

計算機程式設計

C語言 選擇 if/switch

郭忠義

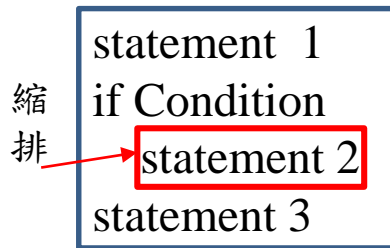
jykuo@ntut.edu.tw

臺北科技大學資訊工程系

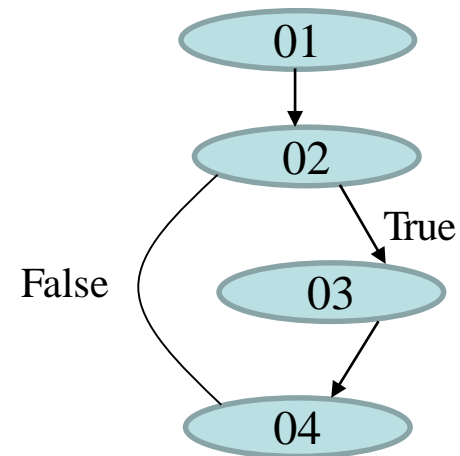
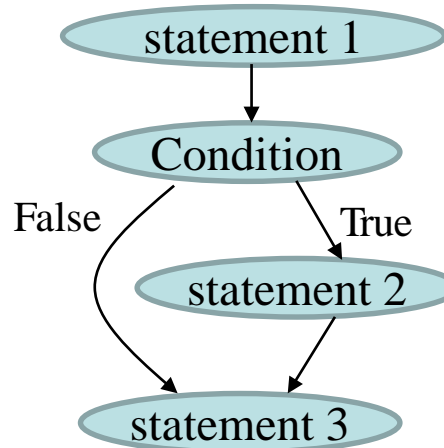
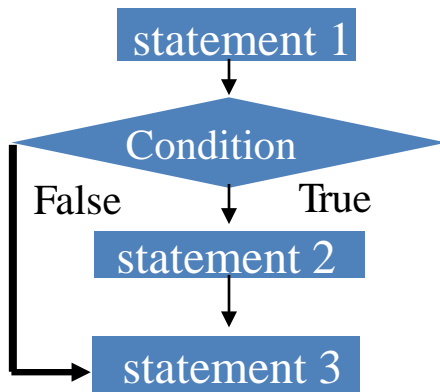
選擇 - if條件判斷

□ 基本語法

- if (condition), if 後跟隨"條件判斷"(Condition)
- statement 2, if本體區塊 (body block), 縮排一層, 若"條件判斷"True, 執行此區指令



```
01 x = 2;  
02 if x*10 == 20  
03     printf("2*10=1024");  
04 x = 3;
```



If 條件判斷

□ 條件判斷，求值是 True/False，1/0

○ 關係比較運算: $x > 10$, $x < 10$, $x == 10$, $x != 10$

比較運算	語法
相等	$A == B$
不等於	$A != B$
大於	$A > B$
小於	$A < B$
大於等於	$A \geq B$
小於等於	$A \leq B$

```
int main() {  
    printf("%d", (2==3));  
    printf("%d", (2!=3));  
    printf("%d", (2 >= 3));  
    printf("%d", (2 <= 3));  
    return 0;  
}
```

If條件判斷

□ 邏輯運算子

not !	A
False	True
True	False

and &&	A B
True	True True
False	True False
False	False True
False	False False

or	A B
True	True True
True	True False
True	False True
False	False False

```
int main() {  
    printf("%d", (2==3) || (3 < 7));  
    printf("%d", (2==3) && (3 < 7));  
    printf("%d", !(3 < 7));  
    return 0;  
}
```

□ 三種關係比較

```
if ((10 < x) && (x < 20)) {  
    printf("%d 在 10~20 範圍內", x);  
}
```

運算子優先順序

- 位置高表示有高優先權
- 同一行，運算時，由左至右
- $a > b + 2$ ， $+$ 優先順序較 $>$ 號高，可表示為 $a > (b + 2)$
- 設計程式最好用括號區別。

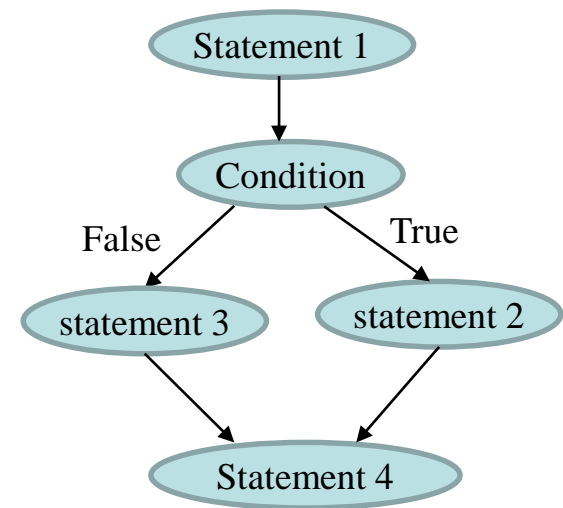
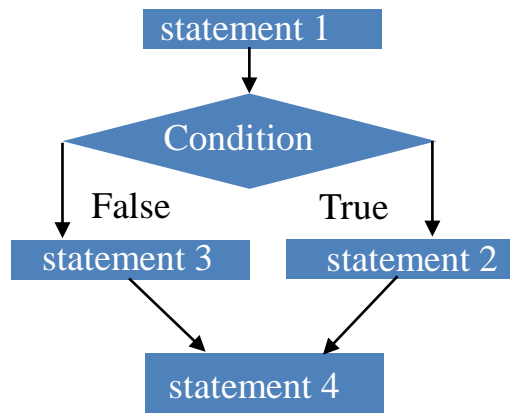
基本運算子優先順序表
！，-（負號）++，--
＊，／，％
＋，－
<，<=，>，>=
==，!=

if-else條件判斷

□ 基本語法

- if (condition) , if 後跟隨"條件"(Condition)
- if "條件"判斷True , 執行statement 2
- if "條件"判斷True , 執行else 的 statement 3
- statement 2, 3 縮排一層

```
statement 1  
if Condition  
    statement 2  
else  
    statement 3  
statement 4
```



if 條件判斷

□ 基本指令

```
void f() {  
    int num;  
    scanf("%d", &num);  
    if (num % 2 == 0) {  
        printf("%d 是偶數", num);  
    }  
    else {  
        printf("%d 是奇數", num);  
    }  
}
```

○ if else 縮寫

- 單行可以不需要{ }
- 一般建議均使用{ }
- 可維護性

```
if (x>y)  
    maxVal = x;  
else  
    maxVal = y;
```

```
maxVal = x ? x>y : y
```

if 條件判斷

- if 條件判斷可以是一個變數

```
void f(int num) {  
    if (num!=0) {  
        printf("%d 不為 0", num);  
    }  
    if (num) {  
        printf("%d 不為 0", num);  
    }  
    if (!num) {  
        printf("%d 為 0", num);  
    }  
}
```


if 條件判斷

❑ 輸入分數，判斷是否及格

```
void myFunction() {  
    int score;  
    printf("Hello~\n輸入分數: ");  
    scanf("%d", &score);  
    if (score >= 60)  
        printf("恭喜你及格");  
    else  
        printf("不及格，要加油");  
}  
  
int main() {  
    myFunction();  
    myFunction();  
    return 0;  
}
```

```
Hello~  
輸入分數:61  
恭喜你及格  
Hello~  
輸入分數:59  
不及格，要加油
```

❑ 輸入超過100，或負數要如何修正？

Exercise

- 若溫度(temperature)高於30而且沒有風wind=0，或濕度(humidity)大於85，印出"開冷氣"，若溫度小於10度，印出"開暖氣"。

- CODE

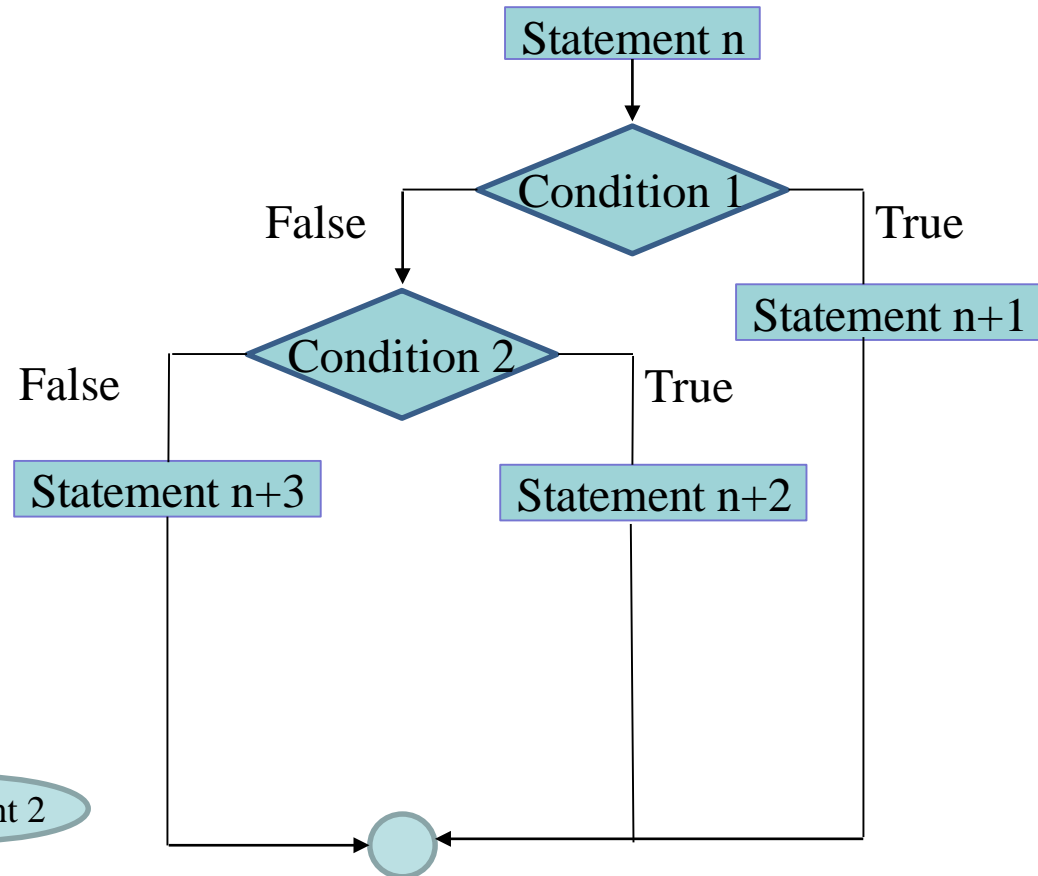
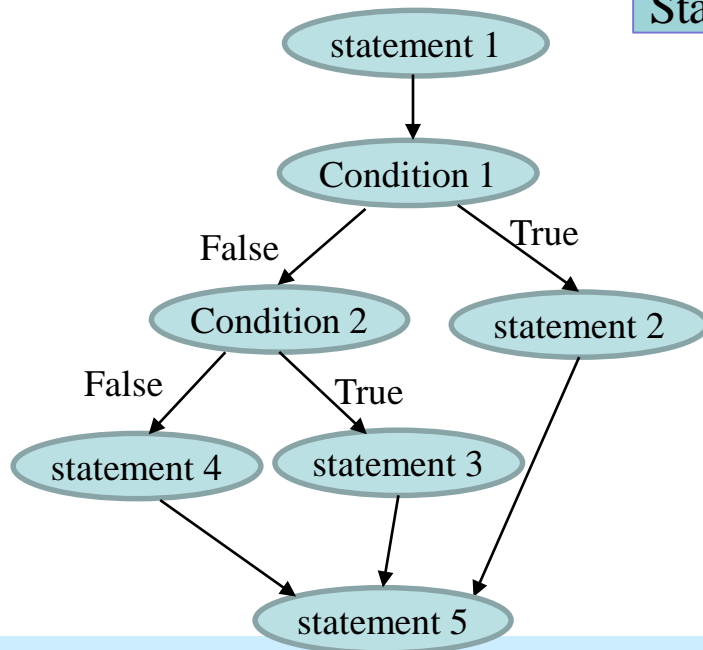
Exercise

- 寫一程式計算一元二次方程式的根 (實根或虛根)

If條件判斷

□ if, else if, else (多選1)

```
Statement n;  
if (Condition 1)  
    Statement n+1;  
else if (Condition 2)  
    Statement n+2;  
else  
    Statement n+3;
```



if條件判斷

□ if, else if, else (多選1)範例(比大小)

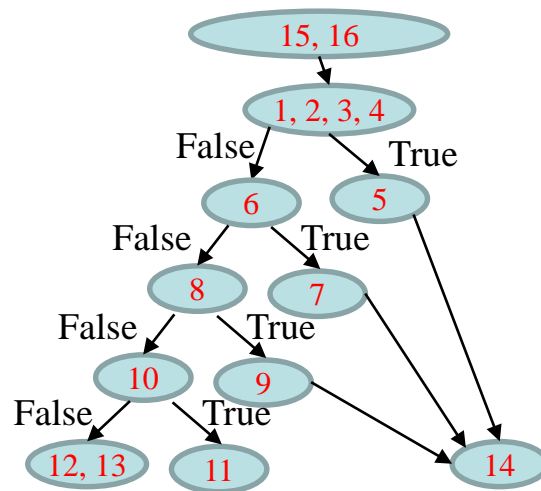
```
void ops() {  
    scanf("%d", &num1);  
    scanf("%d", &num2);  
    if (num1 == num2)  
        printf("%d 等於 %d", num1, num2);  
    else if (num1 < num2)  
        printf("%d 小於 %d", num1, num2);  
    else  
        printf("%d 大於 %d", num1, num2);  
}
```

if條件判斷

❑ 判斷分數等第A, B, C, D, E, F

```
01 void myFunction() {  
02     int score;  
03     scanf("%d", &score);  
04     if (score >= 90)  
05         printf("得 A");  
06     else if (score >= 80 && score < 90)  
07         printf("得 B");  
08     else if (score >= 70 && score < 80)  
09         printf("得 C");  
10     else if (score >= 60 && score < 70)  
11         printf("得 D");  
12     else  
13         printf("不及格");  
14 }  
15 int main(){  
16     myFunction();  
17     myFunction();  
18     myFunction();  
19     return 0;  
20 }
```

輸入分數:50
不及格
輸入分數:90
得 A
輸入分數:70
得 C



這個程式邏輯是否有問題？

if條件判斷

□ 修正判斷邏輯，超過100分或低於 0分之處理

```
01 void myFunction() {
02     int score;
03     scanf("%d", &score);
04     if (score>=90 && score<=100)
05         printf ("得 A");
06     else if (score>=80)
07         printf("得 B");
08     else if (score>=70)
09         printf("得 C");
10     else if (score>=60)
11         printf("得 D");
12     else if (score>=0)
13         printf("不及格");
14     else printf("不是成績範圍0~100");
15 }
16 int main(){
17     myFunction(); myFunction()
18     myFunction();
19     return 0;
20 }
```

Exercise

□ 以下code會輸出？

```
#include <stdio.h>
void f(int key) {
    if (key <1000 && key >=0)
        if (key < 100)
            if (key < 10)
                printf ("1 digit");
            else
                printf("2 digits");
            else
                printf("3 digits");
            else
                printf("Not allowed!");
}
int main() {
    f(99);
    f(0);
    f(7);
    return 0;
}
```


Exercise

□ 判斷何種三角形

○ 當三個邊長能構成三角形時，再判斷該三角形為鈍角、銳角或是直角三角形，其判別方法如下：

- 1. 直角三角形：其中有兩個邊的平方和等於第三邊的平方。
- 2. 鈍角三角形：其中有兩個邊的平方和小於第三邊的平方。
- 3. 銳角三角形：任兩邊的平方和大於第三邊的平方。

○ 輸入三個整數

○ 輸出：顯示直角三角形(Right Triangle)、鈍角三角形(Obtuse Triangle)、銳角三角形(Acute Triangle)或無法構成三角形(Not Triangle)。

□ 測試資料：

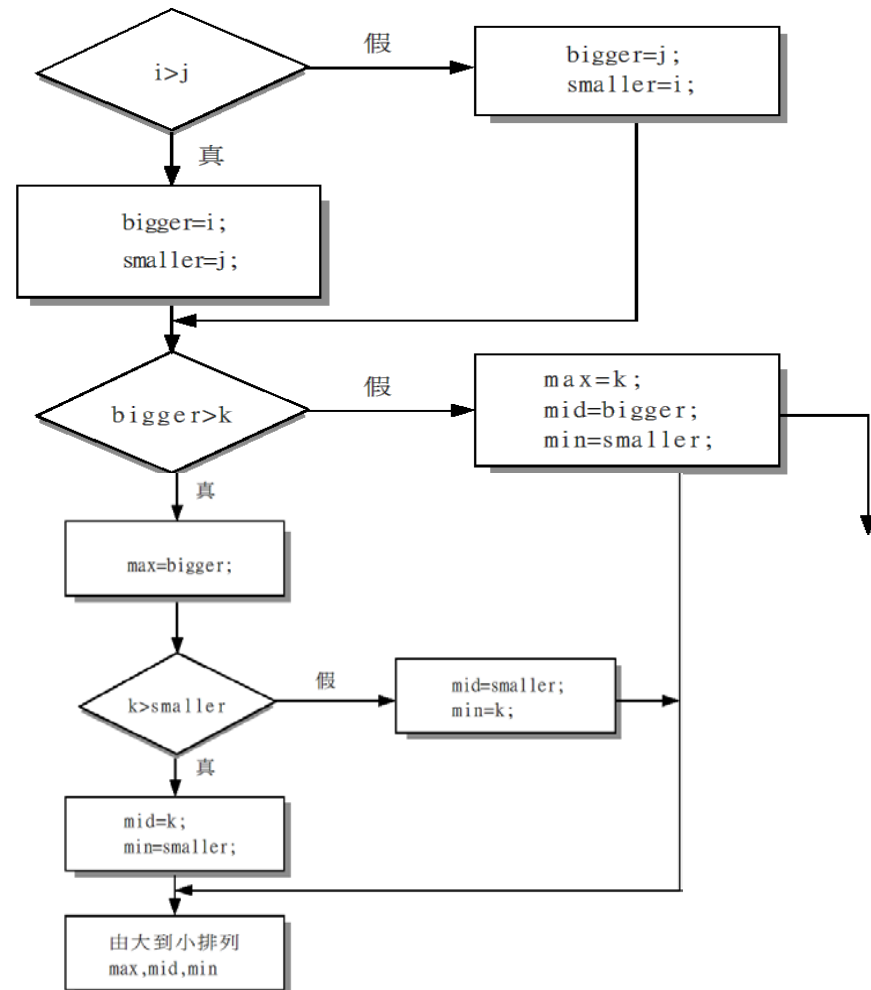
```
input
5 12 13
output
Right Triangle
```

```
input
3 4 5
output
Right Triangle
```

```
input
1 2 3
output
Not Triangle
```

Exercise

- 比較三個數大小流程圖
- 寫一個function比較



Exercise

□ 計算BMI並輸出分級值， $BMI = \text{體重 (kg)} / \text{身高 (m)}^2$

分 級	身體質量指數
體重過輕	$BMI < 18.5$
正常範圍	$18.5 \leq BMI < 24$
過 重	$24 \leq BMI < 27$
輕度肥胖	$27 \leq BMI < 30$
中度肥胖	$30 \leq BMI < 35$
重度肥胖	$BMI \geq 35$

Exercise

- A、B、C三本書價格及折扣表如下，一顧客欲購買A: x 本、B: y 本、C: z 本（ x 、 y 、 z 為使用者輸入），請計算需花費多少錢？

	定價	1~10本	11~20本	21~30本	31本以上
A	380	原價	打9折	打8.5折	打8折
B	1200	原價	打9.5折	打8.5折	打8折
C	180	原價	打8.5折	打8 折	打7折

Exercise

□ Code

```
scanf("%d", &x);
scanf("%d", &y);
scanf("%d", &z);
A_discount = 0;
if (x>=31)
    A_discount = 0.8;
else if (x>=21)
    A_discount = 0.85;
else if (x>=11)
    A_discount = 0.9;
else if (x>=1)
    A_discount = 1;
```

```
B_discount = 0
if (y>=31)
    B_discount = 0.8;
else if (y>=21)
    B_discount = 0.85;
else if (y>=11)
    B_discount = 0.95;
else if (y>=1)
    B_discount = 1;
```

```
C_discount = 0;
if (z>=31)
    C_discount = 0.7;
else if (z>=21)
    C_discount = 0.8;
else if (z>=11)
    C_discount = 0.85;
else if (z>=1)
    C_discount = 1;
```

```
cost= x*380*A_discount + y*1200*B_discount + z*180*C_discount;
printf("The total cost is %d", cost);
```

Exercise

□ 有使用 function的Code

```
def getDiscount(int x, double d0,  
double d1, double d2, double d3){  
    double discount = 0.0;  
    if (x>=31)  
        discount = d0;  
    else if (x>=21)  
        discount = d1;  
    else if (x>=11)  
        discount = d2;  
    else if (x>=1)  
        discount = d3;  
    return discount;  
}
```

```
double A_d0=8.0, A_d1 =8.5, A_d2=9.0, A_d3=1.0;  
double B_d0=8.0, B_d1= 8.5, B_d2= 9.5, B_d =1.0;  
double C_d07, C_d =8.0, C_d2= 8.5, C_d3= 1.0;  
scanf("%d", &x);  
scanf("%d", &y);  
scanf("%d", &z);  
  
A_discount = getDiscount(x, A_d0, A_d1, A_d2, A_d3);  
B_discount = getDiscount(y, B_d0, B_d1, B_d2, B_d3);  
C_discount = getDiscount(z, C_d0, C_d1, C_d2, C_d3);
```

```
cost= x*380*A_discount + y*1200*B_discount + z*180*C_discount  
printf("The total cost is %d" %(cost))
```

Exercise

- A、B、C三本書價格及折扣表如下，一顧客欲購買A: x 本、B: y 本、C: z 本（x、y、z 為使用者輸入），請計算需花費多少錢？
- 幾本區間是否可以輸入，定價表格(區間個數)是否可以輸入？

	定價	1~10本	11~20本	21~30本	31本以上
A	380	原價	打9折	打8.5折	打8折
B	1200	原價	打9.5折	打8.5折	打8折
C	180	原價	打8.5折	打8 折	打7折

HOMework I

- 輸入每月網內、網外、市話、通話時間(sec)及網內、網外簡訊則數，求最佳資費。費率如下表：

資費類型	183型	383型	983型
月租費	183元	383元	983元
優惠內容	月租費可抵等額通信費		
語音 網內	0.08	0.07	0.06
(元/秒) 網外	0.1393	0.1304	0.1087
市話(元/秒)	0.1349	0.1217	0.1018
簡訊 網內	1.1287	1.1127	0.9572
(元/則) 網外	1.4803	1.2458	1.1243

- 輸入

- 網內語音(sec)、網外語音(sec)、市話(sec)、網內簡訊數、網內簡訊數測試資料：

```
input
500 120 13 2 5
output
```


HOMEWORK II

□ 撲克牌

- A, 2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K
- A~10 點數為 1~10，J, K, Q 為 0.5。

□ X, Y 兩個人各發三張撲克牌，加總點數接近 10.5 則贏。

- 超過 10.5 爆掉分數為 0。

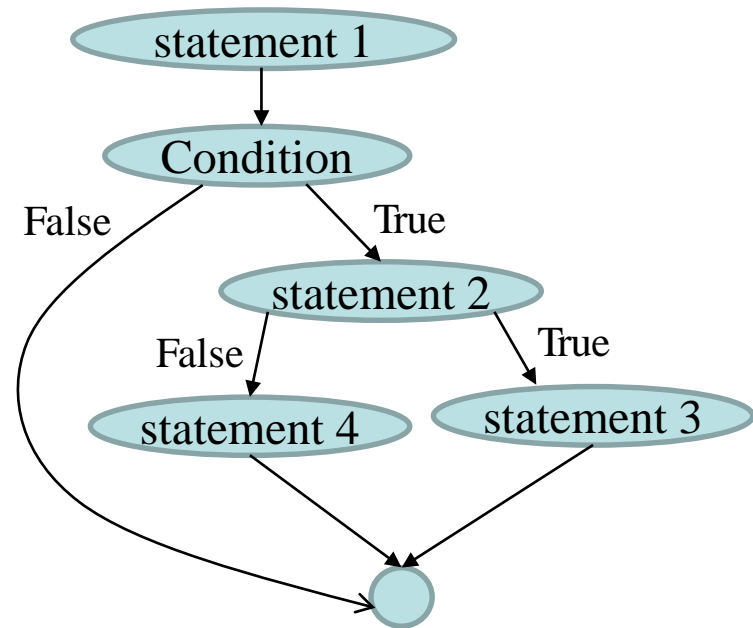
□ 程式

- 輸入 X, Y 兩個人各發的五張撲克牌。
- 輸出兩個人的點數，以及 A 贏或 B 贏或平手。

巢狀 if

- if 內可以再有一層if

```
Statement 1;  
if (Condition 1) {  
    Statement 2;  
    if (Condition 2)  
        Statement 3;  
    else  
        Statement 4;  
}
```



switch-case

- 從數個個案中，選擇出一個執行。

```
switch (條件算式) {  
    case 條件算式值1:  
        動作1:  
        break;  
    case 條件算式值2:  
        動作2:  
        break;  
    default:  
        最後動作:  
        break;  
}
```

switch-case

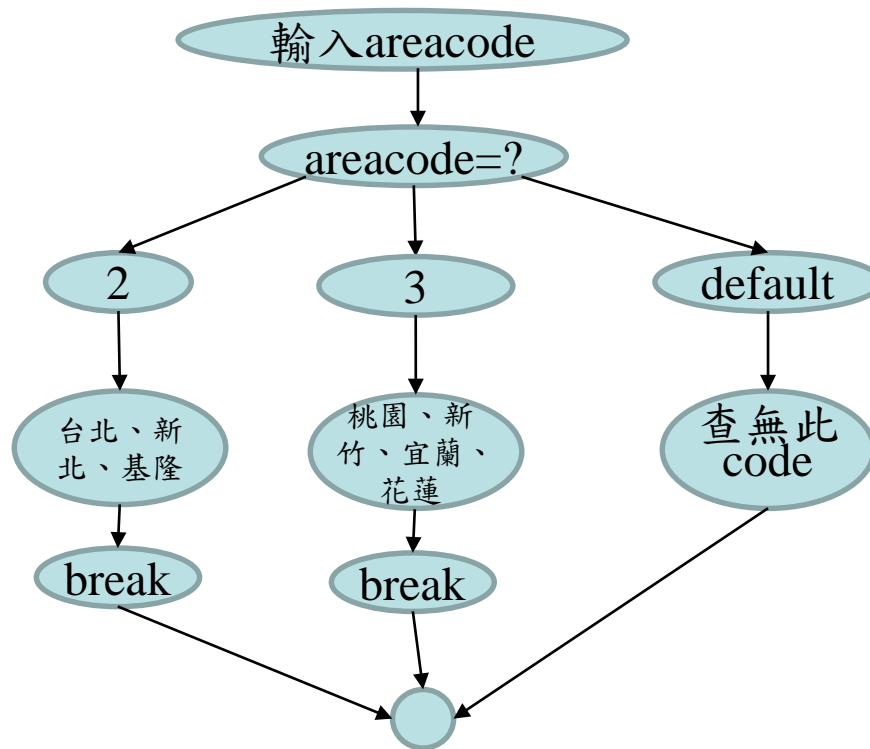
- 從數個個案中，選擇出一個執行。
 - 條件算式：結果為數值或字元的算式，也可以是變數名稱。
 - 根據條件算式，判斷執行哪個 "case"。
 - 適合多選一的條件判斷式。
 - case：在 switch 內的敘述，可同時存在兩個以上。每個 case 都有不同的條件算式值與動作，且應與他 case 不同。
 - break：結束 case 動作，**若沒有 break，會一直執行直到遇見 break**。
 - default：沒有條件算式值。當 switch 找不到符合的 case，便會執行 default 內的程式碼。

switch-case

```
#include <stdio.h>
void findCode(int areaCode) {
    switch(areaCode) { //依區域號碼判斷
        case 2:
            printf("02 是台北市、新北市與基隆市區域號碼\n");
            break;
        case 3:
            printf("03 是桃園市、新竹縣市與宜蘭花蓮縣區域號碼\n");
            break;
        default:
            printf("資料庫無此資料\n");
    }
}
int main() {
    findCode(2);
    findCode(3);
    findCode(9);
    return 0;
}
```

switch-case

- switch 根據 areaCode 的值，選擇執行哪一個 case。
 - 若 areaCode 等於 2，執行 case 2 程式碼；若 areacode 等於 3，則執行 case 3 程式碼。無對應 case 可執行，執行 default 內程式。



Exercise

□ 程式會輸出甚麼？

```
#include <stdio.h>
void findCode(int areaCode) {
    switch(areaCode) { //依區域號碼判斷
        case 2:
            printf("02 是台北市、新北市與基隆市區域號碼\n");
        case 3:
            printf("03 是桃園市、新竹縣市與宜蘭花蓮縣區域號碼\n");
            break;
        default:
            printf("資料庫無此資料\n");
    }
}
int main() {
    findCode(2);
    findCode(3);
    findCode(9);
    return 0;
}
```

Exercise

- ❑ 假設 A~F 六個字元由長度為 4 的二元序列編碼，個程式要從編碼辨識這六個字元。

字元	A	B	C	D	E	F
編碼	0 1 0 1	0 1 1 1	0 0 1 0	1 1 0 1	1 0 0 0	1 1 0 0

輸入

0 1 0 1

正確輸出

A

輸入

0 0 1 0

正確輸出

C

輸入

1 0 0 0

正確輸出

E

Exercise

- 假設 A~F 六個字元由長度為 4 的二元序列編碼，個程式要從編碼辨識這六個字元。

```
#include <stdio.h>
char encode(int a, int b, int c, int d){
    switch(a*1000+b*100+c*10+d) {
    case 101:
        return 'A';
    case 111:
        return 'B';
    case 10:
        return 'C';
    case 1101:
        return 'D';
    case 1000:
        return 'E';
    case 1100:
        return 'F';
    }
}
```

```
int main(){
    int a=0, b=0, c=0, d=0;
    scanf("%d %d %d %d",&a, &b, &c, &d);
    printf("%c\n", encode(a,b,c,d));
    return 0;
}
```

HOMEWORK III

□ 檢查三門課程是否衝堂

- 依序輸入課程編號(數字)、上課小時數(1-3小時)、上課時間(ao 星期1-5, 第1-9節)

輸入說明

1001 (第二門課課程編號)

3 (3小時)

11 (星期1 第1節課)

59 (星期5 第9節課)

25 (星期2 第5節課)

2020 (第二門課課程編號)

...

2030 (第三門課課程編號)

...

輸出說明

(兩課程編號衝突在哪幾節)

1001 and 2020 conflict on 25