CS504D: Object Oriented Programming	Date:

1 Week 1

Problem 1.1 *Write a program in Java to find the prime numbers between 1 to 100*

Code.

```
class prime
{
        static boolean primes[];
        public static void fillFalse()
        {
                int i;
                for(i=0;i<101;i++) {
                        primes[i]=true;
                }
        }
        public static void initialise()
        {
                fillFalse();
                int i,j;
                primes[1]=false;
                for(i=2;i<101;i++) {
                        if(primes[i]==true) {
                                for(j=i+i;j<101;j+=i) {
                                        primes[j]=false;
                                }
                        }
                }
        }
        public static void print()
        {
                int i;
                for(i=1;i<=100;i++) {
                        if(primes[i]==true)
                        System.out.println(i);
                }
        }
        public static void main(String ags[])
        {
                primes=new boolean[101];
                initialise();
                print();
        }
}
```

Problem 1.2 *Write a program in Java to reverse a given number.*

Code.

```
import java.io.*;
class reverse
{
        public static void main(String args[])throws IOException
        {
                BufferedReader br=new BufferedReader(new
                InputStreamReader(System.in));
                int n;
                n=Integer.parseInt(br.readLine());
                int m=n,rev=0;
                while(m>0) {
                        rev=(rev*10)+m%10;
                        m/=10;
                }
                System.out.println("Reversed Number "+rev);
        }
}
```

Problem 1.3 *Write a program in Java to find the sum of digits of a given number.*

Code.

```
import java.io.*;
class sum
{
        public static void main(String args[])throws IOException
        {
                BufferedReader br=new BufferedReader(new
                InputStreamReader(System.in));
                int n;
                n=Integer.parseInt(br.readLine());
                int m=n,rev=0,sum=0;
                while(m>0) {
                        sum+=(m%10);
                        m/=10;
                }
                System.out.println("Sum of each digits "+sum);
        }
}
```

Problem 1.4 *Write a program in Java to print the following pattern.*

```
*
**
***
****
```

Code.

```
class patt1
{
        public static void main(String args[])
        {
```

```
                int n=4,i,j;
                for(i=1;i<=n;i++) {
                        for(j=1;j<=i;j++) {
                                System.out.print("*");
                        }
                        System.out.println();
                }
        }
}
```

Problem 1.5 *Write a program in Java to print the following pattern.*

```
   *
  ***
 *****
*******
```

Code.

```
class patt2
{
        public static void main(String args[])
        {
                int n=4,i,j,k;
                for(i=1;i<=n;i++) {
                        for(j=n-1;j>=i;j--) {
                                System.out.print(" ");
                        }
                        for(j=1;j<=((2*i)-1);j++) {
                                System.out.print("*");
                        }
                        System.out.println();
                }
        }
}
```

Problem 1.6 *Write a program in Java to print the following pattern.*

```
   *
  **
 ***
****
```

Code.

```
class patt3
{
        public static void main(String args[])
        {
                int n=4,i,j;
                for(i=1;i<=n;i++) {
                        for(j=n-1;j>=i;j--) {
                                System.out.print(" ");
                        }
                        for(j=1;j<=i;j++) {
```

```
                                System.out.print("*");
                            }
                            System.out.println();
                        }
                    }
                }
```

2 Week 2 - Function and Constructor Overloading

Problem 2.1 *Write a program in Java to calculate the area of different shapes using function overloading.*

Code.

```
class area1
{
        void area(int sq)
        {
                System.out.println("Area of Square = "+sq*sq);
        }
        void area(int l, int w)
        {
                System.out.println("Area of Rectangle = "+l*w);
        }
        void area(float b,float ht)
        {
                System.out.println("Area of Triangle = "+(0.5)*(b*ht));
        }
        public static void main(String args[])
        {
                area1 a1=new area1();
                a1.area(10);
                a1.area(10,20);
                a1.area(10.0f,25.0f);
        }
}
```

Output.

Problem 2.1 *Write a program in Java to calculate the area of different shapes using Constructor overloading.*

Code.

```
class area2
{
        area2(int sq)
        {
                System.out.println("Area of Square = "+sq*sq);
        }
        area2(int l, int w)
        {
                System.out.println("Area of Rectangle = "+l*w);
        }
        area2(float b,float ht)
        {
                System.out.println("Area of Triangle = "+(0.5)*(b*ht));
```

```
        }
        public static void main(String args[])
        {
                area2 a1=new area2(5);
                a1=new area2(12,20);
                a1=new area2(12.5f,13.0f);
        }
}
```

Output.

3 Week 3 -

Problem 3.1 *Write a program to design a class representing a bank account. The class should have the following data members:*
** a/c no. * customer id * balance amount*
The class should have member methods with the following functions:
** initialize initial value * to deposit amount * to withdraw amount * to display customer id, a/c no. and current balance.*

Code.

```
import java.util.*;
class Bank{
        static Scanner sc=new Scanner(System.in);
        static long acno; static double amt;
        static String id;
        private void init(){
                acno=0; amt=0.0;
                id="";
        }
        private double deposit(double d){ return amt+=d; }
        private double withdraw(double d){
                if(d<amt&&amt!=0)return amt-=d;
                else {
                        System.out.println("Not Enough Balance!!");
                         return amt;
                }
        }
        private void print(){
                System.out.println("Customer ID \t A/c No. \t"+
                "Current Balance");
                System.out.println(id+"\t \t "+acno+"\t \t "+amt);
        }
        public static void main(String[] args){
                Bank obj=new Bank();
                obj.init();
                System.out.println("Enter account no and current balance:");
                id="3000114022";
                acno=sc.nextLong(); amt=sc.nextDouble();
                double d=0.0;
                int choice=0;
                do{
                        System.out.println("Main Menu");
                        System.out.println("0. Deposit");
                        System.out.println("1. Withdrawal");
                        System.out.println("2. Print Statement");
                        System.out.println("3. Exit");
```

```
                        System.out.println("Enter choice:");
                        choice=sc.nextInt();
                        switch(choice){
                                case 0:d=0.0;
                                        System.out.println("Enter "+
                                        "amount to deposit:");
                                        d=sc.nextDouble();
                                         System.out.println("Deposit="+d+
                                         "current balance="+
                                         (double)obj.deposit(d));
                                         break;
                                case 1:d=0.0;
                                       System.out.println("Amount ?");
                                       d=sc.nextDouble();
                                       System.out.println("withdrawal="+d+
                                       "current balance="+
                                       (double)obj.withdraw(d));
                                        break;
                                case 2:obj.print();
                                       break;
                                default:
                                        break;
                        }
                }while(choice<3);
        }
}
```

Output.

Problem 3.2 *Write a program to add two complex numbers.*
Print the result in $x + iy$
form. Use objects as arguments to a method which will perform the addition and use function overloading.

Code.

```
class Complex
{
        double x;
        int y;
        Complex(double a,int b){
                x=a; y=b;
        }
        void print(){ System.out.println(x+"+ i"+y); }
}
class Test{
        double real; int imag;
        private double sum(double a,double b){
                real=a+b; return real;
        }
        private int sum(int a,int b){
                imag=a+b; return imag;
        }
        public static void main(String[] args){
```

```
                Complex obj=new Complex(4,6);
                obj.print();
                Complex obj1=new Complex(1,9);
                obj1.print();
                Test t1=new Test();
                System.out.println("sum = :"+t1.sum(obj.x,obj1.x)+"+" i
                +t1.sum(obj.y,obj1.y));
        }
}
```

Output.

4 Week 4 - Command Line Arguments & static variables

Problem 4.2 *Write a program in Java and create two sub classes "Arts" and "Science", such that all the students have unique roll no*

Code.

```
class Student
{
        static int count;
        int roll;
        Student()
        {
                roll=++count;
        }
}
class Science extends Student
{
        int phy,chem,math;
        Science(int _phy, int _chem, int _math)
        {
                super();
                phy=_phy;
                chem=_chem;
                math=_math;
        }
}
class Art extends Student
{
        int hist,geo,eng;
        Art(int _hist, int _geo, int _eng)
        {
                super();
                hist=_hist;
                geo=_geo;
                eng=_eng;
        }
}
class st
{
        public static void main(String args[])
        {
                Art a1=new Art(10,20,30);
                Science s1=new Science(15,25,31);
                System.out.println("Art\t\t"+a1.hist+
                "\t"+a1.geo+"\t"+a1.eng+"\tRoll "+a1.roll);
                System.out.println("Science\t\t"+s1.phy+"\t"+s1.chem+
```

```
                "\t"+s1.math+"\tRoll "+s1.roll);
        }
}
```

Output.

Problem 4.3 *Write a program in Java to take two integers from the command line and print the largest and smallest among them.*

Code.

```
class op
{
        public static void main(String args[])
        {
                int i;
                int a=Integer.parseInt(args[0]);
                int b=Integer.parseInt(args[1]);
                System.out.println("Sum="+(a+b));
                int max,min;
                if(a>b) {
                        max=a;
                        min=b;
                } else {
                        max=b;
                        min=a;
                }
                System.out.println("Max="+max+"\nMin="+min);

        }
}
```

Output.

Problem 4.4 *Write a program in Java to take command line integers from argument and sort them.*

Code.

```
class sot
{
        public static void main(String args[])
        {
                int i,j;
                int n=args.length;
                int a[]=new int[n];
                for(i=0;i<n;i++) {
                        a[i]=Integer.parseInt(args[i]);
                }
                for(i=0;i<n-1;i++) {
                        for(j=0;j<n-i-1;j++) {
                                if(a[j]>a[j+1]) {
                                        a[j]=(a[j]+a[j+1])-(a[j+1]=a[j]);
                                }
                        }
                }
                for(i=0;i<n;i++) {
```

```
                        System.out.println(a[i]);
                }
        }
}
```

Output.

Problem 4.5 *Write a program in Java to take command line float from argument and sort them.*

Code.

```
class sortF
{
        public static void main(String args[])
        {
                int i,j;
                int n=args.length;
                float a[]=new float[n];
                for(i=0;i<n;i++) {
                        a[i]=Float.parseFloat(args[i]);
                }
                for(i=0;i<n-1;i++) {
                        for(j=0;j<n-i-1;j++) {
                                if(a[j]>a[j+1]) {
                                        a[j]=(a[j]+a[j+1])-
                                        (a[j+1]=a[j]);
                                }
                        }
                }
                for(i=0;i<n;i++) {
                        System.out.println(a[i]);
                }
        }
}
```

Output.

Problem 4.6 *Write a program in Java to take command line strings from argument and sort them.*

Code.

```
class sortAR
{
        public static void main(String args[])
        {
                int i,j;
                for(i=0;i<args.length-1;i++) {
                        for(j=0;j<args.length-i-1;j++) {
                                if(args[j].compareTo(args[j+1])>0) {
                                        String temp=args[j+1];
                                        args[j+1]=args[j];
                                        args[j]=temp;
                                }
                        }
                }
```

```
                    for(i=0;i<args.length;i++) {
                            System.out.println(args[i]);
                    }
            }
}
```

Output.