

Programming Assignment 2

410621208 資工四 龔彥安
410721371 資工三 吳侑達

1.The Problem Description:

Write a simple SIC simulator that reads a standard SIC object program into memory.

2.Highlight of the Way You Write the Program:

將程式中的 s_load、s_show 與 s_unload 完成。

s_load:將程式 load 進 memory 資料結構

```
/* Write your own s_load here. */
void s_load () {

    if(loaded){puts("Program is not released yet!\n");}
    else{
        rd_header();//獲得 start address 與 program length
        while(rd_text());//寫入 memory
        rd_end();
    }
}
```

s_show:顯示記憶體資料結構狀況

```
/* Write your own s_show here. */
void s_show () {
    #include<math.h>
    int i, cur;

    //printf("%x\n", cur);
    if(loaded)
    for(i = cur = start_add; i < 3.5*pow(16,3); ++i){

        if(i % 8 == 0)putchar(' ');

        if(i % 32 == 0){
            printf("\n");
            //printf("%d" , 32);
            printf("%d ", i);
            tentosix(cur);
            cur += 16;
        }
        putchar(memory[i]);
    }
    putchar('\n');
}
```

s_unload:釋放所有資料結構

```
/* Write your own s_unload here. */
void s_unload () {

    free(memory);
    start_add = 0;

    prog_len = 0;
    start_add = 0;
    first_add = 0;
    curr_add = 0;
    control_length = 0;
    mem_size = 0;
    loaded = 0;
    op = 0;
    indexed = 0;
    operand = 0;
    running = 0;
    control_addr = 0;
    filename[100];

}
```

3.The Program Listing:

<https://gist.github.com/peter-kong/9b8c4ad92c06902830a4a5bd0d1af57d>

4.Test Run Result:

Load & Show:

```
C:\Users\peter\Desktop\系統程式\HW02>a
SIC Simulator> load test1.obj
start_add: 1000, prog_len: 107a
SIC Simulator> show

4096 1000 14103348 20390010 36281030 30101548
4128 1010 20613C10 0300102A 0C103900 102D0C10
4160 1020 36482061 0810334C 0000454F 46000003
```

```
12320 2010 XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX
12352 2020 XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX
12384 2030 XXXXXXXX XXXXXXXX XX041030 001030E0
12416 2040 205D3020 3FD8205D 28103030 20575490
12448 2050 392C205E 38203F10 10364C00 00F10010
12480 2060 00041030 E0207930 20645090 39DC2079
12512 2070 2C103638 20644C00 0005XXXX XXXXXXXX
12544 2080 XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX
```

5.Discussion:

這次作業的 spec 讓我有點困惑，後來查了一下網路上的資料發現 load 與 show 的功能沒有那麼複雜，結果一下就寫完了，不過在 coding 途中我也遇到了一些新的啟發，像是 Address 是 16 進位顯示一行，但是一行卻有 32 個字元，其實就是課堂上所提及的 1 byte 是 8-bit，但是一個 16 進位只有 4-bit，因此機器上是以兩個 16 進位碼來做成一個 byte，因此才會有 32 個字元卻直有進位一個 16 的狀況。