

```
/**
 * 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 StringIn = args[0];
        String StringOut = "";
        for (int i = StringIn.length() - 1; i >= 0 ; i--)
        {
            StringOut += StringIn.charAt(i);
        }
        System.out.println(StringOut);
        System.out.println("The middle character is " + StringOut.charAt(StringOut.length() / 2));
    }
}
```

```
/**
 * 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 HighestNum = -1;
        int RandomNum = (int) (Math.random() * 10);
        while (RandomNum >= HighestNum)
        {
            System.out.println(RandomNum);
            HighestNum = RandomNum;
            RandomNum = (int) (Math.random() * 10);
        }
    }
}
```

```

/**
 * Gets a command-line argument n (int), and prints an n-by-n damka board.
 */
public class DamkaBoard {
    public static void main(String[] args) {
        int N = Integer.parseInt(args[0]);
        String Row = "";
        for (int i = 0; i < N; i++)
        {
            Row = "";
            for (int j = 0; j < N; j++)
            {
                if(i % 2 == 0)
                {
                    Row += "* ";
                }
                else
                {
                    Row += " *";
                }
            }
            System.out.println(Row);
        }
    }
}

```

```

/**
 * Gets a command-line argument (int), and checks if the given number is perfect.
 */
public class Perfect {
    public static void main (String[] args) {
        int num = Integer.parseInt(args[0]);
        int DivSum = 0;
        String divisors = "1";
        for (int i = 1; i<num; i++)
        {
            if (num % i == 0)
            {
                DivSum += i;
                if(i != 1)
                {
                    divisors += " + " + i;
                }
            }
        }
        if(num == DivSum && num != 0)
        {
            System.out.println(num + " is a perfect number since " + num + " = " + divisors);
        }
        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);

        //Initialize variables for a single experiment
        Boolean IsBoy = false;
        Boolean IsGirl = false;
        int NumOfChildrenSingleEx = 0;

        //Initialize variables for all experiments
        int TwoChildrenCount = 0;
        int ThreeChildrenCount = 0;
        int FourOrMoreChildrenCount = 0;
        double NumOfChildrenOverall = 0;
        String MostCommonNumOfChildren = "";
    }
}

```

```

for (int i = 0; i < T; i++)
{
    IsBoy = false;
    IsGirl = false;
    NumOfChildrenSingleEx = 0;
    while (!IsBoy || !IsGirl)
    {
        if (generator.nextDouble() < 0.5)
        {
            IsBoy = true;
        }
        else
        {
            IsGirl = true;
        }
        NumOfChildrenSingleEx++;
    }
    if (NumOfChildrenSingleEx == 2) TwoChildrenCount++;
    else if (NumOfChildrenSingleEx == 3) ThreeChildrenCount++;
    else FourOrMoreChildrenCount++;
    NumOfChildrenOverall += NumOfChildrenSingleEx;
}

```

```

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

```

```

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

```

```

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

```

```

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

```

```

        MostCommonNumOfChildren = "4 or more.";

        if (ThreeChildrenCount >= Math.max(TwoChildrenCount, FourOrMoreChildrenCount))
MostCommonNumOfChildren = "3.";

        if (TwoChildrenCount >= Math.max(ThreeChildrenCount, FourOrMoreChildrenCount))
MostCommonNumOfChildren = "2.";

        System.out.println("The most common number of children is " +
MostCommonNumOfChildren);

        //// In the previous version of this program, you used a statement like:
        //// double rnd = Math.random();

        //// Where "rnd" is the variable that stores the generated random value.

        //// In this version of the program, replace this statement with:
        //// double rnd = generator.nextDouble();

        //// This statement will generate a random value in the range [0,1),
        //// just like you had in the previous version, except that the
        //// randomization will be based on the given seed.

        //// This is the only change that you have to do in the program.

    }

}

```