

身分證字號判別

說明：

身分證字號的每個字元代表的意義如下,例如：A123456789

(1)第一個字元代表地區，轉換方式為：A 轉換成 1,0 兩個字元，B 轉換成 1,1.....

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
10	11	12	13	14	15	16	17	34	18	19	20	21	22	35	23	24	25	26	27	28	29	32	30	31	33

(2)第二個字元代表性別，1 代表男性，2 代表女性

(3)第三個字元到第九個字元為流水號碼。

(4)第十個字元為檢查號碼。

檢查碼產生的規則為：

1.

每個相對應數字乘上權數										檢查號碼
A		1	2	3	4	5	6	7	8	9
1	0	x8	x7	x6	x5	x4	x3	x2	x1	
x1	X9									

2.將乘上權數後之積相加

$$\begin{array}{rclcl}
 1 & \times & 1 & = & 1 \\
 0 & \times & 9 & = & 0 \\
 1 & \times & 8 & = & 8 \\
 2 & \times & 7 & = & 14 \\
 3 & \times & 6 & = & 18 \\
 4 & \times & 5 & = & 20 \\
 5 & \times & 4 & = & 20 \\
 6 & \times & 3 & = & 18 \\
 7 & \times & 2 & = & 14 \\
 +) & 8 & \times & 1 & = & 8 \\
 \hline
 & & & & & 121
 \end{array}$$

3.相加後之值除以模數 10 取其餘數

$$121 / 10 = 12 \cdots \text{餘數 } 1$$

4.由模數減去餘數得檢查號碼，若餘數為 0 時，則設定其檢查碼為 0

$$10 - 1 = 9$$

(5) 檢查步驟：

1. 檢查長度，須為十個字元。
2. 檢查格式，第一碼須為字母，後九碼須為數字。
3. 檢查性別，第一個數字須為 1(男性)或 2(女性)。
4. 計算檢查碼是否正確。

以上檢查都正確即為正確之身分證號碼，但不能確定是否已分配出去或尚在使用，僅能確定其「正確」。

Python:

```
ID= input("請輸入身份證字號:").upper()

if(len(ID) != 10):
    print("錯誤，身份證字號須為 10 碼")
elif(not ID[0].isalpha()):
    print("錯誤，第一碼須為英文字母")
elif(not ID[1:].isdigit()):
    print("錯誤，後九碼須為數字")
elif(ID[1]<'1' or ID[1]>'2'):
    print("錯誤，第一個數字須為 1 或 2")
else:
    X= {'A':10, 'B':11, 'C':12, 'D':13, 'E':14,
        'F':15, 'G':16, 'H':17, 'I':34, 'J':18,
        'K':19, 'L':20, 'M':21, 'N':22, 'O':35,
        'P':23, 'Q':24, 'R':25, 'S':26, 'T':27,
        'U':28, 'V':29, 'W':32, 'X':30, 'Y':31,
        'Z':33 }
    num= X[ID[0]]//10 + (X[ID[0]]%10)*9
    for i in range(2,10):
        num += int(ID[-i])*(i-1)
    ans= 10 - num % 10

    if(ans==int(ID[-1])):
        print(ID + " 是正確的身分證字號")
    else:
        print(ID + " 不是正確的身分證字號")
```

C:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>

int check1(char* ID)
{
    for(int i=2; i<10; i++)
        if(!isdigit(ID[i]))
            return 0;
    return 1;
}

int check2(char* ID)
{
    int N, num, A[26];
    //身分證字母對照表
    A[0]=10; A[1]=11; A[2]=12; A[3]=13; A[4]=14;
    A[5]=15; A[6]=16; A[7]=17; A[8]=34; A[9]=18;
    A[10]=19; A[11]=20; A[12]=21; A[13]=22; A[14]=35;
    A[15]=23; A[16]=24; A[17]=25; A[18]=26; A[19]=27;
    A[20]=28; A[21]=29; A[22]=32; A[23]=30; A[24]=31;
    A[25]=33;

    N= ID[0] - 65; //將 ASCII 字母轉成相對應的陣列索引值
    num= A[N]/10 + (A[N]%10)*9;
    for(int j=1,i=8; i>=1; i--,j++)
        num+= (ID[j] & 0x0F)*i; //ID[j] & 0x0F 是將文字'0-9'轉成數值 0-9
    num= 10 - num%10;

    if(num == (ID[9] & 0x0F))
        return 1;
    else
        return 0;
}
```

```
int main(void)
{
    char ID[11];
    printf("請輸入身份證字號：");
    scanf("%s",ID);
    ID[0]= toupper(ID[0]);

    if(strlen(ID)!=10)
        printf("長度須為 10 個字");
    else if(!isalpha(ID[0]))
        printf("第一個字須為英文字母");
    else if(ID[1]<'1' || ID[1]>'2')
        printf("第一個數字須為 1 或 2");
    else if(!check1(ID))
        printf("除第一個字為字母，後九碼須為數字");
    else
        if(check2(ID))
            printf("%s 是正確的身分證字號",ID);
        else
            printf("%s 不是正確的身分證字號",ID);
    return 0;
}
```

VB:

Module Module1

Sub Main()

Dim ID As String

Console.Write("請輸入身份證字號：")

ID = UCase(Console.ReadLine())

If Len(ID) <> 10 Then

Console.WriteLine("長度須為 10 個字")

ElseIf Mid(ID, 1, 1) < "A" Or Mid(ID, 1, 1) > "Z" Then

Console.WriteLine("第一個字須為英文字母")

ElseIf Mid(ID, 2, 1) <> "1" And Mid(ID, 2, 1) <> "2" Then

Console.WriteLine("第一個數字須為 1 或 2")

ElseIf check1(ID) Then

Console.WriteLine("除第一個字為字母，後九碼須為數字")

ElseIf check2(ID) Then

Console.WriteLine(ID & " 是正確的身分證字號")

Else

Console.WriteLine(ID & " 不是正確的身分證字號")

End If

Console.ReadLine()

End Sub

Function check1(ID)

For i = 2 To 10

If Mid(ID, i, 1) < "0" Or Mid(ID, i, 1) > "9" Then Return 1

Next

Return 0

End Function

Function check2(ID)

Dim A(26), N, num, s As Integer

A(1) = 10 : A(2) = 11 : A(3) = 12 : A(4) = 13 : A(5) = 14

A(6) = 15 : A(7) = 16 : A(8) = 17 : A(9) = 34 : A(10) = 18

A(11) = 19 : A(12) = 20 : A(13) = 21 : A(14) = 22 : A(15) = 35

A(16) = 23 : A(17) = 24 : A(18) = 25 : A(19) = 26 : A(20) = 27

A(21) = 28 : A(22) = 29 : A(23) = 32 : A(24) = 30 : A(25) = 31

A(26) = 33

N = Asc(Mid(ID, 1, 1)) - 64

num = Int(A(N) / 10) + (A(N) Mod 10) * 9

s = 2

For i = 8 To 1 Step -1

num += Val(Mid(ID, s, 1)) * i

s += 1

Next

num = 10 - (num Mod 10)

If num <> Val(Mid(ID, 10, 1)) Then

Return 0

Else

Return 1

End If

End Function

End Module