

REPORT PROJECT ASSEMBLER PASS 1

NAME :AYA ABDEL MONEIM KHALIL

ID:5

Design -->

Main data structures:

map<string,int>standardOperations;

A map mapping every mnemonic to its format to facilitate the look-up during validation and updating the location according to the corresponding format.

Vector [<string,long> symTab] :

The vector holds the symbol table containing each label and its location in decimal.

Hash Set [std::set<string>usedLabels] :

The set contains the used label throughout the program to check any new label if it was already used before ...

Main Methods

read_Parse_File()

bool validLabel(string lbl)

bool validOperands(string operands)

bool validOperation(string operation)

Auxiliary Methods

The following methods are used only in the free format :

string getOperands(vector<string>token)

string getLabel(vector<string>token,string operation)

string get Operation(vector<string>token)

bool canBeLabel(string label)

checks if the label starts with letter and doesn't contain special characters such [*,']
+ whitespaces

string int_to_hex(int location)

converts the given number to a hexadecimal string starting with 0

string print(int lineNumber,long location,string label,string operation,string operands,string comment,string errors)

put the final output in a proper format to be printed .

bool wordLiteral(string operands)

checks if the following sequence applies to the given string
=w'some positive or negative number'

bool isNumeric(string operands)

checks if the given string is a number

Algorithm description/Program flow :

1. Menmonics with the corresponding formats are automatically uploaded to the program in an unordered_map [Hash Table] through setOperationVector() method.

2. The input file is read line by line ...where each line goes through the following :

- 1-8 label
- 9 blank
- 10-15 operation code
- 16-17 blank
- 18-35 operand
- 36-66 comment

each of the previous fields is validated by the corresponding validation function and put in strings to be later displayed.

3. If any of the fields contain errors an error message is displayed but the most critical field is the operation field because it is directly connected with the location counter thus if it contains any error the whole line is ignored same goes if the line has a formatting error.

4. Now, that all fields are processed, the fields are printed through print(...) method properly in the output file.

5. Same steps are repeated from step 1 till the end of the input file...

Free Format Algorithms Used:

Same as fixed format but the line is parsed on the operation first and the left side is the label and finally the right side is the operands and each field is validated accordingly using each of the above validation functions.

Assumptions/Handling Special Inputs :

1. Format 4 [+op] the (+) sign is included within the operation field
example :

.23456789012345678901234567890123456

.Label. Opcode The Operand

```
.
      START    0000
BGN   +LDA     ALPHA
      +ADD     INCR
      +SUB     THREE
      STA      BETA
      J        *
```

```
.
THREE  WORD    3
ALPHA  RESW    1
BETA   RESW    1
INCR   RESW    1
      END      BGN |
```

2.Line will be ignored in following cases :

-->Empty

-->Commented

-->ill-Formatted

example :

.23456789012345678901234567890123456

.Label. Opcode The Operand

```
.
      START    0000
BGN   +LDA     ALP,HA
      +ADD     INCR

      +SUB     THREE
      STA      BETA
      J        *|
```

```
.
THREE  WORD    3
ALPHA  RESW    1
BETA   RESW    1
INCR   RESW    1
      END      BGN
```

3.If the start statement is missing no error is displayed and the location counter starts from 0000 but an error is displayed in case of missing end statement.

4.A label can't contain any of the following characters[*,'+]' + whitespaces also must start with letter .

5 . Comments in the free format must start with [.]

Sample Runs [Fixed Format]:

1.Empty File only \n \n \n

Line	LOCATION	LABEL	OPERATION	OPERANDS	COMMENTS

Missing End statement					
>>	e n d	o f	p a s s	1	
>>	*****				
>>	s y m b o l	t a b l e	(values in decimal)		
	name	value			

2.No start statement

Some Fields Missing

No End Statement

```

.Empty Line Coming Up

.Illegal input Coming Up
    sjfknaajklmfklalmfewiufhiulwf
.2345678901234567890
.Missing Start
    LDX    #0
.Missing Operation Field
    A
.Line with Tabs
    TD     =x'aa'
    JEQ     RLOOP
.Missing Operand Field
    RD
    COMP
    JEQ     @1
CHECK2    COMP    #135|
    J       PRINT
UP        JSUB    11

    tix     =w'5'
    wd      4
PRINT    JSUB    STRING
    RMO     X,A
    5
    RMO     A,X
    COMPR   X,S
    JLT     OUTPUT
    J

....Comment Line .....Comment Line....    ?!.....
.Missing Label Field
    RESB    4
int        word    5
INDEV      BYTE    c''
OUTDEV     BYTE    X'06'
.Missing End

```

Line	LOCATION	LABEL	OPERATION	OPERANDS	COMMENTS
1	000000				.Empty Line Comming Up
2	000000				.Illegal input Coming Up
3	000000	--	--	--	
Error : Invalid Mnemonic Sytax Error:Line Format Error					
4	000000				.2345678901234567890
5	000000				.Missing Start
6	000000	LDX	#0		
7	000003				.Missing Operation Field
8	000003		A		
Error:Empty Operation Field					
9	000003				.Line with Tabs
10	000003	--	--	--	
Error:Line Format Error [Illegal Operands]					
11	000003	JEQ	RLOOP		
12	000006				.Missing Operand Field
13	000006	RD			
Operands Required					
14	000009	COMP			
Operands Required					
15	00000C	JEQ	@1		

15	00000C		JEQ	@1
16	00000F	CHECK2	COMP	#135
17	000012		J	PRINT
18	000015	UP	JSUB	11
19	000018		tix	=w'5'
20	00001B		wd	4
21	00001E	PRINT	JSUB	STRING
22	000021		RMO	X,A
23	000023			5
Error:Empty Operation Field				
24	000023		RMO	A,X
25	000025		COMPR	X,S
26	000027		JLT	OUTPUT
27	00002A		J	
Operands Required				
28	00002D		Comment LineComment Line.... ?!....
29	00002D			.Missing Label Field
30	00002D		RESB	4
Operation Requires Label				
31	00002D	int	word	5
32	000030	INDEV	BYTE	c''

24	000023		RMO	A,X
25	000025		COMPR	X,S
26	000027		JLT	OUTPUT
27	00002A		J	
Operands Required				
28	00002D		Comment LineComment Line.... ?!.....
29	00002D			.Missing Label Field
30	00002D		RESB	4
Operation Requires Label				
31	00002D	int	word	5
32	000030	INDEV	BYTE	c''
33	000030	OUTDEV	BYTE	X'06'
34	000031			.Missing End

Missing End statement

```
>> end of pass 1
>> *****
>> symbol table (values in decimal)
    name      value
    -----
    check2     15
    up         21
    print      30
    int        45
    indev      48
    outdev     48
```


3.Duplicate Start statement

Duplicate End statement

ill-Formatted Lines

```
.23456789012345678901234567890123456
.Label. Opcode The Operand
.
      START 0000
BGN   LDA   ALPHA
      ADD   INCR
      SUB   # 3
      STA   BETA
      start 0
      J     *
.
      ALPHA RESW 5
BETA  RESW 2
      end
INCR  RESW 0
      END   BGN
```

Line	LOCATION	LABEL	OPERATION	OPERANDS	COMMENTS
1	000000				.23456789012345678901234567890123456
2	000000				.Label. Opcode The Operand
3	000000				.
4	000000	START	0000		
Misplaced start statement					
5	000000	--	--	--	--
Error:Line Format Error					
Error : Invalid Mnemonic Sytax					
6	000000	ADD	INCR		
7	000003	--	--	--	--
Error:Line Format Error [No whitespace]					
8	000003	--	--	--	--
Error:Line Format Error [No whitespace]					
9	000003	start	0		
Misplaced start statement					
10	000003	J	*		
11	000006				.
12	000006	--	--	--	--
Error:Empty Operation Field					
13	000006	BETA	RESW	2	
14	00000C	end			

```

6      000000      ADD      INCR

7      000003  --      --      --      --
Error:Line Format Error [No whitespace]

8      000003  --      --      --      --
Error:Line Format Error [No whitespace]

9      000003      start    0

Misplaced start statement
10     000003      ]      *

11     000006      .

12     000006  --      --      --      --
Error:Empty Operation Field

13     000006  BETA      RESW    2

14     00000C      end

15     00000C  INCR      RESW    0

16     00000C      END      BGN

Misplaced end statement
>>   e n d    o f    p a s s    1
>>   *****
>>   s y m b o l    t a b l e    (values in decimal)
>>       name          value
>>       -----
>>       beta          6
>>       incr          12

```

4.Invalid Operands

Invalid Labels

Invalid Operations

.23456789012345678901234567890123456

.Label. Opcode The Operand

.

```
4BGN      START    0000
          LDA      ALPH,A
          ADD      INCR
          SUB      #-3
          STA      @string,x
          INVALID *
```

.

```
ALPHA     RESW     -5
BETA      RESW     2
INCR      RESW     0
          END      BGN
```

Line	LOCATION	LABEL	OPERATION	OPERANDS	COMMENTS
1	000000				.23456789012345678901234567890123456
2	000000				.Label. Opcode The Operand
3	000000				.
4	000000	START	0000		
5	000000	LDA	ALPH,A		
Error:Invalid Label Syntax[First Character Must be a Letter]					
Incorrect Operands					
6	000003	ADD	INCR		
7	000006	SUB	#-3		
8	000009	STA	@string,x		
[Op m,x]					
9	00000C	--	--	--	--
Error : Invalid Mnemonic Sytax					
Error:Line Format Error					
10	00000C				.
11	00000C	ALPHA	RESW	-5	
Incorrect Operands					
12	00000C	BETA	RESW	2	
13	000012	INCR	RESW	0	
14	000012	END	BGN		
>> end of pass 1					
>> *****					
>> symbol table (values in decimal)					

```

3      000000
4      000000          START      0000
5      000000          LDA      ALPH,A
Error:Invalid Label Syntax[First Character Must be a Letter]
Incorrect Operands
6      000003          ADD      INCR
7      000006          SUB      #-3
8      000009          STA      @string,x
[Op m,x]
9      00000C  --      --      --
Error : Invalid Mnemonic Sytax
Error:Line Format Error

10     00000C
11     00000C  ALPHA    RESW      -5
Incorrect Operands
12     00000C  BETA     RESW      2
13     000012  INCR     RESW      0
14     000012          END      BGN

>>   e n d      o f      p a s s   1
>>   *****
>>   s y m b o l      t a b l e   (values in decimal)
      name          value
      -----
      alpha          12
      beta            12
      incr            18

```

5.Special Inputs [Literals , addressing modes ,Indexing]

```
.  copy 8 integers from u to v
.  loop index <> 1
prog    start    0000
bgn     lds      #3
        ldt      #24
        +ldx     @3
loop    lda      =w'-11'
        sta      v,x
        addr     s,x
        compr    x,t
        jlt      *-10
        j        *+10

.
.  list of values for u
u        word    1,-2,3
n        byte    c'test
c        byte    x'05'
u        byte    c
.  array to store results
v        resw     8
dr        byte    c''
cr        byte    x'5'
        end      bgn
```

```

14  00001A                                .  list of values for u
15  00001A  u          word      1,-2,3
16  000023  n          byte      c'test      '
17  000029  c          byte      x'05'
18  00002A  --          --          --          --
Error:Label is Already Used
Incorrect Operands
19  00002A                                .  array to store results
20  00002A  v          resw      8
21  000042  dr          byte      c''
22  000042  cr          byte      x'5'
Not Hexa Literal
23  000043                                end      bgn

>>  e n d    o f    p a s s   1
>>  *****
>>  s y m b o l    t a b l e  (values in decimal)
      name          value
      -----
      bgn           0
      loop          10
      u             26
      n             35
      c             41
      v             42
      dr            66
      cr            66

```

Sample Runs [Free-Format]

|.2345678901234567890

```

      LDX      #0
CLEAR      A
RESB      TD      =x'aa'
      JEQ      RLOOP
      RD      5
      COMP     #4
      JEQ      @1
      COMP     #0
      JEQ      HERE
      STCH     5
      TIX     #1000
      JLT      RLOOP
HERE      RMO      X,S
      LDX      #      0
OUTPUT    LDA      int    ,      x
      STA      int    ,x
      COMP     =      W      '      5      '
      JGT      CHECK2
      J        STRING,X
CHECK2     COMP     #      135
      JLT      UP
      J        PRINT
UP         JSUB     11
      STCH     INDEV
      LDCH     INDEV
      tix      =w'5'
      wd      4
PRINT     JSUB     STRING
      RMO      X,A
      ADD      5
      RMO      A      ,      X
      COMPR    X,S
      JLT      OUTPUT
      J

```

:

```

                                JLT      RLOOP
                                RMO      X,S
HERE    LDX      #      0
OUTPUT  LDA      int    ,      x
STA     int     ,x
COMP    =      W      '      5      '
JGT     CHECK2
J       STRING,X
CHECK2  COMP     #      135
JLT     UP
J       PRINT
UP      JSUB     11
STCH    INDEV
LDCH    INDEV
        tix     =w'5'
PRINT   wd      4
        JSUB    STRING
RMO     X,A
        ADD     5
RMO     A      ,      X
        COMPR   X,S
JLT     OUTPUT
        J       *
.....
STRING  RESB     4
int     word
INDEV   BYTE    c''
OUTDEV  BYTE    X      '      06'
.....
        TOUPPER SUB     #32
        RSUB
.....
PRINTC  TD      OUTDEV
        JEQ     PRINTC
        WD      OUTDEV
        END

```

.....OUTPUT.....

Line	LOCATION	LABEL	OPERATION	OPERANDS	COMMENTS
1	000000 /	/	/		.2345678901234567890
2	000000 /	LDX	#0	/	
3	000003 /	CLEAR	A	/	
4	000005 /	/		/	
	ERROR IN OPERANDS FIELD				
5	000005 /	JEQ	RLOOP	/	
6	000008 /	RD	5	/	
7	00000B /	COMP	#4	/	
8	00000E /	JEQ	@1	/	
9	000011 /	COMP	#0	/	
10	000014 /	JEQ	HERE	/	
11	000017 /	STCH	5	/	
12	00001A /	TIX	#1000	/	
13	00001D /	JLT	RLOOP	/	
14	000020 HERE	RMO	X,S	/	
15	000022 /	LDX	#0	/	
16	000025 OUTPUT	LDA	int,x	/	
17	000028 /	STA	int,x	/	

18	00002B	/	COMP	=W'5'	/
19	00002E	/	JGT	CHECK2	/
20	000031	/	J	STRING,X	/
21	000034	CHECK2	COMP	#135	/
22	000037	/	JLT	UP	/
23	00003A	/	J	PRINT	/
24	00003D	UP	JSUB	11	/
25	000040	/	STCH	INDEV	/
26	000043	/	LDCH	INDEV	/
27	000046	/	tix	=w'5'	/
28	000049	/	wd	4	/
29	00004C	PRINT	JSUB	STRING	/
30	00004F	/	RMO	X,A	/
31	000051	/	ADD	5	/
32	000054	/	RMO	A,X	/
33	000056	/	COMPR	X,S	/
34	000058	/	JLT	OUTPUT	/
35	00005B	/	J	*	/

30	00004F	/	RMO	X,A	/
31	000051	/	ADD	5	/
32	000054	/	RMO	A,X	/
33	000056	/	COMPR	X,S	/
34	000058	/	JLT	OUTPUT	/
35	00005B	/	J	*	/
36	00005E	/	/	/
37	00005E	STRING	RESB	4	/
38	000062	int	word	5	/
39	000065	INDEV	BYTE	c''	/
40	000065	OUTDEV	BYTE	c' 06'	/
41	000066	/	/	/
42	000066	TOUPPER	SUB	#32	/
43	000069	/	RSUB		/
44	00006C	/	/	/
45	00006C	PRINTC	TD	OUTDEV	/
46	00006F	/	JEQ	PRINTC	/
47	000072	/	WD	OUTDEV	/

```

40  000065  OUTDEV  BYTE    c' 06'          /
41  000066  /          /          /          .....
/
42  000066  TOUPPER  SUB     #32           /
43  000069  /          RSUB           /
44  00006C  /          /          /          .....
/
45  00006C  PRINTC   TD      OUTDEV       /
46  00006F  /          JEQ     PRINTC     /
47  000072  /          WD      OUTDEV     /
48  000075  /          END              /

>>  e n d   o f   p a s s   1
>>  *****
>>  s y m b o l   t a b l e   (values in decimal)
      name          value
      -----
      here          32
      output        37
      check2        52
      up            61
      print         76
      string        94
      int           98
      indev         101
      outdev        101
      toupper       102
      printc        108

```

.23456789012345678901234567890123456

.Label. Opcode The Operand

.

```
          START  0000
BGN      LDA    ALPHA
          ADD    INCR
          SUB    # 3
          STA    BETA
          start  0
          J      *
```

.

```
          ALPHA  RESW  5
BETA     RESW  2
          end
INCR     RESW  0
          END    BGN
```

.....OUTPUT.....

Line	LOCATION	LABEL	OPERATION	OPERANDS	COMMENTS
1	000000 /	/	/		.23456789012345678901234567890123456
2	/				
2	000000 /	/	/		.Label. Opcode The Operand
3	/				
3	000000 /	/	/		.
4	/				
4	000000 /	START	0000		/
5	000000 BGN	LDA	ALPHA		/
6	000003 /	ADD	INCR		/
7	000006 /	SUB	#3		/
8	000009 /	STA	BETA		/
9	00000C /	start	0		/
Misplaced or Duplicate Start statement					
10	00000C /	J	*		/
11	00000F /	/	/		.
12	/				
12	00000F ALPHA	RESW	5		/
13	00001E BETA	RESW	2		/
14	000024 /	end			/
15	000024 INCR	RESW	0		/
16	000024 /	END	BGN		/
Misplaced or Duplicate end statement					

5	000000	BGN	LDA	ALPHA	/
6	000003	/	ADD	INCR	/
7	000006	/	SUB	#3	/
8	000009	/	STA	BETA	/
9	00000C	/	start	0	/
Misplaced or Duplicate Start statement					
10	00000C	/	J	*	/
11	00000F	/	/	/	.
12	00000F	ALPHA	RESW	5	/
13	00001E	BETA	RESW	2	/
14	000024	/	end		/
15	000024	INCR	RESW	0	/
16	000024	/	END	BGN	/

Misplaced or Duplicate end statement

```
>>  e n d   o f   p a s s   1
>>  *****
>>  s y m b o l   t a b l e   (values in decimal)
      name          value
      -----
      bgn           0
      alpha         15
      beta          30
      incr          36
```


.2345678901234567890

COPY	START	0
FIRST	STL	RETADR
	LDB	#LENGTH
	BASE	LENGTH
CLOOP	+JSUB	RDREC
	LDA	LENGTH
	COMP	#0
	JEQ	ENDFIL
	+JSUB	WRREC
	J	CLOOP
ENDFIL	LDA	EOF
	STA	BUFFER
	LDA	#3
	STA	LENGTH
	+JSUB	WRREC
	J	@RETADR
EOF	BYTE	C'EOF'
RETADR	RESW	1
LENGTH	RESW	1
BUFFER	RESB	4096
.		
.	SUBROUTINE TO READ RECORD INTO BUFFER	
.		
RDREC	CLEAR	X
	CLEAR	A
	CLEAR	S
fourk	equ	4096
	+LDT	#fourk
RLOOP	TD	INPUT
	JEQ	RLOOP
	RD	INPUT
	COMPR	A,S
	JEQ	EXIT
	STCH	BUFFER,X
	TIXR	T

```

RETADR    RESW    1
LENGTH    RESW    1
BUFFER     RESB    4096
.
.         SUBROUTINE TO READ RECORD INTO BUFFER
.
RDREC      CLEAR   X
           CLEAR   A
           CLEAR   S
fourk      equ     4096
           +LDT    #fourk
RLOOP      TD      INPUT
           JEQ     RLOOP
           RD      INPUT
           COMPR   A,S
           JEQ     EXIT
           STCH    BUFFER,X
           TIXR    T
           JLT     RLOOP
EXIT        STX     LENGTH
           RSUB
INPUT       BYTE    X'F1'
.
.         SUBROUTINE TO WRITE RECORD FROM BUFFER
.
WRREC      CLEAR   X
           LDT     LENGTH
WLOOP      TD      OUTPUT
           JEQ     WLOOP
           LDCH    BUFFER,X
           WD      OUTPUT
           TIXR    T
           JLT     WLOOP
           RSUB
OUTPUT     BYTE    X'05'
           END     FIRST

```


.....OUTPUT.....

Line	LOCATION		LABEL	OPERATION	OPERANDS	COMMENTS
1	000000	/	/	/		.2345678901234567890
2	000000	COPY	START	0	/	
3	000000	FIRST	STL	RETADR	/	
4	000003	/	LDB	#LENGTH	/	
5	000006	/	BASE	LENGTH	/	
6	000006	CLOOP	+JSUB	RDREC	/	
7	00000A	/	LDA	LENGTH	/	
8	00000D	/	COMP	#0	/	
9	000010	/	JEQ	ENDFIL	/	
10	000013	/	+JSUB	WRREC	/	
11	000017	/	J	CLOOP	/	
12	00001A	ENDFIL	LDA	EOF	/	
13	00001D	/	STA	BUFFER	/	
14	000020	/	LDA	#3	/	
15	000023	/	STA	LENGTH	/	
16	000026	/	+JSUB	WRREC	/	
17	00002A	/	J	@RETADR	/	

20	000033	LENGTH	RESW	1	/	
21	000036	BUFFER	RESB	4096	/	
22	001036	/	/	/	.	
23	001036	/	/	/	.	SUBROUTINE TO READ RECORD INTO BUFFER
24	001036	/	/	/	.	
25	001036	RDREC	CLEAR	X	/	
26	001038	/	CLEAR	A	/	
27	00103A	/	CLEAR	S	/	
28	00103C	fourk	equ	4096	/	
29	00103C	/	+LDT	#fourk	/	
30	001040	RLOOP	TD	INPUT	/	
31	001043	/	JEQ	RLOOP	/	
32	001046	/	RD	INPUT	/	
33	001049	/	COMPR	A,S	/	
34	00104B	/	JEQ	EXIT	/	
35	00104E	/	STCH	BUFFER,X	/	
36	001051	/	TIXR	T	/	
37	001053	/	JLT	RLOOP	/	

39	001059	/	RSUB		/	
40	00105C	INPUT	BYTE	c'F1'	/	
41	00105D	/	/	/	.	
42	00105D	/	/	/	.	SUBROUTINE TO WRITE RECORD FROM BUFFER
43	00105D	/	/	/	.	
44	00105D	WRREC	CLEAR	X	/	
45	00105F	/	LDT	LENGTH	/	
46	001062	WLOOP	TD	OUTPUT	/	
47	001065	/	JEQ	WLOOP	/	
48	001068	/	LDCH	BUFFER,X	/	
49	00106B	/	WD	OUTPUT	/	
50	00106E	/	TIXR	T	/	
51	001070	/	JLT	WLOOP	/	
52	001073	/	RSUB		/	
53	001076	OUTPUT	BYTE	c'05'	/	
54	001077	/	END	FIRST	/	

```
>> end of pass 1
>> *****
>> symbol table (values in decimal)
>>      name      value
```

48	001068	/	LDCH	BUFFER,X	/
49	00106B	/	WD	OUTPUT	/
50	00106E	/	TIXR	T	/
51	001070	/	JLT	WLOOP	/
52	001073	/	RSUB		/
53	001076	OUTPUT	BYTE	c'05'	/
54	001077	/	END	FIRST	/

>> e n d o f p a s s 1

>> *****

>> s y m b o l t a b l e (values in decimal)

name	value

first	0
cloop	6
endfil	26
eof	45
retadr	48
length	51
buffer	54
rdrec	4150
fourk	4156
rloop	4160
exit	4182
input	4188
wrrec	4189
wloop	4194
output	4214
