

```

1  /*****
2  /*プログラム課題: 2-LineCounter
3  /*名前: 三留 慎太郎
4  /*日付: 20250514
5  /*プログラムの説明: java プログラム全体の LOC, クラス毎の LOC, クラス毎のメソッド数を数える。
6  /*クラスの説明: ファイル毎にLOCをカウントする。
7  *****/
8  import java.io.*;
9  public class LineCounter {
10     public static int[] lineCounter(String dirName){
11         String[] classNames = GetFileName.getClassName(dirName);
12         int[] locs = new int[classNames.length];    //各クラスの行数を保持する配列
13         int count = 0;                               //何個目のクラスを処理しているかを管理
14         for(int i = 0; i < classNames.length; i++){ //すべてのクラスを処理したか?
15             File file = new File("./課題" + GetFileName.getProgramNumber(dirName) + "/" + classNames[i]);
16             BufferedReader reader = null;
17             try {
18                 reader = new BufferedReader(new InputStreamReader(new FileInputStream(file), "UTF-8"));
19                 String line;
20                 while((line = reader.readLine()) != null) { //行数分処理
21                     if(line.isEmpty() || line.startsWith("/*")){ //空行もしくはコメント行はカウントしない
22                         ;;
23                     } else {
24                         locs[count]++; //行数をカウント
25                     }
26                 }
27             } catch (IOException e) {
28                 System.out.println("ファイル読み込み中にエラーが発生しました: " + e.getMessage());
29             }
30             count++;
31         }
32         return locs;
33     }
34     public static int[] methodCounter(String dirName){
35         String[] classNames = GetFileName.getClassName(dirName);
36         int[] methodCount = new int[classNames.length]; //各クラスのメソッド数を保持する配列
37         int count = 0;                               //何個目のクラスをほっじしているかを管理
38         for(int i = 0; i < classNames.length; i++){ //すべてのクラスを処理したか?
39             File file = new File("./課題" + GetFileName.getProgramNumber(dirName) + "/" + classNames[i]);
40             BufferedReader reader = null;
41             try {
42                 reader = new BufferedReader(new InputStreamReader(new FileInputStream(file), "UTF-8"));
43                 String line;
44                 while((line = reader.readLine()) != null) { //行数分処理
45                     if(line.startsWith("    ")){
46                         if(line.startsWith("    ") || line.startsWith("    }")){ //以下6行で4字分のインデントがあり、5文字目が空白もしくは
47                             は)ではないものをメソッドとして数える。
48                             ;;
49                         } else if(line.endsWith(";")) {
50                             ;;
51                         } else {
52                             methodCount[i]++; //メソッド数をカウント
53                             System.out.println(gyou);
54                         }
55                     }
56                 } catch (IOException e) {
57                     System.out.println("ファイル読み込み中にエラーが発生しました: " + e.getMessage());
58                 }
59             }
60             return methodCount;
61         }
62     }

```