

NIRAJ KUMAR

BT19CS031

QUESTION 1

Write a C program to read a SIC (Simplified Instructional Computer) assembly language program and create three text (.txt) files. These three files must content labels, instructions, and operands present in the program separately. Assume that the columns of the SIC program are tab-separated.

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
#define _GNU_SOURCE
#include <assert.h>
#include <stdlib.h>
#include <stdlib.h>
```

```
FILE *fpr=NULL, *fpw=NULL, *fps=NULL, *fpi=NULL, *fpo=NULL;
```

```
char addr(char ch)
{
    int n;
    n=0;      n=ch-'0';
    n+=8;     if(n==8)
    return '8';
    else if(n==9)
    return '9';
    else if(n==10)
    return 'A';
    else if(n==11)
    return 'B';
    else if(n==12)
    return 'C';
    else if(n==13)
    return 'D';
    else if(n==14)
    return 'E';
}
```

```

else if(n==15)
return 'F';
}
int
main() {
    char label[10],mnemonic[10]
,operand[10],program_name[10],sym_label[30][10],sym_address[30][10],

opcodes[1000][3],opcode[10],str[80],st[10],ch,c,chr[10],text[80][80]
;

int
address=0,line=0,length,x=0,first_address=0,last_address=0,pa=0,obj_
code=0,i=0,j=0,k=0,pos1=0,
    pos2=0,flag1=0,flag2=0,p1=0,p2=0,l=0,z=0,n=0,count=0,l1=0,l2=0;

char
op_mnemonic[26][10]={"LDAC","STAC","SUBJ","MULT","STRL","DIVD","ADDA
","STRCH","DT",

"JMPEQ","LOADCH","DW","SUBR","FIXR","FIX","JMPLT","COMP","CLR","LOAD
B","J","DR","COMPR","STREX","LOADT","LOADL","LOADX"};

char
op_opcode[26][10]={"10","1C","58","30","24","34","28","44","F0","40"
,"60","EC",

"4C","C8","3C","39","38","C4","78","3C","E8","D0","20","84","08","04
"};

    /*Adressing*/
    fpr=fopen("new_input.txt","r+");
    fpw=fopen("addressed_file.txt","w+");

    fscanf(fpr,"%s%s%X",&program_name,&mnemonic,&address);
    /*Reading data from input.txt*/
    line++;
    fprintf(fpw,"%d\t\t%s\t%s\t%.4X\n",line
,program_name,mnemonic,adres
s); /*Writing data into addressed_file.txt */
    while(strcmp(mnemonic,"END")!=0)
    {
        if(strcmp(label,".")!=0)                /*For checking comments*/
        {
            fscanf(fpr,"%s%s%s",&label,&mnemonic,&operand);
            line++;

```

```
fprintf(fpw, "%d\t%.4X\t%s\t%s\t%s\n", line, address, label, mnemonic, operand); /*Writing data into addressed_file.txt */
```

```
/*checking conditions for various mnemonics*/
```

```

    if(strcmp(mnemonic, "BYTE")==0)
    {
        length=0;
length=strlen(operand);
        length-=3;          /*excluding C' ,X' */

        if(operand[0]=='C')
address+=length;
        else
    {
            length=(length/2);
address+=length;
        }
    }

    else if(strcmp(mnemonic, "WORD")==0)
        address+=3;
    else if(strcmp(mnemonic, "RESERW")==0)
    {
        x=atoi(operand);    /*converting character into
integer */
        address+=(x*3);      /*1 WORD = 3 BYTES */
    }
    else if(strcmp(mnemonic, "RESERB")==0)
    {
        x=atoi(operand);    /*change RESERB's string
mnemonic into integer --> x */
        address+=x;
    }
    else
        address+=3;
}

fprintf(fpw, "\n");
}
fclose(fpr);
fclose(fpw);

```

```
/*Creating Symtab*/
```

```

fpw=fopen("addressed_file.txt", "r+");
fps=fopen("symtab.txt", "w+");    fpi=fopen("instab.txt", "w+");
fpo=fopen("opetab.txt", "w+");

```

```

fscanf(fpw, "%d%s%s", &line, &program_name, &mnemonic, &operand);

while(strcmp(mnemonic, "END") != 0)
{
fscanf(fpw, "%d%X%s%s", &line, &address, &label, &mnemonic, &operand);

    if(line == 2)
        first_address = address;    /*Will be used during object
file creation */

    if(strcmp(label, "--") != 0)
    {
        fprintf(fps, "%s  %X\n", label, address);
j++;
    }
    if(strcmp(mnemonic, "--") != 0)
    {
        fprintf(fpi, "%s  %X\n", mnemonic, address);
    }
    if(strcmp(operand, "--") != 0)
    {
        fprintf(fpo, "%s  %X\n", operand, address);
    }
}

last_address = address;    /*Will be used during object file
creation */    fclose(fpr);    fclose(fpw);    fclose(fps);
fclose(fpi);    fclose(fpo);
printf("\n\n***** Printing content of - addressed_file.txt
*****\n\n");
fpw = fopen("addressed_file.txt", "r");
char _ch = fgetc(fpw);
while(!feof(fpw))
{
    printf("%c", _ch);
    _ch = fgetc(fpw);
}
printf("\n\n***** Printing content of - symtab.txt
*****\n\n");
fps = fopen("symtab.txt", "r");
_ch = fgetc(fps);
while(!feof(fps))
{
    printf("%c", _ch);
    _ch = fgetc(fps);
}
printf("\n\n***** Printing content of - instab.txt
*****\n\n");

```

```

    fpi=fopen("instab.txt", "r");
    _ch=fgetc(fpi);
while(!feof(fpi))
{
    printf("%c", _ch);
    _ch=fgetc(fpi);
}
    printf("\n\n***** Printing content of - opetab.txt
*****\n\n");
    fpo=fopen("opetab.txt", "r");
    _ch=fgetc(fpo);
while(!feof(fpo))
{
    printf("%c", _ch);
    _ch=fgetc(fpo);
}
fclose(fpr);
fclose(fpw);
fclose(fps);
fclose(fpi);
fclose(fpo);
return 0;
}

```

OUTPUT –

```

***** Printing content of - addressed_file.txt *****
1      COPY      START 1000
2      1000 FIRST STRL RETADR
3      1003 CLOOP SUBJ  RDRAC
4      1006 -- LDAC  LENGTH
5      1009 -- COMP  ZERO
6      100C -- JNPEQ ENDFIL
7      100F -- SUBJ  WRREC
8      1012 -- J     CLOOP
9      1015 ENDFIL LDAC  EOF
10     1018 -- STAC  BUFFER
11     101B -- LDAC  THREE
12     101E -- STAC  LENGTH
13     1021 -- SUBJ  WRREC
14     1024 -- LOADL RETADR
15     1027 -- SUBR  --
16     102A EOF  BYTE  C'EOF'
17     102D THREE WORD  3
18     1030 ZERO  WORD  0
19     1033 RETADR RESERW 1
20     1036 LENGTH RESERW 1
21     1039 BUFFER RESERB 4096
22     2039 RDRAC LOADX  ZERO
23     203C -- LDAC  ZERO
24     203F RLOOP  DT     INPUT
25     2042 -- JNPEQ RLOOP
26     2045 -- DR     INPUT
27     2048 -- COMP  ZERO
28     204B -- JNPEQ EXIT
29     204E -- STREX  BUFFER,X
30     2051 -- FIX   MAXLEN
31     2054 -- JNPLT RLOOP
32     2057 EXIT  STREX  LENGTH
33     205A -- SUBR  --
34     205D INPUT  BYTE  X'F1'
35     205E MAXLEN WORD  4096
36     2061 WRREC  LOADX  ZERO
37     2064 WLOOP  DT     OUTPUT

```

```
"C:\Users\Niraj\Desktop\c++\Question 1\SikParcer.exe"
J 1012
LDAC 1015
STAC 1018
LDAC 101B
STAC 101E
SUBJ 1021
LOADL 1024
SUBR 1027
BYTE 102A
WORD 102D
WORD 1030
RESERW 1033
RESERW 1036
RESERB 1039
LOADX 2039
LDAC 203C
DT 203F
JMPEQ 2042
DR 2045
COMP 2048
JMPEQ 204B
STRCH 204E
FIX 2051
JMPLT 2054
STREX 2057
SUBR 205A
BYTE 205D
WORD 205E
LOADX 2061
DT 2064
JMPEQ 2067
LOADCH 206A
DW 206D
FIX 2070
JMPLT 2073
SUBR 2076
BYTE 2079
END 207A

***** Printing content of - opetab.txt *****
```

```
"C:\Users\Niraj\Desktop\c++\Question 1\SikParcer.exe"
37 2064 WLOOP DT OUTPUT
38 2067 -- JMPEQ WLOOP
39 206A -- LOADCH BUFFER,X
40 206D -- DW OUTPUT
41 2070 -- FIX LENGTH
42 2073 -- JMPLT WLOOP
43 2076 -- SUBR --
44 2079 OUTPUT BYTE X'05'
45 207A -- END FIRST

***** Printing content of - symtab.txt *****

FIRST 1000
CLOOP 1003
ENDFIL 1015
EOF 102A
THREE 102D
ZERO 1030
RETADR 1033
LENGTH 1036
BUFFER 1039
RDREC 2039
RLOOP 203F
EXIT 2057
INPUT 205D
MAXLEN 205E
WRREC 2061
WLOOP 2064
OUTPUT 2079

***** Printing content of - instab.txt *****

STRL 1000
SUBJ 1003
LDAC 1006
COMP 1009
JMPEQ 100C
SUBJ 100F
J 1012
```

```
"C:\Users\Niraj\Desktop\c++\Question 1\SicParcer.exe"
***** Printing content of - opetab.txt *****

RETADR 1000
RDREC 1003
LENGTH 1006
ZERO 1009
ENDFIL 100C
WRREC 100F
CLOOP 1012
EOF 1015
BUFFER 1018
THREE 101B
LENGTH 101E
WRREC 1021
RETADR 1024
C'EOF' 102A
3 102D
0 1030
1 1033
1 1036
4096 1039
ZERO 2039
ZERO 203C
INPUT 203F
RLOOP 2042
INPUT 2045
ZERO 2048
EXIT 2048
BUFFER,X 204E
MAXLEN 2051
RLOOP 2054
LENGTH 2057
X'F1' 205D
4096 205E
ZERO 2061
OUTPUT 2064
WLOOP 2067
BUFFER,X 206A
OUTPUT 206D
LENGTH 2070
WLOOP 2073
```

```
"C:\Users\Niraj\Desktop\c++\Question 1\SicParcer.exe"
0 1030
1 1033
1 1036
4096 1039
ZERO 2039
ZERO 203C
INPUT 203F
RLOOP 2042
INPUT 2045
ZERO 2048
EXIT 2048
BUFFER,X 204E
MAXLEN 2051
RLOOP 2054
LENGTH 2057
X'F1' 205D
4096 205E
ZERO 2061
OUTPUT 2064
WLOOP 2067
BUFFER,X 206A
OUTPUT 206D
LENGTH 2070
WLOOP 2073
X'05' 2079
FIRST 207A

Process returned 0 (0x0) execution time : 0.316 s
Press any key to continue.
```

```
instab.txt - Notepad
File Edit Format View Help
STRL 1000
SUBJ 1003
LDAC 1006
COMP 1009
JMPEQ 100C
SUBJ 100F
J 1012
LDAC 1015
STAC 1018
LDAC 101B
STAC 101E
SUBJ 1021
LOADL 1024
SUBR 1027
BYTE 102A
WORD 102D
WORD 1030
RESERW 1033
RESERW 1036
RESERB 1039
LOADX 2039
LDAC 203C
DT 203F
JMPEQ 2042
DR 2045
COMP 2048
JMPEQ 2048
STRCH 204E
FIX 2051
JMPLT 2054
STREX 2057
SUBR 205A
BYTE 205D
WORD 205E
-----
```

```
instab.txt - Notepad
File Edit Format View Help
SUBJ 1021
LOADL 1024
SUBR 1027
BYTE 102A
WORD 102D
WORD 1030
RESERW 1033
RESERW 1036
RESERB 1039
LOADX 2039
LDAC 203C
DT 203F
JMPEQ 2042
DR 2045
COMP 2048
JMPEQ 2048
STRCH 204E
FIX 2051
JMPLT 2054
STREX 2057
SUBR 205A
BYTE 205D
WORD 205E
LOADX 2061
DT 2064
JMPEQ 2067
LOADCH 206A
DW 206D
FIX 2070
JMPLT 2073
SUBR 2076
BYTE 2079
END 207A
```



```
opetab.txt - Notepad
File Edit Format View Help
RETADR 1000
RDREC 1003
LENGTH 1006
ZERO 1009
ENDFIL 100C
WRREC 100F
CLOOP 1012
EOF 1015
BUFFER 1018
THREE 101B
LENGTH 101E
WRREC 1021
RETADR 1024
C'EOF' 102A
3 102D
0 1030
1 1033
1 1036
4096 1039
ZERO 2039
ZERO 203C
INPUT 203F
RLOOP 2042
INPUT 2045
ZERO 2048
EXIT 204B
BUFFER,X 204E
MAXLEN 2051
RLOOP 2054
LENGTH 2057
X'F1' 205D
4096 205E
ZERO 2061
OUTPUT 2064
.....
```

```
opetab.txt - Notepad
File Edit Format View Help
BUFFER 1018
THREE 101B
LENGTH 101E
WRREC 1021
RETADR 1024
C'EOF' 102A
3 102D
0 1030
1 1033
1 1036
4096 1039
ZERO 2039
ZERO 203C
INPUT 203F
RLOOP 2042
INPUT 2045
ZERO 2048
EXIT 204B
BUFFER,X 204E
MAXLEN 2051
RLOOP 2054
LENGTH 2057
X'F1' 205D
4096 205E
ZERO 2061
OUTPUT 2064
WLOOP 2067
BUFFER,X 206A
OUTPUT 206D
LENGTH 2070
WLOOP 2073
X'05' 2079
FIRST 207A
```

```
symtab.txt - Notepad
File Edit Format View Help
FIRST 1000
CLOOP 1003
ENDFIL 1015
EOF 102A
THREE 102D
ZERO 1030
RETADR 1033
LENGTH 1036
BUFFER 1039
RDREC 2039
RLOOP 203F
EXIT 2057
INPUT 205D
MAXLEN 205E
WRREC 2061
WLOOP 2064
OUTPUT 2079
```

```
new_input.txt - Notepad
File Edit Format View Help
COPY START 1000
FIRST STRL RETADR
CLOOP SUBJ RDREC
-- LDAC LENGTH
-- COMP ZERO
-- JMPEQ ENDFIL
-- SUBJ WRREC
-- J CLOOP
ENDFIL LDAC EOF
-- STAC BUFFER
-- LDAC THREE
-- STAC LENGTH
-- SUBJ WRREC
-- LOADL RETADR
-- SUBR --
EOF BYTE C'EOF'
THREE WORD 3
ZERO WORD 0
RETADR RESERW 1
LENGTH RESERW 1
BUFFER RESERB 4096
RDREC LOADX ZERO
-- LDAC ZERO
RLOOP DT INPUT
-- JMPEQ RLOOP
-- DR INPUT
-- COMP ZERO
-- JMPEQ EXIT
-- STRCH BUFFER,X
-- FIX MAXLEN
-- JMPLT RLOOP
EXIT STREX LENGTH
-- SUBR --
INPUT BYTE X'F1'
```

```
new_input.txt - Notepad
File Edit Format View Help
-- SUBJ WRREC
-- LOADL RETADR
-- SUBR --
EOF BYTE C'EOF'
THREE WORD 3
ZERO WORD 0
RETADR RESERW 1
LENGTH RESERW 1
BUFFER RESERB 4096
RDREC LOADX ZERO
-- LDAC ZERO
RLOOP DT INPUT
-- JMPEQ RLOOP
-- DR INPUT
-- COMP ZERO
-- JMPEQ EXIT
-- STRCH BUFFER,X
-- FIX MAXLEN
-- JMPLT RLOOP
EXIT STREX LENGTH
-- SUBR --
INPUT BYTE X'F1'
MAXLEN WORD 4096
WRREC LOADX ZERO
WLOOP DT OUTPUT
-- JMPEQ WLOOP
-- LOADCH BUFFER,X
-- DW OUTPUT
-- FIX LENGTH
-- JMPLT WLOOP
-- SUBR --
OUTPUT BYTE X'05'
-- END FIRST
```

```
addressed_file.txt - Notepad
File Edit Format View Help
1 1000 COPY START 1000
2 1000 FIRST STRL RETADR
3 1003 CLOOP SUBJ RDREC
4 1006 -- LDAC LENGTH
5 1009 -- COMP ZERO
6 100C -- JMPEQ ENDFIL
7 100F -- SUBJ WRREC
8 1012 -- J CLOOP
9 1015 ENDFIL LDAC EOF
10 1018 -- STAC BUFFER
11 101B -- LDAC THREE
12 101E -- STAC LENGTH
13 1021 -- SUBJ WRREC
14 1024 -- LOADL RETADR
15 1027 -- SUBR --
16 102A EOF BYTE C'EOF'
17 102D THREE WORD 3
18 1030 ZERO WORD 0
19 1033 RETADR RESERW 1
20 1036 LENGTH RESERW 1
21 1039 BUFFER RESERB 4096
22 2039 RDREC LOADX ZERO
23 203C -- LDAC ZERO
24 203F RLOOP DT INPUT
25 2042 -- JMPEQ RLOOP
26 2045 -- DR INPUT
27 2048 -- COMP ZERO
28 204B -- JMPEQ EXIT
29 204E -- STRCH BUFFER,X
30 2051 -- FIX MAXLEN
31 2054 -- JMPLT RLOOP
32 2057 EXIT STREX LENGTH
33 205A -- SUBR --
34 205D INPUT BYTE X'F1'
-- ----
```

```
addressed_file.txt - Notepad
File Edit Format View Help
13 1021 -- SUBJ WRREC
14 1024 -- LOADL RETADR
15 1027 -- SUBR --
16 102A EOF BYTE C' EOF'
17 102D THREE WORD 3
18 1030 ZERO WORD 0
19 1033 RETADR RESERW 1
20 1036 LENGTH RESERW 1
21 1039 BUFFER RESERB 4096
22 1039 RDREC LOADX ZERO
23 103C -- LDAC ZERO
24 103F RLOOP DT INPUT
25 1042 -- JMPEQ RLOOP
26 1045 -- DR INPUT
27 1048 -- COMP ZERO
28 1048 -- JMPEQ EXIT
29 104E -- STRCH BUFFER, X
30 1051 -- FIX MAXLEN
31 1054 -- JMPLT RLOOP
32 1057 EXIT STREX LENGTH
33 105A -- SUBR --
34 105D INPUT BYTE X'F1'
35 105E MAXLEN WORD 4096
36 1061 WRREC LOADX ZERO
37 1064 WLOOP DT OUTPUT
38 1067 -- JMPEQ WLOOP
39 106A -- LOADCH BUFFER, X
40 106D -- DW OUTPUT
41 1070 -- FIX LENGTH
42 1073 -- JMPLT WLOOP
43 1076 -- SUBR --
44 1079 OUTPUT BYTE X'05'
45 107A -- END FIRST
```