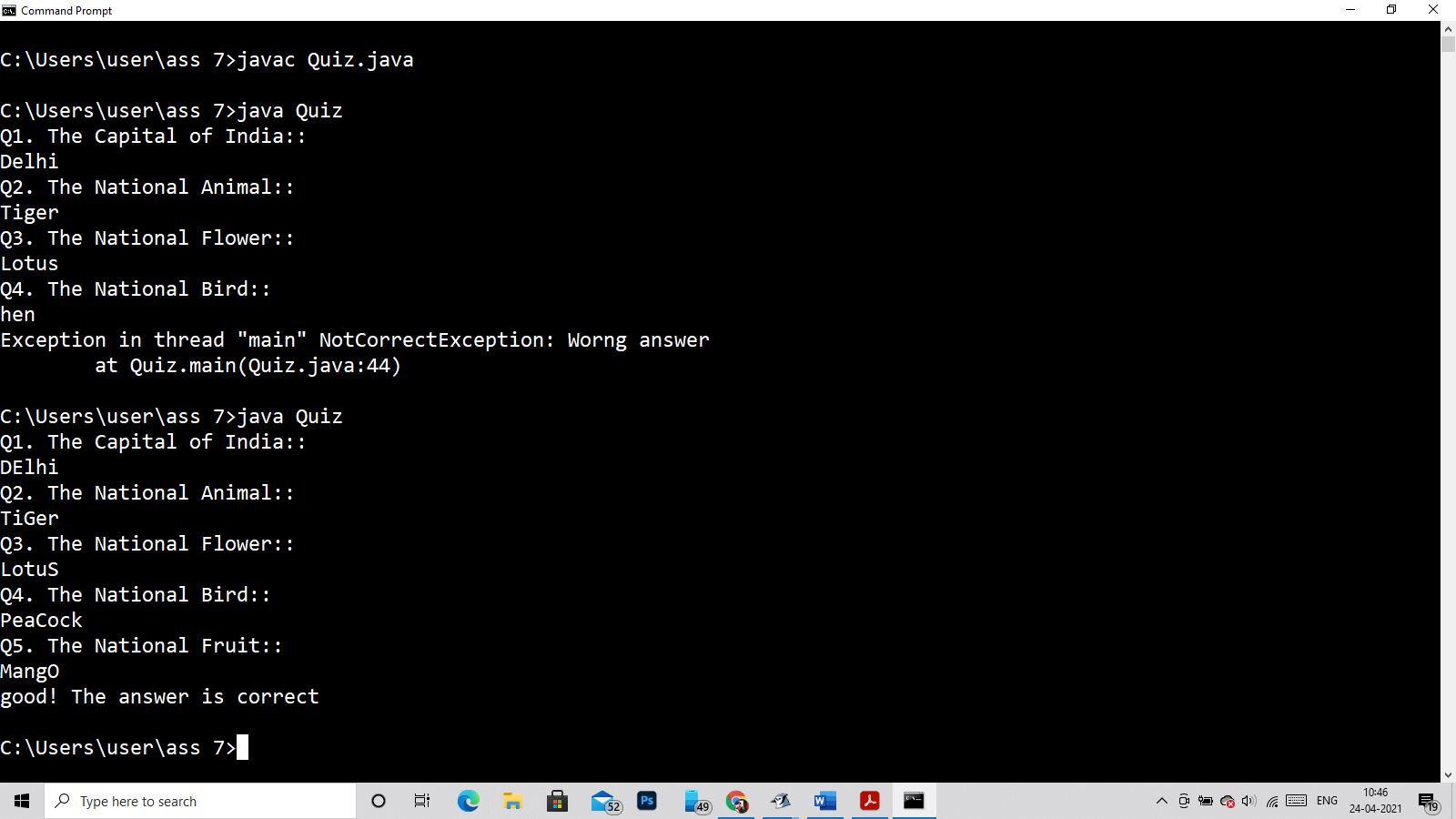
}

}

Exception Case



Correct Case



Q19)

class userNameException extends Exception

{

public userNameException(String msg)

{

super(msg);

}

}

public class Username

{

static void check(String unm,String pass)throws userNameException

{

if(unm.length()<6)

{

throw new userNameException("username less then 6 char");

}

else

if(unm.equals("BipinKumar123")&&pass.equals("Sub@2019"))

{

System.out.println("Successfully login!!!");

}

else

{

throw new userNameException("username / Password Wrong!!");

}

}

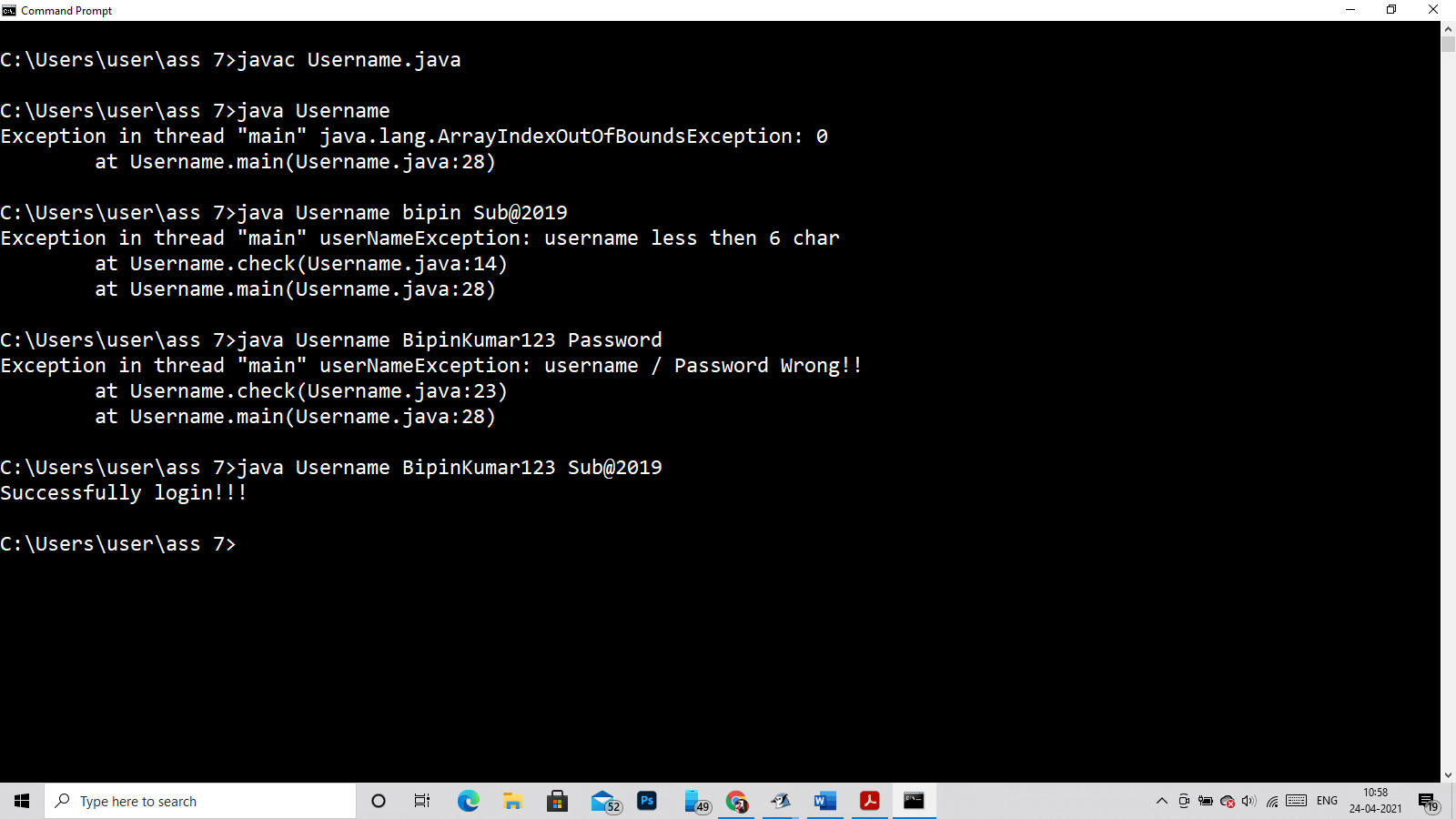
public static void main(String args[])throws userNameException

{

check(args[0],args[1]);

}

}



Q20)

class AuthenticationFailure extends Exception

{

public AuthenticationFailure(String msg)

{

super(msg);

}

}

public class Username

{

static void check(String unm,String pass)throws AuthenticationFailure

{

if(unm.equals("BinodKumar")&&pass.equals("Sub@2019"))

{

System.out.println("Successfully login!!!");

}

else

{

throw new AuthenticationFailure("username / Password Wrong!!");

}

}

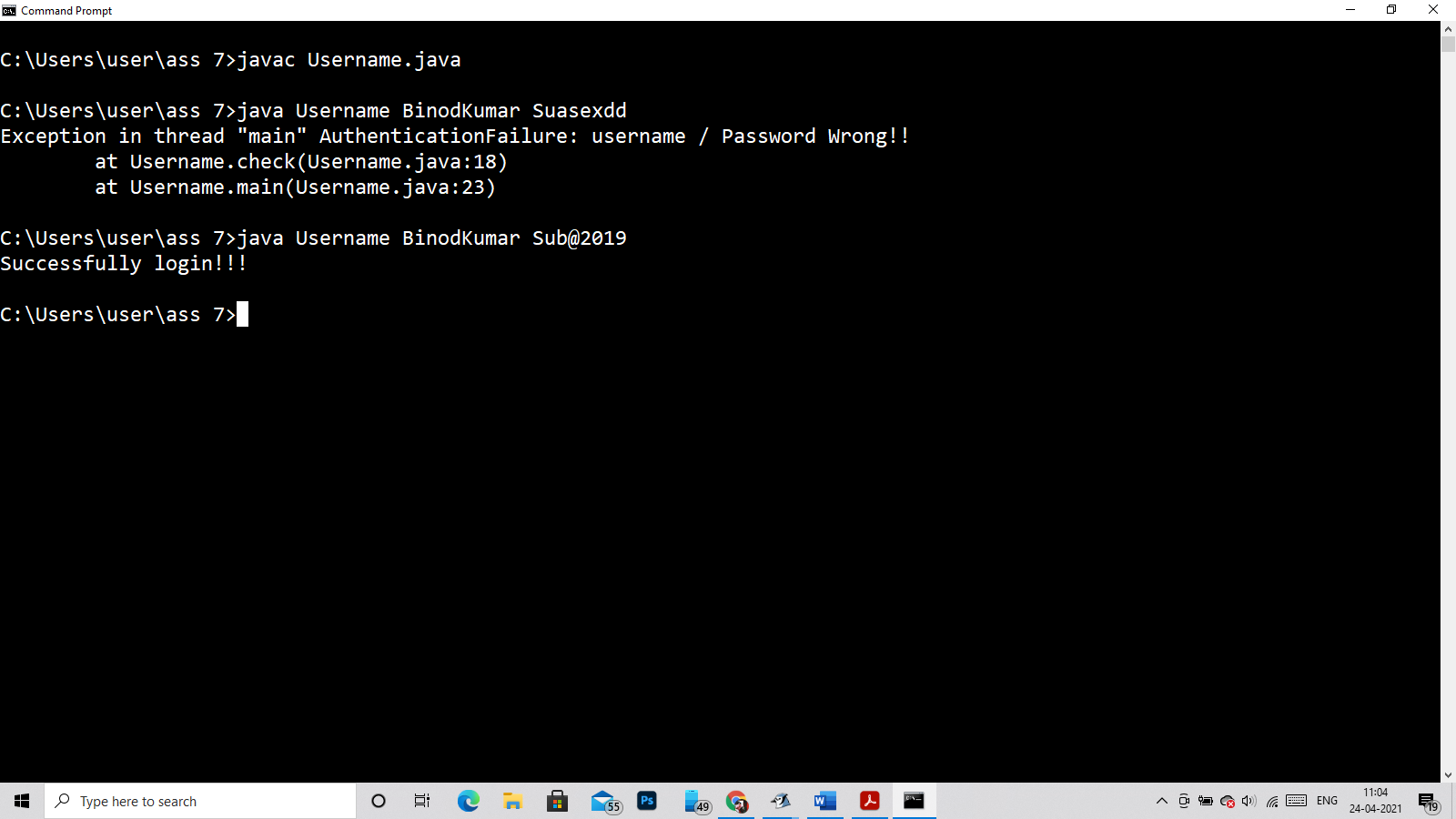
public static void main(String args[])throws AuthenticationFailure

{

check(args[0],args[1]);

}

}



Q21)

import java.util.\*;

class AgeNegativeException extends Exception {

public AgeNegativeException(String msg) {

super(msg);

}

}

public class Agecheck {

public static void main(String[] args) {

Scanner in = new Scanner(System.in);

System.out.print("Enter your name :: ");

String name = in.nextLine();

System.out.print("Enter your age :: ");

int age = in.nextInt();

try {

if(age < 0)

throw new AgeNegativeException("Age can't be (-)ve !! -\_-");

else

System.out.println("Valid age !!!");

}

catch (AgeNegativeException e) {

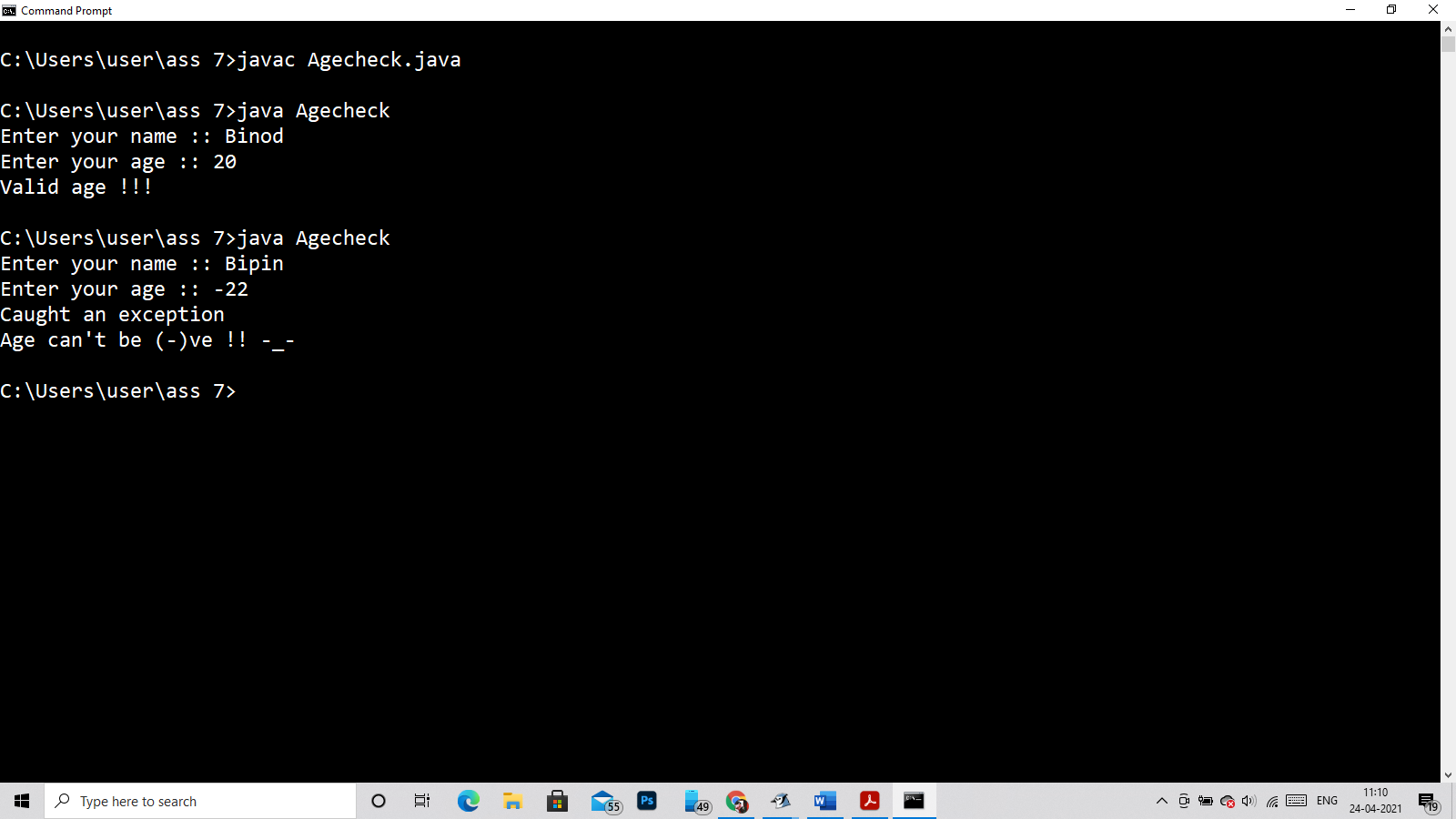
System.out.println("Caught an exception");

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

}

}

}



Q22)

import java.util.\*;

class NegitiveException extends Exception {

public NegitiveException(String msg) {

super(msg);

}

}

public class Numcheck {

public static void main(String[] args) {

Scanner in = new Scanner(System.in);

System.out.print("Enter your Number :: ");

int num = in.nextInt();

try {

if(num < 0)

throw new NegitiveException(" can't be (-)ve !! -\_-");

else

System.out.println("Valid number !!!");

}

catch (NegitiveException e) {

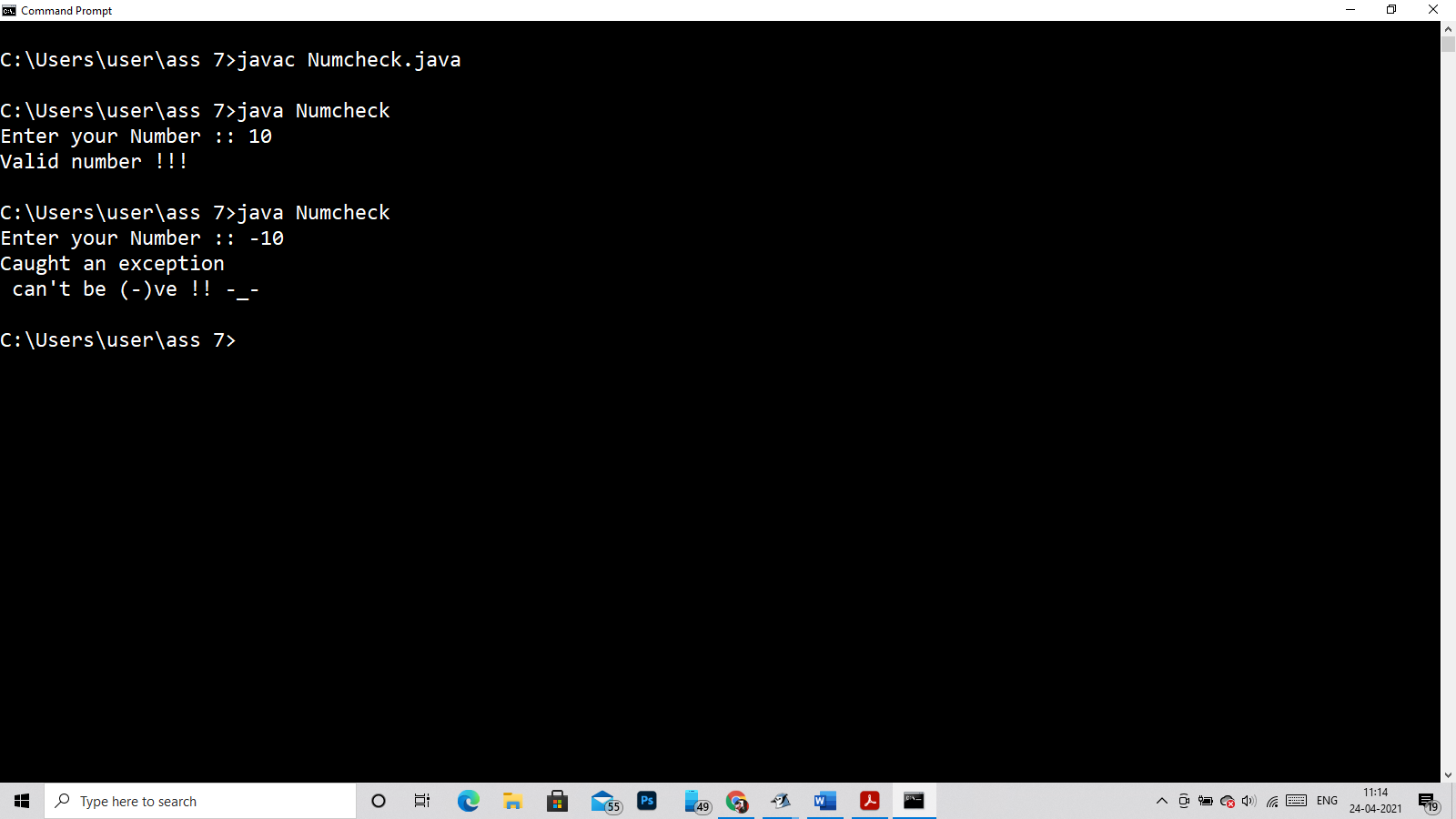
System.out.println("Caught an exception");

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

}

}

}



Q23)

import java.util.\*;

class StringMismatchException extends Exception

{

public StringMismatchException(String msg)

{

super(msg);

}

}

public class Strck

{

public static void main(String asd[])throws StringMismatchException

{

Scanner in = new Scanner(System.in);

String ans;

System.out.println("Enter The String");

ans=in.next();

if(ans.equalsIgnoreCase("Bipin"))

{

System.out.println("Successfull !!!! \*\_\*");

}

else

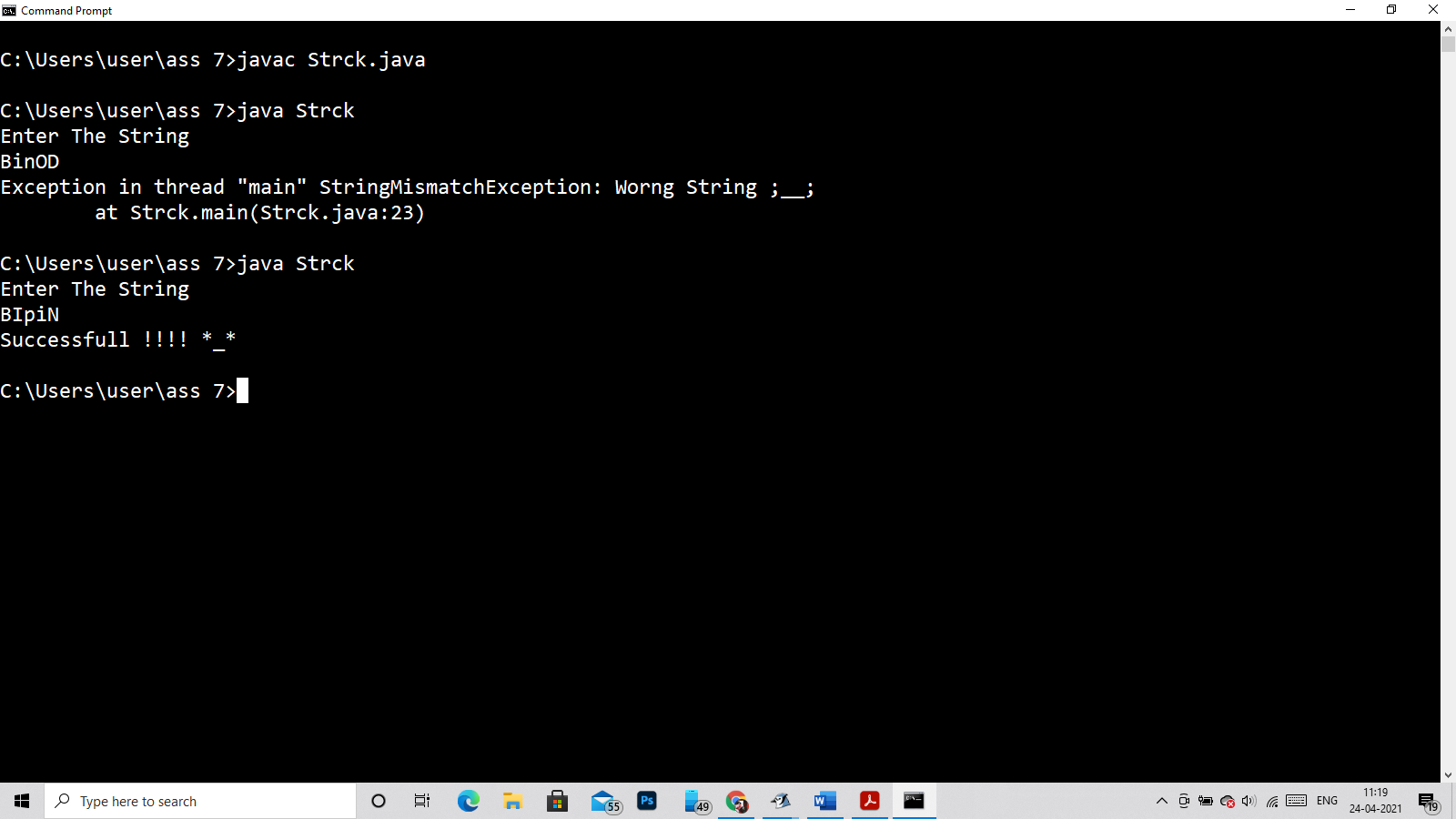
{

throw new StringMismatchException("Worng String ;\_\_;");

}

}

}



Q24)

import java.util.\*;

class pushException extends Exception

{

public pushException(String msg)

{

super(msg);

}

}

class popException extends Exception

{

public popException(String msg)

{

super(msg);

}

}

public class stack

{

private static int stack[]=new int[5];//5

private static int top=-1;//-1

private static void push(int n)throws pushException

{

if(top>=4)//n-1

{

throw new pushException("Overflow!!!");

}else

{

top=top+1;

stack[top]=n;

}

}

private static void pop()throws popException

{

if(top==-1)

{

throw new popException("stack Underflow");

}

else

{

int temp=top;

System.out.println("The value deleted is :"+stack[temp]);

top=top-1;

}

}

private static void display()

{

if(top==-1)

{

System.out.println("stack Underflow");

}

else

{

int temp=top,i=0;

while(temp>=0)

{

System.out.println(stack[temp]);

temp--;

}

}

}

private static void garbage()

{

int t=top;

System.out.println("After deletion top+1 display: "+stack[t+1]);

}

public static void main(String args[])

{

Scanner in = new Scanner(System.in);

int ch,num;

while(true)

{

System.out.println("Enter your choice\n1)push\t2)pop\t3)display\t4)exit");

ch=in.nextInt();

if(ch==1)

{

System.out.println("Enter the number to been pushed:");

num=in.nextInt();

try

{

push(num);

}catch(pushException e)

{

System.out.println("Exception found!! "+e);

}

}

else if(ch==2)

{

try

{

pop();

}catch(popException e)

{

System.out.println("Exception found!! "+e);

}

}

else if(ch==3)

{

display();

}

else if(ch==4)

{

break;

}

else

{

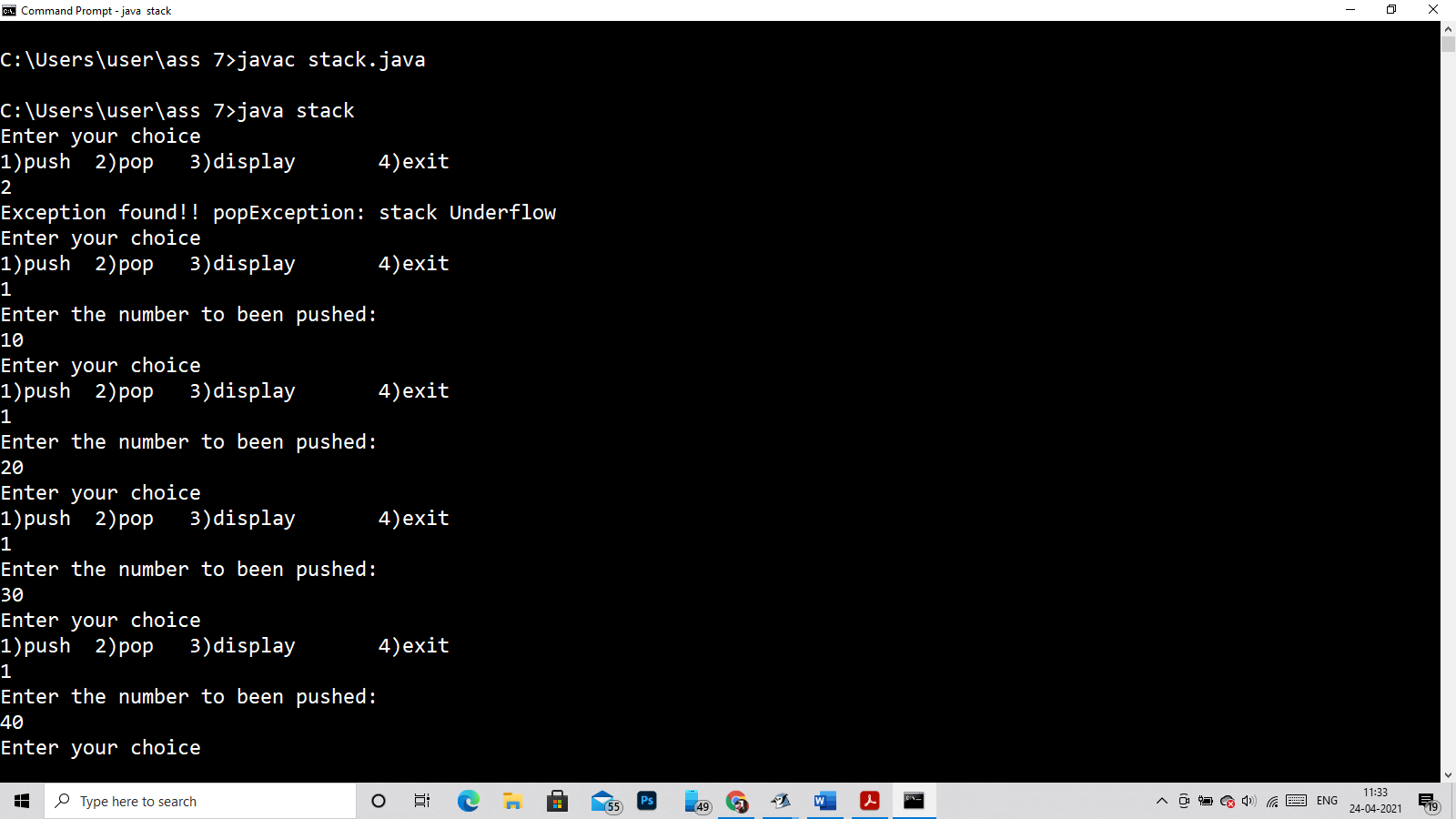
System.out.println("Wrong input");

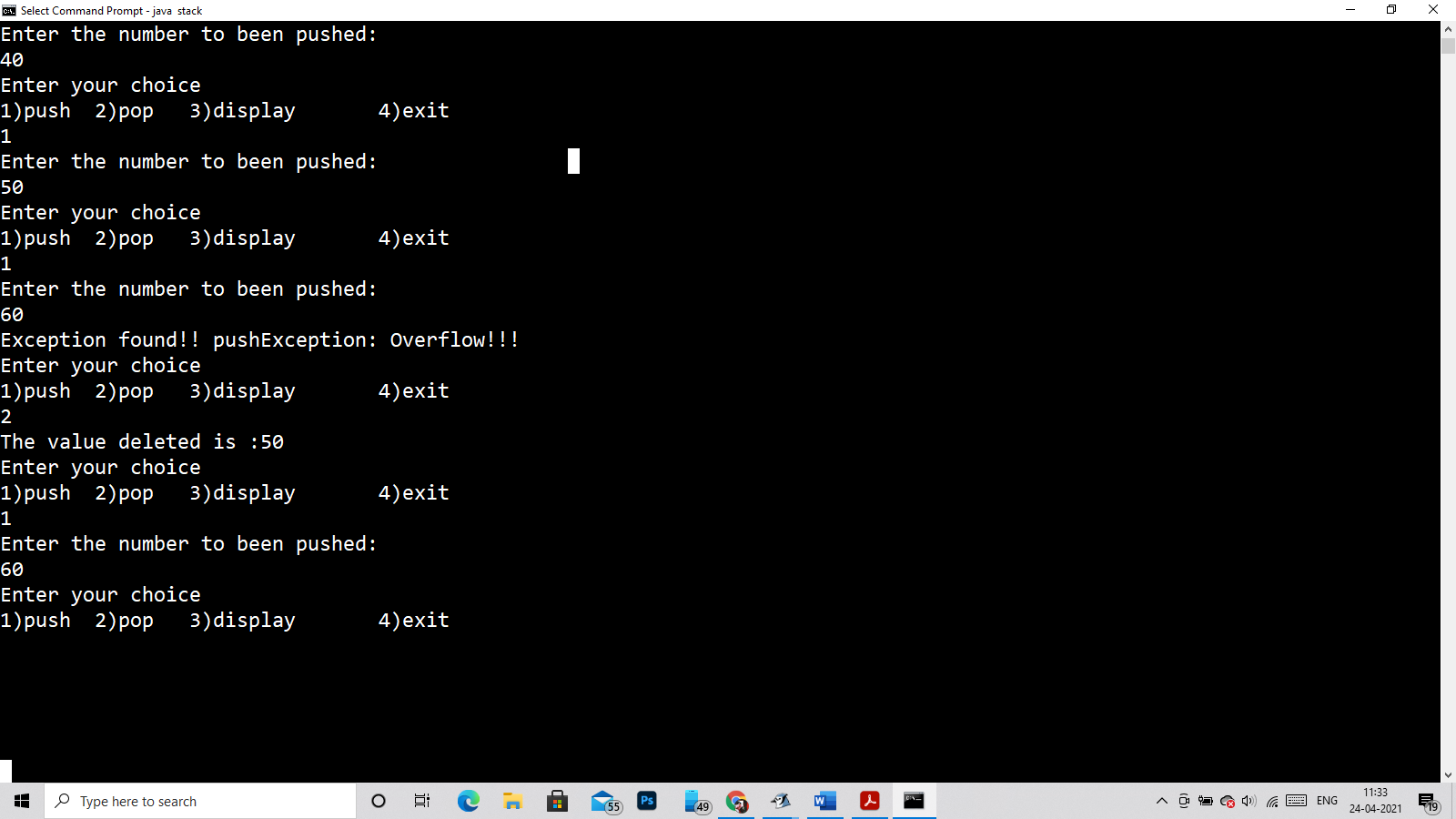
}

}

}

}





Q25)

import java.util.\*;

class ScoreException extends Exception

{

public ScoreException(String msg)

{

super(msg);

}

}

class Students

{

int StuID;

double score;

public Students(int StuID,double score)throws ScoreException

{

if(score<0 || score>100)

{

throw new ScoreException("Invalid Marks");

}

else

{

this.StuID=StuID;

this.score=score;

System.out.println("Student ID:: "+StuID);

System.out.println("Score "+score);

}

}

void display()

{

System.out.println("ID:: "+this.StuID+" marks:: "+this.score);

}

}

public class Main

{

static int i=0;

public static void main(String args[])

{

Scanner in = new Scanner(System.in);

int ch,sid;

double score;

Students obj[] = new Students[10];

while(true)

{

System.out.println("Enter your choice:: \n1)Input New Student \n2)Display all \n3)Exit");

ch=in.nextInt();

switch(ch)

{

case 1:

System.out.println("Enter the Student ID");

sid= in.nextInt();

System.out.println("Enter the Score Marks:: ");

score = in.nextDouble();

try

{

obj[i]= new Students(sid,score);

i++;

}catch(ScoreException e)

{

System.out.println("Error caught "+e);

}

break;

case 2:

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

{

obj[k].display();

}

break;

case 3:

System.exit(1);

default:

System.out.println("Error Input -\_-");

break;

}

}

}

}

