

Government Engineering College Thrissur

System Software Lab

Navaneeth D

TCR18CS043

S5, CSE

1st Pass of Two-Pass Assembler

AIM

Implement pass one of a two pass assembler. *

THEORY

Assembler is a program for converting instructions written in low-level assembly code into relocatable machine code and generating along information for the loader.

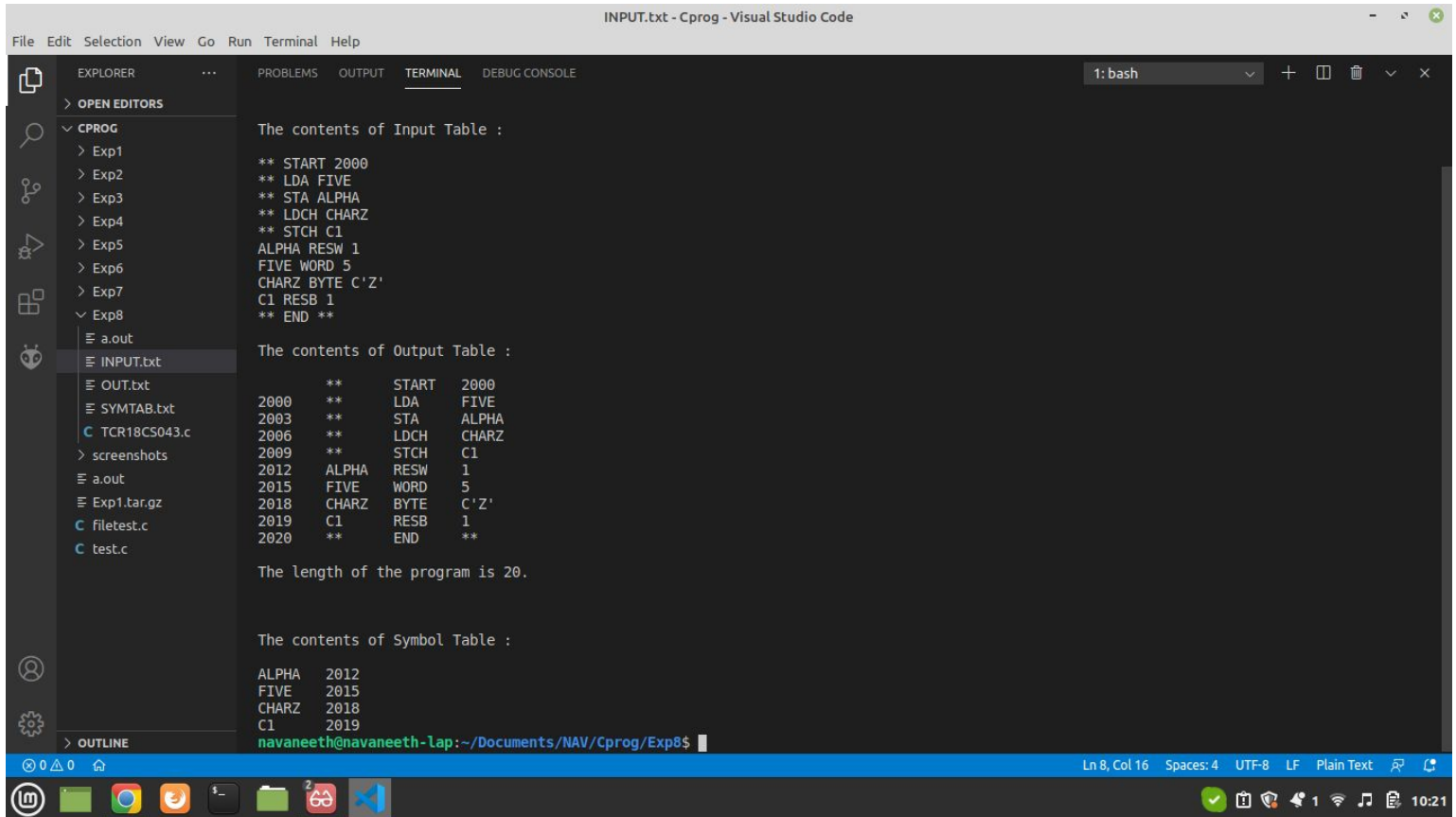
It generates instructions by evaluating the mnemonics (symbols) in the operation field and finding the value of symbols and literals to produce machine code. Now, if an assembler does all this work in one scan then it is called a single pass assembler, otherwise if it does in multiple scans then called multiple pass assembler. Here assembler divide these tasks in two passes:

- **Pass-1:**
 1. Define symbols and literals and remember them in symbol table and literal table respectively.
 2. Keep track of location counter
 3. Process pseudo-operations
- **Pass-2:**
 1. Generate object code by converting symbolic op-code into respective numeric op-code
 2. Generate data for literals and look for values of symbols

RESULT

The 1st pass of a two-pass assembler was implemented with successful output.

Output Screenshots



```
INPUT.txt - Cprog - Visual Studio Code
File Edit Selection View Go Run Terminal Help

EXPLORER
> OPEN EDITORS
CPROG
  > Exp1
  > Exp2
  > Exp3
  > Exp4
  > Exp5
  > Exp6
  > Exp7
  > Exp8
  > a.out
  > INPUT.txt
  > OUT.txt
  > SYMTAB.txt
  > TCR18CS043.c
  > screenshots
  > a.out
  > Exp1.tar.gz
  > filetest.c
  > test.c

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
1: bash

The contents of Input Table :

** START 2000
** LDA FIVE
** STA ALPHA
** LDCH CHARZ
** STCH C1
ALPHA RESW 1
FIVE WORD 5
CHARZ BYTE C'Z'
C1 RESB 1
** END **

The contents of Output Table :

** START 2000
2000 ** LDA FIVE
2003 ** STA ALPHA
2006 ** LDCH CHARZ
2009 ** STCH C1
2012 ALPHA RESW 1
2015 FIVE WORD 5
2018 CHARZ BYTE C'Z'
2019 C1 RESB 1
2020 ** END **

The length of the program is 20.

The contents of Symbol Table :

ALPHA 2012
FIVE 2015
CHARZ 2018
C1 2019
navaneeth@navaneeth-lap:~/Documents/NAV/Cprog/Exp8$
```