請放到Codeblocks看註解=D

typedef struct node

{

struct node \* prev;

struct node \* next;

} dllNode\_t;

void DLL\_init(dllNode\_t \* head); //建構一個空的 list

int DLL\_isEmpty(dllNode\_t \*head); // 傳進我的head1或是裡面沒存data的listone, 如果前一個跟後一個都是空的就回傳1

dllNode\_t \* DLL\_next\_node(dllNode\_t \* node); // 回傳下一個節點

dllNode\_t \* DLL\_prev\_node(dllNode\_t \* node); // 回傳上一個節點

unsigned int DLL\_num\_nodes(dllNode\_t \*head); // 傳進我的listone(裡面沒data), 先跑到尾巴, 再往前記數

void DLL\_add\_before(dllNode\_t \* new\_node, dllNode\_t \* head); // 這邊的head永遠都是傳進去第一個(除了最一開始傳進來時會沒data, 其他的時候有data),

void DLL\_add\_tail(dllNode\_t \* new\_node, dllNode\_t \*head); //(將新node加入到listone的後一個)

void DLL\_delete(dllNode\_t \* node); //(從node所在的 Linked List 中刪除此點, 有三種情況, 要刪除的點分別有, 頭, 尾, 中間的點)

void DLL\_concate(dllNode\_t \*firstList, dllNode\_t \* secondList); //存的時候統一都用add\_tail, 這樣listone和listtwo本身都不會有data, 然後把它們丟進來, (將secondList 串在firstList之後, 讓firstlist走到尾巴後, 讓下一個點等於secondlist的next)