109 學年度 Onsite Exam 1.系統程式

1. 請以上課介紹的 SIC 環境為實作對象
2. 使用語言 C, C++, Java, C#, Python, …
3. Instruction set 可參考教科書附錄 A
4. Assembler 之演算法，請參考教科書 Fig. 2.4
5. 測試之輸入資料與格式，自行假設(SIC program)，如教科書 Fig.2.1 或 Fig.2.5

輸出格式自行假設，如教科書 Fig. 2.2 與 Fig. 2.3

1. 第一階段組譯

```
begin
        read first input line
        if OPCODE = 'START' then
                begin
                        save #[OPERAND] as starting address
                        initialize LOCCTR to starting address
                        write line to intermediate file
                        read next input line
                end {if START}
        else
                initialize LOCCTR to 0
        while OPCODE ≠ 'END' do
                begin
                        if this is not a comment line then
                                begin
                                        if there is a symbol in the LABEL field then
                                                begin
                                                        search SYMTAB for LABEL
                                                        if found then
                                                                set error flag(duplicate symbol)
                                                        else
                                                                insert (LABEL,LOCCTR) into SYMTAB
                                                end {if symbol}
                                        search OPTAB for OPCODE
                                        if found then
                                                add 3 {instruction length} to LOCCTR
                                        else if OPCODE = 'WORD' then
                                                add 3 to LOCCTR

                                        else if OPCODE = 'RESW' then
                                                add3 * #[OPERAND] to LOCCTR
                                        else if OPCODE = 'RESB' then
                                                add #[OPERAND] to LOCCTR
                                        else if OPCODE = 'BYTE' then
                                                begin
                                                        find length of constant in bytes
                                                        add length to LOCCTR
                                                end {if BYTE}
                                        else
                                                set error flag (invalid operation code)
                                end {if not a comment}
                        write line to intermediate file
                        read next input line
                end {while not END}
        write last line to intermediate file
        save (LOCCTR - starting address) as program length
end {Pass 1}
```

2. 第二階段組譯

```
begin
        read first input line {from intermediate file}
        if OPCODE = 'START' then
                begin
                        write listing line
                        read next input line
                end {if START}
        write Header record to object program
        initalize first Text record
        while OPCODE ≠ 'END' do
                begin
                        if this is not a comment line then
                                begin
                                        search OPTAB for OPCODE
                                        if found then
                                                begin
                                                        if there is a symbol in OPERAND field then
                                                                begin
                                                                        search SYMTAB for OPERAND
                                                                        if found then
                                                                                store symbol value as operand address
                                                                        else
                                                                                begin
                                                                                        store 0 as operand address
                                                                                        set error flag (undefined symbol)
                                                                                end
                                                                end {if symbol}
                                                        else
                                                                store 0 as operand address
                                                        assemble the object code instruction
                                                end {if opcode found}
                                        else if OPCODE = 'BYTE' or 'WORD' then
                                                convert constant to object code
                                        if object code will not fit into the current Text record then
                                                begin
                                                        write Text record to object program
                                                        initinlize new Text record
                                                end
                                        add object code to Text record
                                end {if not comment}
                        write listing line
                        read next input line
                end {while not END}
        write last Text record to object program
        write End record to object program
        write last listing line
end {Pass 2}
```

輸入

(SIC 程式)放入「input.txt」內，並直接執行程式

輸出

輸出檔案有三個分別為 loc.txt、output.txt、objectcode.txt

1. 輸出檔名 loc.txt

資料如下

```
1000    COPY    START   1000
1000    FIRST   STL     RETADR
1003    CLOOP   JSUB    RDREC
1006            LDA     LENGTH
1009            COMP    ZERO
100C            JEQ     ENDFIL
100F            JSUB    WRREC
1012            J       CLOOP
1015    ENDFIL  LDA     EOF
1018            STA     BUFFER
101B            LDA     THREE
101E            STA     LENGTH
1021            JSUB    WRREC
1024            LDL     RETADR
1027            RSUB
102A    EOF     BYTE    C'EOF'
102D    THREE   WORD    3
1030    ZERO    WORD    0
1033    RETADR  RESW    1
1036    LENGTH  RESW    1
1039    BUFFER  RESB    4096
2039    .
2039    .       SUBROUTINE TO READ RECORD INTO BUFFER
2039    .
2039    RDREC   LDX     ZERO
203C            LDA     ZERO
203F    RLOOP   TD      INPUT
2042            JEQ     RLOOP
2045            RD      INPUT
2048            COMP    ZERO
204B            JEQ     EXIT
204E            STCH    BUFFER,X
2051            TIX     MAXLEN
2054            JLT     RLOOP
2057    EXIT    STX     LENGTH
205A            RSUB
205D    INPUT   BYTE    X'F1'
205E    MAXLEN  WORD    4096
2061    .
2061    .       SUBROUTINE TO WRITE RECORD FROM BUFFER
2061    .
2061    WRREC   LDX     ZERO
2064    WLOOP   TD      OUTPUT
2067            JEQ     WLOOP
206A            LDCH    BUFFER,X
206D            WD      OUTPUT
2070            TIX     LENGTH
2073            JLT     WLOOP
2076            RSUB
2079    OUTPUT  BYTE    X'05'
207A            END     FIRST
```

2. 輸出檔名 output.txt

資料如下

```
1000    COPY    START   1000
1000    FIRST   STL     RETADR  141033
1003    CLOOP   JSUB    RDREC   482039
1006            LDA     LENGTH  001036
1009            COMP    ZERO    281030
100C            JEQ     ENDFIL  301015
100F            JSUB    WRREC   482061
1012            J       CLOOP   3C1003
1015    ENDFIL  LDA     EOF     00102A
1018            STA     BUFFER  0C1039
101B            LDA     THREE   00102D
101E            STA     LENGTH  0C1036
1021            JSUB    WRREC   482061
1024            LDL     RETADR  081033
1027            RSUB            4C0000
102A    EOF     BYTE    C'EOF'  454F46
102D    THREE   WORD    3       000003
1030    ZERO    WORD    0       000000
1033    RETADR  RESW    1
1036    LENGTH  RESW    1
1039    BUFFER  RESB    4096
        .
        .       SUBROUTINE TO READ RECORD INTO BUFFER
        .
2039    RDREC   LDX     ZERO    041030
203C            LDA     ZERO    001030
203F    RLOOP   TD      INPUT   E0205D
2042            JEQ     RLOOP   30203F
2045            RD      INPUT   D8205D
2048            COMP    ZERO    281030
204B            JEQ     EXIT    302057
204E            STCH    BUFFER,X        549039
2051            TIX     MAXLEN  2C205E
2054            JLT     RLOOP   38203F
2057    EXIT    STX     LENGTH  101036
205A            RSUB            4C0000
205D    INPUT   BYTE    X'F1'   F1
205E    MAXLEN  WORD    4096    001000
        .
        .       SUBROUTINE TO WRITE RECORD FROM BUFFER
        .
2061    WRREC   LDX     ZERO    041030
2064    WLOOP   TD      OUTPUT  E02079
2067            JEQ     WLOOP   302064
206A            LDCH    BUFFER,X        509039
206D            WD      OUTPUT  DC2079
2070            TIX     LENGTH  2C1036
2073            JLT     WLOOP   382064
2076            RSUB            4C0000
2079    OUTPUT  BYTE    X'05'   05
                END     FIRST
```

3. 輸出檔名 objectcode.txt

資料如下：

```
HCOPY  00100000107A
T0010001E141033482039001036281030301015482061 3C100300102A0C103900102D
T00101E150C10364820610810334C0000454F46000003000000
T0020391E041030001030E0205D30203FD8205D2810303020575490392C205E38203F
T0020571C1010364C0000F1001000041030E02079302064509039DC20792C1036
T002073073820644C000005
E001000
```