



GATE INSTITUTE OF TECHNOLOGY
& MANAGEMENT SCIENCES

(Approved by AICTE, Affiliated to Sri Venkateswara University, TIRUPATI)

ICET CODE : GTMT



Master of Computer Applications

SEMESTER - 2

Hall Ticket No: 3124167

Name of the Student: Kondapalli Saiteja

Subject: object oriented programming with java

Year: 1 Semester: 1st Batch 2024-2026



GATE EDUCATIONAL INSTITUTIONS

DEPARTMENT OF COMPUTERS

YEAR I SEMESTER 1st

Reg No. _____

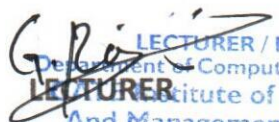
CERTIFICATE

Certified that this is the bonafide record of practical work done in the laboratory

by the candidate Kondapalli SaiTeja during the
academic year 2024-2026 subject object oriented programming with java

No. of Practical's Conducted 13

No. of Practical's Attended 13


LECTURER / HEAD
Department of Computer Applications
GATE Institute of Technology
And Management Sciences
TIRUPATI


LECTURER / HEAD
HEAD OF THE DEPARTMENT
GATE Institute of Technology
And Management Sciences
TIRUPATI

Submitted for the Practical Examination held on _____

Examiners

1.

2.

INDEX

S.No.	Date	Name of the Experiment	Page No.	Marks Awarded	Remarks
1		operators & conditional statements			
2		Arrays			
3		constructors			
4		Method overloading			
5		Packages			
6		exception handling			
7		Multi threading			
8		Vectors			
9		string operations			
10		calendar			
11		Applet			
12		Swings			

INDEX

S.No.	Date	Name of the Experiment	Page No.	Marks Awarded	Remarks
		1. To determine the focal length of a convex lens by the distant object method.	1	10	
		2. To determine the focal length of a convex lens by the u-v method.	2	10	
		3. To determine the focal length of a concave lens by the u-v method.	3	10	
		4. To determine the focal length of a concave lens by the combination method.	4	10	
		5. To determine the focal length of a concave lens by the combination method.	5	10	
		6. To determine the focal length of a concave lens by the combination method.	6	10	
		7. To determine the focal length of a concave lens by the combination method.	7	10	
		8. To determine the focal length of a concave lens by the combination method.	8	10	
		9. To determine the focal length of a concave lens by the combination method.	9	10	
		10. To determine the focal length of a concave lens by the combination method.	10	10	
		11. To determine the focal length of a concave lens by the combination method.	11	10	
		12. To determine the focal length of a concave lens by the combination method.	12	10	
		13. To determine the focal length of a concave lens by the combination method.	13	10	
		14. To determine the focal length of a concave lens by the combination method.	14	10	
		15. To determine the focal length of a concave lens by the combination method.	15	10	
		16. To determine the focal length of a concave lens by the combination method.	16	10	
		17. To determine the focal length of a concave lens by the combination method.	17	10	
		18. To determine the focal length of a concave lens by the combination method.	18	10	
		19. To determine the focal length of a concave lens by the combination method.	19	10	
		20. To determine the focal length of a concave lens by the combination method.	20	10	

INDEX

[illegible]

INDEX

[illegible]

Date :	GATE Educational Institutions	Page No.:
	operators & conditional statements	Practical No. :

Problem Definition:-

Design and develop a java program to implement the operators and conditional statements

Description:-

- * operator in java is a symbol that is used to perform operators like +, -, *, / etc.
- * conditional statements are mostly used in decision making scenarios. The conditional statements are also referred as branching statements. There are different types of Conditional statements

They are :-

1. if statement
2. nested if statement
3. if-else statement
4. if-else-if statement
5. switch case statement

Date :

GATE Educational Institutions

Page No.:

Practical No. :

Algorithm :-

step 1 : start

step 2 : import package

step 3 : Declare variables a, b, result = 0

step 4 : -Accept two values a & b

step 5 : if $a=0$ & $b=0$ then print result

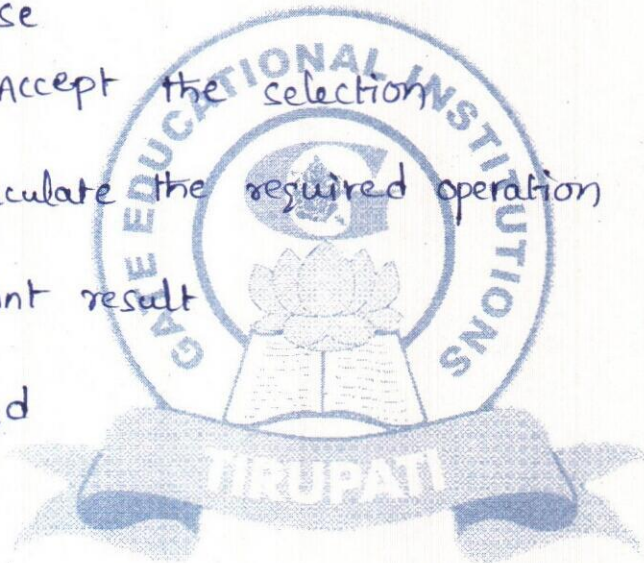
step 6 : else

-Accept the selection

step 7 : calculate the required operation

step 8 : print result

step 9 : end



1.To write a java program to implement operators and conditional statements

Import java.util.*;
Date :

GATE Educational Institutions

Page No. :

Class Experiment{

Public static void main(String args[]){

Practical No. :

Int a,b,result=0,choice=0;

Scanner sc = new Scanner(System.in);

System.out.println("Enter any two numbers :");

a = sc.nextInt();

b = sc.nextInt();

if(a==0&&b==0){

System.out.println("The Result will be :"+result);}

Else{

System.out.println("Please select your option :");

System.out.println("1.addition\n 2.substraction\n 3.multiplication\n 4.division\n");

Choice sc.nextInt();

Switch(choice){

Case 1: result =a+b; break;

Case 2: result =a-b;break;

Case 3: result =a*b;break;

Case 4: result =a/b;break;

Default:System.out.println("Invalid option");}}

System.out.println("The result is :"+result);

}}

Output:

Enter any two numbers

10

2

Please select your option

1.addition

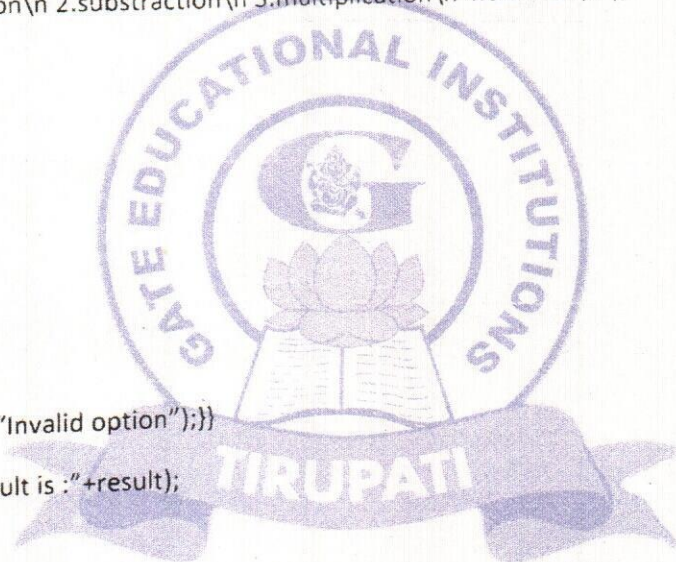
2.substraction

3.multiplication

4.division

3

The result is : 20



CD, math, 2

Date :	GATE Educational Institutions	Page No.:
		Practical No. :

Output :-

Enter any two numbers

26

28

please select your option

1. addition

2. subtraction

3. Multiplication

4. division

1

The result is : 54

Conclusion :

The above java program executed successfully.

Date :	GATE Educational Institutions	Page No.:
	ARRAYS	Practical No. :

Problem definition:-

Design and develop a java program to implement the types of arrays

Description:-

An array is a collection of similar data types.

Array is a container object that holds values of homogeneous types. It is also known as static data structure because size of an array must be specified at the time of its declaration.

Algorithm:-

step 1: start

step 2: declare i, j;

step 3: declare array, a[];

step 4: print "length of given array is" + n

step 5: print "enter elements into array is" + n

step 6: for (i=0; i<n; i++)

step 7: a[i] = sc.nextInt();

step 8: print "elements before sorting"

step 9: for (i=0; i<n; i++)

Date :	GATE Educational Institutions	Page No.:
		Practical No. :

step 10 : print (a[i]);

step 11 : for (i=0; i<n; i++)

step 12 : if (a[i] < a[j])

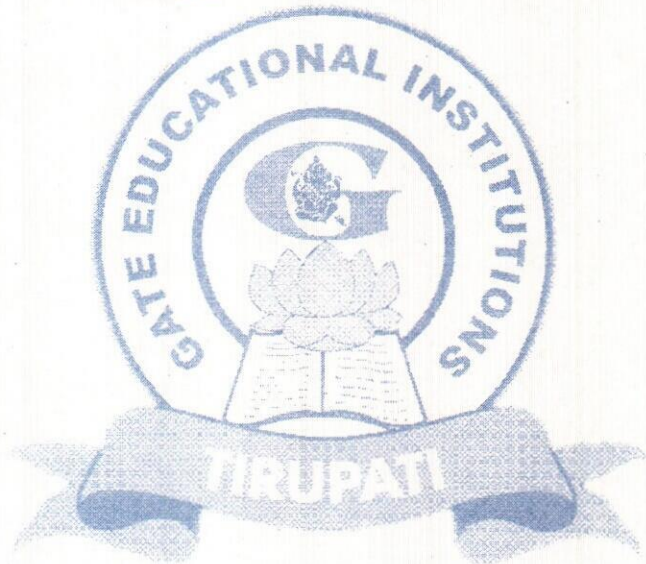
step 13 : for (int j=1; j<n; j++)

step 14 : a[j] to temp $\Rightarrow$ temp = a[i]

a[j] to a[i] $\Rightarrow$ a[i] = a[j]

temp to a[j] $\Rightarrow$ a[j] = temp;

step 15 : end



Date :	GATE Educational Institutions	Page No. :
	1 .Design and develop a java program to implement the types of arrays.	Practical No. :
<pre> import java.util.*; class array1D { public static void main(String args[]) { int a[]=new int[5]; //int b[]; //b=a; int i,j; int n=a.length; Scanner sc=new Scanner(System.in); System.out.println("length of the given array is:"+n); System.out.println("enter elements into array a[:]"); for(i=0;i<n;i++) { a[i]=sc.nextInt(); } System.out.println("elements in the array before sorting:"); for(i=0;i<n;i++) { System.out.println(a[i]); } for(i=0;i<n;i++) { for(j=i+1;j<n;j++) { if(a[i]<a[j]) { int temp=a[i]; a[i]=a[j]; a[j]=temp; } } } } </pre>		

Date :	GATE Educational Institutions	Page No.:
		Practical No. :

output :-

length of the given array is : 3

enter elements in the array a[]:

99

15

67

Elements in the array before sorting:

99

15

67

elements in the array after sorting:

99

67

15

conclusion :-

The above java program executed successfully.

Date :	GATE Educational Institutions	Page No.:
	CONSTRUCTORS	Practical No. :

Problem Definition :-

To design and develop a java program to demonstrate types of constructors.

Description :-

A constructor is a block of codes similar to the method. It is called when an instance of the class is created. At the time of calling constructor, memory for the object is allocated in the memory.

Algorithm :-

Step 1 : start

Step 2 : Import scanner class

Step 3 : Create an object for scanner class.

Step 4 : Declare the area()

Step 5 : Read the radius of the circle then

$$\text{area} = 3.14 * r * r$$

Step 6 : Declare area() for triangle then

$$\text{area of triangle} = (i * j) / 2$$

Step 7 : Find the area of sphere = $3.14 * 6 * g * g$

Step 8 : calculate area of rectangle = $u * v$

Step 9 : Declare the main class and main method.

Date :	GATE Educational Institutions	Page No.:
		Practical No. :

Step 10: declare int ch

Step 11: start the do loop

Step 12: enter the type of constructor you need to perform

Step 13: start the switch statement
switch (ch)

Step 14: if case is 1 then
area a₁ = new area ();

Step 15: if case is 2 then
area a₂ = new area (h, b);

Step 16: if case is 3 then
area a₃ = new area (s);

Step 17: if case is 4 then
area a₄ = new area (l, b);

Step 18: default :
In valid option

Step 19: End the switch statement

Step 20: end the do while loop

Step 21: end

Date :	GATE Educational Institutions	Page No. :
2. Design and develop a java program to demonstrate constructors.	types of	Practical No. :
<pre> import java.util.*; class area { Scanner sc=new Scanner(System.in); area() { System.out.println("\nEnter the radius of the circle"); int r=sc.nextInt(); System.out.println("\nThe area of the circle is : "+(3.14*r*r)); } area(int i,int j) { System.out.println("\nThe area of the triangle is : "+(i*j/2)); } area(float g) { System.out.println("\nThe area of the sphere is : "+(3.14*6*g*g)); } area(float u,float v) { System.out.println("\nThe area of the rectangle is : "+(u*v)); } } class java2 { public static void main(String args[]) { Scanner sc=new Scanner(System.in); int ch; </pre>		

Date :	GATE Educational Institutions	Page No. :
do {		Practical No. :

```

System.out.println("\nTypes of constructors\n1.Circle\n2.Triangle\n3.Sphere\n4.Rectangle\n");
System.out.println("\nEnter your choice");
ch=sc.nextInt();
switch(ch)
{
case 1:area a1=new area();
        break;
case 2:System.out.println("\nEnter the height and breadth of triangle");
        int h=sc.nextInt();
        int b=sc.nextInt();
        area a2=new area(h,b);
        break;
case 3:System.out.println("\nEnter the radius of the sphere");
        int s=sc.nextInt();
        area a3=new area(s);
        break;
case 4:System.out.println("\nEnter the length and breadth of triangle");
        int l=sc.nextInt();
        int br=sc.nextInt();
        area a4=new area(l,br);
        break;
default:System.out.println(" Invalid option");
}
}while(ch<5);
}
}

```


Date :	GATE Educational Institutions	Page No. :
<u>Output:</u>	Command Prompt <pre>C:\Users\HP\Desktop>javac java2.java C:\Users\HP\Desktop>java java2 Types of constructors 1.Circle 2.Triangle 3.Sphere 4.Rectangle Enter your choice 2 Enter the height and breadth of triangle 10 20 The area of the triangle is : 100 Types of constructors 1.Circle 2.Triangle 3.Sphere 4.Rectangle Enter your choice 5 Invalid option C:\Users\HP\Desktop></pre>	Practical No. :



Date :	GATE Educational Institutions	Page No.:
		Practical No. :

Output :-

Types of constructors

1. circle
2. Triangle
3. sphere
4. Rectangle

Enter your choice

4

Enter the length and breadth of rectangle

4

5

The area of rectangle

20

Conclusion :-

The above java program executed successfully.

Date :	GATE Educational Institutions	Page No.:
	METHOD OVERLOADING	Practical No. :

Problem Definition:-

Design and develop a java program for implementation of method overloading

Description:-

If a class has multiple methods having same name but different in parameters, it is known as method overloading.

Algorithm:-

Step 1 : start

Step 2 : declare test() and print "No parameters"

Step 3 : declare test (int a) and print a

Step 4 : declare test (int a, int b) and print "a and b"

Step 5 : declare test (double a) and print "a"

Step 6 : create an object for class

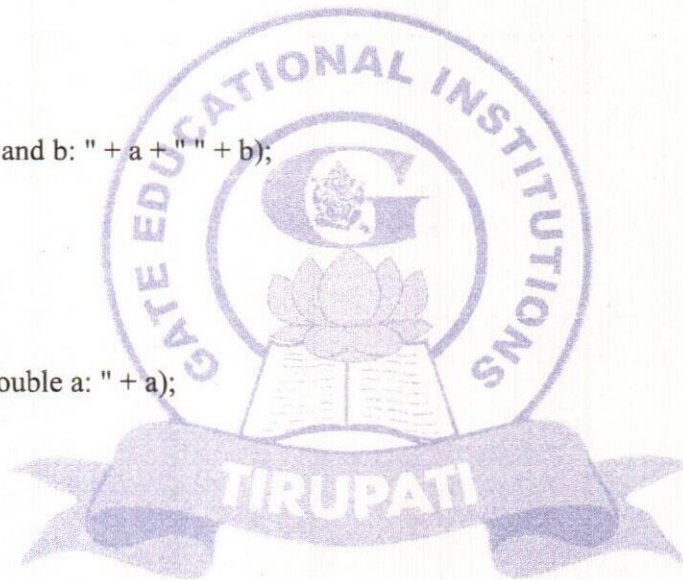
Step 7 : declare result variable

Step 8 : call the methods

Step 9 : print the result

Step 10 : end

Date :	GATE Educational Institutions	Page No. :
3. <u>Design and develop a java program for implementation of method overloading</u>	Practical No. :	
<pre> class OverloadDemo { void test() { System.out.println("No parameters"); } void test(int a) { System.out.println("a: " + a); } void test(int a, int b) { System.out.println("a and b: " + a + " " + b); } double test(double a) { System.out.println("double a: " + a); return a*a; } } class Overload { public static void main(String args[]) { OverloadDemo ob = new OverloadDemo(); double result; ob.test(); ob.test(10); ob.test(10, 20); result = ob.test(123.25); System.out.println("Result of ob.test(123.25): " + resu } </pre>		



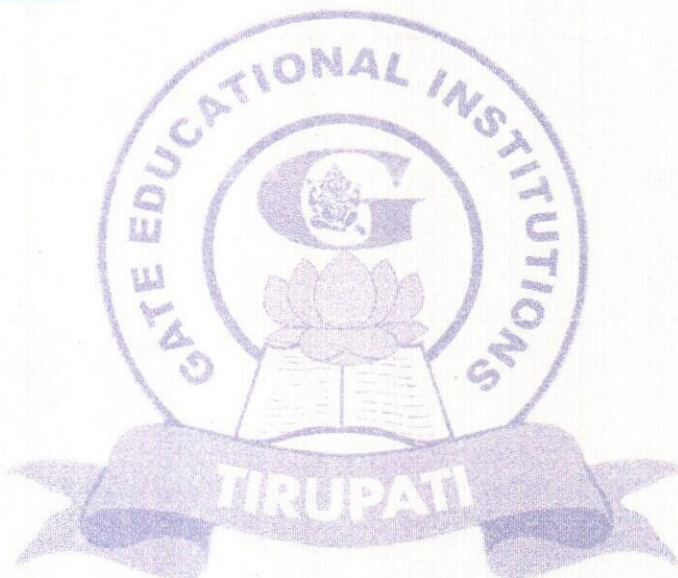
Date : }	GATE Educational Institutions	Page No. :
		Practical No. :

Output:

```

C:\Users\HP\Desktop>javac Overload.java
C:\Users\HP\Desktop>java Overload
No parameters
a: 10
a and b: 10.20
double a: 123.25
Result of ab.test(123.25): 15190.5625
C:\Users\HP\Desktop>

```



Date :	GATE Educational Institutions	Page No.:
		Practical No. :

out put :-

No parameters

a:30

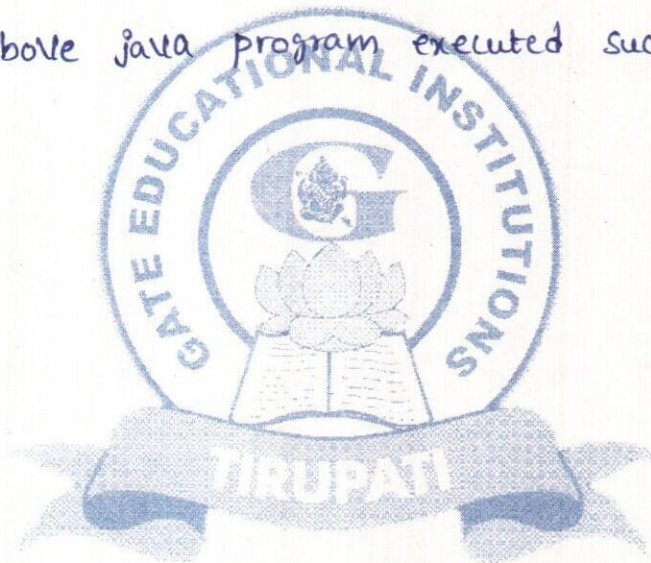
a and b: 40 50

double a:120.25

result of ab test (120.25): 14460.0625

conclusion :-

The above java program executed successfully



Date :	GATE Educational Institutions	Page No.:
	PACKAGES	Practical No. :

Problem Definition:—

Design and develop a Java program to implement the concept of packages.

Description:—

A Java package is a group of similar types of classes interfaces and sub-packages. It can be categorized in two forms built-in-package and user-defined package.

Algorithm:—

Step 1: Start

Step 2: Declare variables name, author, float, price, year

Step 3: create a constructor

```
public BookDetails (String a, String n, float p, int y)
```

Step 4: declare a method

```
print "name"
print "author"
print "price"
print "year"
```

Step 5: Create an object for constructor

Step 6: call the display()

Step 7: end

Date :	GATE Educational Institutions	Page No. :
4. <u>Write a java program to implement the concept of packages.</u>		Practical No. :
package book;		
<pre> public class BookDetails { String name,author; float price; int year; public BookDetails(String n,String a,float p,int y) { name=n; author=a; price=p; year=y; } public void display() { System.out.println("Book Name:"+name); System.out.println("Book Author:"+author); System.out.println("Book Price:Rs "+price); System.out.println("Year of Publishing:"+year); } } import book.*; class BookDemo { public static void main(String args[]) { </pre>		

Date :	GATE Educational Institutions	Page No. :
<pre> BookDetails b=new BookDetails("Java Programming", "Balguruswamy", 190.00f,2007); b.display(); } } </pre>		Practical No. :
<u>Output:</u> Book Name :Java Programming Book Author :Balguruswamy Book Price :Rs 190.0 Year of Publishing:2007 		

Date :	GATE Educational Institutions	Page No.:
		Practical No. :

output :-

Book Name : wings of fire

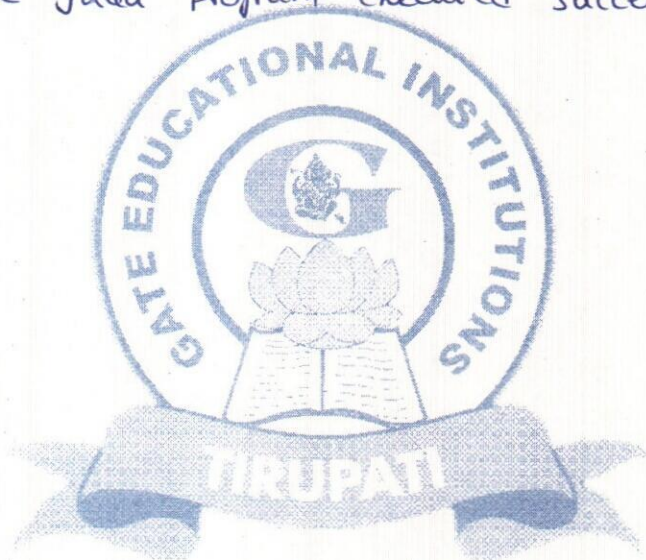
Book Author : Abdul Kalam

price : 2000.00f

Year of publishing : 2009

conclusion :-

The above Java program executed successfully



Date :	GATE Educational Institutions	Page No.:
	EXCEPTION HANDLING	Practical No. :

problem definition:-

design and develop a java program in a java to implement the concept of exception handling.

Description:-

The exception handling in java is one of the powerful mechanism to handle the run time errors so that the normal flow of the application can be maintained.

Algorithm:-

step 1: start

step 2: import java.io.* package and
java.lang.* package

step 3: write exception class

step 4: declare and print msg

step 5: class exception 1 starts

step 6: declare variable

eno, ename, job, sal, count

step 7: try

for (i = count; i < 3; i++)

Date :	GATE Educational Institutions	Page No.:
		Practical No. :

Step 8: print the employee details

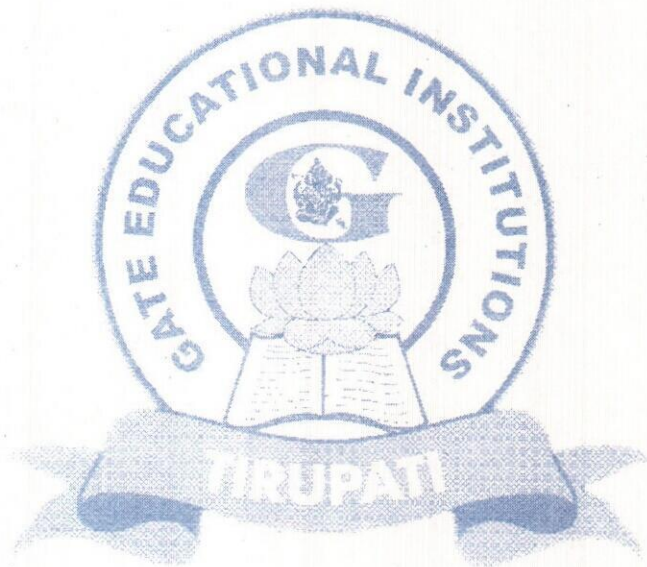
Step 9: catch (myexceptione)

Step 10: print ("exception caught" + e)

Step 11: call disp()

Step 12: print "e.no, ename, job, sal";

Step 13: end



Date :	GATE Educational Institutions	Page No. :
5. Design and develop a program in java to implement the concept of <u>Exception Handling</u>		Practical No. :
<pre> import java.io.*; import java.lang.*; class MyException extends Exception { String msg; public MyException(String message) { msg=message; } public String toString() { return msg; } } class exception9 { static int eno[]=new int[5]; static String ename[]=new String[3]; static String job[]=new String[3]; static int sal[]=new int[3]; </pre>		

Date :	GATE Educational Institutions	Page No. :
static int count=0;		Practical No. :
static void get() throws IOException <pre data-bbox="196 347 884 1942"> { int i; DataInputStream d=new DataInputStream(System.in); try { for(i=count;i<3;i++) { System.out.println("\n enter"+(i+1)+"employee detail"); System.out.println("\n enter employee no. and name:"); eno[i]=Integer.parseInt(d.readLine()); if(eno[i]<=0) throw new MyException("entr valid no."); ename[i]=d.readLine(); System.out.println("\n enter job and salary"); job[i]=d.readLine(); sal[i]=Integer.parseInt(d.readLine()); if(sal[i]<0) throw new MyException("enter valid salary"); count++; } } catch(MyException e) { System.out.println("exception caught:"+e); get(); } } static void disp() </pre>		

Date :	GATE Educational Institutions	Page No. :
{ for(int i=0;i<3;i++) { System.out.println(eno[i]+"\\t\\t"+ename[i]+"\\t\\t"+job[i]+"\\t\\t"+sal[i]); } } public static void main(String args[]) throws IOException { get(); System.out.println("\\n eno \\t\\t ename \\t\\t job \\t\\tsalary\\n"); disp(); } }	Practical No. :	

Output:

Enter 1 employee details:
Enter employee no and name : 5554
Exception caught : enter valid number
Enter 1 employee detail :
Enter employee no and name : 3001
R.Haricharan
Enter job and salary : lecturer 25000
Enter 2 employee detail :
Enter employee no and name : 5001
S.Kalaiselvan
Enter job and salary : reader
20000
Enter 3 employee detail :
Enter employee no and name : 7801
S.Rajadurai
Enter job and salary : director
23000
35

ENO	ENAME	JOB	SALARY
3001	R.Haricharan	lecturer	25000
500	S.Kalaiselvan	reader	20000
7801	S.Rajadurai	director	23000

Date :	GATE Educational Institutions	Page No.:
		Practical No. :

output :-

enter 1 employee details:

enter employee no and name:

1

Satheesh

enter job and salary:

Software

2,00,000

enter 2 employee details:

enter employee no and name:

2

Phruva

enter job and salary:

Manager

2,90,000

enter 3 employee details:

enter employee no and name:

3

Nandha

enter job and salary:

Hardware

2,50,000

ENO

ENAME

JOB

1

Satheesh

Software

2

Phruva

Manager

3

Nandha

Hardware

Date :	GATE Educational Institutions	Page No.:
		Practical No. :

conclusion :-

The above java program executed successfully



Date :	GATE Educational Institutions	Page No.:
	MULTI THREADING	Practical No. :

Problem Definition:—

Design and develop a program in java to demonstrate the concept of multithreading.

Description:—

Multithreading is a model of program execution that allows for multithreads to be created within a process execution independently but concurrently sharing process resources depending on the hardware, threads can run fully parallel if they are distributed to their own CPU core.

Algorithm:—

Step 1:— start

Step 2: create even thread with extends thread

Step 3: create run () method with try and catch blocks

Step 4: create new class odd thread which extend thread i.e, thread 2.

Step 5: create run method with try and catch block of code.

Step 6: try block of code which prints even numbers and exit from thread.

Date :	GATE Educational Institutions	Page No.:
		Practical No. :

Step 7: create main class thread demo

Step 8: call the methods that had initialized in even thread, add thread.

even thread e new;

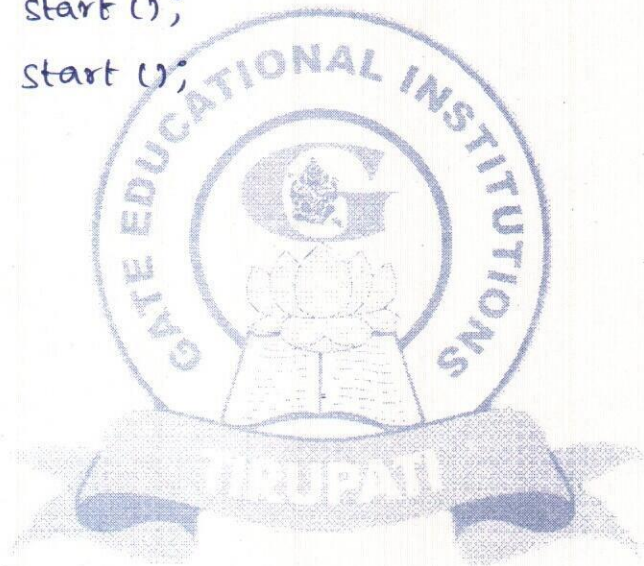
odd thread o new;

Step 9: start summing thread using keyword

e. start ();

o. start ();

Step 10: End



Date :

GATE Educational Institutions

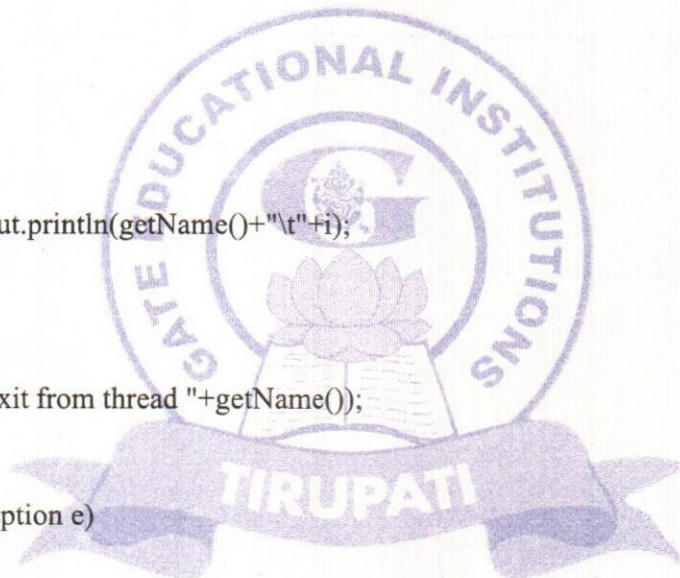
Page No. :

6. Design and develop a program in java to demonstrate the concept of multithreading.

Practical No. :

```
class EvenThread extends Thread
{
    EvenThread(String name)
    {
        super(name);
    }
    public void run()
    {
        try
        {
            for(int i=1;i<15;i++)
            {
                if (i%2==0) System.out.println(getName()+"\t"+i);
                Thread.sleep(1000);
            }
            System.out.println("Exit from thread "+getName());
        }
        catch (InterruptedException e)
        {
            System.out.println("Thread "+getName()+" Interrupted");
        }
    }
}

class OddThread extends Thread
{
    OddThread(String name)
    {
        super(name);
    }
```



Date :	GATE Educational Institutions	Page No. :
public void run() {		Practical No. :


```

try
{
for(int i=1;i<15;i++)
{
if (i%2!=0)
System.out.println(getName()+"\t"+i);
Thread.sleep(500);
}
System.out.println("Exit from thread "+getName());
}
catch(InterruptedException e)
{
System.out.println("Thread "+getName()+" Interrupted");
}
}

class ThreadDemo
{
public static void main(String args[])
{
EvenThread e=new EvenThread("Even");
OddThread o=new OddThread("Odd");
e.start();
o.start();
}
}

```

Output:

Date :

GATE Educational Institutions

Page No. :

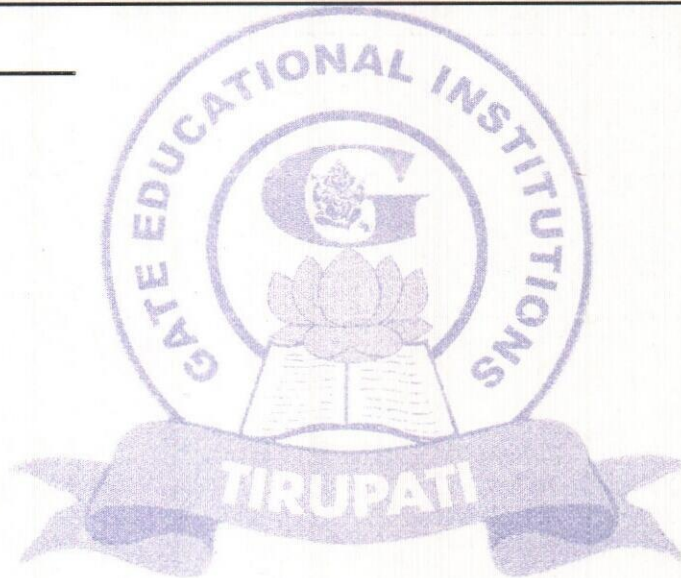
```
C:\Users\System8\Desktop>javac OddThread.java
```

Practical No. :

```
C:\Users\System8\Desktop>java ThreadDemo
```

```
Odd      1
Even     2
Odd      3
Odd      5
Even     4
Odd      7
Odd      9
Even     6
Odd     11
Odd     13
Even     8
Exit from thread Odd
Even    10
Even    12
Even    14
Exit from thread Even
```

```
C:\Users\System8\Desktop>^U^U_
```



Date :	GATE Educational Institutions	Page No.:
		Practical No. :

output :-

odd 1

even 2

odd 3

odd 5

even 4

odd 7

odd 9

even 6

odd 11

odd 13

even 8

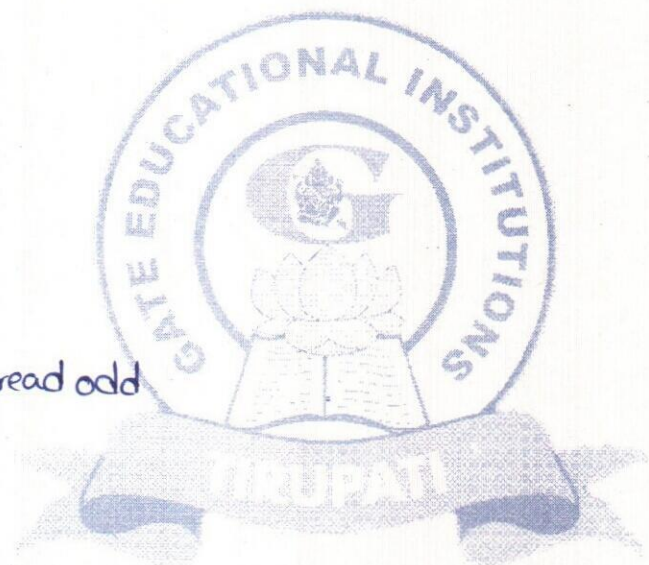
exit from thread odd

even 10

even 12

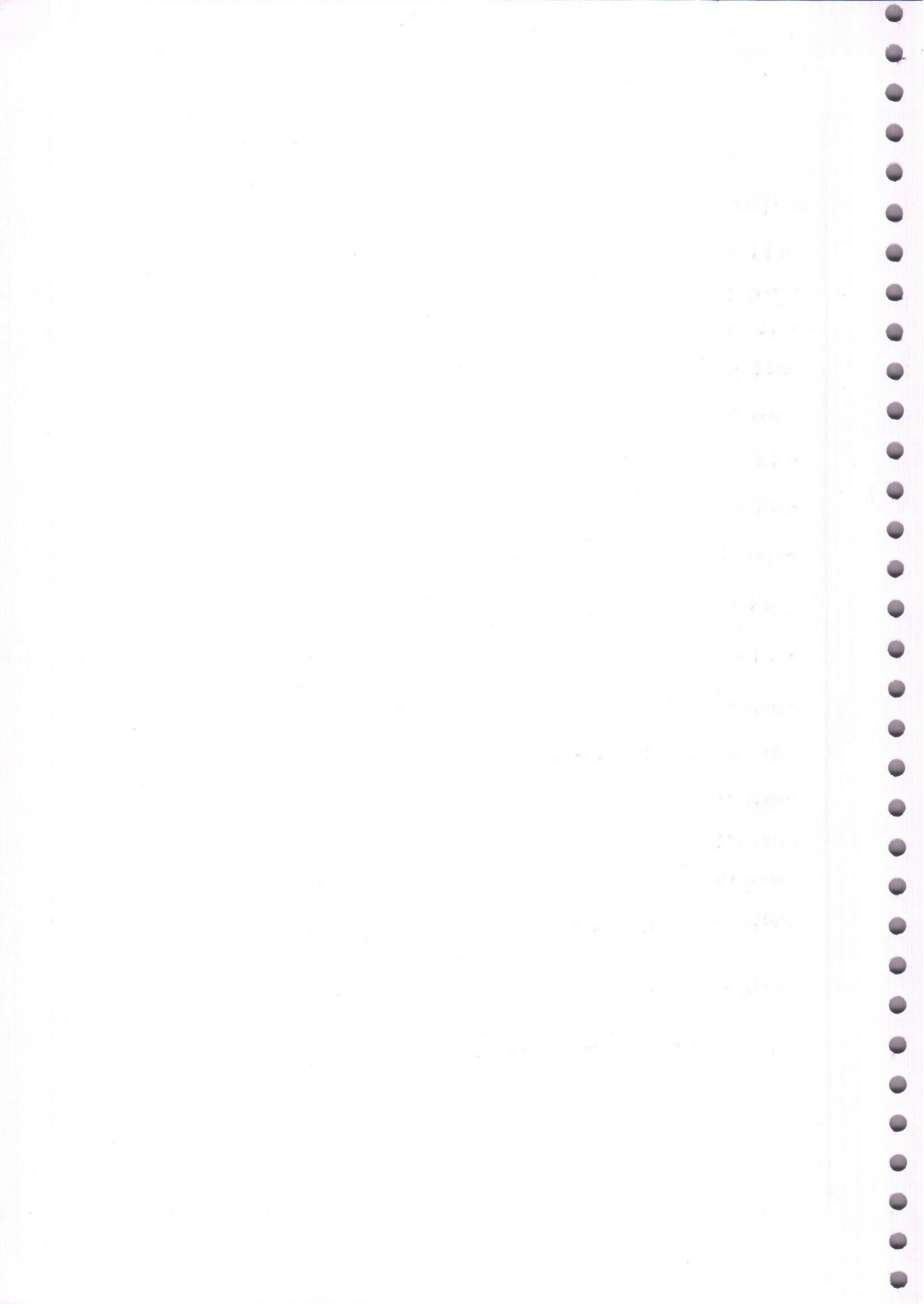
even 14

exit from thread even



conclusion :-

The above java program executed successfully.



Date :	GATE Educational Institutions	Page No.:
	VECTORS	Practical No. :

Problem Definition:-

TO write a menu based program in java, that access a shopping list of items from command line and store in a vector and perform operations.

Description:-

- * It contains components that can be accessed using an integer index similar to an array.
- * The size of a vector can grow or shrink as need to accomodate adding and removing items
- * Vector methods are add elements, element At(), remove element, copyInto (Array)

Algorithm:-

- Step 1: start the program
- Step 2: create a class with name VectorDemo
- Step 3: Create object 'v' for vector class with arguments.
- Step 4: print 'v' size and capacity
- Step 5: Add elements into vector, with integer type
- Step 6: print capacity after 4 additions

Date :	GATE Educational Institutions	Page No.:
		Practical No. :

step 7: Add 6 more elements into 'v' & print the capacity of 'v'.

step 8: print the first & last elements in v.

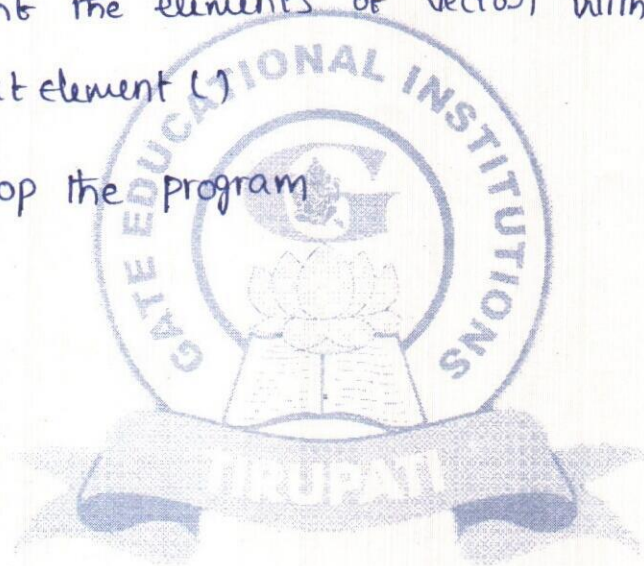
step 9: check the 'v' contains '3'

if (v contains (new Integer(3)))

step 10: Create VE num object for enumeration

step 11: print the elements of vector with VE num
next element (1)

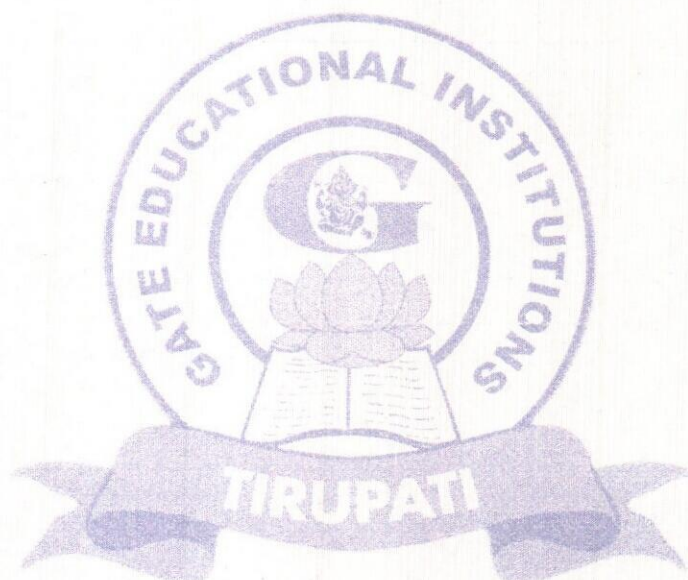
step 12: Stop the program



Date :	GATE Educational Institutions	Page No. :
7. Write a menu based program in java that occurs a shopping list of items from command line and store in a vector & perform operations.		Practical No. :
<pre>import java.util.*; public class VectorDemo { public static void main(String args[]) { Vector v = new Vector(3, 2); System.out.println("Initial size: " + v.size()); System.out.println("Initial capacity: " + v.capacity()); v.addElement(new Integer(1)); v.addElement(new Integer(2)); v.addElement(new Integer(3)); v.addElement(new Integer(4)); System.out.println("Capacity after four additions: " + v.capacity()); v.addElement(new Double(5.45)); System.out.println("Current capacity: " + v.capacity());</pre>		

Date :	GATE Educational Institutions	Page No. :
v.addElement(new Double(6.08)); v.addElement(new Integer(7));		Practical No. :
<pre> System.out.println("Current capacity: " + v.capacity()); v.addElement(new Float(9.4)); v.addElement(new Integer(10)); System.out.println("Current capacity: " + v.capacity()); v.addElement(new Integer(11)); v.addElement(new Integer(12)); System.out.println("First element: " + (Integer)v.firstElement()); System.out.println("Last element: " + (Integer)v.lastElement()); if(v.contains(new Integer(3))) System.out.println("Vector contains 3."); Enumeration vEnum = v.elements(); System.out.println("\nElements in vector:"); while(vEnum.hasMoreElements()) System.out.print(vEnum.nextElement() + " "); System.out.println(); } } </pre>		
<u>Output:</u>		

Date :	GATE Educational Institutions	Page No. :
	Command Prompt Microsoft Windows [Version 10.0.17763.168] (c) 2018 Microsoft Corporation. All rights reserved. C:\Users\HP>cd Downloads C:\Users\HP\Downloads>cd ~\java rem prog C:\Users\HP\Downloads\java rem prog>javac VectorDemo.java Note: VectorDemo.java uses or overrides a deprecated API. Note: Recompile with -Xlint:deprecation for details. Note: VectorDemo.java uses unchecked or unsafe operations. Note: Recompile with -Xlint:unchecked for details. C:\Users\HP\Downloads\java rem prog>java VectorDemo Initial size: 0 Initial capacity: 3 Capacity after four additions: 5 Current capacity: 5 Current capacity: 7 Current capacity: 9 First element: 1 Last element: 12 Vector contains 3. Elements in vector: 1 2 3 4 5.45 6.08 7 9.4 10 11 12 C:\Users\HP\Downloads\java rem prog>_	Practical No. :



Date :	GATE Educational Institutions	Page No.:
		Practical No. :

output :-

Initial size : 0

Initial capacity : 3

Capacity after four additions : 5

Current capacity : 5

current capacity : 7

current capacity : 9

First element : 1

last element : 12

vector contains 3

elements in vector :

1

2

3

4

5.45

6.08

7

9.4

10

11

12

conclusion :-

The above java program executed successfully

Date :	GATE Educational Institutions	Page No.:
	STRING OPERATIONS	Practical No. :

Problem Definition:-

Design and develop a java program to perform string operations.

Description:-

String is a sequence of characters. It is immutable which means it cannot be changed. It is basically an object like an array of characters works same as java string. String class provides a lot of methods to perform operations on string such as `concat()`, `compare()`, `equals()`, `split()` etc.

Algorithm:-

Step 1: start

Step 2: create class stringop

Step 3: create methods `conc()`, `comp()`, `icompe()`, `upper()`, `lower()`, `lenc()`, `sub()`.

Step 4: create class java8 with main method

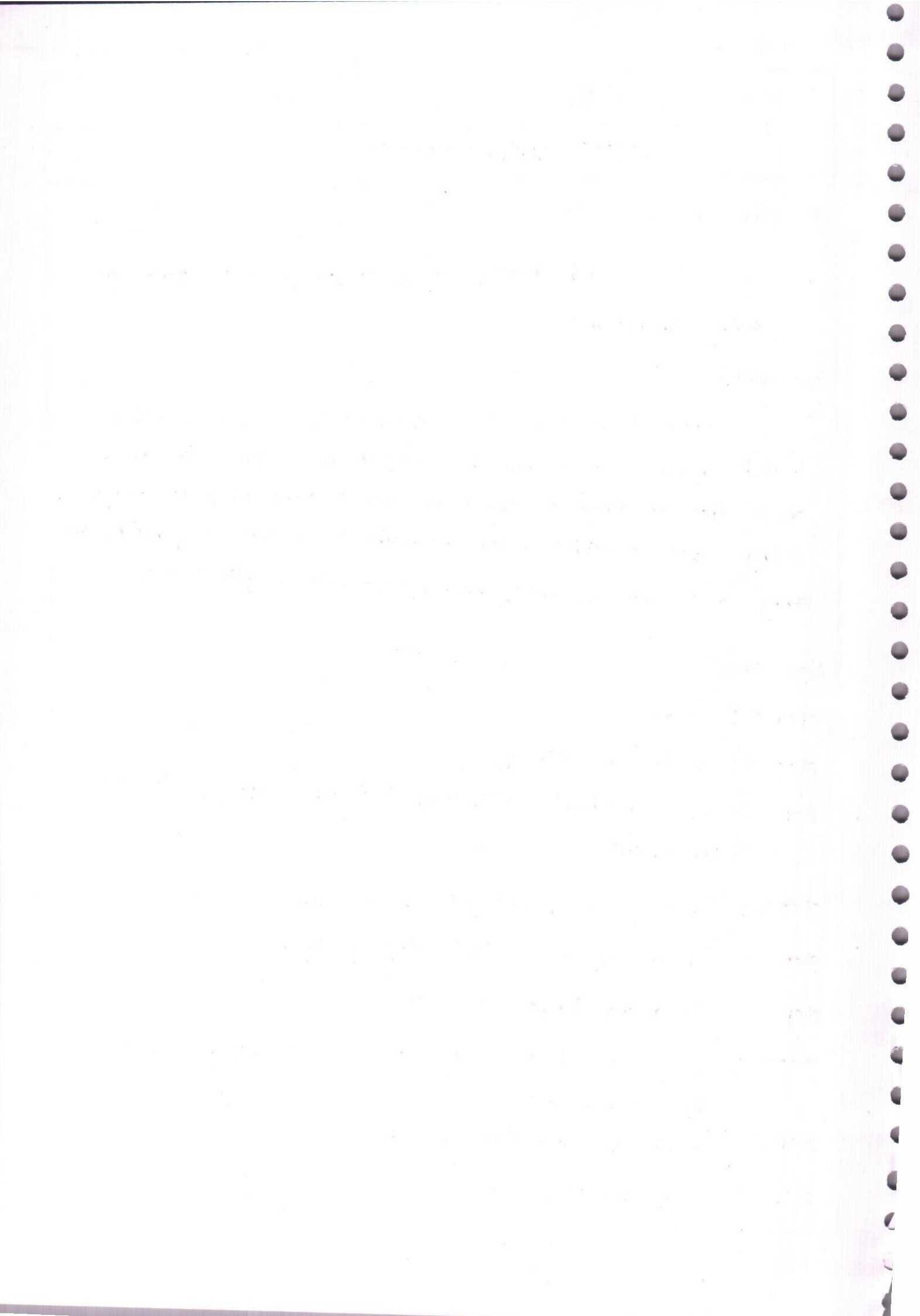
Step 5: create object 's' for string op class.

Step 6: Read the input from user

Step 7: using switch to select the choice of operations by switch cases

Step 8: print the operation result.

Step 9: stop the program.



Date :	GATE Educational Institutions	Page No. :
8. <u>Develop a program in java to perform string operations.</u>	Practical No. :	
<pre> import java.io.*; import java.util.*; import java.lang.*; class stringop { Scanner sc=new Scanner(System.in); void con() { String a; String b; System.out.println("Enter First String"); a=sc.next(); System.out.println("Enter Second String"); b=sc.next(); String c=a+b; System.out.println("\nUsing + Operator : "+c); c=a.concat(b); System.out.println("\n Using concatenation method : "+c); } void comp() { String a; String b; System.out.println("Enter First String"); a=sc.next(); System.out.println("Enter Second String"); b=sc.next(); if(a.equals(b)) { System.out.println("\nBoth strings are exactly equal"); } else { </pre>		

Date :	GATE Educational Institutions	Page No. :
System.out.println("\nBoth strings are exactly not equal"); }		Practical No. :
<pre> void icomp() { String a; String b; System.out.println("Enter First String"); a=sc.next(); System.out.println("Enter Second String"); b=sc.next(); if(a.equalsIgnoreCase(b)) { System.out.println("\nBoth strings are equal"); } else { System.out.println("\nBoth strings are not equal"); } } void upper() { String a; System.out.println("\nEnter a string in lower case"); a=sc.next(); System.out.println("\nOriginal String is : "+a); System.out.println("\nThe String in upper case : "+a.toUpperCase()); } void lower() { String a; System.out.println("\nEnter a string in upper case"); a=sc.next(); System.out.println("\n Original string is : "+a); System.out.println("\n String in to lower case : "+a.toLowerCase()); } void len() { String a; String b; System.out.println("Enter First String"); a=sc.next(); System.out.println("Enter Second String"); b=sc.next(); System.out.println("\n The length of the second string is : "+a.length()); </pre>		

Date :	GATE Educational Institutions	Page No. :
System.out.println("\n\nThe length of the second string is : "+b.length()); }		Practical No. :
<pre> void sub() { String a; a="S.V.University,Tirupati"; System.out.println(a.substring(4,14)); System.out.println(a.substring(4)); } } class java8 { public static void main(String args[]) { int ch; stringop s=new stringop(); Scanner sc=new Scanner(System.in); do { System.out.println("\nString Operations\n1.String Concatination\n2.String Compare\n3.String Compare with ignore case\n4.String to Upper case\n5.String to lower case\n6.String length\n7.String sub string\n8.Exit\n"); System.out.println("\nEnter your option"); ch=sc.nextInt(); switch(ch) { case 1:s.con(); break; case 2:s.comp(); break; case 3:s.icomp(); break; case 4:s.upper(); break; case 5:s.lower(); break; case 6:s.len(); break; case 7:s.sub(); break; default: System.out.println("\n***** Invalid *****"); } }while(ch<8); } } </pre>		

Date :	GATE Educational Institutions	Page No. :
Output:		Practical No. :

String Operations

- 1.String Concatination
- 2.String Compare
- 3.String Compare with ignore case
- 4.String to Upper case
- 5.String to lower case
- 6.String length
- 7.String sub string
- 8.Exit

Enter your option

1

Enter First String

raja

Enter Second String

chandu

Using + Operator : rajachandu

Using concatenation method : rajachandu

String Operations

- 1.String Concatination
- 2.String Compare
- 3.String Compare with ignore case
- 4.String to Upper case
- 5.String to lower case
- 6.String length
- 7.String sub string
- 8.Exit

Enter your option

2

Enter First String

raja

Enter Second String

RAJA

Both strings are exactly not equal

String Operations

- 1.String Concatination
- 2.String Compare
- 3.String Compare with ignore case
- 4.String to Upper case
- 5.String to lower case
- 6.String length
- 7.String sub string
- 8.Exit

Enter your option

3

Enter First String

CHANDU

Date :	GATE Educational Institutions	Page No. :
Enter Second String chandu		Practical No. :

Both strings are equal

String Operations

- 1.String Concatination
- 2.String Compare
- 3.String Compare with ignore case
- 4.String to Upper case
- 5.String to lower case
- 6.String length
- 7.String sub string
- 8.Exit

Enter your option

4

Enter a string in lower case

raja

Original String is : raja

The String in upper case : RAJA

String Operations

- 1.String Concatination
- 2.String Compare
- 3.String Compare with ignore case
- 4.String to Upper case
- 5.String to lower case
- 6.String length
- 7.String sub string
- 8.Exit

Enter your option

5

Enetr a string in upper case

CHANDU

Original string is : CHANDU

String in to lower case : chandu

String Operations

- 1.String Concatination
- 2.String Compare
- 3.String Compare with ignore case
- 4.String to Upper case
- 5.String to lower case
- 6.String length
- 7.String sub string
- 8.Exit

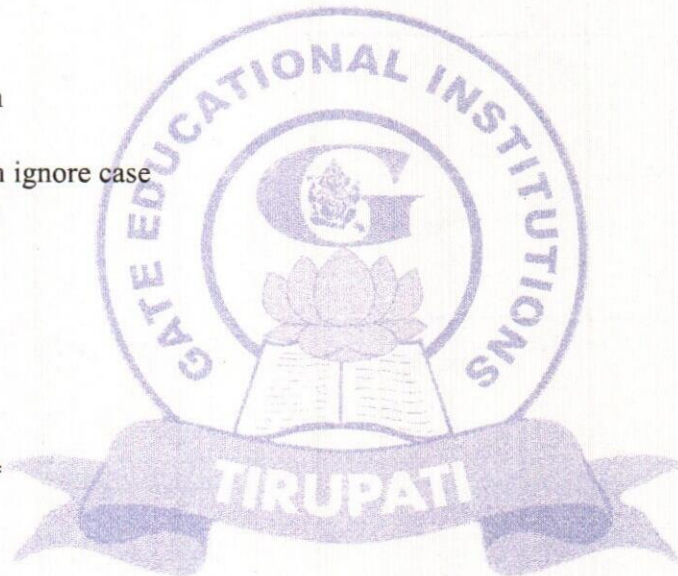
Enter your option

6

Enter First String

rajachandu

Date :	GATE Educational Institutions	Page No. :
Enter Second String chandra		Practical No. :
The length of the second string is : 10 The length of the second string is : 7 String Operations 1.String Concatination 2.String Compare 3.String Compare with ignore case 4.String to Upper case 5.String to lower case 6.String length 7.String sub string 8.Exit Enter your option 7 University University,Tirupati String Operations 1.String Concatination 2.String Compare 3.String Compare with ignore case 4.String to Upper case 5.String to lower case 6.String length 7.String sub string 8.Exit Enter your option 8 ***** Invalid *****		



Date :	GATE Educational Institutions	Page No.:
		Practical No. :

output :-

string operations

1. string concatenation
2. string compare
3. string compare with ignore case
4. string to upper case
5. string to lower case
6. string length
7. string sub string
8. exit

enter your option 1

enter first string raja

enter second string chandhu

using + operator : rajachandhu

using concatenation method :

rajachandhu

conclusion :-

The above java program executed successfully.

Date :	GATE Educational Institutions	Page No.:
	CALENDAR	Practical No. :

problem definition:-

Design and develop a java program to display calendar data by using the java utilities.

Description:-

Java calendar class is an abstract class that provides method for converting data b/w a specific instant in time and set of calendar fields such as month, year, hour etc.

Algorithm:-

step 1 : start

step 2 : declare variables day, month, year

step 3 : declare second, minute, hour

step 4 : create object

step 5 : Read day:

day = date.get(Calendar, DAY-OF-MONTH);

step 6 : read month and year

month = date.get(Calendar, MONTH);

year = date.get(Calendar, YEAR);

step 7 : read second, minute and hour

second = date.get(Calendar, SECOND);

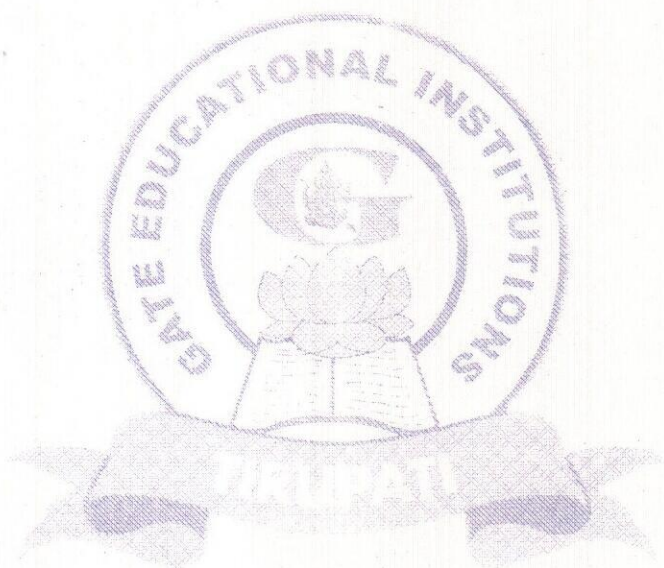
minute = date.get(Calendar, MINUTE);

hour = date.get(Calendar, HOUR);

Date :	GATE Educational Institutions	Page No.:
		Practical No. :

step8: print day, month, year, second, minute, hour.

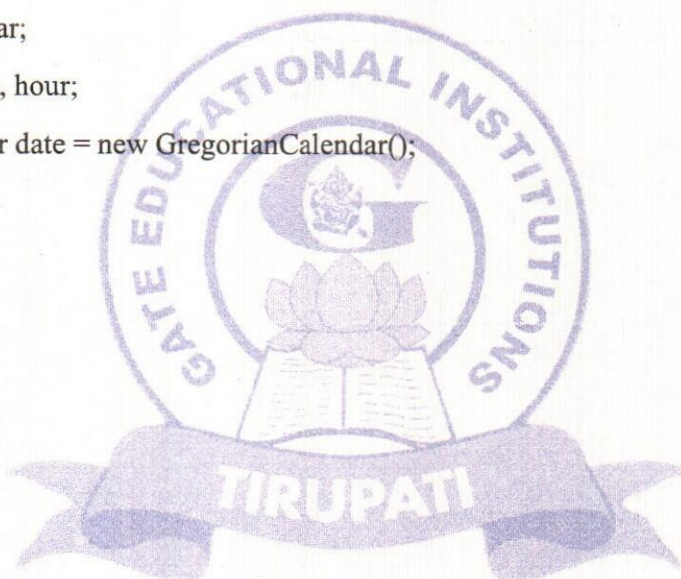
step9: end



Date :	GATE Educational Institutions	Page No. :
		Practical No. :

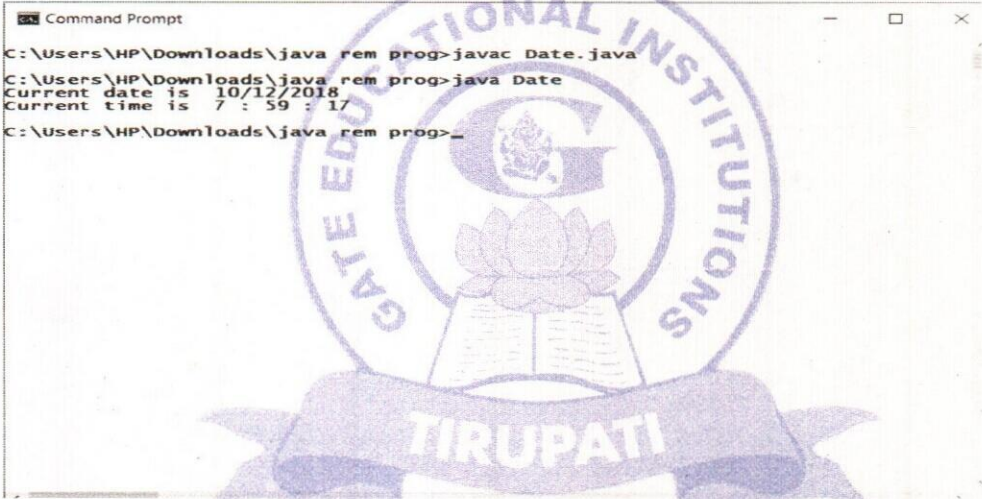
9. Develop a program in java to display calendar data by using the java utilities.

```
import java.util.*;
class Date
{
    public static void main(String args[])
    {
        int day, month, year;
        int second, minute, hour;
        GregorianCalendar date = new GregorianCalendar();
```



Date :	GATE Educational Institutions	Page No. :
day = date.get(Calendar.DAY_OF_MONTH); month = date.get(Calendar.MONTH); year = date.get(Calendar.YEAR); second = date.get(Calendar.SECOND); minute = date.get(Calendar.MINUTE); hour = date.get(Calendar.HOUR); System.out.println("Current date is "+day+"/"+(month+1)+"/"+year); System.out.println("Current time is "+hour+" : "+minute+" : "+second); } }	Practical No. :	

Output:



Date :	GATE Educational Institutions	Page No.:
		Practical No. :

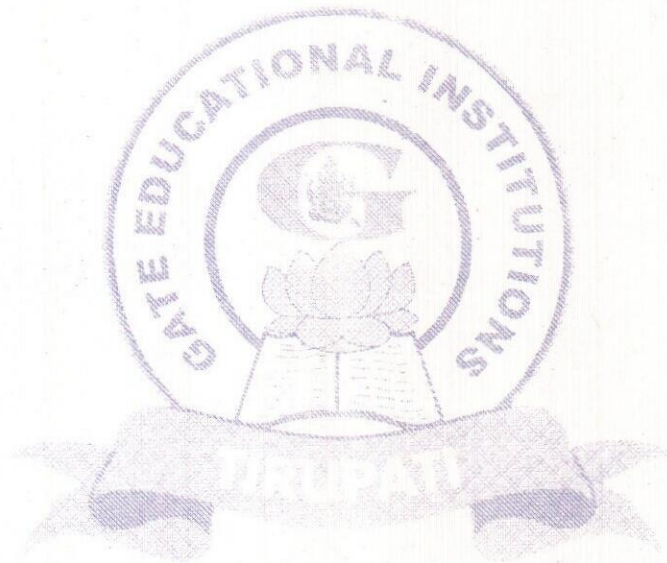
output :-

current date is 26/06/2022

current time is 12:00

conclusion :-

. The above java program executed successfully.



Date :	GATE Educational Institutions	Page No.:
	APPLET	Practical No. :

Problem Definition:-

Design and develop a java program to create an applet to draw a bar chart from the following details.

Description:-

Applets are small java programs that are primarily used in internet computing. They can be transported over the internet from one computer to another and run using the Applet Viewer or any web browser that supports java.

Algorithm:-

Step 1 : start

Step 2 : Import applet and awt package

Step 3 : Initialize n, label, value.

Step 4 : create int()

Step 5 : Read label and value

Step 6 : create point (Graphics g)

Step 7 : for (int i=0; i<n; i++)

Step 8 : set color and draw line

g.setColor (color.red);

g.drawLine (100, 50, 100, 350);

g.setColor (color.blue);

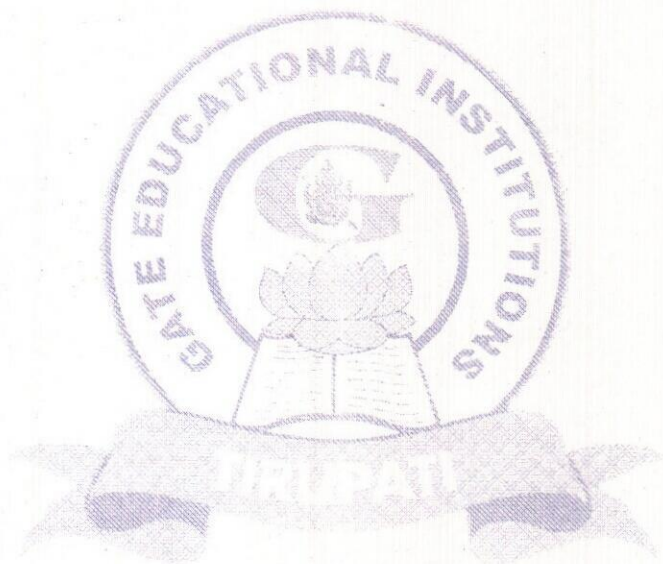
g.drawString (label[i], 40, 145 + (150 * i));

g.setColor (green);

Date :	GATE Educational Institutions	Page No.:
		Practical No. :

step 9 : end

step 10 : create html file



Date :	GATE Educational Institutions					Page No. :
10. Write a java program to create an applet to draw a bar chat from the following details.						Practical No. :
Subjects	OR	JAVA	OS	DS	DBMS	
Marks	60	65	70	85	90	
<pre> import java.applet.*; import java.awt.*; public class BarChart extends Applet { int n=0; String label[]; int value[]; </pre>						



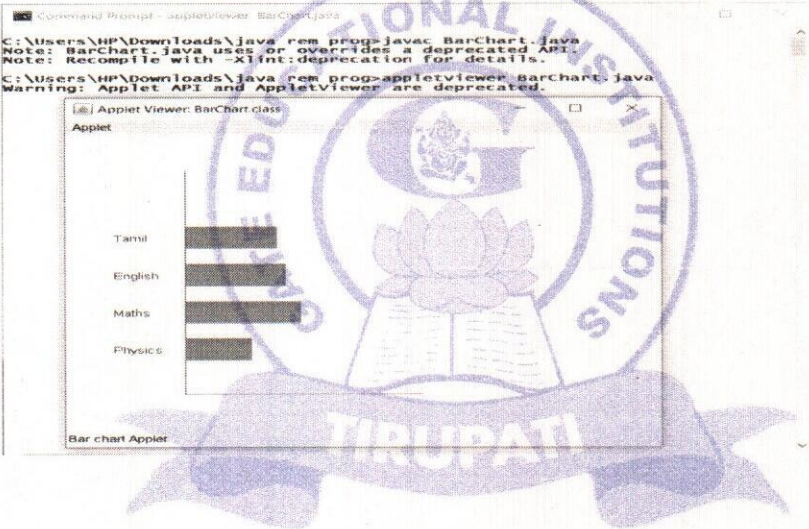
Date :	GATE Educational Institutions	Page No. :
public void init() {		Practical No. :

```

try
{
n= Integer.parseInt(getParameter("columns"));
label= new String[n];
value= new int[n];
label[0]=getParameter("label1");
label[1]=getParameter("label2");
label[2]= getParameter("label3");
label[3]= getParameter("label4");
value[0]=Integer.parseInt(getParameter("m1"));
value[1]=Integer.parseInt(getParameter("m2"));
value[2]=Integer.parseInt(getParameter("m3"));
value[3]= Integer.parseInt(getParameter("m4"));
}
catch(NumberFormatException e){ }
}

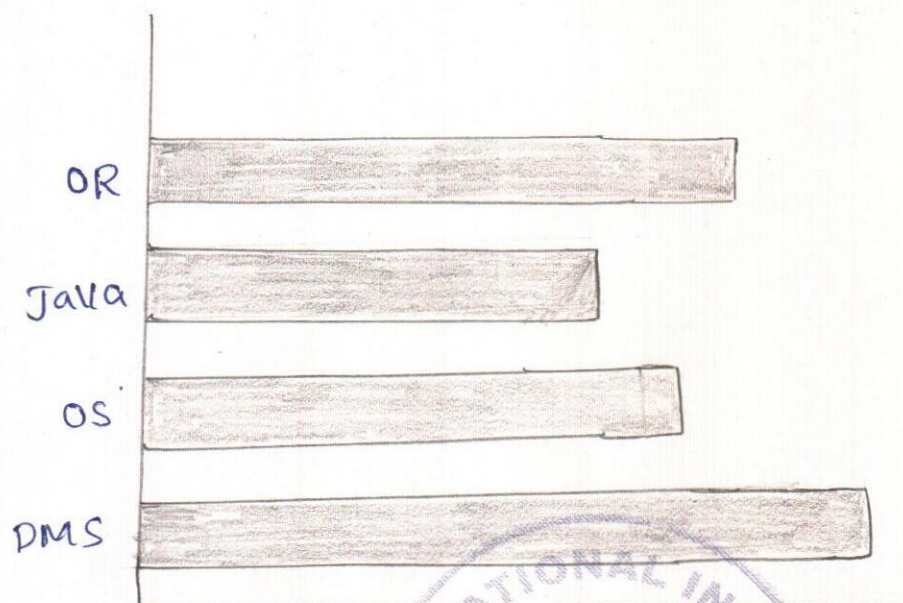
public void paint(Graphics g)
{
String msg=label[0]; for(int i=0;i<n;i++)
{
g.setColor(Color.red);
g.drawLine(100,50,100,350);
g.drawLine(100,350,300,350);
g.setColor(Color.blue);
g.drawString(label[i],40,145+(50*i));
g.setColor(Color.green);
g.fillRect(100,125+(50*i),value[i] ,30); showStatus("Bar chart Applet");
}
}
}

```


Date :	GATE Educational Institutions	Page No. :
/*<APPLET name="columns" VALUE=4>	CODE="BarChart.class" HEIGHT=400 WIDTH=500><PARAM Practical No. :	
<PARAM name="label1" VALUE="Tamil"><PARAM name="label2" VALUE="English"> <PARAM name="label3" VALUE="Maths"> <PARAM name="label4" VALUE="Physics"> <PARAM name="m1" VALUE="78"> <PARAM name="m2" VALUE="85"> <PARAM name="m3" VALUE="98"> <PARAM name="m4" VALUE="56"> </APPLET> */		
<u>Output:</u> 		

Date :	GATE Educational Institutions	Page No.:
		Practical No. :

output:-



conclusion:-

The above java program executed successfully.

Date :	GATE Educational Institutions	Page No.:
	SWINGS	Practical No. :

problem definition:-

Design and develop a java program to implement the concept of swings.

Description:-

Java swings are used to create window based application. It is built on the top of AWT (Abstract Window Toolkit) API and entirely written in java. It provides platform independent and lightweight components.

The java swing package provides classes for java swing API such as JButton, JTextField, JTextArea, JCheckBox etc.,

Algorithm:-

step 1: start

step 2: import java.awt.*;

step 3: declare main class method & extend

step 4: declare

JTabbedPane J1P = new JTabbedPane

step 5: declaring cities, colors and flavours

step 6: using extend keyword to class JPanel

step 7: initialize buttons 1 to 4

add(b1)

add(b2)

add(b3)

add(b4)

Date :	GATE Educational Institutions	Page No.:
		Practical No. :

step 8 : Again using extend keyword for colors panel.

step 9 : Initialize checkbox cb₁ to cb₃

add (cb₁)

add (cb₂)

add (cb₃)

step 10 : extend for flavourspanel class jcb.add item

("vanilla"), ("chocolate"), ("strawberry");

step 11 : Initialize Applet code

< applet code = "JTabbedPane demo",

width = 400 height = 100 >

step 12 : closing the applet with HTML tag

step 13 : end

Date :	GATE Educational Institutions	Page No. :
11. Develop a program in java to implement the concept of Swings.		Practical No. :
<pre> import javax.swing.*; public class JTabbedPaneDemo extends JApplet { public void init() { JTabbedPane jtp = new JTabbedPane(); jtp.addTab("Cities", new CitiesPanel()); jtp.addTab("Colors", new ColorsPanel()); jtp.addTab("Flavors", new FlavorsPanel()); getContentPane().add(jtp); } } class CitiesPanel extends JPanel { public CitiesPanel() { JButton b1 = new JButton("New York"); add(b1); JButton b2 = new JButton("London"); add(b2); JButton b3 = new JButton("Hong Kong"); add(b3); JButton b4 = new JButton("Tokyo"); add(b4); } } class ColorsPanel extends JPanel { public ColorsPanel() { JCheckBox cb1 = new JCheckBox("Red"); add(cb1); JCheckBox cb2 = new JCheckBox("Green"); add(cb2); JCheckBox cb3 = new JCheckBox("Blue"); add(cb3); } } </pre>		

Date :	GATE Educational Institutions	Page No. :
		Practical No. :

```
class FlavorsPanel extends JPanel {
public FlavorsPanel() {
JComboBox jcb = new JComboBox();
jcb.addItem("Vanilla");
jcb.addItem("Chocolate");
jcb.addItem("Strawberry");
add(jcb);
}
}

/*
<applet code="JTabbedPaneDemo" width=400 height=100>
</applet>
*/
```

Output:



Date :

GATE Educational Institutions

Page No.:

Practical No. :

output :-

Applet

Cities	colors	Flavors
Delhi	Hyderabad	Banglore
		chennai

Applet

Cities	colors	Flavors
	☐ black	☐ white
		☐ purple

Cities	colors	Flavours
		☒ vanilla
		vanilla
		chocolate
		strawberry

conclusion:-

The above java program is executed successfully.

Date :	GATE Educational Institutions	Page No.:
	Passing message to client & server	Practical No. :

Problem Definition :-

Design and develop a java program to passing message to client and server.

Description :-

The socket class is used to communicate client and server. Through this class we can read and write message. The server socket class is used at server-side. the accept() method of server socket class blocks the console until the client is connected.

Algorithm :-

- step 1 : start
- step 2 : import packages
- step 3 : create try and catch blocks
- step 4 : print "total bytes received"
- step 5 : if exception
 - print "error in connection"
- step 6 : create file server
- step 7 : create server socket
- step 8 : print "File transfer begins"
- step 9 : if exception
 - print (e.getMessage());
- step 10 : end

Date :	GATE Educational Institutions	Page No. :
13. Design and develop a java program to passing message to client and server.		Practical No. :
<i>FileClient.java</i> <pre> import java.util.*; import java.io.*; import java.net.*; public class FileClient { public static void main(String a[]) { try { Socket con=new Socket("localhost",95); BufferedReader in=new BufferedReader(new InputStreamReader(System.in)); DataOutputStream toserver=new DataOutputStream(con.getOutputStream()); int size; System.out.print("Enter needed the file name:"); String str=in.readLine(); System.out.print("Enter the destination file name with extension:"); String file=in.readLine(); toserver.writeBytes(str+"\n"); DataInputStream fromserver=new DataInputStream(con.getInputStream()); byte myarr[]=new byte[1024]; int tot=0; FileOutputStream fout=new FileOutputStream(file); while((size=fromserver.read(myarr))>=0) { fout.write(myarr,0,size); tot=tot+size; System.out.println("Total size="+tot); } } } </pre>		

Date :	GATE Educational Institutions	Page No. :
fout.close();		Practical No. :

```

System.out.println("Done ....Total bytes received:"+tot);
fromserver.close();
toserver.close();
con.close();
}
catch(UnknownHostException e)
{
System.out.println("Error in connection");
System.out.println(e.getMessage());
}

```


Date :	GATE Educational Institutions	Page No. :
catch(IOException e) {		Practical No. :


```

System.out.println("Error in file transfer");
System.out.println(e.getMessage());
}
}
}

FileServer.java

import java.util.*;
import java.io.*;
import java.net.*;

public class FileServer
{
    public static void main(String a[])
    {
        try
        {
            ServerSocket s=new ServerSocket(95);
            System.out.println("Server Waiting For The Client");
            Socket cs=s.accept();
            BufferedReader fromclient=new BufferedReader(new
            InputStreamReader(cs.getInputStream()));
            String file=fromclient.readLine(); System.out.println("The file requested:"+file); int len,size;
            FileInputStream fin=new FileInputStream(file);
            len=fin.available();
            System.out.println("File Size="+len);
            byte myarr[]=new byte[512];
            int tot=0;
            cs.getOutputStream().flush();
            OutputStream output = cs.getOutptStream();
            System.out.println("File transfer begins. ... ");
            while((size= fin.read(myarr)) != -1)

```


Date :	GATE Educational Institutions	Page No. :
{		Practical No. :

```

output.write(myarr,0,size);

tot=tot+size;
}

System.out.println("File transfer complete. ... Total Bytes transferred:"+tot);
fin.close();
fromclient.close();
output.close();
cs.close();
}
catch(IOException e)
{
System.out.println(e.getMessage());
}
}
}
}

```

Output:

Date :	GATE Educational Institutions	Page No.:
		Practical No. :

output :-

```
c : \users\HP\Desktop > java c File server.java
c : \users\HP\Desktop > java File server
```

Server waiting for the client

The file requested to: hello phruva!

hello phruva! (The system cannot find the file specified)

```
c : \users\HP\Desktop > java c File client.java
```

```
c : \users\HP\Desktop > java File client
```

Enter needed the file name: Hello phruva!

enter the destination file name with extension:

File server.java

error in file transfer

connection reset

conclusion :-

The above java program is executed successfully.

Verified
G. J. C.