

LAB 4 โครงสร้างควบคุมแบบทำซ้ำ

ตอนที่ 1 เริ่มต้นกับ for, while, do-while

1. เขียนโปรแกรมในการพิมพ์ตัวเลขจาก 1 – 30 โดยใช้คำสั่งในการวนลูป
2. จากข้อที่ 1 ให้พิมพ์ตัวเลขบรรทัดละ 10 ตัว

ตัวอย่างผลการทำงาน

```
1 2 3 4 5 6 7 8 9 10
11 12 13 14 15 16 17 18 19 20
21 22 23 24 25 26 27 28 29 30
```

3. เขียนโปรแกรมสูตรคูณเลข 5 ให้แสดงผลดังภาพด้านล่าง (ให้ใช้คำสั่งในการวนลูป)

ตัวอย่างผลการทำงาน

```
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45
5 x 10 = 50
5 x 11 = 55
5 x 12 = 60
```

4. ปรับปรุงโปรแกรมข้อที่ 3 โดยให้แสดงสูตรคูณตัวเลขใดๆ ที่รับจากผู้ใ้

ตัวอย่างผลการทำงาน (สมมติว่าผู้ใช้ใส่เลข 8)

```
Enter number: 8
8 x 1 = 8
8 x 2 = 16
8 x 3 = 24
8 x 4 = 32
8 x 5 = 40
8 x 6 = 48
8 x 7 = 56
8 x 8 = 64
8 x 9 = 72
8 x 10 = 80
8 x 11 = 88
8 x 12 = 96
```

5. เขียนโปรแกรมในการนับจำนวนน้ำหนักรวมของบุคคลในลิฟต์ โดยรับน้ำหนักของบุคคลทีละคน แล้วบวกน้ำหนักรวมไปเรื่อยๆ เมื่อน้ำหนักรวมเกิน 400 ให้ขึ้นข้อความ Overload!! แล้วจบโปรแกรม

```
ตัวอย่างผลการทำงาน
Enter weight: 85
Enter weight: 75
Enter weight: 58
Enter weight: 49
Enter weight: 89
Enter weight: 64
Overload!!
```

6. ปรับเพิ่มโปรแกรมข้อที่ 5 โดยให้แสดงผลรวมของน้ำหนักก่อนที่จะเกิน 400 ว่ามีน้ำหนักรวมเท่าไร
7. เขียนโปรแกรมในการรับค่าเกรด (ตัวอักษร A-E) จากผู้ใช้
โดยให้ตรวจสอบถ้าผู้ใช้ใส่ตัวอักษรอื่นที่ไม่ใช่ A-E ให้วนรับค่าเกรดใหม่จนกว่าจะได้เกรดที่ต้องการ (A_E) แล้วแสดงว่าเกรดที่รับเข้ามาคืออะไร

```
ตัวอย่างผลการทำงาน
Enter grade: A
Grade = A
```

```
ตัวอย่างผลการทำงาน
Enter grade: G
Enter grade: W
Enter grade: B
Grade = B
```

8. พิมพ์เครื่องหมาย * ดังรูปโดยใช้คำสั่งวนลูป


```
*
**
***
****
*****
```

ตอนที่ 2 ฝึกความชำนาญ

1. เขียนโปรแกรมรับค่าจำนวนของตัวเลขที่ต้องการหาผลบวก แล้วรับค่าตัวเลขเท่าจำนวนที่ต้องการ จากนั้นหาผลบวกของตัวเลขที่รับเข้ามา แล้วแสดงผลลัพธ์ออกทางจอภาพ

```
ตัวอย่างผลการทำงาน
Enter the number of integers you want to add: 4
Enter 4 integers
2 5 4 3
Sum of entered integers = 14
```

2. เขียนโปรแกรมรับค่าตัวเลข 1 จำนวน แล้วหาผลบวกของตัวเลขในแต่ละหลักรวมกัน

```
ตัวอย่างผลการทำงาน
Enter an integer: 1234
Sum of digits of 1234 = 10    // มาจาก 1+2+3+4
```

3. เขียนโปรแกรมรับค่าตัวเลขจำนวนเต็ม 1 ค่า แล้วทดสอบว่า ตัวเลขนั้นเป็น จำนวนเฉพาะหรือไม่

ตัวอย่างการทำงาน Enter a number: 7 7 is a prime number.	ตัวอย่างการทำงาน Enter a number: 10 10 is not a prime number.
---	---

4. เขียนโปรแกรมรับค่าตัวเลขจำนวนเต็ม 1 ค่า แล้วพิมพ์ตัวเลขจำนวนเฉพาะที่น้อยกว่าหรือเท่ากับตัวเลขที่รับเข้ามา

```
ตัวอย่างการทำงาน
Enter a number: 19
Prime numbers are 2 3 5 7 11 13 17 19
```

5. พิมพ์เครื่องหมาย * ดังรูปโดยใช้คำสั่งวนลู

☆ ☆ ☆ ☆ ☆
 ☆ ☆ ☆ ☆
 ☆ ☆ ☆
 ☆ ☆
 ☆

6. เขียนโปรแกรมตรวจสอบจำนวนกล่องในตู้ขนส่งสินค้าซึ่งสามารถบรรจุกล่องได้มากที่สุด 50 ใบ
เมื่อเริ่มต้นทำงานให้ตู้ขนส่งสินค้าว่างเปล่า ทุกครั้งเมื่อมีการลำเลียงกล่องแต่ละใบเข้าไปไว้ในตู้ขนส่งสินค้า
ให้พนักงานกดเครื่องหมาย + (กด 1 ครั้งแทนการนับกล่อง 1 ใบ)
และเมื่อมีการลำเลียงกล่องออกจากตู้ขนส่งสินค้าให้กดเครื่องหมาย - (กด 1
ครั้งแทนการขนกล่องออกจากตู้ 1 ใบ) เมื่อสิ้นสุดการขนย้ายกล่องให้พนักงานพิมพ์ q
โปรแกรมจะแสดงจำนวนกล่องทั้งหมดที่มีอยู่ในตู้ขนส่งสินค้าให้ทราบ หรือ
ถ้ามีกล่องเต็มตู้ขนส่งสินค้าก่อนที่พนักงานจะพิมพ์ q ให้โปรแกรมหยุดทำงานแล้ว แสดงข้อความ
“FULL”

ตัวอย่างการทำงานแบบที่ 1	ตัวอย่างการทำงานแบบที่ 2
Box: +	Box: +
Box: +	Box: +
Box: +	:
Box: -	Box: + (ซ้ำทั้งหมด 50 ครั้ง)
Box: q	FULL
Sum = 2	

แบบฝึกหัดทบทวน

1. เขียนโปรแกรมในการพิมพ์ตัวเลขกลับจากหลักสุดท้ายมาจนถึงหลักแรก

```
ตัวอย่างผลการทำงาน
Enter a number to reverse: 1234
Reverse of entered number is 4321
```

2. เขียนโปรแกรมรับจำนวนเต็มบวกจากผู้ใช้ 1 จำนวน แล้วคำนวณหาค่าเฉลี่ย ของผลบวกของค่าตั้งแต่ 1 ถึง จำนวนที่ผู้ใช้ป้อนเข้ามา เช่น ผู้ใช้ป้อน 10 ให้หาค่าเฉลี่ยของผลบวกตั้งแต่ 1 ถึง 10

ตัวอย่างผลการทำงาน

Enter n: **10**

Average of the summation from 1 to 10 is 5.50

3. จงเขียนโปรแกรมรับค่าจำนวนเต็ม 10 จำนวน แล้วสรุปว่า เป็นเลขคู่ (even number) กี่จำนวนและเป็นเลขคี่ (odd number) กี่จำนวน

ตัวอย่างผลการทำงาน

Enter 10 integers: **60 12 -3 9 5 17 1103 8 9 10**

There are 4 even numbers and 6 odd numbers.

4. เขียนโปรแกรมเพื่อคำนวณหาจำนวนรยยนต์และรณมอเตอรืไซดไ์นมหาวิทยาลัยแห่งหนึ่ง โดยกำหนดให้ยามรักษาความปลอดภัยที่ประตูเข้าออกเป็นผู้กำหนดค่าให้แก่โปรแกรกดังนี้
- 'a' เมื่อมีรณมอเตอรืไซดไ์เข้า 1 คัน - 'b' เมื่อมีรณมอเตอรืไซดไ์ออก 1 คัน
 - 'c' เมื่อมีรยยนต์เข้า 1 คัน - 'd' เมื่อมีรยยนต์ออก 1 คัน - 'q' เพื่อจบโปรแกรม
- ทั้งนี้ให้โปรแกรม สรุปผลจำนวนรยยนต์และรณมอเตอรืไซดไ์ทุกๆ 10 ครั้งของการเข้าออก
5. (โปรแกรมลูบซ้อนลูบ) ให้เขียนโปรแกรมรับค่าจำนวนเต็ม (n) จากผู้ใช้ทีละ 1 ค่า แล้วคำนวณค่า n! ของค่าที่ผู้ใช้ป้อนเข้ามา โดยโปรแกรมจะหยุดทำงานเมื่อผู้ใช้ป้อนจำนวนเต็มที่น้อยกว่า 1

ตัวอย่างการรันโปรแกรม

Enter n: 4

4! = 24

Enter n: 7

7! = 5040

Enter n: 3

3! = 6

Enter n: 0

STOP