

344-111 ชุดวิชาการโปรแกรมและขั้นตอนวิธี

# ผังงาน และ คำสั่งควบคุมทิศทาง



## Outline

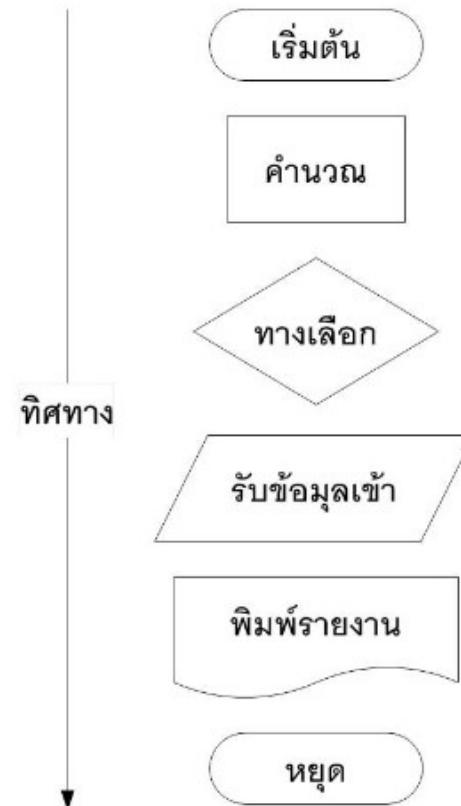
---

- ผังงาน
- คำสั่งควบคุมทิศทาง
  - if
  - if-else
  - switch case

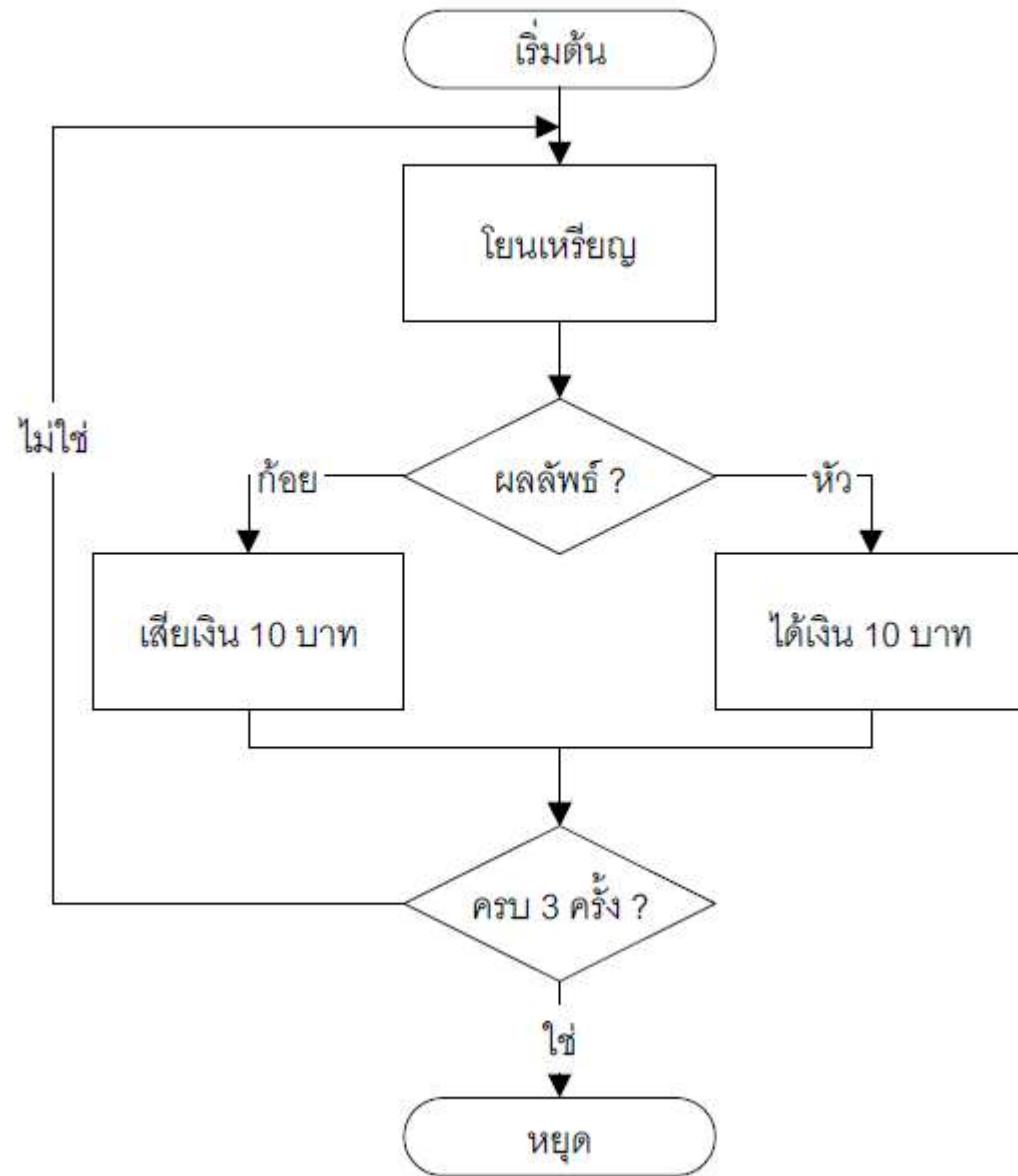


## ผังงาน

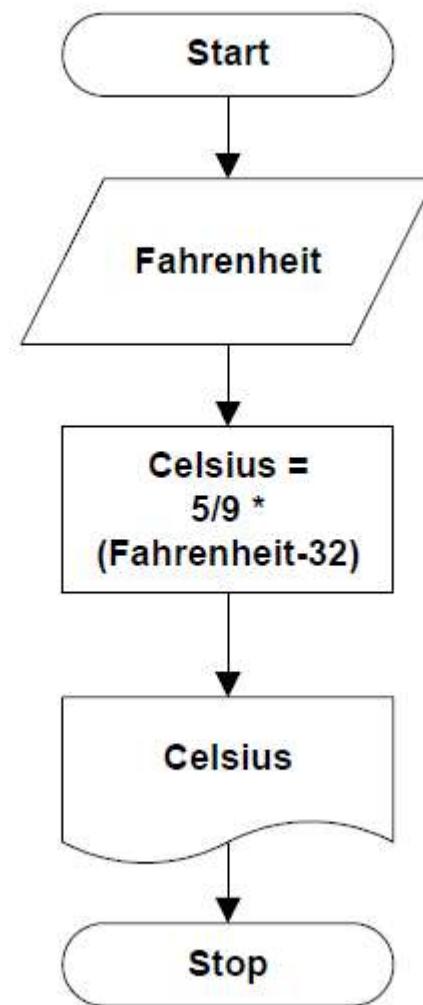
- เพื่อให้เข้าใจการประมวลผลได้สะอาดและดีขึ้นจึงได้มีการนำสัญลักษณ์รูปภาพมาใช้ในการอธิบาย วิธีการประมวลผลแทนคำพูด
- สัญลักษณ์รูปภาพที่นำมาใช้เรียกว่า Flowchart
- ทำไมต้องมี
  - ช่วยลำดับขั้นตอนการทำงาน
  - สร้างความเข้าใจตรงกันในทีม
  - วิเคราะห์การทำงานได้อย่างมีระบบ



## เขียนผังงานสำหรับ โอนหรือถอนสามครั้ง



เขียนผังงานแปลงอุณหภูมิ  
จากองศา พาร์เรนไฮต์ เป็น  
เซลเซียส





## แบบฝึกหัด

---

- เขียน Flowchart สำหรับแปลงค่าเซนติเมตร เป็นเมตร และ กิโลเมตร



## Control statement - if

ในภาษา C คำสั่งสำหรับโครงสร้างการควบคุมเพื่อกำหนดทางเลือก คือประโยค if ซึ่งประกอบด้วยส่วนที่กำหนดน้ำที่ตรวจสอบเงื่อนไขที่เรียกว่า "นิพจน์ทางตรรกะศาสตร์" โดยผลการตรวจสอบมีโอกาสเป็นไปได้ 2 ทาง คือ

จริง หมายความว่า มีค่าไม่เท่ากับศูนย์

ไม่จริง หมายความว่า มีค่าเท่ากับศูนย์

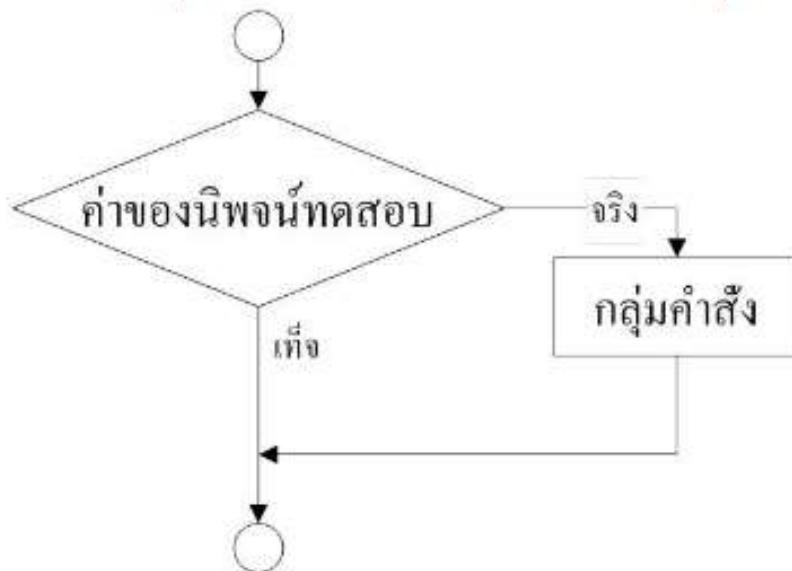
การใช้คำสั่ง if สามารถแบ่งตามลักษณะการทำงานได้ 3 แบบ ดังนี้

- คำสั่ง if แบบทางเดียว (if)
- คำสั่ง if แบบสองทาง (if-else)
- คำสั่ง if แบบหลายทาง (nested- if)



## If แบบทางเดียว

เป็นประโยชน์ควบคุณ if ที่ง่ายที่สุด ในการตัดสินใจประโยชน์ if จะมีทางเลือกให้เพียงทางเดียวเท่านั้น โดยถ้าเงื่อนไขเป็นจริง ก็จะไปทำงานกลุ่มคำสั่งที่กำหนด แต่ถ้าเงื่อนไขเป็นเท็จ ก็จะหลุดออกจากประโยชน์ควบคุณ if



ถ้า นิพจน์ทดสอบ เป็นจริง  
จะทำงานกลุ่มคำสั่ง

ถ้า นิพจน์ทดสอบ เป็นเท็จ  
จะข้ามการทำงานกลุ่มคำสั่ง



## If แบบทางเดียว

### รูปแบบ 1

```
if ( condition )
    statement;
```

### รูปแบบ 2

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

Condition : เงื่อนไขที่กำหนดขึ้นเพื่อใช้พิจารณาว่าจะทำหรือไม่ทำตามคำสั่ง โดยจะต้องเขียนไว้ภายในเครื่องหมาย () ซึ่งเงื่อนไขอาจจะอยู่ในรูปของนิพจน์การคำนวณและเปรียบเทียบ หรือเป็นค่าของตัวแปรก็ได้

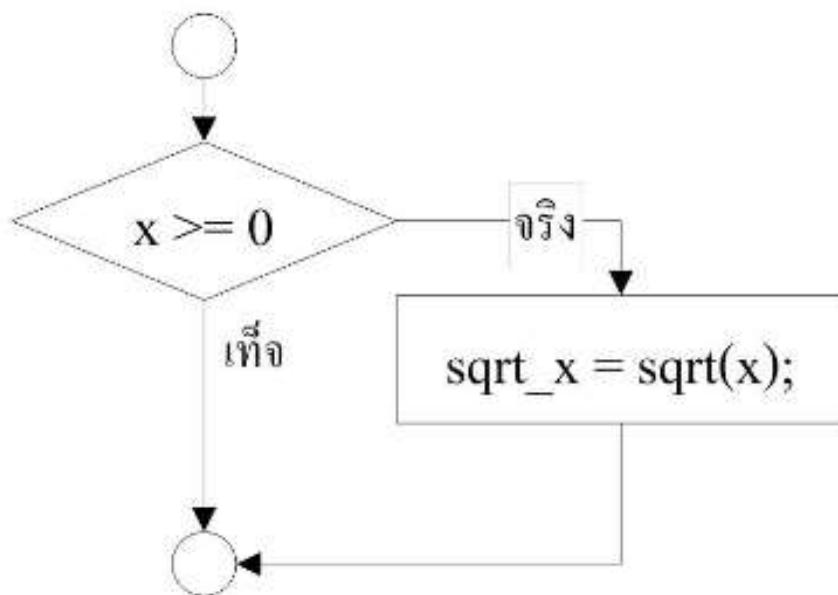
Statement : คำสั่งที่จะให้ทำงานถ้าผลการตรวจสอบเงื่อนไขเป็นจริงโดยอาจจะมีมากกว่า 1 คำสั่งก็ได้แต่ต้องใช้เครื่องหมาย {} ครอบคำสั่งเหล่านั้นไว้ด้วย



## If แบบทางเดียว

ตัวอย่าง : ต้องการหาค่ารากที่สองของ  $x$  โดยมีการตรวจสอบว่า  $x$  มีค่ามากกว่าหรือเท่ากับ 0 จะได้คำสั่งภาษา C เป็นดังนี้

```
if (x>=0)
    sqrt_x = sqrt(x);
```





## If แบบทางเดียว

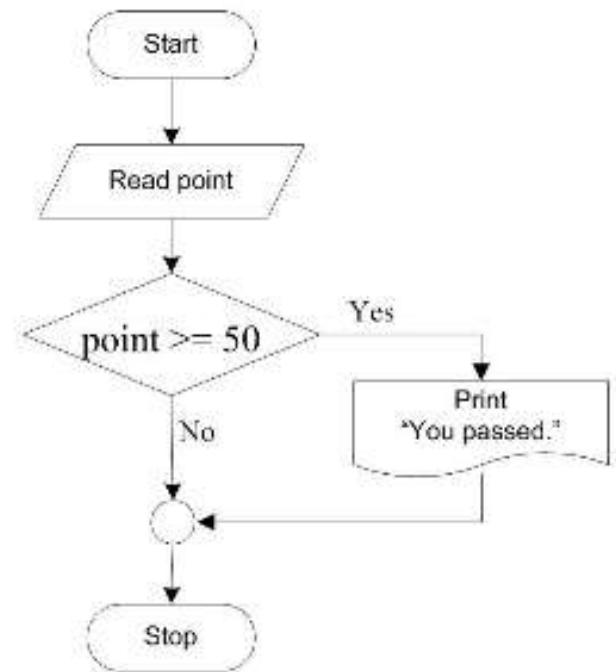
```
#include <stdio.h>
#include <math.h>
int main()
{
    double x, sqrt_x;
    printf("Enter number x : ");
    scanf("%lf", &x);
    if (x >= 0)
    {
        sqrt_x = sqrt(x);
        printf("sqrt(%.2lf) = %.2lf.\n", x, sqrt_x);
    }
    return (0);
}
```

```
graph TD
    Start([Start]) --> Read[/Read x/]
    Read --> Decision{X >= 0}
    Decision -- Yes --> Compute[Compute sqrt(x)]
    Compute --> Print[/Print "sqrt(x)"/]
    Print --> Decision
    Decision -- No --> Stop([Stop])
```



## If แบบทางเดียว

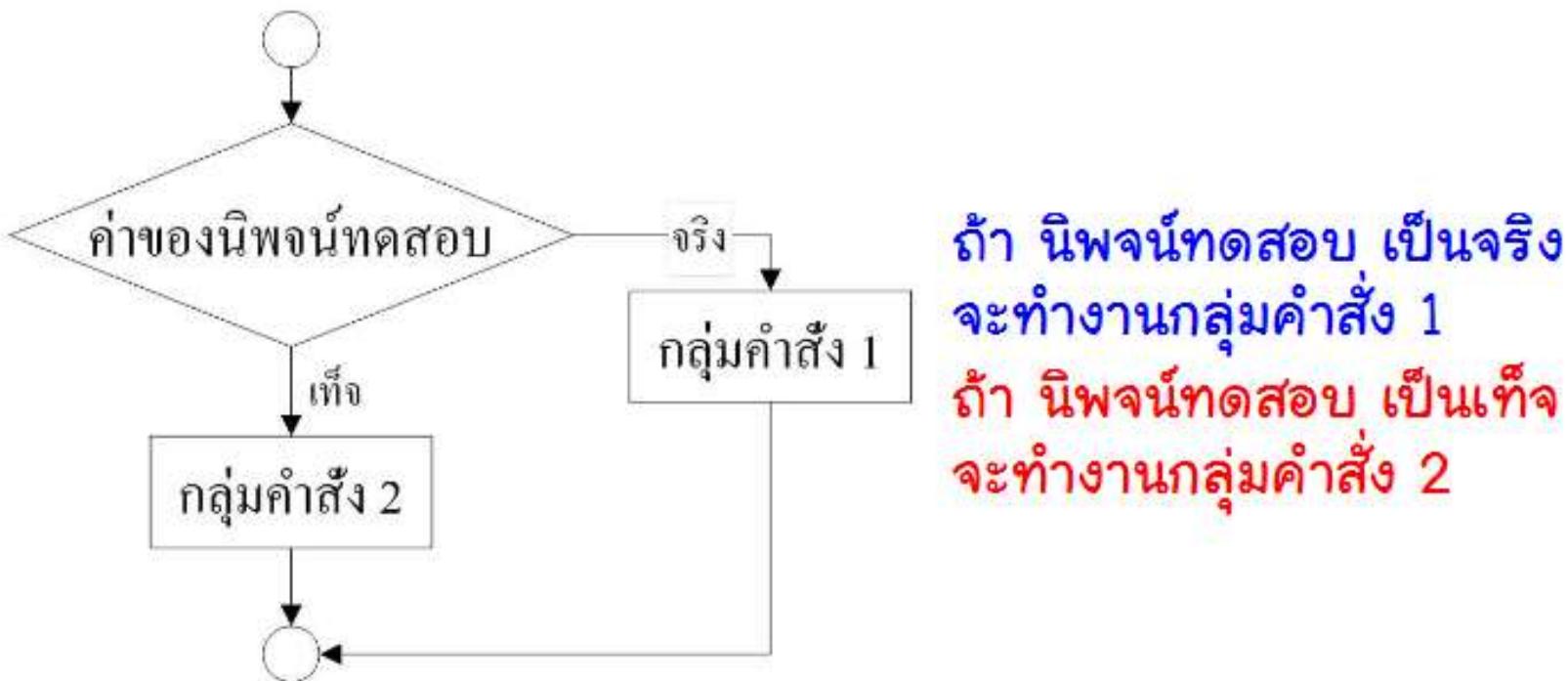
```
#include <stdio.h>
int main()
{
    int point;
    printf("Enter your Exam Points : ");
    scanf("%d",&point);
    if (point >= 50)
        printf("You Passed.\n");
    return (0);
}
```





## If แบบสองทาง (if-else)

- โครงสร้างแบบ if – else คล้ายกับประโยชน์ควบคุณ if แบบทางเดียว เพียงแต่เพิ่มทางเลือกเข้ามา
- ใช้ในการถ้าที่เราต้องการให้มีทางเลือก 2 ทางให้ประมวลผล คือ ทางเลือกที่เงื่อนไขมีค่าเป็นจริง และเงื่อนไขมีค่าเป็นเท็จ





## If แบบสองทาง (if-else)

### รูปแบบ 1

```
if (condition)
    statements1;
else
    statements2;
```

- ถ้าผลการตรวจสอบเงื่อนไขเป็นจริง ก็ทำงานตามคำสั่งของ if
- แต่ถ้าผลการตรวจสอบเงื่อนไขเป็นเท็จ ก็ให้ทำงานตามคำสั่ง ของ else แทน



## If แบบสองทาง (if-else)

### รูปแบบ 2

```
if (condition)
{
    statement-1;
    ...
    statement-n;
}
else
{
    statement-1;
    ...
    statement-n;
}
```

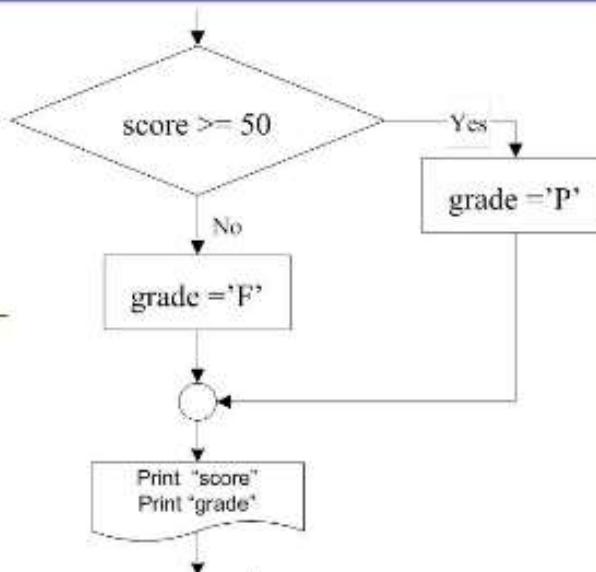
- ถ้าผลการตรวจสอบเงื่อนไขเป็นจริง ก็ทำงานตามคำสั่งของ if
- แต่ถ้าผลการตรวจสอบเงื่อนไขเป็นเท็จ ก็ให้ทำงานตามคำสั่งของ else แทน



ตัวอย่าง : ถ้าค่าในตัวแปร score มากกว่าหรือเท่ากับ 50 ให้ตัวแปร grade มีค่า ‘P’ แต่ถ้าค่าในตัวแปร score น้อยกว่า 50 ให้ตัวแปร grade มีค่า ‘F’

```
if (score >= 50)
    grade = 'P';
else
    grade = 'F';

printf("score = %6.2f\n", score);
printf("grade = %c\n", grade);
```





```
#include <stdio.h>
int main()
{
    int point;
    printf("Enter your Exam Points : ");
    scanf("%d",&point);
    if (point >= 50)
        printf("You Passed.\n");
    else
        printf("You didn't Passed.\n");
    return (0);
}
```

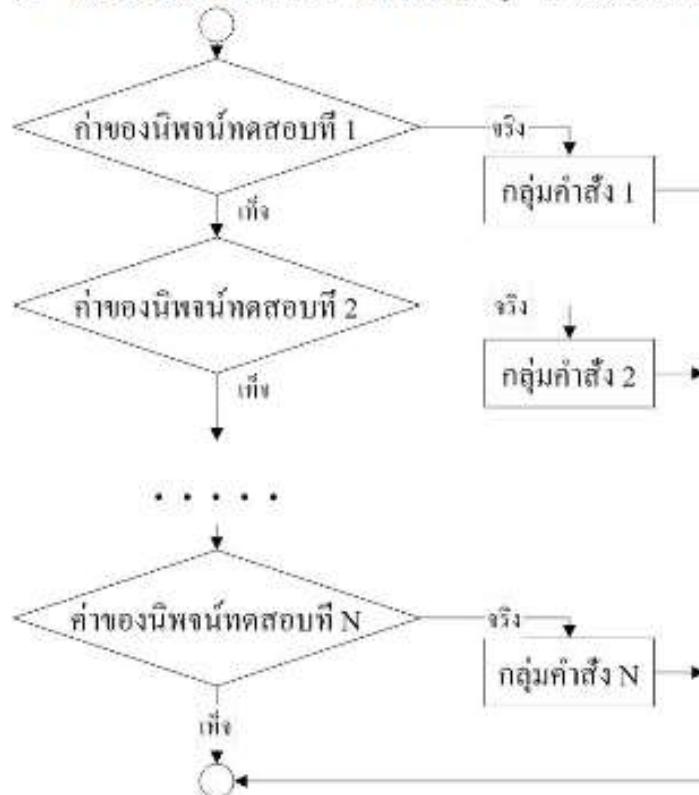
ผลการรัน

Enter your Exam Points : 80  
You Passed.

Enter your Exam Points : 25  
You didn't Passed.

## if-else if

โครงสร้างแบบ if-else if เป็นประโยชน์ควบคุณ if ที่มีความซับซ้อนมากขึ้น จะใช้ในการถ้าที่ทางเลือกมีมากกว่า 2 ทาง ตัวอย่างผังงานประโยชน์ควบคุณ if แบบหลายทางเลือก เป็นดังนี้





## if-else if

### รูปแบบ

```
if (condition-1)
    statement;
else if (condition-2)
    statement;
else if (condition-3)
    statement;
...
else if (condition-n)
    statement;
else
    statement;
```

Statement : คำสั่งที่จะให้ทำงานถ้าผลการตรวจสอบเงื่อนไขเป็นจริง

โดยอาจจะมีมากกว่า 1 คำสั่งก็ได้ แต่ต้องใช้เครื่องหมาย { } ครอบคลุมคำสั่งเหล่านั้นไว้ด้วย

- ทำการตรวจสอบเงื่อนไขแรก ถ้าผลออกมานั้นจริงก็จะทำงานตามคำสั่งของ if
- แต่ถ้าผลออกมานั้นไม่จริงก็จะทำการตรวจสอบเงื่อนไขที่ 2 ซึ่งถ้าผลเป็นจริงก็จะทำงานตามคำสั่งของ else if นั้น
- ถ้าไม่จริงจะทำการตรวจสอบเงื่อนไขอื่นที่เรียงตามลำดับต่อไปจนเมื่อครบทุกเงื่อนไขแล้วถ้าผลยังคงไม่จริง จะทำงานตามคำสั่งที่กำหนดไว้ที่ else



## if-else if

---

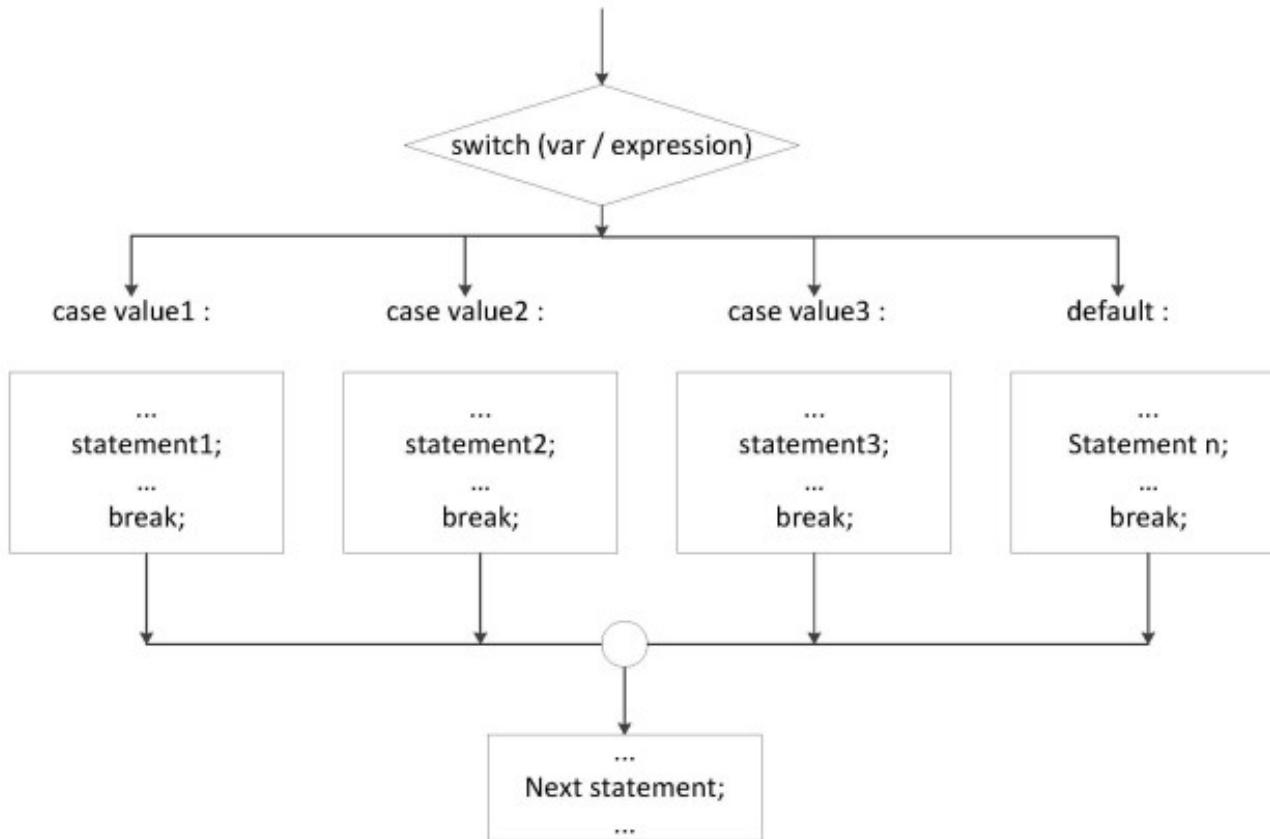
### ตัวอย่าง

```
if (x>0)
    printf("Type Positive Number\n");
else if (x==0)
    printf("Type Zero Number\n");
else
    printf("Type Negative Number\n");
```



## คำสั่ง switch

- ในการเขียนโปรแกรมภาษา C มีประโยชน์ควบคุณ switch ที่ช่วยอำนวยความสะดวกในการเขียนโปรแกรมเพื่อสร้างทางเลือกหลายทาง นونหนีออกจากกรณีใช้ if
- switch มีลักษณะการทดสอบเงื่อนไขซึ่งอาจจะเป็นตัวแปร หรือนิพจน์ กับค่าคงที่ที่อยู่ภายในคำสั่ง case
- ถ้าค่าของตัวแปรหรือนิพจน์ทดสอบ ตรงกับค่าภายในคำสั่ง case ได้ แล้วจะทำงานตามคำสั่งภายในคำสั่ง case นั้น
- แต่หากตรวจสอบแล้วไม่ตรงกับ case ได้เลย จะทำงานภายใต้คำสั่ง default (อาจจะมี หรือ ไม่มีคำสั่ง default ก็ได้ ขึ้นอยู่กับปัญหา)
- จะต้องมีคำสั่ง break เพื่อออกจากการทำงานของ case นั้น โดยไม่ต้องผ่าน case ถัดไป





## รูปแบบ switch

```
switch (variable)
{
    case constant-1 :
        statement sequence;
        break;
    case constant-2 :
        statement sequence;
        break;
    .
    .
    case constant-n :
        statement sequence;
        break;
    default :
        statement sequence;
}
```



## switch: ตัวอย่าง

```
#include <stdio.h>
main()
{ int choice;
    printf("**** Menu ***\n\n");
    printf("1. Coke\n");
    printf("2. PEPSI\n");
    printf("3. Green Tea\n");
    printf("\nPlease Select 1, 2 or 3 : ");
    scanf("%d",&choice);
    switch (choice) {
        case 1 : printf("You Select Coke\n");
        break;
        case 2 : printf("You Select PEPSI\n");
        break;
        case 3 : printf("You Select Green Tea\n");
        break;
        default: printf("Sorry, Out of Menu\n");
    } // end of switch
} // end of main
```



## ตัวอย่าง

จะเขียนโปรแกรมแสดงการรับค่าของประเภทของสินค้า และราคาสินค้า เข้ามาทางแป้นพิมพ์ และนำมาคำนวณหาภาษี โดยคิดแยกตามประเภทสินค้า ดังนี้

ประเภทที่	1	เสียภาษี	10%	ของราคาสินค้า
"	2	"	15 %	" "
"	3	"	17 %	" "
"	4	"	40 %	" "

### ๓.ย. การทำงานของโปรแกรม

Enter type : 2

Enter value : 3500

TYPE = 2

VALUE = 3500.00

TAX = 525.00

Enter type : 1

Enter value : 1500

TYPE = 1

VALUE = 1500.00

TAX = 150.00

Enter type : 7

Enter value : 3000

Unknown type.



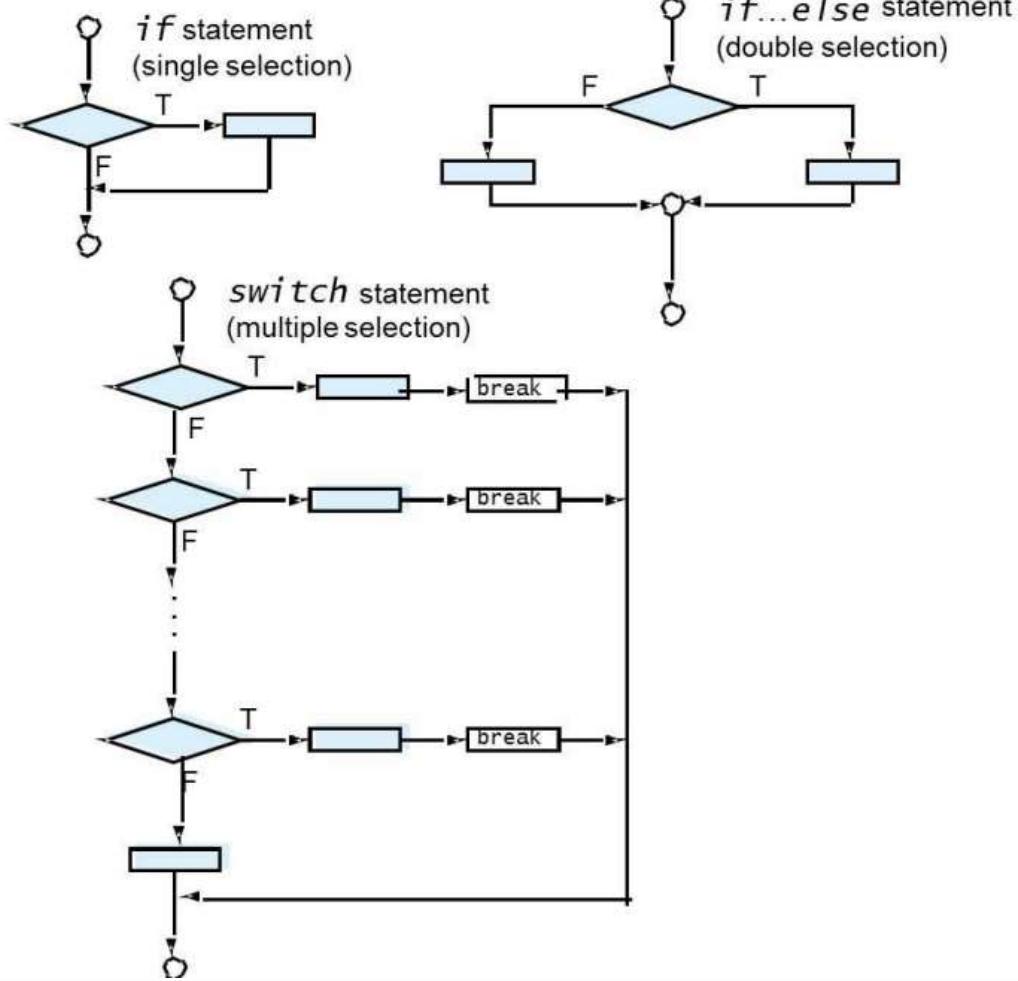


## sequence vs. selection

Sequence



Selection





## Class Work I

---

- แบ่งกลุ่มนักศึกษาเป็น 8 กลุ่ม ให้แต่ละกลุ่มหาผลลัพธ์ที่ได้จากโปรแกรมที่กำหนดให้ตามข้อที่กำหนดให้ในแต่ละกลุ่ม เช่น กลุ่มที่ 1 ทำข้อ 1
- จากแต่ละข้อที่กำหนดให้ต่อไปนี้ พร้อมอธิบายว่า เพราะเหตุใดถึงได้ผลลัพธ์เช่นนั้น



## Problem:Classwork1

---

1

```
#include<stdio.h>

main(){
    int i = 0, j = 0;
    if(i++ == j++)
        printf("%d %d", i--, j--);
    else
        printf("%d %d", i, j);
    return 0;
}
```



## Problem:Classwork1

---

2

```
#include<stdio.h>
main()
{
    int i = 1, j = 0 ;
    if(i-- == j)
        printf("i = %d", --i);
    else
        printf("j = %d", ++j);
    return 0;
}
```



3

```
#include<stdio.h>
main()
{
    int i = 25;
    if(i == 25);
        i = 50;
    if(i == 25)
        i = i + 1;
    else
        i = i + 1;
    printf("%d", i);
    return 0;
}
```



4

```
#include<stdio.h>
main()
{
    int i = 5, j = 6, k = 7;
    if(i > j == k)
        printf("%d %d %d", i++, ++j, --k);
    else
        printf("%d %d %d", i, j, k);
}
```



5

```
#include<stdio.h>
main()
{
    int i = 5, j = 6, k = 7;
    if(i > j == k)
        printf("%d %d %d", i++, ++j, --k);
    else
        printf("%d %d %d", i, j, k);
}
```



---

6

```
#include<stdio.h>
main()
{
    int i = 1;
    i++;
    switch(i--)
    {
        case 1:
        printf("case 1 executed");
        break;
        case 2:
        printf("case 2 executed");
        break;
        default:
        printf("default block executed");
        break;
    }
}
```



7

```
#include<stdio.h>
main()
{
    char ch = 65;
    switch(ch)
    {
        case 'A':
            printf("Apple");
            break;
        case 'B':
            printf("Bing");
            break;
        default:
            printf("Bye");
            break;
    }
}
```



8

```
#include<stdio.h>
main()
{
int i = 1;
switch(i)
{
    case 1:
        printf("Hai ");
    default:
        printf("Bye");
}
}
```



## Class Work 2

---

- แบ่งกลุ่มนักศึกษาเป็น 8 กลุ่ม
- ทำ ClassWork จำนวน 3 ช้อ
- ให้แต่ละกลุ่ม ส่งงานผ่านทาง Ims2 โดยตัวแทนกลุ่ม ตั้งชื่อไฟล์ว่า GroupX.pdf
- นำเสนอในชั้นเรียนรวม โดยเลือกมา 3 กลุ่ม



## เขียนโปรแกรม

### เขียนโปรแกรมสำหรับคำสั่งต่อไปนี้

```
x= 25;  
If(y!=(x-10)  
    x = x-100;  
Else  
    x=x/2;
```

```
x=12.5  
If(y<15.0)  
{  
    if(y>=0.0)  
        x=5*y;  
    else  
        x=2*y;  
}
```



## เขียนผังงาน

### ■ จงเขียนผังงานและวิธีการวิเคราะห์งานในการคำนวณหาค่าจ้างรายวันของพนักงานในบริษัทแห่งหนึ่งมีเงื่อนไขดังนี้

- ถ้าพนักงานคนใดทำงานเกินกว่า 6 ชั่วโมงให้นับชั่วโมงที่เกินกว่า 6 ชั่วโมงเป็นชั่วโมงการทำงานนอกเวลา และให้คิดค่าแรงเป็น 2 เท่าของค่าแรงปกติ
- ถ้าบริษัทให้ค่าแรงปกติชั่วโมงละ 100 บาท