

ฝึกปฏิบัติการครั้งที่ 3

โครงสร้างแบบทางเลือก

วัตถุประสงค์

1. เพื่อให้เข้าใจการทำงานของตัวดำเนินการสัมพันธ์และตัวดำเนินการตรรกะ
2. เพื่อให้รู้จักการใช้ตัวดำเนินการเงื่อนไข (conditional operator)
3. เพื่อให้เข้าใจถึงวิธีการใช้คำสั่งในการควบคุมการไหล (control flow) แบบมีทางเลือกหรือแบบมีเงื่อนไข ซึ่งได้แก่ คำสั่ง if-else และ switch-case

ตอนที่ 1 ตัวดำเนินการสัมพันธ์และตรรกะ (Relational and Logical operators)

ให้พิมพ์โปรแกรมแล้วบันทึก ซึ่งโปรแกรมนี้นี้จะมีส่วนการทำงานใหญ่ๆ 3 ส่วนคือ

ส่วนที่ 1 รับค่าข้อมูลเข้ามาประมวลผล

ส่วนที่ 2 ศึกษาการทำงานของตัวดำเนินการสัมพันธ์

ส่วนที่ 3 ศึกษาการทำงานของตัวดำเนินการตรรกะ

จากนั้นทดลองรันโปรแกรม และพิจารณาผลลัพธ์ที่ได้

```
#include <stdio.h>
int main () {
    int i,j;
    printf("Enter the value of i: ");
    scanf("%d",&i);
    printf("\nEnter the value of j: ");
    scanf("%d",&j);
    /* Relational Operator */
    printf("\nUsing Relational Operator\n");
    printf(" i<j is %d\n", i<j);
    printf(" i<=j is %d\n", i<=j);
    printf(" i==j is %d\n", i==j);
    printf(" i>j is %d\n", i>j);
    printf(" i>=j is %d\n", i>=j);
    printf(" i!=j is %d\n", i!=j);
    /* Logical Operator */
    printf("\nUsing Logical Operator\n");
    printf("AND i&j is %d\n", i&j);
    printf("OR i||j is %d\n", i||j);
    printf("NEGATION !i is %d\n", !i);
    printf("NEGATION !j is %d\n", !j);
    return 0;
}
```

คำถามที่ 1 ให้ทดลองรันโปรแกรม และใส่ค่า i และ j เป็นค่าต่างๆ เช่น 0 1 2 -1 -2 8 7 ฯลฯ ผลลัพธ์ที่เกิดจากตัวดำเนินการสัมพันธ์ (Relational Operator) และตัวดำเนินการตรรกะ (Logical Operator) เป็นอย่างไร

ผลลัพธ์ที่ได้คืออะไร

ตอนที่ 2 ตัวดำเนินการเงื่อนไข (Conditional operator) ? :

โปรแกรม absolute and inverse of integer รับค่าจำนวนเต็ม n แล้วคำนวณหาค่าสัมบูรณ์ (absolute, $|n|$) และค่าส่วนกลับของ n (inverse, $\frac{1}{n}$) กำหนดให้ inverse ของ n เป็น 0 เมื่อ n=0 (ระวัง !!! ห้ามใช้ 0 เป็นตัวหารเด็ดขาด!)

```
//absolute and inverse of integer
#include <stdio.h>
int main (){
    int n, abs_n; float inv_n;
    printf("Enter an integer : ");
    scanf("%d",&n);
    abs_n = n<0 ? -n : n;          //conditional operator
    inv_n = n!=0 ? 1.0/n : 0;      //conditional operator
    printf("|n| = %d\n", abs_n);
    printf("inverse n = %f\n", inv_n);
    return 0;
}
```

คำถามที่ 2 ให้ทดลองรันโปรแกรมและใส่ค่า n เป็นค่าต่างๆ เช่น 0 1 2 -1 -3 -4 7 ฯลฯ และสังเกตผล เพราะเหตุใดจึงใช้ $1.0/n$ เพื่อคำนวณค่า $\frac{1}{n}$? จากนั้นแก้ไขโปรแกรมให้ทำงานเหมือนเดิม แต่ให้ใช้คำสั่ง if-else แทนการใช้ Conditional operator

Hint if(_____) abs_n = _____ else abs_n = _____

ตอนที่ 3 การทำงานโดยใช้คำสั่ง if-else

- ในการทดลองนี้เป็นการใช้คำสั่ง if ในการควบคุมการไหลของโปรแกรมแบบมีทางเลือก จากตัวอย่างเป็นโปรแกรมทดสอบเลข **จำนวนคู่ (even number)** และ**จำนวนคี่ (odd number)** ซึ่งมีวิธีการทดลองดังนี้

- 1) รูปที่ 1 เป็นซอร์สโค้ดของโปรแกรมให้ทำการพิมพ์และรันโปรแกรม
- 2) ทดลองใส่ค่าตัวเลข เช่น 0, 1, 2, -1, -3, 12, -4, 711 สังเกตผลการทดลอง

```
1. #include<stdio.h>
2. int main() {
3.     int num;
4.     printf("num = "); scanf("%d", &num);
5.     if (num%2 == 0)
6.         printf(" %d is even number \n", num);
7.     //else
8.         // printf(" %d is odd number \n", num);
9.     printf("Good Bye");
10.    return 0;
11. }
```

รูปที่ 1 ซอร์สโค้ดของโปรแกรม

- 3) ให้ลบ // ในบรรทัดที่ 7 และ 8 ออก แล้วทำการรันโปรแกรมอีกครั้ง สรุปผลความแตกต่างของข้อ 2) และ 3)

คำถามที่ 3 จงเขียนโปรแกรมเพื่อตรวจสอบตัวเลขจำนวนเต็ม ว่าเป็นเลขคู่ (even) หรือ เลขคี่ (odd) และเป็นจำนวนบวก (positive) หรือจำนวนลบ (negative) หรือเป็นค่าศูนย์ โดยสามารถแสดงผลลัพธ์ซึ่งแบ่งออกเป็น 5 กรณี คือ positive even number, positive odd number, zero, negative even number และ negative odd number

Hint ใช้โครงสร้าง if-else และใช้ ตัวดำเนินการ OR หรือ AND ในการกำหนดเงื่อนไขร่วมด้วย เช่น ถ้าจำนวนเต็มมากกว่าศูนย์และเป็นเลขคู่ แสดงว่าเป็น positive even number เป็นต้น

```
Enter a number: 30  
30 is positive even number  
Bye...
```

```
Enter a number: 0  
0 is zero  
Bye...
```

```
Enter a number: -28  
-28 is negative even number  
Bye...
```

```
Enter a number: -1805  
-1805 is negative odd number  
Bye...
```

รูปที่ 2 ตัวอย่างผลการรันโปรแกรม

ตอนที่ 4 โครงสร้างแบบทางเลือก ด้วยการใช้คำสั่ง switch-case

- ในการทดลองหัวข้อนี้เป็นการใช้คำสั่ง switch ในการควบคุมการไหลของโปรแกรมแบบมีทางเลือก ซึ่งมีวิธีการทดลองดังนี้

- 1) พิมพ์โปรแกรมหาดังรูปที่ 3 จากนั้นทำการรันโปรแกรมและสังเกตผลการทดลอง
- 2) ลบคำสั่ง break ในบรรทัดที่ 12, 15 และ 18 จากนั้นทำการทดลองรันโปรแกรมโดยป้อนค่าต่างๆ อีกครั้ง ผลการรันโปรแกรมแตกต่างไปจากเดิมอย่างไร
- 3) เปลี่ยนชนิดตัวแปรของ group ให้เป็น double แล้วรันโปรแกรมอีกครั้ง สามารถรันโปรแกรมได้สำเร็จหรือไม่เพราะเหตุใด
- 4) ให้เขียนโปรแกรมเพิ่มกรณีดังนี้
 - a. กรณีที่ group = 2 ให้แสดงข้อความว่า poor
 - b. กรณีที่ group = 0 หรือ 1 ให้แสดงข้อความว่า fail
 - c. ในกรณีที่ผู้ใช้ป้อนค่านอกช่วงที่กำหนดให้แสดงข้อความ Out of range

```
1.  #include<stdio.h>
2.  int main() {
3.      int score, group;
4.      printf("Enter your score (0 - 20): ");
5.      scanf("%d",&score);
6.      group = score/4;
7.      switch(group) {
8.          case 5:
9.              printf("Very Good \n");
10.             break;
11.          case 4:
12.              printf("Good \n ");
13.              break;
14.          case 3:
15.              printf("Fair \n ");
16.              break;
17.      }
18.      return 0;
19.  }
```

รูปที่ 3 โปรแกรมคำสั่ง switch-case

คำถามที่ 4 จงเขียนโปรแกรม bmi.c เพื่อรับค่าส่วนสูง (ในหน่วยเมตร) และน้ำหนัก (หน่วยกิโลกรัม kg) ของผู้ใช้ แล้วคำนวณหาค่า BMI (Body Mass Index) ซึ่งจะเป็นตัวระบุว่าผู้ใช้มีรูปร่างอยู่ในเกณฑ์ใด สูตรคำนวณค่า BMI เป็นดังนี้

$$BMI = \frac{weight(kg)}{height(m)^2}$$

BMI	Category	เกณฑ์รูปร่าง
bmi < 18.5	Underweight	ผอมเกินไป
18.5 <= bmi < 25	Normal	ปกติ
25 <= bmi <= 30	Overweight	อ้วนเกินไป
>30	Obese	เป็นโรคอ้วน

ฝึกปฏิบัติทำรายการทดลองนอกห้องเรียน

จงเขียนโปรแกรมเพื่อคำนวณค่านายหน้า โดยโปรแกรมรับค่ายอดขายรวม (Total sales) ของพนักงานขาย แล้วนำมาคำนวณค่านายหน้า (Sales commission) โดยคิดเป็นเปอร์เซ็นต์จากยอดขาย แบ่งออกเป็น 3 ช่วง คือหากยอดขายต่ำกว่า 1000 ให้คิด 5% 1000-5000 ให้คิด 7% และหากเกิน 5000 ให้คิด 10% เช่น ยอดขาย 8000 บาท จะได้ค่านายหน้า 800 บาท, ยอดขาย 2000 บาท จะได้ค่านายหน้า 140 บาท เป็นต้น