/\*

Assignment no: 1

Implement Pass I of II pass Assembler

\*/

import java.util.\*;

import java.io.\*;

class Pass1 {

public String code[][] = new String[20][4];

public int lc, codeSize;

public int systabPtr, optabPtr, littabPtr, pooltabPtr;

public String SYSTAB[][] = new String[20][3];

public String OPTAB[][] = new String[20][3];

public String LITTAB[][] = new String[20][2];

public String POOLTAB[] = new String[20];

public String opcode[][] = new String[28][3];

public Pass1() {

try {

File file = new File("C:\\Users\\Admin\\Desktop\\SQL\\Opcode.txt");

BufferedReader br = new BufferedReader(new FileReader(file));

String temp1;

int j, i = 0;

while ((temp1 = br.readLine()) != null) {

String temp2[] = temp1.split("\t");

for (j = 0; j < 3; j++)

opcode[i][j] = temp2[j];

i++;

}

systabPtr = 0;

optabPtr = 0;

littabPtr = 0;

pooltabPtr = 1;

POOLTAB[0] = "#0";

} catch (Exception e) {

System.out.println(e);

}

}

public void getData() {

try {

File file = new File("C:\\Users\\Admin\\Desktop\\SQL\\input1.txt");

BufferedReader br = new BufferedReader(new FileReader(file));

String temp1;

int j, i = 0;

while ((temp1 = br.readLine()) != null) {

String temp2[] = temp1.split("\t");

for (j = 0; j < temp2.length; j++)

code[i][j] = temp2[j];

while (j < 4) {

code[i][j] = "";

j++;

}

i++;

}

codeSize = i;

} catch (Exception e) {

System.out.println(e);

}

}

public void putData() {

for (int i = 0; i < codeSize; i++) {

System.out.println(code[i][0] + "\t" + code[i][1] + "\t" + code[i][2] + "\t" + code[i][3]);

}

System.out.println("\n\nSys Table:");

for (int i = 0; i < systabPtr; i++) {

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

}

System.out.println("\n\nOp Table:");

for (int i = 0; i < optabPtr; i++) {

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

}

System.out.println("\n\nLit Table:");

for (int i = 0; i < littabPtr; i++) {

System.out.println(LITTAB[i][0] + "\t" + LITTAB[i][1]);

}

System.out.println("\n\nPool Table:");

for (int i = 0; i < pooltabPtr; i++) {

System.out.println(POOLTAB[i]);

}

}

public void tableCreation() {

int i;

for (i = 0; i < codeSize; i++) {

if (code[i][1].equals("START")) {

lc = Integer.parseInt(code[i][2]);

i++;

break;

}

}

while (i < codeSize) {

int length;

if (code[i][0].equals("") == false) {

length = 1;

if (code[i][1].equals("DS"))

length = Integer.parseInt(code[i][2]);

sysTabEntry(i, length);

}

length = 1;

if (code[i][1].equals("DS"))

length = Integer.parseInt(code[i][2]);

optTabEntry(i, length);

if (code[i][3].contains("="))

littabEntry(i);

if (code[i][1].equals("LTORG"))

pooltabEntry();

i++;

lc += length;

}

}

public void sysTabEntry(int pos, int length) {

SYSTAB[systabPtr][0] = code[pos][0];

SYSTAB[systabPtr][1] = Integer.toString(lc);

SYSTAB[systabPtr][2] = Integer.toString(length);

systabPtr++;

}

public void optTabEntry(int pos, int length) {

for (int i = 0; i < optabPtr; i++) {

if (code[pos][1].equals("DS") == true)

break;

if (OPTAB[i][0].equals(code[pos][1]) == true)

return;

}

OPTAB[optabPtr][0] = code[pos][1];

int temp = findClass(code[pos][1]);

OPTAB[optabPtr][1] = opcode[temp][1];

OPTAB[optabPtr][2] = "(" + opcode[temp][2] + "," + length + ")";

optabPtr++;

}

public void littabEntry(int pos) {

LITTAB[littabPtr][0] = code[pos][3];

LITTAB[littabPtr][1] = Integer.toString(lc);

littabPtr++;

}

public void pooltabEntry() {

POOLTAB[pooltabPtr] = "#" + littabPtr;

pooltabPtr++;

}

public int findClass(String mnemonic) {

for (int i = 0; i < 28; i++) {

if (mnemonic.equals(opcode[i][0]) == true)

return i;

}

return -1;

}

public static void main(String[] args) {

Pass1 obj = new Pass1();

obj.getData();

obj.tableCreation();

obj.putData();

}

}

**OUTPUT:**

START 1000

LOOP MOV A, B

MOV C, B

MUL D, E

DS 2

LTORG

ADD A, E

STOP

Sys Table:

Op Table:

Lit Table:

Pool Table:

#0