ASSIGNMENT-3

NAME- PRASANTA KUMAR SAHOO
ROLL NO- 404035
REGD NO- 2105105031

Array

1. Write a program to find average and also find total no. of odd, even, prime and palindrome number available in an array of 30 user defined numbers?

```
import java.util.Scanner;

class A1
{
   public static void main(String[] args)
   {
      int even=0,odd=0,prime=0,palindrome=0,sum=0;
      float avg;
      int arr []=new int [30];
      Scanner sc = new Scanner(System.in);
      for(int i=0;i<30;i++)
      {
         System.out.print("Enter value to "+ (i+1) +" Number in array : ");
         arr[i]=sc.nextInt();
      }
      for(int i=0;i<30;i++)
      {
         //For Sum of element
         int n=arr[i];
         sum=sum+n;

         //For Even & Odd
         if(n%2==0)
         {
            even++;
         }
         else
         {
            odd++;
```

```
        }
        //For prime Number
        int count =0;
        for(int j=1;j<=n/2;j++)
        {
           if(n%j==0)
           {
              count++;
           }
        }
        if(count ==1)
        {
           prime++;
        }

        //For pallindrome Number
        int rev=0;
        int num=n;
        while(n!=0)
        {
           rev= (rev*10)+n%10;
           n=n/10;
        }
        if(num==rev)
        {
           palindrome++;
        }
     }
     avg=(float)sum/30;
     System.out.println("Total Number of Even Number is "+even);
     System.out.println("Total Number of Odd Number is "+odd);
     System.out.println("Total Number of Prime Number is "+prime);
     System.out.println("Total Number of Pallindrome Number is "+palindrome);
     System.out.println("Average of all element present in the array is "+avg);
     sc.close();
  }
}
```

2. Write a program to find average of the numbers available in left and right side of the number required by user if available in an array of 10 user defined number otherwise replace the item of the array placed in middle position with the number required by user and calculate output?

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

```
public class A2
{
   public static void main(String[] args)
   {
      Scanner sc = new Scanner(System.in);
      int arr [] = new int[10];
      int num ,position=-1;
      float avg;
      boolean available =false;
      for(int i=0;i<10;i++)
      {
         System.out.print("Enter "+(i+1)+" Element to array : ");
         arr[i]=sc.nextInt();
      }
      System.out.print("Number you want to search : ");
      num=sc.nextInt();
      for(int i=0;i<10;i++)
      {
         if(num == arr[i])
         {
            position=i;
            available =true;
            break;
         }
      }
      if(available)
      {
         avg= (float) (arr[position-1]+arr[position+1]/2);
         System.out.println("Average of left and right side of Number is "+avg);
      }
      else
      {
         arr[5]=num;
         avg=(float) (arr[4]+arr[6])/2;
         System.out.println("Average of left and right side of Middle Number is "+avg);
      }
      sc.close();
   }
}
```

3. Write a program to sort 10 numbers in ascending order and find the sum and product of 2nd smallest and 3rd largest number?

```
import java.util.Scanner;

public class A3
{
   public static void main(String[] args)
   {
      Scanner sc = new Scanner(System.in);
      int arr[]= new int [10];
      int sum,product;
      for(int i=0;i<10;i++)
      {
         System.out.print("Enter value "+(i+1)+" Number : ");
         arr[i]=sc.nextInt();
      }
      for(int i=0;i<10;i++)
      {
         for(int j=i+1;j<10;j++)
         {
            if(arr[i]>arr[j])
            {
               int temp = arr[i];
               arr[i]=arr[j];
               arr[j]=temp;
            }
         }
      }
      System.out.println("Number in the array in ascending order ");
      for(int i=0;i<10;i++)
      {
         System.out.print(" "+arr[i]);
      }
      sum=arr[1]+arr[7];
      product=arr[1]*arr[7];
      System.out.println("\nSum of 2nd smallest and 3rd largest number is "+sum);
      System.out.println("Product of 2nd smallest and 3rd largest number is "+product);
      sc.close();

   }
}
```

4. Write a program to sort 10 numbers in descending order and average of 3rd largest and 4th smallest number in an array of 10 user defined numbers?

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

```
public class A4
{
   public static void main(String[] args)
   {
      Scanner sc = new Scanner(System.in);
      int arr[]= new int [10];
      float avg;
      for(int i=0;i<10;i++)
      {
         System.out.print("Enter value "+(i+1)+" Number : ");
         arr[i]=sc.nextInt();
      }
      for(int i=0;i<10;i++)
      {
         for(int j=i+1;j<10;j++)
         {
            if(arr[i]<arr[j])
            {
               int temp = arr[i];
               arr[i]=arr[j];
               arr[j]=temp;
            }
         }
      }
      System.out.println("Number in the array in descending order ");
      for(int i=0;i<10;i++)
      {
         System.out.print(" "+arr[i]);
      }
      avg = (float) (arr[2]+arr[6])/2;
      System.out.println("\nAverage of 3rd largest and 4th smallest number is "+avg);
      sc.close();
   }
}
```

5. Write a program to find summation, subtraction and multiplication of two 3X4 matrices?

```
import java.util.Scanner;

public class A5
{
   public static void main(String[] args)
   {
      Scanner sc = new Scanner(System.in);
```

```
int a [][]= new int [3][4];
int b [][]= new int [3][4];
int sum [][]= new int [3][4];
int sub [][]= new int [3][4];
int mul [][]= new int [3][4];
System.out.println("Enter element to first matrix : ");
for(int i=0;i<3;i++)
{
   for(int j=0;j<4;j++)
   {
      a[i][j]=sc.nextInt();
   }
}
System.out.println("Enter element to second matrix : ");
for(int i=0;i<3;i++)
{
   for(int j=0;j<4;j++)
   {
      b[i][j]=sc.nextInt();
   }
}
for(int i=0;i<3;i++)
{
   for(int j=0;j<4;j++)
   {
      sum[i][j]=a[i][j]+b[i][j];
      sub[i][j]=a[i][j]-b[i][j];
      for(int k=0;k<4;k++)
      {
         mul[i][j]+=a[i][k]*b[k][j];
      }
   }
}
System.out.println("Summation of two Martix is ");
for(int i=0;i<3;i++)
{
   for(int j=0;j<4;j++)
   {
      System.out.print(" "+sum[i][j]);
   }
   System.out.println();
}
System.out.println("Subtraction of two Martix is ");
for(int i=0;i<3;i++)
```

```
      {
         for(int j=0;j<4;j++)
         {
            System.out.print(" "+sub[i][j]);
         }
         System.out.println();
      }
      System.out.println("Multiplication of two Martix is ");
      for(int i=0;i<3;i++)
      {
         for(int j=0;j<4;j++)
         {
            System.out.print(" "+sub[i][j]);
         }
         System.out.println();
      }

      sc.close();
   }
}
```

6. Write a program to determinant and inverse of a 3X3 matrix?

```
import java.util.Scanner;

public class A6
{
   public static void main(String[] args)
   {
      Scanner sc = new Scanner(System.in);
      int i, j;
      int det = 0;
      int m[][] = new int[3][3];
      System.out.println("Enter elements of matrix row wise:");
      for (i = 0; i < 3; ++i)
      {
         for (j = 0; j < 3; ++j)
         {
            m[i][j] = sc.nextInt();
         }
      }
      //For Determinant
      int x = (m[1][1] * m[2][2]) - (m[2][1] * m[1][2]);
                  int y = (m[1][0] * m[2][2]) - (m[2][0] * m[1][2]);
```

```
                int z = (m[1][0] * m[2][1]) - (m[2][0] * m[1][1]);

                det = (m[0][0] * x)- (m[0][1] * y) + (m[0][2] * z);
      System.out.println("\ndeterminant = " + det);
      if(det!=0)
      {
         System.out.println("Inverse of matrix is:"); // M (inverse) = 1/(det M) * Adj M
         for (i = 0; i < 3; ++i) {
            for (j = 0; j < 3; ++j)
               System.out.print((((m[(j + 1) % 3][(i + 1) % 3] * m[(j + 2) % 3][(i + 2) % 3])- (m[(j + 1) %
3][(i + 2) % 3] * m[(j + 2) % 3][(i + 1) % 3])) / det) + " ");
            System.out.print("\n");
         }
      }
      sc.close();
   }
}

7.  Write a program to find transpose of a 3X4 matrics? */
import java.util.Scanner;

class A7
{
   public static void main(String[] args)
   {
      Scanner sc = new Scanner(System.in);
      int m[][]= new int[3][4];
      for(int i=0;i<3;i++)
      {
         for(int j=0;j<4;j++)
         {
            System.out.print("Enter element to "+ i+1 + " row and "+ j+1 + " column : ");
            m[i][j]=sc.nextInt();
         }
      }
      System.out.println("Transpose of Martix is ");
      for(int i=0;i<4;i++)
      {
         for(int j=0;j<3;j++)
         {
            System.out.print(" "+m[j][i]);
         }
         System.out.println();
      }
```

```
        sc.close();
    }
}
```

CLASS AND OBJECT

1. Write a program to create simple calculator for addition, subtraction, division, multiplication, modulus, factorial, gcd, lcm, power, square root, cube root using class and object?

```
import java.util.Scanner;
class Calulator
{
    int a,b;
    Calulator(int x,int y)
    {
        a=x;
        b=y;
    }
    void add()
    {
        int r =a+b;
        System.out.println("Sum = "+r);
    }
    void sub()
    {
        int r =a-b;
        System.out.println("Sub = "+r);
    }
    void mul()
    {
        int r =a*b;
        System.out.println("mul = "+r);
    }
    void div()
    {
        int r =a/b;
        System.out.println("div = "+r);
    }
    void mod()
    {
        int r =a%b;
        System.out.println("Mod = "+r);
```

```
}
void factorial()
{
   int r=1;
   for(int i=1;i<=a;i++)
   {
      r=r*i;
   }
   System.out.println("Factorial of "+a+" is "+r);
}
void gcd()
{
   int s=(a<b?a:b);
   int g=0;
   for(int i=1;i<=s;i++)
   {
      if(a%i==0 && b%i==0)
      {
         g=i;
      }
   }
   System.out.println("GCD of "+a+" "+b+" is "+g);
}
void lcm()
{
   int l=(a>b?a:b);
   int m=0;
   for(int i=l;i<=a*b;i++)
   {
      if(i%a==0 && i%b==0)
      {
         m=i;
      }
   }
   System.out.println("LCM of "+a+" "+b+" is "+m);
}
void power()
{
   int p=1;
   for(int i=1;i<b;i++)
   {
      p=p*a;
   }
   System.out.println(a+" to the power "+b+" is "+p);
```

```
    }
    void squareRoot()
    {
        int num=a;
        double temp;
        double sr=num/2;
        do{
            temp=sr;
            sr=(temp+(num/temp))/2;
        }while((temp-sr)!=0);
        System.out.println("Square Root of "+a+" is "+sr);
    }
    void cubeRoot()
    {
        /*int num=a;
        double temp;
        double cr=num/3;
        do{
            temp=cr;
            cr=(temp+(num/temp))/3;
        }while((temp-cr)!=0);*/
        double cr = Math.cbrt(a);
        System.out.println("Cube Root of "+a+" is "+cr);
    }
}
class C1
{
    public static void main(String args[])
    {
        Scanner sc= new Scanner(System.in);
        int a,b=0;
        System.out.println("Welcome To Calculator \n1-addition, 2-subtraction, 3-multiplication, 4-
division,5-modulus, 6-gcd, 7-lcm, 8-power, 9-factorial, 10-square root, 11-cube root");
        System.out.print("Choose one operation(1-11):");
        int ch=sc.nextInt();
        if(ch==9 || ch==10 || ch==11) //bcs Factorial , Squre Root and Cube Root need one operand
        {
            System.out.print("\nEnter a Number : ");
            a=sc.nextInt();
        }
        else
        {
            System.out.print("\nEnter First Number : ");
            a=sc.nextInt();
```

```
   System.out.print("\nEnter Second Number : ");
   b=sc.nextInt();
}
Calulator c = new Calulator(a,b);
//Perform operation as per user chooice
if(ch==1)
{
   c.add();
}
else if(ch==2)
{
   c.sub();
}
else if(ch==3)
{
   c.mul();
}
else if(ch==4)
{
   c.div();
}
else if(ch==5)
{
   c.mod();
}
else if(ch==6)
{
   c.gcd();
}
else if(ch==7)
{
   c.lcm();
}
else if(ch==8)
{
   c.power();
}
else if(ch==9)
{
   c.factorial();
}
else if(ch==10)
{
   c.squareRoot();
```

```
    }
    else if(ch==11)
    {
      c.cubeRoot();
    }
    else
    {
      System.out.println("You Choose an invalid option");
    }
    sc.close();
  }
}
```

2. Write a program to give information about any number such as whether it is even odd, prime or not prime, or positive or negative, palindrome or not using class and obect?

```
import java.util.Scanner;
class Number
{
  int num;
  void input()
  {
    Scanner sc = new Scanner(System.in);
    System.out.print("Enter a Number :");
    num=sc.nextInt();
    sc.close();
  }
  void evenOdd()
  {
    if(num%2==0)
    {
      System.out.println("Number is Even");
    }
    else
    {
      System.out.println("Number is Odd");
    }
  }
  void positiveNegative()
  {
    if(num>0)
    {
      System.out.println("Number is Positive");
    }
```

```
    else if(num<0)
    {
        System.out.println("Number is Negative");
    }
    else
    {
        System.out.println("Number is Zero");
    }
}
void primeOrNot()
{
    int count=0;
    for(int i=1;i<=num/2;i++)
    {
        if(num%i==0)
        {
            count++;
        }
    }
    if(count==1)
    {
        System.out.println("Number is Prime");
    }
    else
    {
        System.out.println("Number is not Prime");
    }
}
void pallindrome()
{
    int n=num;
    int rev=0;
    while(n!=0)
    {
        rev=rev*10+(n%10);
        n=n/10;
    }
    if(num==rev)
    {
        System.out.println("Number is Pallindrome");
    }
    else
    {
        System.out.println("Number is Not Pallindrome");
```

```
      }
   }
}
public class C2
{
   public static void main(String[] args)
   {
      Number n1 = new Number();
      n1.input();
      n1.evenOdd();
      n1.positiveNegative();
      n1.primeOrNot();
      n1.pallindrome();
   }
}
```

3. Write a program to find area and perimeter of rectangle using class and object?

```
import java.util.Scanner;
class Rectangle
{
   int ln,br;
   void input()
   {
      Scanner sc = new Scanner(System.in);
      System.out.print("Enter Length and Breadth of Rectangle :");
      ln=sc.nextInt();
      br=sc.nextInt();
      sc.close();
   }
   void area()
   {
      int a=ln*br;
      System.out.println("Area of the Rectangle is "+a);
   }
   void perimeter()
   {
      int p=2*(ln+br);
      System.out.println("Perimeter of the Rectangle is "+p);
   }
}
public class C3
{
```

```
   public static void main(String[] args)
   {
      Rectangle r1 = new Rectangle();
      r1.input();
      r1.area();
      r1.perimeter();
   }
}
```

4. Write a program to find area and perimeter of square using class and object?

```
import java.util.Scanner;
class Square
{
   int s;
   void input()
   {
      Scanner sc = new Scanner(System.in);
      System.out.print("Enter side of Square :");
      s=sc.nextInt();
      sc.close();
   }
   void area()
   {
      int a=s*s;
      System.out.println("Area of the Square is "+a);
   }
   void perimeter()
   {
      int p=4*s;
      System.out.println("Perimeter of the Square is "+p);
   }
}public class C4
{
   public static void main(String[] args)
   {
      Square s1 = new Square();
      s1.input();
      s1.area();
      s1.perimeter();

   }
}
```

5. Write a program to find area and perimeter of triangle having 3 sides using class and object?

```
import java.util.Scanner;
class Triangle
{
   int x,y,z;
   void input()
   {
      Scanner sc = new Scanner(System.in);
      System.out.print("Enter Three side of the Triangle :");
      x=sc.nextInt();
      y=sc.nextInt();
      z=sc.nextInt();
      sc.close();
   }
   void area()
   {
      float s =(float) (x+y+z)/2;
      //S = Semi perimeter
      float a=(s*(s-x)*(s-y)*(s-z));
      System.out.println("Area of the Triangle is "+Math.sqrt(a));
   }
   void perimeter()
   {
      int p=x+y+z;
      System.out.println("Perimeter of the Triangle is "+p);
   }
}public class C5
{
   public static void main(String[] args)
   {
      Triangle T1 = new Triangle();
      T1.input();
      T1.area();
      T1.perimeter();

   }
}
```

6. Write a program to display type of triangle using sides of triangle?

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

```
class Striangle
{
   int a,b,c;
   Striangle(int x,int y,int z)
   {
      a=x;
      b=y;
      c=z;
   }
   void typeOfTriangle()
   {
      if(a==b && b==c)
      {
         System.out.println("Equilateral Triangle");
      }
      else if (a==b || b==c || c==a)
      {
         System.out.println("Isosceles Triangle");
      }
      else
      {
         System.out.println("Scalene Triangle");
      }
   }
}
public class C6
{
   public static void main(String[] args)
   {
      Scanner sc = new Scanner(System.in);
      System.out.print("Enter three side of trianle :");
      int a=sc.nextInt();
      int b=sc.nextInt();
      int c=sc.nextInt();
      Striangle st = new Striangle(a, b, c);
      st.typeOfTriangle();
      sc.close();
   }
}
```

7. Write a java program to create banking application to perform following using using class and object?

a) Initialize with initial balance Rs 3000

b) Deposit amount if balance is more than 1000 otherwise take Rs 100 as penalty for deposit
c) Withdraw amount if balance is more than 1000 otherwise alert user for low balance
d) check for balance

```
class Bank
{
   int bal;
   public Bank (int a)
   {
      bal=a;
   }
   public void deposit(int a)
   {
      if(a>1000)
      {
         bal=bal+a;
         System.out.println("Your account has been creadited with Rs "+a+" and AVl Balance is Rs
"+bal);
      }
      else
      {
         bal=bal+a-100;
         System.out.println("Your account has been creadited with Rs "+a+" , penalty 100 and AVl
Balance is Rs "+bal);
      }
   }
   public void withdraw(int a)
   {
      if (bal>1000)
      {
         bal=bal-a;
         System.out.println("Your account has been debited wit Rs "+a+" and Avl Balance is Rs
"+bal);
      }
      else
      {
         System.out.println("You do not have sufficcient balance in your account");
      }
   }
   public void checkbal()
   {
      System.out.println("Current Balance is Rs "+bal);
   }
}
```

```
class C7
{
   public static void main(String[] args)
   {
      Bank b = new Bank(3000);
      b.deposit(2500);
      b.withdraw(1200);
      b.checkbal();
   }
}
```

8. Write a program to search for a user defined number in array of 20 numbers using linear search and binary search using class and object? Allow user to choose searching method.

```
import java.util.Scanner;
class Searching
{
   int linear(int arr[],int x)
   {
      int index=-1;
      for(int i=0;i<20;i++)
      {
         if(arr[i]==x)
         {
            index = i;
         }
      }
      return index;
   }
   int binary(int arr[],int x)
   {
      //Binary search implemented only sorted arry
      int low=0,high=20,mid,index=-1;
      while(low<=high)
      {
         mid = (low+high)/2;
         if(arr[mid]==x)
           index = mid;
         else if(arr[mid]<x)
           low=mid+1;
         else
           high=mid-1;
      }
      return index;
```

```
   }
}

public class C8
{
   public static void main(String[] args)
   {
      Scanner sc = new Scanner(System.in);
      Searching s = new Searching();
      int arr[]= new int [20];
      int index=-1;
      for(int i=0;i<20;i++)
{
         System.out.print("Enter "+(i+1)+" Element to array : ");
         arr[i]=sc.nextInt();
      }
      //sorting array
      for(int i=0;i<20;i++)
      {
         for(int j=0;j<19;j++)
         {
            if(arr[j]>arr[j+1])
            {
               int temp = arr[j];
               arr[j]=arr[j+1];
               arr[j+1]=temp;
            }
         }
      }
      System.out.print("Enter a number to search in the array : ");
      int n = sc.nextInt();
      System.out.println("Choose Searching Method :\n Linear Search (type =1) \n Binary Search
(type 2)");
      int choose = sc.nextInt();
      if(choose == 1)
      {
         index=s.linear(arr,n);
      }
      else if(choose ==2)
      {
         index= s.binary(arr,n);
      }
      else
      {
```

```
            System.out.println("You choose invalid option for searching ");
        }
        if(index == -1)
        {
            System.out.println(n+" is not present inside array ");
        }
        else
        {
            System.out.println(n+" is present at "+index+" position in the array");
        }
        sc.close();
    }
}
```

9. import java.util.Scanner;
Write a program to implement stack and perform following using class and object?
a) create an stack of size 10 using array
b) insert 10 elements into stack using push and isfull method
c) find factorial of difference between largest and smallest element of stack
d) search any user defined element in stack using peak method
e) delete 3 elements from stack using pop and isempty method
f) display remaining element of stack

```
class Stack
{
    int num[],top,size;
    Stack(int n)
    {
        size =n;
        num = new int [size];
        top= -1;
    }
    boolean isfull()
    {
        if(top==size-1)
            return true;
        else
            return false;
    }
    boolean isempty()
    {
        if(top==-1)
            return true;
        else
```

```
        return false;
}
void push (int a)
{
   boolean st= isfull();
   if(st)
      System.out.println("Overflow");
   else
   {
      top=top+1;
      num[top]=a;
      System.out.println(num[top]+" has been inserted into stack");
   }
}
void pop ()
{
   boolean st =isempty();
   if(st)
     System.out.println("Underflow");
   else
   {
      int t=num[top];
      top =top-1;
      System.out.println("Popped item from stack is "+t);
   }
}
void display()
{
   boolean st = isempty();
   if(st)
   {
      System.out.println("Stack is empty , Nothing to display");
   }
   else
   {
      System.out.println("Element in Stack are ");
      for(int i=0;i<=top;i++)
      {
         System.out.println("Top position = "+i+" and Available data is "+num[i]);
      }
   }
}
void peak(int n)
{
```

```
    boolean Available=false;
    for(int i=0;i<10;i++)
    {
        if(num[i]==n)
        {
            Available =true;
        }
    }
    if(Available)
    {
        System.out.println(n+" is Available in Stack");
    }
    else
    {
        System.out.println(n+" is Not Available in Stack");

    }
}
int factorial(int n)
{
    int fact=1;
    for(int i=2;i<=n;i++)
    {
        fact=fact*i;
    }
    return fact;
}
int largest()
{
    int l=num[0];
    for(int i=1;i<10;i++)
    {
        if(l<num[i])
        {
            l=num[i];
        }
    }
    int fl = factorial(l);
    return fl;
}
int smallest()
{
    int s=num[0];
    for(int i=1;i<10;i++)
```

```
      {
         if(s>num[i])
         {
            s=num[i];
         }
      }
      int fs = factorial(s);
      return fs;
   }
}

class C9
{
   public static void main(String[] args)
   {
      //create an stack of size 10 using array
      Stack s = new Stack(10);
      Scanner sc = new Scanner(System.in);
      //insert 10 elements into stack using push and isfull method
      for(int i=1;i<=10;i++)
      {
         System.out.print("Enter "+i+" Element to stack :");
         int n = sc.nextInt();
         s.push(n);
      }
      //find factorial of difference between largest and smallest
      int diff = s.largest()-s.smallest();
      System.out.println("factorial of difference between largest and smallest element of stack is
"+diff);
      //search any user defined element in stack using peak method
      System.out.print("Enter a number to search in the stack :");
      int se=sc.nextInt();
      s.peak(se);
      //delete 3 elements from stack using pop and isempty method
      s.pop();
      s.pop();
      s.pop();
      // display remaining element of stack
      s.display();
      sc.close();
   }
}
```

10. Write a program to implement queue and perform following using class and object?
a) create an queue of size 10 using array
b) insert 10 elements into queue using insert method
c) replace even numbers available in queue with nearest prime numbers
d) display number of odd and prime numbers.
e) delete 3 elements from queue using delete method
f) display remaining element of queue.

```
import java.util.Scanner;

class Queue
{
   int item[];
   int size;
   int front =-1,rear =-1;
   Queue(int n)
   {
      size =n;
      item= new int[size];
   }
   boolean isFull()
   {
      if(front==0 && rear==size-1)
      {
         return true;
      }
      else
      {
         return false;
      }
   }
   boolean isEmpty()
   {
      if(front==-1)
        return true;
      else
        return false;
   }
   void insert(int n)
   {
      if(isFull())
      {
         System.out.println("Queue is Full");
      }
      else
```

```
    {
        if(front == -1)
        {
            front=0;
        }
        rear++;
        item[rear]=n;
        System.out.println(n+" is Inserted into Queue");
    }
}
void delete()
{
    int element;
    if(isEmpty())
    {
        System.out.println("Queue is empty");
    }
    else
    {
        element=item[front];
        if(front>=rear)
        {
            front =-1;
            rear =-1;
        }
        else
        {
            front++;
        }
        System.out.println(element+" is Remove from the Queue");
    }
}
void display()
{
    if(isEmpty())
    {
        System.out.println("Empty Queue");
    }
    else
    {
        System.out.println("Front index " + front);
        System.out.println("Items in the Queue are  ");
        for(int i=front;i<=rear;i++)
        {
```

```
            System.out.println(item[i]+" ");
        }
        System.out.println(" Rear index "+rear);
    }
}
void peak(int n)
{
    boolean available = false;
    for(int i=0;i<size;i++)
    {
        if(item[i]==n)
        {
            available =true;
        }
    }
    if(available)
    {
        System.out.println(n+" is Avalable in the Queue");
    }
    else
    {
        System.out.println(n+" is Not Avalable in the Queue");

    }

}
void displayPrimeOdd()
{
    if(isEmpty())
    {
        System.out.println("Empty Queue");
    }
    else
    {
        System.out.println("Prime Number in the Queue are ");
        for(int i=0;i<size;i++)
        {
            int count =0;
            for(int j=1;j<=item[i];j++)
            {
                if(item[i]%j==0)
                {
                    count++;
                }
```

```
        }
        if(count==2)
        {
            System.out.print(item[i]+" ");
        }
    }
    System.out.println("Odd Number in the Queue are ");
    for(int i=0;i<size;i++)
    {
        if(item[i]%2!=0)
        {
            System.out.print(item[i]+" ");
        }
    }
  }
}

void replaceEvenNum()
{
  for (int i = 0; i < 10; i++)
  {
    if (item[i] % 2 == 0)
    {
      int l = item[i] + 4;
      for (int j = item[i]; j <= l; j++)
      {
        int count = 0;
        for (int k = 1; k <= j; k++)
        {
          if ( j% k == 0)
          {
            count++;
          }
        }
        if (count == 2)
        {
          item[i]=j;
        }
      }
    }
    else
    {
      int s = item[i] - 4;
      for (int j = item[i]; j >=s; j--)
```

```
            {
               int count = 0;
               for (int k = 1; k <= j; k++)
               {
                  if ( j% k == 0)
                  {
                     count++;
                  }
               }
               if (count == 2)
               {
                  item[i]=j;
               }
            }
         }
      }
   }
}

class C10
{
   public static void main(String[] args)
   {
      Scanner sc = new Scanner(System.in);
      //create an queue of size 10 using array
      Queue q = new Queue(10);
      //insert 10 elements into queue using insert method
      for(int i=1;i<=10;i++)
      {
         System.out.print("Enter "+i+" Element to the Queue :");
         int n=sc.nextInt();
         q.insert(n);
      }
      //replace even numbers available in queue with nearest prime numbers
      q.replaceEvenNum();
      //display number of odd and prime numbers.
      q.displayPrimeOdd();
      //delete 3 elements from queue using delete method
      q.delete();
      q.delete();
      q.delete();
      //display remaining element of queue
      q.display();
      sc.close();
```

```
   }
}
```

11. Write a program to swap two numbers without using 3rd variable and with using 3rd variable using class and object?

```
class Swap
{
   void swapWith(int a,int b)
   {
      System.out.println("Before Swapping with using 3rd variable A is "+a+" and B is "+b);

      int t;
      t=a;
      a=b;
      b=t;
      System.out.println("After Swapping with using 3rd variable A is "+a+" and B is "+b);
   }
   void swapWithout(int a,int b)
   {
      System.out.println("Before Swapping without using 3rd variable A is "+a+" and B is "+b);

      a=a+b;
      b=a-b;
      a=a-b;
      System.out.println("After Swapping without using 3rd variable A is "+a+" and B is "+b);
   }
}
public class C11
{
   public static void main(String[] args)
   {
      Swap s = new Swap();
      s.swapWith(10, 15);
      s.swapWithout(7, 8);
   }
}
```

POLYMORPHISM

1. Write a program to find area and perimeter of circle,square,rectangle and triangle using method overloading?

```
class Shape
{
```

```
   void areaPerimeter(int a,int b, int c,int h)
   {
      int p = a+b+c;
      int ar =(b*h)/2;
      System.out.println("Area of Triangle is "+ar);
      System.out.println("Perimeter of Triangle is "+p);
   }
   void areaPerimeter(int l,int b)
   {
      int p = 2*(l+b);
      int a =l*b;
      System.out.println("Area of Rectangle is "+a);
      System.out.println("Perimeter of Rectangle is "+p);
   }
   void areaPerimeter(int s)
   {
      int p = 4*s;
      int a =s*s;
      System.out.println("Area of Rectangle is "+a);
      System.out.println("Perimeter of Rectangle is "+p);
   }
}
class P1
{
   public static void main(String[] args)
   {
      Shape s = new Shape();
      s.areaPerimeter(6, 7, 12, 10);
      s.areaPerimeter(15, 8);
      s.areaPerimeter(13);
   }
}
```

2. /*Write program to display following pattern using method overloading and constructor overloading?
Odd number based pyramid starts from value n received from user

```
       n+1
     n+3 n+6
   n+5 n+10 n+15
 n+7 n+14 n+21 n+28
```

even number based pyramid starts from value n received from user

```
     n+2
```

```
    n+4 n+6
   n+6 n+10 n+14
 n+8 n+14 n+20 n+26
 user defined character suppose entered character is k
    k
   k k
  k k k
 k k k k
 */
class Pyramid
{
  void pattern(int n)
  {
    if(n%2==0) //For Even Number
    {
      int i=0;
      for(int row=1;row<5;row++)
      {
        for(int space =1;space<5-row;space++)
        {
          System.out.print(" ");
        }
        for(int col=1;col<=row;col++)
        {
          System.out.print(n+((col*i)+2)+" ");
        }
        i=i+2;
        System.out.println();
      }
    }
    else
    {
      int i=1; //For Odd Number
      for(int row=1;row<5;row++)
      {
        for(int space =1;space<5-row;space++)
        {
          System.out.print(" ");
        }
        for(int col=1;col<=row;col++)
        {
          System.out.print(n+(col*i)+" ");
        }
        i=i+2;
```

```
      System.out.println();
    }
   }
  }
  void pattern(char k)
  {
   for(int row=1;row<5;row++)
   {
    for(int space =1;space<5-row;space++)
    {
     System.out.print(" ");
    }
    for(int col=1;col<=row;col++)
    {
     System.out.print(k+" ");
    }
    System.out.println();
   }
  }
}

public class P2
{
   public static void main(String[] args)
   {
    Pyramid p = new Pyramid();
    p.pattern(2);
    p.pattern(3);
    p.pattern('K');
   }
}
```

3. Write a program to sort 10 numbers in the array using following sorting algorithm and method overloading?

a) merge sort

b) quick sort

c) heap sort

```
import java.util.Scanner;

class Sorting
{
   // Merge Sort
   void merge(int a[], int beg, int mid, int end)
```

```
{
    int i, j, k;
    int n1 = mid - beg + 1;
    int n2 = end - mid;

    int LeftArray[] = new int[n1];
    int RightArray[] = new int[n2];

    for (i = 0; i < n1; i++)
    {
        LeftArray[i] = a[beg + i];
    }
    for (j = 0; j < n2; j++)
    {
        RightArray[j] = a[mid + 1 + j];
    }
    i = 0;
    j = 0;
    k = beg;

    while (i < n1 && j < n2)
    {
        if (LeftArray[i] <= RightArray[j])
        {
            a[k] = LeftArray[i];
            i++;
        } else {
            a[k] = RightArray[j];
            j++;
        }
        k++;
    }
    while (i < n1)
    {
        a[k] = LeftArray[i];
        i++;
        k++;
    }

    while (j < n2)
    {
        a[k] = RightArray[j];
        j++;
        k++;
```

```
    }
}

void mergeSort(int a[], int beg, int end)
{
    if (beg < end)
    {
        int mid = (beg + end) / 2;
        mergeSort(a, beg, mid);
        mergeSort(a, mid + 1, end);
        merge(a, beg, mid, end);
    }
}

// Quick Sort
int partition(int a[], int start, int end)
{
    int pivot = a[end];
    int i = (start - 1);

    for (int j = start; j <= end - 1; j++)
    {
        if (a[j] < pivot)
        {
            i++;
            int t = a[i];
            a[i] = a[j];
            a[j] = t;
        }
    }
    int t = a[i + 1];
    a[i + 1] = a[end];
    a[end] = t;
    return (i + 1);
}

/* function to implement quick sort */
void quicksort(int a[], int start, int end)
{
    if (start < end)
    {
        int p = partition(a, start, end);
        quicksort(a, start, p - 1);
        quicksort(a, p + 1, end);
```

```
    }
}

// Heap Sort
void heapify(int a[], int n, int i)
{
    int largest = i;
    int left = 2 * i + 1;
    int right = 2 * i + 2;
    if (left < n && a[left] > a[largest])
        largest = left;
    // If right child is larger than root
    if (right < n && a[right] > a[largest])
        largest = right;
    // If root is not largest
    if (largest != i)
    {
        int temp = a[i];
        a[i] = a[largest];
        a[largest] = temp;

        heapify(a, n, largest);
    }
}

/* Function to implement the heap sort */
void heapSort(int a[], int n)
{
    for (int i = n / 2 - 1; i >= 0; i--)
        heapify(a, n, i);

    for (int i = n - 1; i >= 0; i--)
    {
        int temp = a[0];
        a[0] = a[i];
        a[i] = temp;

        heapify(a, i, 0);
    }
}

/* Function to print the array */
void printArray(int a[])
{
```

```
        System.out.println("After Sorting Element in the array are :");
        for (int i = 0; i < 10; i++)
        {
           System.out.print(a[i] + " ");
        }
     }

}

public class P3
{
   public static void main(String[] args)
   {
        int arr[] = new int[10];
        Scanner sc = new Scanner(System.in);
        Sorting s = new Sorting();
        for (int i = 0; i < 10; i++)
        {
           System.out.print("Enter " + (i + 1) + " Element to array :");
           arr[i] = sc.nextInt();
        }
        System.out.println("1- merge sort 2- quick sort 3- heap sort");
        System.out.print("Choose one sorting : ");
        int choice =sc.nextInt();
        if(choice==1)
        {
           s.mergeSort(arr, 0, 9); //0 is first index , 9 is last indes
           s.printArray(arr);
        }
        else if(choice == 2)
        {
           s.quicksort(arr, 0, 9);
           s.printArray(arr);
        }
        else if(choice == 3)
        {
           s.heapSort(arr, 10);
           s.printArray(arr);
        }
        else
        {
           System.out.println("Sorry , You are entered a wrong option");
        }
        sc.close();
```

```
    }
}
```

4. Write a program to calculate interest that may be simple or compound using method overloading?

```
class Bank
{
    void interest(int p,int t,int r)
    {
        int si = (p*t*r)/100;
        System.out.println("Simple Interest is "+si);
    }
    void interest(int p,int t,int r,int n)
    {   float t2=(float)r/(n*100);
        double t1=(double) Math.pow((1 + t2), t*n);
        double ci =(double) p*t1 - p;
        System.out.println("Compound Interest is "+ci);
    }
}

public class P4
{
    public static void main(String[] args)
    {
        Bank b = new Bank();
        b.interest(1000, 2, 10);
        b.interest(1000, 2, 10,1);
    }
}
```

5. /*Write a program to override sum method if arguments are numbers than it will add numbers or
string than concat two strings using constructor overloading
*/

```
class Add
{
    void sum(int a,int b)
    {
        int c=a+b;
        System.out.println("Sum of two number is : "+c);
    }
    void sum(String x,String y)
    {
        String z=x+y;
```

```
      System.out.println("Concatination of Two String is "+z);
   }
}

public class P5
{
   public static void main(String[] args)
   {
      Add a = new Add();
      a.sum(34, 45);
      a.sum("Prabhu", "Datta");
   }
}
```

6. Write a program to check whether a number or string is palindrome or not using method overloading?

```
class Number
{
   void palindrome(int n)
   {
      int tn=n;
      int rev=0;
      while(tn!=0)
      {
         rev=rev*10+(tn%10);
         tn=tn/10;
      }
      if(rev==n)
      {
         System.out.println(n+" is Palindrome");
      }
      else
      {
         System.out.println(n+" is Not Palindrome");

      }
   }
   void palindrome(String nm)
   {
      String str = nm;
      String rev="";
      int len =str.length();
      for (int i = (len - 1); i >=0; --i)
```

```
      {
         rev = rev + str.charAt(i);
      }

      if(str.toLowerCase().equals(rev.toLowerCase()))
      {
         System.out.println(str + " is a Palindrome String.");
      }
      else
      {
         System.out.println(str + " is not a Palindrome String.");
      }
   }
}
public class P6
{
   public static void main(String[] args)
   {
      Number n = new Number();
      n.palindrome("prabhu");
      n.palindrome(3245423);
   }
}
```

7. Write a program to find volume of rectangle and square box using constructor overloading?

```
class Box
{
   Box(int l,int b,int h)
   {
      int vol=l*b*h;
      System.out.println("Volume of the Rectangle box is "+vol);
   }
   Box(int s)
   {
      int vol=s*s*s;
      System.out.println("Volume of the Square box is "+vol);
   }
}
public class P7
{
   public static void main(String[] args)
   {
```

```
        Box b1 = new Box(5, 8, 7);
        Box b2 = new Box(4);
    }
}
```