

OBJECT ORIENTED JAVA LAB RECORD

NAME :- VALLISHA M

USN :- 1BM19CS177

SECTION :- 3D

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.

Writeup:

VALLISHAM
IBM19CS177
09-10-2020

discuss

- Roots of Quadratic Equation

```
import java.util.Scanner;
public class roots
{
    double a,b,c,d;
    roots(double a, double b, double c, double d)
    {
        this.a=a;
        this.b=b;
        this.c=c;
    }
    public static void main(String[] args)
    {
        Scanner in = new Scanner(System.in);
        System.out.println("Enter a,b,c : ");
        double a1 = in.nextDouble();
        double b1 = in.nextDouble();
        double c1 = in.nextDouble();
        in.close();
        roots obj = new roots(a1,b1,c1);
        obj.d = obj.discriminant();
        if(obj.d > 0)
            obj.distinct();
        else if (obj.d == 0)
            obj.equal();
        else
            obj.imaginary();
    }
}
```

```
public double discriminant()  
{  
    return (b*b - math.sqrt(4*a*c));  
}
```

```
public void distinct()  
{
```

```
    double x = Math.sqrt(d);  
    double r1 = (-1*b + x) / (2*a);
```

```
    double r2 = (-1*b - x) / (2*a);
```

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

```
    System.out.println("First root : "+r1);
```

```
    System.out.println("Second root : "+r2);
```

```
}
```

```
public void equal()  
{
```

```
    double x = Math.sqrt(d);
```

```
    double r1 = (-1*b + x) / (2*a);
```

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

```
    System.out.println("Root : "+r1);
```

```
}
```

```
public void imaginary()  
{
```

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

```
    double x = Math.sqrt(-d);
```

```
    double r = (-b) / (2*a);
```

```
    double i = Math.abs(x / (2*a));
```

classmate

Date _____

Page _____

if ($a == 0$) $r = 0$;

System.out.println ("The roots are : " + r + "
" + (-) i + " + " + i + ")";

}

Output;

```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Vallisha>cd C:\Users\Vallisha\Desktop\OOJ Lab\09-10-2020

C:\Users\Vallisha\Desktop\OOJ Lab\09-10-2020>java roots
Enter a, b, c :
2
3
2

The roots are imaginary
The roots are : -0.75 <+/-> i * 0.6614378277661477

C:\Users\Vallisha\Desktop\OOJ Lab\09-10-2020>java roots
Enter a, b, c :
1
2
1

The roots are real and equal;

Root : -1.0

C:\Users\Vallisha\Desktop\OOJ Lab\09-10-2020>java roots
Enter a, b, c :
1
-3
2

The roots are real and distinct;

First Root : 2.0
Second Root : 1.0
```

- 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.

Writeup:

VALLISHA.M, 1BM19CS177, 16-10-2020

Program to calculate SGPA

```
import java.util.Scanner;  
class Student {  
    int numberofcourses;  
    String usn, name;  
    int creditsArray[];  
    int marksArray[];  
    int gradesArray[];  
    double SGPA;  
    Student() { SGPA = 0.0D; }  
    void input()  
    {  
        Scanner in = new Scanner(System.in);  
        System.out.print("Enter your  
        name : ");  
        name = in.next();  
        System.out.print("Enter number of  
        courses taken : ");  
        numberofcourses = in.nextInt();  
    }
```

```
System.out.print("Enter your user % ");  
user = in.nextInt();  
marksArray = new int[numberOfCourses];  
creditsArray = new int[numberOfCourses];  
gradesArray = new int[numberOfCourses];  
  
int i = 0;  
System.out.println();  
for (i = 0; i < numberOfCourses; i++)  
{  
    System.out.print(" Enter credits for  
course " + (i + 1) + " : ");  
    creditsArray[i] = in.nextInt();  
    System.out.print(" Enter marks obtained  
in " + (i + 1) + " : ");  
    marksArray[i] = in.nextInt();  
}  
in.close();  
  
void computeGradesArray()  
{  
    int i = 0;  
    for (i = 0; i < numberOfCourses; i++)  
    {  
        if (marksArray[i] == 10)  
        {  
            gradesArray[i] = 10;  
            continue;  
        }  
    }  
}
```

```
If (marksArray[i] >= 50)
{
    gradesArray[i] = (marksArray[i]/10) + 1;
    continue;
}

If (marksArray[i] >= 35)
{
    gradesArray[i] = 4;
    continue;
}

gradesArray[i] = 0;
}

void computeSGPA()
{
    int i = 0;
    int netredits = 0;
    for(i = 0; i < numberofCourses; i++)
    {
        SGPA = SGPA + creditsArray[i] * gradeArray[i];
        netredits = netredits + creditsArray[i];
    }
    SGPA = SGPA / netredits;
}
```

```
void display()
```

```
{
```

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

```
    System.out.print("USN : " + usn);
```

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

```
    System.out.println("Number of courses  
taken this semester : " + numberofcourses);
```

```
    int i = 0;
```

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

```
        System.out.print("Marks scored
```

```
in course " + (i + 1) + " is " + marksArray[i]);
```

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

```
}
```

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

```
{
```

```
    Student object = new obj Student();
```

```
    object.Input();
```

```
    object.computeGradesArray();
```

```
    object.computeSGPA();
```

```
    object.display();
```

```
}  
}
```

Output:

```
C:\WINDOWS\system32\cmd.exe
C:\Users\Vallisha\Desktop\OOJ Lab\SGPA 16-10-2020>java Student
Enter your name : Vallisha M
Enter your USN : 1BM19CS177
Enter number of courses taken(MAX 10) : 5

Enter Credits for course 1 : 5
Enter marks obtained in course 1 : 80

Enter Credits for course 2 : 5
Enter marks obtained in course 2 : 35

Enter Credits for course 3 : 4
Enter marks obtained in course 3 : 95

Enter Credits for course 4 : 6
Enter marks obtained in course 4 : 100

Enter Credits for course 5 : 2
Enter marks obtained in course 5 : 30

USN : 1BM19CS177
Name : Vallisha M
Number of Courses taken this semester : 5
Marks scored in course 1 is 80
Marks scored in course 2 is 35
Marks scored in course 3 is 95
Marks scored in course 4 is 100
Marks scored in course 5 is 30
SGPA : 7.5

C:\Users\Vallisha\Desktop\OOJ Lab\SGPA 16-10-2020>
```

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.

Writeup;

CLASSMATE
Date _____
Page _____

Vallisha - M
IBM19CS177

Book Program

```
public class Book {
    private String name, author;
    private int numPages, price;

    Book() {
        this.name = null;
        this.author = null;
        this.numPages = 0;
        this.price = 0;
    }

    Book (String name, String author, int price, int numPages) {
        this.author = author;
        this.name = name;
        this.price = price;
        this.numPages = numPages;
    }

    @Override
    public String toString() {
        return String.format ("Book Name : " + name + " Author Name : " + author +
            " Price : " + price + " Number of Pages : " + numPages);
    }
}
```

public class Main

import java.util.Scanner;
public class Main {

public static void main(String args[]) {

Scanner in = new Scanner(System.in);
System.out.print("Enter number of
books : ");

int n = in.nextInt();

Book books[] = new Book[n];

String name, authorName;

int numPages, price;

int i;

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

System.out.print("Enter
name of book " + (i + 1) + ": ");

name = in.next();

System.out.print("Enter name of
author of book " + (i + 1) + ": ");

authorName = in.next();

System.out.print("Enter price of
book " + (i + 1) + ": ");

price = in.nextInt();

System.out.print("Enter number
of pages in book " + (i + 1) + ": ");

numPages = in.nextInt();

```
books[i] = new Book(name, author, name  
piece, num - pages);
```

```
}  
System.out.println("The details of  
books are; ");
```

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

```
System.out.println("Book " + (i + 1) + "
```

```
System.out.println(books[i].toString()  
+ "String());
```

```
" --END--
```

Output;

```
C:\WINDOWS\system32\cmd.exe
C:\Users\Vallisha\Desktop\OOJ Lab\book, bank, abstractArea\book>java Main
Enter number of books : 2

Enter name of book 1 : Harry
Enter name of author of book 1 : Rowling
Enter price of book 1 : 1200
Enter number of pages in book 1 : 562

Enter name of book 2 : Protector
Enter name of author of book 2 : Davis
Enter price of book 2 : 500
Enter number of pages in book 2 : 2000

The details of books are ;

Book 1
Book Name : Harry
Author Name : Rowling
Book Price : 1200
Number of Pages : 562

Book 2
Book Name : Protector
Author Name : Davis
Book Price : 500
Number of Pages : 2000
C:\Users\Vallisha\Desktop\OOJ Lab\book, bank, abstractArea\book>
```

- 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.

Writeup:

```
abstract class Area {  
    int dim1, dim2;  
    Area (int dim1, int dim2){  
        this.dim1 = dim1;  
        this.dim2 = dim2;  
    }  
    abstract void printArea();  
}  
public class Triangle extends Area {  
    Triangle (int dim1, int dim2){  
        super(dim1, dim2);  
    }  
    void printArea(){  
        double area = dim1 * dim2 / 2;  
        System.out.println("Area of  
        triangle is " + area);  
    }  
}
```

```

public class Rectangle extends Area {
    Rectangle (int dim1, int dim2) {
        super(dim1, dim2);
    }
    void printArea() {
        double area = dim1 * dim2;
        System.out.println ("Area of rectangle
                            is " + area);
    }
}

public class Circle extends Area {
    Circle (int dim) {
        super();
        super(dim, dim);
    }
    void printArea() {
        double area = 3.142 * dim * dim;
        System.out.println ("Area of circle is
                            " + area);
    }
}

import java.util.Scanner;
public class Main {
    public static void main (String args[])
    {
        Scanner sc = new Scanner (System
                                .in);
        System.out.println ("Enter 1 for Triangle");
        System.out.println ("Enter 2 for Rectangle");
    }
}

```

```
System.out.println("Enter 3 for circle: ");
int choice = sc.nextInt();
switch(choice){
```

case 1 of

```
    System.out.println("Enter height of
triangle : ");
```

```
    dim1 = in.nextInt();
```

```
    System.out.print("Enter base of
triangle : ");
```

```
    dim2 = in.nextInt();
```

```
    Triangle triangle = new Triangle(dim1,
```

```
    dim2);
```

```
    triangle.printArea();
```

```
    break;
```

```
}
```

case 2 : {

```
    System.out.println("Enter width of
rectangle : ");
```

```
    dim1 = sc.nextInt();
```

```
    System.out.println("Enter length of
rectangle : ");
```

```
    dim2 = sc.nextInt();
```

```
    Rectangle rectangle = new Rectangle
(dim1, dim2);
```

```
    rectangle.printArea();
```

```
    break;
```

```
}
```

case 3: {

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

dim1 = in.nextInt();

Circle circle = new Circle(dim1);

circle.printArea();

break;

}

default: {

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

break;

}

} sc.close();

}

}

Output;

```
C:\Users\Vallisha\Desktop\OOJ Lab\book, bank, abstractArea\area>java Main
Enter 1 for Triangle
Enter 2 for Rectangle
Enter 3 for Circle
Enter your choice : 1
Enter height of triangle : 2
Enter base of triangle : 4
Area of triangle is 4.0

C:\Users\Vallisha\Desktop\OOJ Lab\book, bank, abstractArea\area>java Main
Enter 1 for Triangle
Enter 2 for Rectangle
Enter 3 for Circle
Enter your choice : 2
Enter width of Rectangle : 3
Enter length of Rectangle : 2
Area of rectangle is 6.0

C:\Users\Vallisha\Desktop\OOJ Lab\book, bank, abstractArea\area>java Main
Enter 1 for Triangle
Enter 2 for Rectangle
Enter 3 for Circle
Enter your choice : 3
Enter radius of circle : 10
Area of circle is 314.2
```

- 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.

Writeup:

Vallitha M
1BM19CS177

classmate

Date _____
Page _____

Bank Program

public class Bank {

private String customerName;
private String accountType;
private int accountNumber;

Bank () {

customerName = null;
accountType = null;
accountNumber = 0;

}

Bank (String customerName, int
accountNumber, String accountType) {

this.customerName = customerName;
this.accountNumber = accountNumber;
this.accountType = accountType;

@Override

public String toString () {

return String.format ("Customer Name :
" + customerName + " Account number :
" + accountNumber + " Account Type : "
+ accountType);

}

public class CurrentAccount extends Bank

private double balance;

CurrentAccount()

balance = 0.0D;

}

public void credit(double amount)

balance = balance + amount;

this.minimumBalance();

}

public int debit(double amount)

if (balance > 100 && balance - amount >= 0)

~~this~~.balance = balance - amount;

this.minimumBalance();

return 0;

}

return -1;

public void minimumBalance()

int fine = 0;

if (balance < 1000)

fine = 100;

balance = balance - fine;

public double getBalance()

return ~~this~~.balance;

{ } { }

public class SavingsAccount extends
Bank {

private double balance, rate;
Savings Account () {
rate = 4.567;
balance = 0.0D;
}

public double getBalance () {
return balance;
}

public void credit (double amount) {
balance = balance + amount;
~~This minimum Balance~~;

public int debit (double amount) {
~~If balance > 1000~~

If (balance - amount >= 0) {
balance = balance - amount;
return 0;
}
return -1;

public void calculateInterest (double time) {

rate = rate / 100;
if (time > 5) rate = 4.9 / 100;
if (time > 8) rate = 5.00 / 100;

$\text{balancee} = \text{balance} * \text{Math.pow}(1 + \text{rate}, \text{time});$

}

import java.util.Scanner;
public class Main

public static void main(String args[]);

Scanner in = new Scanner(System.in);

Bank bank;

String n, ch;

int amo, choice;

double amt;

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

cN = in.next();

System.out.print("Enter account
number : ");

amo = in.nextInt();

CurrentAccount account = new
CurrentAccount();

SavingsAccount account = new
SavingsAccount();

System.out.print("Enter 0 for
current account or ~~zero~~ 1 for
savings account : ");

boolean flag = (in.nextInt() == 0) true/false

```
if (flag)
```

```
    at = "current";
```

```
else
```

```
    at = "Savings"
```

```
bank = new Bank (n, amo, at);
```

```
while (true) {
```

```
    System.out.println ("Enter 1 to deposit");
```

```
    System.out.println ("Enter 2 to withdraw",  
    );
```

```
    System.out.println ("Enter 3 to check  
    balance");
```

```
    if (at
```

```
    if (!flag)
```

```
        System.out.println ("Enter 4 to  
        calculate interest");
```

```
    System.out.println ("Enter -1 to quit");
```

```
    System.out.print ("Enter your choice : ");
```

```
    choice = In.nextInt();
```

```
    if (choice == -1)
```

```
        break;
```

```
    if (choice == 1) {
```

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

```
        amt = Math.abs (In.nextDouble());
```

```
    if (flag) account.credit (amt);
```

```
    else account1.credit (amt);
```

```
}
```

```
If (choice == 2){
```

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

```
    amt = Math.abs(in.nextDouble());  
    int status = 0;
```

```
    if (flag) account.setStatus(account.debit(amt));  
    else status = account1.debit(amt);
```

```
    if (status == -1)
```

```
        System.out.println("Could not  
debit");
```

```
}  
else if (choice == 3){
```

```
    System.out.println(bank.toString());
```

```
    if (flag)
```

```
        System.out.println("Balance : "  
+ account.getBalance());
```

```
    else
```

```
        System.out.println("Balance : "  
+ account1.getBalance());
```

```
}  
else if (choice == 4 & & !flag){
```

```
    System.out.println("Enter number of  
years : ");
```

```
    amt = Math.abs(in.nextDouble());
```

```
    System.out.println("Balance before  
interest : " + account.getBalance());
```

```
    account1.calculateInterest(amt);
```

```
    System.out.println("Balance after  
interest : " + account1.getBalance());
```

```
}
```

```
else if (choice > 5 || choice == 0) choice = -1  
    || (flag && choice == 4)  
        System.out.println("Invalid input");  
    }  
    System.out.println("----- DONE -----");  
    in.close();  
}
```

Output:

```
C:\WINDOWS\system32\cmd.exe
C:\Users\Vallisha\Desktop\OOJ Lab\book, bank, abstractArea\bankProgram>java Main
Enter customer name : Vallisha
Enter account number : 1234
Enter 0 for current or 1 for savings : 0
Enter 1 to deposit
Enter 2 to withdraw
Enter 3 to check balance
Enter -1 to quit
Enter your choice : 1
Enter amount to be deposited :
200

Fine : 100

Enter 1 to deposit
Enter 2 to withdraw
Enter 3 to check balance
Enter -1 to quit
Enter your choice : 3
Customer Name : Vallisha
Account Number : 1234
Account Type : current
Balance : 100.0
Enter 1 to deposit
Enter 2 to withdraw
Enter 3 to check balance
Enter -1 to quit
Enter your choice : 2
Enter amount to be debited :
300
Could not debit
Enter 1 to deposit
Enter 2 to withdraw
Enter 3 to check balance
Enter -1 to quit
Enter your choice : 3
Customer Name : Vallisha
Account Number : 1234
Account Type : current
Balance : 100.0
Enter 1 to deposit
Enter 2 to withdraw
Enter 3 to check balance
Enter -1 to quit
Enter your choice : -1
-----DONE-----

C:\Users\Vallisha\Desktop\OOJ Lab\book, bank, abstractArea\bankProgram>
```

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.

Vallisha.M, 1BM19CS177
Program - 6
CLASSMATE
Date _____
Page _____

package CIE;

```

public class Student {
    private String name, usn;
    private int sem;
    public Student (String name, String usn, int sem) {
        this.name = name;
        this.usn = usn;
        this.sem = sem;
    }
    public Student () {
        name = usn = null;
        sem = 0;
    }
    @Override
    public String toString () {
        return String.format ("Name : " + name
            + "\nUSN : " + usn + " in Semester : "
            + sem + "\n");
    }
}

```

package CIE;

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

```
}  
public int getMark(int i) {  
    return marks[i];  
}
```

package SEE;

```
import CIE.Student; import CIE.Student;  
public class External extends Student {  
    private int marks[];  
    public External (int arr[]) {  
        marks = new int[5];  
        for (int i = 0; i < 5; i++)  
            marks[i] = arr[i];  
    }  
    public int getMark (int i) {  
        return marks[i];  
    }
```

```
import java.util.Scanner;  
import CIE.*;  
import SEE.*;  
  
public class Main {  
    public static void main (String args[]) {
```

```
        Scanner in = new Scanner (System.in);  
        int n, i;  
        String name, usn;  
        System.out.print ("Enter number of  
        Students : Students : ");  
        n = in.nextInt();  
        int marks[][] = new int [n][5];  
        int arr[] = new int[5];  
        Student stu[] = new Student[n];  
        Internal internal[] = new Internal[n];  
        External ext[] = new External[n];  
        for (i = 0; i < n; i++) {
```

```
            System.out.println ("---- Student " + (i+1) +  
                " ---- \n");  
            System.out.print ("Enter name : ");  
            name = in.nextLine();  
            name = in.next();  
            System.out.print ("Enter USN : ");  
            usn = in.next();
```

```

System.out.print("Enter semester : ");
Sem = in.nextInt();
stu[i] = new Student(name, user, Sem);
for (j = 0; j < 5; j++) {
    System.out.print("Enter CIE marks  
in course " + (j+1) + " : ");
    arr[j] = in.nextInt();
}
System.out.println("\n\n");
internal[i] = new Internal(arr);
for (j = 0; j < 5; j++) {
    System.out.print("Enter SEE marks  
in course " + (j+1) + " : ");
    arr[j] = in.nextInt();
}
ext[i] = new External(arr);
in.close();
for (i = 0; i < n; i++) {
    for (j = 0; j < 5; j++) {
        marks[i][j] = internal.getMark(i)
        + ((ext[i].getMark(j)) / 2);
    }
}
System.out.println("----- STUDENT  
DETAILS ----- \n\n");
for (i = 0; i < n; i++) {
    System.out.println("----- Student  
" + (i+1) + " ----- \n\n");
    System.out.println(stu[i].toString());
}

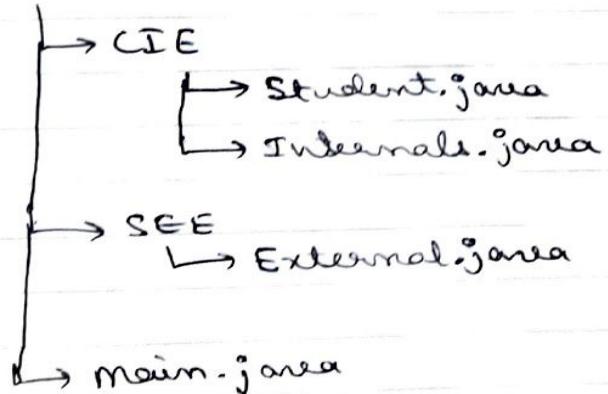
```

Intermediate
Java

```
for (j=0; j<5; j++) {  
    System.out.println("Marks scored  
in course " + (j+1) + " : " + marks[i][j]);  
}  
System.out.println("-----\n");  
}
```

FOLDER STRUCTURE:

prog 6



Output:

```
C:\WINDOWS\system32\cmd.exe
C:\Users\Vallisha\Desktop\00J Lab\prog6>java Main
Enter number of students : 2
----Student 1----

Enter Name : Vallisha
Enter USN : 1BM19CS177
Enter Semester : 3
Enter CIE marks in course 1 : 50
Enter CIE marks in course 2 : 50
Enter CIE marks in course 3 : 50
Enter CIE marks in course 4 : 50
Enter CIE marks in course 5 : 50

Enter SEE marks in course 1 : 75
Enter SEE marks in course 2 : 100
Enter SEE marks in course 3 : 100
Enter SEE marks in course 4 : 100
Enter SEE marks in course 5 : 100
----Student 2----

Enter Name : Vineet
Enter USN : 1BM19CS990
Enter Semester : 5
Enter CIE marks in course 1 : 50
Enter CIE marks in course 2 : 50
Enter CIE marks in course 3 : 45
Enter CIE marks in course 4 : 50
Enter CIE marks in course 5 : 50

Enter SEE marks in course 1 : 50
Enter SEE marks in course 2 : 100
Enter SEE marks in course 3 : 100
Enter SEE marks in course 4 : 100
Enter SEE marks in course 5 : 100
----STUDENT DETAILS----

----Student 1----

Name : Vallisha
USN : 1BM19CS177
Semester : 3

Marks scored in course 1 : 87
Marks scored in course 2 : 100
Marks scored in course 3 : 100
Marks scored in course 4 : 100
Marks scored in course 5 : 100
```

```
C:\WINDOWS\system32\cmd.exe
Enter CIE marks in course 2 : 50
Enter CIE marks in course 3 : 45
Enter CIE marks in course 4 : 50
Enter CIE marks in course 5 : 50

Enter SEE marks in course 1 : 50
Enter SEE marks in course 2 : 100
Enter SEE marks in course 3 : 100
Enter SEE marks in course 4 : 100
Enter SEE marks in course 5 : 100
-----STUDENT DETAILS-----

-----Student 1-----

Name : Vallisha
USN : 1BM19CS177
Semester : 3

Marks scored in course 1 : 87
Marks scored in course 2 : 100
Marks scored in course 3 : 100
Marks scored in course 4 : 100
Marks scored in course 5 : 100
-----Student 2-----

Name : Vineet
USN : 1BM19CS990
Semester : 5

Marks scored in course 1 : 75
Marks scored in course 2 : 100
Marks scored in course 3 : 95
Marks scored in course 4 : 100
Marks scored in course 5 : 100
C:\Users\Vallisha\Desktop\OOJ Lab\prog6>
```

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

Writeup:

Vallisha.m ; IBM19CS177 CLASSMATE Date _____
Page _____

Program - 7
Generics

```
class TwoGen<T, V>
{
    T obj1;
    V obj2;
    TwoGen(T obj1, V obj2)
    {
        obj1 = obj1;
        obj2 = obj2;
    }
    void showType()
    {
        System.out.println("Type of T is "
            + obj1.getClass().getName());
        System.out.println("Type of V is "
            + obj2.getClass().getName());
    }
    T getObj1()
    {
        return obj1;
    }
    V getObj2()
    {
        return obj2;
    }
}
```

```
class Main{
```

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

```
        TwoGen<Integer, String> obj = new
```

```
        TwoGen<Integer, String> (99, "Java");
```

```
        obj.showTypes();
```

```
        int v = obj.getObj1();
```

```
        System.out.println("Value of integer = "  
                           + v);
```

```
        String str = obj.getObj2();
```

```
        System.out.println("Value of String : "  
                           + str);
```

```
}
```

```
}
```

Output:

```
C:\Users\Vallisha\Desktop\OOJ Lab\program_7>java Main
Type of T is java.lang.Integer
Type of V is java.lang.String
Integer value : 99
String value : Java
```

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.

Writeup:

Vallisha.M . IBM19CS177

Program -8 -Exception

```
import java.util.Scanner;
```

```
class Main {
```

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

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

```
System.out.print ("Enter father's age : ");
```

```
int fatherAge = in.nextInt();
```

```
System.out.print ("Enter son's age : ");
```

```
int sonAge = in.nextInt();
```

```
in.close();
```

Son son;

try {

son = new Son (sonAge, fatherAge);

System.out.println("No errors");

}

catch (AgelessThanZeroException exception) {

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

}

catch (SonOlderThanFatherException exception) {

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

}

~~class~~

public class AgelessThanZeroException extends

Exception {

AgelessThanZeroException(^{String} message) {

super(message);

}

public class SonOlderThanFatherException extends

Exception {

SonOlderThanFatherException(^{String} message) {

super(message);

}

public class Father {

int age = 0;

message = "Father's Age cannot be less than zero";

Father() {}

Father (int age) throws AgeLessThanZeroException

{ if (age < 0)

 throws new AgeLessThanZeroException(

 message);

 this.age = age;

}

public class Son extends Father {

String message1 = "Son's age cannot be greater than father's age";

String message2 = "Son's age cannot be less than zero";

int age;

Son (int sonAge, int fatherAge) throws

AgeLessThanZeroException, SonOlderThanFatherException

{

 super(fatherAge);

 if (sonAge < 0)

 throws new AgeLessThanZeroException(message1);

 if (sonAge >= fatherAge)

 throws new SonOlderThanFatherException(message2);

 age = sonAge;

}

Output:

```
C:\Users\Vallisha\Desktop\Labs\00J Lab\program_8>java Main
Enter father's age : 25
Enter son's age : 2
No Errors

C:\Users\Vallisha\Desktop\Labs\00J Lab\program_8>java Main
Enter father's age : -5
Enter son's age : 6
Father's age cannot be less than zero

C:\Users\Vallisha\Desktop\Labs\00J Lab\program_8>java Main
Enter father's age : -5
Enter son's age : -3
Father's age cannot be less than zero

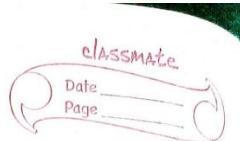
C:\Users\Vallisha\Desktop\Labs\00J Lab\program_8>java Main
Enter father's age : 25
Enter son's age : -55
Son's age cannot be less than zero

C:\Users\Vallisha\Desktop\Labs\00J Lab\program_8>
```

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.

Writeup:

Vallisha M
1BM19CS177



- Thread program

```
class thread1 extends Thread {
```

```
    thread1() {
```

```
        : new Thread ("thread1");  
        start();
```

```
}
```

```
public void run() {
```

```
    while (true) {
```

```
        try {
```

```
            System.out.println ("BMS
```

```
College of Engineering");
```

```
Thread.sleep(1000);
```

```
}
```

```
        catch (InterruptedException ie) {
```

```
            System.out.println (
```

```
                "Interrupted");
```

```
}
```

```
}
```

```
class thread2 extends Thread {
```

```
    thread2() {
```

```
        : new Thread ("thread2");
```

```
        start();
```

```
}
```

```
public void run()
```

{

```
    while(true)
```

{

```
        try{
```

```
            System.out.println("ESE");
```

```
            Thread.sleep(2000);
```

{

```
        catch (InterruptedException e){
```

```
            System.out.println("Interrupted");
```

{

{

{

```
class Main{
```

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

```
        System.out.println("Enter Ctrl+C to stop");
```

~~██████████~~

```
        Thread t1 = new Thread();
```

```
        Thread t2 = new Thread();
```

{

{

Output:

```
BMS College Of Engineering
CSE
CSE
CSE
CSE
BMS College Of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College Of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College Of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College Of Engineering
```