M	O	N	A	R
C	H	Y	B	D
E	F	G	I/J	K
L	P	Q	S	T
U	V	W	X	Z

Figure 1: Playfair Key Matrix

輸出結果

```
Please input the plaintext:
I liked bubble sort. I meet you.
IL IK ED BU BX BL ES OR TI ME ET YO UX
ES KE KC CX IA CS IL NM SK CL KL HN VZ
C:\Users\joyce\Desktop\組合語言\test\Debug\test.exe (處理序 19876) 已結束，出現代碼 0。
若要在偵錯停止時自動關閉主控台，請啟用 [工具] -> [選項] -> [偵錯] -> [偵錯停止時，自動關閉主控台]。
按任意鍵關閉此視窗...
```

說明

存放資料

```
5       INCLUDE Irvine32.inc
6
7       BUFMAX = 128
8       .data
9       val         BYTE 5
10      val2        WORD ?
11      val3        WORD ?
12      val5        WORD ?
13      val6        WORD ?
14      val4        WORD -1
15      addre       DWORD ?
16      Welc        BYTE "Please input the plaintext:",0
17      Modi        BYTE "Modified plaintext:",0
18      Cipher      BYTE "The ciphertext is:",0
19      Plaintext   BYTE 128 DUP(?)
20      Ciphertext  BYTE 128 DUP(?)
21      Ciphertext2 BYTE 128 DUP(?)
22      Modif       BYTE 128 DUP(?)
23      PlayfairKey BYTE 'M','O','N','A','R'
24      Rowsize = ($ - PlayfairKey)
25              BYTE 'C','H','Y','B','D'
26              BYTE 'E','F','G','I','K'
27              BYTE 'L','P','Q','S','T'
28              BYTE 'U','V','W','X','Z'
29      bufSize     DWORD  ?
```

主程式

先印出提示字，輸入明文，將 key、明文、密文的位子放到 stack，call 加密的 procedure

```
31      .code
32      main PROC
33          mov     edx,OFFSET Welc
34          call    WriteString
35          call    Crlf
36
37          mov     ecx,BUFMAX
38          mov     edx,OFFSET Plaintext
39          call    ReadString
40          ;call   WriteString
41          push    OFFSET PlayfairKey
42          push    OFFSET Ciphertext
43          push    OFFSET Plaintext
44          call    Playfair
45
46          exit
47      main ENDP
```

Playfair 函式

將 ebp 位子放到 stack 保存，esp 移到 ebp，Plaintext、Ciphertext 分別移到 esi、edi，call lowertoCap 將字母改成大寫，Modified 將空格，標點符號移除並改成兩兩放一起，然後印出；Ciphertext2 用來存放最終結果，因為暫存器不夠所以先將位子存到記憶體，61 行是將 key 的位置放到 edi，call ToCipher 將明文轉成密文

```
49      Playfair PROC
50          push    ebp
51          mov     ebp,esp
52          mov     esi,[ebp+8]     ;Plaintext
53          mov     edi,[ebp+12]    ;Ciphertext
54          call    lowertoCap
55          call    Modified
56          mov     edx,edi
57          call    WriteString
58          call    Crlf
59          mov     edi,OFFSET Ciphertext2
60          mov     addre,edi
61          mov     edi,[ebp+16]
62          call    ToCipher
63          mov     edi,OFFSET Ciphertext2
64          mov     edx, edi
65          call    WriteString
66          pop     ebp
67      ret
68      Playfair ENDP
69
```

lowertoCap

將字母改成大寫，並且將 J 改成 I

```
71      lowertoCap PROC
72          mov     ecx,eax
73          push    edi
74      L1:
75          mov     bl, [esi]
76          cmp     bl, 'a'
77          jb      checkU
78          cmp     bl, 'z'
79          ja      checkU
80          sub     bl, 32
81      checkU:
82          cmp     bl,'J'
83          je      toI
84          jmp     con
85      toI:
86          mov     bl,'I'
87      con:
88          cmp     bl, 'A'
89          jb      notUL
90          cmp     bl, 'Z'
91          ja      notUL
92          mov     [edi], bl
93          inc     edi
94      notUL:
95          inc     esi
96          loop    L1
97          pop     edi
98      ret
99      lowertoCap ENDP
```

Modified

將空格，標點符號移除並改成兩兩放一起

107~109 行：確認是否是結尾

110~113 行：確認兩個字是否一樣，相等跳至 addX 加 X，否則跳至 two 放一起

addX：加 X，再加一個空格

two：將兩個字放一起，再加一個空格

goout：確認結尾是否是只有一個字，是加 X，否跳至 goout2 離開

```
102     Modified PROC
103         mov     esi,edi
104         mov     edi,OFFSET Modif
105         push    edi
106     L3:
107         mov     al,[esi]
108         cmp     al,0
109         je      goout
110         mov     bl,[esi+1]
111         cmp     al,bl
112         je      addX
113         jmp     two
114
115     addX:
116         mov     [edi],al
117         mov     BYTE PTR [edi+1],'X'
118         mov     BYTE PTR [edi+2],' '
119         add     esi,1
120         add     edi,3
121         jmp     L3
```

```
122     two:
123         mov     [edi],al
124         mov     [edi+1],bl
125         mov     BYTE PTR [edi+2],' '
126         add     esi,2
127         add     edi,3
128         jmp     L3
129     goout:
130         sub     edi,3
131         cmp     bl,0
132         jne     goout2
133         mov     BYTE PTR [edi+1],'X'
134     goout2:
135         pop     edi
136     ret
137     Modified ENDP
138
```

ToCipher

esi 要加密的字串、eax,edi key 的起始位置

149~151：確認是否是結尾

153~162 找是哪個 row 跟 column，scasb 會把找到的值的位子存在 edi

運用(eax-edi)(-1)/5 就可知道 row 跟 column

因為一次要比兩個字所以 164~184 再做一次

```
140     ToCipher PROC
141
142         mov     esi,OFFSET Modif
143         mov     eax,edi
144
145         push    edi
146         push    eax
147     L4:
148         mov     ecx,LENGTHOF Modif
149         mov     al,[esi]
150         cmp     al,0
151         je      goout3
152         cld
153         repne   scasb
154         pop     eax
155         push    eax
156         dec     edi
157         sub     eax,edi
158         mul     val4
159         div     val
160
161         mov     BYTE PTR val2,al
162         mov     BYTE PTR val3,ah
163
164         pop     eax
165         pop     edi
166         push    edi
167         push    eax
168
```

183~186 比較 row 是否一樣，是跳到 samer 處理
187~190 比較 column 是否一樣，是跳到 samec 處理
都不是就跳到 dif 處理

```
169         mov     al,[esi+1]
170         cld
171         repne   scasb
172
173         pop     eax
174         push    esi
175         dec     edi
176         push    eax
177         sub     eax,edi
178         mul     val4
179         div     val
180         mov     BYTE PTR val5,al
181         mov     BYTE PTR val6,ah
182
183         mov     ax,val2
184         mov     bx,val5
185         cmp     ax,bx
186         je      samer
187         mov     ax,val3
188         mov     bx,val6
189         cmp     ax,bx
190         je      samec
191         jmp     dif
```

samer

將 PlayfaieKey 的起始位子放在 ebx，eax 放 array 的 rowsize，將第一個字的 row 乘上 rowsize 加上 column 再加 1 就會是他應該的值，若加 1 後值超過 5 就要改成 0，兩個字都是如此

```
192     samer:
193         mov     ebx,OFFSET PlayfairKey
194         mov     eax,Rowsize
195         Imul    ax,WORD PTR val2
196         add     ebx,eax
197         movzx   esi,val3
198         add     esi,1
199         cmp     esi,5
200         jne     neq
201         sub     esi,5
202     neq:
203         mov     al,[ebx+esi]
204         mov     edi,addre
205         mov     [edi],al
206         movzx   esi,val6
207         add     esi,1
208         cmp     esi,5
209         jne     neq2
210         sub     esi,5
211     neq2:
212         mov     al,[ebx+esi]
213         mov     [edi+1],al
214         jmp     L5
```

samec

將 PlayfaieKey 的起始位子放在 ebx，eax 放 array 的 rowsize，將第一個字的 row 加 1，若加 1 後值超過 5 就要改成 0，乘上 rowsize 加上 column 再就會是他應該的值，兩個字都是如此

```
216   samec:
217       mov     ebx,OFFSET PlayfairKey
218       mov     eax,Rowsize
219       movzx   esi,val2
220       add     esi,1
221       cmp     esi,5
222       jne     neq3
223       sub     esi,5
224   neq3:
225       mov     WORD PTR val2,si
226       Imul    ax,WORD PTR val2
227       add     ebx,eax
228       movzx   esi,val3
229       mov     al,[ebx+esi]
230       mov     edi,addre
231       mov     [edi],al
232       mov     ebx,OFFSET PlayfairKey
233       mov     eax,Rowsize
234       movzx   esi,val5
235       add     esi,1
236       cmp     esi,5
237       jne     neq4
238       sub     esi,5
239   neq4:
240       mov     WORD PTR val5,si
241       Imul    ax,WORD PTR val5
242       add     ebx,eax
243       movzx   esi,val3
244       mov     al,[ebx+esi]
245       mov     [edi+1],al
246
247       jmp     L5
```

dif

將 PlayfaieKey 的起始位子放在 ebx，eax 放 array 的 rowsize，將第一個字的 row 乘上 rowsize 加上另一個字的 column 就會是他應該的值，兩個字都是如此

```
248     dif:
249         mov     ebx,OFFSET PlayfairKey
250         mov     eax,Rowsize
251         Imul    ax,WORD PTR val2
252         add     ebx,eax
253         movzx   esi,val6
254         mov     al,[ebx+esi]
255         mov     edi,addre
256         mov     [edi],al
257         mov     ebx,OFFSET PlayfairKey
258         mov     eax,Rowsize
259         Imul    ax,WORD PTR val5
260         add     ebx,eax
261         movzx   esi,val3
262         mov     al,[ebx+esi]
263         mov     [edi+1],al
264
```

兩個字都改完後要再加一個空格

```
265     L5:
266         mov     BYTE PTR [edi+2],' '
267         add     edi,3
268         mov     addre,edi
269         pop     edi
270
271         pop     esi
272         pop     eax
273         add     esi,3
274         push    eax
275         push    eax
276
277         jmp     L4
278     goout3:
279         pop     eax
280         pop     eax
281         ;pop        eax
282     ret
283     ToCipher ENDP
284
285     END main
```