**Name:** Sanika Kendre **Batch:** C

**Class:** TE Computer A **Roll No.:**21CO059

**Problem Statement:**

**Design suitable Data structures and implement Pass-II of a two-pass assembler for pseudo-machine. Implementation should consist of a few instructions from each category and few assembler directives.The output of Pass-I (intermediate code file and symbol table) should be input for Pass-II.**

**Program:**

import java.io.BufferedReader;

import java.io.BufferedWriter;

import java.io.FileNotFoundException;

import java.io.FileReader;

import java.io.FileWriter;

import java.util.ArrayList;

public class PassTwo {

ArrayList<Integer> symaddr=new ArrayList<>();

ArrayList<String> littab=new ArrayList<>();

ArrayList<Integer> litaddr=new ArrayList<>();

ArrayList<Integer> pooltab=new ArrayList<>();

public void readFile() throws Exception {

BufferedReader br=new BufferedReader(new FileReader("sym.txt"));

String line=" ";

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

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

symaddr.add(Integer.parseInt(split[1]));

}

BufferedReader br2=new BufferedReader(new FileReader("lit.txt"));

while((line=br2.readLine())!=null) {

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

littab.add(split[0]);

litaddr.add(Integer.parseInt(split[1]));

}

BufferedReader br3=new BufferedReader(new FileReader("pool.txt"));

while((line=br3.readLine())!=null) {

pooltab.add(Integer.parseInt(line));

}

br.close();

br2.close();

br3.close();

}

public void generateMachineCode() throws Exception {

String line=" ";

BufferedWriter wr=new BufferedWriter(new FileWriter("output.txt"));

BufferedReader br=new BufferedReader(new FileReader("intercode.txt"));

int p=0;

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

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

String sp[]=split[0].replace("(","").replace(")", "").split("\\,");

if(sp[0].equals("DL")) {

String arr[]=split[1].replace("(", "").replace(")",

"").split("\\,");

wr.write("+00 0 "+String.format("%03d",Integer.parseInt(arr[1]))+"\n");

}

else if(sp[0].equals("IS")) {

wr.write(sp[1]+" ");

int i=1;

while(i<split.length) {

String arr[]=split[i].replace("(",

"").replace(")","").split("\\,");

if(arr[0].equals("S")) {

wr.write(symaddr.get(Integer.parseInt(arr[1]))+"\n");

}

else if(arr[0].equals("L")) {

wr.write(litaddr.get(Integer.parseInt(arr[1]))+"\n");

}

else if(arr[0].equals("C")) {

wr.write((Integer.parseInt(arr[1]))+" ");

}

else if(arr[0].equals("RG")){

wr.write((Integer.parseInt(arr[1])+" "));

}

else {

wr.write(arr[0]+" ");

}

i++;

}

}

else if(sp[0].equals("AD") && (sp[1].equals("04") ||

sp[1].equals("05"))){

if(sp[1].equals("05")) {

for(int i=pooltab.get(p);i<pooltab.get(p+1);i++) {

wr.write("+00 0"+Integer.parseInt(littab.get(i))+"\n");

}

}

else {

for(int i=pooltab.get(p);i<littab.size();i++) {

wr.write("+000"+Integer.parseInt(littab.get(i))+"\n");

}

}

p++;

}

}

br.close();

wr.close();

}

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

PassTwo p=new PassTwo();

p.readFile();

p.generateMachineCode();

}

}

**Input:**

**intercode.txt**

(AD,01)(C,100)

(IS,04) (RG,1) (S,0)

(IS,01) (RG,2) (L,0)

(IS,05) (RG,1) (S,1)

(IS,02) (RG,3) (L,1)

(AD,05)

(IS,01) (RG,4) (L,2)

(DL,01) (C,10)

(AD,05)

(DL,02) (C,1)

(AD,02)

**lit.txt**

6 105

1 106

5 120

**sym.txt**

B 119

A 123

**pool.txt**

0

2

3

**Output:**

04 1 119

01 2 105

05 1 123

02 3 106

+00 06

+00 01

01 4 120

+00 0 010

+00 05

+00 0 001