

```
/**
 * 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]);
        if (num == 0) {
            System.out.println("Every non-zero numbers is a divisor
of 0");
        } else {
            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 input = args[0];
```

```
        int len = input.length();
```

```
        for (int i = (len - 1); i >= 0; i--) {
```

```
            char c = input.charAt(i);
```

```
            System.out.print(c);
```

```
        }
```

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

```
        if (len % 2 == 0 ){
```

```
            System.out.println("The middle character is " +  
input.charAt((len / 2) - 1));
```

```
        } else {
```

```
            System.out.println("The middle character is " +  
input.charAt(len / 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 random = (int) (Math.random() * 10);
        int temp = -1;
        do {
            System.out.println(random);
            temp = random;
            random = (int) (Math.random() * 10);
        } while (random >= temp);
    }
}
```

```

/**
 * 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 = 1;
        String perfect = num + " is a perfect number since " + num + " =
        1";
        for(int i = 2; i < num; i++) {
            if(num % i == 0) {
                perfect = (perfect + " + " + i);
                sum = (sum + i);
            }
        }
        if (sum == num) {
            System.out.println(perfect);
        } else {
            System.out.println(num + " is not a perfect number");
        }
    }
}

```

```

/**
 * 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 board = Integer.parseInt(args[0]);
        for (int i = 1; i <= board; i++) {
            if ((i % 2) == 0) {
                for (int j = 1; j <= board; j++ ) {
                    System.out.print(" *");
                }
            } else {
                for (int j = 1; j <= board; j++ ) {
                    System.out.print("* ");
                }
            }
            System.out.println();
        }
    }
}

```

```

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);
        int two_children = 0;
        int three_children = 0;
        int four_or_more = 0;
        int total_sum = 0;
        for (int i = 0; i < T; i++) {
            int count_boy = 0;
            int count_girl = 0;
            do {
                // Generates a random number between 0
                to 1, that determines the gender of the child
                double random = generator.nextDouble();

                boolean girl = (random < 0.5);
                if (girl) {
                    total_sum++;
                    count_girl++;
                } else {
                    total_sum++;
                    count_boy++;
                }
            } while (true);
        }
    }
}

```

```

    }
    } while ((count_girl < 1) || (count_boy < 1));

    // Determines the number of children in each family
    and updates the counts accordingly
    if (count_girl + count_boy == 2) {
        two_children++;
    } else if (count_girl + count_boy == 3) {
        three_children++;
    } else {
        four_or_more++;
    }
}

```

```

// Determines the most common number of children in a
family
String most_common = "";
if ((two_children >= three_children) && (two_children >=
four_or_more)) {
    most_common = "2";
} else if ((three_children > two_children) &&
(three_children >= four_or_more)) {
    most_common = "3";
} else {
    most_common = "4 or more";
}

double average = ((double) (total_sum) / T);

```

```

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

```

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

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

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

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

```
}
```

```
}
```