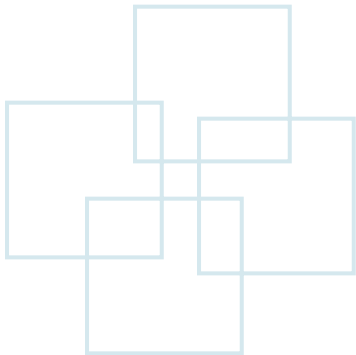


Chapter 16

Bit Operations





Number System

- Decimal number system:

- 0~9

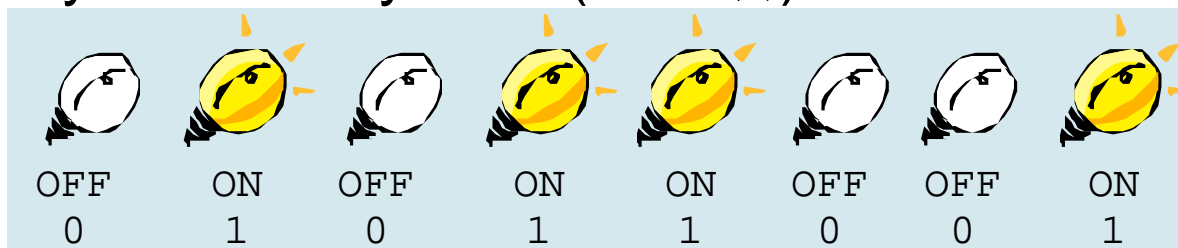
- E.g. $6935_{10} = 6 \times 10^3 + 9 \times 10^2 + 3 \times 10^1 + 5 \times 10^0$

- Week (7進制) vs. decimal number system:

十進位、七進位與星期系統的對照

十進位	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
七進位	0	1	2	3	4	5	6	10	11	12	13	14	15	16	20	21	22
星期系統	日	一	二	三	四	五	六	日	一	二	三	四	五	六	日	一	二

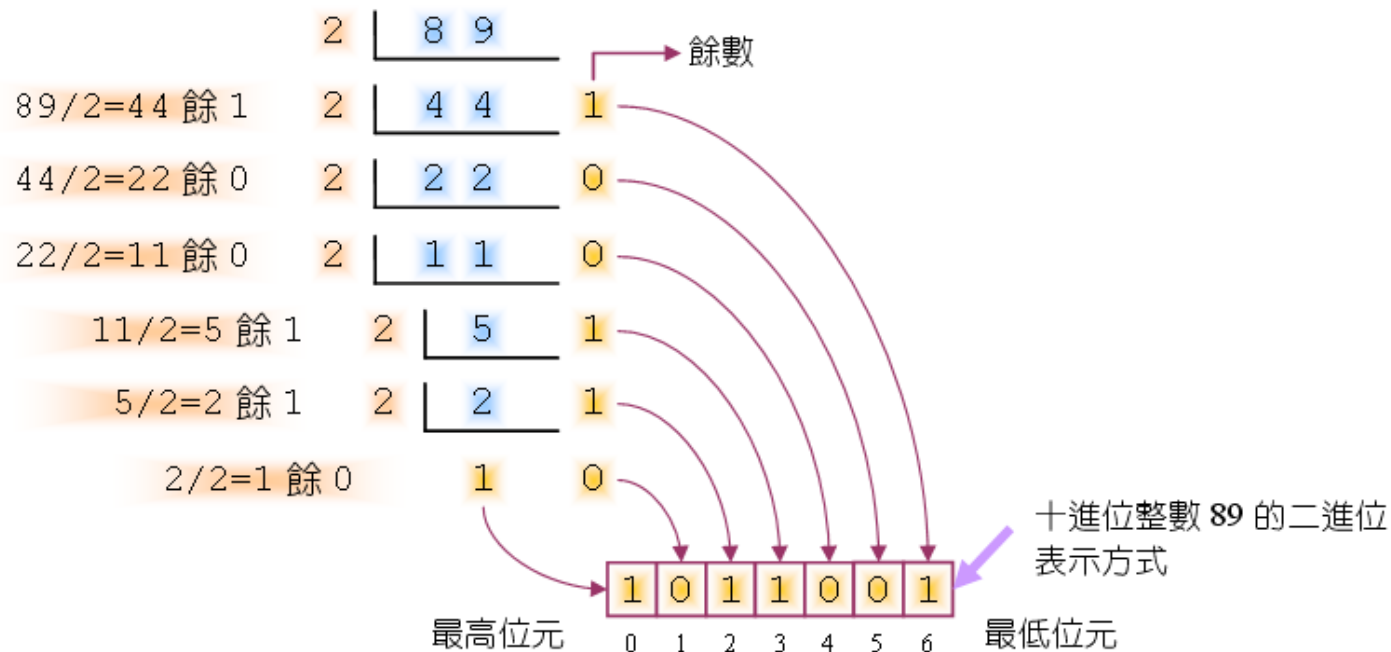
- Binary number system (二進制):





Binary (Number) System (二進制)

• Decimal to binary



$$1011001_2 = 1 \times 2^6 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^0 = 89_{10}$$



Example: Number System Conversion

- Convert a decimal number to a binary number

```
01  /* 將十進位整數以二進位來表示 */
02  #include <stdio.h>
03  #include <stdlib.h>
04  #define SIZE 8          /* 定義 SIZE 為 8, 代表以 8 個數字顯示二進位 */
05  void show_binary(int); /* 宣告 show_binary() 函數的原型 */
06  int main(void)
07  {
08      printf("89 的二進位為: ");
09      show_binary(89);    /* 顯示 89 的二進位 */
10
11      system("pause");
12      return 0;
13  }
```



Example: Number System Conversion (Cont.)

```

14 void show_binary(int num) /* 函數 show_binary() 的定義 */
15 {
16     int i,b[SIZE]={0}; /* 宣告陣列 b，並設定元素的初值都是 0 */
17     for(i=1;i<=SIZE;i++)
18     {
19         b[SIZE-i]=num%2; /* 將 num%2 的餘數設定給 b[SIZE-i] 存放 */
20         num=num/2; /* 將 num/2 的值設回給 num */
21     }
22     for(i=0;i<SIZE;i++)
23         printf("%d",b[i]);
24     printf("\n");
25 }

```

/* OUTPUT--

89 的二進位為: 01011001

-----*/



Octal Number (八進制)

• Octal number: 0~7

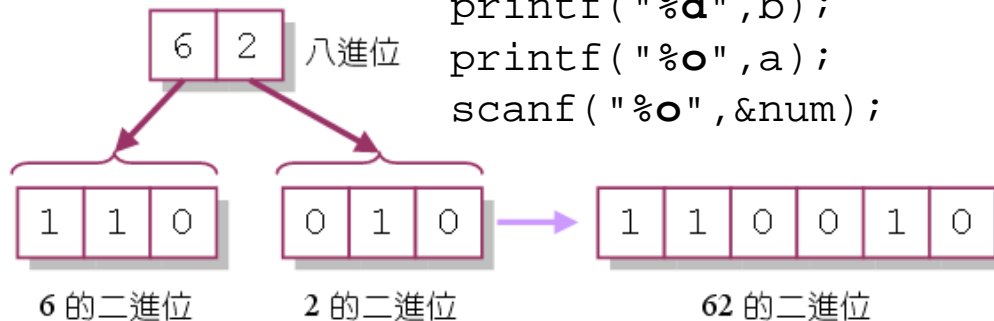
八進位和二進位的相對應值

八進位	0	1	2	3	4	5	6	7
二進位	000	001	010	011	100	101	110	111

Octal number

```
int num, a=38, b=062;
printf("%d", b);
printf("%o", a);
scanf("%o", &num);
```

```
/* 設定 a=38, b為八進位的62 */
/* 將變數b的值以十進位印出 */
/* 將變數a的值以八進位印出 */
/* 從鍵盤輸入一個八進位的整數 */
```





Hexadecimal Number (16進制)

十六進位、十進位、八進位與二進位的對照表

十進位	二進位	八進位	十六進位
0	0000	00	0
1	0001	01	1
2	0010	02	2
3	0011	03	3
4	0100	04	4
5	0101	05	5
6	0110	06	6
7	0111	07	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F



Hexadecimal Number (16進制) (Cont.)

$$5D_{16} = 5 \times 16^1 + 13 \times 16^0 = 93_{10}$$

```
int num, a=25, b=0x6A; /* 設定 a=25, b 的值為十六進位的 6A */
printf("%d", b); /* 將變數 b 的值以十進位印出 */
printf("%x", a); /* 將變數 a 的值以十六進位印出 */
scanf("%x", &num); /* 從鍵盤輸入一個十六進位的整數 */
```




Bit Operator (位元運算子)

位元邏輯運算子

位元邏輯運算子	意義
~	位元 NOT 運算子
&	位元 AND 運算子
	位元 OR 運算子
^	位元 XOR 運算子

NOT 運算子「~」之真值表

a	~a
0	1
1	0



Operator Example

AND 運算子「&」之真值表

a	b	a&b
0	1	0
0	0	0
1	1	1
1	0	0

OR 運算子「|」之真值表

a	b	a b
0	1	1
0	0	0
1	1	1
1	0	1

XOR 運算子「^」之真值表

a	b	a ^ b
0	1	1
0	0	0
1	1	0
1	0	1

$$\begin{array}{r}
 \begin{array}{cccccccc} 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{array} & \text{十進位的值為 } 105 \\
 \begin{array}{cccccccc} 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{array} & \text{十進位的值為 } 26 \\
 \hline
 \begin{array}{cccccccc} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{array} & \text{十進位的值為 } 8
 \end{array}$$

a&b

$$\begin{array}{r}
 \begin{array}{cccccccc} 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{array} & \text{十進位的值為 } 105 \\
 \begin{array}{cccccccc} 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{array} & \text{十進位的值為 } 26 \\
 \hline
 \begin{array}{cccccccc} 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \end{array} & \text{十進位的值為 } 123
 \end{array}$$

a|b

$$\begin{array}{r}
 \begin{array}{cccccccc} 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{array} & \text{十進位的值為 } 105 \\
 \begin{array}{cccccccc} 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{array} & \text{十進位的值為 } 26 \\
 \hline
 \begin{array}{cccccccc} 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 \end{array} & \text{十進位的值為 } 115
 \end{array}$$

a^b



AND Operator (Example)

```
01  /* prog14_2, 位元AND運算子「&」的範例說明 */
02  #include <stdio.h>
03  int main(void)
04  {
05      int a=105,b=26;
06      printf("%d&%d=%d\n", a,b, a&b);    /* 計算 a&b 的值 */
07
08      system("pause");
09      return 0;
10  }

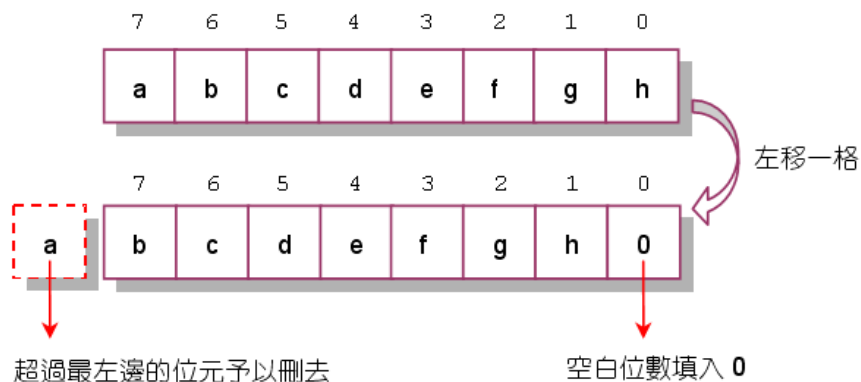
/* OUTPUT---
105&26=8
-----*/
```



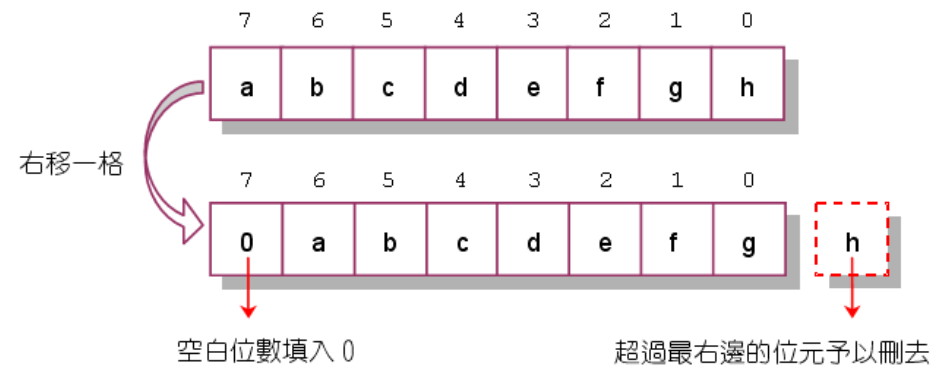
Shift Operator

位元位移運算子

位元位移運算子	意義
$\text{num} \ll n$	左移，將 num 的位元向左移 n 個位元
$\text{num} \gg n$	右移，將 num 的位元向右移 n 個位元



左移運算子



右移運算子



Shift Operator Example

```

01  /* 左移運算子「<<」的使用範例 */
02  #include <stdio.h>
03  #include <stdlib.h>
04  #define SIZE 8          /* 定義 SIZE 為 8，代表以 8 個數字來顯示二進位 */
05  void show_binary(int); /* 宣告 show_binary() 函數的原型 */
06  int main(void)
07  {
08      int a;
09      a=(89<<1);        /* 將整數 89 往左移一個位元，然後設定給變數 a 存放 */
10      printf("89 二進位的值為: ");
11      show_binary(89);          /* 顯示數字 89 的二進位 */
12      printf("左移一個位元後: ");
13      show_binary(a);          /* 顯示 89 左移一個位元後的二進位 */
14      printf("左移一個位元後的十進位值為: %d\n", a);
15      system("pause");
16      return 0;
17  }

```

```

89 二進位的值為: 01011001
左移一個位元後: 10110010
左移一個位元後的十進位值為: 178

```



Bit Fields (位元欄位)

- Each field of a structure only occupies the specified number of bits, instead of bytes.

Declaration format

```
struct StructName
{
    DataType FieldName1: BitNum;
    DataType FieldName1: BitNum;
    ...
    DataType FieldName1: BitNum;
};
```

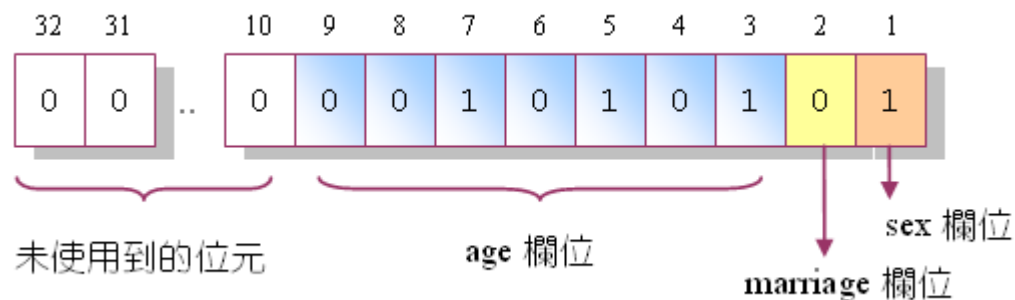


Example of Bit Fields

```

struct status                                     /* 定義位元欄位結構 */
{
    unsigned sex:1;                               /* sex欄位，佔1個位元 */
    unsigned marriage:1;                         /* marriage欄位，佔1個位元 */
    unsigned age:7;                              /* age欄位，佔7個位元 */
};
struct status tom;                               /* 宣告 struct status 的結構變數 tom */

```





Example of Bit Fields (Cont.)

```

01  /* 位元欄位結構的使用 */
02  #include <stdio.h>
03  int main(void)
04  {
05      struct status          /* 定義位元欄位結構 */
06      {
07          unsigned sex:1;
08          unsigned marriage:1;
09          unsigned age:7;
10      };
11      struct status tom={1,0,21}; /* 宣告並設定結構變數的初值 */
12
13      if(tom.sex==0)          /* 判別 sex 欄位的位元是否為 0 */
14          printf("性別:女,");
15      else
16          printf("性別:男,");
17
18      if(tom.marriage==0)    /* 判別 marriage 欄位的位元是否為 0 */
19          printf("未婚,");
20      else
21          printf("已婚,");
22
23      printf("%d歲\n",tom.age); /* 印出 age 欄位的值 */
24
25      printf("sizeof(tom)=%d\n",sizeof(tom)); /* 印出變數 tom 的大小 */
26
27      system("pause");
28      return 0;
29  }

```

/* OUTPUT--

性別:男,未婚,21歲

sizeof(tom)=4

***/**



Lab 16

- 一年有12個月，所以月份是一個十二進位的系統，16個月就相當於一年又4個月，如果十二進位系統是以數字0~9與英文字母A與B來代表。請設計一個程式讓使用者輸入一個十進位整數，然後把些整數轉換為12進位之後輸出在螢上。