

流程圖與 選擇性敘述

4

CHAPTER

4-1 | 流程圖之表示符號說明

我們現在所使用的流程圖（Flow Chart）是在西元 1940 年由 John von Neumann 所訂定的。在進行較複雜的程式設計時，往往可藉由繪製流程圖，來輔助程式的設計。藉由流程圖的繪製，可以讓程式設計者更清楚地掌握程式的運作流程，並且對於程式的邏輯思考架構，有更深刻的瞭解。

由於電腦只能依照「程式」指示，逐步完成指定的工作，因此在設計程式時，必須先將問題分解成許多小步驟，然後再依一定的次序逐步執行，而這個描述問題解決程序的方法便稱做**演算法**（algorithm）。在此引用 Horowitz、Sahni 和 Dinesh 在《Fundamental of Data Structures in C++》一書對「演算法」的定義：**為解決某一問題或完成特定工作，一系列有次序且明確的指令集合**，所有演算法都會包含以下特性：

- 輸入（input）：演算法在運算前通常會有一些事先給定的輸入資料，這些資料大多是由使用者事先給予，或是在演算法的執行步驟中指定。
- 輸出（output）：演算法的目的就是產生結果，至少要有一項的輸出結果。

- 明確性（**definiteness**）：每個執行步驟都必須明確而清楚，不可存在模稜兩可的情況。
- 有限性（**finiteness**）：在任何情況下演算法一定要在有限的步驟內完成，不能無限期執行。
- 有效性（**effectiveness**）：演算法所描述的執行過程可以用人工的方式（例如用紙、筆），在一定時間內推算出相同的結果。

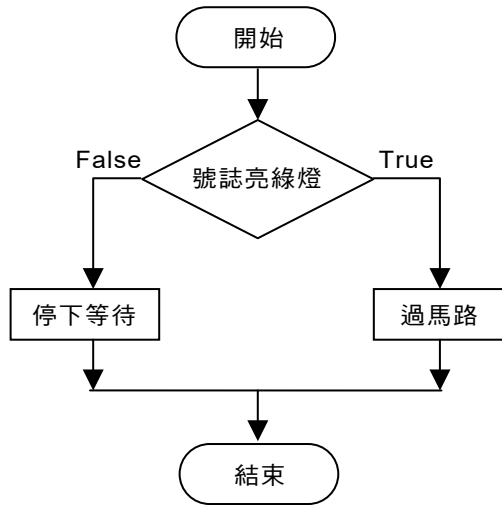
比起文字敘述，使用流程圖描述演算法有以下的好處：

- 對於複雜的演算法結構，流程圖比文字描述更能瞭解整個流程，方便進行除錯。
- 修改方便，可隨時加入或刪減流程圖的圖形符號。
- 可增加演算法的可讀性。

移交程式給他人進行維護，附上流程圖可幫助對方瞭解整個程式的處理流程。此處針對繪製流程圖所常使用的流程圖符號，進行列表說明：

流程圖符號	名稱	說明
	程序	代表一個程式區塊，具備執行特定工作的能力。
	決策	依條件式內容來進行條件判斷，以決定程序的下一條路徑。
	流程	用來表示程式的執行方向，具有方向性。
	開始/結束	用來表示演算法的開始或是結束。
	輸出/輸入	用來表示輸出或輸入資料。

我們用右邊的例子來說明如何使用流程圖來幫助我們思考與解決問題。例如當走到十字路口時，我們需要依據號誌的燈號，做一個判斷動作，以決定是否要過馬路或停下等待。參考的流程圖如右，假如在十字路口，遇到號誌亮綠燈才可以過馬路，否則就停下等待。當號誌亮綠燈時，條件判斷式為真（True），隨後進入過馬路動作，否則，條件判斷式為偽（False），必須進入停下等待動作。



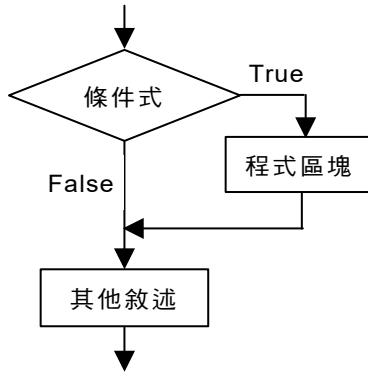
基本上流程圖所表達的是一套我們制定好的規則，並且用圖形來表示，善用它可以讓一些複雜的規則更容易傳達，也可以幫助我們去思考問題的解決方式。使用流程圖的優點在於讓人較容易了解整個作業流程，易於程式的除錯，也有助於程式的修改與維護。本書的程式流程圖為了減少頁面篇幅空間，以下的參考流程圖將會省略開始及結束等流程圖符號。

4-2 | 判斷結構之 if 敘述

if 敘述的用途是用來判斷是否進入程式區塊，如果條件式成立，則會進入程式區塊，如果條件式不成立，則會進入 if 敘述之後的其他敘述。其使用的語法如下，如果程式區塊只有一行程式敘述，則可以省略左大括號與右大括號：

```
if(條件式) {  
    程式區塊;  
}
```

使用流程圖的表示法如下：



if 敘述並不複雜，可以直接解釋為：如果條件式為真，就進入程式區塊執行敘述。參考下面的程式碼：

```
if(a%2==0)
  a=a/2;
```

上面的兩行敘述會判斷變數 *a* 是否為偶數，此處是使用『取餘數』算術運算子，如果餘數等於 (`==`) 0，則進入 if 敘述將其值除以 2；也就是說如果 *a* 對 2 取餘數其值等於 0，則表示 *a* 是 2 的倍數，也代表 *a* 為偶數。

在上面這個例子中並沒有加上大括號，因為如果只有一行程式敘述，大括號可以省略，如果超過一行，就必須加上大括號。再來看一個類似的例子：

```
if(a>=10 && a<=99) {
  b=a/10;
  c=a%10;
}
```

上面這四行敘述（假設變數 *a, b, c* 為整數型態）會判斷變數 *a* 是否為二位數，如果 *a* 為二位數（*a* 介於 10~99 之間），則 if 敘述會將變數 *a* 的十位數部分，指定給變數 *b*，再將變數 *a* 的個位數部分，指定給變數 *c*。

**程式範例：百貨公司週年慶打折程式**

學習重點：熟悉 if 敘述的使用／參考檔案：4-2-1.c

程式設計目標

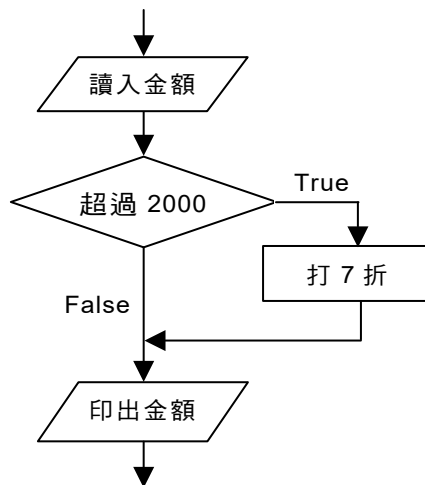
真神氣百貨公司週年慶，公司決定在週年慶期間對消費超過 2000 元的顧客打 7 折，來增加買氣，請幫該公司寫出一個收銀台程式，輸入顧客購買總金額後，計算顧客實際需付的錢。如圖為輸入『6999』的執行結果。


```
D:\C_Examples\...  -  □  ×
請輸入消費金額：6999
實需付 4899 元
請按任意鍵繼續 . . .
```

如圖為輸入『1800』的執行結果。


```
D:\C_Examples\...  -  □  ×
請輸入消費金額：1800
實需付 1800 元
請按任意鍵繼續 . . .
```

參考流程圖：



 參考程式碼

```
01: /*百貨公司週年慶打折程式 */
02: #include <stdio.h>
03: #include <stdlib.h>
04: int main(void)
05: {
06:     int money;
07:     printf("請輸入消費金額：");
08:     scanf("%d",&money); //輸入買了多少錢
09:     if(money > 2000) //判斷消費金額是否超過 2000
10:         money = money*0.7; //打 7 折
11:     printf("實需付 %d 元\n",money);
12:     system("pause");
13:     return 0;
14: }
```

 程式碼解說

第 6 行：宣告整數變數 `money`。

第 8 行：使用 `scanf()` 函式將使用者輸入的消費金額存入 `money` 整數變數。

第 9 行：用 `if` 敘述來判斷消費金額 `money` 是否超過 2000 元，如超過 2000 元，就進入第 10 行程式敘述來打七折。

第 10 行：計算打七折後的金額。

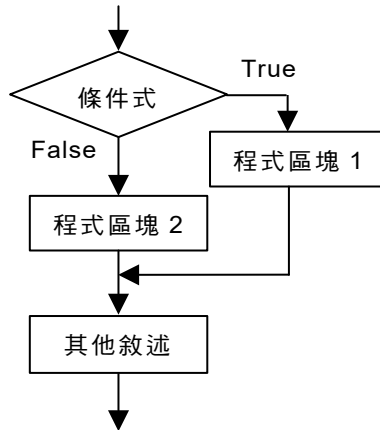
第 11 行：印出整數變數 `money`。

4-3 | 判斷結構之 `if...else...敘述`

`if...else...敘述`會根據條件式的真偽，決定要進入的程式區塊為何。如果 `if` 條件式成立，則會執行 `if` 部分的『程式區塊 1』；如果不成立則會執行 `else` 部分的『程式區塊 2』，兩個程式區塊會依據條件式成立與否，選擇其一來執行，使用的語法如下：

```
if(條件式) {
    程式區塊 1;
} else {
    程式區塊 2;
}
```

使用流程圖的表示法如下：




if...else...敘述有一個特點，就是在的程式區塊和 else 之後的程式區塊，其中有一段必會被執行，如果判斷式為真，就執行 if 之後的程式區塊，如果為偽，就執行 else 之後的程式區塊，參考下面的程式碼：【參考檔案：4-3-1.c】

```
if(a%2)
    printf("This is odd");
else
    printf("This is even");
```

上面的四行敘述會判斷 a 為偶數或奇數。如為奇數，對 2 取餘數會得到 1，也就是 if 敘述內的條件式為 True，因此會執行 if 下的『printf(“a is odd”);』敘述。如為偶數，對 2 取餘數會得到 0，也就是 if 敘述內的條件式為 False，因此會執行 else 下的『printf(“a is even”);』敘述。

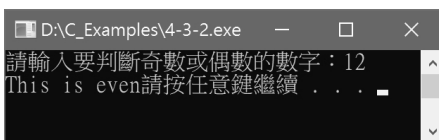
小提示 還記得三元運算子嗎？判斷印出 `a` 是奇數或偶數的 `if...else...` 程式，也可以改用三元運算子簡潔地撰寫出來，參考程式碼如下：

 **參考檔案：4-3-2.c**

```
01: /* 奇數偶數判斷程式 */
02: #include <stdio.h>
03: #include <stdlib.h>
04: int main(void)
05: {
06:     int a;
07:     printf("請輸入要判斷奇數或偶數的數字：");
08:     scanf("%d",&a);
09:     a%2 ?printf("This is odd") :printf("This is even");
10:     system("pause");
11:     return 0;
12: }
```

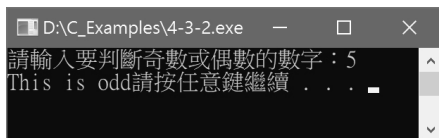
執行結果：

1. 輸入偶數 12。



```
D:\C_Examples\4-3-2.exe
請輸入要判斷奇數或偶數的數字：12
This is even請按任意鍵繼續 . . .
```

2. 輸入奇數 5。



```
D:\C_Examples\4-3-2.exe
請輸入要判斷奇數或偶數的數字：5
This is odd請按任意鍵繼續 . . .
```


**程式範例：雨天帶傘判斷程式**

學習重點：熟悉 if...else...敘述的使用／參考檔案：4-3-3.c

程式設計目標

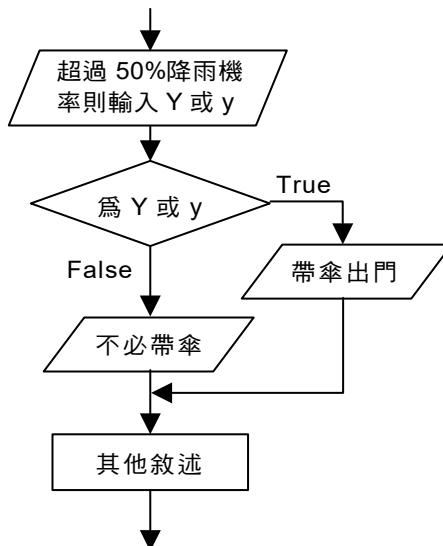
寫出一個程式，請問使用者今天降雨機率是否大於 50%，根據使用者的回答，輸出今天是否應該帶傘。下圖為使用者回答『Y』的執行結果。


```
D:\C_Examples\4-3-3...  -  □  ×  
今天降雨機率大於50%嗎?Y  
你最好要帶傘!  
請按任意鍵繼續 . . .
```

下圖為使用者回答『N』的執行結果。

```
D:\C_Examples\4-3-3...  -  □  ×  
今天降雨機率大於50%嗎?N  
可能不會下雨，不必帶傘了!  
請按任意鍵繼續 . . .
```

參考流程圖：




 參考程式碼

```

01: /* 雨天帶傘程式 */
02: #include <stdio.h>
03: #include <stdlib.h>
04: int main(void)
05: {
06:     char rain;
07:     printf("今天降雨機率大於 50%%嗎?");
08:     scanf("%c",&rain);    //字元型態變數，格式符號用%c
09:     if(rain=='y' || rain=='Y')    //判斷輸入是否為 y 或是 Y
10:         printf("你最好要帶傘!\n");
11:     else    //當使用者沒有按下 y 或 Y 時，進入 else 敘述中
12:         printf("可能不會下雨，不必帶傘了!\n");
13:     system("pause");    //使程式暫停在執行畫面讓我們看到結果
14:     return 0;
15: }

```

 程式碼解說

第 6 行：宣告字元變數 `rain`。

第 7 行：使用 `printf()` 函式印出『今天降雨機率大於 50%嗎?』，因為『%』符號為跳脫字元，因此，此處需要使用 2 個『%%』符號。

第 8 行：使用 `scanf()` 函式讀入使用者的回答，將結果存到 `rain` 變數。因為是字元型態變數，所以格式符號用『%c』。

第 9~12 行：用 `if` 判斷式來判斷使用者的輸入。如果是輸入『y』或『Y』，表示今天降雨機率大於 50%，程式會告訴使用者『你最好要帶傘!』；如果使用者輸入非『y』或『Y』，表示降雨機率沒有大於 50%，則會回應『可能不會下雨，不必帶傘了!』。

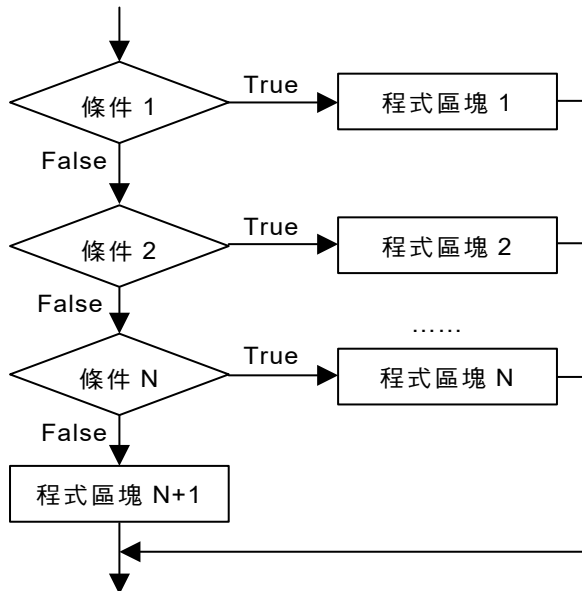
4-4 | 判斷結構之 `if...else if...else...` 敘述

`if...else if...else...` 敘述提供了多種狀況的選擇判斷，根據不同的條件判斷式，一一判斷要進入哪一個程式區塊。

使用的語法如下：

```
if(條件式 1) {  
    程式區塊 1;  
}  
else if(條件式 2) {  
    程式區塊 2;  
}  
...  
else if(條件式 N) {  
    程式區塊 N;  
}  
else {  
    程式區塊 N+1;  
}
```

使用流程圖的表示法如下：



參考下面的程式碼，此範例程式會依據條件式來判斷 a 是正數、零或是負數。當 a 值大於 0 時，會印出『**a is positive**』；當 a 值等於 0 時，會印出『**a is zero**』；當 a 值小於 0 時，則會印出『**a is negative**』。

```

if(a>0)
    printf("a is positive");
else if (a==0)
    printf("a is zero");
else
    printf("a is negative");

```

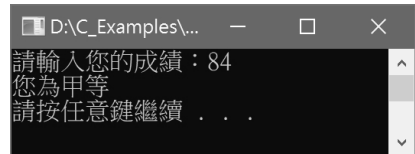


程式範例：成績區間判斷程式

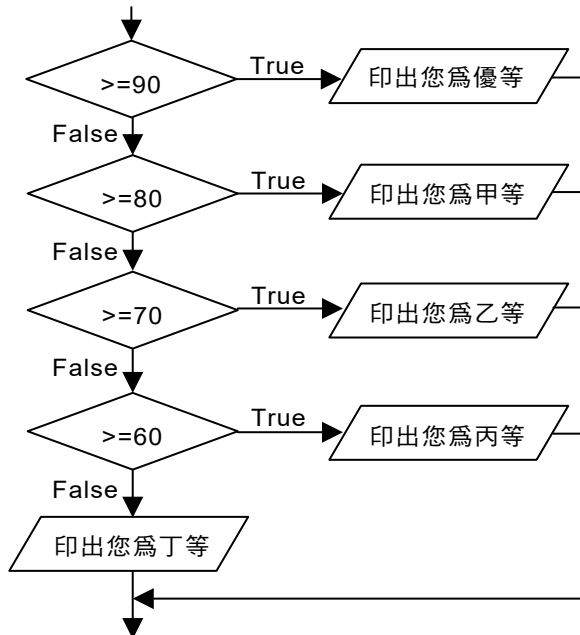
學習重點：熟悉 if...else if...else...敘述的使用／參考檔案：4-4-1.c


程式設計目標

依據使用者輸入的成績，判斷其成績等第為何？90 分以上為優等，80 分到未滿 90 分之間為甲等，70 分到未滿 80 分之間為乙等，60 分到未滿 70 分之間為丙等，未滿 60 分為丁等。右圖為輸入『84』的執行結果，程式會告知使用者『您為甲等』。




參考流程圖：



 參考程式碼


```
01: /* 成績區間判斷程式 */
02: #include <stdio.h>
03: #include <stdlib.h>
04: int main(void)
05: {
06:     int score;
07:     printf("請輸入您的成績：");
08:     scanf("%d",&score);
09:     if(score >= 90)           //是否為 90 分以上
10:         printf("您為優等\n");
11:     else if(score >= 80)     //小於 90 但 80 以上
12:         printf("您為甲等\n");
13:     else if(score >= 70)     //小於 80 但 70 以上
14:         printf("您為乙等\n");
15:     else if(score >= 60)     //小於 70 但 60 以上
16:         printf("您為丙等\n");
17:     else
18:         printf("您為丁等\n");
19:     system("pause");
20:     return 0;
21: }
```

 程式碼解說

第 8 行：使用 `scanf()` 函式讀入使用者的分數，將結果存到 `score` 變數。

第 9~18 行：為 `if...else if...else...` 敘述，如果成績高於等於 90 分，會執行第 10 行，印出『您為優等』。如果成績介於 80 分~89 分，會執行第 12 行，印出『您為甲等』。如果成績介於 70 分~79 分，會執行第 14 行，印出『您為乙等』。如果成績介於 60 分~69 分，會執行第 16 行，印出『您為丙等』。其他的成績則會印出『您為丁等』。

小提示 我們在輸入成績資料時，有時候會不小心輸入超出範圍的分數，比方說，百分制的成績等第計算，不小心輸入『105』分的時候，程式會輸出『優等』，並不會提醒『成績格式不合』。因此，我們可以修改原本的程式，讓程式可以正確處理區間內的資料，此處我們搭配邏輯運算子『&&』，來確保分數的區間是正確的，參考程式碼如下：

 參考檔案：4-4-2.c

```
01: /* 進階版成績區間判斷程式 */
02: #include <stdio.h>
03: #include <stdlib.h>
04: int main(void)
05: {
06:     int score;
07:     printf("請輸入您的成績：");
08:     scanf("%d",&score);
09:     if(score>=90 && score<=100)    //是否為 90 分以上
10:         printf("您為優等\n");
11:     else if(score>=80 && score<90) //小於 90 但 80 以上
12:         printf("您為甲等\n");
13:     else if(score>=70 && score<80) //小於 80 但 70 以上
14:         printf("您為乙等\n");
15:     else if(score>=60 && score<70) //小於 70 但 60 以上
16:         printf("您為丙等\n");
17:     else if(score>=0 && score<60) //小於 60 分
18:         printf("您為丁等\n");
19:     else
20:         printf("成績格式不合\n");
21:     system("pause");
22:     return 0;
23: }
```

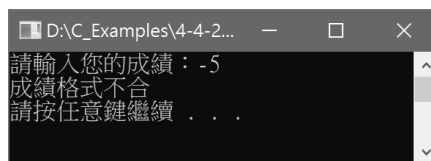
執行結果：

1. 輸入數字『105』。



```
D:\C_Examples\4-4-2...
請輸入您的成績：105
成績格式不合
請按任意鍵繼續 . . . .
```

2. 輸入數字『-5』。



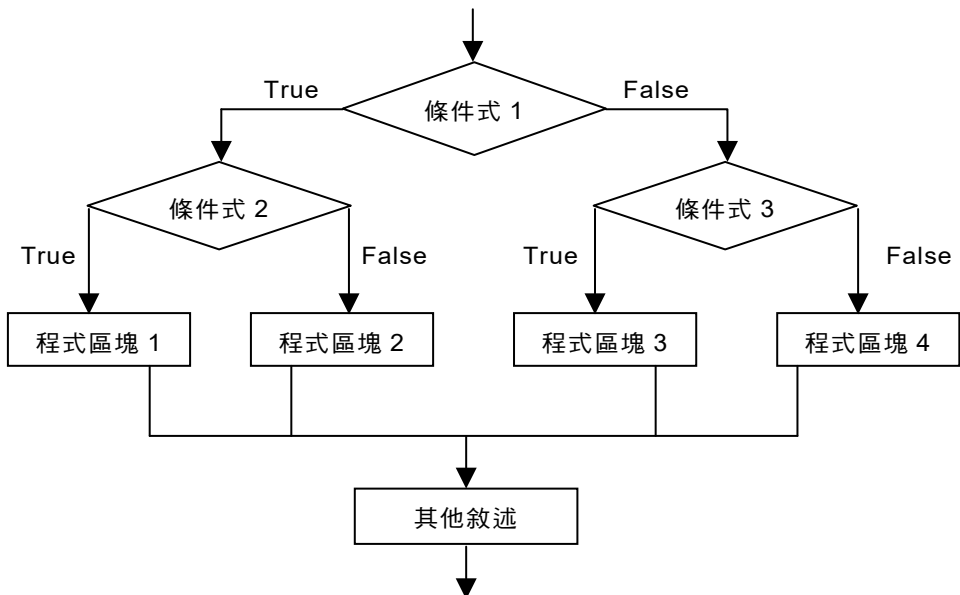
```
D:\C_Examples\4-4-2...
請輸入您的成績：-5
成績格式不合
請按任意鍵繼續 . . .
```

4-5 | 判斷結構之巢狀 if

當 if 敘述的程式區塊中，包含了其他的 if 敘述時，我們稱此種 if 敘述為巢狀 if 敘述，我們舉一個巢狀 if 的語法架構如下：

```
if(條件式 1) {  
    if(條件式 2) {  
        程式區塊 1;  
    }  
    else {  
        程式區塊 2;  
    }  
else {  
    if(條件式 3) {  
        程式區塊 3;  
    }  
    else {  
        程式區塊 4;  
    }  
}
```

使用流程圖的表示法如下：



當我們在撰寫巢狀 if 的時候，盡量將相對應的 if...else...敘述縮排對齊，以利未來程式之閱讀、維護或修改。



程式範例：閏年判斷程式

學習重點：熟悉巢狀 if 敘述的使用／參考檔案：4-5-1.c

程式設計目標

請寫出一程式，輸入西元年份，判斷該年是否為閏年（閏年判斷方式為：四年一閏，逢百年不閏，四百年又閏）。如果輸入『2100』，程式會輸出『2100 年為平年』。如果輸入『2000』，程式會輸出『2000 年為閏年』。

如圖為輸入『480』的執行結果。

```
D:\C_Example... - □ ×
請輸入西元年：480
480 年為閏年
請按任意鍵繼續 . . .
```

參考程式碼

```
01: /*閏年判斷程式 */
02: #include <stdio.h>
03: #include <stdlib.h>
04: int main(void)
05: {
06:     int year;
07:     print("請輸入西元年：");
08:     scanf("%d",&year);
09:     if(year%4==0) {           //可被 4 整除
10:         if(year%100==0 && year%400!=0) //被 4 和 100 整除且不被 400 整除
11:             printf("%d 年為平年\n",year);
12:         else                 //被 4 和 400 都整除，但不被 100 整除
13:             printf("%d 年為閏年\n",year);
14:     }
15:     else                     //無法被 4 整除
16:         printf("%d 年為平年\n",year);
17:     system("pause");
18:     return 0;
19: }
```


程式碼解說

第 8 行：使用 `scanf()` 函式讀入使用者輸入的西元年，將結果存到 `year` 變數。

第 9～14 行：此段 `if` 敘述處理 `year` 是 4 的倍數的情況，用的運算是取餘數運算，如果餘數為 0，表示該數為倍數。第 10、11 行處理被 100 整除，但不被 400 整除的狀況，印出『為平年』。第 12、13 行處理被 4 和 400 都整除，但不被 100 整除的狀況，印出『為閏年』。

第 15、16 行：此段 `else` 敘述處理 `year` 不是 4 的倍數的情況，只要不是 4 的倍數，就一定也不可能是 100 的倍數或是 400 的倍數，在這種情況下印出『為平年』。

4-6 | 判斷結構之 switch 敘述

在我們想要提供兩個選擇時，我們可以用 `if` 來做判斷，選擇要執行的程式區塊。當有多重選擇要進行判斷，要多選一時，我們可以用 `if...else if...else...` 敘述來做判斷。另外，C 語言為了將多選一的情況簡化，還提供了 `switch` 敘述，`switch` 會根據某一個字元或是整數變數，來判斷要進入哪一個程式區塊，其語法架構如下：

```
switch(變數或運算式) {
    case 值 1:
        程式區塊 1;
        break;
    case 值 2:
        程式區塊 2;
        break;
    ...
    case 值 N:
        程式區塊 N;
        break;
    default:
        程式區塊 N+1;
}
```

4-8 | 習題

選擇題：

() 1. 使用 switch 多重選擇結構，我們通常會在一個 case 結束時，加上何種指令？

- (a) more (b) break
(c) then (d) continue

() 2. 請問在流程圖中，「決策」所使用的符號為下列何者？

- (a)  (b) 
(c)  (d) 

() 3. 請問下列哪一種並非 C 的判斷流程語法？

- (a) switch...case (b) select...case
(c) if...else (d) if...else if...else

() 4. 如果 a 的值為 2，則執行下列程式後會輸出何值？

```
if (a==3)
a=3;
a=4;
```

- (a) 2 (b) 3
(c) 4 (d) 5

() 5. 如果 a 的值為 2，則執行下列程式後 a 為多少？

```
if (a==3)
a=3;
else
a=4;
```

- (a) 2 (b) 3
(c) 4 (d) 5

- (b) `if (x==10) y = 'a';`
`else if (x==20 || x==30) y = 'b';`
`else y = 'c';`
- (c) `if (x==10) y = 'a';`
`if (x >=20 && x <=30) y = 'b';`
`y = 'c';`
- (d) `if (x==10) y = 'a';`
`else if(x >=20 && x <=30) y = 'b';`
`else y = 'c';`

() 12. 下側是依據分類 s 評定等第的程式碼片段，正確的等第公式應為：

- 90~100 判為 A 等
 80~89 判為 B 等
 70~79 判為 C 等
 60~69 判為 D 等
 0~59 判為 F 等

這段程式碼在處理 0~100 的分數時，有幾個分數的等第是錯的？

【105 年 10 月 APCS 觀念題第 9 題】

```
if (s>=90) {
    printf ("A \n");
}
else if (s>=80) {
    printf ("B \n");
}
else if (s>=60) {
    printf ("D \n");
}
else if (s>=70) {
    printf ("C \n");
}
else {
    printf ("F\n");
}
```

- (a) 20
 (b) 11
 (c) 2
 (d) 10