```
/**
 *  Gets a command-line argument (int), and prints all the divisors
of the given number.
 */
public class Divisors {
     public static void main (String[] args) {
           int num = Integer.parseInt(args[0]);
           for (int i = 1; i <= num; i++) {
            if (num % i == 0) {
                System.out.println(i);
            }
        }
    }
}
```

```
/**
 * Prints a given string, backward. Then prints the middle character
in the string.
 * The program expects to get one command-line argument: A string.
 */
public class Reverse {
     public static void main (String[] args){
          String str1 = args[0];
          String str2 = "";
          int n = str1.length();

          for (int i = 0; i < n; i++) {
               str2 = str2 + str1.charAt(n - i - 1);
          }

          System.out.println(str2);
          System.out.println("The middle character is " +
str1.charAt((n - 1) / 2));
     }
}
```

```java
import java.util.Random;

/**
 *  Generates and prints random integers in the range [0,10),
 *  as long as they form a non-decreasing sequence.
 */
public class InOrder {
    public static void main(String[] args) {
        int num1 = (int) (Math.random() * 10);
        System.out.print(num1);

        int num2;
        while ((num2 = (int) (Math.random() * 10)) > num1) {
            System.out.print(" " + num2);
            num1 = num2;
        }

        System.out.println();
    }
}
```

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

```
/**
 *  Gets a command-line argument (int), and chekcs if the given
number is perfect.
 */
public class Perfect {
    public static void main(String[] args) {

        int num = Integer.parseInt(args[0]);
        int sum = 0;
        String div = "";

        for (int i = 1; i <= num / 2; i++) {
            if (num % i == 0) {
                sum += i;
                if (!(div.length() == 0)) {
                    div = div + " + ";
                }
                div = div + i;
            }
        }

        if (sum == num) {
            System.out.println(num + " is a perfect number since " +
num + " = " + div);
        } else {
            System.out.println(num + " is not a perfect number");
        }
    }
}
```

```
import java.util.Random;
/**
 *  Computes some statistics about families in which the parents 
decide
 *  to have children until they have at least one child of each 
gender.
 *  The program expects to get two command-line arguments: an int 
value
 *   that determines how many families to simulate, and an int 
value
 *  that serves as the seed of the random numbers generated by the 
program.
 *  Example usage: % java OneOfEachStats 1000 1
 */
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);

           int totalChildren = 0;
        int twoChildren = 0;
        int threeChildren = 0;
        int fourOrMoreChildren = 0;
        int mostCommonNumber = 0;
        int largestCount = 0;

        for (int i = 0; i < T; i++) {
            boolean boy = false;
            boolean girl = false;
            int numberOfChildren = 0;

            while (!boy || !girl) {
                double rand = generator.nextDouble();
                if (rand < 0.5) {
                    boy = true;
                } else {
                    girl = true;
                }

                numberOfChildren++;
            }

            totalChildren += numberOfChildren;

            if (numberOfChildren == 2) {
                twoChildren++;
            } else if (numberOfChildren == 3) {
                threeChildren++;
            } else if (numberOfChildren >= 4) {
                fourOrMoreChildren++;
            }

            if (numberOfChildren > largestCount) {
                largestCount = numberOfChildren;
                mostCommonNumber = numberOfChildren;
            }
        }

        double averageChildren = (double) totalChildren / T;

           if (twoChildren > threeChildren && twoChildren >
fourOrMoreChildren) {
            mostCommonNumber = 2;
        } else if (threeChildren > twoChildren && threeChildren >
fourOrMoreChildren) {
            mostCommonNumber = 3;
        } else if (fourOrMoreChildren > twoChildren &&
fourOrMoreChildren > threeChildren) {
            mostCommonNumber = 4;
        }

        System.out.println("Average: " + averageChildren + "
children to get at least one of each gender.");
        System.out.println("Number of families with 2 children: " +
twoChildren);
        System.out.println("Number of families with 3 children: " +
threeChildren);
        System.out.println("Number of families with 4 or more
children: " + fourOrMoreChildren);
        System.out.println("The most common number of children is "
+ mostCommonNumber + ".");
    }
}
```