

Data Structure HW2

讀檔

Step1.將執行檔和測試資料放在同一個資料夾

```
#include <stdio.h>

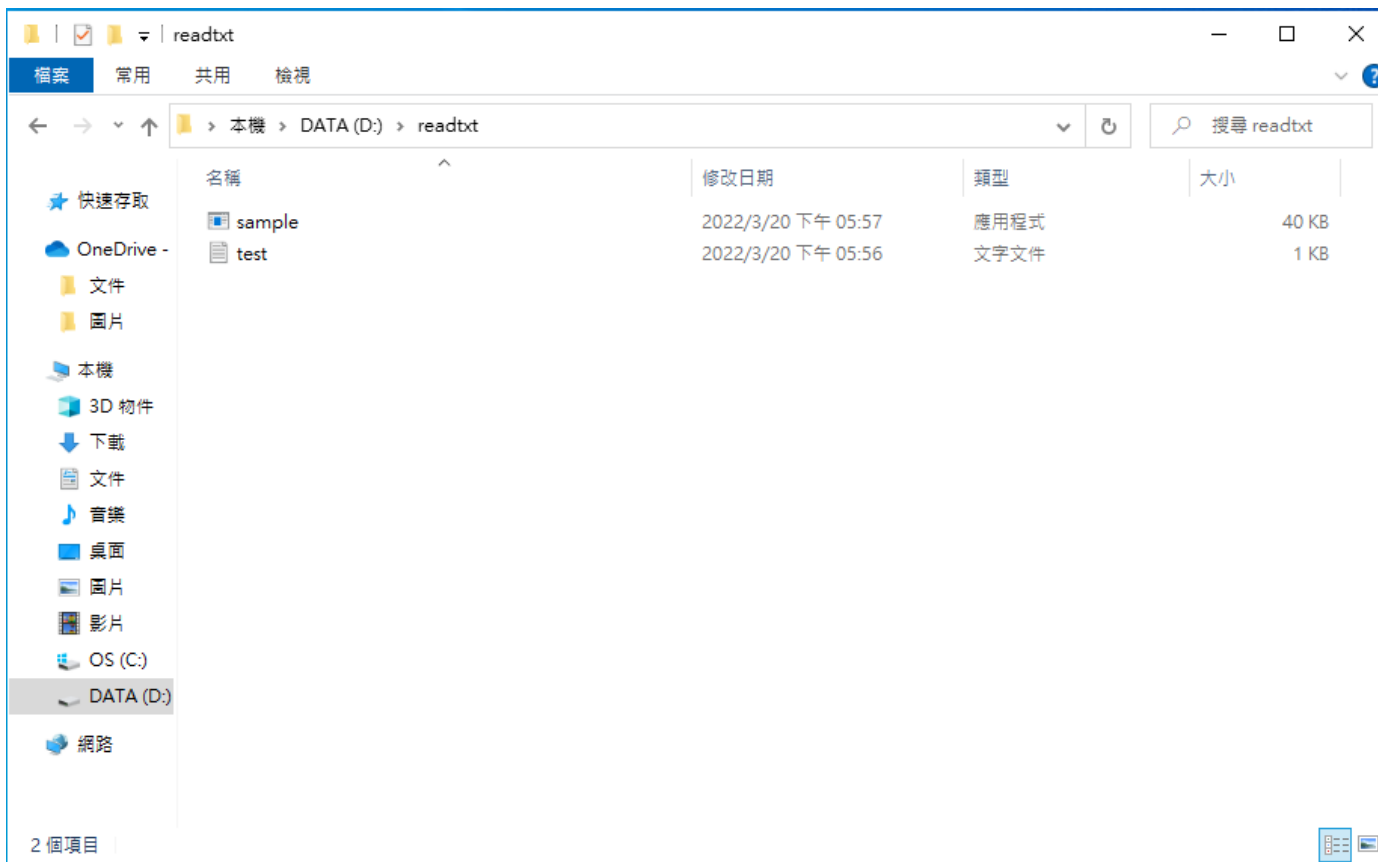
int main()
{
    char input;
    while (scanf("%c",&input)==1) {
        printf("%c", input);
    }
}
```

test - 記事本

檔案(F) 編輯(E) 格式(O) 檢視(V) 說明

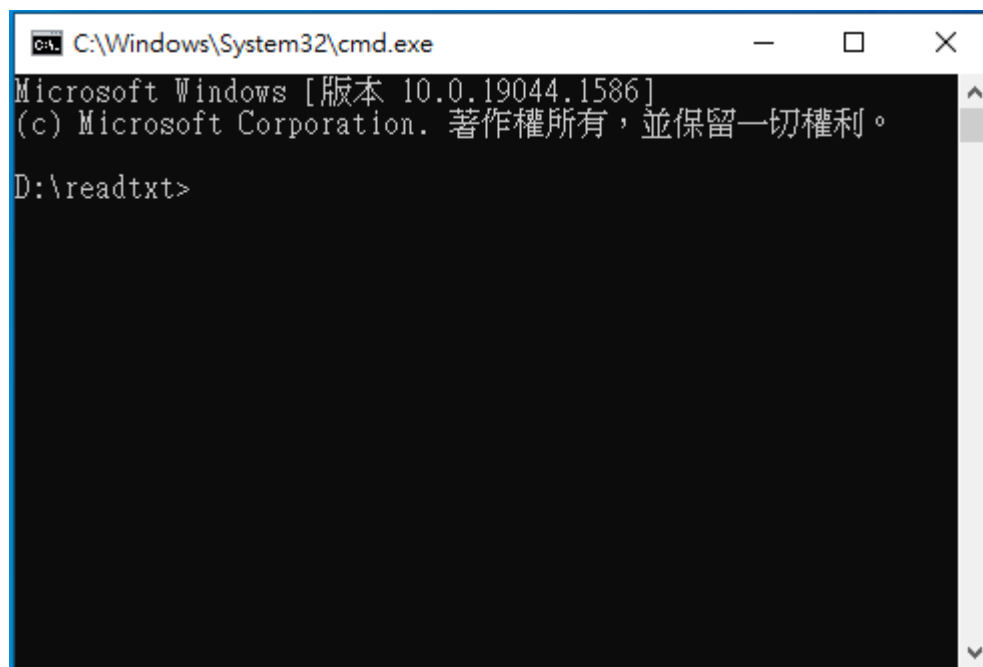
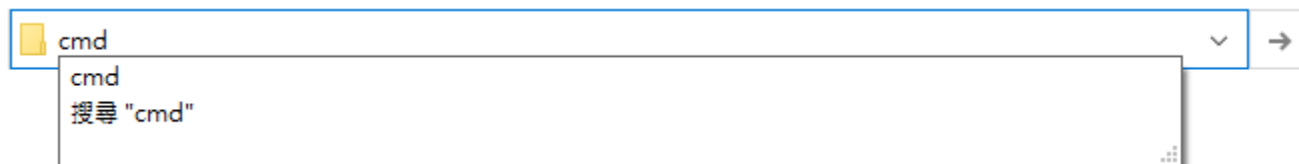
one
two
three
four
five

第 6 100% Windows (CRLF) UTF-8



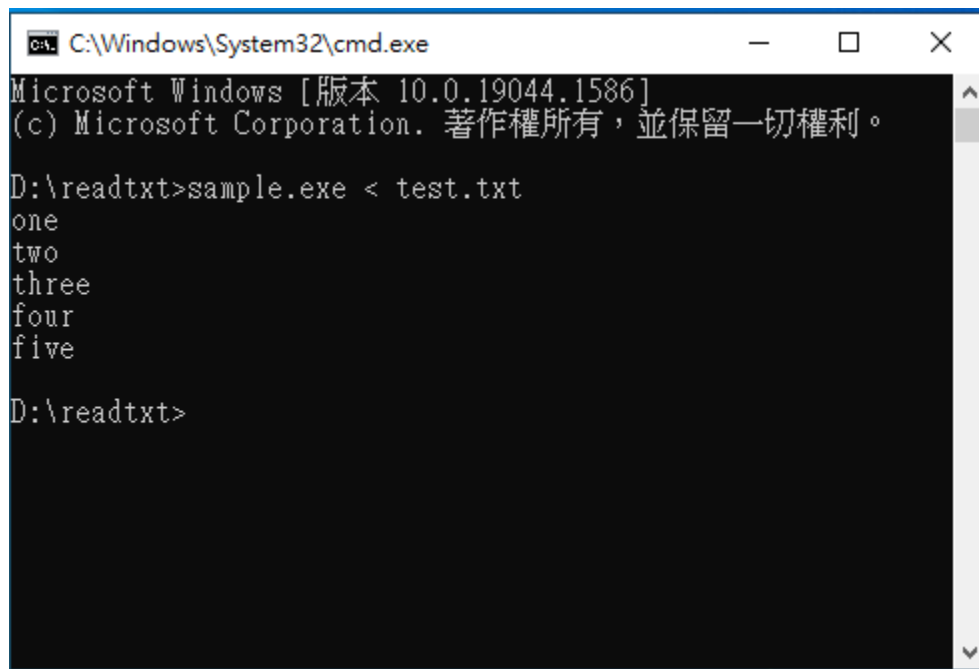
讀檔

Step2.開啟cmd並設定路徑在該資料夾(在路徑欄位打cmd後按Enter)



讀檔

Step3.輸入指令(執行檔檔名.exe < 測試檔檔名.txt)



```
C:\Windows\System32\cmd.exe
Microsoft Windows [版本 10.0.19044.1586]
(c) Microsoft Corporation. 著作權所有，並保留一切權利。

D:\readtxt>sample.exe < test.txt
one
two
three
four
five

D:\readtxt>
```

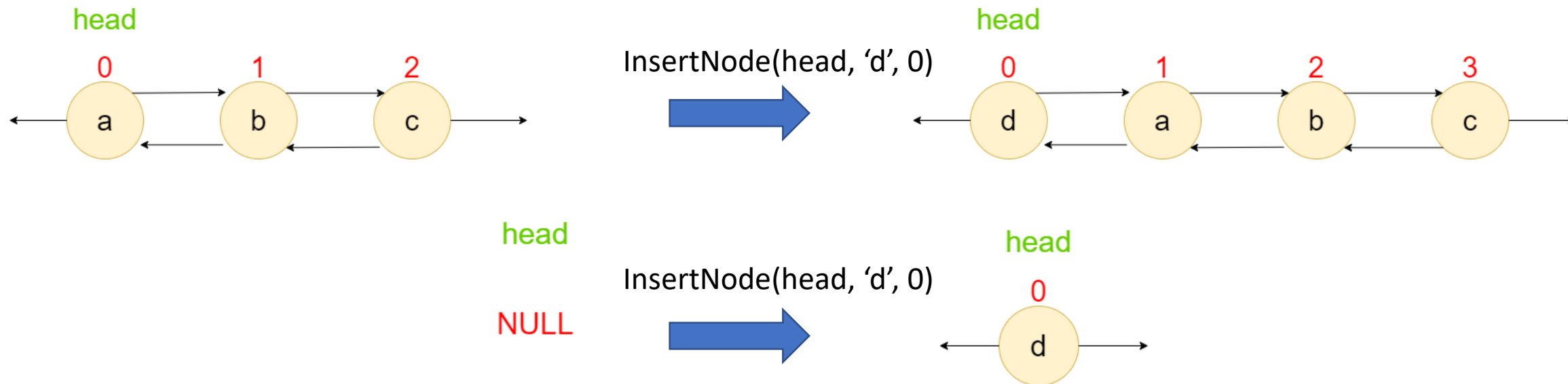
1.雙向鏈結串列

1. `node* InsertNode(node* head, char data, int pos):` 插入一個資料為data的node
2. `node* DeleteNode(node* head, int pos):` 刪除位於pos的node
3. `node* DeleteData(node* head, char data):` 刪除linked list內所有資料為data的node
4. `void PrintList(node* head):` 印出linked list內容
5. `void NumOfOccur(node* head, char data):` 找出資料為data的node總數量
6. `void MaxNumOfOccur(node* head):` 找出node數量最多的data
7. `void SearchWord(node* head, char* word):` 找出word是否存在linked list內

1.雙向鏈結串列

```
node* InsertNode(node* head, char data, int pos)
```

將資料為data的node加在編號pos的node前面(成為新的Linked List的編號第pos個node)



格式規範

InsertNode

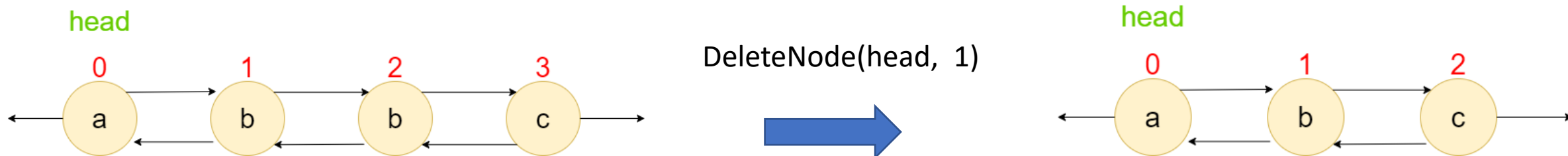
Linked list	data	pos	result
ANY	NOT ASCII 32~126	ANY	印出" Insert wrong data"
NULL	ASCII 32~126	0	插入資料為data的node成為Linked list的head
NULL	ASCII 32~126	NOT 0	印出" Insert wrong position"
N node	ASCII 32~126	0~N	插入資料為data的node成為新的Linked list的編號pos的node
N node	ASCII 32~126	NOT 0~N	印出" Insert wrong position"

<https://zh.wikipedia.org/wiki/ASCII>

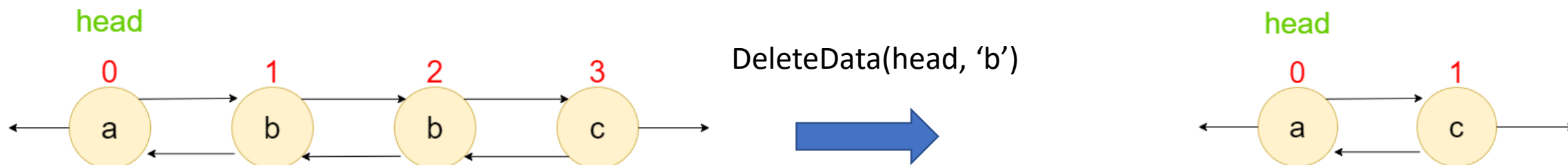
1. 雙向鏈結串列

`node* DeleteNode(node* head, int pos)`

- 刪除位於編號pos的node



`node* DeleteData(node* head, char data)`



格式規範

DeleteNode

Linked list	pos	result
NULL	ANY	印出 "Linked list is empty"
N node	NOT 0~N-1	印出 "Delete wrong position"
N node	0~N-1	刪除第pos個node

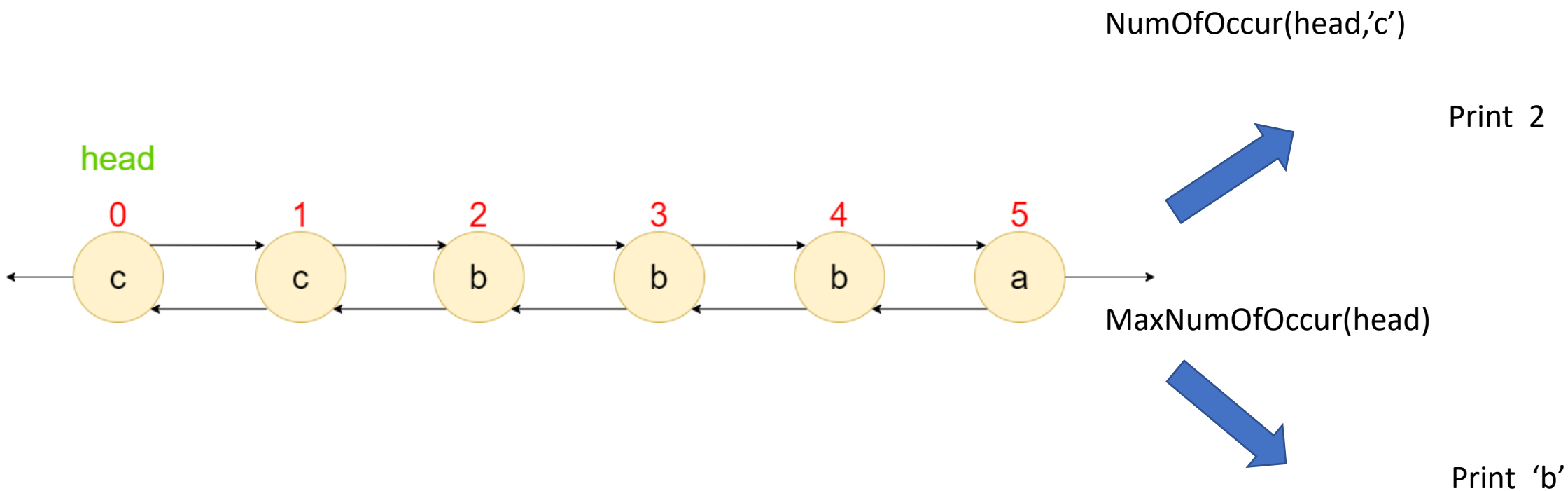
DeleteData

Linked list	data	result
NULL	ANY	印出 "Linked list is empty"
N node	NOT ASCII 32~126	印出 "Delete wrong data"
N node	ASCII 32~126	刪除所有資料為data的node

1.雙向鏈結串列

```
void NumOfOccur(node* head, char data)
```

```
void MaxNumOfOccur (node* head)
```



格式規範

NumOfOccur

Linked list	data	result
NULL	ANY	印出 “Linked list is empty”
N node	NOT ASCII 32~126	印出” Search wrong data”
N node	ASCII 32~126	找出資料為data的node出現次數

MaxNumOfOccur

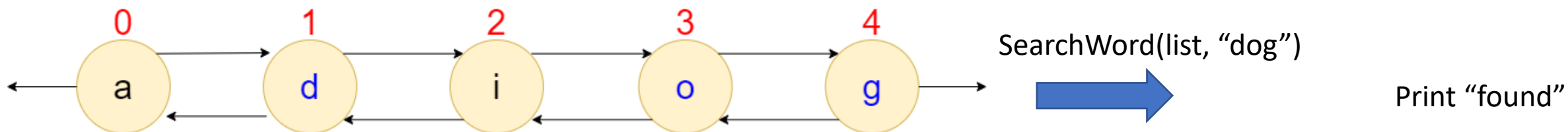
Linked list	result
NULL	印出 “Linked list is empty”
N node	找出出現次數最多的data，若有多個則都要印出，中間隔一個空白字元

1.雙向鏈結串列

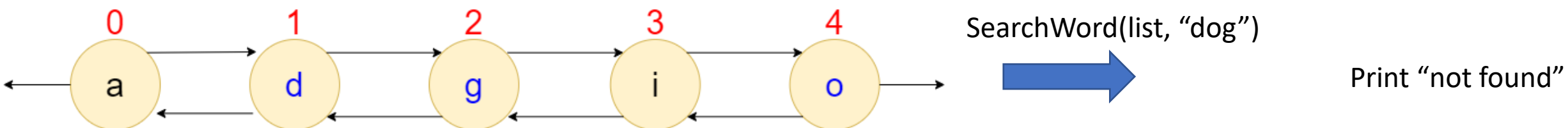
```
void SearchWord(node* head, char* word)
```

找出word內所有字母是否都在linked list內部(需按順序，不用連續)

head



head



格式規範

SearchWord

Linked list	word	result
NULL	ANY	印出 “Linked list is empty”
N node	NOT ASCII 32~126	有任何字元不符合規定，印出” Search wrong word”
N node	ASCII 32~126	找出word內每一個char是否存在於Linked list 須按順序，不用連續 若找到印出”found” ，否則印出”not found”

測試資料

```
test - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明
InsertNode a 0
InsertNode a 1
InsertNode p 2
InsertNode p 3
InsertNode p 4
InsertNode l 5
InsertNode e 6
InsertNode z 7
InsertNode w 8
InsertNode x 0
InsertNode y 0
PrintList
SearchWord apple
DeleteNode 0
PrintList
SearchWord apple
DeleteData p
PrintList
SearchWord apple
NumOfOccur a
MaxNumOfOccur
PrintList
```

`PrintList (node* head)`

從頭開始印出linked list所有資料，中間間隔一個空白字元

```
C:\Windows\System32\cmd.exe
Microsoft Windows [版本 10.0.19044.1586]
(c) Microsoft Corporation. 著作權所有，並保留一切權利。

D:\>hw2.exe < test.txt
y x a a p p p l e z w
found
x a a p p p l e z w
found
x a a l e z w
not found
2
a
x a a l e z w

D:\>
```

2.大數運算

將兩筆輸入資料(可能是正數或負數)相加或是相減，輸出運算結果

測試資料

```
test - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明
123456789123456789123456789
+
987654321987654321987654321|
第 3 列, 第 28 行 100% Windows (CRLF) UTF-8
```

```
C:\Windows\System32\cmd.exe
Microsoft Windows [版本 10.0.19044.1586]
(c) Microsoft Corporation. 著作權所有，並保留一切權利。

D:\data>hw2_2.exe < test.txt
11111111111111111111111111111110

D:\data>
```

```
test - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明
-987654321123456789987654321
-
123456789987654321123456789
第 4 列, 第 1 行 100% Windows (CRLF) UTF-8
```

```
C:\Windows\System32\cmd.exe
Microsoft Windows [版本 10.0.19044.1586]
(c) Microsoft Corporation. 著作權所有，並保留一切權利。

D:\data>hw2_2.exe < test.txt
11111111111111111111111111111110

D:\data>hw2_2.exe < test.txt
-864197531135802468864197532

D:\data>
```

要注意運算結果不能印出多餘的0
e.g. 1000 – 999必須印出1而非0001

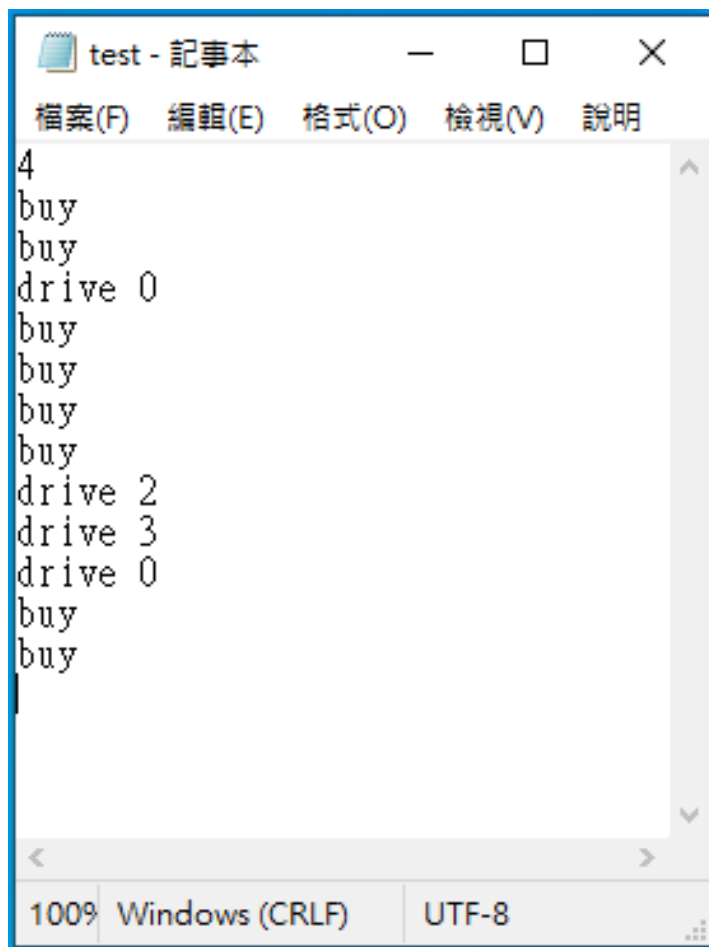
3.富翁的車庫

有一位富翁購買了一塊土地作為私人車庫，車庫內可以停放 N 輛車，富翁有時會選擇駕駛車庫內的一輛車出去兜風，有時則會購買新車，並給新車編號(第一輛購買的車編號為0，第二輛購買的車編號為1，以此類推)。當購買新車時車庫滿了則選擇車庫內的一輛舊車賣出，選擇的標準為從購買當時往回推算最久沒有開的車(購買時也算開過一次)。

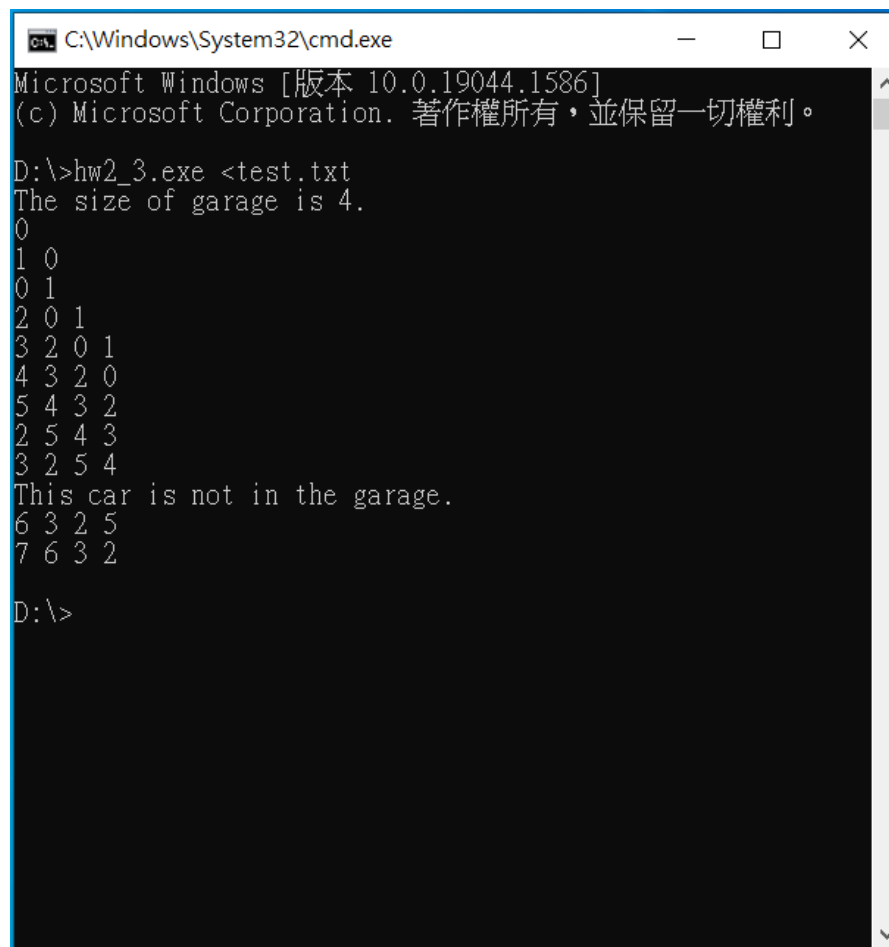
格式規範

- 先讀取車庫大小(1~100) 並印出“The size of garage is N.”
- 購買新車輸入“buy” ，駕駛某輛車輸入“drive ”+汽車編號
- 請印出富翁每次行為後車庫內所有車的編號(沒有順序限制)
- 若想要駕駛的車不在車庫內則印出“This car is not in the garage.”

測試資料



```
test - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明
4
buy
buy
drive 0
buy
buy
buy
buy
drive 2
drive 3
drive 0
buy
buy
100% Windows (CRLF) UTF-8
```



```
C:\Windows\System32\cmd.exe
Microsoft Windows [版本 10.0.19044.1586]
(c) Microsoft Corporation. 著作權所有，並保留一切權利。
D:\>hw2_3.exe <test.txt
The size of garage is 4.
0
1 0
0 1
2 0 1
3 2 0 1
4 3 2 0
5 4 3 2
2 5 4 3
3 2 5 4
This car is not in the garage.
6 3 2 5
7 6 3 2
D:\>
```

注意事項

1. 請使用C和C++程式語言完成作業
2. 有任何問題請寄信給助教
3. 禁止使用C++ STL函式庫(包括vector)，否則0分
4. 第一題回傳值和參數設定皆為參考，只要能夠正確讀檔和執行功能即可
5. 全部三題請都使用linked list實作，否則0分
6. 若因語法或版本問題導致助教無法執行程式，會請同學帶著電腦來實驗室demo，不會直接拿不到分數

注意事項

C++可以使用string class，但不能用以下member functions

- insert
- erase
- append
- push_back
- pop_back
- += operator
- + operator
- 名稱內有find的functions

C可以使用string.h，同理C++可以使用cstring，但不能使用以下functions

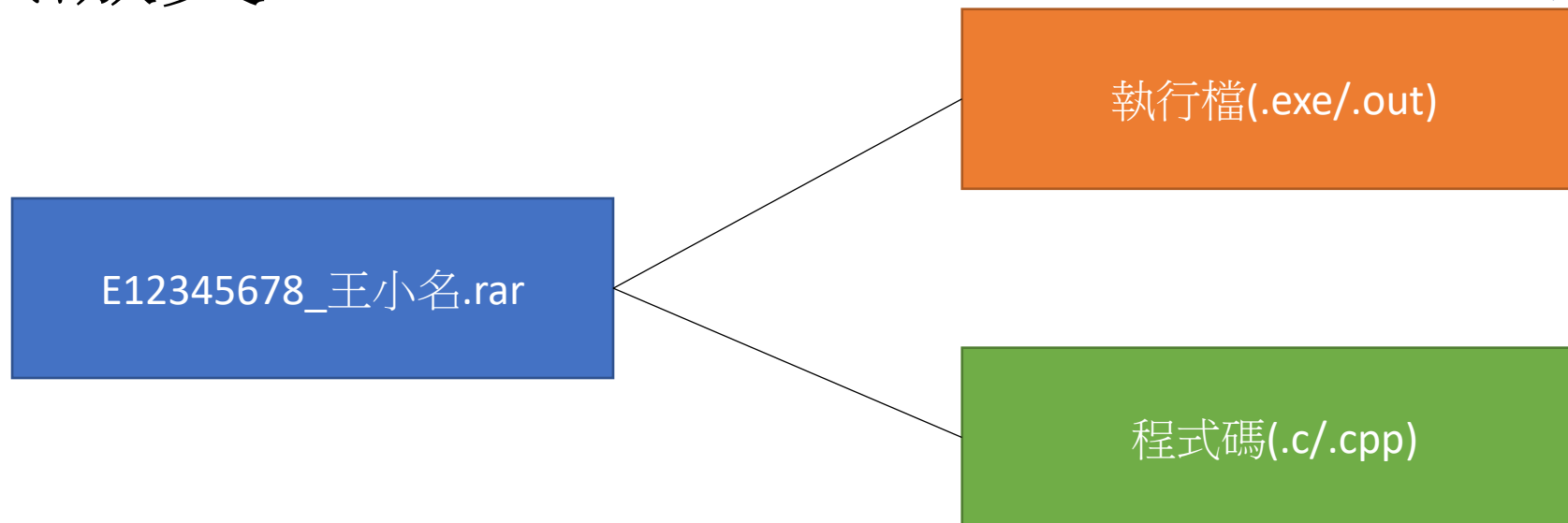
- memmove
- strcat
- strncat
- memchr
- strchr
- strcspn
- strpbrk
- strrchr
- strspn
- strstr

評分標準

- 雙向鏈結串列(40%)
- 大數運算(30%)
- 富翁的車庫(30%)

作業繳交

請確保執行畫面
不會跳掉



格式範例:

hw2_1.cpp/ hw2_2.cpp/ hw2_3.cpp/ hw2_1.exe/ hw2_2.exe/hw2_3.exe

格式錯誤該次作業總成績扣十分

繳交期限:

4/11(一) 23:59前至 moodle 上傳作業

不接受遲交