import java.io.\*;

import java.util.\*;

class intWrapper {

public int value;

public intWrapper(int value) {

this.value = value;

}

}

class MDTEntry {

String line;

public MDTEntry(String line) {

this.line = line;

}

@Override

public String toString() {

return line;

}

}

class tableentry {

public int index;

public String name;

public int mdtindex;

public tableentry(int index, String name, int mdtindex) {

this.index = index;

this.name = name;

this.mdtindex = mdtindex;

}

@Override

public String toString() {

return index + "\t" + name + "\t" + mdtindex;

}

}

class Pass1 {

public static void main(String[] args) {

intWrapper mntc = new intWrapper(1);

List<tableentry> MNT = new ArrayList<>();

List<MDTEntry> MDT = new ArrayList<>();

List<String> ALA = new ArrayList<>();

boolean inMacro = false;

int mdtIndex = 0;

try (

BufferedWriter macroIntermediateWriter = new BufferedWriter(new FileWriter("macro\_intermediate.txt"));

BufferedWriter mdtWriter = new BufferedWriter(new FileWriter("MDT.txt"));

BufferedWriter mntWriter = new BufferedWriter(new FileWriter("MNT.txt"));

BufferedWriter alaWriter = new BufferedWriter(new FileWriter("ALA.txt"));

) {

File file = new File("macro\_source.txt");

Scanner scanner = new Scanner(file);

while (scanner.hasNextLine()) {

String line = scanner.nextLine().trim();

if (line.isEmpty()) continue;

String[] tokens = line.split("[,\\s]+");

if (tokens[0].equalsIgnoreCase("MACRO")) {

inMacro = true;

String newline = scanner.nextLine().trim();

String[] newtokens = newline.split("[,\\s]+");

for (int i = newtokens.length - 1; i >= 0; i--) {

if (newtokens[i].startsWith("&")) {

ALA.add(mdtIndex, newtokens[i]);

} else {

MNT.add(new tableentry(mntc.value, newtokens[i], mdtIndex + 1));

mntc.value++;

}

}

MDT.add(new MDTEntry(newline));

continue;

}

if (inMacro && !tokens[0].equalsIgnoreCase("MEND")) {

for (int i = 0; i < tokens.length; i++) {

if (tokens[i].startsWith("&")) {

int alaIndex = ALA.indexOf(tokens[i]) + 1;

tokens[i] = "#" + alaIndex;

}

}

MDT.add(new MDTEntry(String.join(" ", tokens)));

mdtIndex++;

}

if (!inMacro) {

macroIntermediateWriter.write(line);

macroIntermediateWriter.write("\n");

}

if (tokens[0].equalsIgnoreCase("MEND")) {

MDT.add(new MDTEntry(line));

inMacro = false;

mdtIndex++;

}

}

scanner.close();

for (tableentry entry : MNT) {

mntWriter.write(entry.toString());

mntWriter.newLine();

}

for (int i = 0; i < MDT.size(); i++) {

mdtWriter.write(i + 1 + "\t" + MDT.get(i).toString());

mdtWriter.newLine();

}

for (int i = 0; i < ALA.size(); i++) {

alaWriter.write(i + 1 + "\t" + ALA.get(i));

alaWriter.newLine();

}

} catch (FileNotFoundException e) {

System.err.println("Error: File not found.");

e.printStackTrace();

} catch (IOException e) {

e.printStackTrace();

}

System.out.println("Macro Name Table (MNT):");

System.out.println("Index\tName\tMDT Index");

for (tableentry entry : MNT) {

System.out.println(entry);

}

System.out.println("\nMacro Definition Table (MDT):");

System.out.println("Index\tLine");

for (int i = 0; i < MDT.size(); i++) {

System.out.println(i + 1 + "\t" + MDT.get(i));

}

System.out.println("\nArgument List Array (ALA):");

for (int i = 0; i < ALA.size(); i++) {

System.out.println(i + 1 + "\t" + ALA.get(i));

}

}

}