

```

1  /*
2  Name : Vasu Kalariya
3  Roll : PE29
4  Sub  : SSC ( Macro Pass 1)
5  */
6
7  import java.io.*;
8  import java.util.Hashtable;
9
10 public class MacroPass1 {
11     public static void main(String[] args) throws
12     IOException {
13         BufferedReader br = null;
14         FileReader fr = null;
15         Hashtable<String, String> ala = new Hashtable
16         <String, String>();
17         boolean macrodef = false;           //
18         //flag for macro definition
19         boolean alaParameter = false;       //
20         //flag for macro inserting ala parameters name
21         boolean endMacro = false;           //
22         //flag for end of macro definition
23         boolean macroName = false;          //
24         //flag for name of macro definition
25
26         int mdtptr = 1, mntptr = 1, alaptr = 1;
27         // pointers for MDT, MNT, ALA
28
29         String inputfilename = "F:\\T9\\SSC\\Assi 3\\
30 src\\Input.txt"; //Input File
31         fr = new FileReader(inputfilename);
32         br = new BufferedReader(fr);
33
34         String f1 = "F:\\T9\\SSC\\Assi 3\\src\\MDT.
35 txt"; // MDT file
36         FileWriter fw1 = new FileWriter(f1);
37         BufferedWriter bw1 = new BufferedWriter(fw1);
38
39         String f2 = "F:\\T9\\SSC\\Assi 3\\src\\MNT.
40 txt"; // MNT file
41         FileWriter fw2 = new FileWriter(f2);
42         BufferedWriter bw2 = new BufferedWriter(fw2);

```

```

35
36     String f3 = "F:\\T9\\SSC\\Assi 3\\src\\ALA.
    txt";           // ALA file
37     FileWriter fw3 = new FileWriter(f3);
38     BufferedWriter bw3 = new BufferedWriter(fw3);
39
40     String f4 = "F:\\T9\\SSC\\Assi 3\\src\\Output
    .txt";           // OUTPUT file without macro
    expansion
41     FileWriter fw4 = new FileWriter(f4);
42     BufferedWriter bw4 = new BufferedWriter(fw4);
43
44     try {
45
46
47         String sCurrentLine, s0, s1, s2;
48         while ((sCurrentLine = br.readLine()) !=
    null) {
49             s0 = sCurrentLine.split(" |\\,")[0];
50
51             if (s0.equals("MACRO"
    )) {           // check for Macro definition
52                 macrodef = true;
53                 alaParameter = true;
54                 endMacro = true;
55             }
56             else if (macrodef == true ||
    alaParameter == true || endMacro == true) {
57
58                 if (macrodef == true
    ) {           // writing in MNT
59                     bw2.write(mntptr + "\\t" + s0
    + "\\t" + mdtptr + "\\n");
60                     macrodef = false;
61                     macroName = true;
62                     mntptr++;
63                 }
64                 if (alaParameter == true
    ) {           // writing in ALA
65                     for (int i = 1; i < (
    sCurrentLine.split(" |\\,").length); i++) {
66                         s1 = sCurrentLine.split(
    " |\\,")[i];
67                         String temp;

```

```

68         bw3.write("#" + alaptr
    + "\t" + s1 + "\n");
69         temp = "#" + alaptr;
70         ala.put(s1, temp);
71         alaptr++;
72     }
73     alaParameter = false;
74 }
75     if (endMacro == true
){
    // writing in MDT
76         bw1.write(mdtptr+" ");
77         for (int i = 0; i < (
sCurrentLine.split(" |\\,").length); i++){
78
79             s2 = sCurrentLine.split(
" |\\,")[i];
80
81             if (ala.containsKey(s2
) && macroName == false){
    // replacing
Parameters with # number
82                 bw1.write(ala.get(s2
)+" ");
83             }
84             else {
85                 bw1.write(s2+" ");
86             }
87         }
88         bw1.write("\n");
89         mdtptr++;
90         macroName = false;
91     }
92 }
93
94     else {
95         bw4.write(sCurrentLine+"\n");
96     }
97
98     if (s0.equals("MEND")){
99         endMacro = false;
100    }
101 }
102
103     } catch (FileNotFoundException
fileNotFoundException) {

```

```
104         fileNotFoundException.printStackTrace();
105     } catch (IOException ioException) {
106         ioException.printStackTrace();
107     }
108
109     bw1.close();
110     bw2.close();
111     bw3.close();
112     bw4.close();
113
114 }
115 }
116
```