

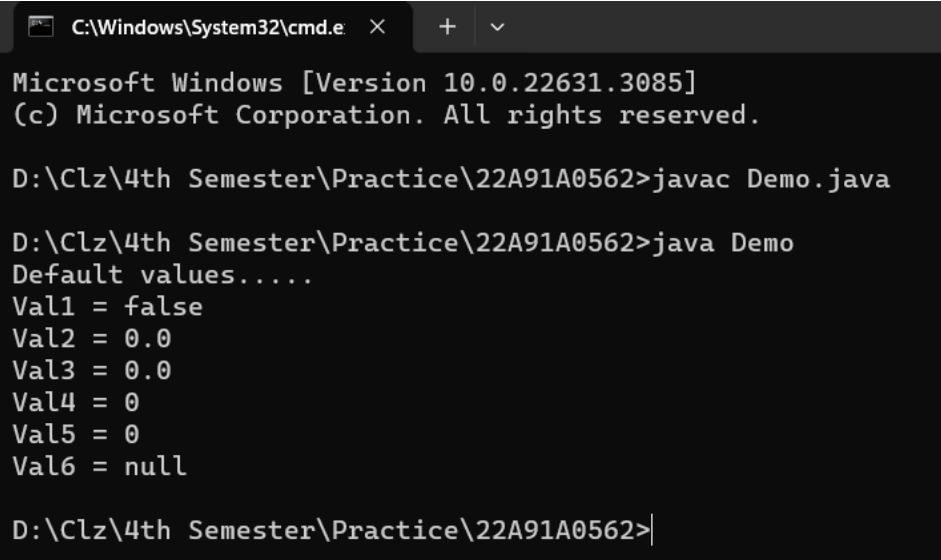
Date:

Week – 1**1. BASIC PROGRAM:**

- 1.1) Write a Java program to display default value of all primitive data type of JAVA.
- 1.2) Write a Java program to find the discriminant value D and find out the roots of the quadratic equation of the form $ax^2+bx+c=0$.
- 1.3) Five Bikers Compete in a race such that they drive at a constant speed which may or may not be the same as the other. To qualify the race, the speed of a racer must be more than the average speed of all 5 racers. Take as input the speed of each racer and print back the speed of qualifying racers.

1.1) Write a Java program to display default value of all primitive data type of JAVA.Program

```
public class Demo {  
    static boolean val1;  
    static double val2;  
    static float val3;  
    static int val4;  
    static long val5;  
    static String val6;  
    public static void main(String[] args)  
    {  
        System.out.println("Default values.....");  
        System.out.println("Val1 = " + val1);  
        System.out.println("Val2 = " + val2);  
        System.out.println("Val3 = " + val3);  
        System.out.println("Val4 = " + val4);  
        System.out.println("Val5 = " + val5);  
        System.out.println("Val6 = " + val6);  
    }  
}
```

Output

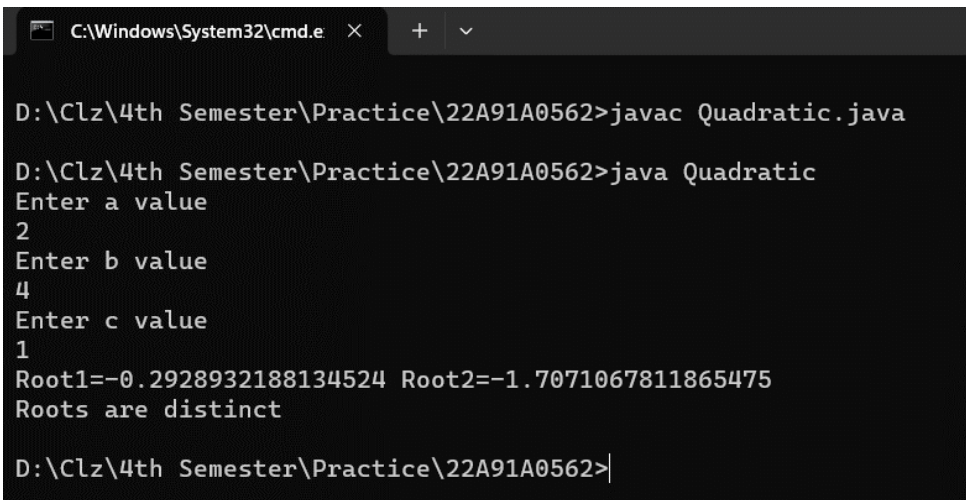
```
C:\Windows\System32\cmd.e  X  +  v  
Microsoft Windows [Version 10.0.22631.3085]  
(c) Microsoft Corporation. All rights reserved.  
  
D:\Clz\4th Semester\Practice\22A91A0562>javac Demo.java  
  
D:\Clz\4th Semester\Practice\22A91A0562>java Demo  
Default values.....  
Val1 = false  
Val2 = 0.0  
Val3 = 0.0  
Val4 = 0  
Val5 = 0  
Val6 = null  
  
D:\Clz\4th Semester\Practice\22A91A0562>|
```

Date:

1.2) Write a Java program to find the discriminant value D and find out the roots of the quadratic equation of the form $ax^2+bx+c=0$.

Program

```
import java.util.*;
class Quadratic
{
public static void main(String args[])
{
double a,b,c,d;
double root1,root2;
Scanner sc=new Scanner(System.in);
System.out.println("Enter a value");
a=sc.nextDouble();
System.out.println("Enter b value");
b=sc.nextDouble();
System.out.println("Enter c value");
c=sc.nextDouble();
d=b*b-4*a*c; if(d>0)
{
root1=(-b+Math.sqrt(d))/(2*a);
root2=(-b-Math.sqrt(d))/(2*a);
System.out.println("Root1="+root1+" Root2="+root2);
System.out.println("Roots are distinct");
}
else if(d==0)
{
root1=root2=(-b)/(2*a);
System.out.println("Root1="+root1+" Root2="+root2);
System.out.println("Roots are equal");
}
else
{
System.out.println("Roots are imaginary");
}
}
}
```

Output

```
C:\Windows\System32\cmd.e  X  +  v

D:\Clz\4th Semester\Practice\22A91A0562>javac Quadratic.java

D:\Clz\4th Semester\Practice\22A91A0562>java Quadratic
Enter a value
2
Enter b value
4
Enter c value
1
Root1=-0.2928932188134524 Root2=-1.7071067811865475
Roots are distinct

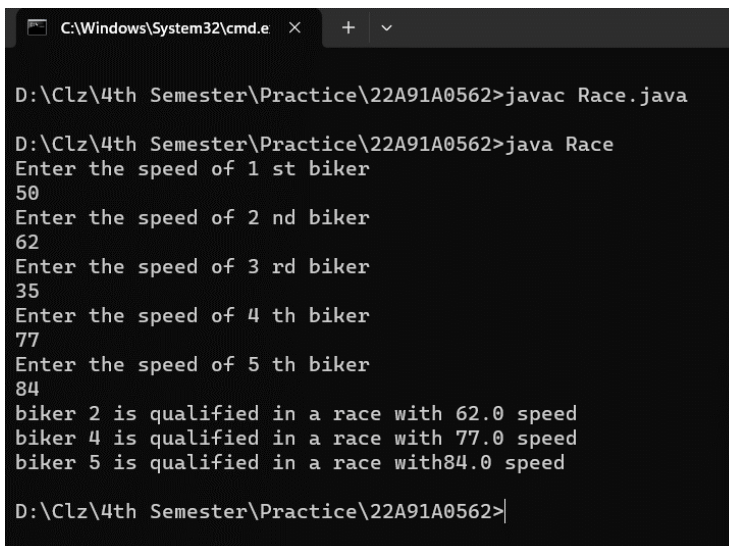
D:\Clz\4th Semester\Practice\22A91A0562>|
```

Date:

1.3) Five Bikers Compete in a race such that they drive at a constant speed which may or may not be the same as the other. To qualify the race, the speed of a racer must be more than the average speed of all 5 racers. Take as input the speed of each racer and print back the speed of qualifying racers.

Program

```
import java.util.*;
class Race
{
    public static void main(String args[])
    {
        double b1,b2,b3,b4,b5,avg;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the speed of 1 st biker");
        b1=sc.nextDouble();
        System.out.println("Enter the speed of 2 nd biker");
        b2=sc.nextDouble();
        System.out.println("Enter the speed of 3 rd biker");
        b3=sc.nextDouble();
        System.out.println("Enter the speed of 4 th biker");
        b4=sc.nextDouble();
        System.out.println("Enter the speed of 5 th");
        b5=sc.nextDouble();
        avg=(b1+b2+b3+b4+b5)/5;
        if(b1>avg)
            System.out.println("biker 1 is qualified in a race with "+b1+" speed");
        if(b2>avg)
            System.out.println("biker 2 is qualified in a race with "+b2+" speed");
        if(b3>avg)
            System.out.println("biker 3 is qualified in a race with "+b3+" speed");
        if(b4>avg)
            System.out.println("biker 4 is qualified in a race with "+b4+" speed");
        if(b5>avg)
            System.out.println("biker 5 is qualified in a race with "+b5+" speed");
    }
}
```

Output

```
C:\Windows\System32\cmd.e  x  +  v

D:\Clz\4th Semester\Practice\22A91A0562>javac Race.java

D:\Clz\4th Semester\Practice\22A91A0562>java Race
Enter the speed of 1 st biker
50
Enter the speed of 2 nd biker
62
Enter the speed of 3 rd biker
35
Enter the speed of 4 th biker
77
Enter the speed of 5 th biker
84
biker 2 is qualified in a race with 62.0 speed
biker 4 is qualified in a race with 77.0 speed
biker 5 is qualified in a race with 84.0 speed

D:\Clz\4th Semester\Practice\22A91A0562>
```

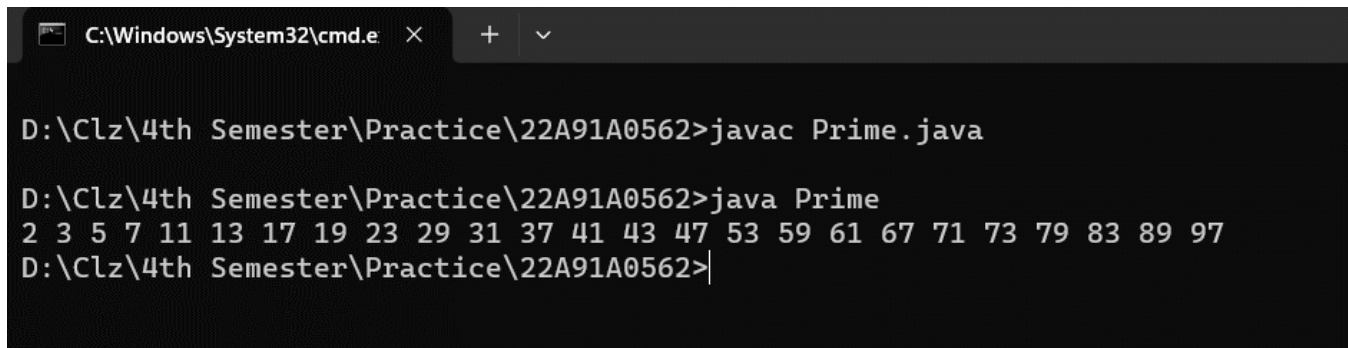
Date:

Week – 2**2. CONTROL FLOW STATEMENTS**

- 2.1) Write a Java program to select all the prime numbers within the range of 1 to 100.
- 2.2) Write a Java program to Find the sum of all even terms in the Fibonacci sequence up to the given range N.
- 2.3) Write a Java program to check whether a given number is Armstrong or not.

2.1) Write a Java program to select all the prime numbers within the range of 1 to 100.**Program**

```
class Prime
{
public static void main(String args[])
{
int i,j;
for(i=1;i<=100;i++)
{
int factor=0; for(j=1;j<=i;j++)
{
if(i%j==0)
factor++;
}
if(factor==2)
System.out.print(i+" ");
}
}
}
```

Output

```
C:\Windows\System32\cmd.e  ×  +  ▾

D:\Clz\4th Semester\Practice\22A91A0562>javac Prime.java

D:\Clz\4th Semester\Practice\22A91A0562>java Prime
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97
D:\Clz\4th Semester\Practice\22A91A0562>
```

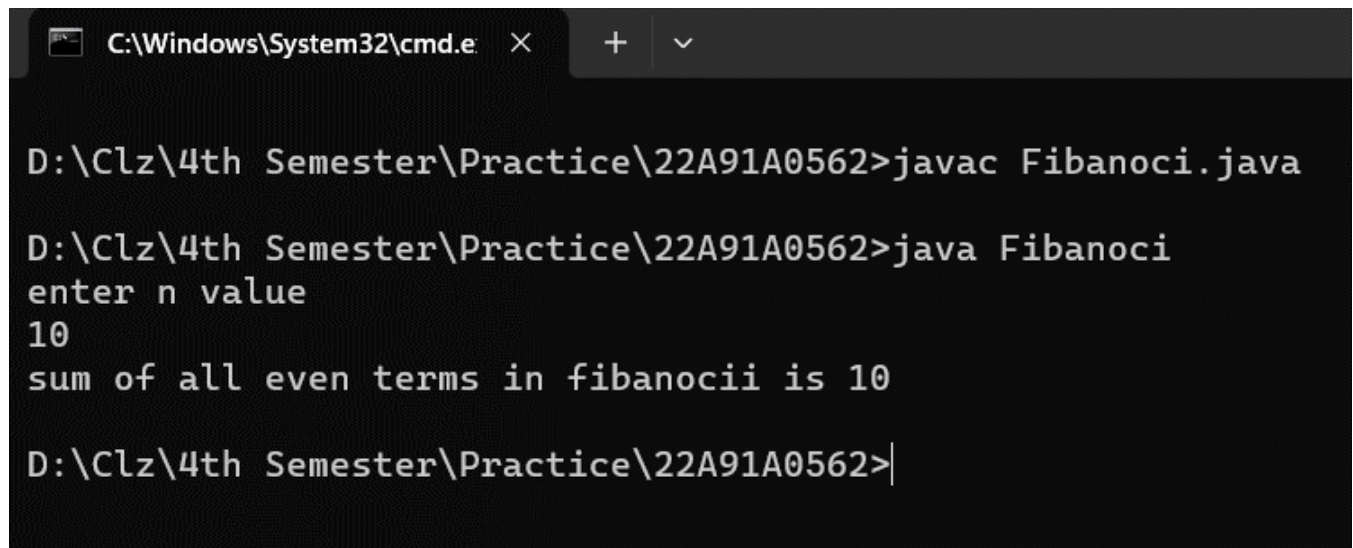
Date:

2.2) Write a Java program to Find the sum of all even terms in the Fibonacci sequence up to the given range N.

Program

```
import java.util.*;
class Fibanoci
{
public static void main(String args[])
{
int n,sum=0,a=0,b=1,c;
Scanner sc=new Scanner(System.in);
System.out.println("enter n value");
n=sc.nextInt();
c=a+b;
while(c<=n)
{
if(c%2==0)
sum=sum+c;
a=b;
b=c; c=a+b;
}
System.out.println("sum of all even terms in fibanocii is "+sum);
}
}
```

Output



```
C:\Windows\System32\cmd.e  X  +  v

D:\Clz\4th Semester\Practice\22A91A0562>javac Fibanoci.java

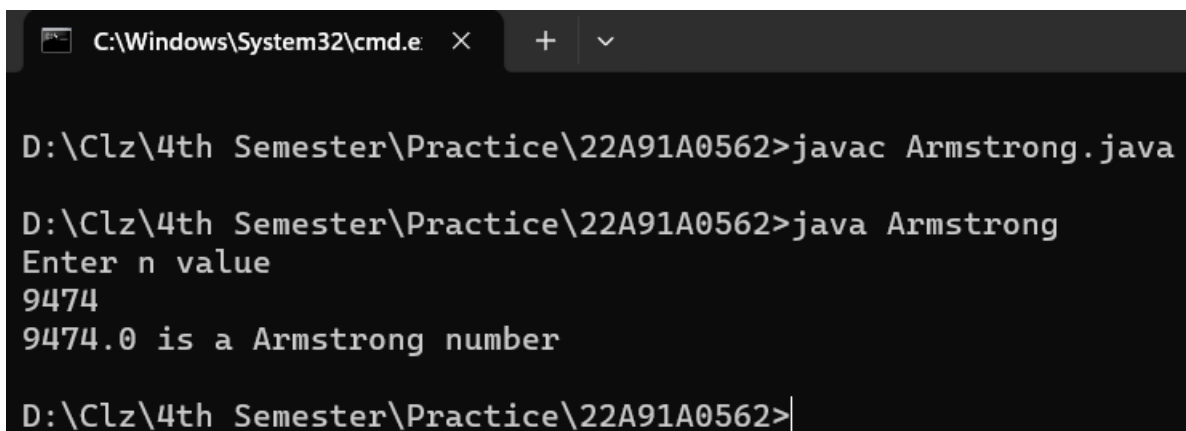
D:\Clz\4th Semester\Practice\22A91A0562>java Fibanoci
enter n value
10
sum of all even terms in fibanocii is 10

D:\Clz\4th Semester\Practice\22A91A0562>|
```

Date:

2.3) Write a Java program to check whether a given number is Armstrong or not.**Program**

```
import java.util.*;
import java.lang.*;
class Armstrong
{
public static void main(String args[])
{
int n,r,temp,count=0,rem;
double sum=0,num;
System.out.println("Enter n value");
Scanner sc=new Scanner(System.in);
n=sc.nextInt();
temp=n;
num=n;
while(temp!=0)
{
rem=temp%10;
count++;
temp=temp/10;
}
while(n!=0)
{
r=n%10;
sum=sum+Math.pow(r,count);
n=n/10;
}
if(sum==num)
System.out.println(num+" is a Armstrong number");
else
System.out.println(num+" is not an Armstrong number");
}
}
```

Output

```
C:\Windows\System32\cmd.e  X  +  v

D:\Clz\4th Semester\Practice\22A91A0562>javac Armstrong.java

D:\Clz\4th Semester\Practice\22A91A0562>java Armstrong
Enter n value
9474
9474.0 is a Armstrong number

D:\Clz\4th Semester\Practice\22A91A0562>|
```

Date:

Week – 3**2. ARRAYS**

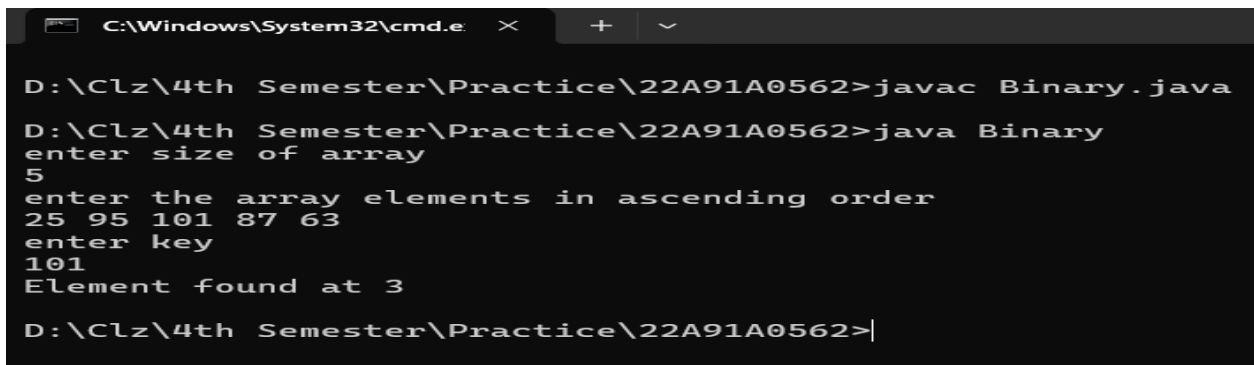
3.1) Write a Java program to implement binary search.

3.2) Write a Java program to sort for an element in a given list of elements using bubble sort.

3.3) Write a Java program to sort for an element in a given list of elements using merge sort.

3.1) Write a Java program to implement binary search.**Program**

```
import java.util.*;
class Binary
{
    public static void main(String args[])
    {
        int a[]=new int[20];
        int n,key,mid,l,h;
        Scanner sc=new Scanner(System.in);
        System.out.println("enter size of array");
        n=sc.nextInt();
        System.out.println("enter the array elements in ascending order");
        for(int i=0;i<n;i++)
            a[i]=sc.nextInt();
        System.out.println("enter key");
        key=sc.nextInt();
        l=0;
        h=n-1;
        while(l<=h)
        {
            mid=(l+h)/2;
            if(key==a[mid])
            {
                System.out.println("Element found at "+(mid+1));
                break;
            }
            else if(key<a[mid])
                h=mid-1;
            else
                l=mid+1;
        }
        if(l>h)
            System.out.println("Element not found");
    }
}
```

Output

```
C:\Windows\System32\cmd.e  X  +  v

D:\Clz\4th Semester\Practice\22A91A0562>javac Binary.java

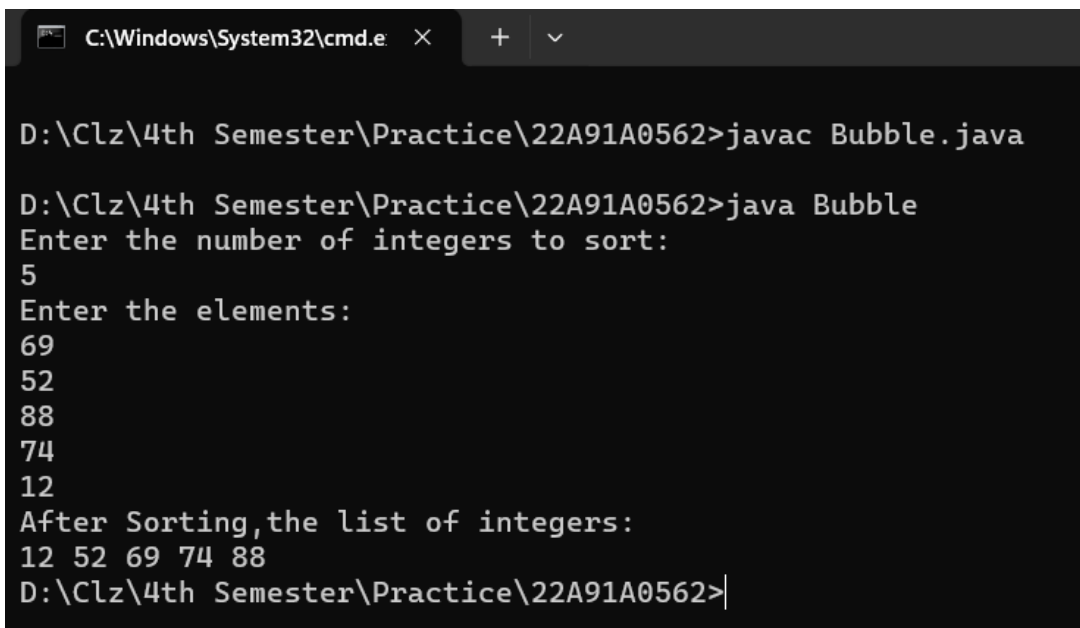
D:\Clz\4th Semester\Practice\22A91A0562>java Binary
enter size of array
5
enter the array elements in ascending order
25 95 101 87 63
enter key
101
Element found at 3

D:\Clz\4th Semester\Practice\22A91A0562>|
```

Date:

3.2) Write a Java program to sort for an element in a given list of elements using bubble sort.Program

```
import java.util.*;
class Bubble
{
    public static void main(String []args)
    {
        int n, i, j, temp;
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number of integers to sort:");
        n=sc.nextInt();
        int a[] = new int[n];
        System.out.println("Enter the elements:");
        for (i = 0; i < n; i++)
            a[i] = sc.nextInt();
        for (i = 0; i < n - 1 ; i++)
        {
            for (j = 0; j < n - i - 1; j++)
            {
                if (a[j] > a[j+1])
                {
                    temp = a[j];
                    a[j] = a[j+1];
                    a[j+1] = temp;
                }
            }
        }
        System.out.println("After Sorting,the list of integers:");
        for (i = 0; i < n; i++)
            System.out.print(a[i]+" ");
    }
}
```

Output

```
C:\Windows\System32\cmd.e  X  +  v

D:\Clz\4th Semester\Practice\22A91A0562>javac Bubble.java

D:\Clz\4th Semester\Practice\22A91A0562>java Bubble
Enter the number of integers to sort:
5
Enter the elements:
69
52
88
74
12
After Sorting,the list of integers:
12 52 69 74 88
D:\Clz\4th Semester\Practice\22A91A0562>
```

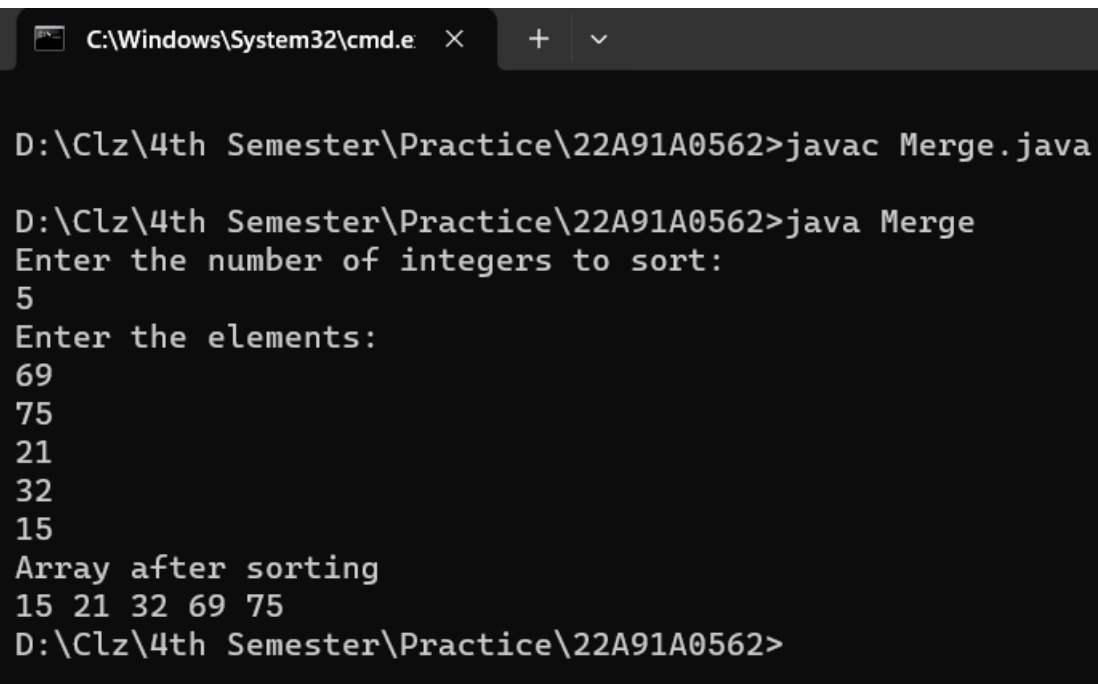

Date:

3.3) Write a Java program to sort for an element in a given list of elements using merge sort.Program

```
import java.util.*;
class Merge
{
    public static void main(String []args)
    {
        int n, i;
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number of integers to sort:");
        n=sc.nextInt();
        int a[] = new int[n];
        System.out.println("Enter the elements:");
        for (i = 0; i < n; i++)
            a[i] = sc.nextInt();
        Method m= new Method();
        m.ms(a,0,n-1);
        System.out.println("Array after sorting");
        for(i=0;i<n;i++)
            System.out.print(a[i]+" ");
    }
}
class Method
{
    void ms(int a[],int first,int last)
    {
        int mid;
        if(first<last)
        {
            mid=(first+last)/2;
            ms(a,first,mid);
            ms(a,mid+1,last);
            merge(a,first,mid,last);
        }
    }
    void merge(int a[],int first,int mid,int last)
    {
        int i,h,j,k;
        int b[]=new int[50];
        h=first;
        i=first;
        j=mid+1;
        while((h<=mid)&&(j<=last))
        {
            if(a[h]<=a[j])
            {
                b[i]=a[h];
                h++;
            }
            else
```

Date:

```
{
    b[i]=a[j];
    j++;
}
i++;
}
if(h>mid)
{
    for(k=j;k<=last;k++)
    {
        b[i]=a[k];
        i++;
    }
}
else
{
    for(k=h;k<=mid;k++)
    {
        b[i]=a[k];
        i++;
    }
}
for(k=first;k<=last;k++)
a[k]=b[k];
}
}
}
```

Output

```
C:\Windows\System32\cmd.e  X  +  v

D:\Clz\4th Semester\Practice\22A91A0562>javac Merge.java

D:\Clz\4th Semester\Practice\22A91A0562>java Merge
Enter the number of integers to sort:
5
Enter the elements:
69
75
21
32
15
Array after sorting
15 21 32 69 75
D:\Clz\4th Semester\Practice\22A91A0562>
```

Date:

Week – 4**2. CLASS MECHANISM**

4.1) Write a Java program to display the details of a person. Personal details should be given in one method and the qualification details in another method.

4.2) Write a Java program to implement constructor and constructor overloading.

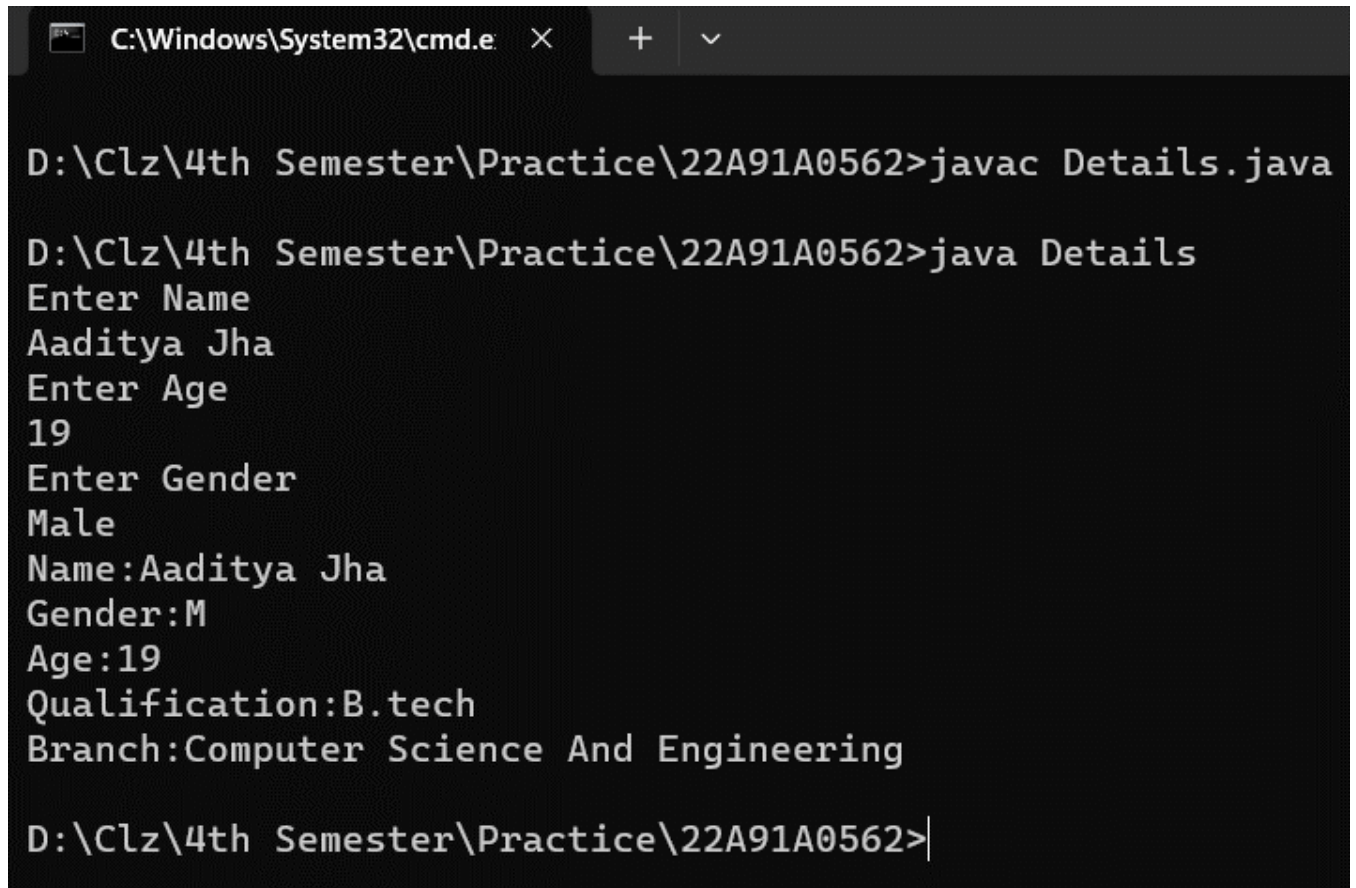
4.3) Write a Java program to implement method overloading.

4.1) Write a Java program to display the details of a person. Personal details should be given in one method and the qualification details in another method.

Program

```
import java.util.*;
class Person
{
char gender;
String name;
int age;
String quali;
String branch;
void personal()
{
Scanner sc=new Scanner(System.in);
System.out.println("Enter Name");
name=sc.nextLine();
System.out.println("Enter Age");
age=sc.nextInt();
System.out.println("Enter Gender");
gender=sc.next().charAt(0);
}
void qualification()
{
quali="B.tech";
branch="Computer Science And Engineering";
}
void display()
{
System.out.println("Name:"+name);
System.out.println("Gender:"+gender);
System.out.println("Age:"+age);
System.out.println("Qualification:"+quali);
System.out.println("Branch:"+branch);
}
}
class Details
{
public static void main(String args[])
{
Person p=new Person();
p.personal();
p.qualification();
p.display();
}}
```

Date:

Output

A screenshot of a Windows command prompt window. The title bar shows the file path 'C:\Windows\System32\cmd.e' with a close button and window controls. The command prompt displays the following sequence of commands and outputs:

```
D:\Clz\4th Semester\Practice\22A91A0562>javac Details.java

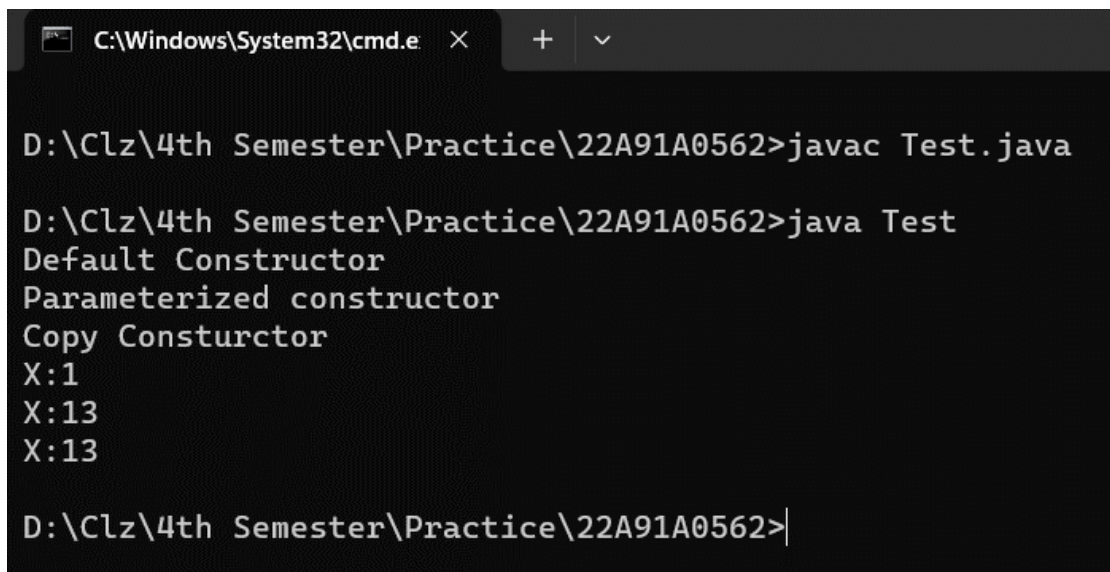
D:\Clz\4th Semester\Practice\22A91A0562>java Details
Enter Name
Aaditya Jha
Enter Age
19
Enter Gender
Male
Name:Aaditya Jha
Gender:M
Age:19
Qualification:B.tech
Branch:Computer Science And Engineering

D:\Clz\4th Semester\Practice\22A91A0562>|
```

Date:

4.2) Write a Java program to implement constructor and constructor overloading.Program

```
class Test
{
    int x;
    public Test() // default constructor
    {
        System.out.println("Default Constructor");
        x = 1;
    }
    public Test(int x) // parameterized constructor
    {
        System.out.println("Parameterized constructor");
        this.x = x;
    }
    public Test(Test t) // copy constructor
    {
        x = t.x;
        System.out.println("Copy Consturctor");
    }
    void show()
    {
        System.out.println("X:"+this.x);
    }
    public static void main (String args [])
    {
        Test t1 = new Test(); // default constructor
        Test t2 = new Test(13); // parameterized
        Test t3 = new Test(t2); // copy constructor
        t1.show (); // 1
        t2.show (); // 13
        t3.show (); // 13
    }
}
```

Output

```
C:\Windows\System32\cmd.e  X  +  v

D:\Clz\4th Semester\Practice\22A91A0562>javac Test.java

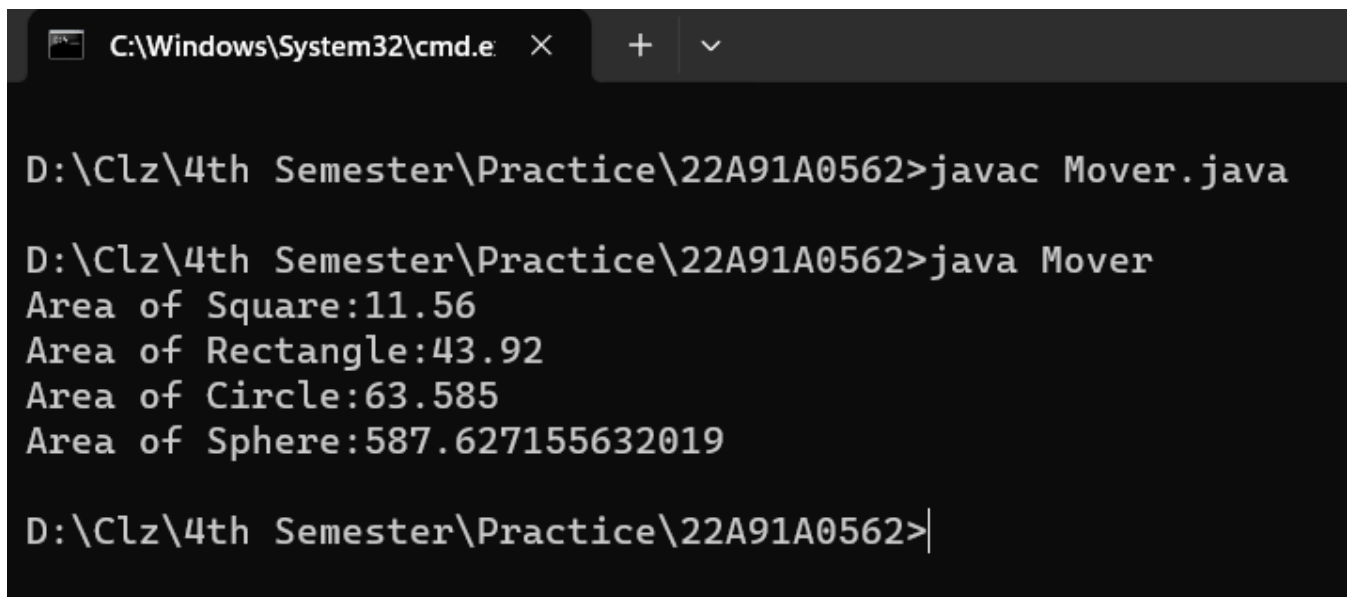
D:\Clz\4th Semester\Practice\22A91A0562>java Test
Default Constructor
Parameterized constructor
Copy Consturctor
X:1
X:13
X:13

D:\Clz\4th Semester\Practice\22A91A0562>
```

Date:

4.3) Write a Java program to implement method overloading.Program

```
class Shapes
{
void area(float side)
{
System.out.println("Area of Square:"+(side*side));
}
void area(float length,float breadth)
{
System.out.println("Area of Rectangle:"+(length*breadth));
}
void area(double radius)
{
System.out.println("Area of Circle:"+(3.14*radius*radius));
}
void area(double radius,float pi)
{
System.out.println("Area of Sphere:"+(4*pi*radius*radius));
}
}
class Mover
{
public static void main(String args[])
{
Shapes s=new Shapes(); s.area(3.4f);
s.area(6.1f,7.2f);
s.area(4.5);
s.area(6.84,3.14f);
}
}
```

Output

```
C:\Windows\System32\cmd.e  ×  +  v

D:\Clz\4th Semester\Practice\22A91A0562>javac Mover.java

D:\Clz\4th Semester\Practice\22A91A0562>java Mover
Area of Square:11.56
Area of Rectangle:43.92
Area of Circle:63.585
Area of Sphere:587.627155632019

D:\Clz\4th Semester\Practice\22A91A0562>|
```