

Divisors.java

```
public class Divisors {  
    public static void main (String[] args) {  
        int x = Integer.parseInt(args[0]);  
        for (int i = 1 ; i <= x ; i++) {  
            if (x % i == 0) {  
                System.out.println(i);  
            }  
        }  
    }  
}
```

Reverse.java

```
public class Reverse {  
    public static void main (String[] args){  
        String A = args[0];  
        int last = A.length() - 1;  
        for (int i = last ; i >= 0 ; i--){  
  
            System.out.print(A.charAt(i));  
  
        }  
        System.out.print("\nThe middle character is " + (A.charAt(last/2)));  
    }  
}
```

InOrder.java

```
public class InOrder {  
    public static void main (String[] args) {  
        int x = (int) (10 * Math.random());  
        int y = (int) (10 * Math.random());  
        do {  
            System.out.print( x + " ");  
            y = x;  
            x = (int) (10 * Math.random());  
        } while (y<=x);  
    }  
}
```

DamkaBoard.java

```
public class DamkaBoard {  
    public static void main(String[] args) {  
        int x = Integer.parseInt(args[0]);  
        for (int i = 0 ; i < x ; i++) {  
            for (int j = 0 ; j < x ; j++) {  
                if (i % 2 == 1) {  
                    System.out.print(" *");  
                }  
                else {  
                    System.out.print("* ");  
                }  
            }  
            System.out.println();  
        }  
    }  
}
```

Perfect.java

```
public class Perfect {  
    public static void main (String[] args) {  
        int x = Integer.parseInt(args[0]);  
        int sum = 0;  
        String message = x + " is a perfect number since " + x + " = ";  
        for (int i = 1 ; i < x ; i++) {  
            if (x % i == 0) {  
                sum += i;  
                message += i + " + ";  
            }  
        }  
        if (sum != x)  
        {  
            message = x + " is not a perfect number";  
        }  
        else {  
            message = message.substring(0, message.length() - 2);  
        }  
        System.out.println(message);  
    }  
}
```

OneOfEachStats.java

```
import java.util.Random;

public class OneOfEachStats {
    public static void main (String[] args) {
        // Gets the two command-line arguments
        int T = Integer.parseInt(args[0]);
        int seed = Integer.parseInt(args[1]);
        // Initailizes a random numbers generator with the given seed
value
        Random generator = new Random(seed);

        double average = 0;
        int kids = 0;
        int twoKids = 0;
        int threeKids = 0;
        int sumKids = 0;
        int fourOrMoreKids = 0;

        String mostCommon = "The most common number of children is
        ",

        for (int i = 0; i < T; i++) {
            boolean haveBoys = false;
            boolean haveGirls = false;
            int count = 0;
            while (!haveGirls || !haveBoys) {
                double random = generator.nextDouble();
                count++;
                if (random>0.5){
                    haveGirls = true;
                }
                else{
                    haveBoys = true;
                }
            }
        }
    }
}
```

```

        }
    }
    if (count==2){
        twoKids++;
    }
    else if(count==3){
        threeKids++;
    }
    else{
        fourOrMoreKids++;
    }
    sumKids += count;
}

```

```

if (twoKids>threeKids) {
    if (twoKids>fourOrMoreKids) {
        mostCommon = mostCommon + "2";
    }
}
else if (threeKids>fourOrMoreKids) {
    mostCommon = mostCommon + "3";
}
else{
    mostCommon = mostCommon + "4";
}

```

```

average = sumKids/(double)T;

```

```

System.out.println("Average: " + average + " children to get at
least one of each gender.");

```

```

System.out.println("Number of families with 2 children: " +
twoKids);

```

```
        System.out.println("Number of families with 3 children: " +
threeKids);

        System.out.println("Number of families with 4 or more children: "
+ fourOrMoreKids);

        System.out.println(mostCommon + ".");

    }

}
```