Practical no: 1

Problem Statement: Design suitable data structures and implement pass-I of a two-pass assembler for pseudo-machine in Java using object oriented feature. Implementation should consist of a few instructions from each category and few assembler directives.

Name: Aditi Dinesh Mulay

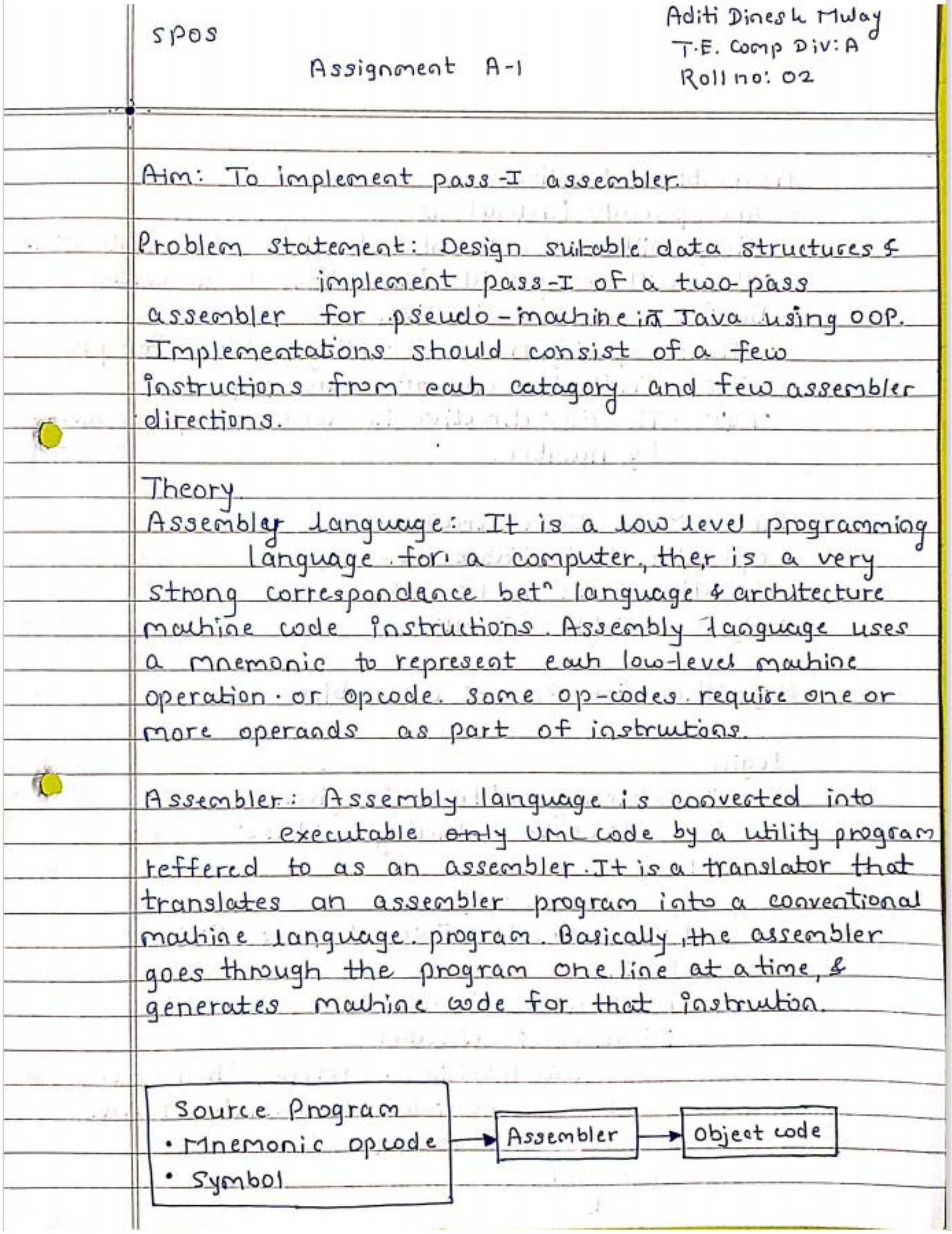
Class: T.E. Computer

Subject: SPOS

Div: A

Roll no: 02

PRN No. 71918146B



SPOS

Assignment A-1

Aditi Dinesh Mway
T.E. Comp Div: A
Roll no: 02

Aim: To implement pass-I assembler.

Problem statement: Design suitable data structures & implement pass-I of a two-pass assembler for pseudo-machine in Java using OOP. Implementations should consist of a few instructions from each catagory and few assembler directions.

Theory.

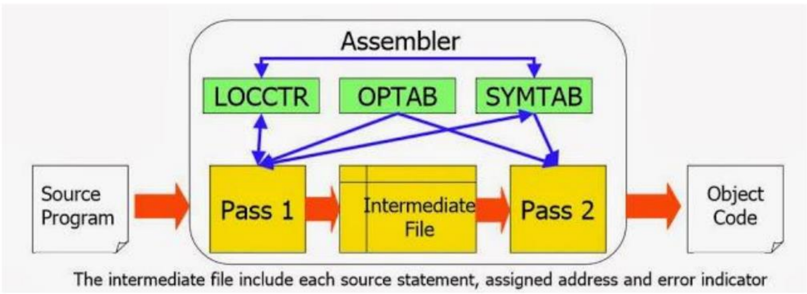
Assembler language: It is a low level programming language for a computer, ther is a very strong correspondance betn language & architecture machine code instructions. Assembly language uses a mnemonic to represent each low-level machine operation or opcode. Some op-codes require one or more operands as part of instrutions.

Assembler: Assembly language is converted into executable ~~only~~ UML code by a utility program teffered to as an assembler. It is a translator that translates an assembler program into a conventional machine language program. Basically, the assembler goes through the program one line at a time, & generates machine code for that instrution.

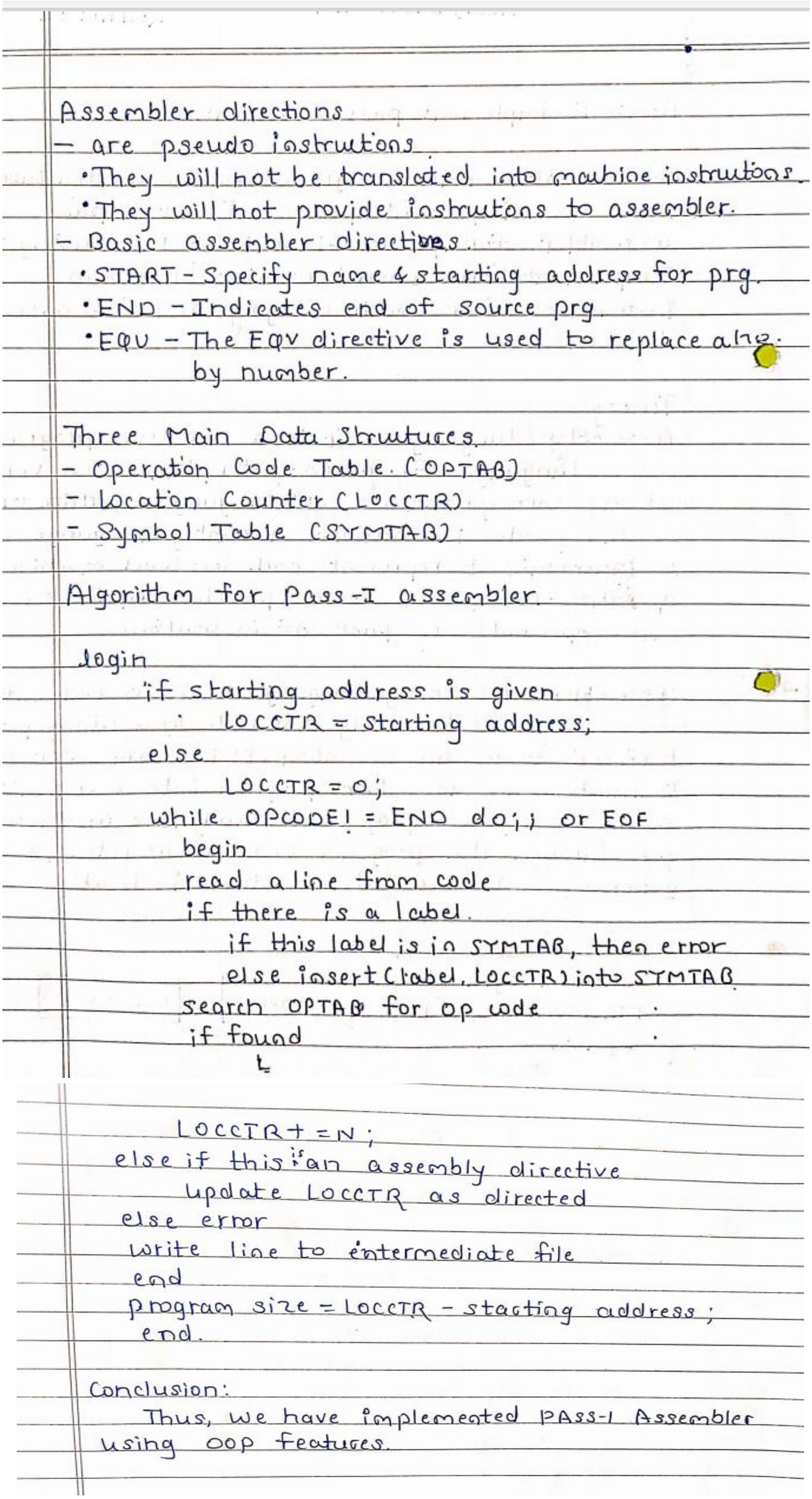
Source Program
• Mnemonic opcode
• Symbol

Assembler

Object code



The intermediate file include each source statement, assigned address and error indicator



Assembler directions

- are pseudo instrutions.
  - They will not be translated into machine instrutions.
  - They will not provide instrutions to assembler.
- Basic assembler directives.
  - START - Specify name & starting address for prg.
  - END - Indicates end of source prg.
  - EQU - The EQV directive is used to replace a1ng. by number.

Three Main Data Structures.

- Operation Code Table. (OPTAB)
- Location Counter (LOCCTR)
- Symbol Table (SYMTAB)

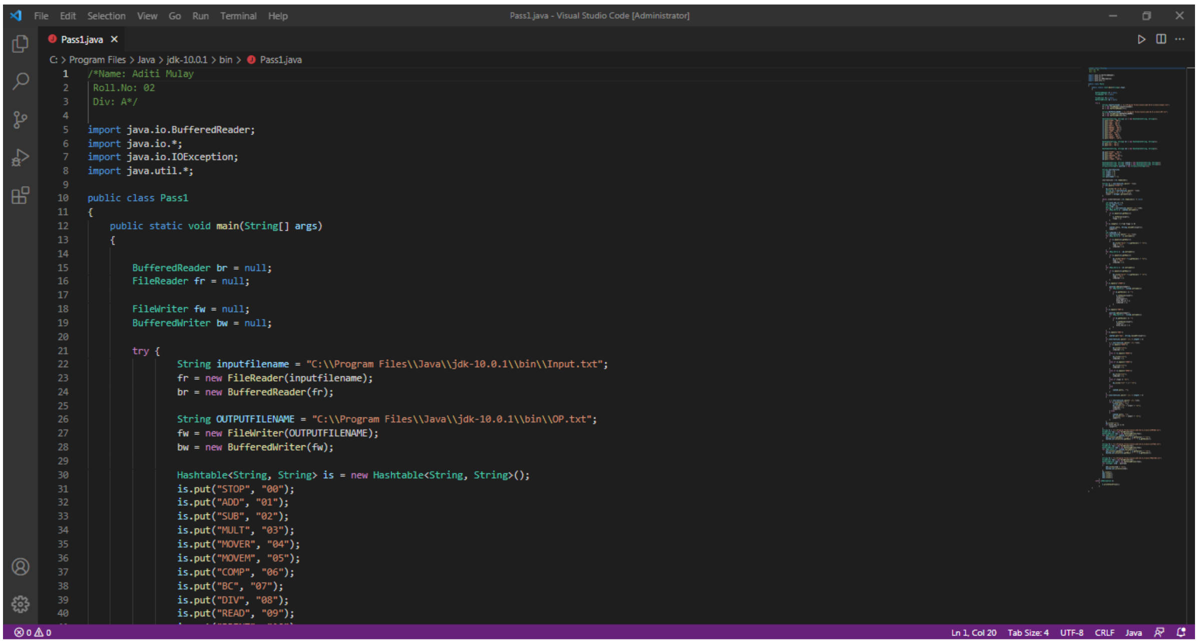
Algorithm for Pass-I assembler.

```
login
    if starting address is given
        LOCCTR = starting address;
    else
        LOCCTR = 0;
    while OPCODE! = END do;; or EOF
      begin
      read a line from code
      if there is a label.
          if this label is in SYMTAB, then error
          else insert (label, LOCCTR) into SYMTAB.
      search OPTAB for op code
      if found
          LOCCTR+ = N;
      else if this is an assembly directive
          update LOCCTR as directed
      else error
      write line to intermediate file
      end
    program size = LOCCTR - starting address;
    end.
```

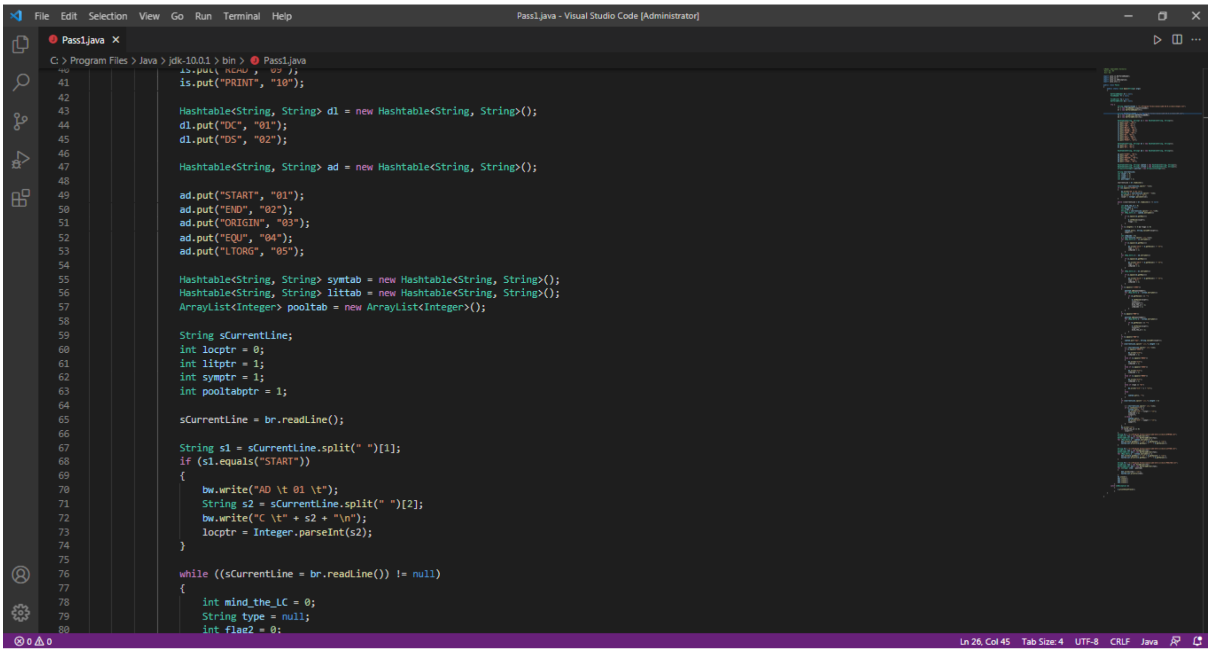
Conclusion:

Thus, we have implemented PASS-1 Assembler using OOP features.

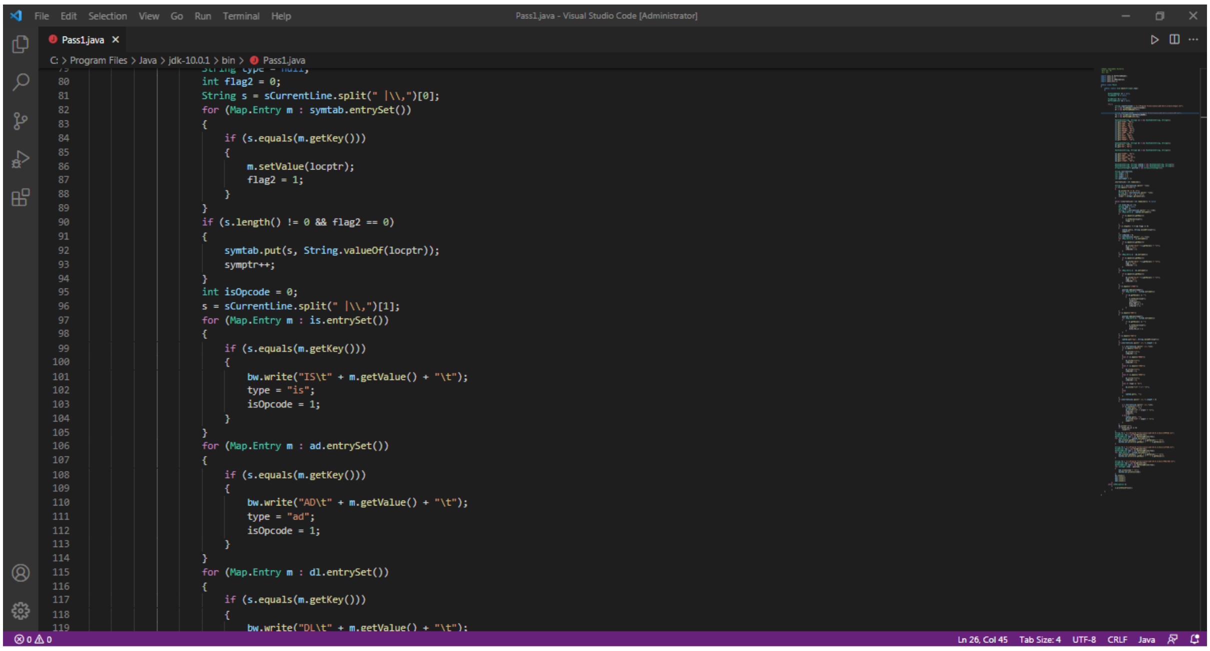
Program:


File Edit Selection View Go Run Terminal Help
Pass1.java - Visual Studio Code [Administrator]
Pass1.java
C: > Program Files > Java > jdk-10.0.1 > bin > Pass1.java
⊗ 0 ⚠ 0
Ln 1, Col 20 Tab Size: 4 UTF-8 CRLF Java

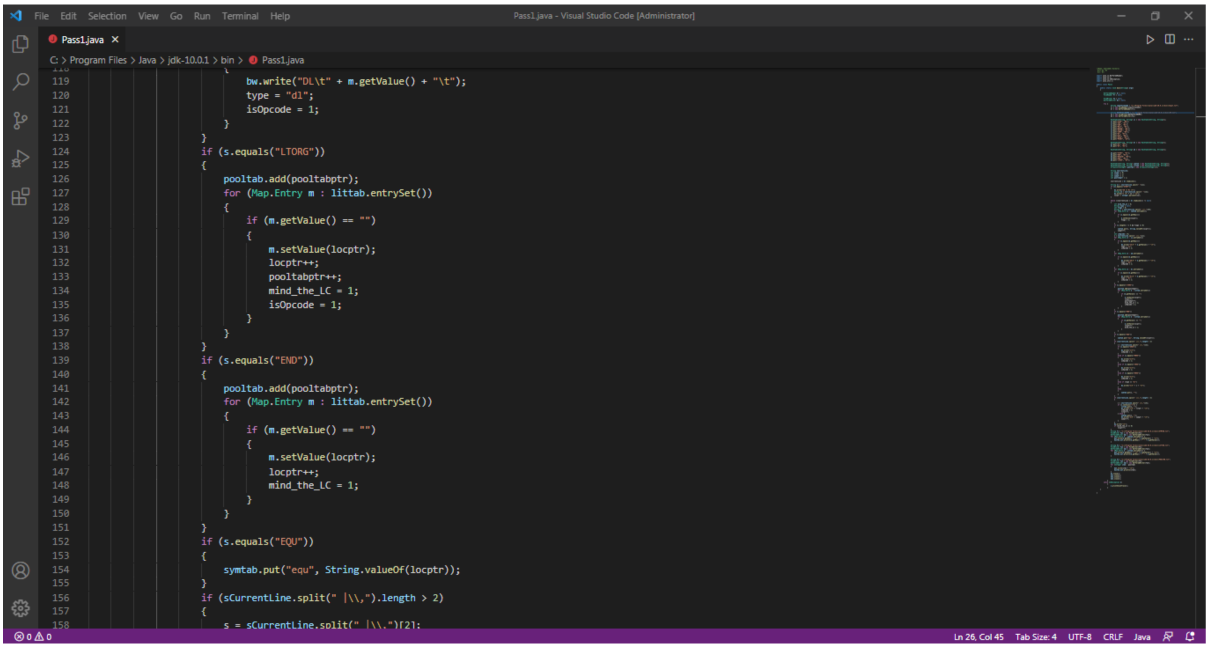
```java
/*Name: Aditi Mulay
 Roll.No: 02
 Div: A*/

import java.io.BufferedReader;
import java.io.*;
import java.io.IOException;
import java.util.*;

public class Pass1
{
    public static void main(String[] args)
    {

        BufferedReader br = null;
        FileReader fr = null;

        FileWriter fw = null;
        BufferedWriter bw = null;

        try {
                String inputfilename = "C:\\Program Files\\Java\\jdk-10.0.1\\bin\\Input.txt";
                fr = new FileReader(inputfilename);
                br = new BufferedReader(fr);

                String OUTPUTFILENAME = "C:\\Program Files\\Java\\jdk-10.0.1\\bin\\OP.txt";
                fw = new FileWriter(OUTPUTFILENAME);
                bw = new BufferedWriter(fw);

                Hashtable<String, String> is = new Hashtable<String, String>();
                is.put("STOP", "00");
                is.put("ADD", "01");
                is.put("SUB", "02");
                is.put("MULT", "03");
                is.put("MOVER", "04");
                is.put("MOVEM", "05");
                is.put("COMP", "06");
                is.put("BC", "07");
                is.put("DIV", "08");
                is.put("READ", "09");
```


File Edit Selection View Go Run Terminal Help
Pass1.java - Visual Studio Code [Administrator]
Pass1.java
C: > Program Files > Java > jdk-10.0.1 > bin > Pass1.java
⊗ 0 ⚠ 0
Ln 26, Col 45 Tab Size: 4 UTF-8 CRLF Java

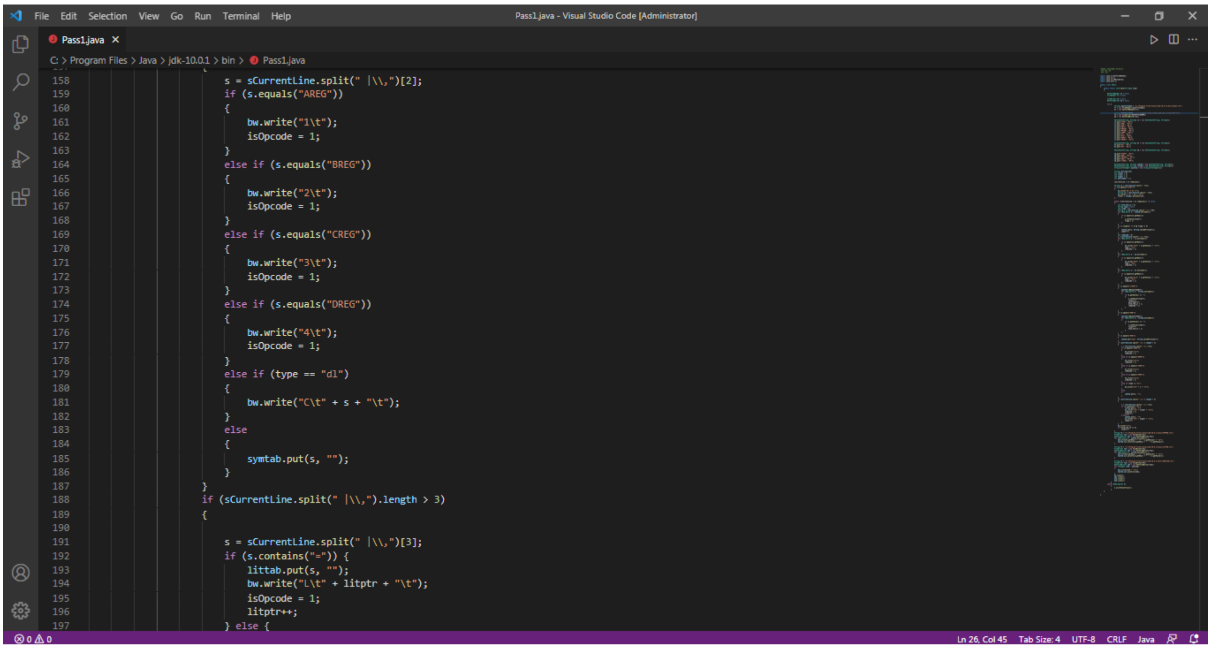
```java
                is.put("PRINT", "10");

                Hashtable<String, String> dl = new Hashtable<String, String>();
                dl.put("DC", "01");
                dl.put("DS", "02");

                Hashtable<String, String> ad = new Hashtable<String, String>();

                ad.put("START", "01");
                ad.put("END", "02");
                ad.put("ORIGIN", "03");
                ad.put("EQU", "04");
                ad.put("LTORG", "05");

                Hashtable<String, String> symtab = new Hashtable<String, String>();
                Hashtable<String, String> littab = new Hashtable<String, String>();
                ArrayList<Integer> pooltab = new ArrayList<Integer>();

                String sCurrentLine;
                int locptr = 0;
                int litptr = 1;
                int symptr = 1;
                int pooltabptr = 1;

                sCurrentLine = br.readLine();

                String s1 = sCurrentLine.split(" ")[1];
                if (s1.equals("START"))
                {
                    bw.write("AD \t 01 \t");
                    String s2 = sCurrentLine.split(" ")[2];
                    bw.write("C \t" + s2 + "\n");
                    locptr = Integer.parseInt(s2);
                }

                while ((sCurrentLine = br.readLine()) != null)
                {
                    int mind_the_LC = 0;
                    String type = null;
                    int flag2 = 0;
```


File Edit Selection View Go Run Terminal Help
Pass1.java - Visual Studio Code [Administrator]
Pass1.java
C: > Program Files > Java > jdk-10.0.1 > bin > Pass1.java
Ln 26, Col 45 Tab Size: 4 UTF-8 CRLF Java

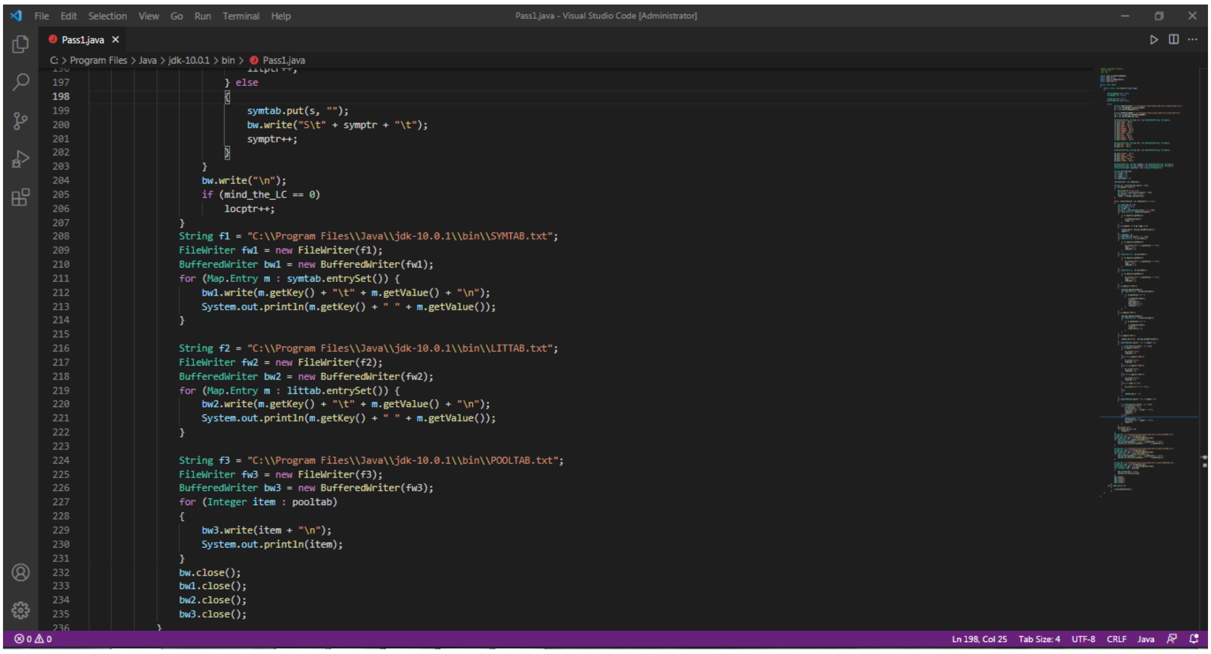
```java
String type = null;
int flag2 = 0;
String s = sCurrentLine.split(" |\\,")[0];
for (Map.Entry m : symtab.entrySet())
{
    if (s.equals(m.getKey()))
    {
        m.setValue(locptr);
        flag2 = 1;
    }
}
if (s.length() != 0 && flag2 == 0)
{
    symtab.put(s, String.valueOf(locptr));
    symptr++;
}
int isOpcode = 0;
s = sCurrentLine.split(" |\\,")[1];
for (Map.Entry m : is.entrySet())
{
    if (s.equals(m.getKey()))
    {
        bw.write("IS\t" + m.getValue() + "\t");
        type = "is";
        isOpcode = 1;
    }
}
for (Map.Entry m : ad.entrySet())
{
    if (s.equals(m.getKey()))
    {
        bw.write("AD\t" + m.getValue() + "\t");
        type = "ad";
        isOpcode = 1;
    }
}
for (Map.Entry m : dl.entrySet())
{
    if (s.equals(m.getKey()))
    {
        bw.write("DL\t" + m.getValue() + "\t");
```


File Edit Selection View Go Run Terminal Help
Pass1.java - Visual Studio Code [Administrator]
Pass1.java
C: > Program Files > Java > jdk-10.0.1 > bin > Pass1.java
Ln 26, Col 45 Tab Size: 4 UTF-8 CRLF Java

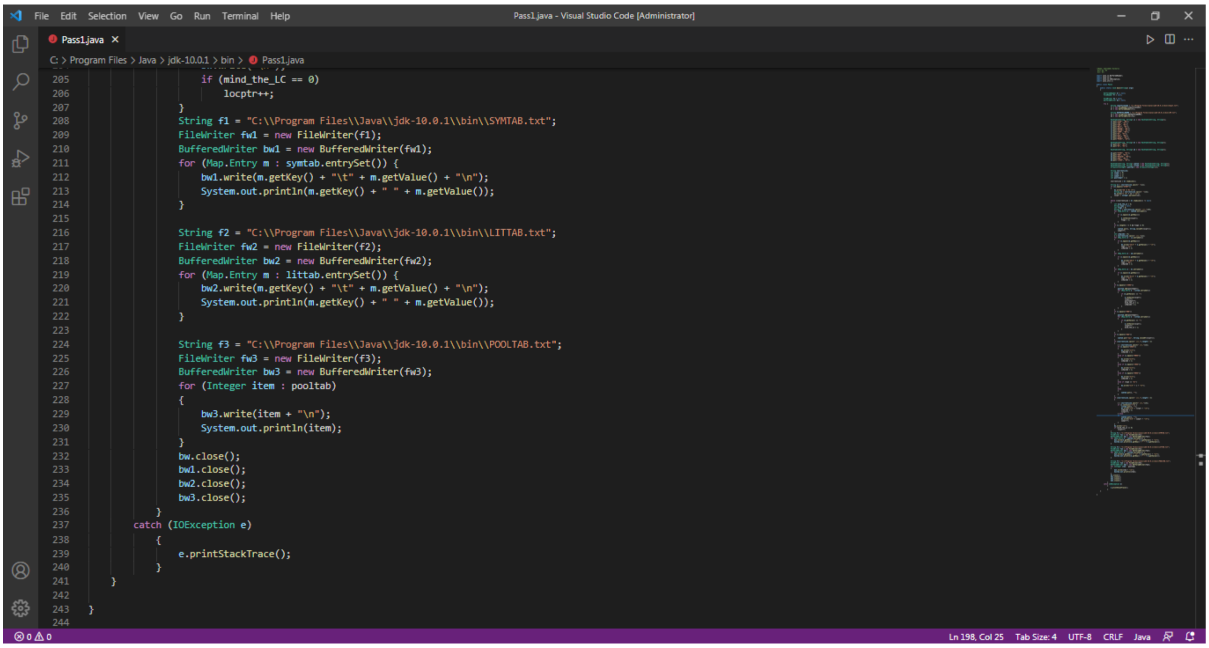
```java
        bw.write("DL\t" + m.getValue() + "\t");
        type = "dl";
        isOpcode = 1;
    }
}
if (s.equals("LTORG"))
{
    pooltab.add(pooltabptr);
    for (Map.Entry m : littab.entrySet())
    {
        if (m.getValue() == "")
        {
            m.setValue(locptr);
            locptr++;
            pooltabptr++;
            mind_the_LC = 1;
            isOpcode = 1;
        }
    }
}
if (s.equals("END"))
{
    pooltab.add(pooltabptr);
    for (Map.Entry m : littab.entrySet())
    {
        if (m.getValue() == "")
        {
            m.setValue(locptr);
            locptr++;
            mind_the_LC = 1;
        }
    }
}
if (s.equals("EQU"))
{
    symtab.put("equ", String.valueOf(locptr));
}
if (sCurrentLine.split(" |\\,").length > 2)
{
    s = sCurrentLine.split(" |\\,")[2];
```


File Edit Selection View Go Run Terminal Help
Pass1.java - Visual Studio Code [Administrator]
Pass1.java
C: > Program Files > Java > jdk-10.0.1 > bin > Pass1.java
⊗ 0 ⚠ 0
Ln 26, Col 45 Tab Size: 4 UTF-8 CRLF Java

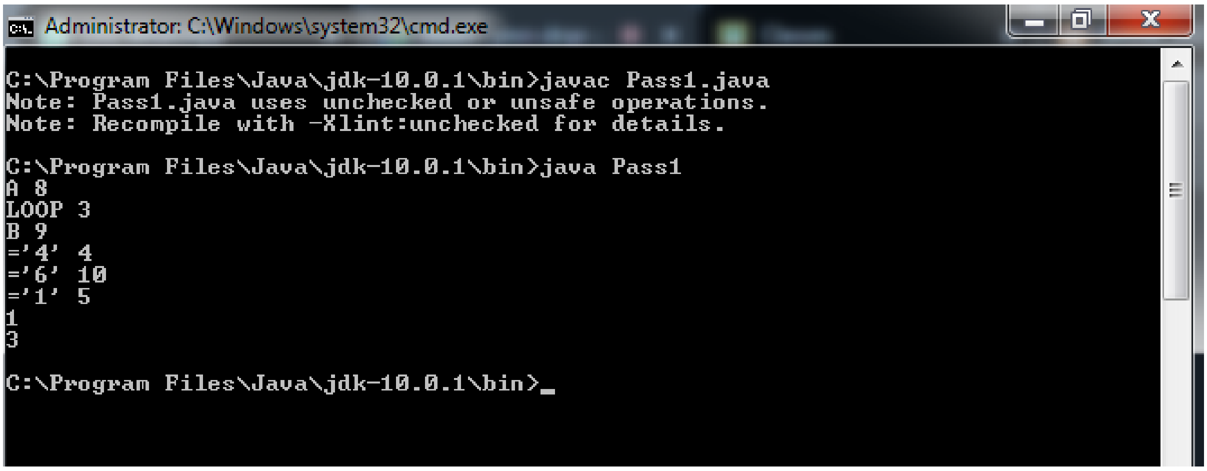
```java
            s = sCurrentLine.split(" |\\,")[2];
            if (s.equals("AREG"))
            {
                bw.write("1\t");
                isOpcode = 1;
            }
            else if (s.equals("BREG"))
            {
                bw.write("2\t");
                isOpcode = 1;
            }
            else if (s.equals("CREG"))
            {
                bw.write("3\t");
                isOpcode = 1;
            }
            else if (s.equals("DREG"))
            {
                bw.write("4\t");
                isOpcode = 1;
            }
            else if (type == "dl")
            {
                bw.write("C\t" + s + "\t");
            }
            else
            {
                symtab.put(s, "");
            }
        }
        if (sCurrentLine.split(" |\\,").length > 3)
        {

            s = sCurrentLine.split(" |\\,")[3];
            if (s.contains("=")) {
                littab.put(s, "");
                bw.write("L\t" + litptr + "\t");
                isOpcode = 1;
                litptr++;
            } else {
```


File Edit Selection View Go Run Terminal Help
Pass1.java - Visual Studio Code [Administrator]
Pass1.java
C: > Program Files > Java > jdk-10.0.1 > bin > Pass1.java
⊗ 0 ⚠ 0
Ln 198, Col 25 Tab Size: 4 UTF-8 CRLF Java

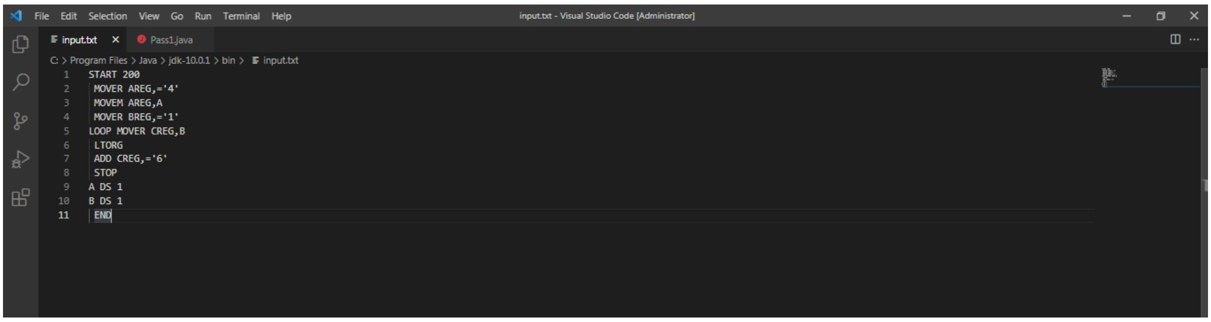
```java
            } else
            {
                symtab.put(s, "");
                bw.write("S\t" + symptr + "\t");
                symptr++;
            }
        }
        bw.write("\n");
        if (mind_the_LC == 0)
            locptr++;
    }
    String f1 = "C:\\Program Files\\Java\\jdk-10.0.1\\bin\\SYMTAB.txt";
    FileWriter fw1 = new FileWriter(f1);
    BufferedWriter bw1 = new BufferedWriter(fw1);
    for (Map.Entry m : symtab.entrySet()) {
        bw1.write(m.getKey() + "\t" + m.getValue() + "\n");
        System.out.println(m.getKey() + " " + m.getValue());
    }

    String f2 = "C:\\Program Files\\Java\\jdk-10.0.1\\bin\\LITTAB.txt";
    FileWriter fw2 = new FileWriter(f2);
    BufferedWriter bw2 = new BufferedWriter(fw2);
    for (Map.Entry m : littab.entrySet()) {
        bw2.write(m.getKey() + "\t" + m.getValue() + "\n");
        System.out.println(m.getKey() + " " + m.getValue());
    }

    String f3 = "C:\\Program Files\\Java\\jdk-10.0.1\\bin\\POOLTAB.txt";
    FileWriter fw3 = new FileWriter(f3);
    BufferedWriter bw3 = new BufferedWriter(fw3);
    for (Integer item : pooltab)
    {
        bw3.write(item + "\n");
        System.out.println(item);
    }
    bw.close();
    bw1.close();
    bw2.close();
    bw3.close();
}
```


File Edit Selection View Go Run Terminal Help
Pass1.java - Visual Studio Code [Administrator]
Pass1.java
C: > Program Files > Java > jdk-10.0.1 > bin > Pass1.java
Ln 198, Col 25 Tab Size: 4 UTF-8 CRLF Java

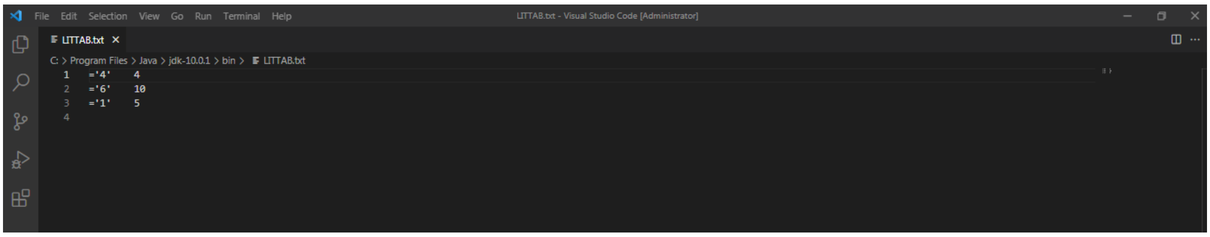
```java
                    if (mind_the_LC == 0)
                        locptr++;
                }
                String f1 = "C:\\Program Files\\Java\\jdk-10.0.1\\bin\\SYMTAB.txt";
                FileWriter fw1 = new FileWriter(f1);
                BufferedWriter bw1 = new BufferedWriter(fw1);
                for (Map.Entry m : symtab.entrySet()) {
                    bw1.write(m.getKey() + "\t" + m.getValue() + "\n");
                    System.out.println(m.getKey() + " " + m.getValue());
                }

                String f2 = "C:\\Program Files\\Java\\jdk-10.0.1\\bin\\LITTAB.txt";
                FileWriter fw2 = new FileWriter(f2);
                BufferedWriter bw2 = new BufferedWriter(fw2);
                for (Map.Entry m : littab.entrySet()) {
                    bw2.write(m.getKey() + "\t" + m.getValue() + "\n");
                    System.out.println(m.getKey() + " " + m.getValue());
                }

                String f3 = "C:\\Program Files\\Java\\jdk-10.0.1\\bin\\POOLTAB.txt";
                FileWriter fw3 = new FileWriter(f3);
                BufferedWriter bw3 = new BufferedWriter(fw3);
                for (Integer item : pooltab)
                {
                    bw3.write(item + "\n");
                    System.out.println(item);
                }
                bw.close();
                bw1.close();
                bw2.close();
                bw3.close();
            }
        catch (IOException e)
            {
                e.printStackTrace();
            }
    }

}
```



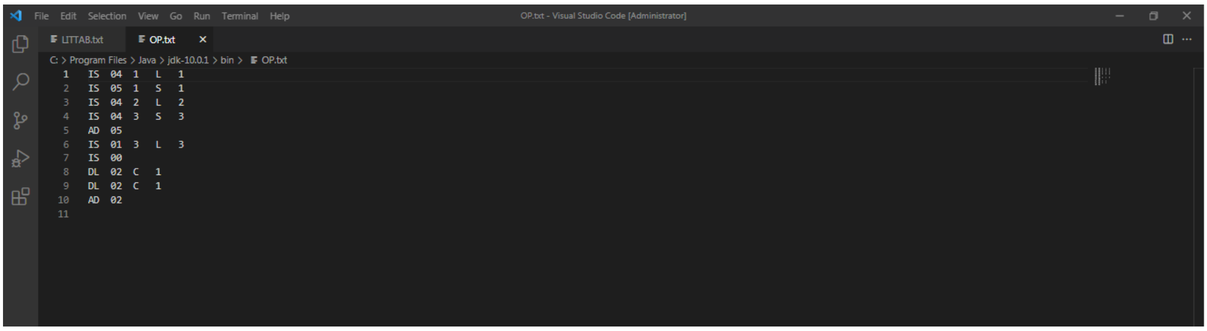
```
C:\Program Files\Java\jdk-10.0.1\bin>javac Pass1.java
Note: Pass1.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.

C:\Program Files\Java\jdk-10.0.1\bin>java Pass1
A 8
LOOP 3
B 9
='4' 4
='6' 10
='1' 5
1
3

C:\Program Files\Java\jdk-10.0.1\bin>_
```


File Edit Selection View Go Run Terminal Help
input.txt - Visual Studio Code [Administrator]
input.txt
Pass1.java
C: > Program Files > Java > jdk-10.0.1 > bin > input.txt

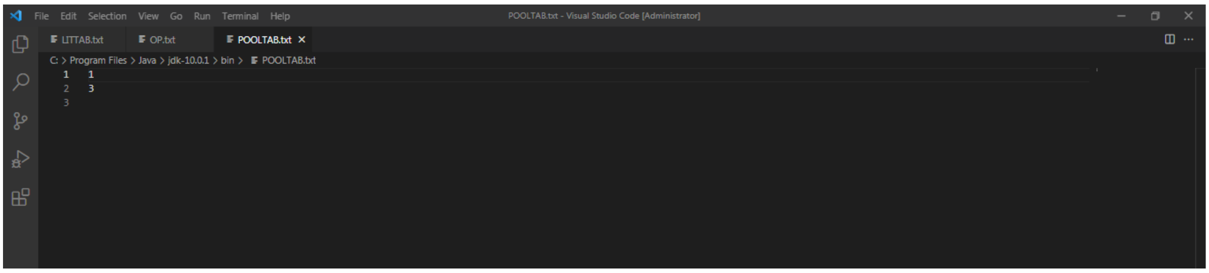
```
START 200
 MOVER AREG,='4'
 MOVEM AREG,A
 MOVER BREG,='1'
LOOP MOVER CREG,B
 LTORG
 ADD CREG,='6'
 STOP
A DS 1
B DS 1
 END
```


File Edit Selection View Go Run Terminal Help
LITTAB.txt - Visual Studio Code [Administrator]
LITTAB.txt
C: > Program Files > Java > jdk-10.0.1 > bin > LITTAB.txt

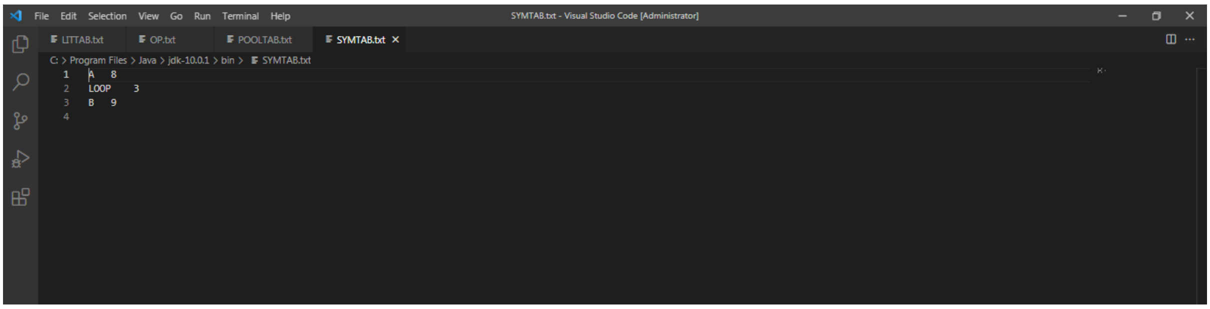
```
='4'    4
='6'    10
='1'    5
```


File Edit Selection View Go Run Terminal Help
OP.txt - Visual Studio Code [Administrator]
LITTAB.txt
OP.txt
C: > Program Files > Java > jdk-10.0.1 > bin > OP.txt

```
IS  04  1   L   1
IS  05  1   S   1
IS  04  2   L   2
IS  04  3   S   3
AD  05
IS  01  3   L   3
IS  00
DL  02  C   1
DL  02  C   1
AD  02
```


File Edit Selection View Go Run Terminal Help
POOLTAB.txt - Visual Studio Code [Administrator]
LITTAB.txt
OP.txt
POOLTAB.txt
C: > Program Files > Java > jdk-10.0.1 > bin > POOLTAB.txt

```
1
3
```


File Edit Selection View Go Run Terminal Help
SYMTAB.txt - Visual Studio Code [Administrator]
LITTAB.txt
OP.txt
POOLTAB.txt
SYMTAB.txt
C: > Program Files > Java > jdk-10.0.1 > bin > SYMTAB.txt

```
A   8
LOOP    3
B   9
```