System Programming

< Project Phase 1 >

Prepared Date :

April, 29, 2019

Prepared by :

Hagar Usama 4970

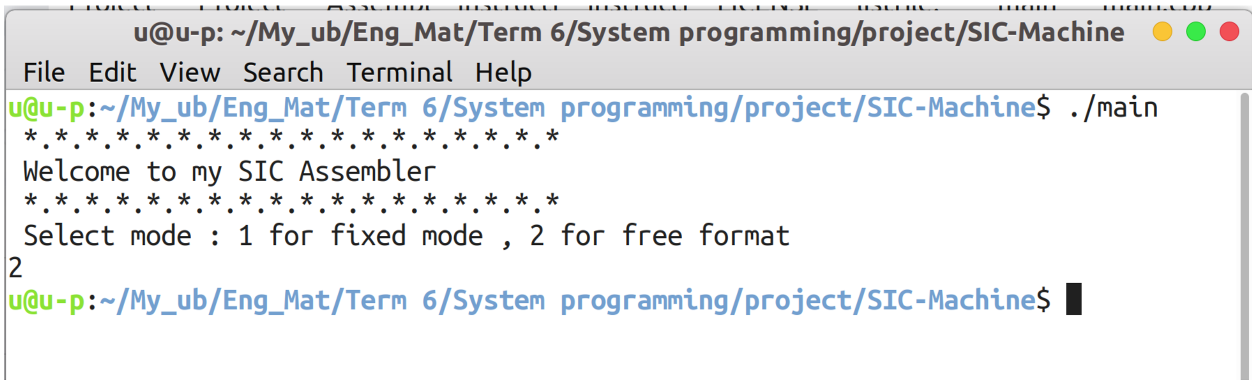
Contents:

Requirements Specification:

To run the program you need a c++ compiler (Recommended : g++)
Also attached exe file (sic_ass.exe) to run the program, however it may not work on windows due to different architecture.

Design:

The design is pretty simple; cmd window to just ask for format of the source file to be assembled, whether it is fixed format or free.


u@u-p: ~/My_ub/Eng_Mat/Term 6/System programming/project/SIC-Machine
File Edit View Search Terminal Help

```
u@u-p:~/My_ub/Eng_Mat/Term 6/System programming/project/SIC-Machine$ ./main
 *.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*
 Welcome to my SIC Assembler
 *.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*
 Select mode : 1 for fixed mode , 2 for free format
2
u@u-p:~/My_ub/Eng_Mat/Term 6/System programming/project/SIC-Machine$ 
```

Main data structures:

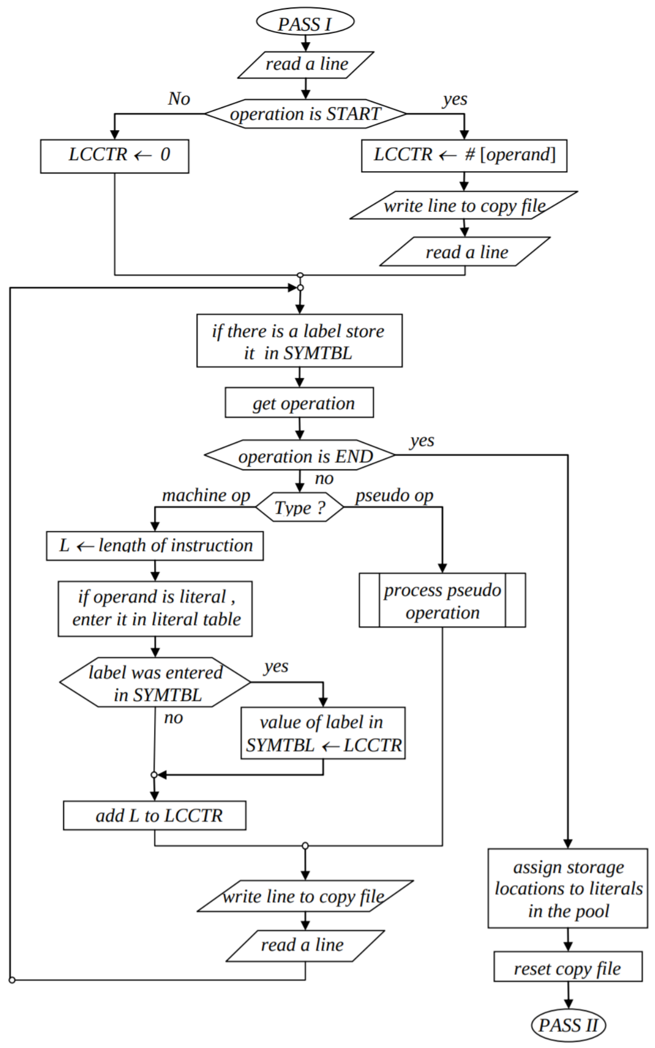
SYMTAB `map<string , int>` : I used a to store symbols and their addresses.
LOCCTR `int` :

- To assign addresses for labels.
- To initialize the beginning address of the program specified by start statement.

Copy File : (`write_line()`) to store a copy of file to be input during Pass II.

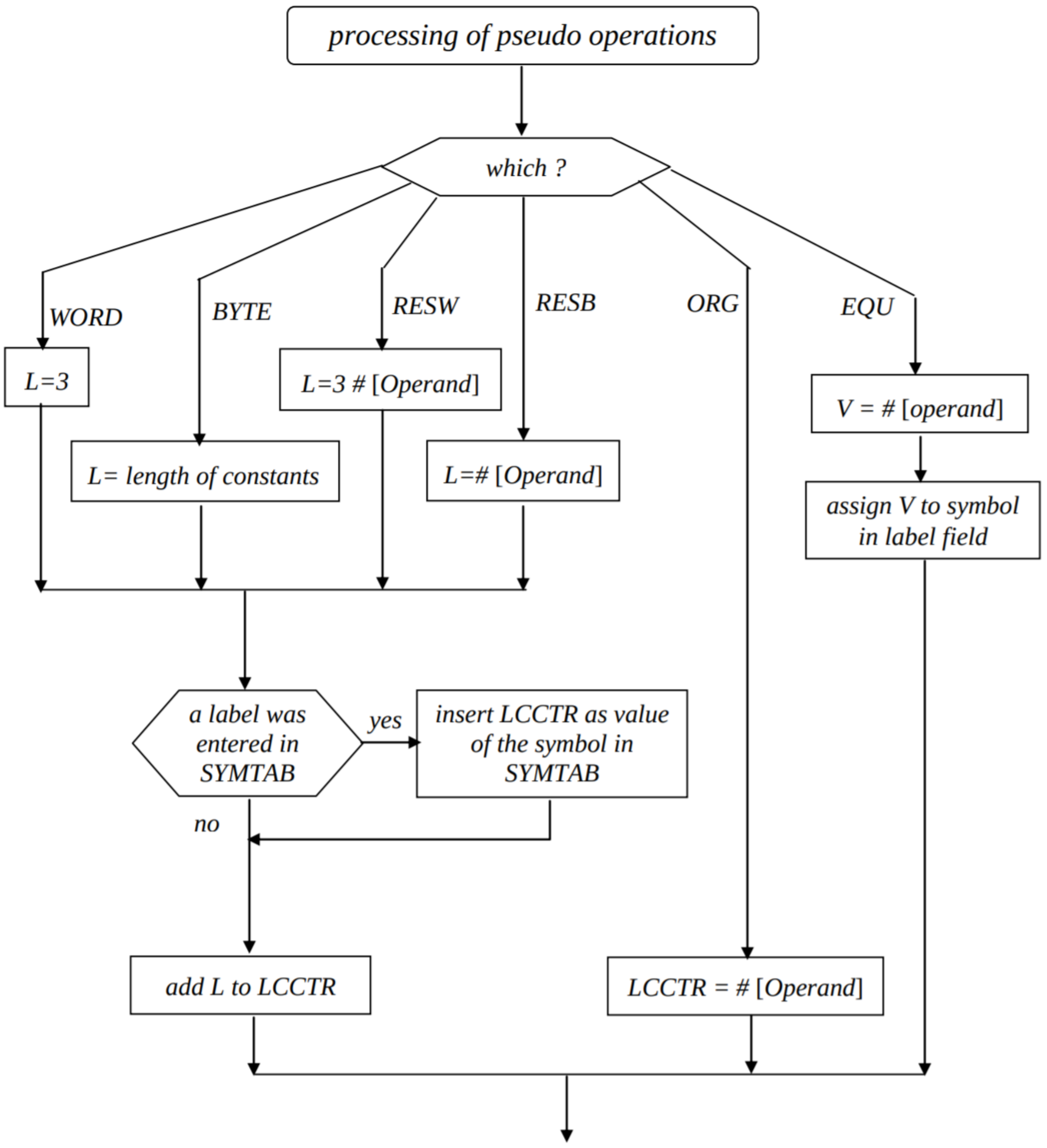
Algorithm description:

I used a to store symbols and their addresses. My reference to algorithm is the pseudo code and flowchart in lecture. The algorithm has nothing about validation, I used regex validation to check the correctness of each statement and partitioned valid ones using regex.



PASS I :   Define Symbols

37   *Systems Programming*



processing of pseudo operations
which ?
WORD
BYTE
RESW
RESB
ORG
EQU
L=3
L=3 # [Operand]
V = # [operand]
L= length of constants
L=# [Operand]
assign V to symbol in label field
a label was entered in SYMTAB
yes
insert LCCTR as value of the symbol in SYMTAB
no
add L to LCCTR
LCCTR = # [Operand]

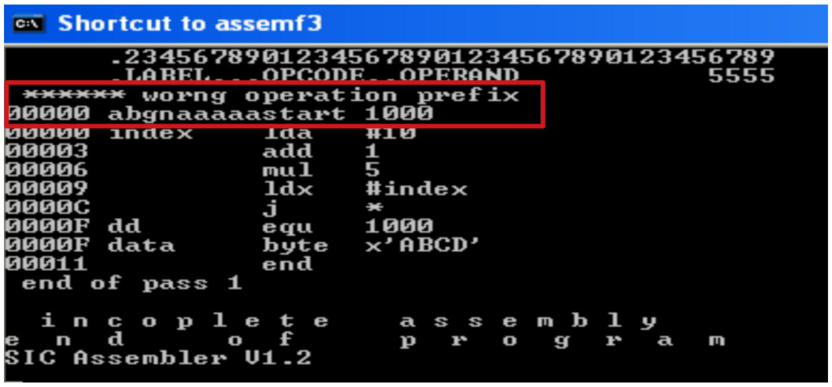
Pass I: Continued

Pass 1:

```
begin
  initialize SYMTAB
  read input line
  if opcode = 'START'
     then begin
            starting_address = #[operand]
            LOCCTR = starting_address
            write line to copy file
            read next line
          end
     else LOCCTR = 0

  while opcode ≠ 'END' do
    begin
      if line is an instruction then                    //  processing of instruction
        begin  if there is a symbol in label field then
                   insert [symbol, LOCCTR]  into SYMTAB
               L = length of instruction
               LOCCTR = LOCCTR + L
               if there is a literal in operand field then insert literal into LITTAB
        end
      else                                              //  processing of directives
        if      opcode = 'ORG' then LOCCTR = #[operand]
        elseif  opcode = 'EQU'  then
                begin V = #[operand]
                      insert [symbol, V]  into SYMTAB
                end
        else
          begin
            if there is a symbol in label field then
                    store [symbol, LOCCTR] in SYMTBL
            if      opcode= 'WORD' then L = 3
            elseif  opcode = 'BYTE'  then L = length of constant in bytes
            elseif  opcode = ' RESW' then L = 3 * #[operand]
            elseif  opcode = 'RESB'  then L = #[operand]
            LOCCTR = LOCCTR + L
          end

      write line to copy file
      read next line

    end while

  assign storage to literals in the pool , if any
  reset copy file
  program length = LOCCTR – starting address
end
```

Assumptions and Notes:

- You should use ' ; ' after operand if you're going to leave a comment.
- Assumed error[11] : if used ' + ' prefix for format 2
- Assumed error[7] : if used position 9 isn't blank (as in our sic_assembler)



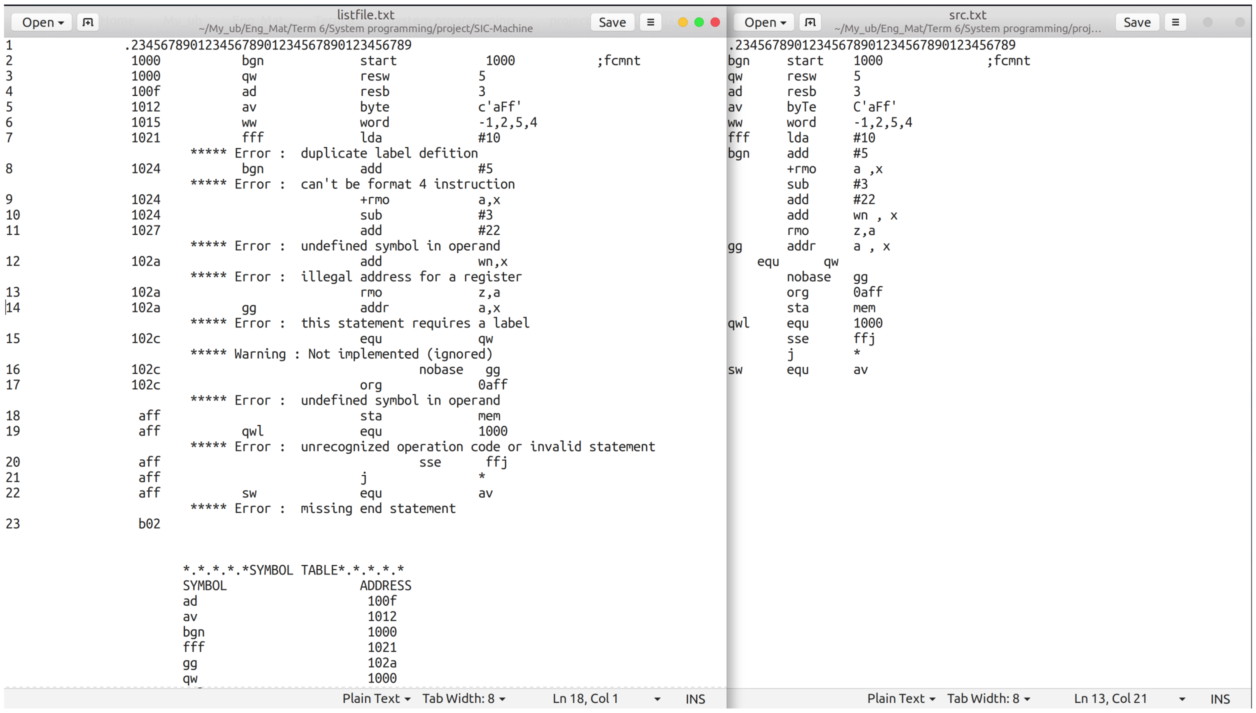
```
      .23456789012345678901234567890123456789
      .LABEL...OPCODE..OPERAND             5555
 ****** worng operation prefix
00000 abgnaaaaastart 1000
00000 index      lda    #10
00003            add    1
00006            mul    5
00009            ldx    #index
0000C            j      *
0000F dd         equ    1000
0000F data       byte   x'ABCD'
00011            end
 end of pass 1

  i n c o p l e t e    a s s e m b l y
e  n  d     o  f       p  r  o  g  r  a  m
SIC Assembler V1.2
```

- Any errors not mentioned in the project will be considered error [8] :
- `"***** Error :  unrecognized operation code or invalid statement"`

- You may find many redundant or unused code, this will be used later for pass II.
- I am using Ubuntu OS & Ubuntu mate OS
- In case something went wrong 'unexpected', you can easily define the statement that caused so from listfile.txt :
  - The line number that may cause an error is the line next to last line. (ie: if core dumped and last line in listfile is 5 then, we have a problem in line 6 in src.txt)

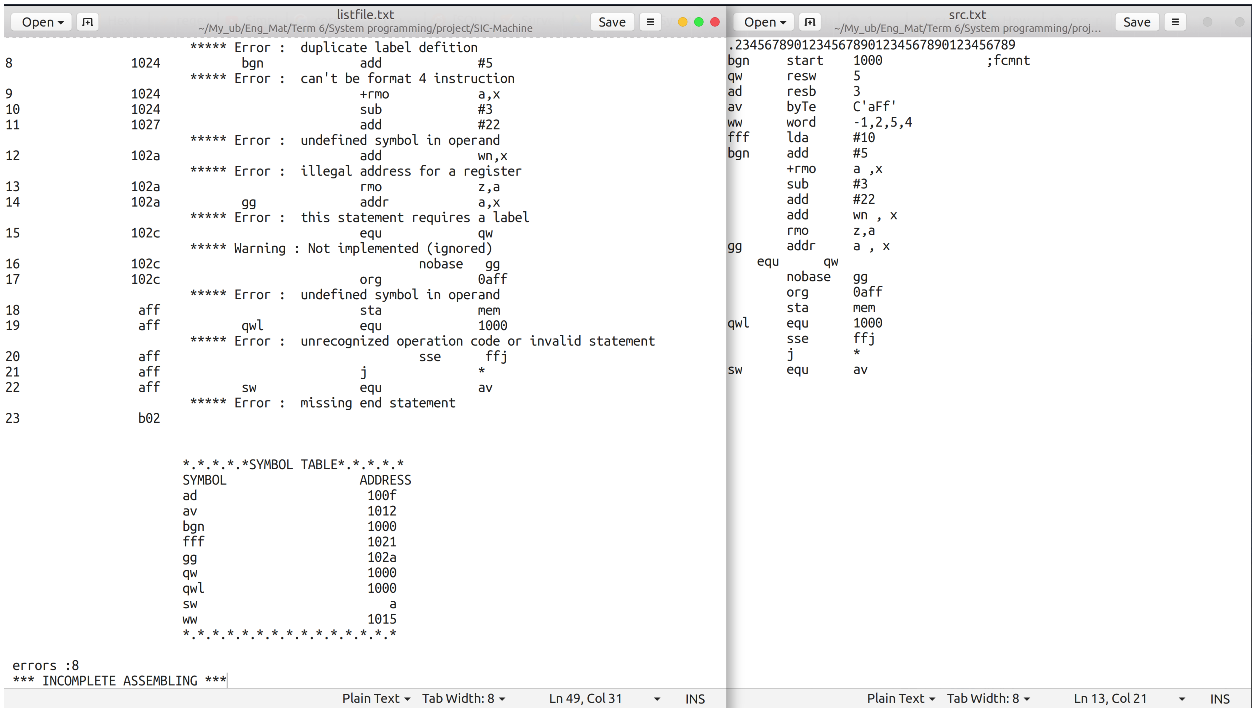
Sample Runs:

Free-format :


Open
listfile.txt
~/My_ub/Eng_Mat/Term 6/System programming/project/SIC-Machine
Save
Plain Text
Tab Width: 8
Ln 18, Col 1
INS
src.txt
~/My_ub/Eng_Mat/Term 6/System programming/proj...
Ln 13, Col 21

```
1               .23456789012345678901234567890123456789
2                1000          bgn          start          1000          ;fcmnt
3                1000          qw           resw           5
4                100f          ad           resb           3
5                1012          av           byte           c'aFf'
6                1015          ww           word           -1,2,5,4
7                1021          fff          lda            #10
         ***** Error :  duplicate label defition
8                1024          bgn          add            #5
         ***** Error :  can't be format 4 instruction
9                1024                       +rmo           a,x
10               1024                       sub            #3
11               1027                       add            #22
         ***** Error :  undefined symbol in operand
12               102a                       add            wn,x
         ***** Error :  illegal address for a register
13               102a                       rmo            z,a
14               102a          gg           addr           a,x
         ***** Error :  this statement requires a label
15               102c                       equ            qw
         ***** Warning : Not implemented (ignored)
16               102c                              nobase   gg
17               102c                       org            0aff
         ***** Error :  undefined symbol in operand
18                aff                       sta            mem
19                aff          qwl          equ            1000
         ***** Error :  unrecognized operation code or invalid statement
20                aff                              sse      ffj
21                aff                       j              *
22                aff          sw           equ            av
         ***** Error :  missing end statement
23                b02


                     *.*.*.*.*SYMBOL TABLE*.*.*.*.*
                     SYMBOL                 ADDRESS
                     ad                      100f
                     av                      1012
                     bgn                     1000
                     fff                     1021
                     gg                      102a
                     qw                      1000
```

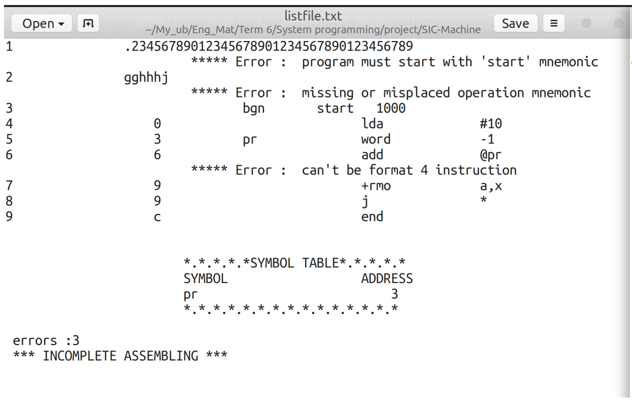
```
.23456789012345678901234567890123456789
bgn     start     1000                 ;fcmnt
qw      resw      5
ad      resb      3
av      byTe      C'aFf'
ww      word      -1,2,5,4
fff     lda       #10
bgn     add       #5
        +rmo      a ,x
        sub       #3
        add       #22
        add       wn , x
        rmo       z,a
gg      addr      a , x
    equ        qw
        nobase    gg
        org       0aff
        sta       mem
qwl     equ       1000
        sse       ffj
        j         *
sw      equ       av
```


Open
listfile.txt
~/My_ub/Eng_Mat/Term 6/System programming/project/SIC-Machine
Save
Plain Text
Tab Width: 8
Ln 49, Col 31
INS
src.txt
~/My_ub/Eng_Mat/Term 6/System programming/proj...
Ln 13, Col 21

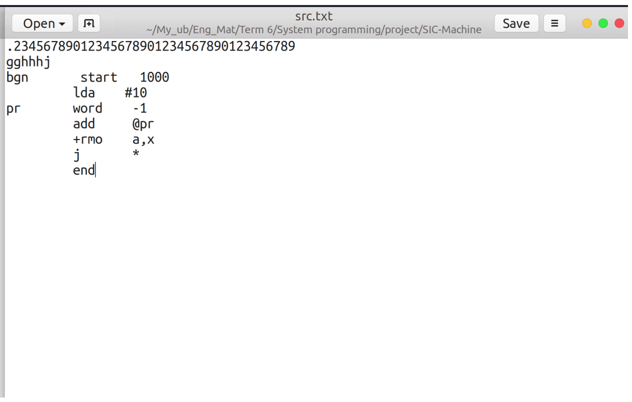
```
         ***** Error :  duplicate label defition
8                1024          bgn          add            #5
         ***** Error :  can't be format 4 instruction
9                1024                       +rmo           a,x
10               1024                       sub            #3
11               1027                       add            #22
         ***** Error :  undefined symbol in operand
12               102a                       add            wn,x
         ***** Error :  illegal address for a register
13               102a                       rmo            z,a
14               102a          gg           addr           a,x
         ***** Error :  this statement requires a label
15               102c                       equ            qw
         ***** Warning : Not implemented (ignored)
16               102c                              nobase   gg
17               102c                       org            0aff
         ***** Error :  undefined symbol in operand
18                aff                       sta            mem
19                aff          qwl          equ            1000
         ***** Error :  unrecognized operation code or invalid statement
20                aff                              sse      ffj
21                aff                       j              *
22                aff          sw           equ            av
         ***** Error :  missing end statement
23                b02


                     *.*.*.*.*SYMBOL TABLE*.*.*.*.*
                     SYMBOL                 ADDRESS
                     ad                      100f
                     av                      1012
                     bgn                     1000
                     fff                     1021
                     gg                      102a
                     qw                      1000
                     qwl                     1000
                     sw                         a
                     ww                      1015
                     *.*.*.*.*.*.*.*.*.*.*.*.*.*.*.*


 errors :8
 *** INCOMPLETE ASSEMBLING ***
```

```
.23456789012345678901234567890123456789
bgn     start     1000                 ;fcmnt
qw      resw      5
ad      resb      3
av      byTe      C'aFf'
ww      word      -1,2,5,4
fff     lda       #10
bgn     add       #5
        +rmo      a ,x
        sub       #3
        add       #22
        add       wn , x
        rmo       z,a
gg      addr      a , x
    equ        qw
        nobase    gg
        org       0aff
        sta       mem
qwl     equ       1000
        sse       ffj
        j         *
sw      equ       av
```

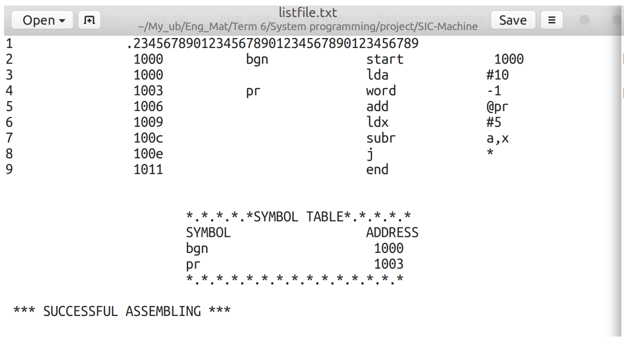
Fixed format :


Open
listfile.txt
~/My_ub/Eng_Mat/Term 6/System programming/project/SIC-Machine
Save

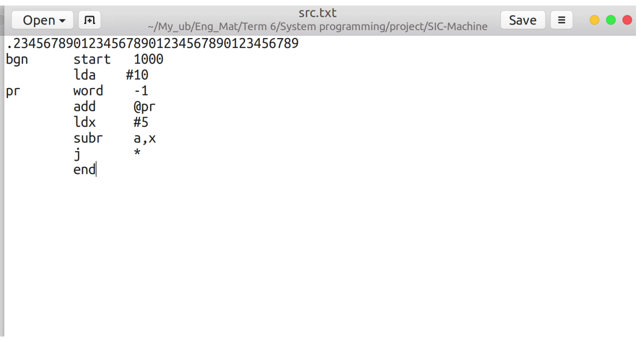
```
1               .234567890123456789012345678901234567890123456789
                         ***** Error :  program must start with 'start' mnemonic
2               gghhhj
                         ***** Error :  missing or misplaced operation mnemonic
3                               bgn       start   1000
4                   0                           lda             #10
5                   3           pr              word            -1
6                   6                           add             @pr
                         ***** Error :  can't be format 4 instruction
7                   9                           +rmo            a,x
8                   9                           j               *
9                   c                           end


                        *.*.*.*.*SYMBOL TABLE*.*.*.*.*
                        SYMBOL                  ADDRESS
                        pr                          3
                        *.*.*.*.*.*.*.*.*.*.*.*.*.*.*


 errors :3
 *** INCOMPLETE ASSEMBLING ***
```


Open
src.txt
~/My_ub/Eng_Mat/Term 6/System programming/project/SIC-Machine
Save

```
.234567890123456789012345678901234567890123456789
gghhhj
bgn       start   1000
         lda    #10
pr       word    -1
         add     @pr
         +rmo    a,x
         j       *
         end
```


Open
listfile.txt
~/My_ub/Eng_Mat/Term 6/System programming/project/SIC-Machine
Save

```
1               .234567890123456789012345678901234567890123456789
2                1000           bgn             start            1000
3                1000                           lda             #10
4                1003           pr              word            -1
5                1006                           add             @pr
6                1009                           ldx             #5
7                100c                           subr            a,x
8                100e                           j               *
9                1011                           end


                        *.*.*.*.*SYMBOL TABLE*.*.*.*.*
                        SYMBOL                  ADDRESS
                        bgn                      1000
                        pr                       1003
                        *.*.*.*.*.*.*.*.*.*.*.*.*.*.*

 *** SUCCESSFUL ASSEMBLING ***
```


Open
src.txt
~/My_ub/Eng_Mat/Term 6/System programming/project/SIC-Machine
Save

```
.234567890123456789012345678901234567890123456789
bgn      start   1000
         lda    #10
pr       word    -1
         add     @pr
         ldx     #5
         subr    a,x
         j       *
         end
```

See this video : project wasn't finished yet