Roll No: 3260

Experiment No:2

Title: .Design Suitable data structures and implement Pass-I and Pass-II of tow-pass macroprocessor. The output of Pass-I (MNT,MDT and intermediate code file without any macro definitions) should be input for Pass-II.

PASS:1

Macropass

```
import java.io.*;

import java.util.*;


class MNT {

        String macroname;

        int mdtc;

        public MNT(String m,int mdp)

        {

                macroname=m;

                mdtc=mdp;

        }

}


class MacroPass{

        static List <MNT>mnt=new LinkedList<MNT>();

        static List<String>ala=new LinkedList<String>();

        static List<String>mdt=new ArrayList<String>();

        static int mntc=0;

        static int mdtc=0;
```

```
    static BufferedReader br;

    static BufferedWriter bw;

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

        String line;

        br=new BufferedReader(new FileReader("C:\\Exp2\\ip"));

        bw=new BufferedWriter(new FileWriter("C:\\Exp2\\op.txt"));


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

        {

            if(line.equalsIgnoreCase("MACRO"))

                process_Def(line);

            else

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

        }

        System.out.println("ALA");

        printala();

        System.out.println("MNT");

        printmnt();

        System.out.println("MDT");

        printmdt();

        bw.close();

    }


    static void printala() {

        int i=0;

        for(String l:ala)
```

```
          {
                    System.out.println(i+" "+l);
                    i++;
          }
     }

     static void printmnt() {
          int i=0;
          for(MNT l:mnt) {
                    System.out.println(i+" "+l.macroname+" "+l.mdtc);
                    i++;
          }
     }

     static void printmdt() {
          int i=0;
          for(String l:mdt) {
                    System.out.println(i+" "+l);
                    i++;
          }
     }

static void process_Def(String line) throws IOException{

          String l;
          l=br.readLine();
```

```
String tk[]=l.split(" ");
mnt.add(new MNT(tk[0],mdtc));
mdtc++;
String arg[]=tk[1].split(",");

for(int i=0;i<arg.length;i++)
        ala.add(arg[i]);
mdt.add(l);
mdtc++;

while(!l.equalsIgnoreCase("MEND"))
{
        int i=0,ind;
        String opline=" ";
        l=br.readLine();
        ind=l.indexOf("&");
        if(ind>0)
        {
                String wrd[]=l.split(" ");
                opline=opline+wrd[0];
                String margs[]=wrd[1].split(",");
                opline=opline+" "+margs[0];

                while(i<margs.length)
                {
                        if(margs[i].startsWith("&"))
```

```
                    {
                        ind=ala.indexOf(margs[i]);
                        opline=opline+"#"+ind;

                    }
                    i++;
                }
            }
            else
                opline=l;
            mdt.add(opline);
            mdtc++;

        }

    }
}
```

Ip(input file)

```
MACRO
INCR &FIRST,&SECOND
ADD AREG,&FIRST
SUB BREG,&SECOND
MUL AREG A
MEND
```

Output :

```
ALA
0 &FIRST
1 &SECOND
```

```
MNT
0 INCR 0
MDT
0 INCR &FIRST,&SECOND
1  ADD AREG#0
2  SUB BREG#1
3 MUL AREG A
4 MEND
```

PASS2:

Mpass2

```
import java.util.*;

import java.io.*;


class MNT {

String name;

int index;


MNT(String s, int i) {

        name = s;

        index = i;

}

}

class ALA

{

        String formal;

        String actual;

        ALA(String f,String a){
```

```
                        formal=f;
                        actual=a;
            }
}
public class Mpass2 {
            static List<MNT> mnt;
            static List<String> mdt;
            static int mntc;
            static int mdtc;
            static int mdtp;
            static List<ALA> ala;
    static BufferedReader br;
    static BufferedReader br1;
    static BufferedWriter bw;

public static void main(String args[]) throws Exception {
bw=new BufferedWriter(new FileWriter("C:\\Users\\DELL\\eclipse-
workspace\\Exp2_Pass2\\pass2"));

                        String line=" ";
                        initializeTables();
                        System.out.println("ALA:");
                        showAla(1);
                        System.out.println("\nMNT:");
                        showMnt();
                        System.out.println("\nMDT:");
                        showMdt();
```

```
        System.out.println("\n===== PASS 2 =====\n");
      br1=new BufferedReader(new FileReader("C:\\Users\\DELL\\eclipse-workspace\\Exp2_Pass2\\op"));

  while((line=br1.readLine())!=null)
        { int flag=0;

            for(MNT l : mnt){
                if(line.contains(l.name))
                {   //macro call found process macro call

                    mdtp=l.index;
                    System.out.println(line);
                    process_call(mdtp,line); //call expansion
                    flag=1;
                    break;
                }
            }
            if(flag==0)
            {
                bw.write(line+"\n");
                System.out.println(line);
            }

        }
 bw.close();
```

```
    }


    static void process_call(int mdtp,String s) throws Exception
    {
        String mname[]=s.split(" ");
        String actual_args[]=mname[1].split(",");
        String mdt_words[]=mdt.get(mdtp).split(" "); //read line from
MDT and split
        String args[]=mdt_words[1].split(",");


        for(int i=0;i<args.length;i++)
        {
            for(int j=0;j<ala.size();j++) {
                ALA l=ala.get(j);
              if(l.formal.equals(args[i]))
              {
                  //formal argument found, so set actual one
                  ala.set(j,new ALA(l.formal,actual_args[i]));
              }
            }
        }
        //Show ALA After setting Actual arguments
        mdtp++;
        String final1="";
        while(!mdt.get(mdtp).equals("MEND"))
        {
            String op_line=mdt.get(mdtp);
```

```
                              mdtp++;
                              if(op_line.contains("#"))
                              { int ind=op_line.indexOf("#");
                                final1=op_line.substring(0,ind);

ind=Integer.parseInt(op_line.substring(ind+1,op_line.length()));

                                ALA l=ala.get(ind);
                                final1=final1+l.actual;
                              }
                              else
                                      final1=op_line;
                          System.out.println(final1);
                    bw.write(final1+"\n");


                    }
          }
          static void showAla(int pass) throws Exception {
                    int i=0;
                    for(ALA l : ala) {
                              System.out.println(i+" "+l.formal+" "+l.actual);
                              i++;
                    }
          }

          static void showMnt() throws Exception {
                    int i=0;
```

```
        for(MNT l : mnt) {
            System.out.println(i+" "+l.name+" "+l.index);
            i++     }
    }

    static void showMdt() throws Exception {
        int i=0;
        for(String l : mdt) {
            System.out.println(i+" "+l);
            i++;

        }
    }

    static void initializeTables() throws Exception{
        mnt = new LinkedList<MNT>();
        mdt = new ArrayList<String>();
        ala = new LinkedList<ALA>();

        String mname=new String();
        //Load MNT
        String s="";
      br=new BufferedReader(new FileReader("C:\\Users\\DELL\\eclipse-
workspace\\Exp2_Pass2\\MNT"));

        while((s=br.readLine())!=null) {
                String words[]=s.split(" ");
                mnt.add(new MNT(words[0],Integer.parseInt(words[1])));

            }
            //load MDT
          br=new BufferedReader(new FileReader("C:\\Users\\DELL\\eclipse-workspace\\Exp2_Pass2\\MDT"));

            while((s=br.readLine())!=null) {
                mdt.add(s);
            }
            //Load ALA pass1
          br=new BufferedReader(new FileReader("C:\\Users\\DELL\\eclipse-workspace\\Exp2_Pass2\\ALA"));
            while((s=br.readLine())!=null) {

                String words[]=s.split(" ");
                for(int i=0;i<words.length;i++)
                  ala.add(new ALA(words[i],"-"));
            }
         br.close();

        }
}//end of class
```

ALA (input file)

```
&FIRST
&SECOND
```

MDT (Input file)

```
INCR1 &FIRST,&SECOND
A 1,#0
L 2,#1
ST 1,#0
MEND
```

MNT(Input file)

```
INCR1 0
```

op (Input file)

```
PRG2 START
USING *,14
INCR1 TEMP,RES
SR 1,1
INCR1 FOUR,FIVE
FOUR DC  F'4'
FIVE DC  F'5'
RES  DS  1F
TEMP  DC  F'2'
END
```

pass2 (output file)

```
PRG2 START
USING *,14
A 1,TEMP
L 2,RES
ST 1,TEMP
SR 1,1
A 1,FOUR
L 2,FIVE
ST 1,FOUR
FOUR DC  F'4'
FIVE DC  F'5'
RES  DS  1F
TEMP  DC  F'2'
END
```

Output:

```
ALA:
0 &FIRST -
1 &SECOND -

MNT:
0 INCR1 0

MDT:
0 INCR1 &FIRST,&SECOND
1 A 1,#0
2 L 2,#1
3 ST 1,#0
4 MEND

===== PASS 2 =====

PRG2 START
USING *,14
INCR1 TEMP,RES
A 1,TEMP
L 2,RES
ST 1,TEMP
SR 1,1
INCR1 FOUR,FIVE
A 1,FOUR
L 2,FIVE
ST 1,FOUR
FOUR DC  F'4'
FIVE DC  F'5'
RES  DS  1F
TEMP  DC  F'2'
END
```