

if

方淑玲 @ Sprout 2020

如果...

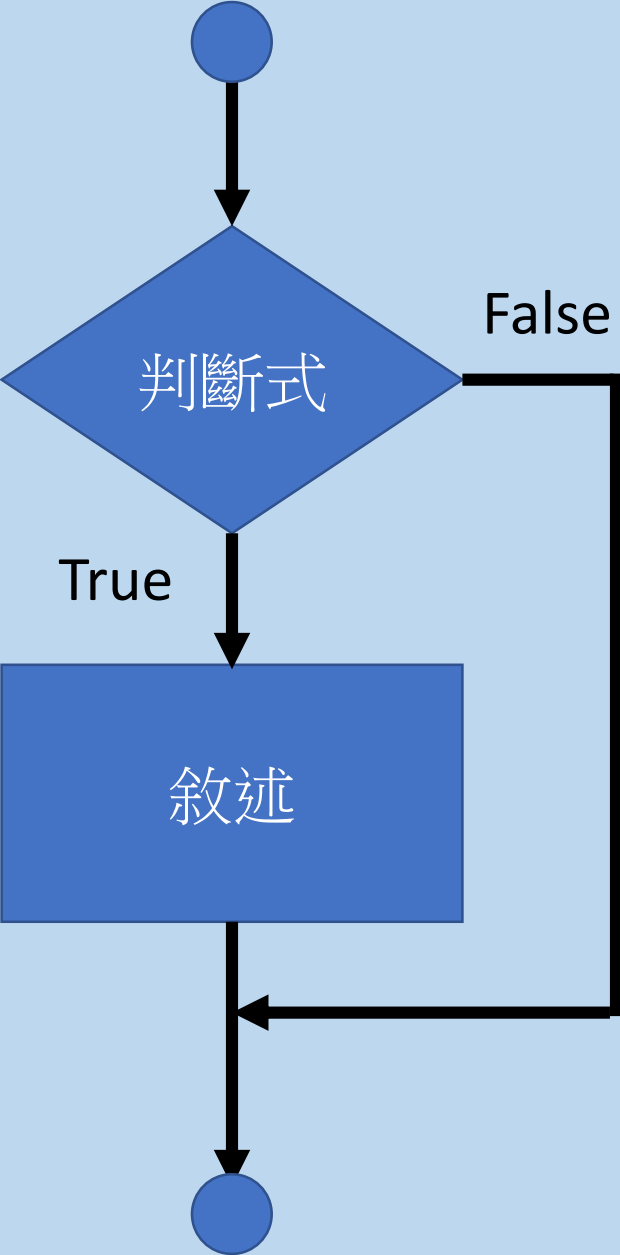
- 如果下雨又沒帶傘，就會淋濕
 - 如果下禮拜段考，週末就不能出去玩
 - 如果學校有兩個人確診，就要停課
 - 如果參加資芽，就會有完美的備審
-
- 如果〔條件〕，〔事件的描述〕
 - 條件為真 → 事件發生
 - 條件為偽 → 事件不發生

格式

```
if ( condition ){  
    statement;  
}
```

- 小括號內的條件成立 (**True**) → 執行大括號的敘述
- 小括號內的條件不成立 (**False**) → 不執行大括號的敘述

流程圖



生活中的 if

```
if( 下禮拜段考 ){  
    週末就不能出去玩;  
}
```

```
if( 有兩個人確診 ){  
    停課;  
}
```

程式中的 if

```
int a = 20200314;  
if( a%2 == 0 ){  
    std::cout << a << " 是偶數" << std::endl;  
}
```

```
int a = 20200314;  
int b = 5021314;  
if( a > b ){  
    std::cout << a << " > " << b << std::endl;  
}
```

一些名詞

- C++ 中用 1 代表 True，0 代表 False → 布林 (boolean) 值
 - 只要 非0 就是 True
- condition 要可以判斷 True 或 False → 布林 (boolean) 運算式
 - 不用分號
- 常見的布林運算式
 - 比較運算式
 - 邏輯運算式

比較運算子

運算子	意義	範例 (a = 5)	結果
<	小於	a < 0	false
<=	小於或等於	a <= 0	false
>	大於	a > 0	true
>=	大於或等於	a >= 0	true
==	等於	a == 0	false
!=	不等於	a != 0	true

注意 !!

邏輯運算子

運算子	意義	打法
&&	且	[shift + 7] [shift + 7]
	或	[shift + \] [shift + \]
!	否 (反)	[shift + 1]

真值表(Truth table)

A	B	A && B
1	1	1
1	0	0
0	1	0
0	0	0

A	B	A B
1	1	1
1	0	1
0	1	1
0	0	0

A	! A
1	0
0	1

運算子的優先順序

優先順序	運算子	說明
	!	否 (邏輯運算子)
	* 、 / 、 %	乘、除、取餘
	+ 、 -	加、減
	< 、 <= 、 > 、 >=	比較運算子
	== 、 !=	相等、不相等 (比較運算子)
	&&	且 (邏輯運算子)
		或 (邏輯運算子)

運算子的優先順序

優先順序	運算子	說明
	!	否 (邏輯運算子)
	* 、 / 、 %	乘、除、取餘
	+ 、 -	加、減
	< 、 <= 、 > 、 >=	比較運算子
	== 、 !=	相等、不相等 (比較運算子)
	&&	且 (邏輯運算子)
		或 (邏輯運算子)

記得把括號括好!!!

例子 – 求圓面積

- Input : 一個整數 R ，代表半徑
- Output : 一個整數 A ，代表圓面積
- Note : $\pi = 3$

參考解答 – 求圓面積

```
#include <iostream>
int main(){
    int R;
    std::cin >> R;
    if(R >= 0){
        std::cout << 3*R*R << std::endl;
    }
    return 0;
}
```

else

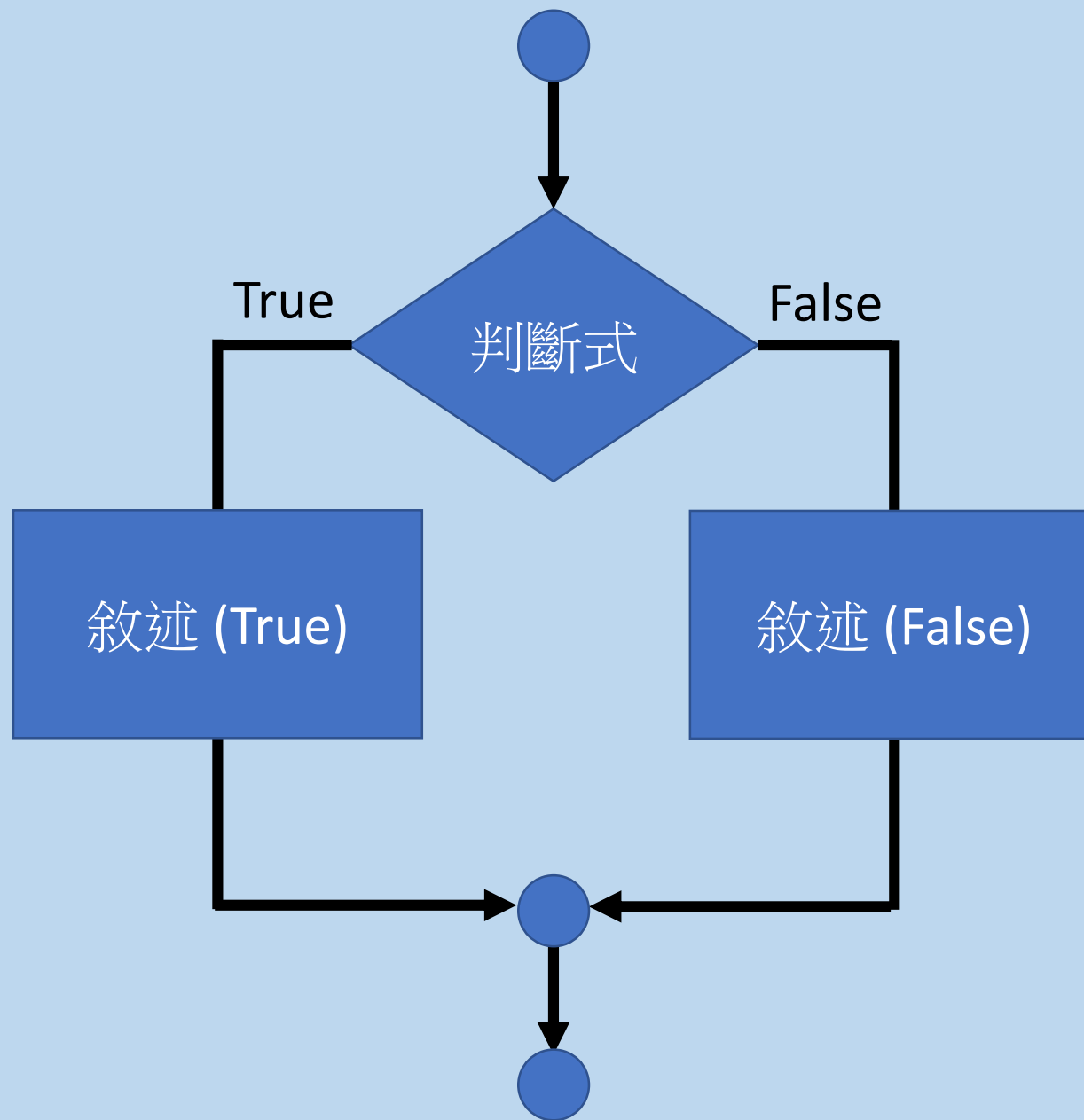
如果我們想要在 $R < 0$ 的時候輸出錯誤訊息 ...

格式

```
if ( condition ){  
    statement_for_the_true_case;  
}  
else{  
    statement_for_the_false_case;  
}
```

- 注意：else 後面不用小括號

流程圖



例子 – 求圓面積2

- Input : 一個整數 R，代表半徑
- Output : 一個整數A，代表圓面積。
如果輸入不合法，輸出 “invalid input”
- Note : $\pi = 3$

參考解答 – 求圓面積2

```
#include <iostream>
int main(){
    int R;
    std::cin >> R;
    if(R >= 0){
        std::cout << 3*R*R << std::endl;
    }
    else{
        std::cout <<"invalid input"<< std::endl;
    }
    return 0;
}
```

練習 - 成績判斷

- Input : 一個整數 x ($0 \leq x \leq 100$) , 代表成績
- Output : 如果 $x \geq 60$, 輸出 “pass”
如果 $x < 60$, 輸出 “failed”

參考解答

```
#include <iostream>
int main(){
    int x;
    std::cin >> x;
    if(x >= 60){
        std::cout << "pass" << std::endl;
    }
    else{
        std::cout << "failed" << std::endl;
    }
    return 0;
}
```

練習 - 成績判斷2

- Input : 一個整數 x ($0 \leq x \leq 100$) , 代表成績
- Output : 如果 $x \geq 80$, 輸出 "A"
如果 $80 > x \geq 70$, 輸出 "B"
如果 $70 > x \geq 60$, 輸出 "C"
如果 $x < 60$, 輸出 "failed"

一個錯誤的程式

```
#include <iostream>
int mian(){

    std::cin << x
    if(x >= 80;){
        std::cout << "A" << std::endl
    }
    if(80 > x >= 70;){
        std::cout << "B" << std::endl
    }
    if(70 > x >= 60;){
        std::cout << "C" << std::endl
    }
    else(60 > x;){
        std::cout << "failed" << std::endl
    }
    return 0
}
```

一個錯誤的程式

```
#include <iostream>
int mian(){

    std::cin << x
    if(x >= 80;){
        std::cout << "A" << std::endl
    }
    if(80 > x >= 70;){
        std::cout << "B" << std::e
    }
    if(70 > x >= 60;){
        std::cout << "C" << std::e
    }
    else(60 > x;){
        std::cout << "failed" << s
    }
    return 0
}
```

1. $80 > x \geq 70$
2. else後面小括號
3. if - else if – else
4. statement 分號
5. 宣告
6. cin >>
7. 小括號的分號
8. main

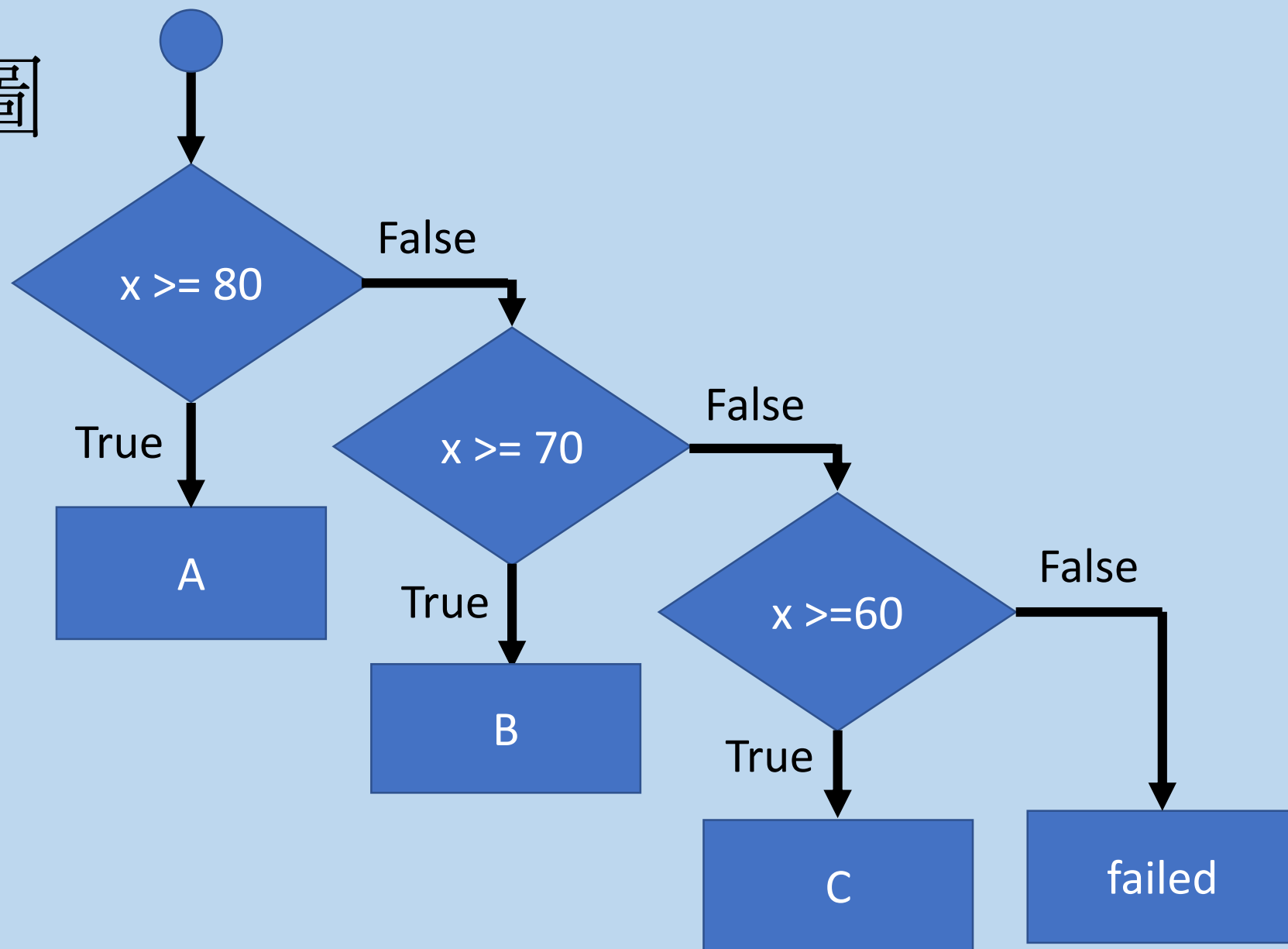
if - else if - else

```
if (condition_1){  
    statement_1;  
}  
else if(condition_2){  
    statement_2;  
}  
...  
else if(condition_N){  
    statement_N;  
}  
else{  
    statement_otherwise;  
}
```

參考解答

```
#include <iostream>
int main(){
    int x;
    std::cin >> x;
    if(x >= 80){
        std::cout << "A" << std::endl;
    }
    else if(x >= 70){
        std::cout << "B" << std::endl;
    }
    else if(x >= 60){
        std::cout << "C" << std::endl;
    }
    else{
        std::cout << "failed" << std::endl;
    }
    return 0;
}
```

流程圖



Nested (巢狀) if
if 裡面有 if

一個栗子(X)

```
#include <iostream>
int main(){
    int a = 12, b = 7;
    if(a > 10){
        if(b > 10){
            std::cout << "both a and b are bigger than 10" << std::endl;
        }
        else{
            std::cout << "a is bigger than 10, but b is not" << std::endl;
        }
    }
    else{
        if(b > 10){
            std::cout << "b is bigger than 10, but a is not" << std::endl;
        }
        else{
            std::cout << "neither a or b is bigger than 10" << std::endl;
        }
    }
    return 0;
}
```

各種提示的整理

- 如果 **statement** 只有一行可以不加大括號，但不建議
- 條件的判斷不加分號，敘述一定要加分號
- 分清楚 = 和 ==
- 善用邏輯運算子
- 只要不是0，電腦就會認為是True
- **else** 總是對應到最接近且尚未被對應的 **if**

else 總是對應到最接近且尚未被對應的 if

```
#include <iostream>
int main(){
    int i = 1, j = 2, k = 3;
    if(i > j)
        if(j > k)
            std::cout << "A" << std::endl;
        else std::cout << "B" << std::endl;
    return 0;
}
```

練習

208 – 季節判斷

季節判斷 - 2015

問題敘述

1-3月是春天，4-6月是夏天，7-9月是秋天，10-12月是冬天。

現在輸入一個整數(1-12)，代表月份，請你判斷此月屬於春夏秋冬哪個季節。

輸入說明

一個整數，代表月份

輸出說明

判斷輸入的月份屬於哪個季節(輸出為英文:spring/summer/fall/winter；記得換行)

輸入範例#1

5

輸出範例#1

summer

輸入範例#2

12

輸出範例#2

winter

補充

其他跟 if 相關的東西

switch

switch

```
switch(variable){  
    case value_0: statement_0;  
    |           |           |  
    |           |           | break;  
    case value_1: statement_1;  
    |           |           |  
    |           |           | break;  
    ...  
    case value_N: statement_N;  
    |           |           |  
    |           |           | break;  
    default: statement_default;  
}
```

- switch 根據變數或運算式的值，執行相對應的敘述內容

例題 – 12生肖

- Input：一個正整數 x ，代表年份
- Output：該年對應的生肖
- Note：1900是鼠年

十二生肖：rat, ox, tiger, rabbit, dragon, snake, horse, sheep,
monkey, rooster, dog, pig

- 想想用 if 怎麼寫？

參考解答 - 12生崗

<https://ideone.com/T5Fl9t>