

00J LAB TEST 2
RECORD EVALUATION

Program 1

Develop a Java program that prints all real solutions to the quadratic equation $ax^2 + bx + c = 0$. Read in a , b , c and use the quadratic formula. If the discriminant $b^2 - 4ac$ is negative, display a message stating that there are no real solutions.

```
Lab 1
import java.util.*;
class quadratic
{
    public static void main (String[] args)
    {
        Scanner sc = new Scanner (System.in);
        System.out.println ("Enter the values");
        double a = sc.nextInt();
        double b = sc.nextInt();
        double c = sc.nextInt();
        while (a != 0)
        {
            double d = (b*b) - (4*a*c);
            if (d == 0)
            {
                double x = (-1*b)/(2*a);
                System.out.println ("The roots are : " + x);
                break;
            }
            else if (d > 0)
            {
                double xa = (-1*b + Math.sqrt(d))/(2*a);
                double xb = (-1*b - Math.sqrt(d))/(2*a);
                System.out.println ("The roots are " + xa + "&" + xb);
                break;
            }
            else if (d < 0)
            {
                System.out.println ("The roots are imaginary");
                break;
            }
        }
    }
}
```

```
quadratic.java
1  import java.util.*;
2  class quadratic
3  {
4      public static void main(String[] args)
5      {
6          Scanner sc=new Scanner (System.in);
7          System.out.println("enter the values");
8          double a=sc.nextInt();
9          double b=sc.nextInt();
10         double c=sc.nextInt();
11         while(a!=0)
12         {
13             double d=(b*b)-(4*a*c);
14             if(d==0)
15             {
16                 double x=(-1*b)/(2*a);
17                 System.out.println("the roots are: "+x);
18                 break;
19             }
20             else if(d>0)
21             {
22                 double xa=(-1*b+Math.sqrt(d))/(2*a);
23                 double xb=(-1*b-Math.sqrt(d))/(2*a);
24                 System.out.println("the roots are "+xa +"&"+xb);
25                 break;
26             }
27             else if(d<0)
28             {
29                 System.out.println("the roots are imaginary");
30                 break;
31             }
32         }
33     }
34 }
```

```
Command Prompt
Microsoft Windows [Version 10.0.18363.1082]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\91992>cd desktop
C:\Users\91992\Desktop>javac quadratic.java
C:\Users\91992\Desktop>java quadratic
enter the values
1
-2
1
the roots are: 1.0

C:\Users\91992\Desktop>java quadratic
enter the values
4
4
1
the roots are: -0.5

C:\Users\91992\Desktop>
```

Program 2

Develop a Java program to create a class Student with members usn, name, an array credits and an array mark. Include methods to accept and display details and a method to calculate SGPA of a student.

Lab 2

```
import java.util.*;
```

```
class student
```

```
{ private String usn;
```

```
private String Name;
```

```
private int n;
```

```
private double sg SGPA = 0;
```

```
private int totalcredits = 0;
```

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

```
void Details ()
```

```
{
```

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

```
usn = ss.nextLine();
```

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

```
Name = ss.nextLine();
```

```
System.out.println("Enter no. of subjects");
```

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

```
int credits[] = new int[n];
```

```
double marks[] = new double[n];
```

```
System.out.println("Enter details of the subject:");
```

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

```
{
```

```
System.out.println("Enter credits alloted  
to the subject " + (i + 1));
```

```
credits[i] = ss.nextInt();
```

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

```
marks[i] = ss.nextInt();
```

```
calculate(credits[i], marks[i], i);
```

```
}
```

```
}
```

```

void calculate (int credit, double mark, int j)
{
    totalcredits = totalcredits + credit;
    if (mark >= 90 && mark <= 100)
    if (mark >= 90 && mark <= 100)
        SGPA = SGPA + (10 * credit);
    else if (mark >= 80 && mark <= 89)
    else if (mark >= 80 && mark <= 89)
        SGPA = SGPA + (9 * credit);
    else if (mark >= 70 && mark <= 79)
        SGPA = SGPA + (8 * credit);
    else if (mark >= 60 && mark <= 69)
        SGPA = SGPA + (7 * credit);
    else if (mark >= 50 && mark <= 59)
        SGPA = SGPA + (6 * credit);
    else if (mark >= 40 && mark <= 49)
        SGPA = SGPA + (5 * credit);
    else
        System.out.println("Failed in subjects" + (j+1));
}

```

```

}
void Display()

```

```

{
    System.out.println("Details of the student");
    System.out.println("Name: " + name);
    System.out.println("USN: " + USN);
    System.out.println("SGPA of student " + (SGPA / totalcredits));
}

```

```

}
}
class main

```

```

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

```

```

        Student s1 = new Student();

```

```

        s1.Details();

```

```

        s1.Display();
    }
}

```



```

1  import java.util.*;
2  class Student
3  {
4      private String USN;
5      private String name;
6      private int n;
7      private double SGPA = 0;
8      private int totalCredits = 0;
9      Scanner ss = new Scanner(System.in);
10
11     void Details()
12     {
13         System.out.println("Enter USN of the student");
14         USN = ss.nextLine();
15         System.out.println("Enter Name of the student");
16         name = ss.nextLine();
17         System.out.println("Enter no of subjects");
18         n = ss.nextInt();
19         int credits[] = new int[n];
20         double marks[] = new double[n];
21         System.out.println("Enter details of the subjects:");
22         for(int i=0;i<n;i++)
23         {
24             System.out.println("Enter credits allotted to the subject "+(i+1));
25             credits[i] = ss.nextInt();
26             System.out.println("Enter marks in the subject "+(i+1));
27             marks[i] = ss.nextInt();
28             Calculate(credits[i],marks[i],i);
29         }
30     }
31     void Calculate(int credit,double mark,int j)
32     {
33         totalCredits = totalCredits + credit;
34         if(mark>=90&&mark<=100)
35             SGPA = SGPA + (10*credit);
36         else if(mark>=80 && mark<=89)
37             SGPA = SGPA + (9*credit);
38         else if(mark>=70&&mark<=79)
39             SGPA = SGPA + (8*credit);
40         else if(mark>=60&&mark<=69)
41             SGPA = SGPA + (7*credit);
42         else if(mark>=50 && mark<=59)
43             SGPA = SGPA + (6*credit);
44         else if(mark>=40&&mark<=49)
45             SGPA = SGPA + (5*credit);
46         else
47             System.out.println("Failed in ubject "+(j+1));
48     }
49     void Display()
50     {
51         System.out.println("Details of the Student");
52         System.out.println("Name : "+name);
53         System.out.println("USN: "+USN);
54         System.out.println("SGPA of student "+(SGPA/totalCredits));
55     }
56 }
57 class Main
58 {
59     public static void main(String args[])
60     {
61         Student s1 = new Student();
62         s1.Details();
63         s1.Display();
64     }
65 }

```

```
C:\Users\91992\Desktop>javac Student.java
```

```
C:\Users\91992\Desktop>java Main
```

```
Enter USN of the student
```

```
1bm19cs023
```

```
Enter Name of the student
```

```
arjun balu
```

```
Enter no of subjects
```

```
4
```

```
Enter details of the subjects:
```

```
Enter credits allotted to the subject 1
```

```
4
```

```
Enter marks in the subject 1
```

```
85
```

```
Enter credits allotted to the subject 2
```

```
3
```

```
Enter marks in the subject 2
```

```
75
```

```
Enter credits allotted to the subject 3
```

```
3
```

```
Enter marks in the subject 3
```

```
50
```

```
Enter credits allotted to the subject 4
```

```
2
```

```
Enter marks in the subject 4
```

```
35
```

```
Failed in ubject 4
```

```
Details of the Student
```

```
Name :arjun balu
```

```
USN: 1bm19cs023
```

```
SGPA of student 6.5
```

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.

```
Lab 3
import java.util.*;

class Book {
    String name;
    String Author;
    int price;
    int num_pages;
    Book()
    {
    }
    Book (String name, String author, int price, int num_pages)
    {
        this.name = name;
        this.author = author;
        this.price = price;
        this.num_pages = num_pages;
    }
    void accept()
    {
        Scanner s = new Scanner (System.in);
        System.out.println ("Enter the name of the book");
        name = s.next();
        System.out.println ("Enter the author of the book");
        Author = s.next();
        System.out.println ("Enter the price of the book");
        price = s.nextInt();
        System.out.println ("Enter the number of pages of the book");
        num_pages = s.nextInt();
    }
    public String toString()
    {
        return ("Name: " + name + "\n" + "Author: " + author +
            "\n" + "price: " + price + "\n" + "number of pages: " + num_pages);
    }
}
```

```

class BMain {
    public static void main(String ss [])
    {
        Scanner a = new Scanner(System.in);
        Book b1 = new Book("Heights", "Anne", 299, 345);
        System.out.println("Sample input: \n" + b1);
        System.out.println("Enter the number of books");
        int n = a.nextInt();
        Book b[] = new Book[n];
        for(int i=0; i<n; i++)
        {
            b[i] = new Book();
            System.out.println("Enter the details of " + (i+1) + " book");
            b[i].accept();
        }
        for(int i=0; i<n; i++)
        {
            System.out.println("Details of book " + (i+1));
            System.out.println(b[i]);
        }
    }
}

```

```

import java.util.*;
class Book {
    String name;
    String author;
    int price;
    int num_pages;
    Book()
    {}
    Book(String name, String author, int price, int num_pages)
    {
        this.name=name;
        this.author=author;
        this.price=price;
        this.num_pages=num_pages;
    }
    void accept()
    {
        Scanner s=new Scanner(System.in);
        System.out.println("Enter the name of the book");
        name=s.next();
        System.out.println("Enter the author of the book");
        author=s.next();
        System.out.println("Enter the price of the book");
        price=s.nextInt();
        System.out.println("Enter the number of pages of the book");
        num_pages=s.nextInt();
    }
    public String toString()
    {
        return ("Name: " + name + "\n" + "Author: " + author + "\n" + "Price: " + price + "\n" + "Number of pages: " + num_pages);
    }
}
class BMain {
    public static void main(String ss[])
    {
        Scanner a=new Scanner(System.in);
        Book b1=new Book("Heights", "Anne", 299, 345);
        System.out.println("Sample input: \n" + b1);
        System.out.println("Enter the number of books");
        int n=a.nextInt();
        Book b[]=new Book[n];
        for(int i=0; i<n; i++)
        {
            b[i]=new Book();
            System.out.println("Enter the details of " + (i+1) + " book");
            b[i].accept();
        }
        for(int i=0; i<n; i++)
        {
            System.out.println("Details of book " + (i+1));
            System.out.println(b[i]);
        }
    }
}

```



```
C:\Users\91992\Desktop>javac Book.java
```

```
C:\Users\91992\Desktop>java BMain
```

```
Sample input:
```

```
Name: Heights
```

```
Author: Anne
```

```
Price: 299
```

```
Number of pages: 345
```

```
Enter the number of books
```

```
3
```

```
Enter the details of 1 book
```

```
Enter the name of the book
```

```
book1
```

```
Enter the author of the book
```

```
auth1
```

```
Enter the price of the book
```

```
499
```

```
Enter the number of pages of the book
```

```
350
```

```
Enter the details of 2 book
```

```
Enter the name of the book
```

```
book2
```

```
Enter the author of the book
```

```
auth2
```

```
Enter the price of the book
```

```
420
```

```
Enter the number of pages of the book
```

```
120
```

```
Enter the details of 3 book
```

```
Enter the name of the book
```

```
book3
```

```
Enter the author of the book
```

```
auth3
```

```
Enter the price of the book
```

```
499
```

```
Enter the number of pages of the book
```

```
299
```

```
Details of book 1
```

```
Name: book1
```

```
Author: auth1
```

```
Price: 499
```

```
Number of pages: 350
```

```
Details of book 2
```

```
Name: book2
```

```
Author: auth2
```

```
Price: 420
```

```
Number of pages: 120
```

```
Details of book 3
```

```
Name: book3
```

```
Author: auth3
```

```
Price: 499
```

```
Number of pages: 299
```

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.

Lab 4

```
import java.util. * ;
abstract class Shape {
    int length, breadth, radius;
    Scanner input = new Scanner (System.in);
    abstract void printArea();
}

class Rectangle extends Shape {
    void printArea() {
        System.out.println(" *** Finding Area of Rectangle ***");
        System.out.println("Enter length and breadth: ");
        length = input.nextInt();
        breadth = input.nextInt();
        System.out.println("Area of rectangle is: " + length * breadth);
    }
}

class Triangle extends Shape {
    void printArea() {
        System.out.println("\n *** finding Area of Triangle ***");
        System.out.println("Enter Base and Height: ");
        length = input.nextInt();
        breadth = input.nextInt();
        System.out.println("Area of Triangle is: " + (length * breadth) / 2);
    }
}
```

```

class Circle extends Shape {
    void printArea() {
        System.out.println("\n *** Finding Area of circle ***");
        System.out.println("Enter Radius:");
        radius = input.nextInt();
        System.out.println("The area of the circle is: "
            + 3.14f * radius * radius);
    }
}

```

```

public class AbstractClassMain {
    public static void main (String[] args) {
        Rectangle rec = new Rectangle();
        rec.printArea();
        Circle new cir = new Circle();
        cir.printArea();
        Triangle tri = new Triangle();
        tri.printArea();
    }
}

```

```

1  import java.util.*;
2
3  abstract class Shape {
4      int length, breadth, radius;
5
6      Scanner input = new Scanner(System.in);
7
8      abstract void printArea();
9  }
10
11
12  class Rectangle extends Shape {
13      void printArea() {
14          System.out.println("*** Finding the Area of Rectangle ***");
15          System.out.print("Enter length and breadth: ");
16          length = input.nextInt();
17          breadth = input.nextInt();
18          System.out.println("The area of Rectangle is: " + length * breadth);
19      }
20  }
21
22  class Triangle extends Shape {
23      void printArea() {
24          System.out.println("\n*** Finding the Area of Triangle ***");
25          System.out.print("Enter Base And Height: ");
26          length = input.nextInt();
27          breadth = input.nextInt();
28          System.out.println("The area of Triangle is: " + (length * breadth) / 2);
29      }
30  }
31
32  class Cricle extends Shape {
33      void printArea() {
34          System.out.println("\n*** Finding the Area of Cricle ***");
35          System.out.print("Enter Radius: ");
36          radius = input.nextInt();
37          System.out.println("The area of Cricle is: " + 3.14f * radius * radius);
38      }
39  }
40
41  public class AbstractClassMain {
42      public static void main(String[] args) {
43          Rectangle rec = new Rectangle();
44          rec.printArea();
45
46          Triangle tri = new Triangle();
47          tri.printArea();
48
49          Cricle cri = new Cricle();
50          cri.printArea();
51      }
52  }
53

```

```

C:\Users\91992\Desktop>java AbstractClassMain
*** Finding the Area of Rectangle ***
Enter length and breadth: 10
20
The area of Rectangle is: 200

*** Finding the Area of Triangle ***
Enter Base And Height: 34
23
The area of Triangle is: 391

*** Finding the Area of Cricle ***
Enter Radius: 12
The area of Cricle is: 452.16

```


Program 5

Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, account number and type of account. From this derive the classes Curr-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks:

- Accept deposit from customer and update the balance.
- Display the balance.
- Compute and deposit interest
- Permit withdrawal and update the balance
- Check for the minimum balance, impose penalty if necessary and update the balance

```
1  import java.util.Scanner;
2
3  abstract class Account {
4      String cName, accType;
5      long accNo;
6      double bal;
7      final double minBal = 1000.0;
8
9      Account(String cName, long accNo, double bal, String accType) {
10         this.accNo = accNo;
11         this.cName = cName;
12         this.bal = bal;
13         this.accType = accType;
14     }
15
16     abstract void addBal(double amt);
17
18     abstract void dispBal();
19
20     abstract void withBal(double amt);
21 }
22
23 class Curr_acct extends Account {
24     Curr_acct(String cName, long accNo, double bal) {
25         super(cName, accNo, bal, "Current");
26         System.out.println("Name: " + cName + "\taccno: " + accNo + "\tbal: " + bal + "\ttype: " + accType);
27     }
28
29     void addBal(double amt) {
30         this.bal += amt;
31     }
32
33
34     void dispBal() {
35         System.out.println("Your balance is: " + this.bal);
36     }
37
38     void checkBal() {
39         if (this.bal < minBal) {
40             System.out.println("Insufficient balance, penalty imposed");
41             this.bal -= this.bal * 0.02;
42         }
43     }
44
45     void withBal(double amt) {
46         this.bal -= amt;
47         checkBal();
48     }
49 }
50
51 class Sav_acct extends Account {
52     Sav_acct(String cName, long accNo, double bal) {
53         super(cName, accNo, bal, "Savings");
54         System.out.println("Name: " + cName + "\taccno: " + accNo + "\tbal: " + bal + "\ttype: " + accType);
55     }
56
57     void addBal(double amt) {
```

```

58         this.bal += amt;
59         addIntr();
60     }
61
62     void addIntr() {
63         this.bal += this.bal * 0.07;
64     }
65
66     void dispBal(){
67         System.out.println("Your balance is (Including Interest): " + this.bal );
68     }
69
70
71     void withBal(double amt){
72         this.bal -= amt;
73     }
74
75
76
77
78 }
79
80
81 class Bank {
82     public static void main(String[] args) {
83         Scanner sc = new Scanner(System.in);
84         Double amt;
85         System.out.println("Enter your details:");
86         System.out.println("Name:");
87         String x=sc.next();
88         System.out.println("Account Number:");
89         long y=sc.nextLong();
90         for(;;)
91         {
92             System.out.println("Type of account:\n1.Current account\n2.Savings account\n3.Exit");
93             int t=sc.nextInt();
94
95
96             if(t==1){
97                 System.out.println("The current account provides cheque book facility but no interest.");
98                 Curr_acct c = new Curr_acct(x, y, 50000);
99                 for(;;)
100                 {
101                     System.out.println("1:Deposit\n2:Display Balance\n3:Withdraw\n4:Exit");
102                     int ch = sc.nextInt();
103
104                     switch (ch) {
105                         case 1:
106                             System.out.println("Enter the amount to be added:");
107                             amt = sc.nextDouble();
108                             c.addBal(amt);
109                             break;
110
111                         case 2:
112                             c.dispBal();
113                             break;
114
115                         case 3:
116                             System.out.println("Enter the amount to be withdrawn:");
117                             amt = sc.nextDouble();
118                             c.withBal(amt);
119                             break;
120                         case 4: System.exit(0);
121                         default: System.out.println("Invalid choice! Try again");
122                     }
123                 }
124             }
125
126
127             else if(t==2){
128                 System.out.println("The savings account provides compound interest and withdrawal facilities but no cheque book facility.");
129                 Sav_acct s = new Sav_acct(x, y, 5000);
130                 for(;;) {
131                     System.out.println("1:Deposit\n2:Display Balance\n3:Withdraw\n4:Exit");
132                     int ch = sc.nextInt();
133                     switch (ch) {
134                         case 1:
135                             System.out.println("Enter the amount to be added:");
136                             amt = sc.nextDouble();
137                             s.addBal(amt);
138                             break;
139
140                         case 2:
141                             s.dispBal();
142                             break;
143
144                         case 3:
145                             System.out.println("Enter the amount to be withdrawn:");
146                             amt = sc.nextDouble();
147                             s.withBal(amt);
148                             break;
149                         case 4: System.exit(0);
150                         default: System.out.println("Invalid choice! Try again");
151                     }
152                 }
153             }
154             else if(t==3)
155                 System.exit(0);
156             else
157                 System.out.println("Invalid choice! Try again");
158
159         }
160     }
161 }
162
163

```

```
C:\Users\91992\Desktop>java Bank
```

```
Enter your details:
```

```
Name:
```

```
arjun
```

```
Account Number:
```

```
12
```

```
Type of account:
```

```
1.Current account
```

```
2.Savings account
```

```
3.Exit
```

```
1
```

```
The current account provides cheque book facility but no interest.
```

```
Name: arjun    accno: 12    bal: 50000.0    type: Current
```

```
1:Deposit
```

```
2:Display Balance
```

```
3:Withdraw
```

```
4:Exit
```

```
4
```

```
C:\Users\91992\Desktop>java Bank
```

```
Enter your details:
```

```
Name:
```

```
arjun
```

```
Account Number:
```

```
12
```

```
Type of account:
```

```
1.Current account
```

```
2.Savings account
```

```
3.Exit
```

```
2
```

```
The savings account provides compound interest and withdrawal facilities but no cheque book facility.
```

```
name: arjun    accno: 12    bal: 5000.0    type: Savings
```

```
1:Deposit
```

```
2:Display Balance
```

```
3:Withdraw
```

```
4:Exit
```

```
1
```

```
Enter the amount to be added:
```

```
5000
```

```
1:Deposit
```

```
2:Display Balance
```

```
3:Withdraw
```

```
4:Exit
```

```
2
```

```
Your balance is (Including Interest): 10700.0
```

```
1:Deposit
```

```
2:Display Balance
```

```
3:Withdraw
```

```
4:Exit
```

PROGRAM 6

Question-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.

```
1  import CIE.*;
2  import SEE.*;
3  import java.util.*;
4  public class pack
5  {
6      public static void main(String args[])
7      {
8          int n;
9          Scanner ss=new Scanner(System.in);
10         System.out.println("Enter the number of students:");
11         n=ss.nextInt();
12         CIE.Student[] st =new CIE.Student[n];
13         CIE.Internals[] in =new CIE.Internals[n];
14         SEE.Externals[] e =new SEE.Externals[n];
15         for(int i=0;i<n;i++)
16         {
17             st[i]=new CIE.Student();
18             st[i].display();
19             in[i]=new CIE.Internals();
20             in[i].display();
21             e[i]=new SEE.Externals();
22             e[i].display();
23             System.out.println("Total marks of student "+st[i].name+" in 5 subjects are:");
24             for(int j=0;j<5;j++)
25             {
26                 System.out.println(in[i].cie_m[j]+(e[i].see_m[j]/2));
27             }
28         }
29     }
30 }
31 }
```



```
Command Prompt - java pack
Enter the number of students:
3
Name:
AQW
USN:
12QM
Semester:
2
CIE Marks for 5 subjects(out of 50):
33
22
44
11
34
SEE Marks for 5 subjects(out of 100):
55
66
77
88
99
Total marks of student AQW in 5 subjects are:
60.5
55.0
82.5
55.0
83.5
Name:
ZXC
USN:
34ER
Semester:
4
CIE Marks for 5 subjects(out of 50):
12
32
43
23
44
SEE Marks for 5 subjects(out of 100):
67
78
87
99
78
Total marks of student ZXC in 5 subjects are:
45.5
71.0
86.5
72.5
83.0
```

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

```
1  class Generics<T, U,S,B,D>{
2      T o1;
3      U o2;
4      S o3;
5      B o4;
6      D o5;
7      Generics(T o1, U o2,S o3,B o4,D o5){
8          this.o1 = o1;
9          this.o2 = o2;
10         this.o3 = o3;
11         this.o4 = o4;
12         this.o5 = o5;
13     }
14     public void print(){
15         System.out.println(o1);
16         System.out.println(o2);
17         System.out.println(o3);
18         System.out.println(o4);
19         System.out.println(o5);
20     }
21 }
22
23 class Gmain{
24     public static void main (String[] args){
25         Generics <String,Integer, String, Boolean, Double> o =
26             new Generics<String, Integer, String, Boolean, Double>("Hello", 9,"World",true,6.45);
27
28         o.print();
29     }
30 }
```

```
Command Prompt
Microsoft Windows [Version 10.0.18363.1198]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\arkas>cd onedrive

C:\Users\arkas\OneDrive>cd desktop

C:\Users\arkas\OneDrive\Desktop>cd java

C:\Users\arkas\OneDrive\Desktop\java>javac Gmain.java

C:\Users\arkas\OneDrive\Desktop\java>javac Gmain.java

C:\Users\arkas\OneDrive\Desktop\java>java Gmain
Hello
9
World
true
6.45

C:\Users\arkas\OneDrive\Desktop\java>
```

LAB PROGRAM 8

Question-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=father’s age.

```

1  import java.util.*;
2
3  class WrongAge extends Exception
4  {
5      public String toString()
6      {
7          return "Please enter the correct age:"+ "Son's age cannot be greater than Father's age";
8      }
9  }
10 }
11 class Father
12 {
13     int age;
14     Father(int age1)
15     {
16         age=age1;
17         System.out.println("Father age:"+age);
18     }
19 }
20 class Son extends Father
21 {
22     Son(int age2)
23     {
24         System.out.println("Son age:"+age2);
25     }
26 }
27 class Agemain1
28 {
29     public static void main(String args[]) throws WrongAge
30     {
31         Scanner ss=new Scanner(System.in);
32         System.out.print("Enter the age of father: ");
33         int j=ss.nextInt();
34         System.out.print("Enter the age of son: ");
35         int k=ss.nextInt();
36
37         if(k>=j)
38         {
39             throw new WrongAge();
40         }
41         else
42         {
43             Father f=new Father(j);
44             Son s=new Son(k);
45         }
46     }
47 }

```

LAB PROGRAM 9

Question-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.

LAB PROGRAM 10 Question- Write a program that creates a user interface

```

1  class Threadt implements Runnable {
2      String name;
3      Thread t;
4      int time;
5      Threadt(String threadname,int time) {
6          name = threadname;
7          this.time=time;
8          t = new Thread(this, name);
9          System.out.println("thread:"+ t);
10         t.start();
11     }
12     public void run() {
13         try {
14             for(int i = 5; i > 0; i--) {
15                 System.out.println(name);
16                 Thread.sleep(time);
17             }
18         } catch (InterruptedException e) {
19             System.out.println(name + "Interrupted");
20         }
21         System.out.println(name + " exiting.");
22     }
23 }
24 class Week11 {
25     public static void main(String args[]) {
26         Threadt t1=new Threadt("BMS COLLEGE OF ENGINEERING",10000);
27         Threadt t2=new Threadt("COMPUTER SCIENCE OF ENGINEERING",2000);
28     }
29 }

```

```

C:\Users\arkas\OneDrive\Desktop\java>javac Week11.java
constructor Thread.Thread(Runnable,AccessControlContext) is not applicable
(argument mismatch; String cannot be converted to Runnable)
constructor Thread.Thread(ThreadGroup,Runnable) is not applicable
(argument mismatch; String cannot be converted to ThreadGroup)
constructor Thread.Thread(ThreadGroup,String) is not applicable
(argument mismatch; String cannot be converted to ThreadGroup)
constructor Thread.Thread(Runnable,String) is not applicable
(argument mismatch; String cannot be converted to Runnable)
Note: Some messages have been simplified; recompile with -Xdiags:verbose to get full output
2 errors

C:\Users\arkas\OneDrive\Desktop\java>java Week11
thread:Thread[BMS COLLEGE OF ENGINEERING,5,main]
thread:Thread[COMPUTER SCIENCE OF ENGINEERING,5,main]
BMS COLLEGE OF ENGINEERING
COMPUTER SCIENCE OF ENGINEERING
COMPUTER SCIENCE OF ENGINEERING
COMPUTER SCIENCE OF ENGINEERING
COMPUTER SCIENCE OF ENGINEERING
BMS COLLEGE OF ENGINEERING
COMPUTER SCIENCE OF ENGINEERING exiting.
BMS COLLEGE OF ENGINEERING
BMS COLLEGE OF ENGINEERING
BMS COLLEGE OF ENGINEERING
BMS COLLEGE OF ENGINEERING exiting.

C:\Users\arkas\OneDrive\Desktop\java>

```

e to perform integer divisions. The user enters two numbers in the textfields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were

Zero, the program would throw an Arithmetic Exception Display the exception in a message dialog box.

```
1  import java.awt.*;
2  import java.awt.event.*;
3  class DivisionInteger extends Frame implements ActionListener{
4      TextField num1TextField;
5      TextField num2TextField;
6      Button calculate;
7      int a,b;
8      float result;
9      String msg="Enter the numbers";
10     public DivisionInteger(){
11
12         setLayout(new FlowLayout());
13
14         calculate=new Button("Calculate");
15         num1TextField=new TextField(5);
16         Label num1Label=new Label("Number 1",Label.RIGHT);
17         num2TextField=new TextField(5);
18         Label num2Label=new Label("Number 2",Label.RIGHT);
19
20         add(num1Label);
21         add(num1TextField);
22         add(num2Label);
23         add(num2TextField);
24         add(calculate);
25         num1TextField.addActionListener(this);
26         num2TextField.addActionListener(this);
27         calculate.addActionListener(this);
28
29         addWindowListener(new MyWindowAdapter());
30     }
31     public void actionPerformed(ActionEvent ae){
32         try{
33             result=divideNumbers();
34             msg=("The result is "+result);
35             repaint();
36         }catch(NumberFormatException e){
37             mse="Number is not Integer."+e:
38             47             if(b==0){
39             48                 throw new ArithmeticException();
40             49             }
41             50             return (float)a/b;
42             51         }
43             52         public void paint(Graphics g){
44             53             g.drawString(msg,50,100);
45             54         }
46
47         public static void main(String args[]){
48             DivisionInteger div=new DivisionInteger();
49             div.setSize(new Dimension(500,500));
50             div.setTitle("Division Calculater");
51             div.setVisible(true);
52         }
53     }
54     class MyWindowAdapter extends WindowAdapter{
55         public void windowClosing(WindowEvent event){
56             System.exit(0);
57         }
58     }
59 }
```

