import java.io.\*;

import java.util.\*;

public class MacroPass1 {

    public static void main(String[] args) throws IOException

    {

        BufferedReader br = new BufferedReader(new FileReader("macro\_input.asm"));

        FileWriter mnt = new FileWriter("mnt.txt");

        FileWriter mdt = new FileWriter("mdt.txt");

        FileWriter kpdt = new FileWriter("kpdtab.txt");

        FileWriter pnt = new FileWriter("pntab.txt");

        LinkedHashMap<String, Integer> pntab = new LinkedHashMap<>();

        String line;

        String Macroname = null;

        int mdtp = 1, kpdtp = 0, paramNo = 1, pp = 0, kp = 0, flag = 0;

        mnt.write("Name PP KP MDTP KPDTP\n");

        while ((line = br.readLine()) != null) 6

        {

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

            if (parts[0].equalsIgnoreCase("MACRO"))

            {

                flag = 1;

                line = br.readLine();

                parts = line.split("\\s+");

                Macroname = parts[0];

                if (parts.length <= 1)

                {

                    mnt.write(parts[0] + "\t" + pp + "\t" + kp + "\t" + mdtp + "\t" + (kp == 0 ? kpdtp : (kpdtp + 1)) + "\n");

                    continue;

                }

                for (int i = 1; i < parts.length; i++)

                {

                    parts[i] = parts[i].replaceAll("[&,]", "");

                    if (parts[i].contains("="))

                    {

                        ++kp;

                        String keywordParam[] = parts[i].split("=");

                        pntab.put(keywordParam[0], paramNo++);

                        if (keywordParam.length == 2)

                        {

                            kpdt.write(keywordParam[0] + "\t" + keywordParam[1] + "\n");

                        }

                        else

                        {

                            kpdt.write(keywordParam[0] + "\t-\n");

                        }

                    }

                    else

                    {

                        pntab.put(parts[i], paramNo++);

                        pp++;

                    }

                }

                mnt.write(parts[0] + "\t" + pp + "\t" + kp + "\t" + mdtp + "\t" + (kp == 0 ? kpdtp : (kpdtp + 1)) + "\n");

                kpdtp = kpdtp + kp;

            }

            else if (parts[0].equalsIgnoreCase("MEND"))

            {

                mdt.write(line + "\n");

                flag = kp = pp = 0;

                mdtp++;

                paramNo = 1;

                // Write macro name and parameters with indices to pntab.txt

                pnt.write(Macroname + ":\t");

                for (String param : pntab.keySet()) {

                    pnt.write(param + "(" + pntab.get(param) + ")\t");

                }

                pnt.write("\n");

                pntab.clear();

            } else if (flag == 1)

            {

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

                {

                    if (parts[i].contains("&"))

                    {

                        parts[i] = parts[i].replaceAll("[&,]", "");

                        mdt.write("(P," + pntab.get(parts[i]) + ")\t");

                    } else

                    {

                        mdt.write(parts[i] + "\t");

                    }

                }

                mdt.write("\n");

                mdtp++;

            }

        }

        br.close();

        mdt.close();

        mnt.close();

        pnt.close();

        kpdt.close();

        System.out.println("Macro Pass1 Processing done. :)");

    }

}

/\*

macro\_input.asm

MACRO

M1 &X,  &Y,  &A=AREG,  &B=

MOVER &A,  &X

ADD &A,  =’1’

MOVER &B,  &Y

ADD &A,  =’5’

MEND

MACRO

M2 &P,  &Q,  &U=CREG,  &V=DREG

MOVER &U,  &P

MOVER &V,  &Q

ADD &U,  =’15’

ADD &V,  =’10’

MEND

\*/