**ASSIGNMENT A2**

**Roll no. : 31164** **Name: Prathamesh KS**

**Title:**

Pass II of a two pass assembler

**Problem statement:**

Implement Pass-II of two pass assembler for pseudo-machine in Java using object oriented features. The output of assignment-1 (intermediate file and symbol table) should be input for this assignment.

**Objective:**

* Synthesis of the object code.
* Understand the use of data structures required in the design of assembler.

**Outcomes:**

We will be able to learn:

* Parse and tokenize the intermediate code file
* Perform the LC processing
* Generate the target code file
* Demonstrate the use of symbol table, literal table, pooltab

**Theory:**

Assembler is a program which converts assembly language instructions into machine language form. A two pass assembler takes two scans of source code to produce the machine code from assembly language program.

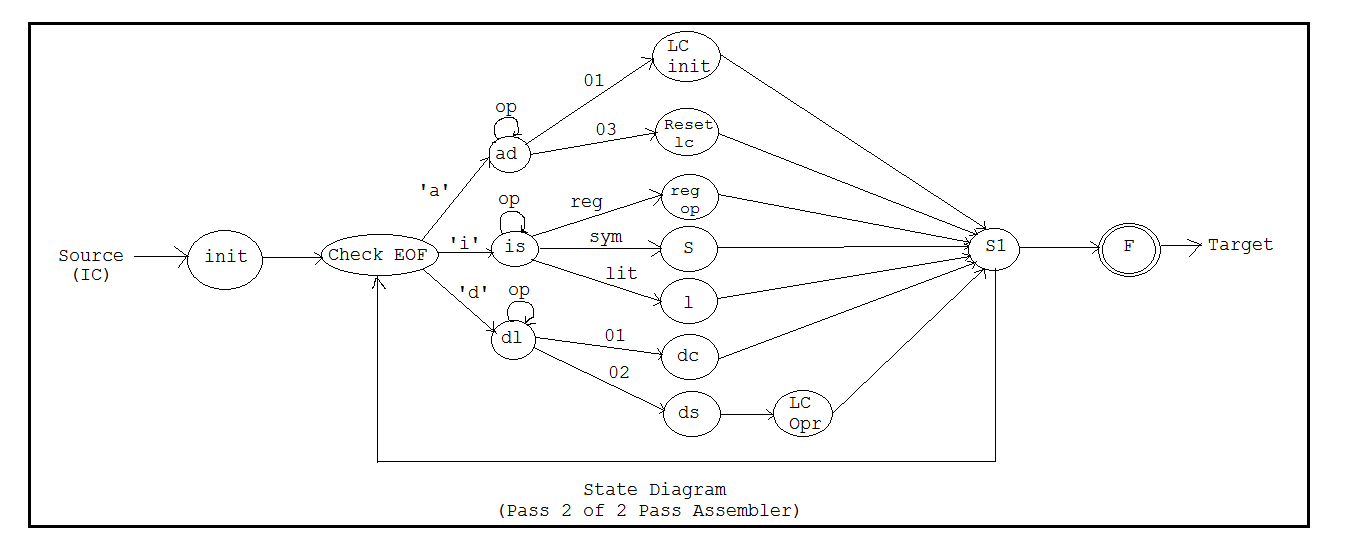
Assembly process consists of following activities:

* Convert mnemonics to their machine language opcode equivalents
* Convert symbolic (i.e. variables, jump labels) operands to their machine addresses
* Translate data constants into internal machine representations
* Output the object program and provide other information required for linker and loader

Pass II Tasks:

* Assemble instructios(generate opcode and look up addresses)
* Generate data values defined by BYTE, WORD
* Perform processing of assembler directives(not done in pass I)
* Write the object program and the assembly listing

**Turing machine/state diagram:**



**Steps to do /algorithm:**

* Read the intermediate code file generated in pass I.
* Search symbol and literal tables to use in machine code generation.
* Generate the machine code

**Conclusion:**

Thus, we successfully implemented pass-II or two pass assembler for pseudo-machine in Java using object oriented features.