import java.io.\*;

public class M\_pass1 {

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

BufferedReader br1 = new BufferedReader(new FileReader("K:\\Drive\\OneDrive\\Desktop\\LP1 SPOS\\Code\\input.txt"));

String line;

mdt[] MDT = new mdt[20];

mnt[] MNT = new mnt[4];

arglist[] ARGLIST = new arglist[10];

boolean macro\_start = false,macro\_end=false,fill\_arglist=false;

int mdt\_cnt=0,mnt\_cnt=0,arglist\_cnt=0;

while((line=br1.readLine())!=null)

{

line=line.replaceAll(","," ");

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

MDT[mdt\_cnt] = new mdt();

String stmnt = "";

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

{

if(tokens[i].equalsIgnoreCase("mend"))

{

MDT[mdt\_cnt++].stmnt = "\t"+tokens[i];

macro\_end = true;

}

if(tokens[i].equalsIgnoreCase("macro"))

{

macro\_start = true;

macro\_end = false;

}

else if(!macro\_end)

{

if(macro\_start)

{

MNT[mnt\_cnt++] = new mnt(tokens[i],mdt\_cnt);

macro\_start = false;

fill\_arglist = true;

}

if(fill\_arglist)

{

while(i<tokens.length)

{

MDT[mdt\_cnt].stmnt = MDT[mdt\_cnt].stmnt+"\t"+tokens[i];

stmnt = stmnt+"\t"+tokens[i];

if(tokens[i].matches("&[a-zA-Z]")||tokens[i].matches("&[a-zA-Z]+[0-9]"))

{

ARGLIST[arglist\_cnt++] = new arglist(tokens[i]);

}

i++;

}

fill\_arglist = false;

}

else

{

if(tokens[i].matches("[a-zA-Z]+")||tokens[i].matches("[a-zA-Z]+[0-9]+")|| tokens[i].matches("[0-9]+"))

{

MDT[mdt\_cnt].stmnt = MDT[mdt\_cnt].stmnt+"\t"+tokens[i];

stmnt = stmnt +"\t"+ tokens[i];

}

if(tokens[i].matches("&[a-zA-Z]+")||tokens[i].matches("&[a-zA-z]+[0-9]"))

{

for(int j=0;j<arglist\_cnt;j++)

{

if(tokens[i].equals(ARGLIST[j].argname))

{

MDT[mdt\_cnt].stmnt = MDT[mdt\_cnt].stmnt+"\t #"+(j+1);

stmnt = stmnt +"\t #"+(j+1);

}

}}

}}}

if(stmnt!="" && !macro\_end)

{

mdt\_cnt++;

}}

br1.close();

BufferedWriter bw1 = new BufferedWriter(new FileWriter("K:\\Drive\\OneDrive\\Desktop\\LP1 SPOS\\Code\\mnt.java"));

System.out.println("\n\n\*\*\*\*\*MACRO NAME TABLE\*\*\*\*\*");

System.out.println("\n\tINDEX\tNAME\tADDRESS");

for(int i = 0 ; i<mnt\_cnt;i++){

System.out.println("\t"+i+"\t"+MNT[i].name+"\t"+MNT[i].addr);

bw1.write(MNT[i].name+"\t"+MNT[i].addr+"\n");

}

bw1.close();

bw1 = new BufferedWriter(new FileWriter("K:\\Drive\\OneDrive\\Desktop\\LP1 SPOS\\Code\\arglist.java"));

System.out.println("\n\n\*\*\*\*\*ARGUMENT LIST\*\*\*\*\*");

System.out.println("\n\tINDEX\tNAME\tADDRESS");

for(int i=0;i<arglist\_cnt;i++){

System.out.println("\t"+i+"\t"+ARGLIST[i].argname);

bw1.write(ARGLIST[i].argname+"\n");

}

bw1.close();

System.out.println("\n\n\*\*\*\*\*MACRO DEFINATION TABLE\*\*\*\*\*");

System.out.println("\n\tINDEX\tSTATEMENT");

bw1 = new BufferedWriter(new FileWriter("K:\\Drive\\OneDrive\\Desktop\\LP1 SPOS\\Code\\mdt.java"));

for(int i=0;i<mdt\_cnt;i++)

{

System.out.println("\t"+i+"\t"+MDT[i].stmnt);

bw1.write(MDT[i].stmnt+"\n");

}

bw1.close();

}

}

**Input.txt**

MACRO

INCR &X,&Y,&REG1=AREG

MOVER &REG1,&X

ADD &REG1,&Y

MOVEM &REG1,&X

MEND

MACRO

DECR &A,&B,&REG2=BREG

MOVER &REG2,&A

SUB &REG2,&B

MOVEM &REG2,&A

MEND

START 100

READ N1

READ N2

DECR N1,N2

INCR N1,N2

STOP

N1 DS 1

N2 DS 2

END

**Mdt.java**

**public** **class** mdt {

String stmnt;

**public** mdt()

{

stmnt="";

}

}

**Arglist.java**

**public** **class** arglist {

String argname;

arglist(String argument){

**this**.argname=argument;

}

}

**Mnt.java**

**public** **class** mnt {

String name;

**int** addr;

**int** arg\_cnt;

mnt(String nm, **int** address)

{

**this**.name=nm;

**this**.addr=address;

**this**.arg\_cnt=0;

}

}

**MDT.txt**

**MNT.txt**

**ARGLIST.txt**

**OUTPUT:-**

