

## Project 3 – Command Line System

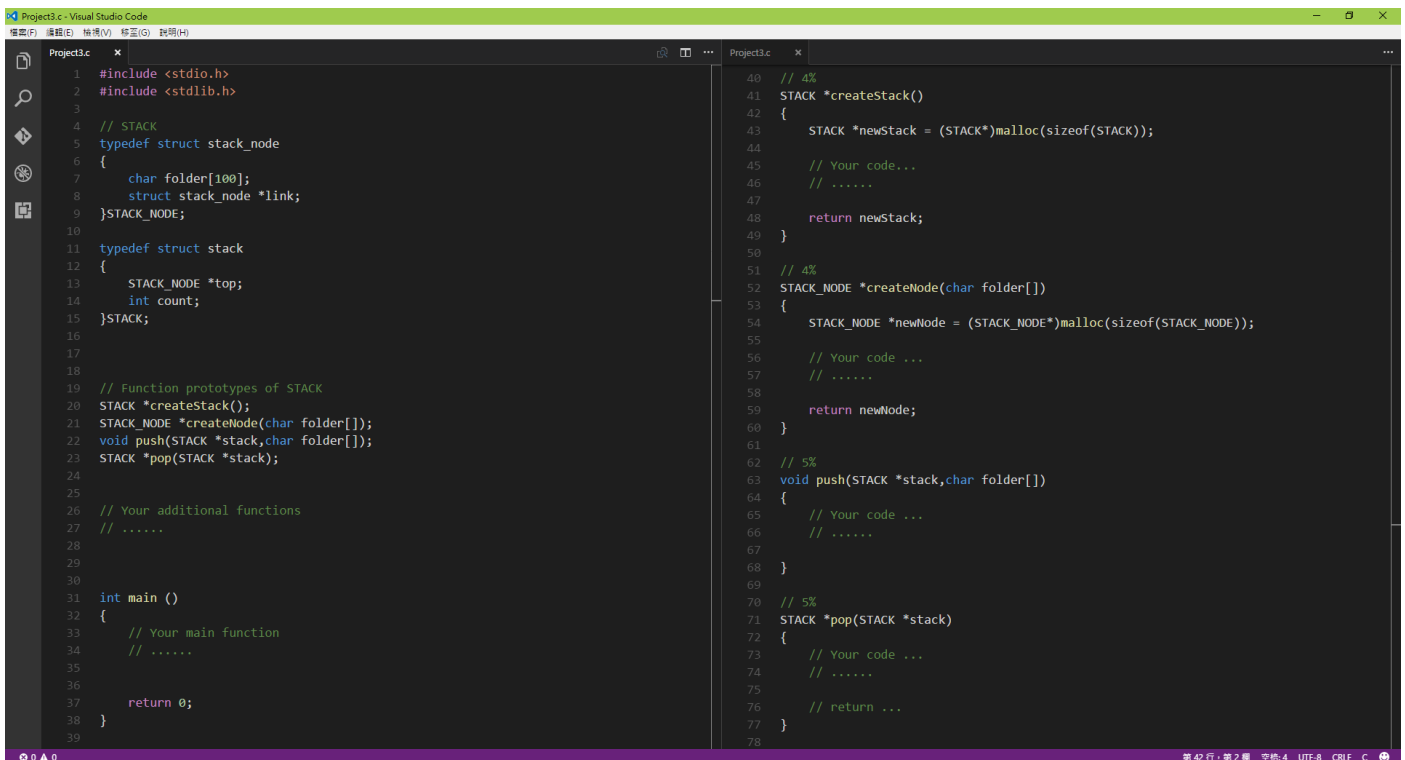
Microsoft Disk Operating System (MS-DOS)，是早期微軟收購來的作業系統，當時圖形化介面 (Graphical user interface, GUI) 還沒很盛行，所有的操作比需透過指令運行，也就是俗稱的指令欄介面 (Command line interface, CLI)。

※問題描述：

請設計一個利用 CLI 執行的檔案系統，包含虛擬的資料夾、檔案，可以利用指令創建資料夾、檔案。指令有 mkdir、rmdir、cd、dir、del 與 find，必須被完成。

※功能需求 – Basic functions：

Stack functions (18%)，在討論區的 Project3.c 檔，完成剩下的程式碼。配分於程式碼中。



```
Project3.c - Visual Studio Code
檔案(F) 編輯(E) 檢視(V) 格式(O) 說明(H)

Project3.c
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 // STACK
5 typedef struct stack_node
6 {
7     char folder[100];
8     struct stack_node *link;
9 }STACK_NODE;
10
11 typedef struct stack
12 {
13     STACK_NODE *top;
14     int count;
15 }STACK;
16
17 // Function prototypes of STACK
18 STACK *createStack();
19 STACK_NODE *createNode(char folder[]);
20 void push(STACK *stack, char folder[]);
21 STACK *pop(STACK *stack);
22
23 // Your additional functions
24 // .....
25
26 int main ()
27 {
28     // Your main function
29     // .....
30
31     return 0;
32 }
33
34
35
36
37
38
39
40 // 4%
41 STACK *createStack()
42 {
43     STACK *newStack = (STACK*)malloc(sizeof(STACK));
44
45     // Your code...
46     // .....
47
48     return newStack;
49 }
50
51 // 4%
52 STACK_NODE *createNode(char folder[])
53 {
54     STACK_NODE *newNode = (STACK_NODE*)malloc(sizeof(STACK_NODE));
55
56     // Your code ...
57     // .....
58
59     return newNode;
60 }
61
62 // 5%
63 void push(STACK *stack, char folder[])
64 {
65     // Your code ...
66     // .....
67 }
68
69
70 // 5%
71 STACK *pop(STACK *stack)
72 {
73     // Your code ...
74     // .....
75
76     // return ...
77 }
78
```

檔案系統：mkdir \$folder (6%)，創造名稱為\$folder 的虛擬資料夾指令，例如想在根目錄 (DS:\) 下，新增一個名稱為 A 的資料夾，下達指令：

mkdir A



```
C:\Users\Power\Desktop\project3.exe
DS:\>mkdir A
Folder A created !
DS:\>
```

檔案系統：rmdir \$folder (6%)，刪除名為\$folder 的虛擬資料夾，例如想刪除名為 A 的虛擬資料夾，下達指令：

```
rmdir A
```



```
C:\Users\Power\Desktop\project3.exe
DS:\>mkdir A
Folder A created !
DS:\>rmdir A
Folder A removed !
DS:\>
```

如果不含此資料夾，回傳錯誤訊息。



```
C:\Users\Power\Desktop\project3.exe
DS:\>mkdir A
Folder A created !
DS:\>rmdir B
Error : Folder B not found !
DS:\>
```

ps. 如果資料夾中擁有其它檔案，必須連同檔案一起刪除。

※功能需求 – Main functions：

檔案系統：cd \$folder (5%)，進入此目錄中的\$folder 資料夾，如果沒有此資料夾，回傳錯誤訊息，必須使用 stack 的 push：



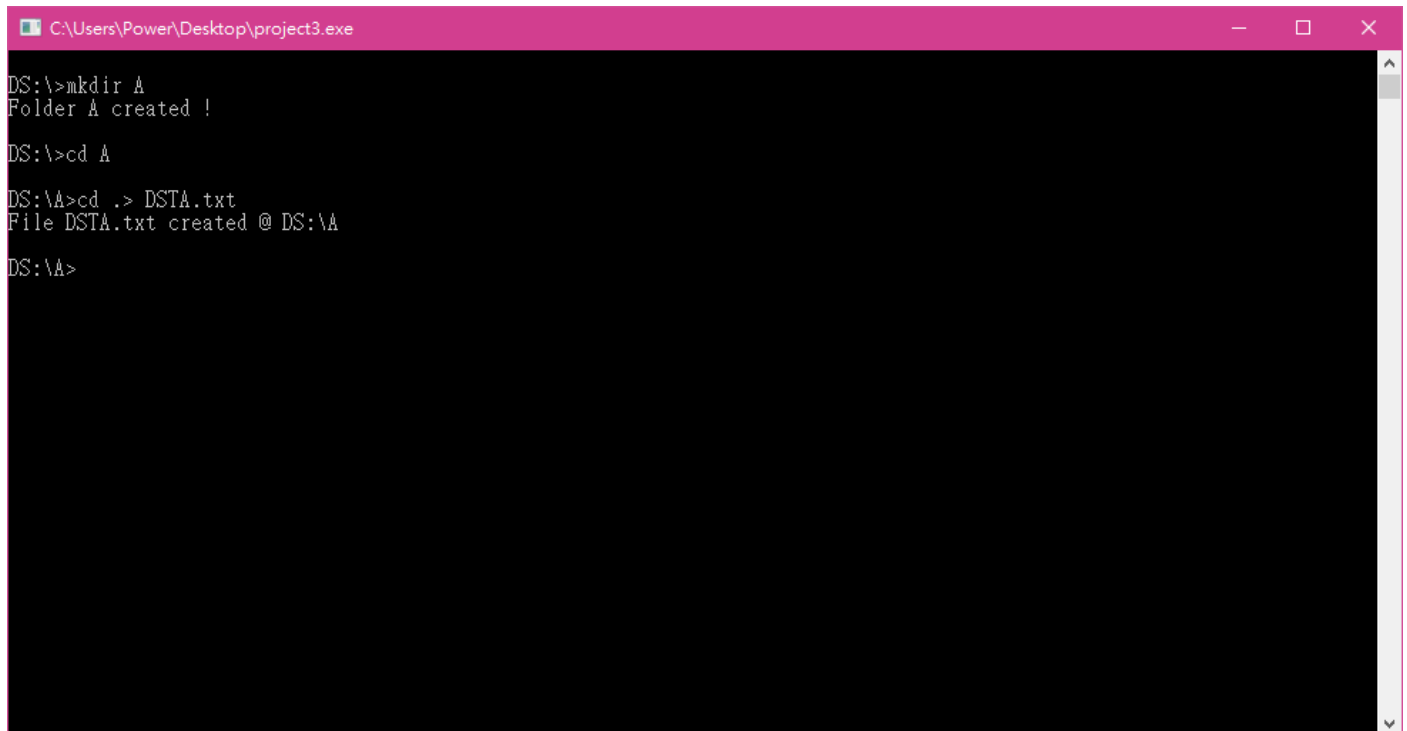
```
C:\Users\Power\Desktop\project3.exe
DS:\>mkdir A
Folder A created !
DS:\>cd A
DS:\A>
```



```
C:\Users\Power\Desktop\project3.exe
DS:\>mkdir A
Folder A created !
DS:\>cd B
Error : Folder B not found !
DS:\>
```

檔案系統：cd .> \$file (3%)，在此資料夾中創立名為\$*file*的虛擬檔案，例如要在目錄底下創立名字為DSTA.txt的檔案，下達指令：

```
cd .> DSTA.txt
```



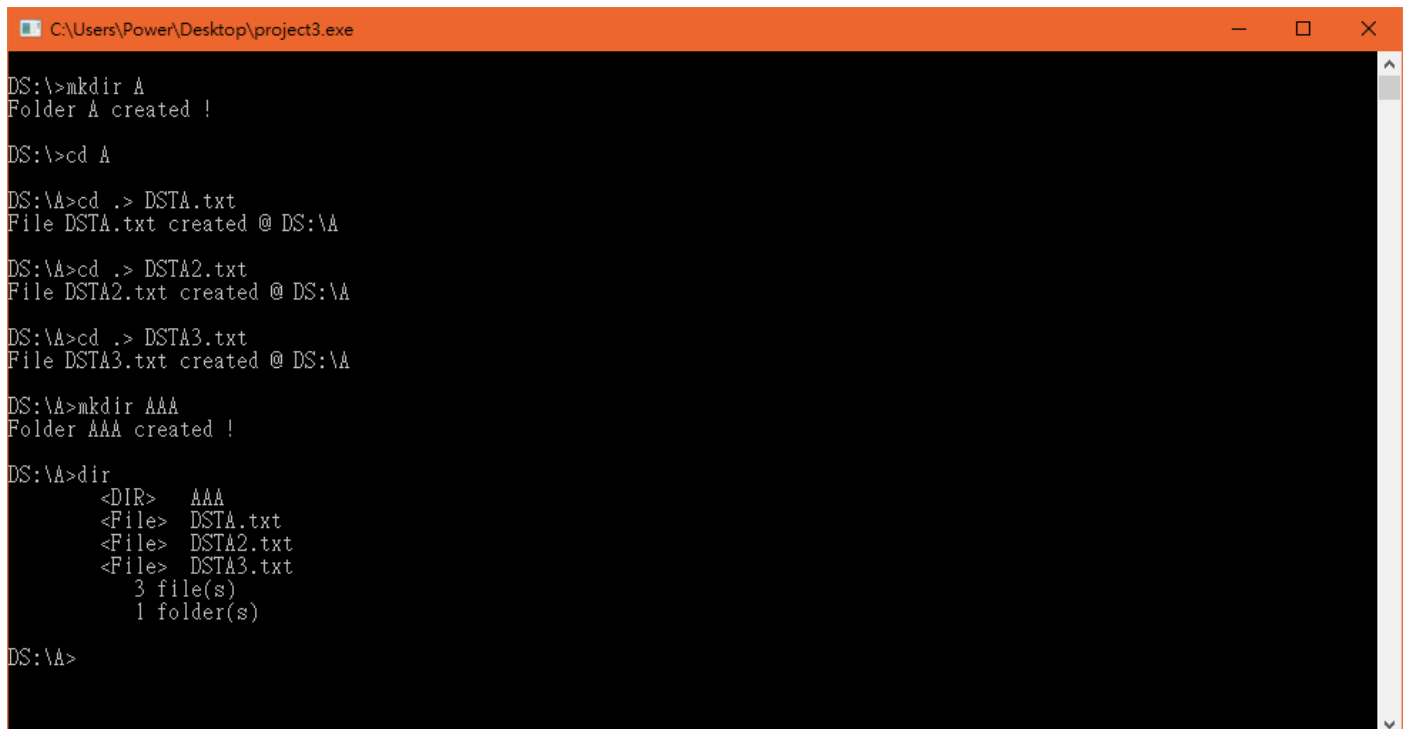
```
C:\Users\Power\Desktop\project3.exe
DS:\>mkdir A
Folder A created !

DS:\>cd A

DS:\A>cd .> DSTA.txt
File DSTA.txt created @ DS:\A

DS:\A>
```

檔案系統：dir (3%)，顯示此資料夾中的所有檔案與資料夾，以及統計檔案數目與資料夾數目：



```
C:\Users\Power\Desktop\project3.exe
DS:\>mkdir A
Folder A created !

DS:\>cd A

DS:\A>cd .> DSTA.txt
File DSTA.txt created @ DS:\A

DS:\A>cd .> DSTA2.txt
File DSTA2.txt created @ DS:\A

DS:\A>cd .> DSTA3.txt
File DSTA3.txt created @ DS:\A

DS:\A>mkdir AAA
Folder AAA created !

DS:\A>dir
<DIR>    AAA
<File>   DSTA.txt
<File>   DSTA2.txt
<File>   DSTA3.txt
          3 file(s)
          1 folder(s)

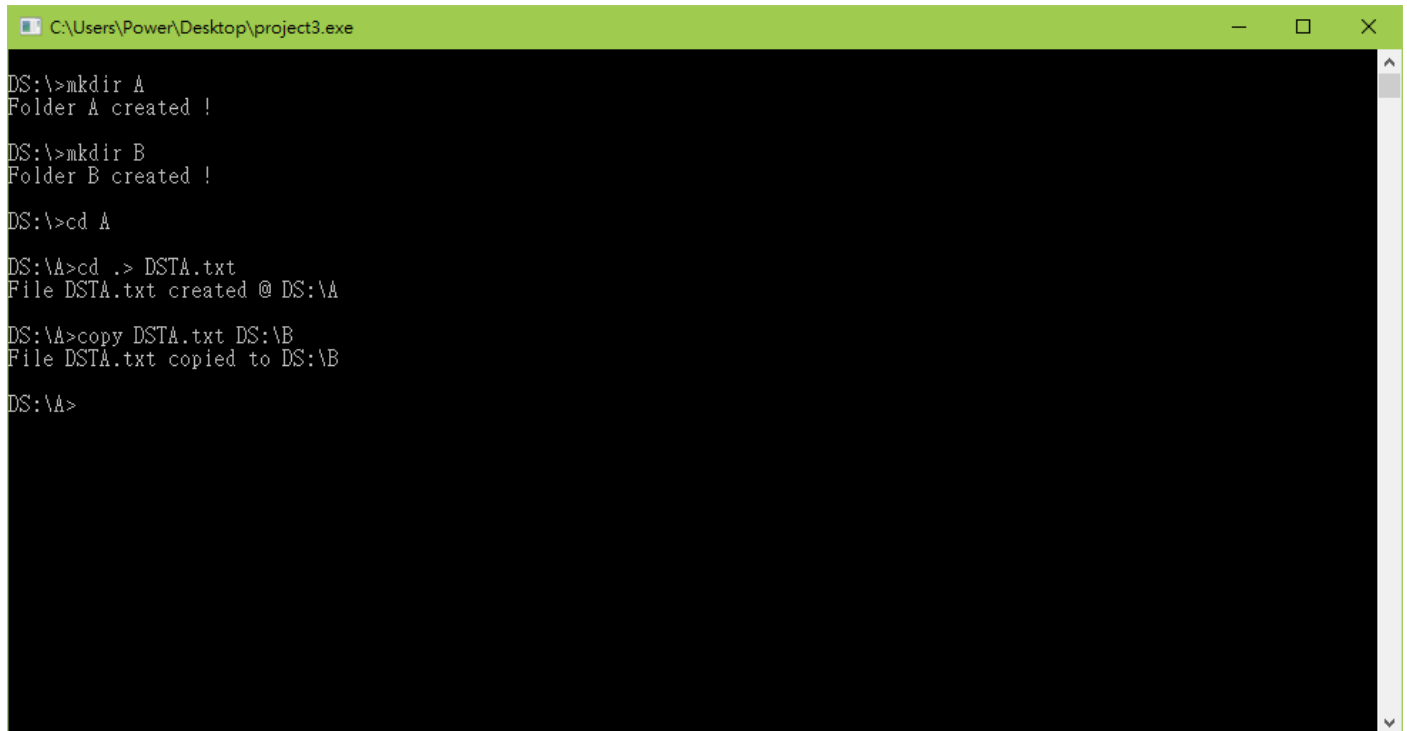
DS:\A>
```

檔案系統：del \$file (2%)，刪除目錄中檔名為\$file的檔案，如果沒有檔案回傳錯誤訊息。

檔案系統：cd .. (4%)，回到上一層目錄，必須使用 stack 中的 pop。

檔案系統：find \$file (5%)，尋找\$file檔名，並列出所存在的資料夾路徑為何。

檔案系統：copy \$file \$path (3%)，複製檔案至\$path路徑。

A screenshot of a Windows command prompt window titled "C:\Users\Power\Desktop\project3.exe". The window has a green title bar and standard Windows window controls. The command history is as follows:

```
DS:\>mkdir A
Folder A created !

DS:\>mkdir B
Folder B created !

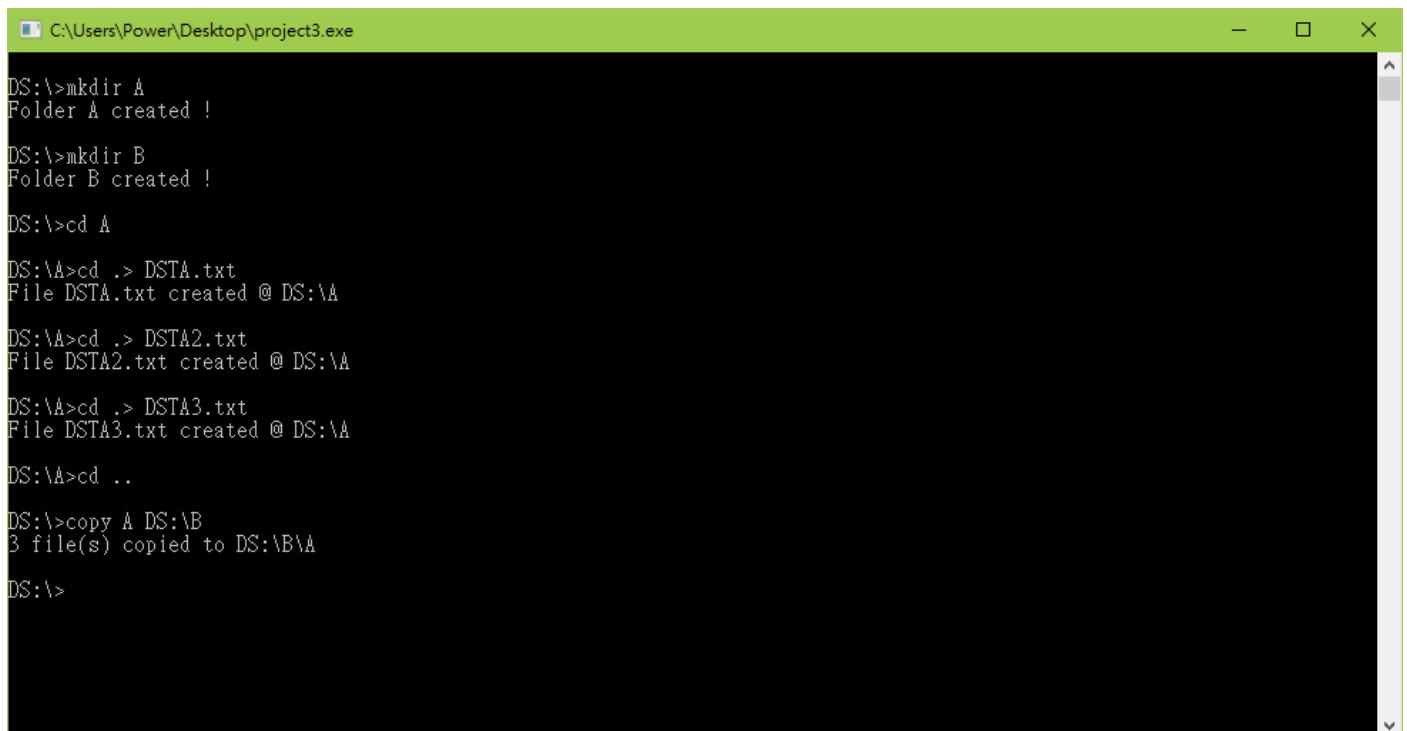
DS:\>cd A

DS:\A>cd .> DSTA.txt
File DSTA.txt created @ DS:\A

DS:\A>copy DSTA.txt DS:\B
File DSTA.txt copied to DS:\B

DS:\A>
```

檔案系統：copy \$folder \$path (5%)，複製資料夾至\$path路徑，資料夾中的檔案必須被複製。

A screenshot of a Windows command prompt window titled "C:\Users\Power\Desktop\project3.exe". The window has a green title bar and standard Windows window controls. The command history is as follows:

```
DS:\>mkdir A
Folder A created !

DS:\>mkdir B
Folder B created !

DS:\>cd A

DS:\A>cd .> DSTA.txt
File DSTA.txt created @ DS:\A

DS:\A>cd .> DSTA2.txt
File DSTA2.txt created @ DS:\A

DS:\A>cd .> DSTA3.txt
File DSTA3.txt created @ DS:\A

DS:\A>cd ..

DS:\>copy A DS:\B
3 file(s) copied to DS:\B\A

DS:\>
```

#### ※功能需求 – Extra Bonus：

所有創建資料夾、檔案；刪除資料夾、檔案，必須實際在電腦中的 C 槽 DS 資料夾下，產生相對應的檔案系統結構。

※配分 (110%) :

Basic functions (30%) :

Stack functions (18%)

檔案系統 : mkdir \$folder (6%)

檔案系統 : rmdir \$folder (6%)

Main functions (30%) :

檔案系統 : cd \$folder (5%)

檔案系統 : cd .> \$file (3%)

檔案系統 : dir (3%)

檔案系統 : del \$file (2%)

檔案系統 : cd .. (4%)

檔案系統 : find \$file (5%)

檔案系統 : copy \$file \$path (3%)

檔案系統 : copy \$folder \$path (5%)

Extra bonus (10%) :

所有指令必須於「C:\DS\」資料夾中，實際執行。

Documents + Interview (40%) :

Design Document (flow chart, functions)

User manual (how to use your project)

Time log & team work (every team member)

Interview (every team member)

※說明 :

1. 關閉程式後，不需要記錄所有的虛擬資料夾、檔案。也就是說不需要讀檔。
2. 資料夾、檔案可以使用 Linked list 實作，比較簡單（無限 node）。
3. Bonus 可以使用 system() 來開發。

※條件限制 :

1. 所有的 cd 指令必須使用 Stack，到新目錄使用 push，回到上一個目錄 pop。
2. 必須使用 C 語言實作。
3. 禁止使用「全域變數」。

※重要 :

1. Deadline : 2016/12/8(四) 上午 10 點，以 e-course 最後上傳的版本為主，不接受遲交。
2. Compile 不過，Project 0 分，上傳之前自己要對自己的 Project 負責，整合後務必自行測試（做不出來的功能就先 Comment 起來，以免整個 Project 沒有分數）。
3. Interview deadline : 2016/12/15(四) 下午 3 點前。
4. Interview 時間最晚必須提早前一天與助教約，助教答覆後才算約成。遲到者該次 Interview 5 折，遲到超過 30 分鐘或爽約該次 Interview 0 分。
5. 需符合以上條件限制，功能才算達成。

## 補充

DOS 指令參考：

<http://www.computerhope.com/msdos.htm>