# System Programming
# < Project Phase 1 & 2 >

Prepared Date :

May 19, 2019

Prepared by :

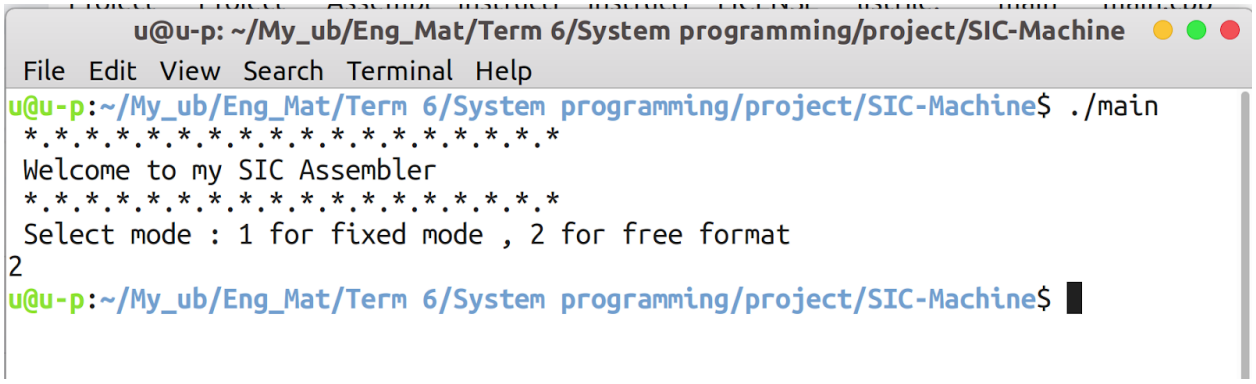
Hagar Usama  4970

# Requirements Specification:

To run the program you need a c++ compiler (Recommended : g++ )
Also attached exe file ( sic_ass.exe [ for phase 1 ] ) 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. Or GUI design.




# Main data structures:

**SYMTAB** `map<string , int>` : I used a to store symbols and their addresses.
**OPTAB** `map<string , int>` : I used a to get opcode for each instruction.

**LOCCTR** `int` :

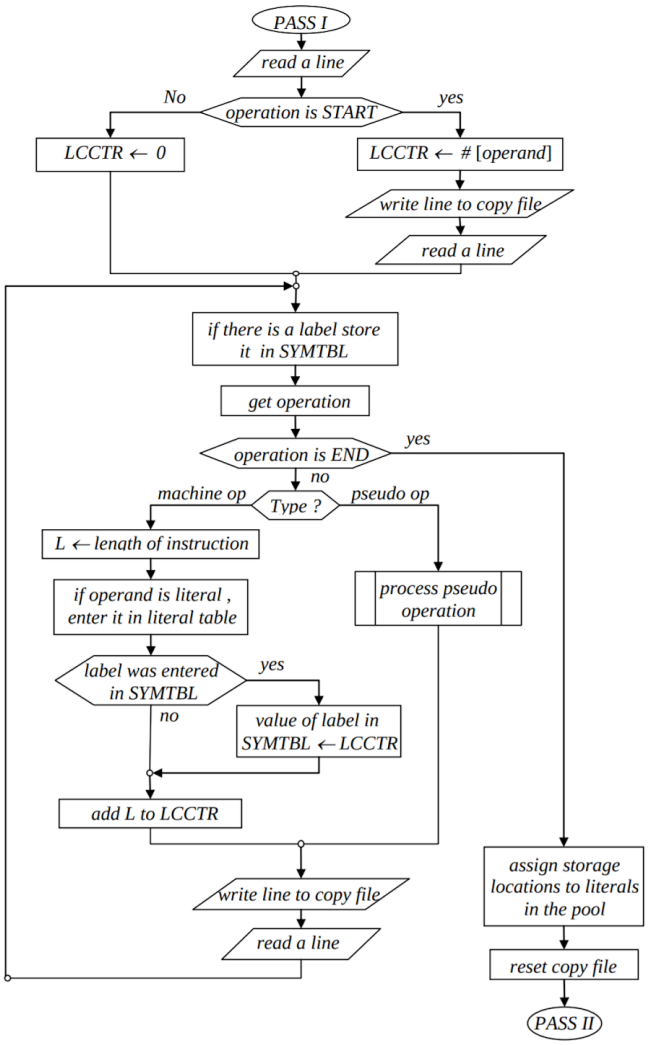
- To assign addresses for labels.
- To initialize the beginning address of the program specified by start statement.

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

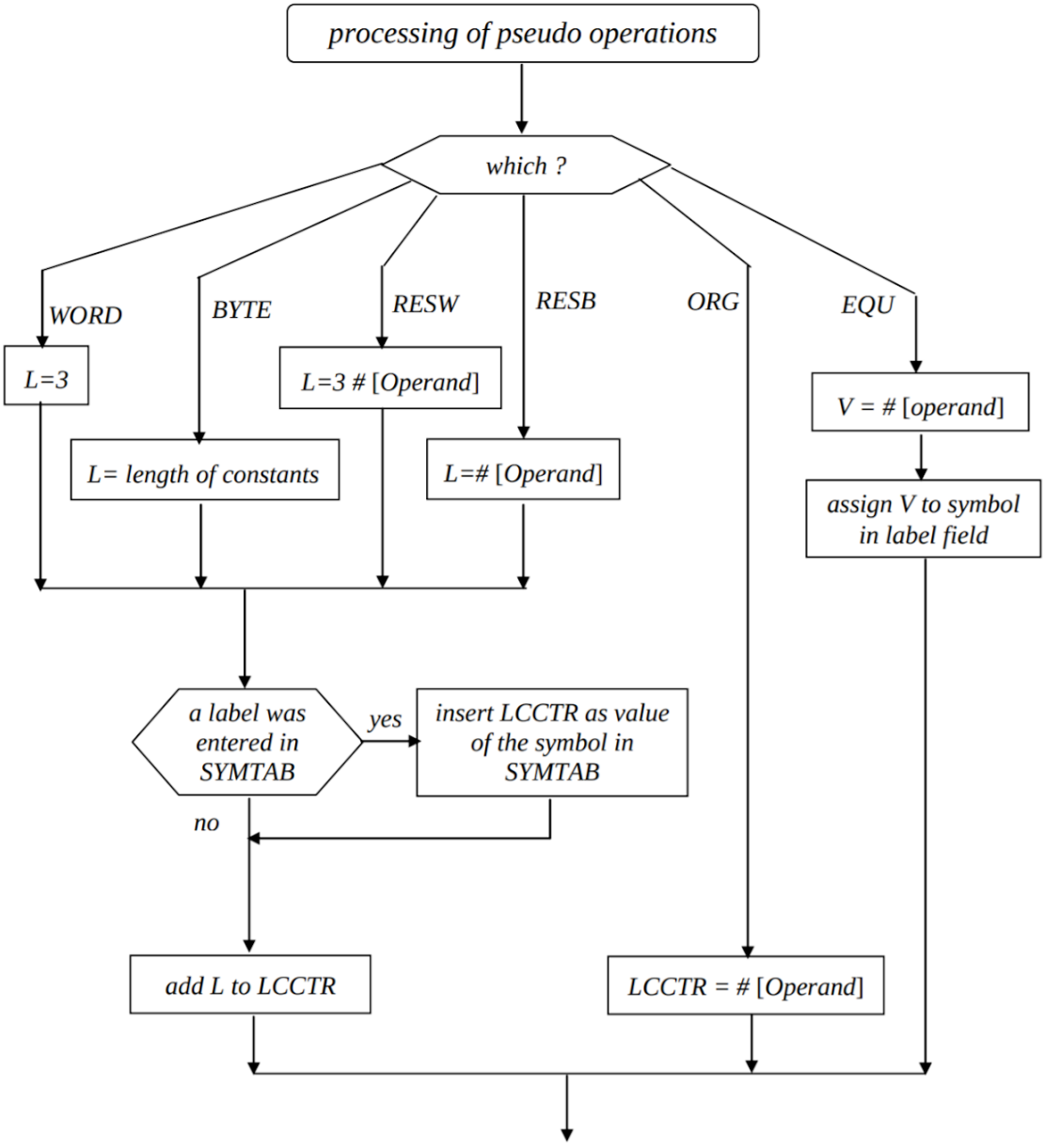
**Object File** : ( `objectize()` ) prints Header record, Text record and End record to be input during Pass II

# 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




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
```

# Pass 2:

***begin***

read input *line*
***if*** *opcode* = '*START*'
***then*** ***begin***
write listing line
read next *line*
***end***

write *Header* record to object program
initialize first *text* record

***while*** *opcode* ≠ '***END***' ***do***
***begin***

***if*** *line* is an instruction ***then*** // processing of instruction

***begin*** **replace mnemonic operation by the binary equivalent**
**process operand according to format**
**include assembled Instruction in object code**
***end***

***else*** // processing of directives

***if*** *opcode*='***WORD***' ***or*** *opcode* ='***BYTE***' ***then***
***begin*** convert constants to internal form
**insert constants in object code**
***end***
***elseif*** *opcode* ='***BASE***' ***then*** evaluate operand for relative addressing
***else if*** *opcode* ='***NOBASE***' ***then*** indicate that base register is not available

list line
**read next *line from* copy file**

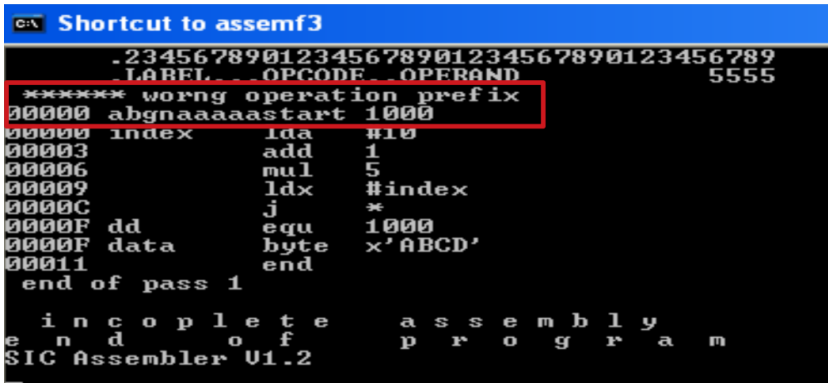
***end while***

*insert literals in object code*

***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)



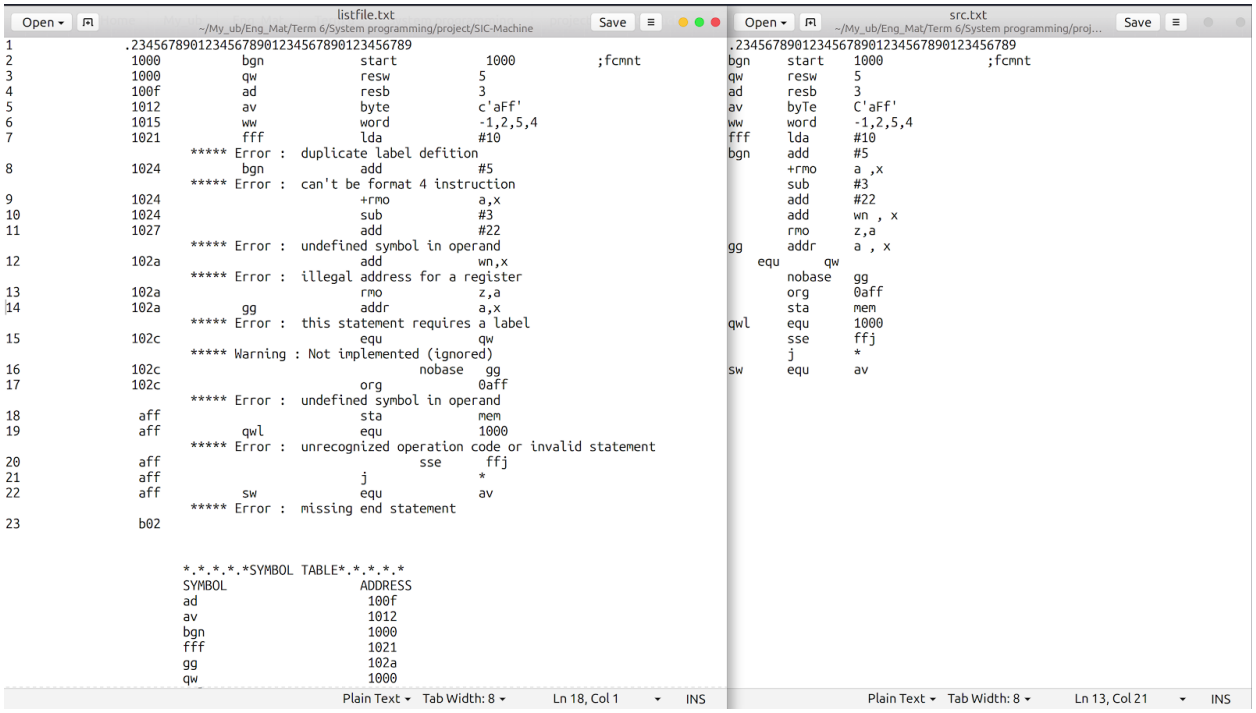

- Any errors not mentioned in the project will be considered error [8] :
- `"***** Error : unrecognized operation code or invalid statement"`

- You may find many redundant or unused code, this will be used later for pass II.
- You cannot using 'resb' , 'resw' , 'org' following each other (just for proper object code)
- Simple expression is implemented as in : equ label + 1000
- You can either run the program in shell mode or GUI mode
- 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 :



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

```
1               .2345678901234567890123456789012345678901234567890123456789
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
```

Plain Text  Tab Width: 8  Ln 18, Col 1  INS

src.txt
~/My_ub/Eng_Mat/Term 6/System programming/proj...

```
.2345678901234567890123456789012345678901234567890123456789
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
```

Plain Text  Tab Width: 8  Ln 13, Col 21  INS

listfile.txt

```
                        ***** 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 ***
```

Plain Text  Tab Width: 8  Ln 49, Col 31  INS

src.txt

```
.2345678901234567890123456789012345678901234567890123456789
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
```

Plain Text  Tab Width: 8  Ln 13, Col 21  INS

## Fixed format :

listfile.txt

```
1               .2345678901234567890123456789012345678901234567890123456789
                        ***** 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 ***
```

src.txt

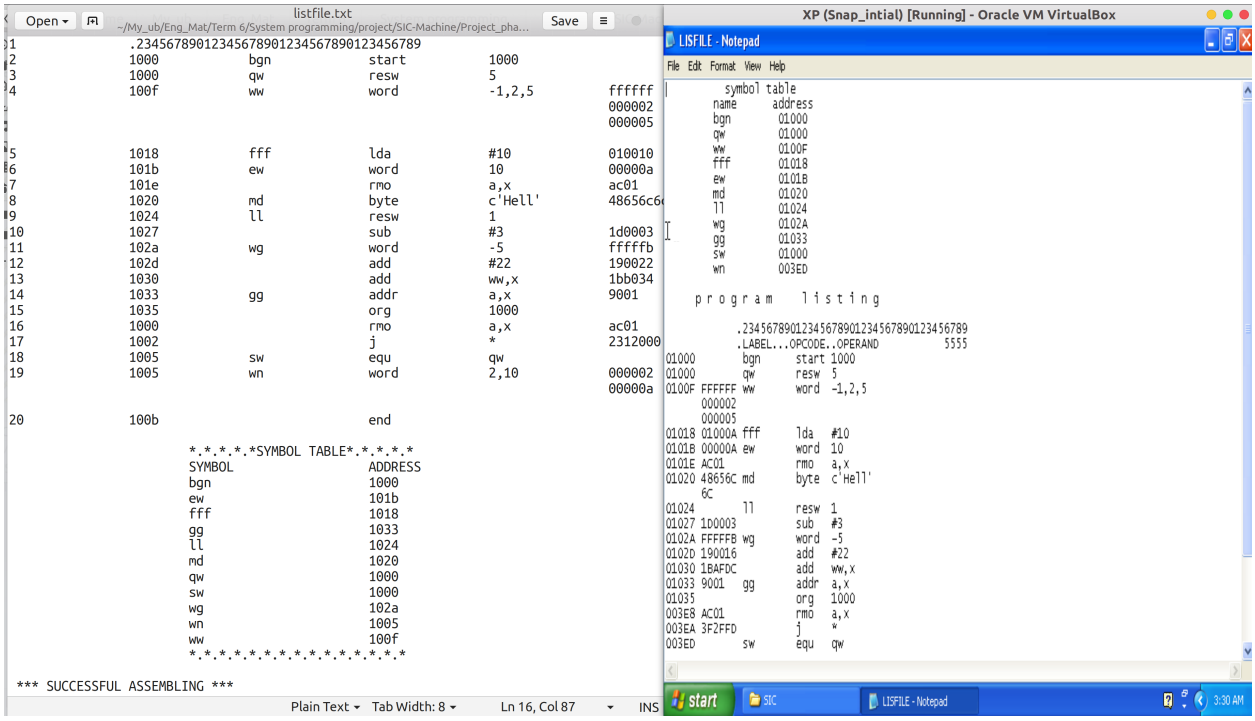
```
.2345678901234567890123456789012345678901234567890123456789
gghhhj
bgn       start   1000
          lda     #10
pr        word    -1
          add     @pr
          +rmo    a,x
          j       *
          end
```

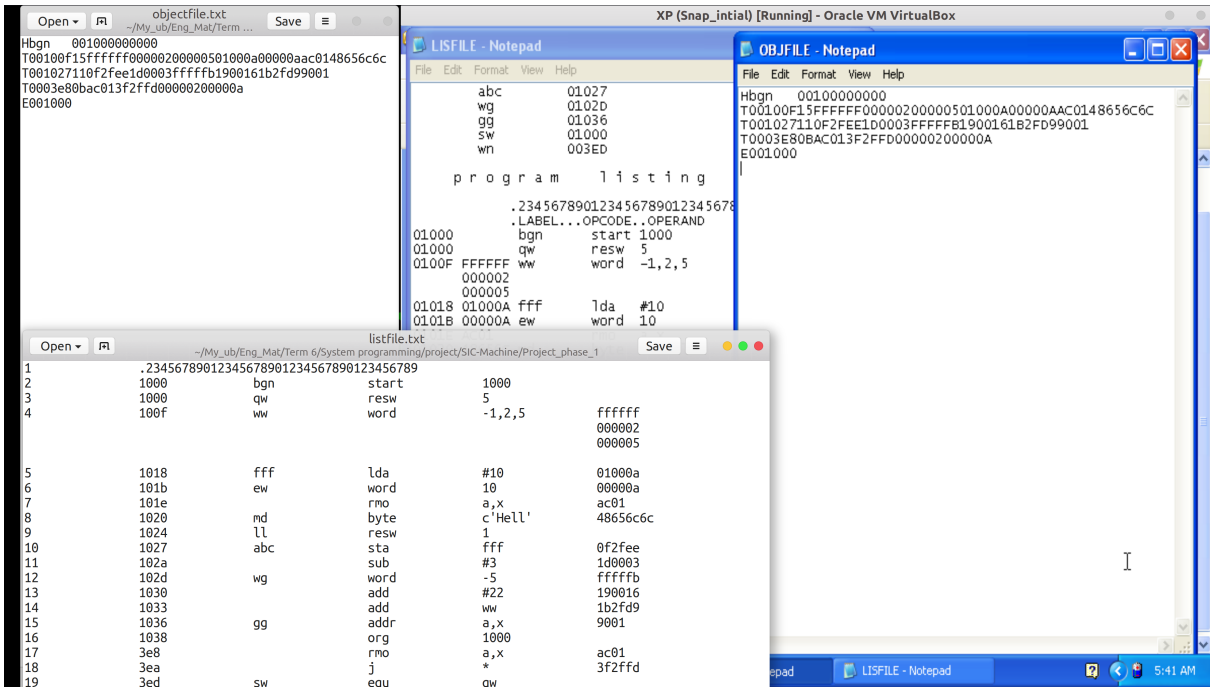
listfile.txt

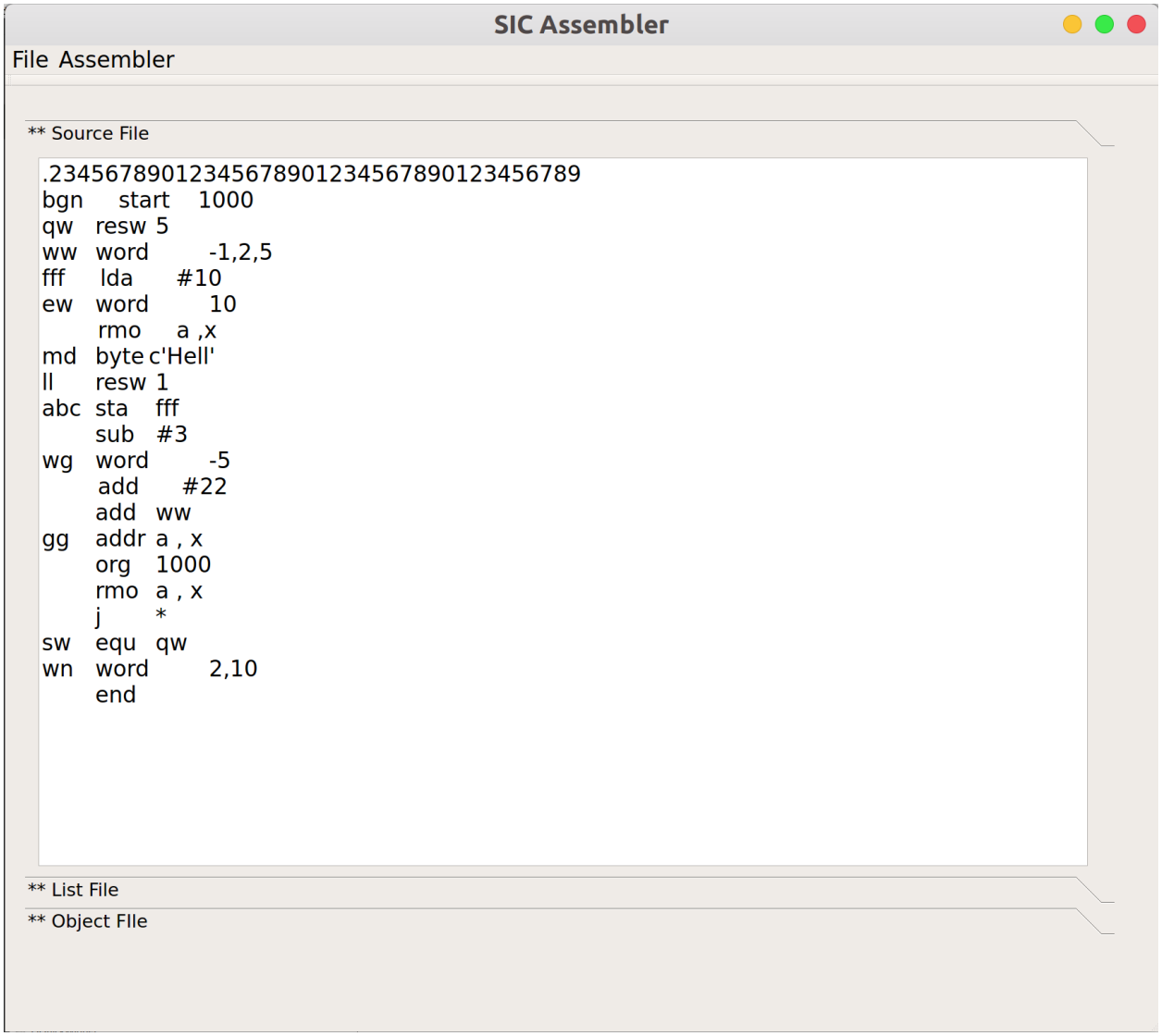
```
1               .2345678901234567890123456789012345678901234567890123456789
2               1000      bgn           start            1000
3               1000                    lda             #10
4               1003      pr            word            -1
5               1006                    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 ***
```

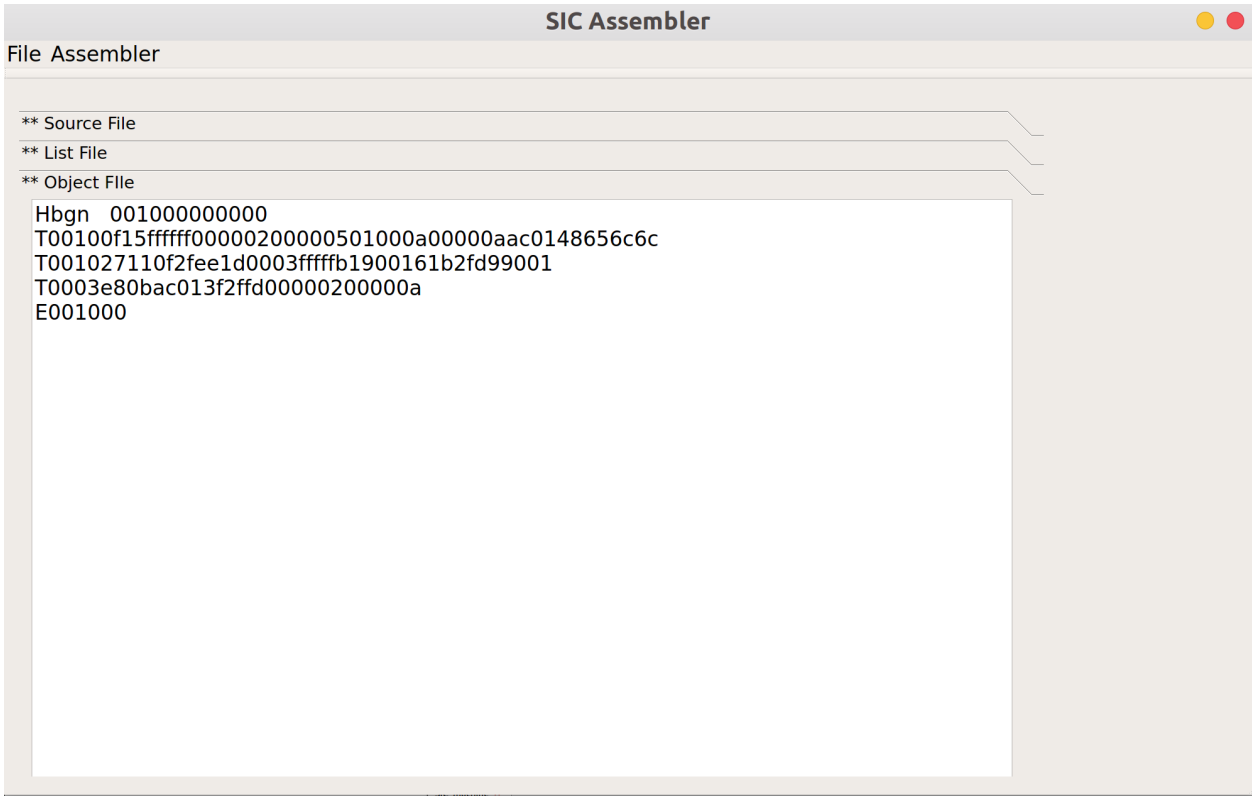
src.txt

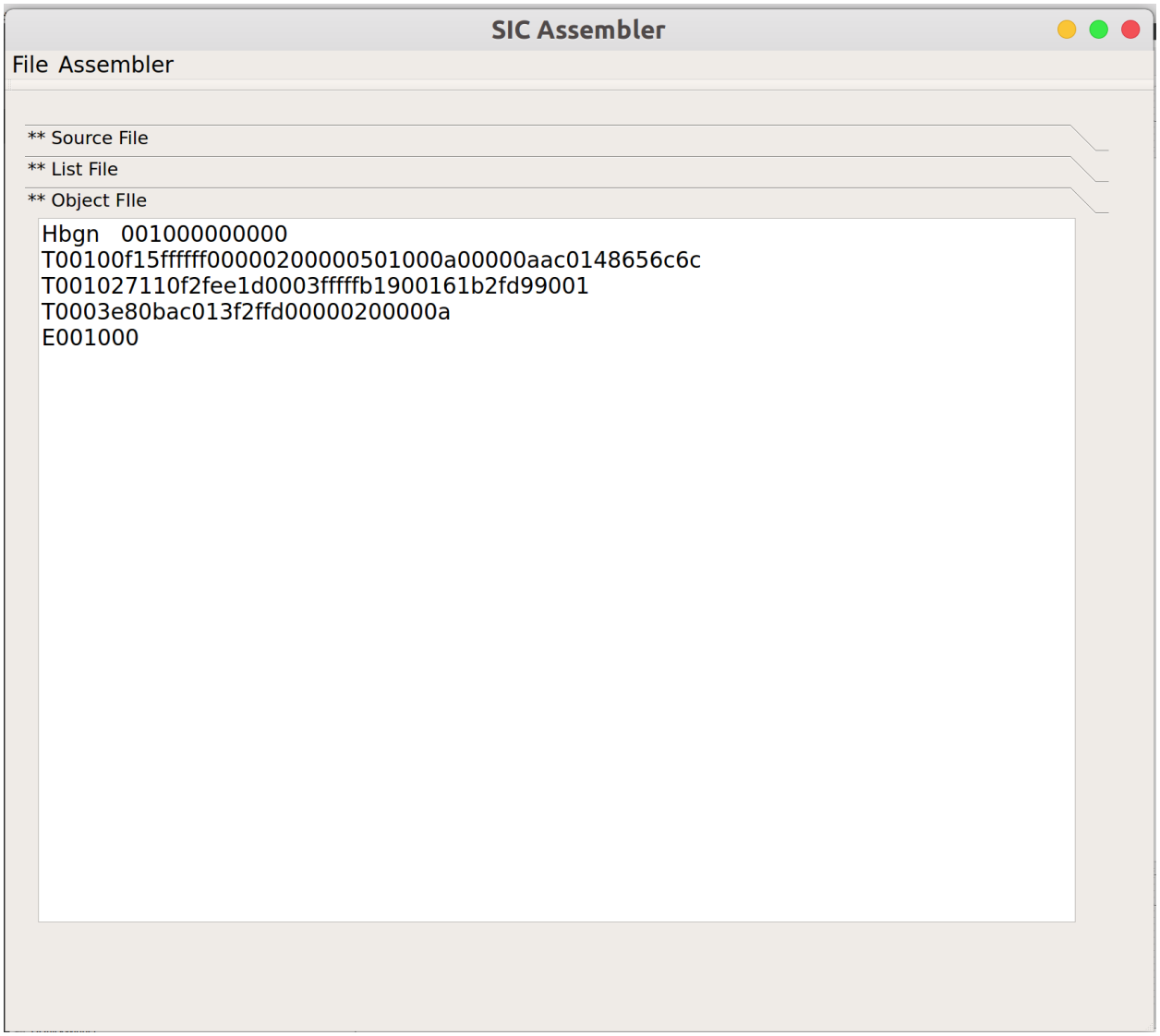
```
.2345678901234567890123456789012345678901234567890123456789
bgn       start   1000
          lda     #10
pr        word    -1
          add     @pr
          ldx     #5
          subr    a,x
          j       *
          end
```


listfile.txt
~/My_ub/Eng_Mat/Term 6/System programming/project/SIC-Machine/Project_pha...
Open
Save
1 .2345678901234567890123456789012345678 9
2 1000 bgn start 1000
3 1000 qw resw 5
4 100f ww word -1,2,5 ffffff
000002
000005
5 1018 fff lda #10 010010
6 101b ew word 10 00000a
7 101e rmo a,x ac01
8 1020 md byte c'Hell' 48656c6
9 1024 ll resw 1
10 1027 sub #3 1d0003
11 102a wg word -5 fffffb
12 102d add #22 190022
13 1030 add ww,x 1bb034
14 1033 gg addr a,x 9001
15 1035 org 1000
16 1000 rmo a,x ac01
17 1002 j * 2312000
18 1005 sw equ qw
19 1005 wn word 2,10 000002
00000a
20 100b end
*.*.*.*.*SYMBOL TABLE*.*.*.*.*
SYMBOL ADDRESS
bgn 1000
ew 101b
fff 1018
gg 1033
ll 1024
md 1020
qw 1000
sw 1000
wg 102a
wn 1005
ww 100f
*.*.*.*.*.*.*.*.*.*.*.*.*.*
*** SUCCESSFUL ASSEMBLING ***
Plain Text
Tab Width: 8
Ln 16, Col 87
INS
XP (Snap_intial) [Running] - Oracle VM VirtualBox
LISFILE - Notepad
File Edit Format View Help
symbol table
name address
bgn 01000
qw 01000
ww 0100F
fff 01018
ew 0101B
md 01020
ll 01024
wg 0102A
gg 01033
sw 01000
wn 003ED
p r o g r a m l i s t i n g
.2345678901234567890123456789012345678 9
.LABEL...OPCODE..OPERAND 5555
01000 bgn start 1000
01000 qw resw 5
0100F FFFFFF ww word -1,2,5
000002
000005
01018 01000A fff lda #10
0101B 00000A ew word 10
0101E AC01 rmo a,x
01020 48656C md byte c'Hell'
6C
01024 ll resw 1
01027 1D0003 sub #3
0102A FFFFFB wg word -5
0102D 190016 add #22
01030 1BAFDC add ww,x
01033 9001 gg addr a,x
01035 org 1000
003E8 AC01 rmo a,x
003EA 3F2FFD j *
003ED sw equ qw
start
SIC
LISFILE - Notepad
3:30 AM


objectfile.txt
~/My_ub/Eng_Mat/Term ...
Open
Save
Hbgn 001000000000
T00100f15ffffff000002000005010000a00000aac0148656c6c
T001027110f2fee1d0003fffffb1900161b2fd99001
T0003e80bac013f2ffd00000200000a
E001000
XP (Snap_intial) [Running] - Oracle VM VirtualBox
LISFILE - Notepad
File Edit Format View Help
abc 01027
wg 0102D
gg 01036
sw 01000
wn 003ED
p r o g r a m l i s t i n g
.2345678901234567890123456789012345678
.LABEL...OPCODE..OPERAND
01000 bgn start 1000
01000 qw resw 5
0100F FFFFFF ww word -1,2,5
000002
000005
01018 01000A fff lda #10
0101B 00000A ew word 10
OBJFILE - Notepad
File Edit Format View Help
Hbgn 001000000000
T00100F15FFFFFF000002000005010000A00000AAC0148656C6C
T001027110F2FEE1D0003FFFFFB1900161B2FD99001
T0003E80BAC013F2FFD00000200000A
E001000
listfile.txt
~/My_ub/Eng_Mat/Term 6/System programming/project/SIC-Machine/Project_phase_1
Open
Save
1 .2345678901234567890123456789012345678 9
2 1000 bgn start 1000
3 1000 qw resw 5
4 100f ww word -1,2,5 ffffff
000002
000005
5 1018 fff lda #10 01000a
6 101b ew word 10 00000a
7 101e rmo a,x ac01
8 1020 md byte c'Hell' 48656c6c
9 1024 ll resw 1
10 1027 abc sta fff 0f2fee
11 102a sub #3 1d0003
12 102d wg word -5 fffffb
13 1030 add #22 190016
14 1033 add ww 1b2fd9
15 1036 gg addr a,x 9001
16 1038 org 1000
17 3e8 rmo a,x ac01
18 3ea j * 3f2ffd
19 3ed sw equ qw
LISFILE - Notepad
5:41 AM


SIC Assembler
File Assembler
** Source File
.23456789012345678901234567890123456789
bgn start 1000
qw resw 5
ww word -1,2,5
fff lda #10
ew word 10
rmo a ,x
md byte c'Hell'
ll resw 1
abc sta fff
sub #3
wg word -5
add #22
add ww
gg addr a , x
org 1000
rmo a , x
j *
sw equ qw
wn word 2,10
end
** List File
** Object FIle


SIC Assembler
File Assembler
** Source File
** List File
** Object FIle
Hbgn 001000000000
T00100f15ffffff0000020000050100a00000aac0148656c6c
T001027110f2fee1d0003fffffb1900161b2fd99001
T0003e80bac013f2ffd00000200000a
E001000


SIC Assembler
File Assembler
** Source File
** List File
** Object FIle
Hbgn 001000000000
T00100f15ffffff00000200000501000a00000aac0148656c6c
T001027110f2fee1d0003fffffb1900161b2fd99001
T0003e80bac013f2ffd00000200000a
E001000