



## ใบงานที่ 7

### เรื่อง Recursion

เสนอ

อาจารย์ ปิยพล ยืนยงสถาวร

จัดทำโดย

นายธีรเดช ประเสริฐวงศ์พนา

65543206016-9

ใบงานนี้เป็นส่วนหนึ่งของรายวิชา โครงสร้างข้อมูลและขั้นตอนวิธี

หลักสูตรวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

มหาวิทยาลัยเทคโนโลยีราชมงคลล้านนา

ประจำภาคที่ 1 ปีการศึกษา 2566

## คำสั่ง/คำชี้แจง

\* ให้ระบุคำสั่ง/คำชี้แจง/คำอธิบาย ตามที่กำหนดให้

1. สร้างโปรแกรม Convert infix to postfix ตามตัวอย่างในเอกสารประกอบการสอน 4
2. แสดงโค้ดโปรแกรมเป็นส่วนๆ พร้อมอธิบาย
3. แสดงผลการรันโปรแกรม พร้อมอธิบายการทำงาน
4. สรุปผลการทดลอง

## ลำดับขั้นการทดลอง

\* ให้ระบุขั้นตอนการทดลอง ผลลัพธ์ที่ได้ โดยใช้รูปภาพประกอบ และอธิบายอย่างละเอียด

## แสดงโค้ดโปรแกรมเป็นส่วนๆ พร้อมอธิบาย

### ส่วนที่ 1

```
#include <stdio.h> //use printf()
#include <conio.h> //use getch()
int Number , ans ;
```

ส่วนที่ 1 เป็นที่ประกาศการใช้ library และประกาศตัวแปร ในโค้ดดังนี้

-Library ที่ใช้มี 2 ตัวดังนี้

<stdio.h> คือ ใช้สำหรับฟังก์ชัน printf() ที่ใช้แสดงผลบนหน้าจอ

<conio.h> คือ ใช้สำหรับฟังก์ชัน getch() ที่รอรับอักขระที่ผู้ใช้ป้อนจากแป้นพิมพ์โดยไม่แสดงผลบนหน้าจอ

-ส่วนตัวแปร Number และ ans เป็นตัวแปรที่ถูกประกาศเพื่อใช้เก็บค่าที่ใช้ในการคำนวณแฟกทอเรียล และผลลัพธ์ โดยมีประเภทข้อมูลเป็น int

## ส่วนที่ 2 Factorial()

```
int Factorial( int N ) { //Factorial Function
    int x , y ;
    if( N == 0 ) {
        printf( ".....Roll Back Point\n" ) ;
        return( 1 ) ;
    } else {
        x = N - 1 ;
        printf( "%2d! = %2d * %2d!\n" , N , N , x ) ; //Displaybefore Recursive
        y = Factorial( x ) ;
        printf( "%2d! = %2d * %3d = %5d\n" , N , N , y , y * N )
        ; //Display After Recursive
        return( N * y ) ;
    }
} //End Fn
```

ส่วนที่ 2 ฟังก์ชัน int Factorial(int N) เป็นการคำนวณแฟกทอเรียลของจำนวนเต็มบวก N โดยใช้วิธีการเรียกซ้ำตัวเอง (recursive) ในการทำงาน มีพารามิเตอร์ N ที่เป็นตัวแปรที่รับค่าจำนวนเต็มบวกที่ต้องการคำนวณแฟกทอเรียล

1. ในส่วนของเงื่อนไข if (N == 0) ถ้าค่าของ N เท่ากับ 0 แสดงว่าเรามาถึงจุดสิ้นสุดของกระบวนการแล้ว โปรแกรมจะแสดงข้อความ ".....Roll Back Point" และคืนค่า 1 ออกจากฟังก์ชัน
2. ในกรณีที่เงื่อนไขไม่เป็นจริง โปรแกรมจะทำงานในบล็อก else ดังนี้
  - 2.1  $X = N - 1$  คือ การประกาศตัวแปร x เพื่อเก็บค่า  $N - 1$  ซึ่งเป็นค่าที่ใช้ในการเรียกฟังก์ชัน Factorial ต่อไป
  - 2.2 โปรแกรมจะแสดงผลลัพธ์ในรูปแบบของขั้นตอนก่อนที่จะทำการเรียกฟังก์ชัน Factorial โดยใช้ printf ในการแสดงผลลัพธ์นั้นออกทางหน้าจอ
  - 2.3  $Y = \text{Factorial}(x)$  คือ จะทำการเรียกฟังก์ชัน Factorial อีกครั้งโดยส่งค่า x เข้าไป เมื่อฟังก์ชัน Factorial สิ้นสุดการทำงานแล้ว โปรแกรมจะทำการแสดงผลลัพธ์ในรูปแบบของขั้นตอนหลังจากการเรียกฟังก์ชัน Factorial โดยแสดงผลลัพธ์แฟกทอเรียลของ N และผลคูณระหว่างผลลัพธ์ของการเรียกฟังก์ชัน Factorial และ N
  - 2.4  $\text{return}(N * y)$  คือ จะคืนค่า  $N * y$  ซึ่งก็คือผลคูณระหว่าง N (ค่าปัจจุบัน) และผลลัพธ์ของการเรียกฟังก์ชัน Factorial ที่น้อยลงไป

### ส่วนที่ 3 main

```
int main() {
    printf( "RECRSIVE(FACTORIAL) PROGRAM\n" );
    printf( "=====\n" );
    //N=0;
    while( Number != -999 ) {
        printf( "Enter Number (-999 is END) : " );
        scanf( "%d" , &Number );
        if( Number >= 0 ) {
            printf( "N! = N(N-1)!\n" );
            printf( "-----\n" );
            ans = Factorial( Number ); //Recursive it self
            printf( "\nAnswer N! = %d\n" , ans );
            printf( "-----Finished\n" );
            getch();
        }
    } //End while
    return(0);
} //End Main
```

ส่วนที่ 3 ฟังก์ชัน int main() เป็นส่วนที่ทำหน้าที่รับข้อมูลจำนวนเต็มจากผู้ใช้และเรียกใช้ฟังก์ชัน Factorial เพื่อคำนวณแฟกทอเรียลของจำนวนนั้น ๆ

1. printf("RECRSIVE(FACTORIAL) PROGRAM\n"); คือ การแสดงข้อความบนหน้าจอ เพื่อแสดงชื่อโปรแกรม

2. while (Number != -999) คือวนลูปทำงานตามเงื่อนไขที่ Number ไม่เท่ากับ -999 ซึ่งเป็นเงื่อนไขการหยุดการทำงานของโปรแกรม ดังนี้

2.1 printf("Enter Number (-999 is END) : ") คือการแสดงข้อความให้ผู้ใช้ป้อนจำนวนเต็มที่ต้องการคำนวณแฟกทอเรียล

2.2 scanf("%d", &Number); คือการรับค่าจำนวนเต็มที่ผู้ใช้ป้อนเข้ามาและเก็บไว้ในตัวแปร Number

2.3 if (Number >= 0) คือเป็นการตรวจสอบว่าค่าที่ป้อนเข้ามาไม่น้อยกว่า 0 ซึ่งเป็นเงื่อนไขที่ใช้ในการคำนวณแฟกทอเรียล เพื่อป้องกันการคำนวณด้านลบ

2.3.1 printf("N! = N(N-1)!\n"); คือการแสดงข้อความทางหน้าจอเพื่อแสดงแบบสูตรการคำนวณแฟกทอเรียล

2.3.2 ans = Factorial(Number) คือ เป็นการเรียกใช้ฟังก์ชัน Factorial โดยส่งค่าจำนวนเต็มที่ผู้ใช้ป้อนเข้ามาเป็นอาร์กิวเมนต์ และเก็บผลลัพธ์ที่ได้ลงในตัวแปร ans

2.3.3 printf("\nAnswer N! = %d\n", ans); คือการแสดงผลลัพธ์ของแฟกทอเรียลที่คำนวณได้ทางหน้าจอ

2.3.4 getch(); เป็นการรอผู้ใช้กดปุ่มใดๆ บนคีย์บอร์ดเพื่ดำเนินการต่อหลังจากแสดงผลลัพธ์

2.3.5 return (0); คือการสิ้นสุดการทำงานของฟังก์ชัน main และคืนค่า 0 ซึ่งเป็นค่าที่บอกว่าโปรแกรมทำงานสมบูรณ์และสิ้นสุดการทำงานโดยไม่มีข้อผิดพลาด

## แสดงผลการรันโปรแกรม พร้อมอธิบายการทำงาน

1. โปรแกรมจะแสดงข้อความ "RECRSIVE(FACTORIAL) PROGRAM" และ  
"===== " เพื่อแสดงชื่อโปรแกรมและเส้นขึ้นระหว่างส่วนข้อมูล  
นำเข้าและผลลัพธ์
2. โปรแกรมจะถามผู้ใช้ให้ป้อนจำนวนเต็มเพื่อคำนวณแฟกทอเรียล โดยแสดงข้อความ "Enter  
Number (-999 is END) :

```
RECRSIVE(FACTORIAL) PROGRAM
=====
Enter Number (-999 is END) : 10
```

3. ถ้าจำนวนที่ป้อนเป็นค่าที่ไม่น้อยกว่า 0 โปรแกรมจะทำงานดังนี้
  - 3.1 โปรแกรมจะแสดงข้อความ " $N! = N(N-1)!$ " และ "-----" เพื่อแสดงสูตรการคำนวณแฟกทอเรียล
  - 3.2 โปรแกรมจะเรียกใช้ฟังก์ชัน Factorial(Number) เพื่อคำนวณแฟกทอเรียลของจำนวนที่ผู้ใช้ป้อนเข้ามา โดยเก็บผลลัพธ์ในตัวแปร ans
  - 3.3 โปรแกรมจะแสดงผลลัพธ์ของแฟกทอเรียลที่คำนวณได้และจะรอผู้ใช้กดปุ่มใดๆ เพื่อดำเนินการต่อ

```
Enter Number (-999 is END) : 10
N! = N(N-1)!
-----
10! = 10 * 9!
9! = 9 * 8!
8! = 8 * 7!
7! = 7 * 6!
6! = 6 * 5!
5! = 5 * 4!
4! = 4 * 3!
3! = 3 * 2!
2! = 2 * 1!
1! = 1 * 0!
.....Roll Back Point
1! = 1 * 1 = 1
2! = 2 * 1 = 2
3! = 3 * 2 = 6
4! = 4 * 6 = 24
5! = 5 * 24 = 120
6! = 6 * 120 = 720
7! = 7 * 720 = 5040
8! = 8 * 5040 = 40320
9! = 9 * 40320 = 362880
10! = 10 * 362880 = 3628800
Answer N! = 3628800
-----Finished
```

4. กระบวนการที่ 2-3 จะทำซ้ำไปเรื่อยๆ จนกว่าผู้ใช้จะป้อนค่า -999 เพื่อสิ้นสุดการทำงานของโปรแกรม
5. เมื่อผู้ใช้ป้อนค่า -999 โปรแกรมจะสิ้นสุดการทำงานและจบการทำงานของฟังก์ชัน main โดยคืนค่า 0

### สรุปผลการทดลอง

\* ให้ระบุข้อมูลสรุปผลการดำเนินงาน สิ่งที่ได้เรียนรู้ ปัญหา และวิธีการแก้ไข

สรุปผล : โปรแกรมนี้เป็นโปรแกรมที่กำหนดให้ผู้ใช้งานป้อนจำนวนเต็มเพื่อคำนวณแฟกทอเรียล โดยแสดงผลลัพธ์ของแฟกทอเรียลที่คำนวณได้ทางหน้าจอ และสามารถแสดงกระบวนการคำนวณแบบเรียกตัวเอง (recursive) ได้อย่างชัดเจน

### ตอบคำถามท้ายการทดลอง

\* ตอบคำถามท้ายการทดลอง (ถ้ามี)

.....

.....

.....

.....

### สื่อ / เอกสารอ้างอิง

\* ให้ระบุที่มาของข้อมูลที่ใช้ประกอบการทดลอง เช่น หนังสือ เว็บไซต์ youtube เป็นต้น

-ในเอกสารประกอบการสอนสัปดาห์ที่ 4 ของอาจารย์ปิยพล ยืนยงสถาวร