# MACRO PASS 2

import java.io.\*;

class arglist {

String argname,value; arglist(String argument) {

// TODO Auto-generated constructor stub

this.argname=argument; this.value="";

}

}

class mnt {

String name; int addr; int arg\_cnt;

mnt(String nm, int address,int total\_arg)

{

this.name=nm; this.addr=address; this.arg\_cnt=total\_arg;

}

}

class mdt { String stmnt; public mdt() {

// TODO Auto-generated constructor stub

stmnt="";

}

}

public class Mpass2 {

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

// TODO Auto-generated method stub mdt[] MDT=new mdt[20]; mnt[] MNT=new mnt[4];

arglist[] formal\_parameter=new arglist[10];

int macro\_addr = -1;

boolean macro\_start=false,macro\_end=false; int macro\_call = -1;

int

mdt\_cnt=0,mnt\_cnt=0,formal\_arglist\_cnt=0,actual\_arglist\_cnt=0,temp\_cnt=0,temp\_cnt1=0;

BufferedReader br1=new BufferedReader(new FileReader("C:\Users\vrudr\OneDrive\Desktop\MPass1\MPass1\\MNT.txt"));

String line;

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

{

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

System.out.println("\t"+"\t"+parts[0]+"\t"+parts[1]+"\t\t"+parts[2]);

MNT[mnt\_cnt++]=new mnt(parts[0], Integer.parseInt(parts[1]),Integer.parseInt(parts[2]));

}

br1.close();

System.out.println("\n\t\*\*\*\*\*\*\*\*MACRO NAME TABLE\*\*\*\*\*\*\*\*\*\*"); System.out.println("\n\tINDEX\tNAME\tADDRESS\tTOTAL ARGUMENTS"); for(int i=0;i<mnt\_cnt;i++)

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

br1=new BufferedReader(new

FileReader("C:\Users\vrudr\OneDrive\Desktop\MPass1\MPass1\\argmnt.txt")); while((line = br1.readLine())!=null)

{

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

formal\_parameter[formal\_arglist\_cnt++]=new arglist(parameters[0]);

if(parameters.length>1)

formal\_parameter[formal\_arglist\_cnt-1].value = parameters[1];

}

br1.close();

System.out.println("\n\n\t\*\*\*\*\*\*\*\*FORMAL ARGUMENT LIST\*\*\*\*\*\*\*\*\*\*"); System.out.println("\n\tINDEX\tNAME\tVALUE"); for(int i=0;i<formal\_arglist\_cnt;i++)

System.out.println("\t"+i+"\t"+formal\_parameter[i].argname+"\t"+formal\_parameter[i].valu e);

br1=new BufferedReader(new

FileReader("C:\Users\vrudr\OneDrive\Desktop\MPass1\MPass1\\MDT.txt")); while((line = br1.readLine())!=null)

{

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

MDT[mdt\_cnt++].stmnt=line;

}

br1.close();

System.out.println("\n\t\*\*\*\*\*\*\*\*MACRO DEFINITION TABLE\*\*\*\*\*\*\*\*\*\*"); System.out.println("\n\tINDEX\t\tSTATEMENT"); for(int i=0;i<mdt\_cnt;i++)

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

br1=new BufferedReader(new

FileReader("C:\Users\vrudr\OneDrive\Desktop\MPass1\MPass1\\input4.txt"));

arglist[] actual\_parameter=new arglist[10];

BufferedWriter bw1 = new BufferedWriter(new

FileWriter("C:\Users\vrudr\OneDrive\Desktop\MPass1\MPass1\\output4.txt"));

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

{

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

temp\_cnt1=0;

for(String current\_token:tokens)

{

if(current\_token.equalsIgnoreCase("macro"))

{

macro\_start=true; macro\_end=false;

}

if(macro\_end && !macro\_start)

{

if(macro\_call != -1 && temp\_cnt<formal\_arglist\_cnt-1)

{

if(formal\_parameter[actual\_arglist\_cnt].value != "")

actual\_parameter[actual\_arglist\_cnt++]=new arglist(formal\_parameter[actual\_arglist\_cnt-

1].value);

actual\_parameter[actual\_arglist\_cnt++]=new arglist(current\_token);

//System.out.println("current token="+(actual\_arglist\_cnt-1));

//System.out.println("current para="+actual\_parameter[actual\_arglist\_cnt-1].argname);

if(formal\_parameter[actual\_arglist\_cnt].value != "")

actual\_parameter[actual\_arglist\_cnt++]=new arglist(formal\_parameter[actual\_arglist\_cnt-

1].value);

}

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

{

if(current\_token.equals(MNT[i].name))

{

macro\_call=i;

temp\_cnt1 = temp\_cnt1 +MNT[i].arg\_cnt;

break;

}

temp\_cnt1 = temp\_cnt1 + MNT[i].arg\_cnt;

}

if(macro\_call == -1) bw1.write("\t" + current\_token);

}

if(current\_token.equalsIgnoreCase("mend"))

{

macro\_end=true; macro\_start=false;

}

}

if(macro\_call != -1)

{

macro\_addr=MNT[macro\_call].addr+1;

while(true)

{

if(MDT[macro\_addr].stmnt.contains("mend") ||

MDT[macro\_addr].stmnt.contains("MEND"))

{

macro\_call = -1;

break;

}

else

{

bw1.write("\n");

String[]

temp\_tokens=MDT[macro\_addr++].stmnt.split("\\s+");

for(String temp:temp\_tokens)

{

if(temp.matches("#[0-9]+"))

{

int num =

Integer.parseInt(temp.replaceAll("[^0-9]+", ""));

bw1.write(actual\_parameter[num-

1].argname+"\t");

}

else

bw1.write(temp + "\t");

}

}

}

}

if(!macro\_start )

bw1.write("\n"); macro\_call= -1;

}

br1.close(); bw1.close();

System.out.println("\n\n\t\*\*\*\*\*\*\*\*ACTUAL ARGUMENT LIST\*\*\*\*\*\*\*\*\*\*"); System.out.println("\n\tINDEX\tNAME"); for(int i=0;i<actual\_arglist\_cnt;i++)

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

}

}

# Input4.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

INCR N1,N2

DECR N1,N3

STOP

N1 DS 1

N2 DS 2

N3 DS 1

END

# Output



