

韌體實驗第4題

Ting-Yu Lin, Adjunct Lecturer (林庭宇)

Office: AMOS Lab E517

Email: tonylin0413@gmail.com



AMOS Lab.
Advanced Mixed-Operation System Lab.
Dept. of Electrical and Computer Engineering,
Tamkang University, Taiwan



**BIDAS Technology
GrounTruth Workshop**
Computer Vision
Image Processing
Edge Computing

第四題

淡江大學電機工程學系 106 學年度軟體實驗題目

4. 【迴文詞/Palindromes】

輸入一個字串，判斷它是否為迴文串以及鏡像串。輸入字串保證不含數字 0。所謂迴文串就是反轉以後和原串相同，如 abba 和 madam。所謂鏡像串，就是左右鏡像之後和原串相同，如 2S 和 3AIAE。注意，並不是每個字元在鏡像之後都能得到一個合法字元。在本題中，每個字元的鏡像如下表所示（空白項表示該字元鏡像後不能得到一個合法字元）。

Character	Reverse	Character	Reverse	Character	Reverse
A	A	H	H	Y	Y
B		N		Z	5
C		O	O	1	1
D		P		2	S
E	3	Q		3	E
F		R		4	
G		S	2	5	Z
H	H	T	T	6	
I	I	U	U	7	
J	L	V	V	8	8
K		W	W	9	
L	J	X	X		

鏡像字元

輸入的每行包含一個字串（至多只有 20 個字元）；且保證只有上述合法字元，不含空白字元，判斷它是否為迴文串和鏡像串（共 4 種組合）。每組資料之後輸出一個空行。



第四題

輸入 (註 1):

NOTAPALINDROME
ISAPALINILAPASI
2A3MEAS
ATOYOTA

迴文

鏡像

palindFlag

mirrorFlag

0	0
1	0
0	1
1	1

輸出 (註 2):

NOTAPALINDROME -- is not a palindrome.

ISAPALINILAPASI -- is a regular palindrome.

2A3MEAS -- is a mirrored string.

ATOYOTA -- is a mirrored palindrome.

- 輸入字串長度奇數偶數皆可，如果為奇數則以中位數為觀察點左右比對，如果為偶數則左右直接比對。
- 迴文：不一定要鏡像反轉，但左右字元位置要對應。
- 鏡像：一定要鏡像反轉且左右字元位置有對應。

➤ 四種結果：

- ✓ not a palindrome: 不是迴文也不是鏡像
- ✓ a regular palindrome: 是迴文不是鏡像
- ✓ a mirrored string: 不是迴文是鏡像
- ✓ a mirrored palindrome: 是迴文也是鏡像

Character Reverse Character Reverse Character Reverse

A	A	H	H	Y	Y
B		N		Z	5
C		O	O	1	1
D		P		2	S
E	3	Q		3	E
F		R		4	
G		S	2	5	Z
H	H	T	T	6	
I	I	U	U	7	
J	L	V	V	8	8
K		W	W	9	
L	J	X	X		

鏡像字元

第四題

需要修改或自行撰寫的部分

```

1  #define _CRT_SECURE_NO_WARNINGS
2  //[迴文詞/Palindromes](1/2)
3  #define IN "P04IN.txt"
4  #define OUT "P04OUT.txt"
5  //*****
6  #include <iostream>
7  #include <ctime>
8  using namespace std;
9  void redir(void);
10 //*****
11 /* Work Space*/
12 char *rev = "A 3 HIL JM O 2TUVWXY51SE Z 8 ";
13 char *meg[] = { "not a palindrome",
14                 "a regular palindrome",
15                 "a mirrored string",
16                 "a mirrored palindrome" };
17
18 char mirrored(char ch); //傳回字元的鏡像
19 //*****
20 int main(void)
21 {
22     redir(); //redirection
23     //*****

```

- 初行加入crt secure no warnings。
- 記得修改輸入和輸出txt檔案名稱。
- #define為C++直接定義，變數型態會依照定義自動判定。
- redir函式標頭檔。
 - Line 12:
 - ✓ 宣告鏡像指標字元(利用指標找字元)。
 - Line 13-16:
 - ✓ 宣告結果訊息指標字元陣列(利用指標找字串)。
 - Line 18:
 - ✓ 自定義mirrored函式標頭檔。
- 重新導向，將輸入輸出設定為讀取txt檔案。

第四題

```

45  //*****
46      freopen("CON", "r", stdin); //取消重新導向
47      freopen("CON", "w", stdout);
48
49      printf("Time used = %.2f\n", (double)clock() / CLK_TCK); //傳回程式目前為止執行的時間
50
51      system("pause");
52      return 0; //the end...
53  }
54
55  void redir(void)
56  {
57      freopen(IN, "r", stdin);
58      freopen(OUT, "w", stdout);
59  }
60  //*****

```

➤ redir函式。

第四題

```
12 char *rev = "A 3 HIL JM O 2TUVWXY51SE Z 8 "; //注意空白字元的數目
```

- Line 12:
 - ✓ 宣告**鏡像指標字元**(利用指標找字元)。
 - ✓ 也可以使用char rev[] = “...”改寫。

輸入 (註 1):

NOTAPALINDROME
ISAPALINILAPASI
2A3MEAS
ATOYOTA

Character	Reverse	Character	Reverse	Character	Reverse
A	A	M	M	Y	Y
B		N		Z	5
C		O	O	1	1
D		P		2	S
E	3	Q		3	E
F		R		4	
G		S	2	5	Z
H	H	T	T	6	
I	I	U	U	7	
J	L	V	V	8	8
K		W	W	9	
L	J	X	X		

鏡像字元

鏡像

A				3				H	I	L		J	M		O				2	T	U	V	W	X	Y	5	1	S	E		Z			8	
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	1	2	3	4	5	6	7	8	9	

指標陣列 (1/2)

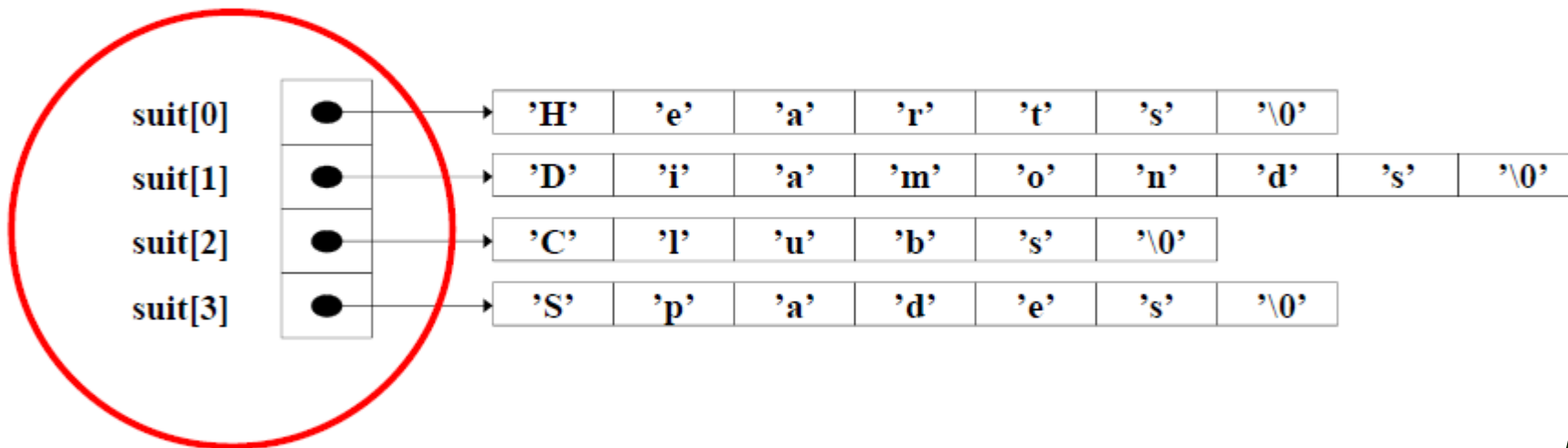
- Arrays can contain pointers
 - Commonly used to store an array of strings
 - E.g. `char *suit[4] = {"Hearts",
"Diamonds",
"Clubs",
"Spades"};`
 - Each element of suit is a pointer to a string (`char *`)
 - The strings are not in the array, only pointers to the strings are in the array

- 每一個suit的元素都是一個pointer 指向string(char*)。
- 字串不在array當中，只有pointers 指向字串時才在array當中。

指標陣列

指標陣列 (2/2)

array suit of type char * \Rightarrow suit is of type char **



- 只有array suit有固定的大小，字串可以是任何大小(長度)。

- array **suit** has a fixed size; but strings can be of any size (length)



指標陣列


- ✓ 字元陣列 = 字元型態的陣列 → (字串)。
- ✓ 指標字元陣列 = 含有指標字元型態的陣列 → (指標字串)。
- Line 13-16:
- ✓ 宣告結果訊息指標字元陣列(利用指標找字串)。

```
13 char *meg[] = { "not a palindrome",
14                 "a regular palindrome",
15                 "a mirrored string",
16                 "a mirrored palindrome" };
17
```

- ✓ meg[0]: "not a palindrome"
- ✓ meg[1]: "a regular palindrome"
- ✓ meg[2]: "a mirrored string"
- ✓ meg[3]: "a mirrored palindrome"

短路運算

需要修改或自行撰寫的部分



```

23  //*****
24  /* Work Space*/
25  char s[21]; //輸入規定至多20個字元
26  int length, p, m, i;
27
28  while (scanf("%s", s) == 1) {
29      length = strlen(s);
30      p = m = 1;
31
32      for (i = 0; i < (double)length / 2; i++) { //注意轉型
33          if (p && s[i] != s[(length - 1) - i]) { //short-circuit
34              p = 0;
35          }
36          if (m && mirrored(s[i]) != s[(length - 1) - i]) {
37              m = 0;
38          }
39      }
40
41      printf("%s -- is %s.\n\n", s, meg[m * 2 + p]);
42  }
43  //*****
    
```

● Line 25:

✓ 宣告 **字元陣列** 型態 s 為輸入字元。

● Line 26:

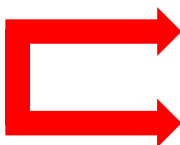
✓ 宣告 **整數** 型態 length, p, m, i。

✓ p **迴文旗標**: 1，初始設定為有迴文。

✓ m **鏡像旗標**: 1，初始設定為有鏡像。

短路運算

需要修改或自行撰寫的部分



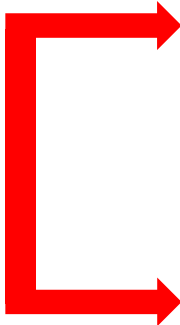
```

23  //*****
24  /* Work Space*/
25  char s[21]; //輸入規定至多20個字元
26  int length, p, m, i;
27
28  while (scanf("%s", s) == 1) {
29      length = strlen(s);
30      p = m = 1;
31
32      for (i = 0; i < (double)length / 2; i++) { //注意轉型
33          if (p && s[i] != s[(length - 1) - i]) { //short-circuit
34              p = 0;
35          }
36          if (m && mirrored(s[i]) != s[(length - 1) - i]) {
37              m = 0;
38          }
39      }
40
41      printf("%s -- is %s.\n\n", s, meg[m * 2 + p]);
42  }
43  //*****
    
```

- Line 28:
 - ✓ `while`迴圈一次`scanf`一行字串，如果有擷取到則回傳1，`while`判斷式成立。
- Line 29-30:
 - ✓ `strlen()`: 計算字串長度函式，並指定給整數型態變數`length`。
 - ✓ 令整數變數`p = m = 1`。初始設定皆有迴文和鏡像。

短路運算

需要修改或自行撰寫的部分



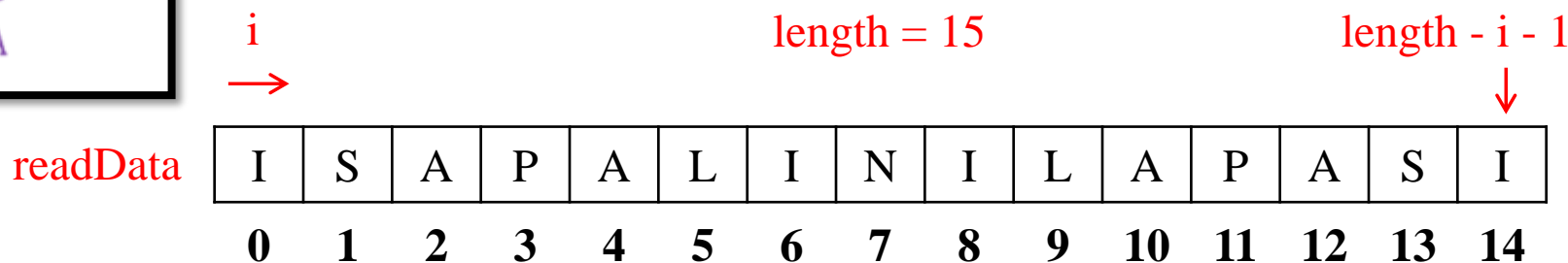
```

23  //*****
24  /* Work Space*/
25  char s[21]; //輸入規定至多20個字元
26  int length, p, m, i;
27
28  while (scanf("%s", s) == 1) {
29      length = strlen(s);
30      p = m = 1;
31
32      for (i = 0; i < (double)length / 2; i++) { //注意轉型
33          if (p && s[i] != s[(length - 1) - i]) { //short-circuit
34              p = 0;
35          }
36          if (m && mirrored(s[i]) != s[(length - 1) - i]) {
37              m = 0;
38          }
39      }
40
41      printf("%s -- is %s.\n\n", s, meg[m * 2 + p]);
42  }
43  //*****
    
```

- Line 32:
 - ✓ for迴圈限定範圍在0到(double)length / 2
→ 除出來才有小數點，奇數偶數分別設置，如果length為奇數會比偶數多跑一次for迴圈到中位數位置。
- Line 33-35:
 - ✓ 判斷是否為迴文。
- Line 36-38:
 - ✓ 判斷是否為鏡像。

迴文判斷

輸入 (註1):
NOTAPALINDROME
ISAPALINILAPASI
2A3MEAS
ATOYOTA



```

32 for (i = 0; i < (double)length / 2; i++) { //注意轉型
33     if (p && s[i] != s[(length - 1) - i]) { //short-circuit
34         p = 0;
35     }
36     if (m && mirrored(s[i]) != s[(length - 1) - i]) {
37         m = 0;
38     }
39 }
40
41 printf("%s -- is %s.\n\n", s, meg[m * 2 + p]);
42 }
43 //*****

```

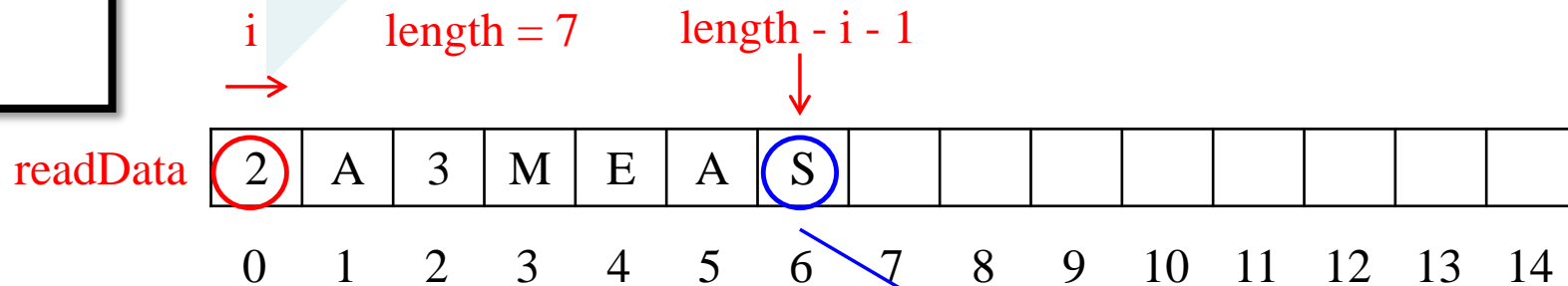
- Line 33-35:
 - ✓ 預設未判斷前都令字串為迴文以及鏡像。
 - ✓ 如果[p為1(預設已是True)] && [第i字母不等於第 (length - i) - 1 字母(True)]，則p = 0不為迴文。

輸入 (註1):
NOTAPALINDROME
ISAPALINILAPASI
2A3MEAS
ATOYOTA

程式 詳解

輸出 結果

鏡像判斷



```
32 for (i = 0; i < (double)length / 2; i++) { //注意轉型
33     if (p && s[i] != s[(length - 1) - i]) { //short-circuit
34         p = 0;
35     }
36     if (m && mirrored(s[i]) != s[(length - 1) - i]) {
37         m = 0;
38     }
39 }
40
41 printf("%s -- is %s.\n\n", s, meg[m * 2 + p]);
42 }
43 //*****
```

mirrored()

● Line 36-38:

- ✓ 預設未判斷前都令字串為迴文以及鏡像。
- ✓ 如果[m為1(預設已是True)] && [第i字母的鏡像不等於第 (length - i) - 1 字母(True)]，則m = 0不為鏡像。

✓ Line 41:

- ✓ 顯示結果，注意旗標順序。

```
13 char *meg[] = { "not a palindrome",
14                 "a regular palindrome",
15                 "a mirrored string",
16                 "a mirrored palindrome" };
```

鏡像 迴文
mirrorFlag palindFlag

0	0
0	1
1	0
1	1

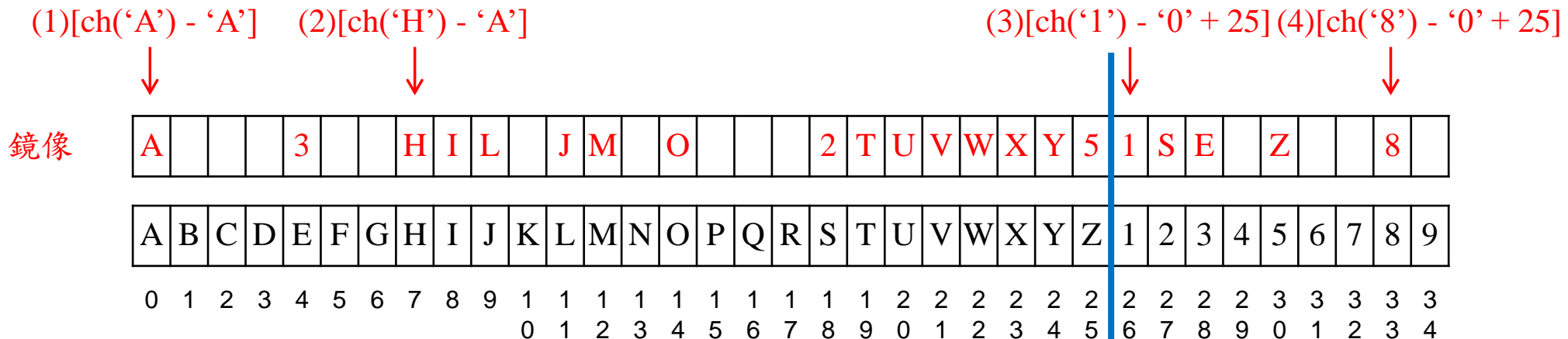
鏡像字母數字回傳

需要修改或自行撰寫的部分

```

60 // [迴文詞/Palindromes](2/2)
61 /* Work Space */
62 char mirrored(char ch) // 傳回字元的鏡像
63 {
64     if (isalpha(ch)) {
65         return rev[ch - 'A'];
66     } else {
67         return rev[ch - '0' + 25];
68     }
69 }
    
```

- Line 62-69: 鏡像自定義函式
- ✓ isalpha() 判斷是否為英文字母函式
- ✓ Line 65: 回傳 **字母** 鏡像結果
- ✓ Line 67: 回傳 **數字** 鏡像結果



```
12 char *rev = "A 3 HIL JM O 2TUVWXY51SE Z 8 "; // 注意空白字元的數目
```

鏡像字母數字回傳

需要修改或自行撰寫的部分

```

60 // [迴文詞/Palindromes] (2/2)
61 /* Work Space */
62 char mirrored(char ch) // 傳回字元的鏡像
63 {
64     if (isalpha(ch)) {
65         return rev[ch - 'A'];
66     } else {
67         return rev[ch - '0' + 25];
68     }
69 }

```

- 指標字元陣列

- ✓ ch為**字母**時ch在rev中為ascii碼
- ✓ (1)[ch('A') - 'A'] → **rev [65 - 65] → rev [0]**
- ✓ (2)[ch('H') - 'A'] → **rev [72 - 65] → rev [7]**
- ✓ ch為**數字**時ch在rev中為ascii碼
- ✓ (3)[ch('1') - '0' + 25] → **rev [49 - 48 + 25] → rev [26]**
- ✓ (4)[ch('8') - '0' + 25] → **rev [56 - 48 + 25] → rev [33]**

Diagram illustrating the generation of a 1D barcode from a 2D barcode. The 2D barcode is a 10x26 grid of cells. The top row contains the letters A through Z. The bottom row contains the numbers 0 through 9. A vertical blue line separates the letters from the numbers. The 1D barcode is a single row of 26 cells. The first 10 cells contain the letters A through J, and the next 16 cells contain the numbers 1 through 8, followed by a space, and then the numbers 1 through 8. The 1D barcode is generated by taking the first row of the 2D barcode and applying a transformation: (1) [ch('A') - 'A'], (2) [ch('H') - 'A'], (3) [ch('1') - '0' + 25], and (4) [ch('8') - '0' + 25]. The 1D barcode is labeled "鏡像" (Mirror Image) on the left.

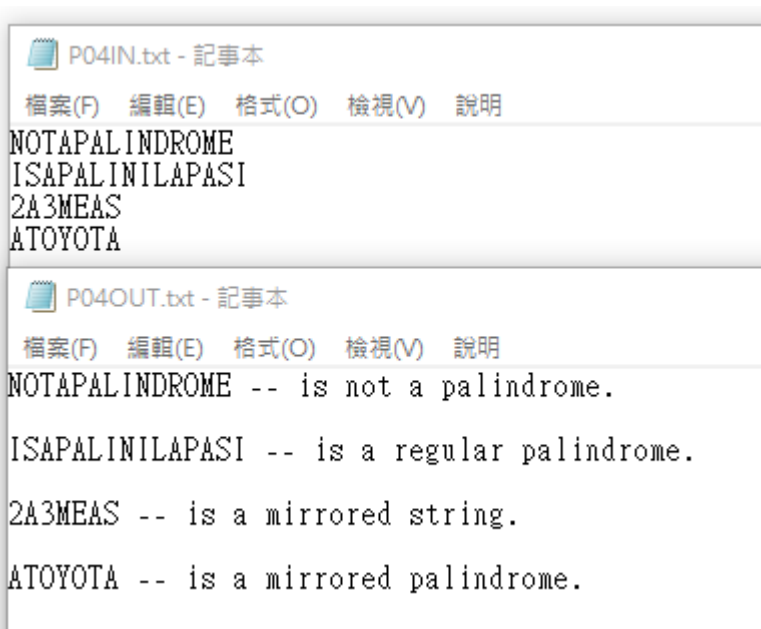
```
12 char *rev = "A 3 HIL JM O 2TUVWXY51SE Z 8 "; //注意空白字元的數目
```


第四題

➤ 解題思考：

- ✓ (1)宣告鏡像指標字元
- ✓ (2)宣告結果訊息指標字元陣列
- ✓ (3)scanf讀取輸入字串，短路運算
- ✓ (4)判斷是否為迴文
- ✓ (5)判斷是否為鏡像
- ✓ (6)自定義鏡像回傳函式
- ✓ (7)顯示輸出注意指標字元陣列表示型式

第四題



The screenshot shows two Notepad windows. The top window, titled 'P04IN.txt - 記事本', contains the input strings: NOTAPALINDROME, ISAPALINILAPASI, 2A3MEAS, and ATOYOTA. The bottom window, titled 'P04OUT.txt - 記事本', shows the corresponding output analysis for each string.

```
P04IN.txt - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明
NOTAPALINDROME
ISAPALINILAPASI
2A3MEAS
ATOYOTA

P04OUT.txt - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明
NOTAPALINDROME -- is not a palindrome.
ISAPALINILAPASI -- is a regular palindrome.
2A3MEAS -- is a mirrored string.
ATOYOTA -- is a mirrored palindrome.
```



Q&A

