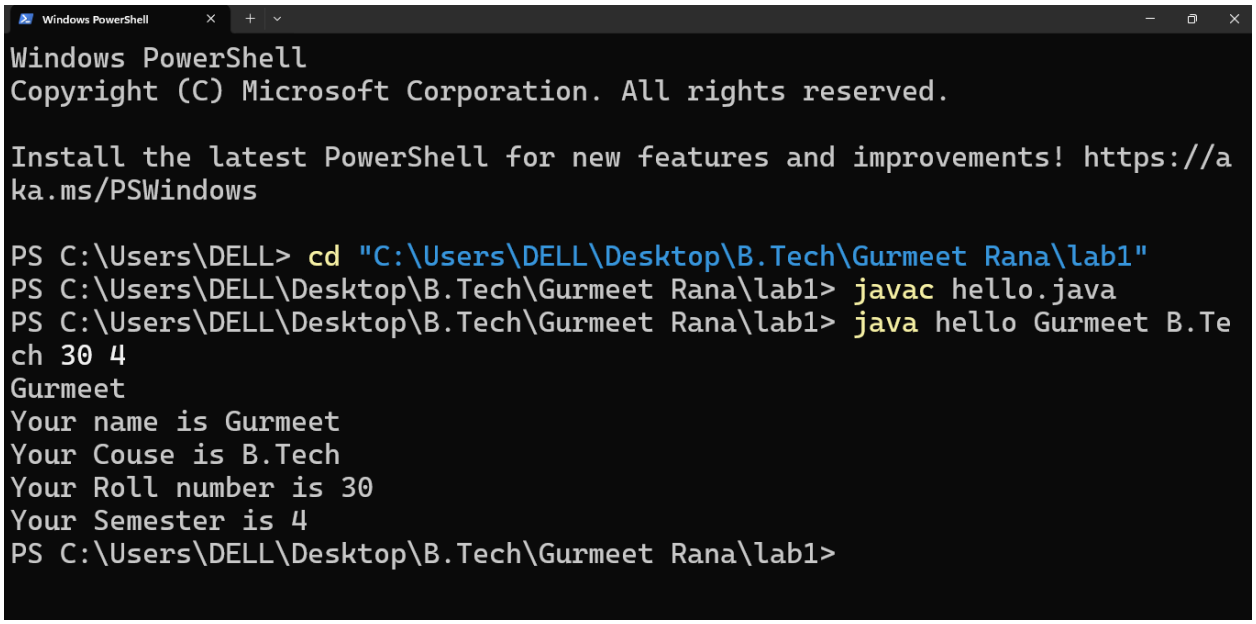
Practical No. 1 :

Write a java program to take input as a command line argument.Your name,course,universityrollno and semester. Display the information.
Name:
UniversityRollNo:
Course:

SOURCE CODE :

```
import java.util.Scanner;
public class hello
{
   public static void main(String[] args)
   {
      String name,course;
      int rollNo,sem;
      Scanner sc=new Scanner(System.in);

      if(args.length<4)
      {
         System.out.println("Enter all the details");
         System.exit(0);
      }

      name = args[0];

      course=args[1];
      sc.nextLine();
      rollNo=Integer.parseInt(args[2]);
      sem=Integer.parseInt(args[3]);

      System.out.println("Your name is "+name);
      System.out.println("Your Couse is "+course);
      System.out.println("Your Roll number is "+rollNo);
      System.out.println("Your Semester is "+sem);
   }
};
```

OUTPUT :


Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows
PS C:\Users\DELL> cd "C:\Users\DELL\Desktop\B.Tech\Gurmeet Rana\lab1"
PS C:\Users\DELL\Desktop\B.Tech\Gurmeet Rana\lab1> javac hello.java
PS C:\Users\DELL\Desktop\B.Tech\Gurmeet Rana\lab1> java hello Gurmeet B.Tech 30 4
Gurmeet
Your name is Gurmeet
Your Couse is B.Tech
Your Roll number is 30
Your Semester is 4
PS C:\Users\DELL\Desktop\B.Tech\Gurmeet Rana\lab1>

Practical No . 2 :

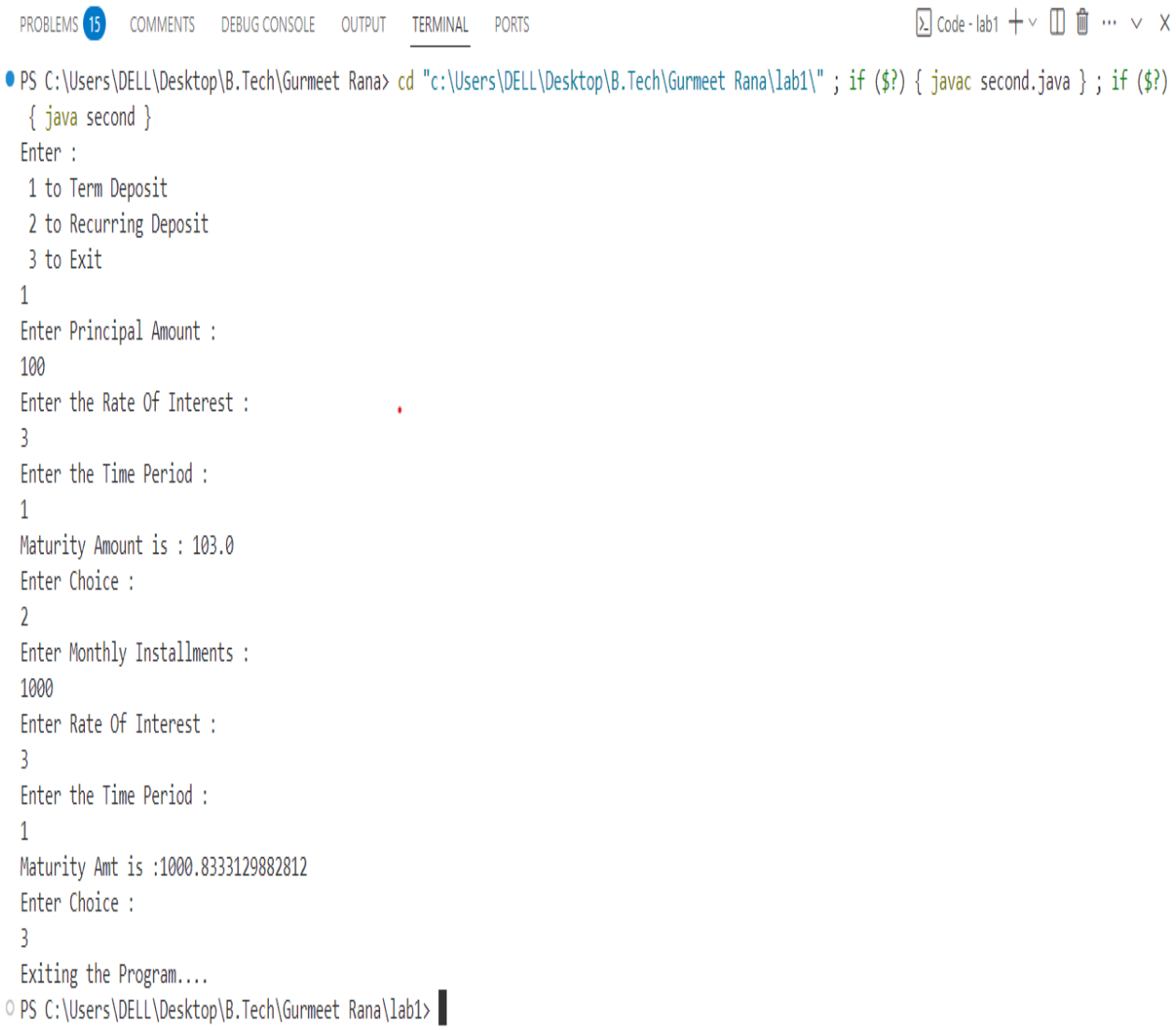
Using the switch statement, write a menu-driven program to calculate the maturity amount of a bank deposit.The user is
(i) Term Deposit
(ii) Recurring Deposit
For option (i) accept Principal (p), rate of interest ® and time period in years (n). Calculate and output the
maturity amount (a) receivable using the formula a =p[1 + r / 100]n.
For option (ii) accept monthly installment (p), rate of interest (r) and time period in months (n). Calculate and
output the maturity amount (a) receivable using the formula a = p * n + p * n(n + 1) / 2 * r / 100 * 1 / 12.
For an incorrect option, an appropriate error message should be displayed.

SOURCE CODE :

```
import java.util.Scanner;
public class hello
{
   public static void main(String[] args)
   {
      String name,course;
      int rollNo,sem;
      Scanner sc=new Scanner(System.in);

      if(args.length<4)
      {
         System.out.println("Enter all the details");
         System.exit(0);
      }

      name = args[0];

      course=args[1];
      sc.nextLine();
      rollNo=Integer.parseInt(args[2]);
      sem=Integer.parseInt(args[3]);

      System.out.println("Your name is "+name);
      System.out.println("Your Couse is "+course);
```

```
        System.out.println("Your Roll number is "+rollNo);
        System.out.println("Your Semester is "+sem);
    }
};
```

OUTPUT :



Practical No. 3 :

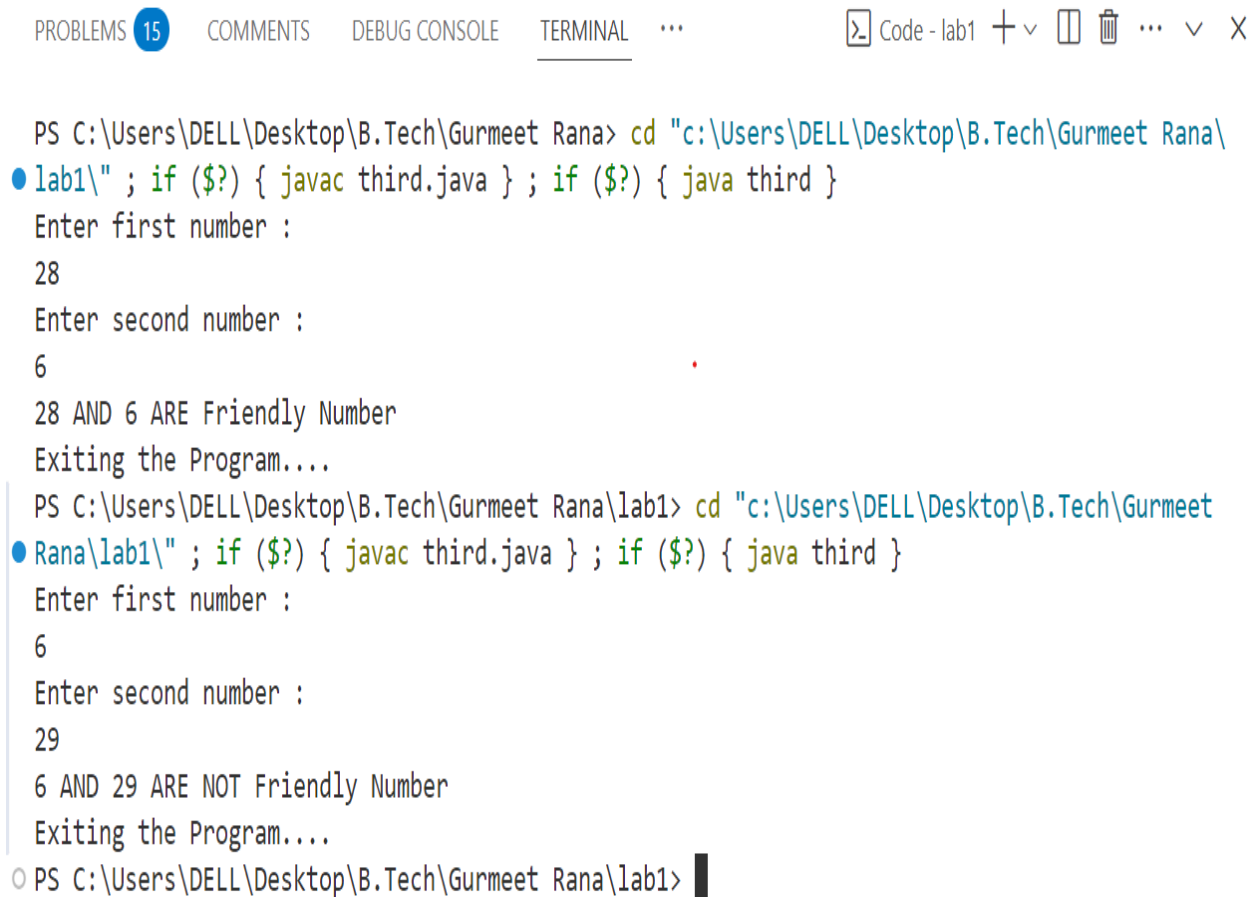
Program to find if the given numbers are Friendly pair or not (Amicable or not).Friendly Pair are two or more numbers with a common abundance.

SOURCE CODE :

```
import java.util.Scanner;
public class third {

   private static int findDivisorSum(int num)
   {
      int sum=0;
      for(int i=1;i<=num/2;i++)
      {
         if(num%i==0)
         {
            sum+=i;
         }
      }
      return sum;
   }
   public static void main(String [] args)
   {
      int num1,num2;
      Scanner sc=new Scanner(System.in);
      System.out.println("Enter first number :");
      num1=sc.nextInt();

      System.out.println("Enter second number :");
      num2=sc.nextInt();

      int divSum1=findDivisorSum(num1);
      int divSum2=findDivisorSum(num2);

      if(divSum1/num1 == divSum2/num2)
      {
         System.out.println(num1+" AND "+num2+" ARE "+"Friendly Number");
      }
      else
      {
```

```
            System.out.println(num1+"AND"+num2+"ARE"+"NOT Friendly Number");
        }
        System.out.println("Exiting the Program....");
}
    }
```

OUTPUT :



Practical No. 4 :

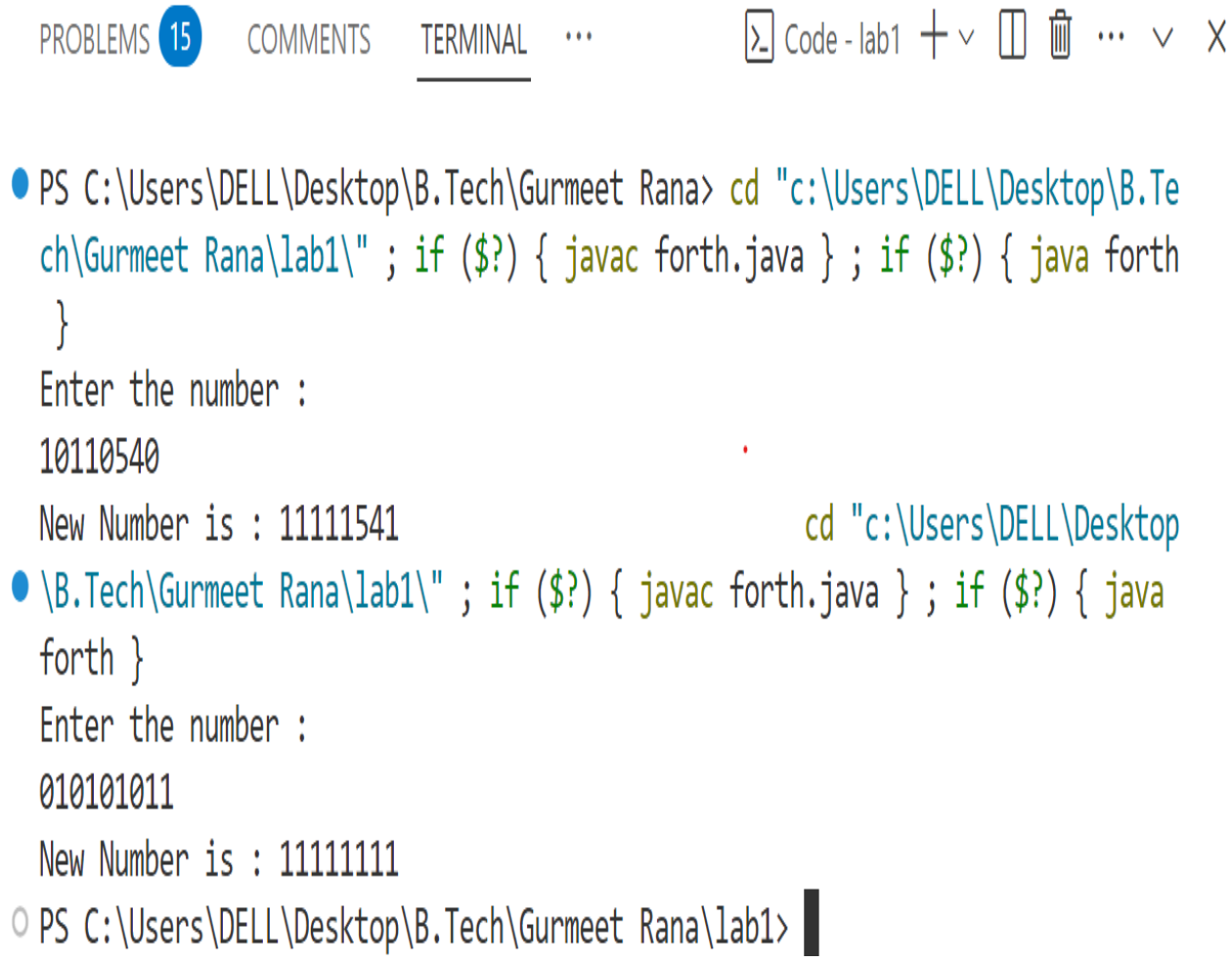
Program to replace all 0's with 1 in a given integer. Given an integer as an input, all the 0's in the number has to be replaced with 1.

SOURCE CODE :

```
import java.util.Scanner;
public class forth {
   public static void main(String [] args)
   {
      int number;
      Scanner sc=new Scanner(System.in);
      System.out.println("Enter the number : ");
      number=sc.nextInt();

      int temp=0;
      while(number>0)
      {
         int digit=number%10;
         if(digit==0)
         {
            temp=temp*10+1;
         }
         else{
            temp=temp*10+digit;
         }
         number/=10;
      }

      int ans=0;
      while(temp>0)
      {
         ans=ans*10+temp%10;
         temp/=10;
      }


      System.out.println("New Number is : "+ans);
```

```
    }
}
```

OUTPUT :


PROBLEMS 15
COMMENTS
TERMINAL
Code - lab1
PS C:\Users\DELL\Desktop\B.Tech\Gurmeet Rana> cd "c:\Users\DELL\Desktop\B.Tech\Gurmeet Rana\lab1\" ; if ($?) { javac forth.java } ; if ($?) { java forth }
Enter the number :
10110540
New Number is : 11111541
cd "c:\Users\DELL\Desktop\B.Tech\Gurmeet Rana\lab1\" ; if ($?) { javac forth.java } ; if ($?) { java forth }
Enter the number :
010101011
New Number is : 11111111
PS C:\Users\DELL\Desktop\B.Tech\Gurmeet Rana\lab1>

Practical No. 5 :

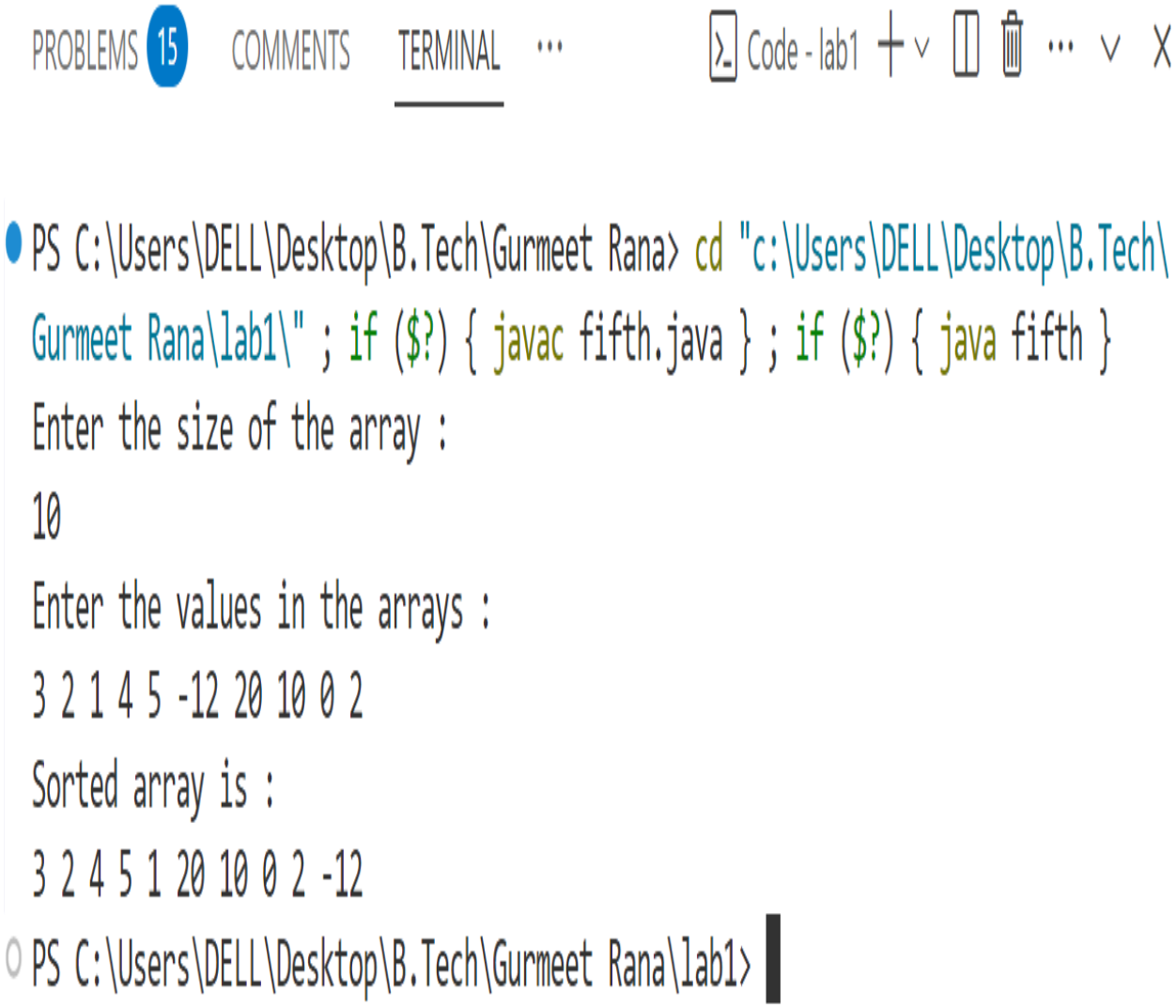
Printing an array into Zigzag fashion. Suppose you were given an array of integers, and you are told to sort the integers in a zigzag pattern. In general, in a zigzag pattern, the first integer is less than the second integer, which is greater than the third integer, which is less than the fourth integer, and so on. Hence, the converted array
should be in the form of e1 < e2 > e3 < e4 > e5 < e6.

SOURCE CODE :

```
import java.util.Scanner;
public class fifth {
   public static void swap(int a,int b)
   {
      a=a^b;
      b=b^a;
      a=a^b;
   }

   void show(int arr[],int n)
   {
      for(int x:arr)
      {
         System.out.println(x);
      }
   }

   void zigZagSort(int arr[],int n)
   {
      boolean flag=false;
      for(int i=0;i<n-1;i++)
      {
         if(flag)
         {
            if(arr[i+1]<arr[i])
            {
               int temp=arr[i+1];
               arr[i+1]=arr[i];
               arr[i]=temp;
            }
         }
         else
```

```
        {
            if(arr[i+1]>arr[i])
            {
                int temp=arr[i+1];
                arr[i+1]=arr[i];
                arr[i]=temp;
            }
        }
    }
  }
  public static void main(String [] args)
  {
    int n;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter the size of the array : ");
    n=sc.nextInt();

    int arr[]=new int[n];

    System.out.println("Enter the values in the arrays : ");
    for(int i=0;i<n;i++)
    {
       arr[i]=sc.nextInt();
    }
    fifth obj=new fifth();
    obj.zigZagSort(arr,n);


    System.out.println("Sorted array is :");
    obj.show(arr, n);

  }
}
```

OUTPUT :


PROBLEMS 15
COMMENTS
TERMINAL
Code - lab1
PS C:\Users\DELL\Desktop\B.Tech\Gurmeet Rana> cd "c:\Users\DELL\Desktop\B.Tech\Gurmeet Rana\lab1\" ; if ($?) { javac fifth.java } ; if ($?) { java fifth }
Enter the size of the array :
10
Enter the values in the arrays :
3 2 1 4 5 -12 20 10 0 2
Sorted array is :
3 2 4 5 1 20 10 0 2 -12
PS C:\Users\DELL\Desktop\B.Tech\Gurmeet Rana\lab1>

Practical No. 6 :

Program to rearrange positive and negative numbers in an array.

SOURCE CODE :

```
import java.util.Scanner;
public class sixth {
   public static void sortNegatives(int arr[],int n)
   {
      int i=0,j=n-1;
      while(i<j)
      {
         if(arr[i]>0 && arr[j]<0)
         {
            int temp=arr[i];
            arr[i]=arr[j];
            arr[j]=temp;
         }
         i++;
         j--;

      }
   }
   public static void main(String [] args)
   {
      Scanner sc=new Scanner(System.in);
      int n;
     System.out.println("Enter the size of the array : ");
      n=sc.nextInt();
      int arr[]=new int[n];
      System.out.println("Enter the values in the array : ");
      for(int i=0;i<n;i++)
      {
         arr[i]=sc.nextInt();
      }
        n=arr.length;
      sortNegatives(arr,n);
```

```
        System.out.println("The sorted array is : ");
        for(int i=0;i<n;i++)
        {
           System.out.print(arr[i]+" ");
        }
        System.out.println("Exiting the program....");
    }
}
```

OUTPUT :