

Computational Science

# Chapter 4: Fundamental of Algorithm Design and Analysis

Act 2: Iteration Structure

## **DTI1306 Computational Science**

Department of Digital Technology for Education

Faculty of Education, Suan Sunandha Rajabhat University

**Content Credit By:** Asst.Prof.Nutthapat Kaewrattanapat, PhD.



**Pasawut Cheerapakorn**

Suan Sunandha Rajabhat University

## Course Description:

วิเคราะห์ เทคนิค วิธีการขั้นตอนการแก้ปัญหา ทักษะการคิดเชิงคำนวณ เชิงนามธรรม ฝึกทักษะในการแก้ปัญหาโดยใช้ขั้นตอนการแก้ปัญหา การย่อยปัญหา การแสดงขั้นตอน การแก้ปัญหา โดยการเขียน บอกเล่า วาดภาพ หรือใช้สัญลักษณ์ ออกแบบและเขียนโปรแกรมโดยใช้ซอฟต์แวร์หรืออุปกรณ์ เทคโนโลยีเบื้องต้น เพื่อไปประยุกต์ใช้ในการแก้ปัญหาในชีวิตประจำวัน ในการตัดสินใจได้อย่างมีประสิทธิภาพและตระหนักถึงการใช้งานสารสนเทศอย่างปลอดภัย พัฒนาโครงงานทางเทคโนโลยีดิจิทัล เพื่อการศึกษาที่มีการบูรณาการกับสาขาอื่น ๆ อย่างสร้างสรรค์และเชื่อมโยงกับชีวิตจริง

The study analyzed how the process solutions, abstract thinking skills, computational skills to solve problems by using the steps to solve the problem of small steps to solve the problem by writing a story or painting the symbol, designers and programmers using software or technology introduction, to use the solution on a daily basis, decisions efficiently and realize the information securely, technological development project.

System Theory

Computational Thinking

Decomposition

Abstraction

Pattern Recognition

Algorithm Design

Design Thinking

Flowchart Design Standard

Flowgorithm

Computer Programming

## Course Outline:

- Chapter 1 - Fundamental of Computational Science
- Chapter 2 - Digital Technology
- Chapter 3 - Digital Literacy
- **Chapter 4 - Algorithm Design and Analysis**
- Chapter 5 - Block-Based Programming
- Chapter 6 - Microbit for Learning
- Chapter 7 - Introduction to Computer Programming
- Chapter 8 - Project Design

# Measurement and Evaluation:

## การวัดและประเมินผล

### 1. ระหว่างการจัดการเรียนรู้

- สอบ Pre-test 0%
- การมอบหมายงาน 20%
- สอบ Post-test 15%
- การมีส่วนร่วมในชั้นเรียน 5%

### 2. การสอบกลางภาค (Midterm Examination)

- ปรนัย 35 ข้อ (35 คะแนน) อัตนัย 1 ข้อ (5 คะแนน) 20%

### 3. โครงการประจำภาคเรียน (Term Project)

- โครงการและการนำเสนอ 20%

### 4. การสอบปลายภาค (Final Examination)

- ปรนัย 35 ข้อ (35 คะแนน) อัตนัย 1 ข้อ (5 คะแนน) 20%

ร้อยละ	ระดับผลการเรียน	ความหมาย
86 – 100	A	ดีเยี่ยม
82 – 85	A-	ดีเยี่ยม
78 – 81	B+	ดีมาก
74 – 77	B	ดี
70 – 73	B-	ค่อนข้างดี
66 – 69	C+	ปานกลางค่อนข้างดี
62 – 65	C	ปานกลาง
58 – 61	C-	ปานกลางค่อนข้างอ่อน
54 – 57	D+	ค่อนข้างอ่อน
50 – 53	D	อ่อน
46 – 49	D-	อ่อนมาก
0 – 45	F	ตก

## Measurement and Evaluation:

ครั้งที่ / สัปดาห์	บทเรียน / หัวข้อ
1	แนะนำรายวิชา การวัดและการประเมินผล หัวข้อเรียนรู้ (Introduction to Course)
2	บทที่ 1 พื้นฐานวิทยาการคำนวณ (Fundamental of Computational Science)
3	บทที่ 2 พื้นฐานเทคโนโลยีดิจิทัล (Digital Technology)
4	บทที่ 3 พื้นฐานการรู้เท่าทันดิจิทัล (Digital Literacy)
5	บทที่ 4 พื้นฐานการวิเคราะห์และออกแบบอัลกอริทึม (Algorithm Design and Analysis)
6	<b>บทที่ 4 พื้นฐานการวิเคราะห์และออกแบบอัลกอริทึม (Algorithm Design and Analysis) [ต่อ]</b>
7	บทที่ 5 การโปรแกรมแบบ Block-Based ด้วย Scratch (Block-Based Programming) [ต่อ]
8	บทที่ 5 การโปรแกรมแบบ Block-Based ด้วย Scratch (Block-Based Programming) [ต่อ]

## Measurement and Evaluation:

ครั้งที่ / สัปดาห์	บทเรียน / หัวข้อ
9	สอบกลางภาค (Midterm Examination)
10	บทที่ 6 การโปรแกรมไมโครคอนโทรลเลอร์เบื้องต้นด้วย Microbit (Microbit for Learning)
11	บทที่ 6 การโปรแกรมไมโครคอนโทรลเลอร์เบื้องต้นด้วย Microbit (Microbit for Learning) [ต่อ]
12	บทที่ 7 การเขียนโปรแกรมคอมพิวเตอร์เบื้องต้น (Introduction to Computer Programming)
13	บทที่ 8 การออกแบบโครงงานทางเทคโนโลยีเพื่อการศึกษา (Project Design)
14	สอบปลายภาค (Final Examination)
15	นำเสนอและส่งโครงงาน (Project Pitching and Presentation)
16	

# Pre Test

**Question:**

**1. ข้อใดอธิบายความหมายของ “อัลกอริทึม (Algorithm)”  
ได้ถูกต้องที่สุด**

**A**

โปรแกรมที่เขียนด้วย  
ภาษา Python

**C**

ชุดขั้นตอนที่ชัดเจนเพื่อแก้  
ปัญหาหรือดำเนินงาน

**B**

ชุดคำสั่งที่ทำงานแบบสุ่ม

**D**

ผังงานที่ใช้วาดแทนโปรแกรม

**Question:**

## 2. ข้อใด ไม่ใช่ คุณลักษณะของอัลกอริทึมที่ดี

**A**

มีจุดสิ้นสุดการทำงาน

**C**

ขึ้นกับภาษาโปรแกรม

**B**

ไม่คลุมเครือ

**D**

มีขั้นตอนตายตัว

**Question:**

**3. อัลกอริทึมหรือผังงานที่มีโครงสร้างแบบใด  
ที่ทำงานแบบเรียงลำดับ**

**A** Sequential Structure

**C** Iteration Structure

**B** Conditional Structure

**D** Decision Structure

**Question:**

## 4. อัลกอริทึมหรือผังงานที่มีโครงสร้างแบบใด ที่ทำงานแบบทำซ้ำ

**A** Sequential Structure

**B** Conditional Structure

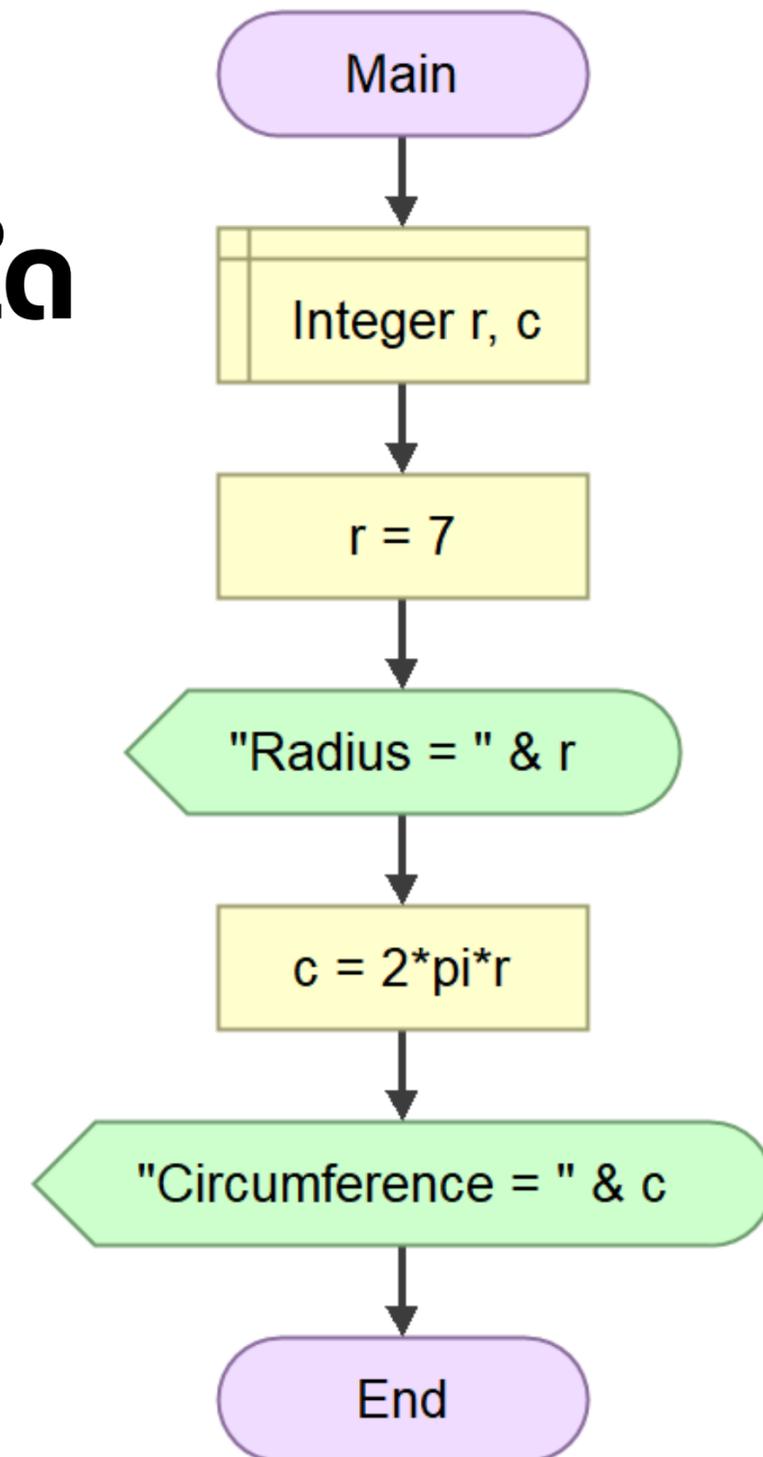
**C** Iteration Structure

**D** Decision Structure

**Question:**

## 5. จากภาพเป็นโครงสร้างผังงานแบบใด

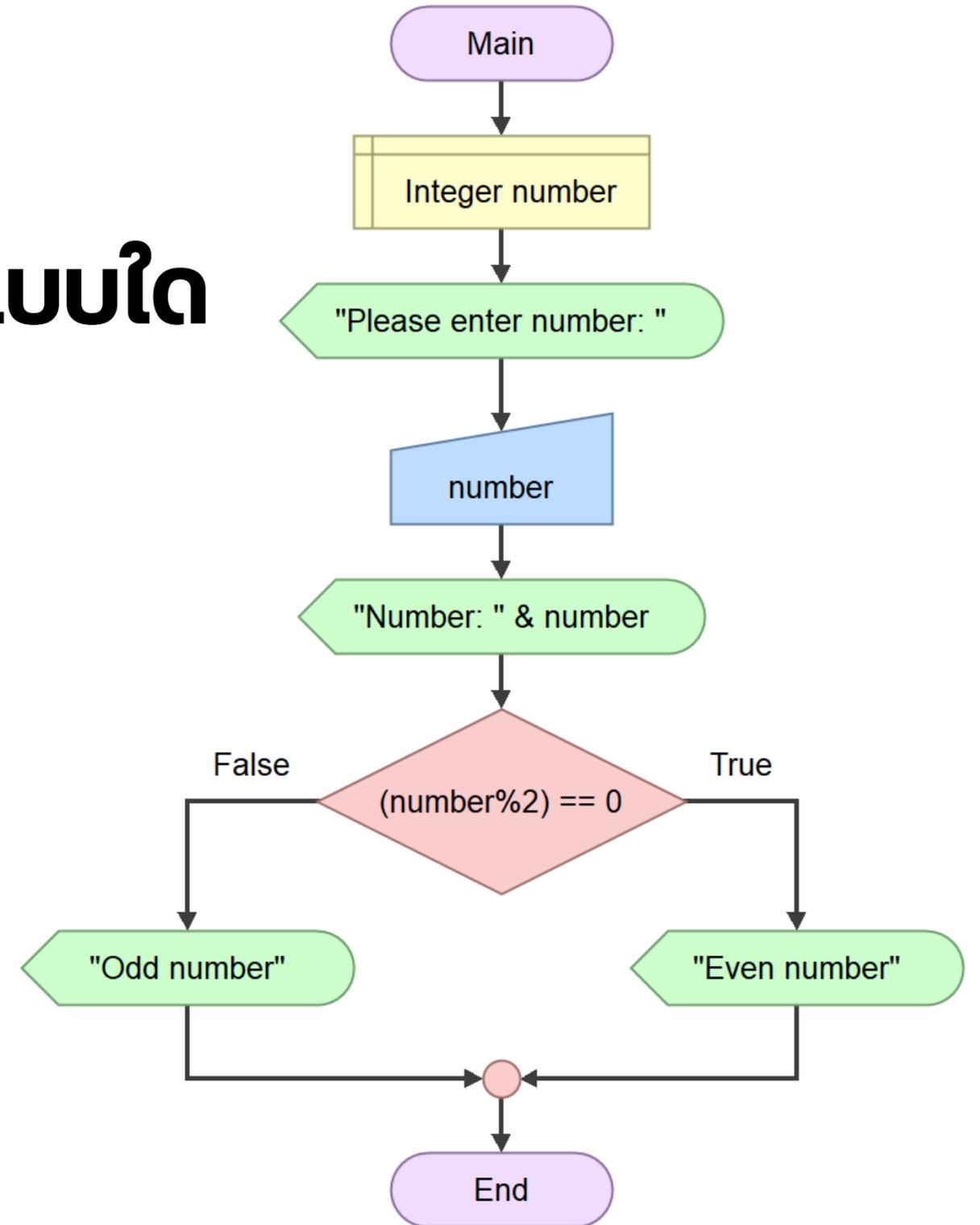
- A** Iteration Structure
- B** Sequential Structure
- C** Conditional Structure
- D** Decision Structure



**Question:**

## 6. จากภาพเป็นโครงสร้างผังงานแบบใด

- A** Iteration Structure
- B** Sequential Structure
- C** Conditional / Decision Structure
- D** ไม่มีข้อใดถูก



**Question:**

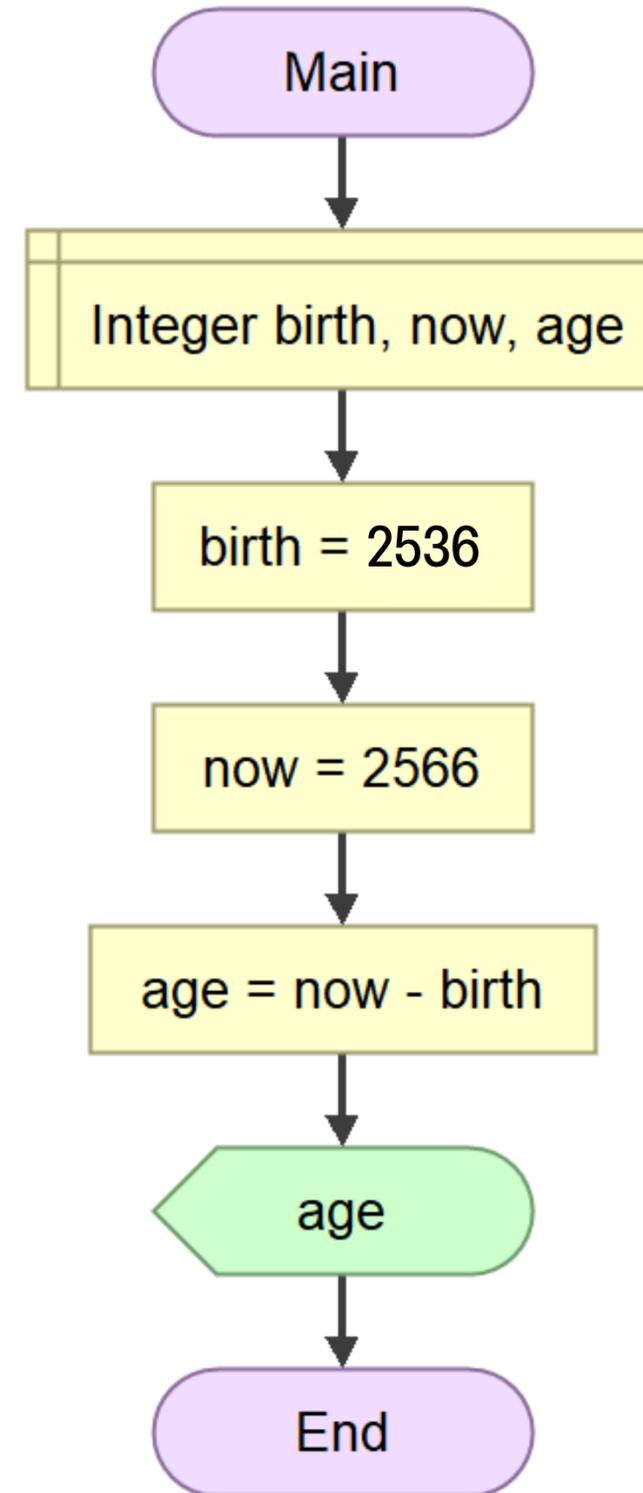
**7. จาก Algorithm นี้ผลลัพธ์ที่  
แสดงออกมา คือข้อใด**

**A** 25

**B** 27

**C** 30

**D** 33



**Question:**

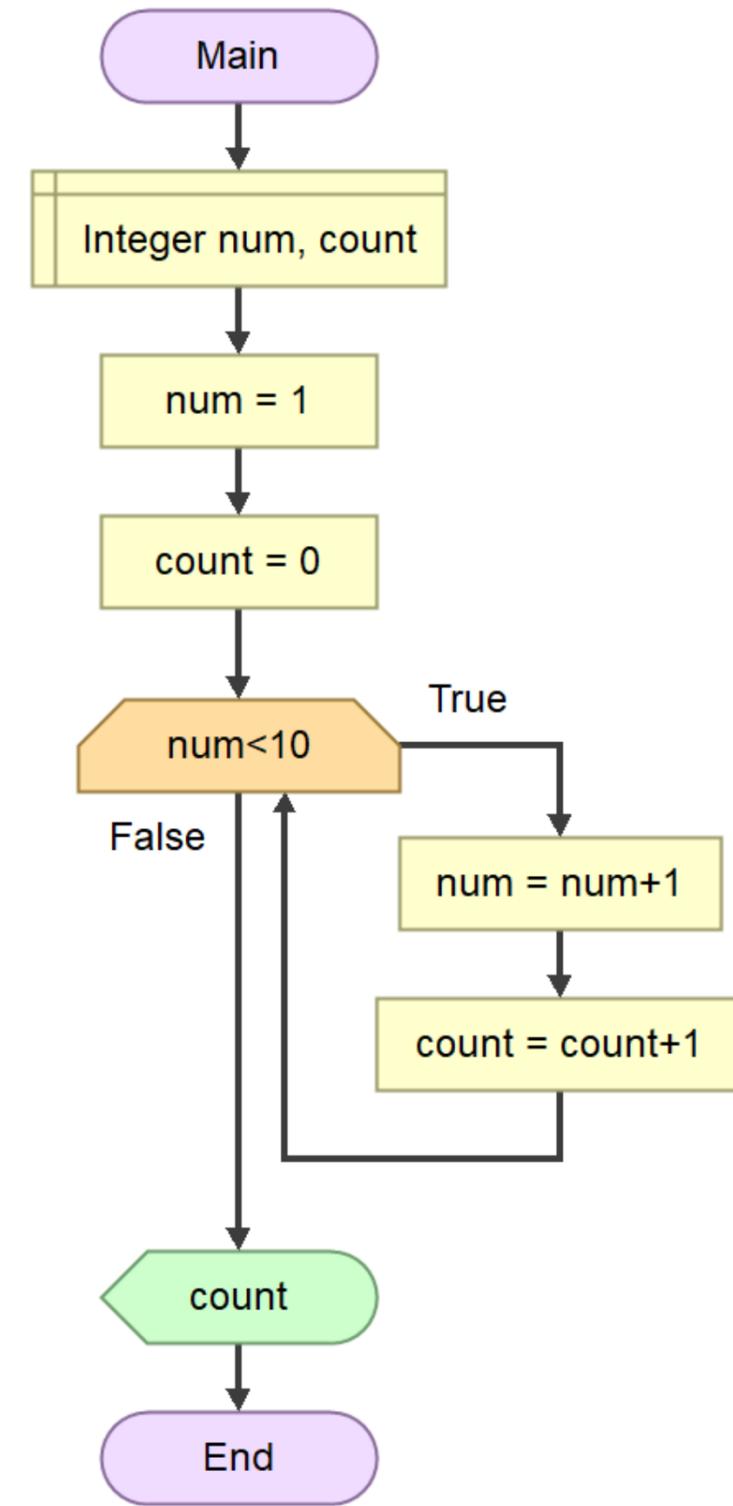
**8. จาก Algorithm นี้ผลลัพธ์ที่  
แสดงออกมา คือข้อใด**

**A** 0

**B** 9

**C** 10

**D** 11



**Question:**

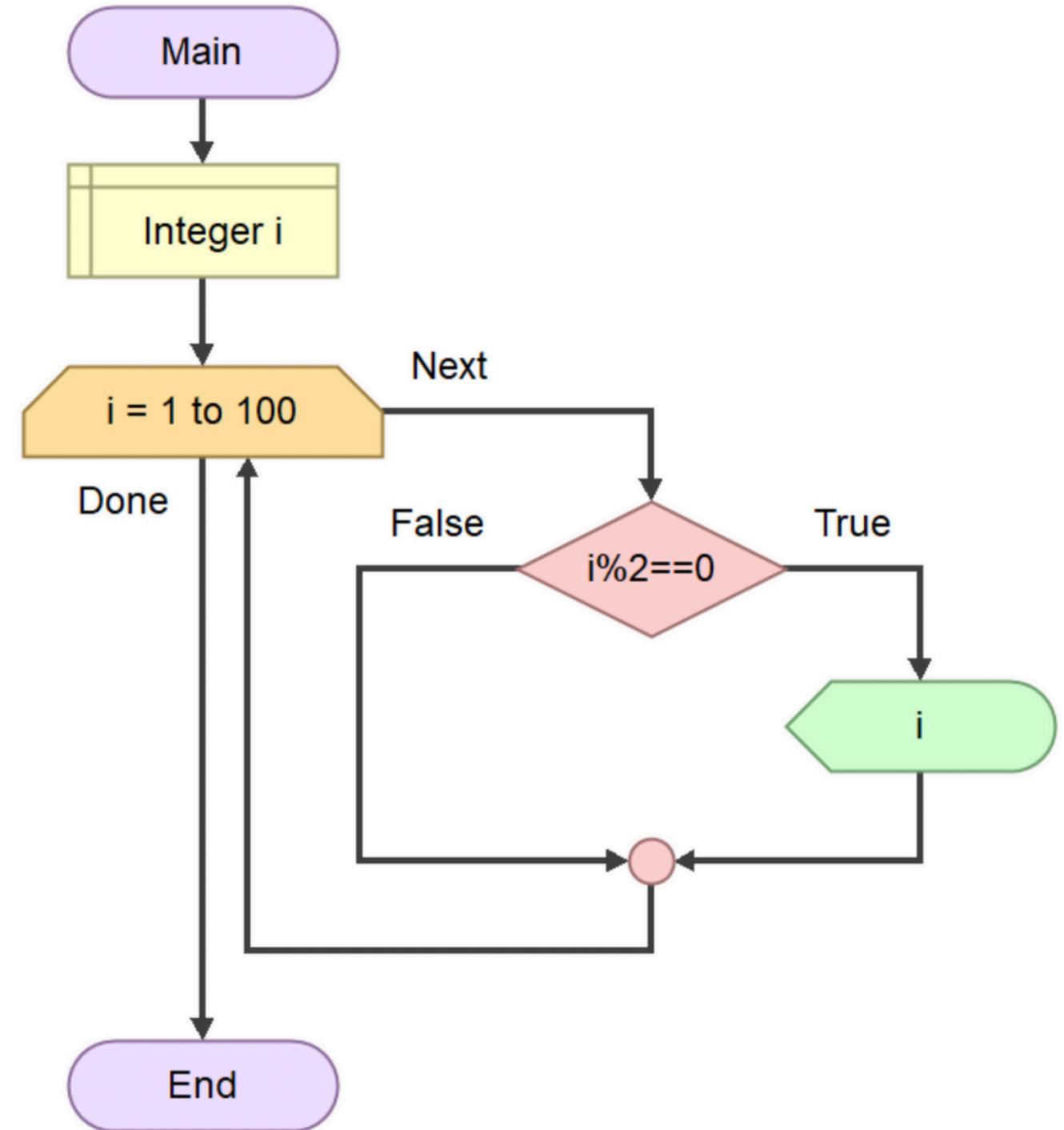
**9. จาก Algorithm นี้ผลลัพธ์ที่  
แสดงออกมา คือข้อใด**

**A** 1, 2, 3, ..., 100

**C** 2, 4, 6, ..., 98, 100

**B** 1, 2, 3, ..., 99

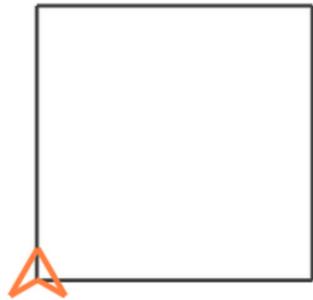
**D** 1, 3, 5, ..., 97, 99



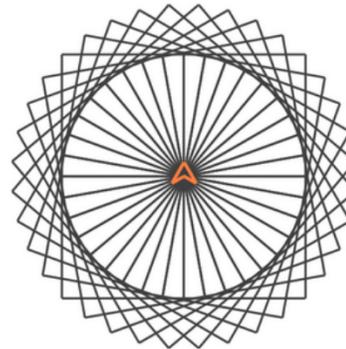
Question:

10. จาก Algorithm นี้ผลลัพธ์ที่  
แสดงออกมา คือข้อใด

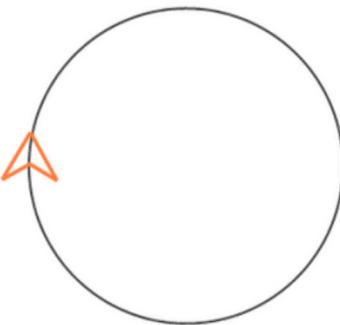
A



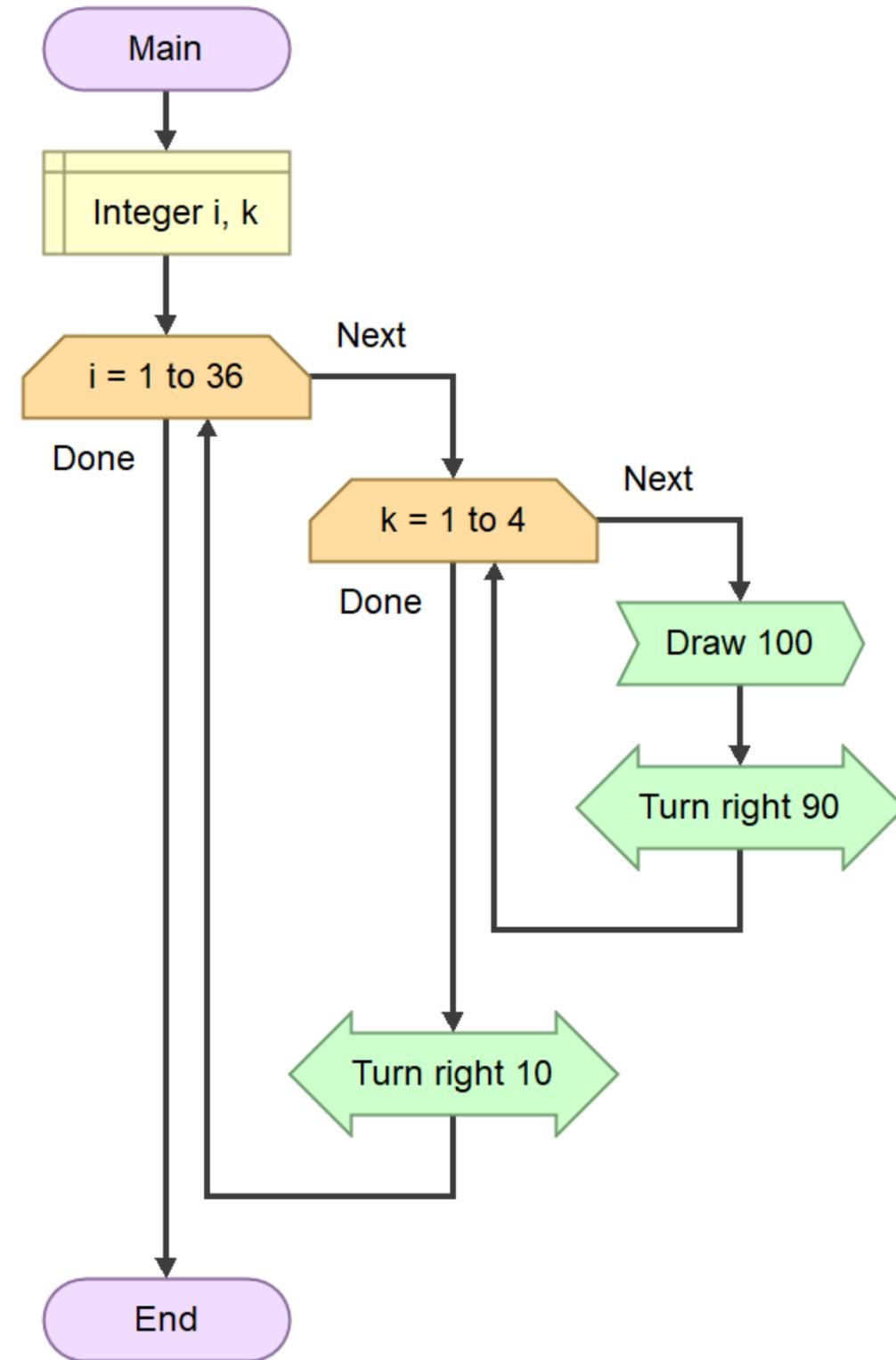
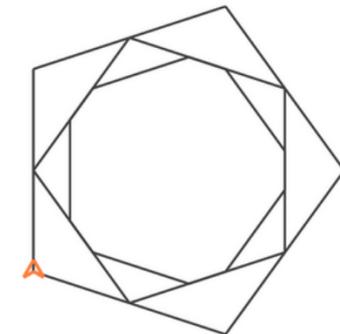
C



B



D



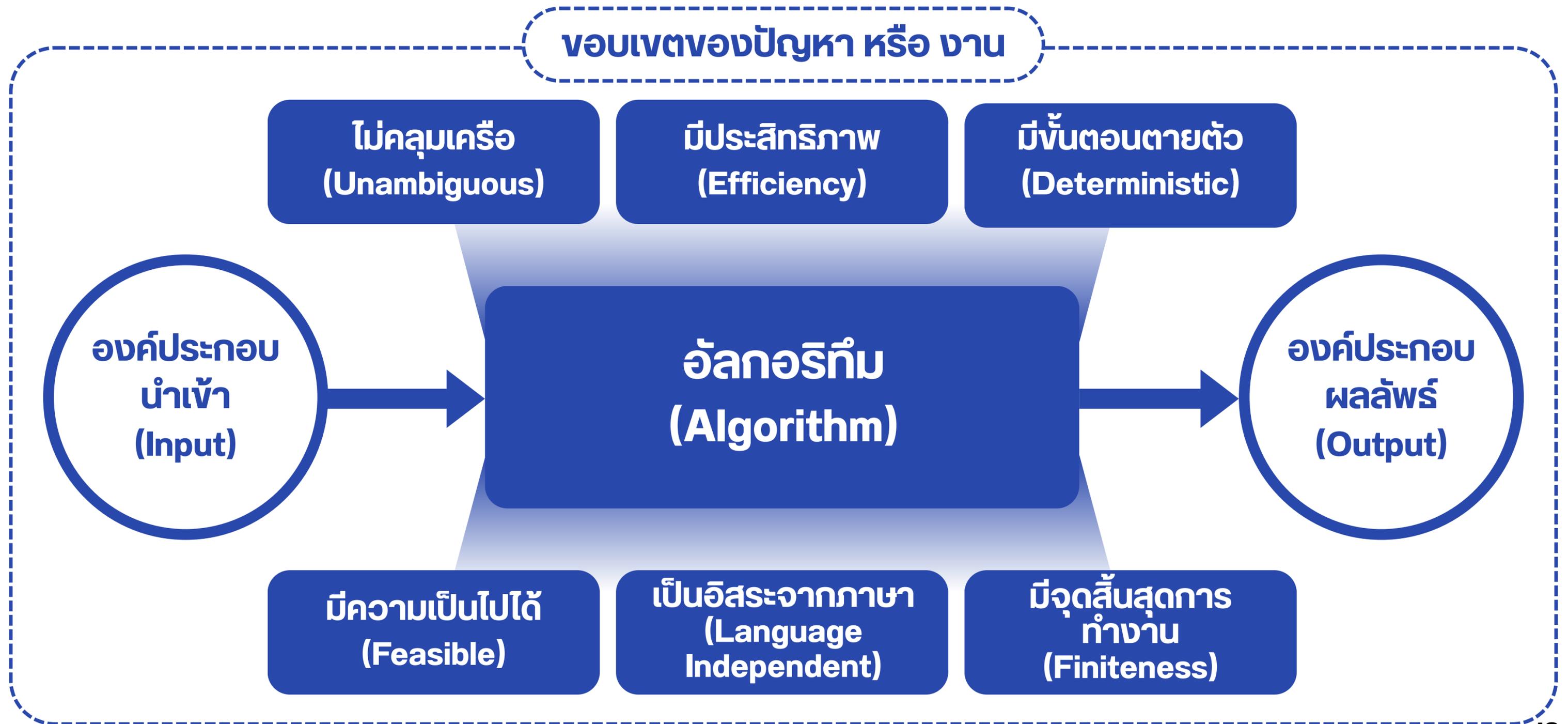
# อัลกอริทึม (Algorithm)

**อัลกอริทึม (Algorithm)** คือ ชุดของขั้นตอนและกฎกติกาที่ถูกกำหนดมา เพื่อแก้ปัญหาหรือดำเนินกระบวนการที่สามารถทำงานได้ด้วยมนุษย์ คอมพิวเตอร์หรือเครื่องจักรอัตโนมัติ ซึ่งขั้นตอน (Algorithm) เป็นการปฏิบัติตามลำดับและหลักการที่แน่นอน เพื่อให้ได้ผลลัพธ์ที่ถูกต้องและเชื่อถือได้ทุกครั้งที่ทำซ้ำ หรือได้ผลลัพธ์ภายใต้เงื่อนไขที่เหมือนกันเสมอ อัลกอริทึมเป็นสิ่งสำคัญที่ส่วนใหญ่ในการแก้ปัญหา การพัฒนาซอฟต์แวร์ การวิเคราะห์ข้อมูล การเรียนรู้ของเครื่อง การค้นหาข้อมูล และงานอื่น ๆ ที่เกี่ยวข้องกับการประมวลผลทางด้านคอมพิวเตอร์.

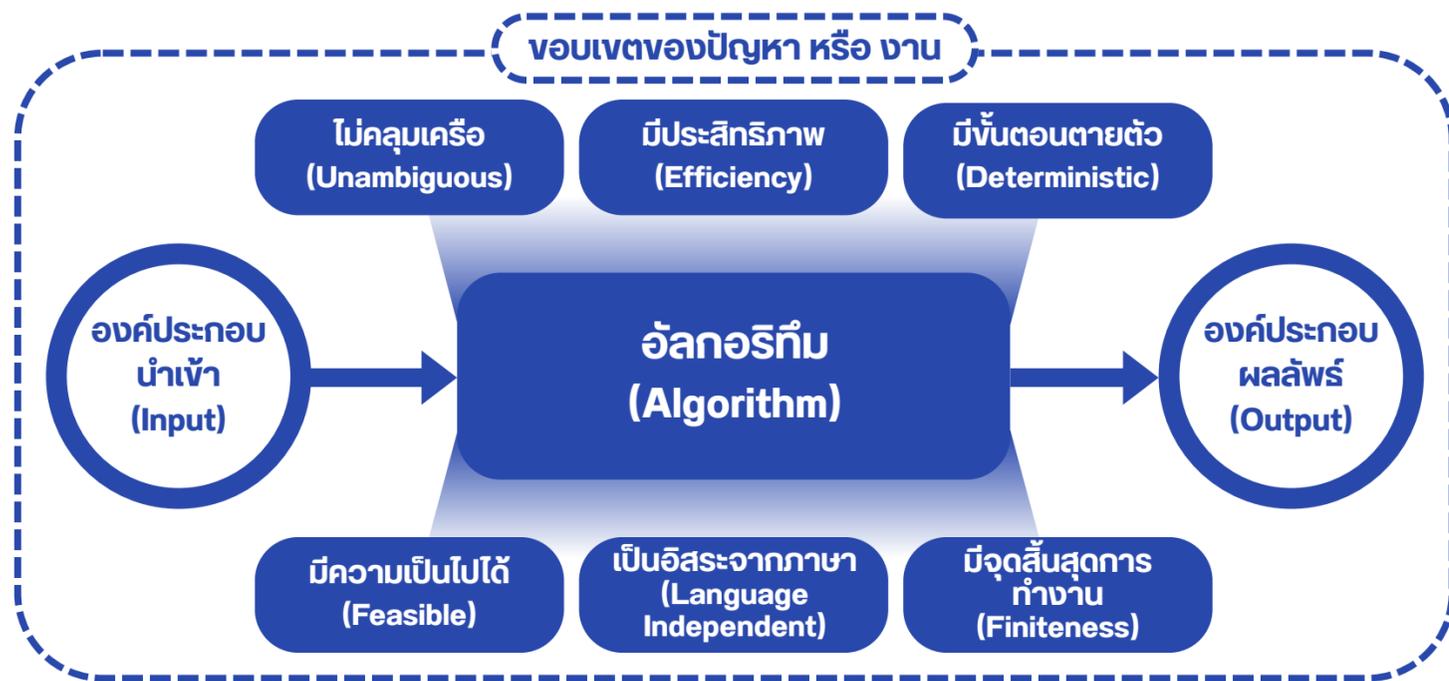
## อ้างอิง

- 1.Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to Algorithms (3rd ed.). MIT Press. ISBN: 978-0262033848.
- 2.Sedgewick, R., & Wayne, K. (2011). Algorithms (4th ed.). Addison-Wesley Professional. ISBN: 978-0321573513.
- 3.Dasgupta, S., Papadimitriou, C. H., & Vazirani, U. V. (2008). Algorithms. McGraw-Hill Science/Engineering/Math. ISBN: 978-0073523408.

# คุณลักษณะของอัลกอริทึม (Characteristics of an Algorithm)



# คุณลักษณะของอัลกอริทึม (Characteristics of an Algorithm)



ลักษณะของอัลกอริทึมที่ดีที่ประกอบด้วยคุณสมบัติ ดังนี้

- 1. ไม่คลุมเครือ (Unambiguous)** อัลกอริทึมต้องไม่มีความกำกวมที่ไม่ชัดเจน และต้องมีชุดคำสั่งที่เข้าใจง่ายและไม่ทำให้เกิดความสับสนในการประมวลผล
- 2. มีประสิทธิภาพ (Efficiency)** อัลกอริทึมควรมีประสิทธิภาพในการทำงาน คือต้องให้ผลลัพธ์ที่ถูกต้อง และใช้เวลาน้อยที่สุดในการแก้ปัญหา

**3. มีขั้นตอนตายตัว (Deterministic)** อัลกอริทึมต้องเป็น Deterministic หมายความว่า จะต้องให้ผลลัพธ์ที่เหมือนกัน สำหรับข้อมูลนำเข้าที่เหมือนกันทุกครั้งที่รับ

**4. มีความเป็นไปได้ (Feasible)** อัลกอริทึมต้องเป็นไปได้ในการนำไปใช้งานจริง สามารถทำงานบนเครื่องคอมพิวเตอร์หรือสภาพแวดล้อมที่เหมาะสม

**5. เป็นอิสระจากภาษา (Language Independent)** อัลกอริทึมควรมิเกี่ยวข้องับภาษาเฉพาะเสียเพียงอย่างเดียว และสามารถนำไปใช้งานในหลายภาษาโปรแกรมได้

**6. มีจุดสิ้นสุดการทำงาน (Finiteness)** อัลกอริทึมต้องสามารถสิ้นสุดการทำงานหลังจากทำขั้นตอนทุกขั้นตอนเสร็จสมบูรณ์ และไม่ต้องเข้าสู่วงจรไม่สิ้นสุด (Infinity Loop)

## อ้างอิง

- Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to Algorithms (3rd ed.). MIT Press. ISBN: 978-0262033848.
- Sedgewick, R., & Wayne, K. (2011). Algorithms (4th ed.). Addison-Wesley Professional. ISBN: 978-0321573513.

# อัลกอริทึม (Algorithm)

การแสดงผลอัลกอริทึมด้วยผังงาน หรือ โฟลวชาร์ต (Flowchart)

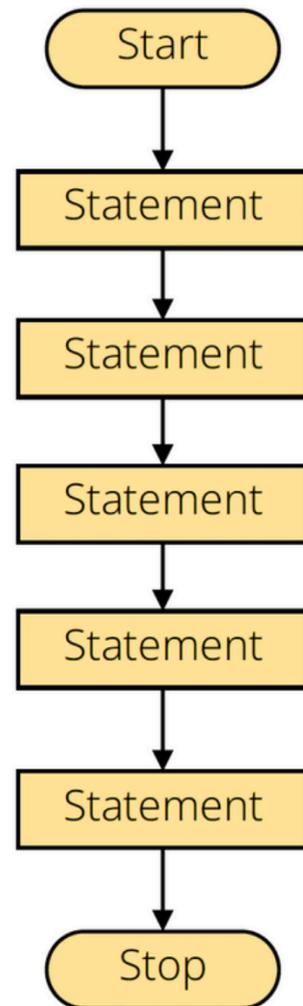


เมื่อรวมกันแล้ว จึงหมายถึง แผนผังที่แสดงการไหลของงาน หรือ แผนผังที่แสดงขั้นตอนการทำงานอย่างชัดเจน นั่นเอง ซึ่งสามารถนำผังงานมาใช้แสดงขั้นตอนการทำงานของโปรแกรมได้

# โครงสร้างการควบคุมทิศทางอัลกอริทึม (Control Flow in Algorithm)

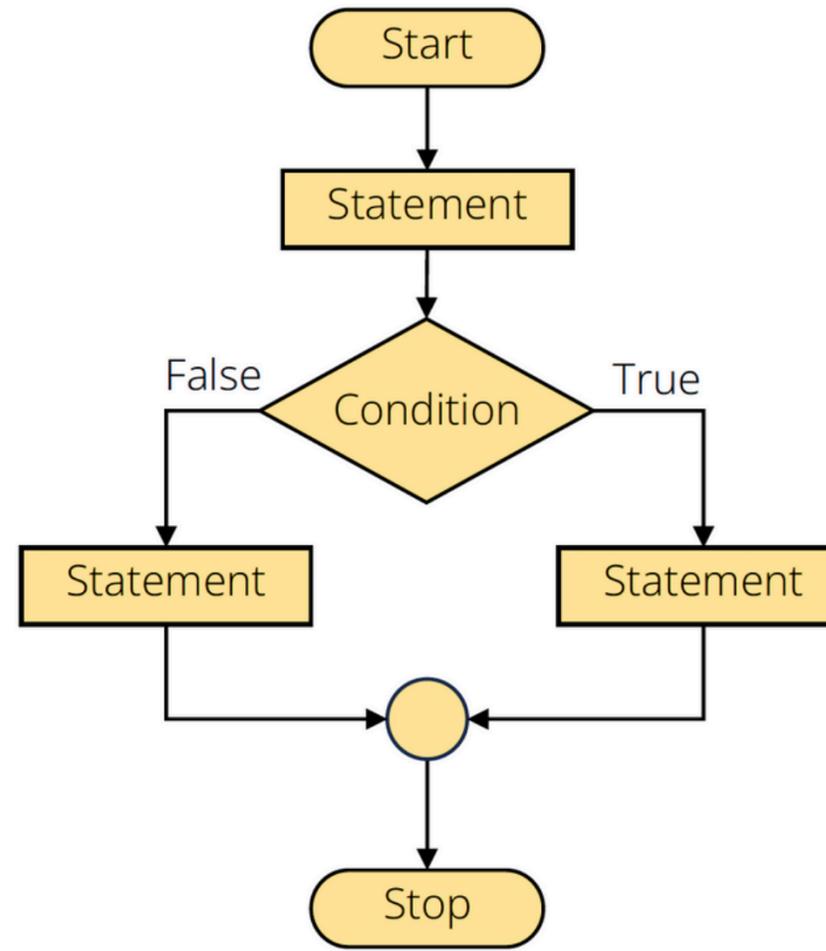
## โครงสร้างการควบคุมแบบลำดับ Sequential Control Structure

โครงสร้างการควบคุมแบบบนลงล่าง  
Top-Down Control Structure



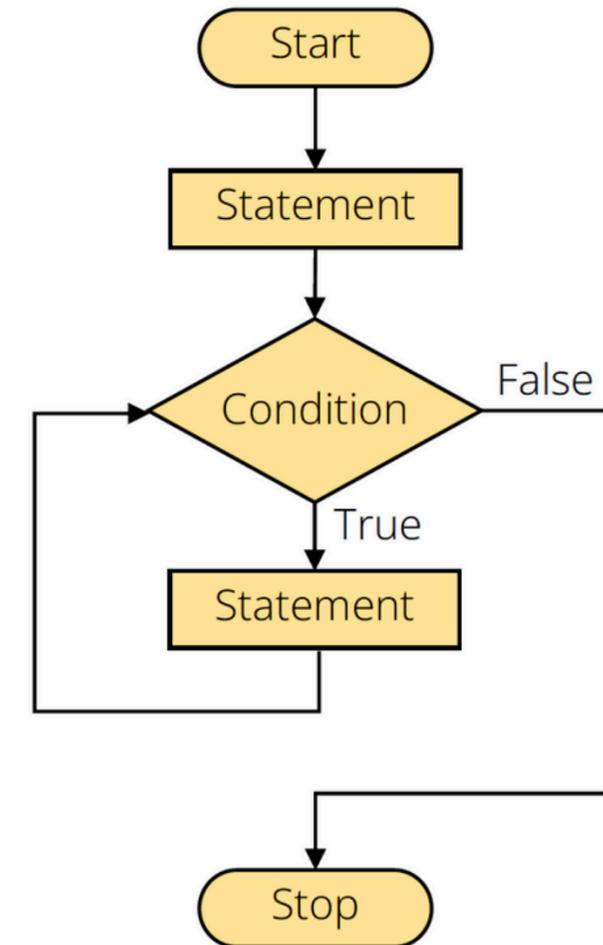
## โครงสร้างการควบคุมแบบทางเลือก Selection Control Structure

โครงสร้างการควบคุมแบบเงื่อนไข  
Conditional Control Structure  
โครงสร้างการควบคุมแบบตัดสินใจ  
Decision Control Structure



## โครงสร้างการควบคุมแบบทำซ้ำ Iteration Control Structure

โครงสร้างการควบคุมแบบวนซ้ำ  
Looping Control Structure



# โครงสร้างการควบคุมแบบทำซ้ำ (Iteration Control Structure)

โครงสร้างการควบคุมแบบทำซ้ำ (Iteration Control Structure) มีประโยชน์และความสำคัญในการออกแบบและเขียนโปรแกรมคอมพิวเตอร์เป็นอย่างมาก ดังนี้

- ช่วยให้การเขียนโปรแกรมง่ายขึ้น โครงสร้างการควบคุมแบบทำซ้ำช่วยให้โปรแกรมเมอร์ไม่ต้องเขียนคำสั่งซ้ำ ๆ กันหลายครั้ง ทำให้โปรแกรมมีความกระชับ เข้าใจง่าย และตรวจสอบความผิดพลาดได้ง่าย
- ช่วยให้โปรแกรมมีประสิทธิภาพมากขึ้น โครงสร้างการควบคุมแบบทำซ้ำช่วยให้โปรแกรมสามารถทำงานซ้ำ ๆ ได้อย่างรวดเร็วและมีประสิทธิภาพ โดยไม่จำเป็นต้องเขียนคำสั่งเดิมหลายครั้ง
- ช่วยให้โปรแกรมมีความยืดหยุ่นมากขึ้น โครงสร้างการควบคุมแบบทำซ้ำช่วยให้โปรแกรมเมอร์สามารถกำหนดเงื่อนไขในการวนซ้ำได้ เพื่อตรวจสอบข้อมูลหรือแก้ปัญหาเฉพาะได้

แสดงผลจำนวน 1 ถึง 10,000

จำนวน 1 ถึง 10,000  
มีจำนวนใดบ้างหาร 2 ลงตัว?

จำนวน 1 ถึง 10,000  
มีจำนวนใดบ้างหาร 2 และ 4 ลงตัว?

แสดงรายการเบอร์โทรศัพท์ที่ขึ้นต้นด้วย 086  
ทั้งหมดในประเทศไทย

มีนักเรียนกี่คนในประเทศไทยที่ผู้ปกครองมี  
รายได้ต่ำกว่า 180,000 บาทต่อปี

# โครงสร้างการควบคุมแบบทำซ้ำ (Iteration Control Structure)

โครงสร้างการควบคุมแบบทำซ้ำมี 3 รูปแบบหลัก ดังนี้

## 1. โครงสร้างแบบวนซ้ำแบบ while

โครงสร้างนี้ใช้ตรวจสอบเงื่อนไขในการวนซ้ำก่อนแต่ละรอบ หากเงื่อนไขเป็นจริง โปรแกรมจะวนซ้ำต่อไป หากเงื่อนไขเป็นเท็จ โปรแกรมจะออกจากการวนซ้ำ

## 2. โครงสร้างแบบวนซ้ำแบบ do-while

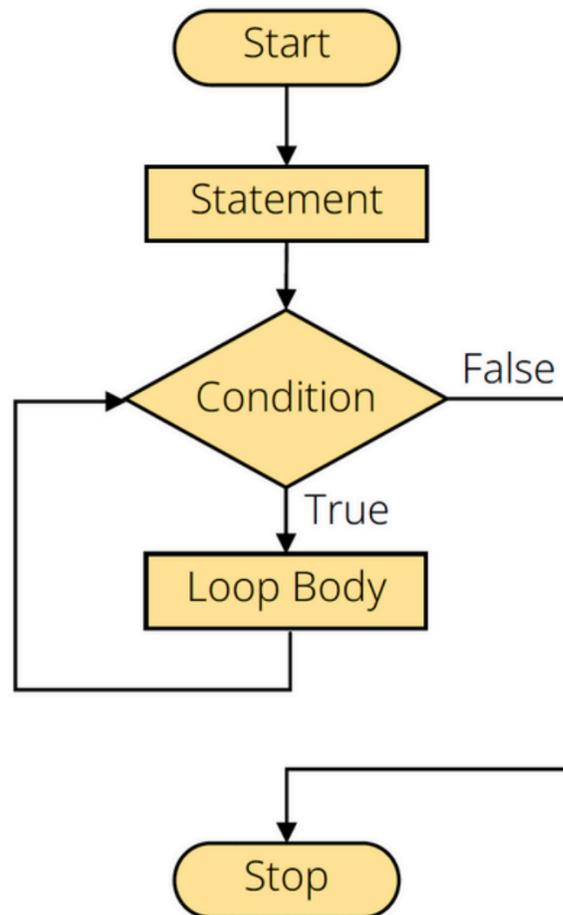
โครงสร้างนี้คล้ายกับโครงสร้างแบบ while แต่โปรแกรมจะวนซ้ำก่อนตรวจสอบเงื่อนไข หากเงื่อนไขเป็นเท็จ โปรแกรมจะวนซ้ำต่อไป หากเงื่อนไขเป็นจริง โปรแกรมจะออกจากการวนซ้ำ

## 3. โครงสร้างแบบวนซ้ำแบบ for

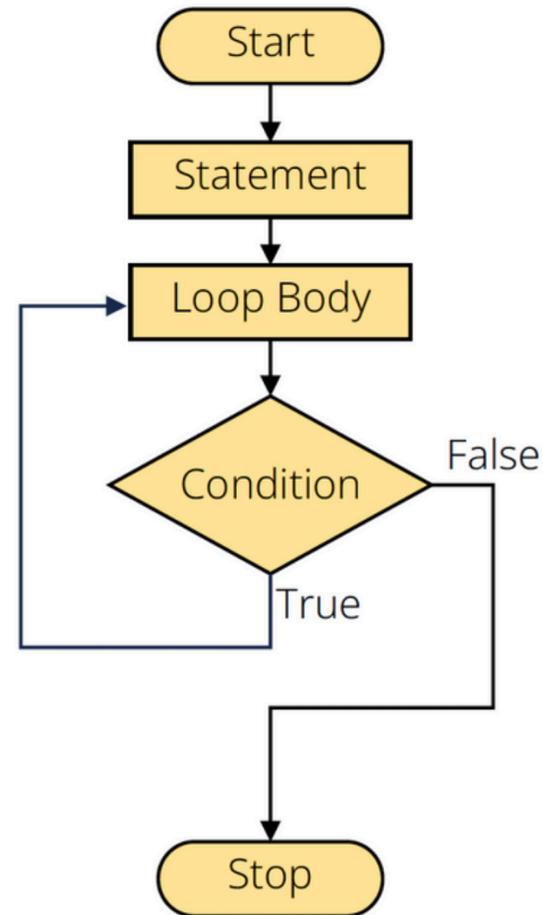
โครงสร้างนี้ใช้กำหนดจำนวนรอบในการวนซ้ำไว้ล่วงหน้า โดยระบุจำนวนรอบในตัวแปร

# โครงสร้างการควบคุมแบบทำซ้ำ (Iteration Control Structure)

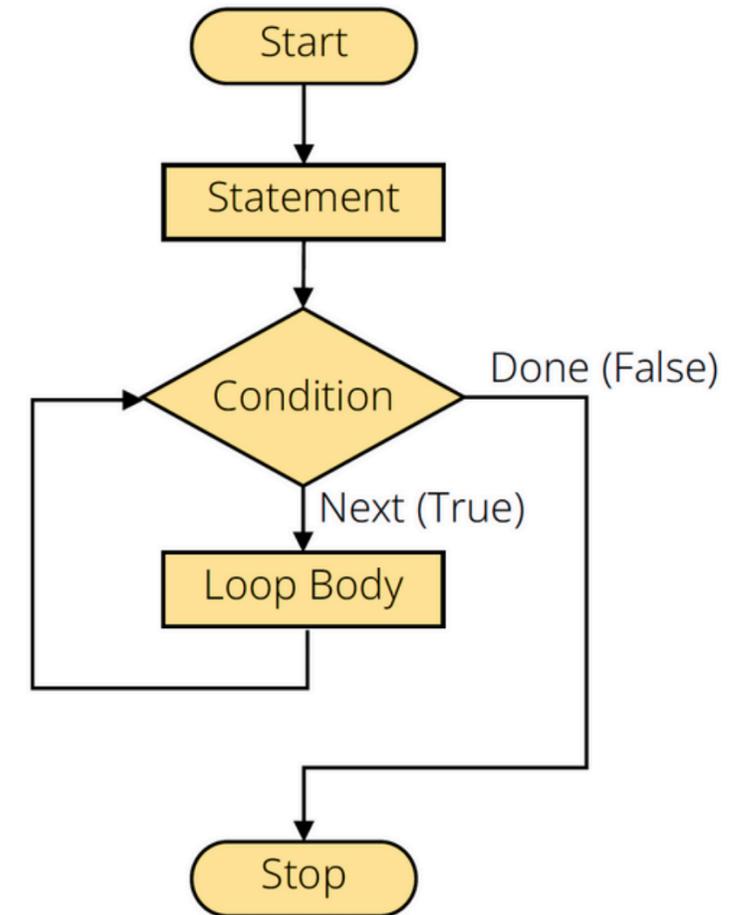
โครงสร้างการควบคุมทำซ้ำแบบ while  
While Statement



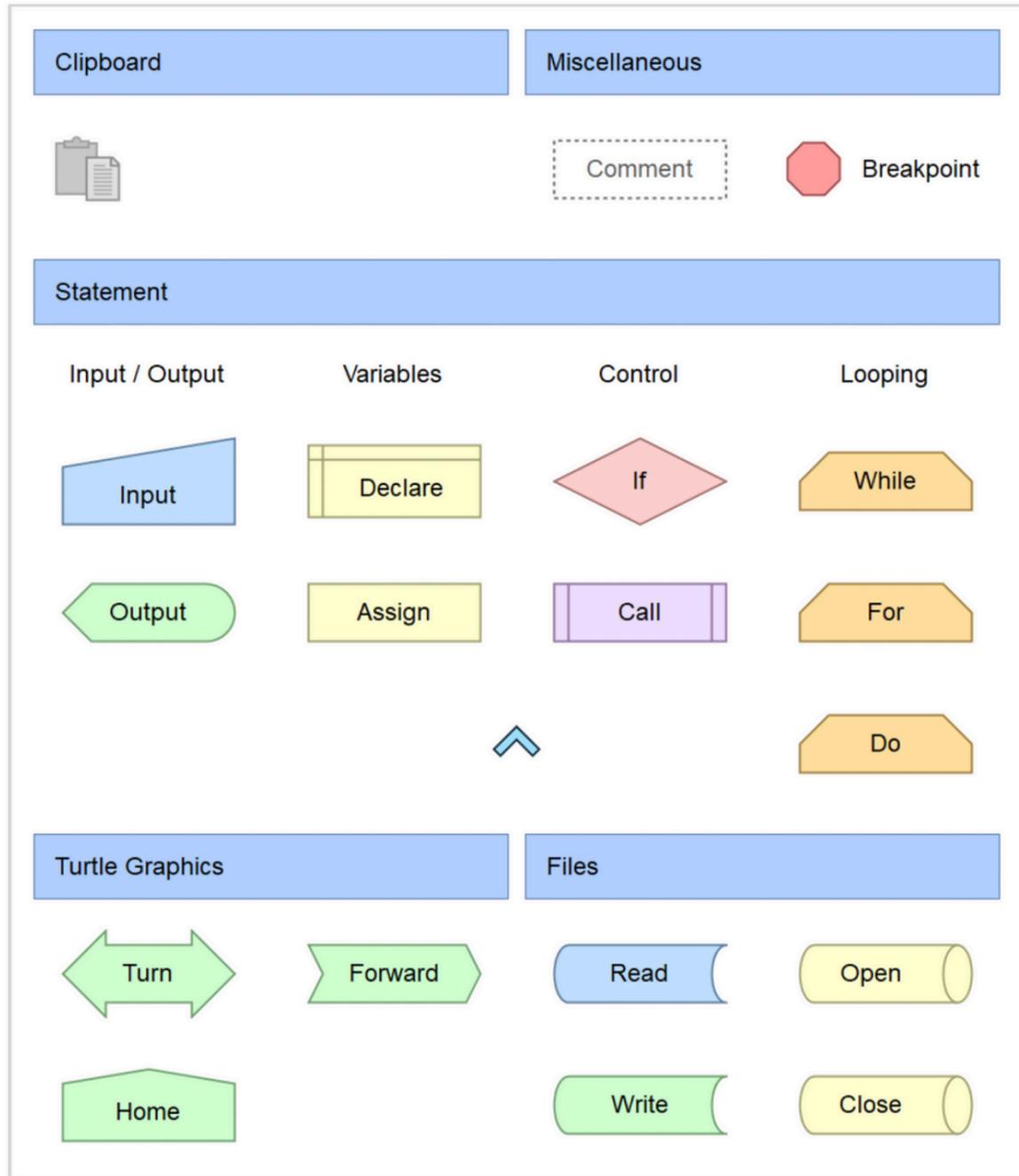
โครงสร้างการควบคุมทำซ้ำแบบ do-while  
Do-while Statement



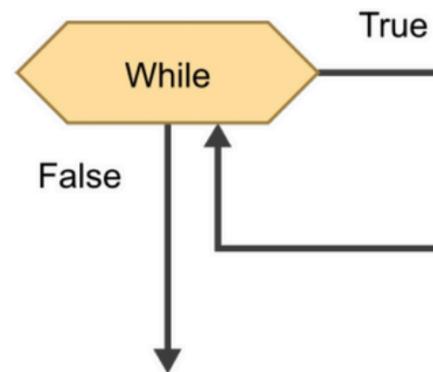
โครงสร้างการควบคุมทำซ้ำแบบ for  
For Statement



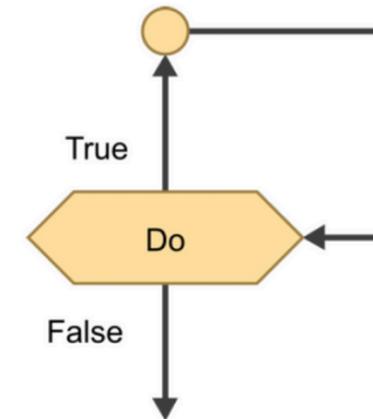
# โครงสร้างการทำซ้ำใน Flowgorithm



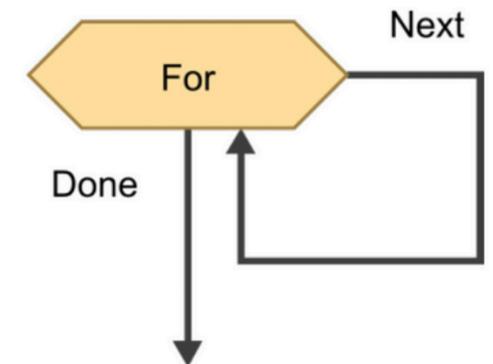
โครงสร้างการควบคุมทำซ้ำ  
แบบ while  
While Statement



โครงสร้างการควบคุมทำซ้ำ  
แบบ do-while  
Do-while Statement



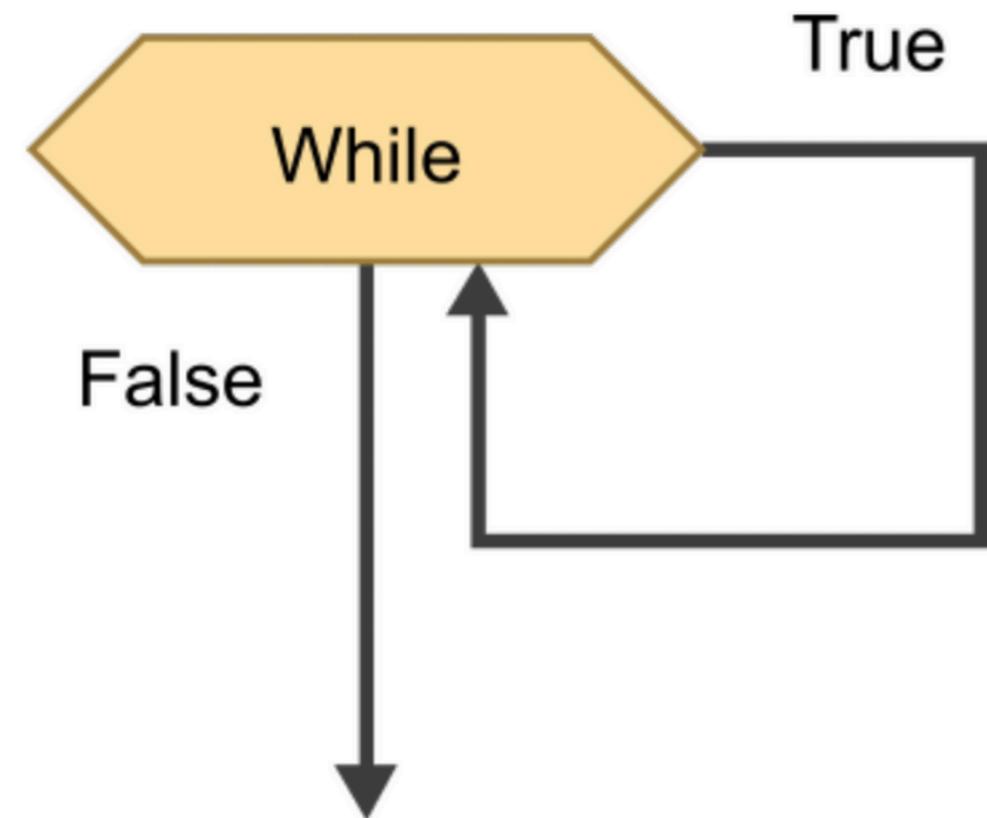
โครงสร้างการควบคุมทำซ้ำ  
แบบ for  
For Statement



# โครงสร้างการทำซ้ำแบบ while

A While Loop evaluates a Boolean expression and then, **if true, executes a block of statements.** After the statements are executed, the While Statement rechecks the expression. **When the expression is false, the loop ends.**

การทำซ้ำแบบ while จะใช้บูลีน (ค่า True/False) ในการประเมินว่าจะวนซ้ำอีกหรือไม่ ถ้าตรวจสอบเงื่อนไขแล้วยังเป็นจริง (True) จะวนซ้ำ แต่ถ้าไม่จริง (False) จะทำการหลุดออกจากการวนซ้ำทันที



# โครงสร้างการทำซ้ำแบบ while

## สถานการณ์ปัญหา:

ต้องการแสดงค่า 1 ถึง 10 ออกทางหน้าจอด้วยการวนซ้ำแบบ while

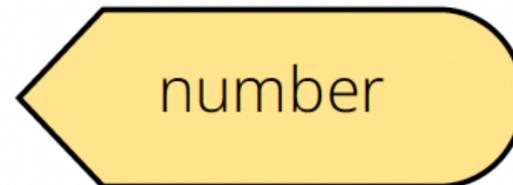
## วิเคราะห์ปัจจัยนำเข้า (Input):

ตัวแปรที่ช่วยเก็บค่าตั้งแต่ 1 ถึง 10



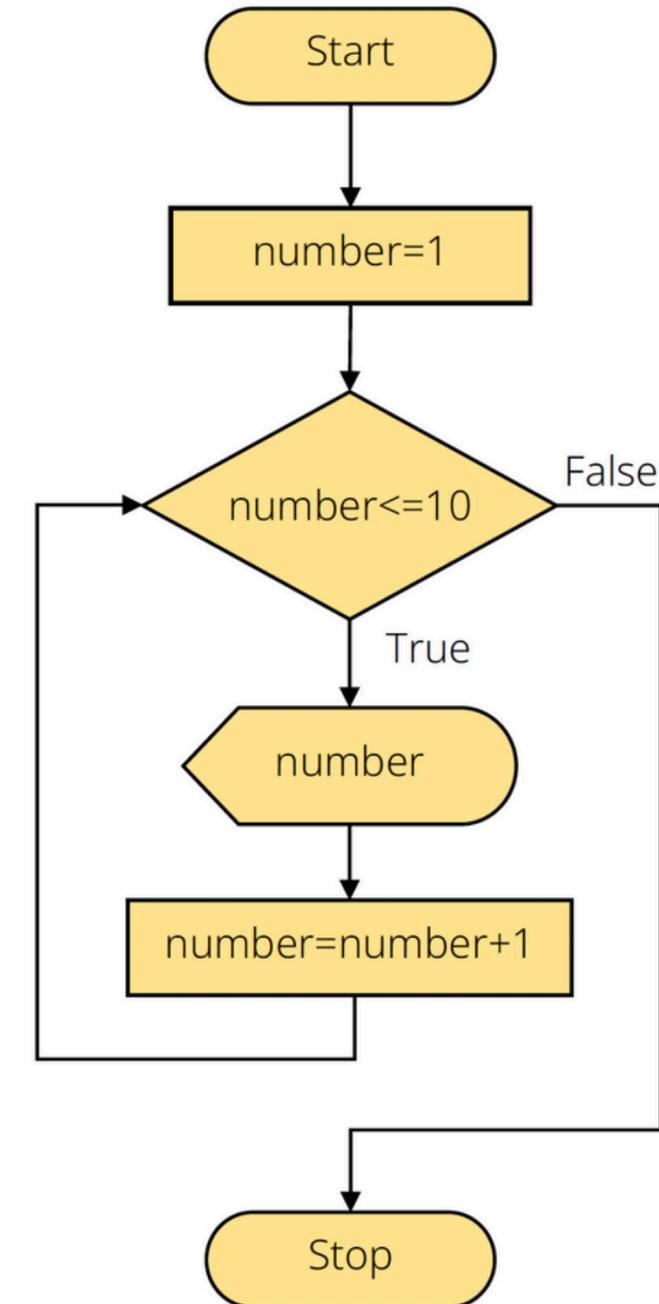
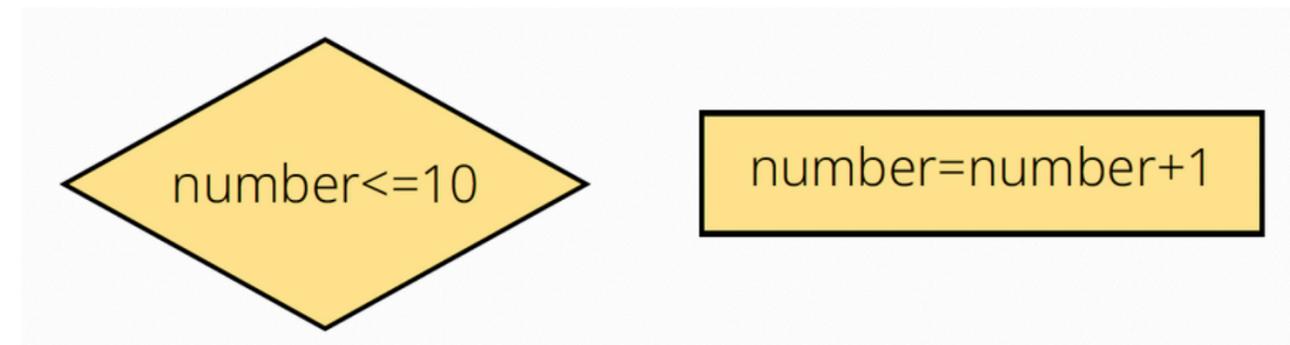
## วิเคราะห์ปัจจัยผลลัพธ์ (Output):

แสดงค่าในตัวแปร



## วิเคราะห์ปัจจัยประมวลผล (Process):

ทำการเพิ่มจำนวนในตัวแปรรอบละ +1



# โครงสร้างการทำซ้ำแบบ while - Code #8

## สถานการณ์ปัญหา:

ต้องการแสดงค่า 1 ถึง 10 ออกทางหน้าจอด้วยการวนซ้ำแบบ while

## วิเคราะห์ปัจจัยนำเข้า (Input):

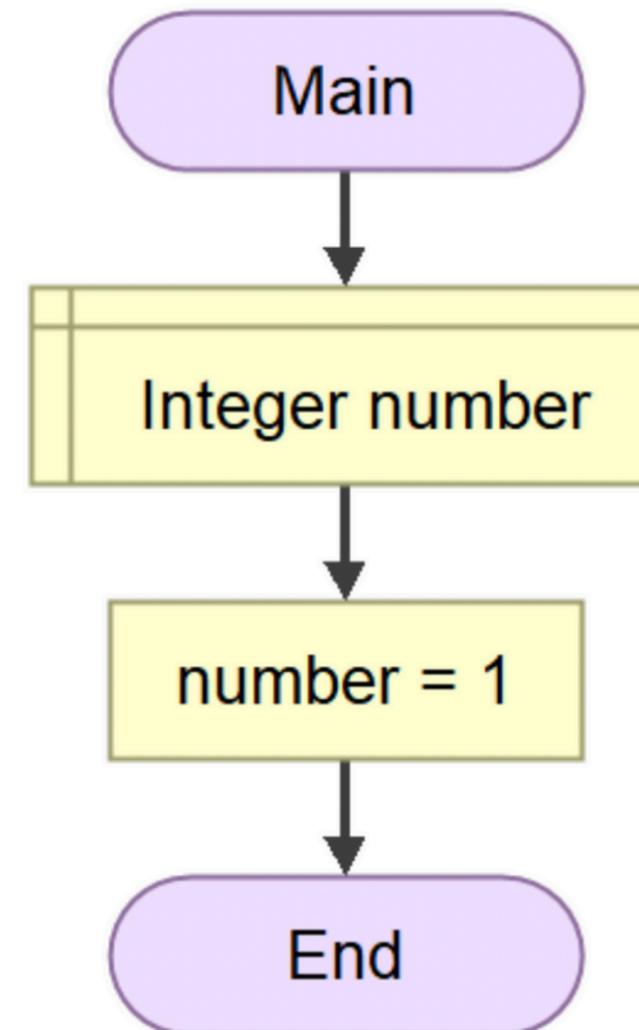
ตัวแปรที่ช่วยเก็บค่าตั้งแต่ 1 ถึง 10

## วิเคราะห์ปัจจัยผลลัพธ์ (Output):

แสดงค่าในตัวแปร

## วิเคราะห์ปัจจัยประมวลผล (Process):

ทำการเพิ่มจำนวนในตัวแปรรอบละ +1



# โครงสร้างการทำซ้ำแบบ while - Code #8

## สถานการณ์ปัญหา:

ต้องการแสดงค่า 1 ถึง 10 ออกทางหน้าจอด้วยการวนซ้ำแบบ while

## วิเคราะห์ปัจจัยนำเข้า (Input):

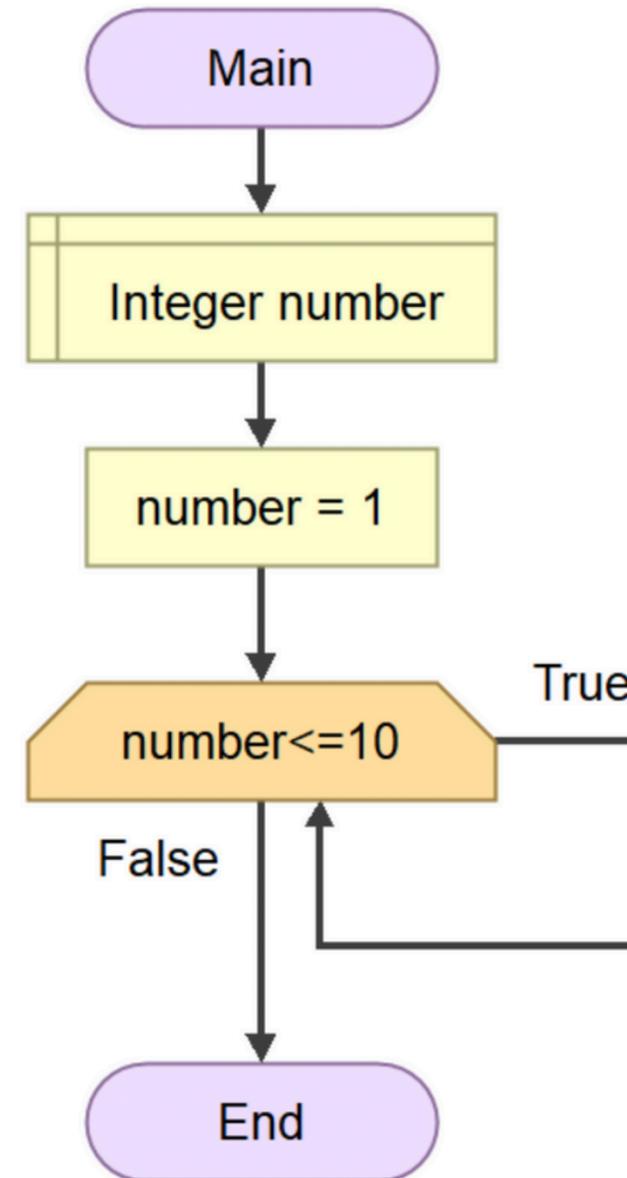
ตัวแปรที่ช่วยเก็บค่าตั้งแต่ 1 ถึง 10

## วิเคราะห์ปัจจัยผลลัพธ์ (Output):

แสดงค่าในตัวแปร

## วิเคราะห์ปัจจัยประมวลผล (Process):

ทำการเพิ่มจำนวนในตัวแปรรอบละ +1



# โครงสร้างการทำซ้ำแบบ while - Code #8

## สถานการณ์ปัญหา:

ต้องการแสดงค่า 1 ถึง 10 ออกทางหน้าจอด้วยการวนซ้ำแบบ while

## วิเคราะห์ปัจจัยนำเข้า (Input):

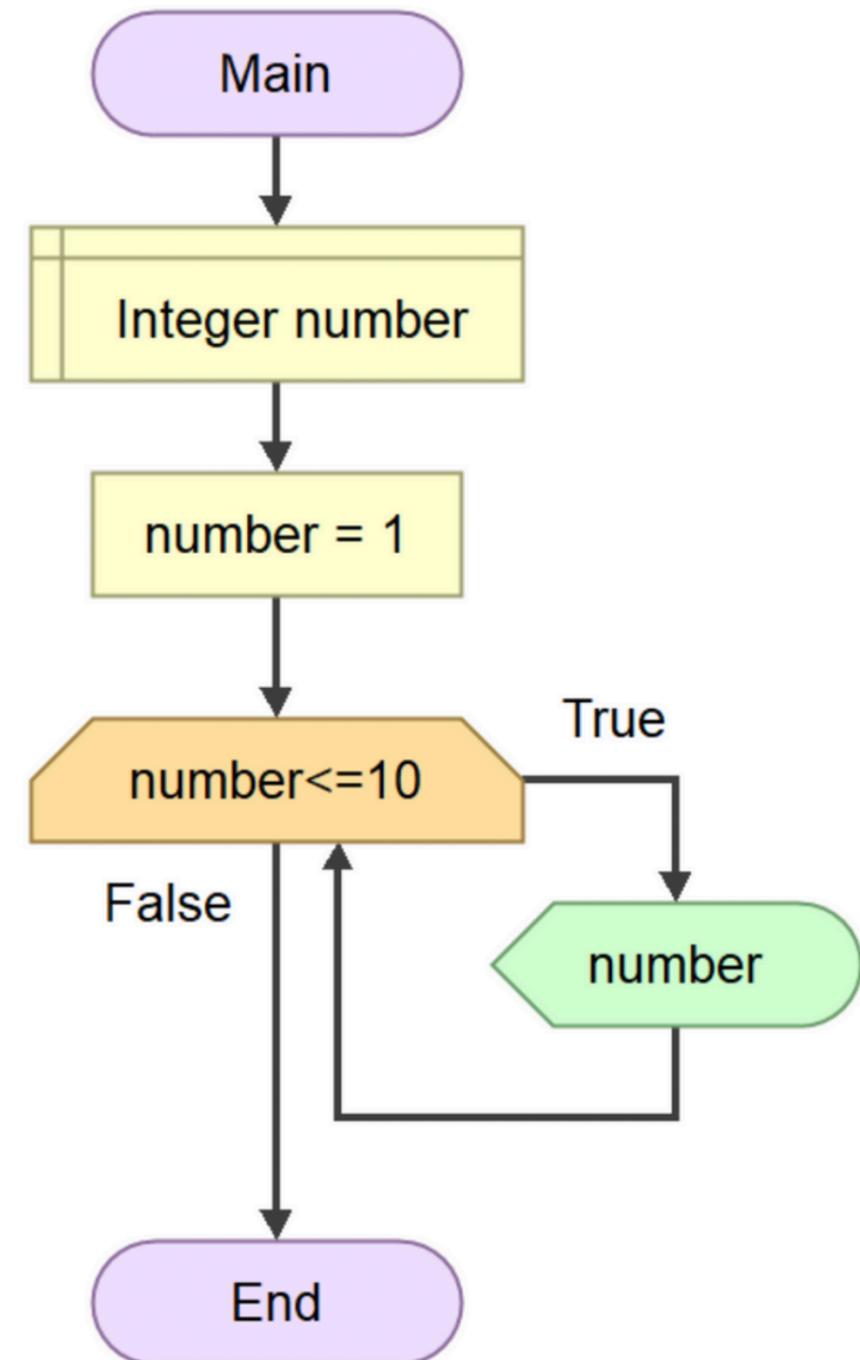
ตัวแปรที่ช่วยเก็บค่าตั้งแต่ 1 ถึง 10

## วิเคราะห์ปัจจัยผลลัพธ์ (Output):

แสดงค่าในตัวแปร

## วิเคราะห์ปัจจัยประมวลผล (Process):

ทำการเพิ่มจำนวนในตัวแปรรอบละ +1



# โครงสร้างการทำซ้ำแบบ while - Code #8

## สถานการณ์ปัญหา:

ต้องการแสดงค่า 1 ถึง 10 ออกทางหน้าจอด้วยการวนซ้ำแบบ while

## วิเคราะห์ปัจจัยนำเข้า (Input):

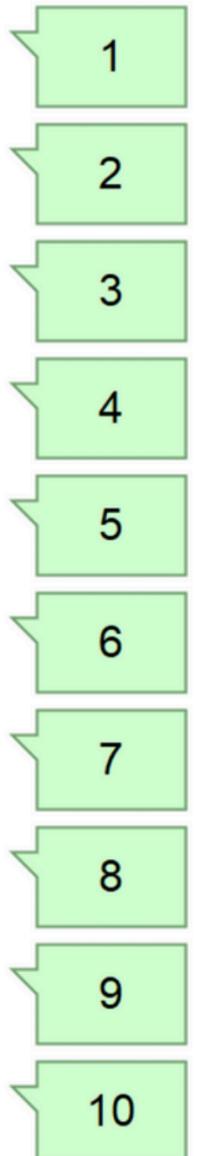
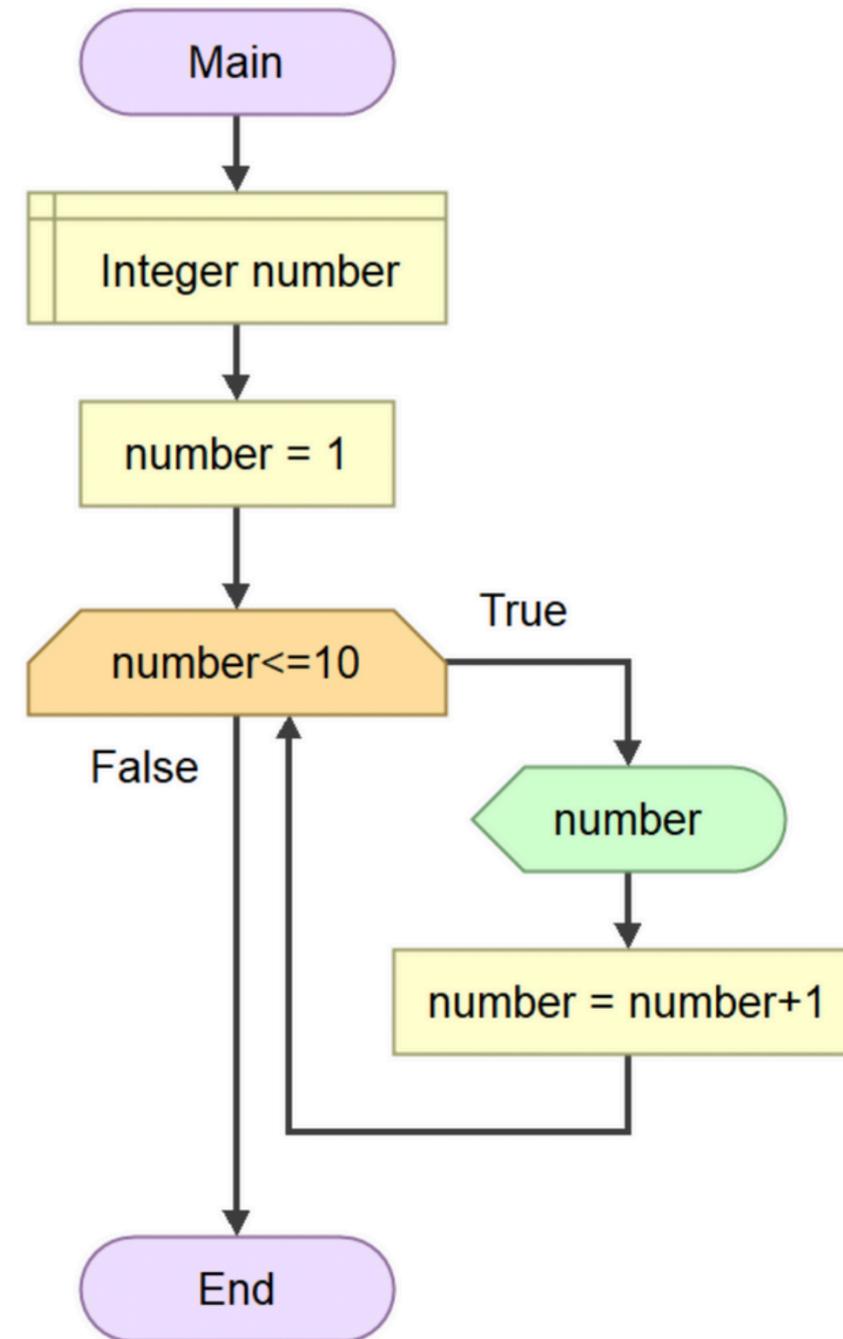
ตัวแปรที่ช่วยเก็บค่าตั้งแต่ 1 ถึง 10

## วิเคราะห์ปัจจัยผลลัพธ์ (Output):

แสดงค่าในตัวแปร

## วิเคราะห์ปัจจัยประมวลผล (Process):

ทำการเพิ่มจำนวนในตัวแปรรอบละ +1

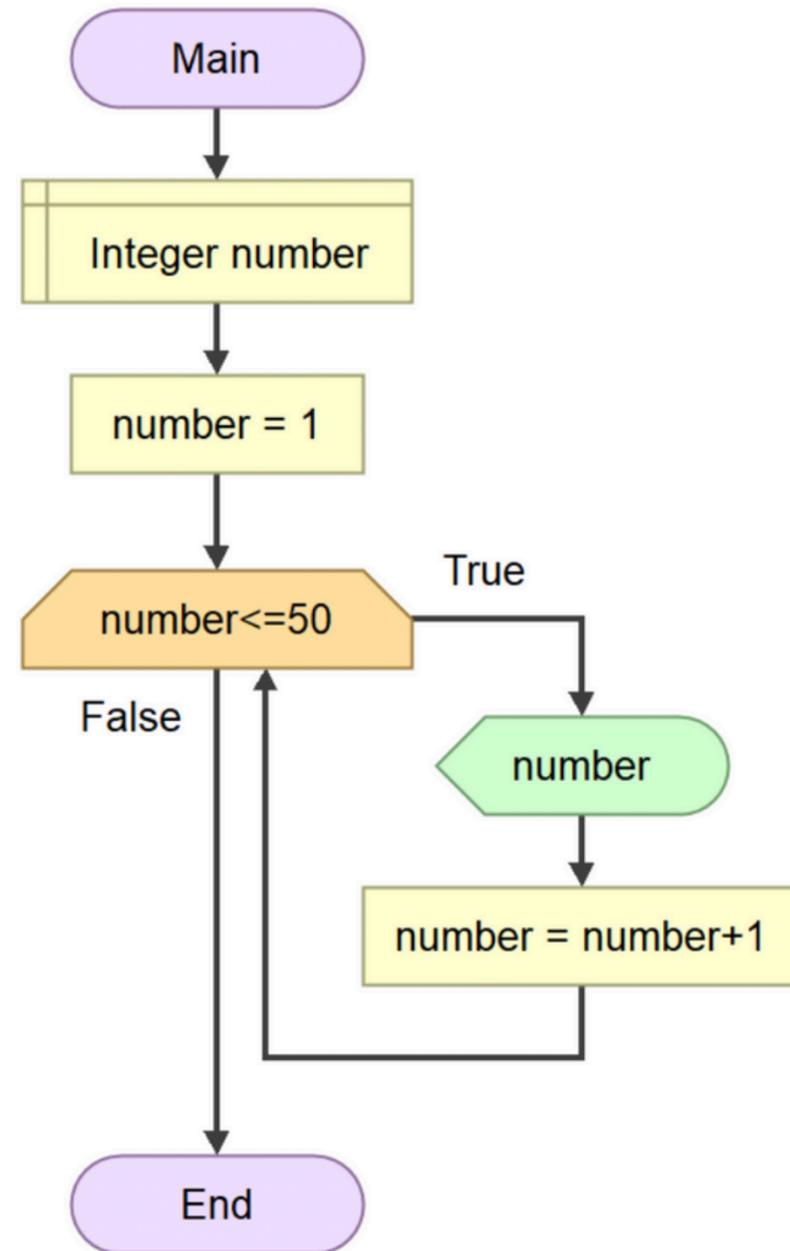


## โครงสร้างการทำซ้ำแบบ while - Code #9

สถานการณ์ปัญหา: แสดงจำนวนตั้งแต่ 1 ไปเรื่อยๆ แต่ห้ามเกิน 50 โดยใช้ while loop

# โครงสร้างการทำซ้ำแบบ while - Code #9

สถานการณ์ปัญหา: แสดงจำนวนตั้งแต่ 1 ไปเรื่อยๆ แต่ห้ามเกิน 50 โดยใช้ while loop



39

40

41

42

43

44

45

46

47

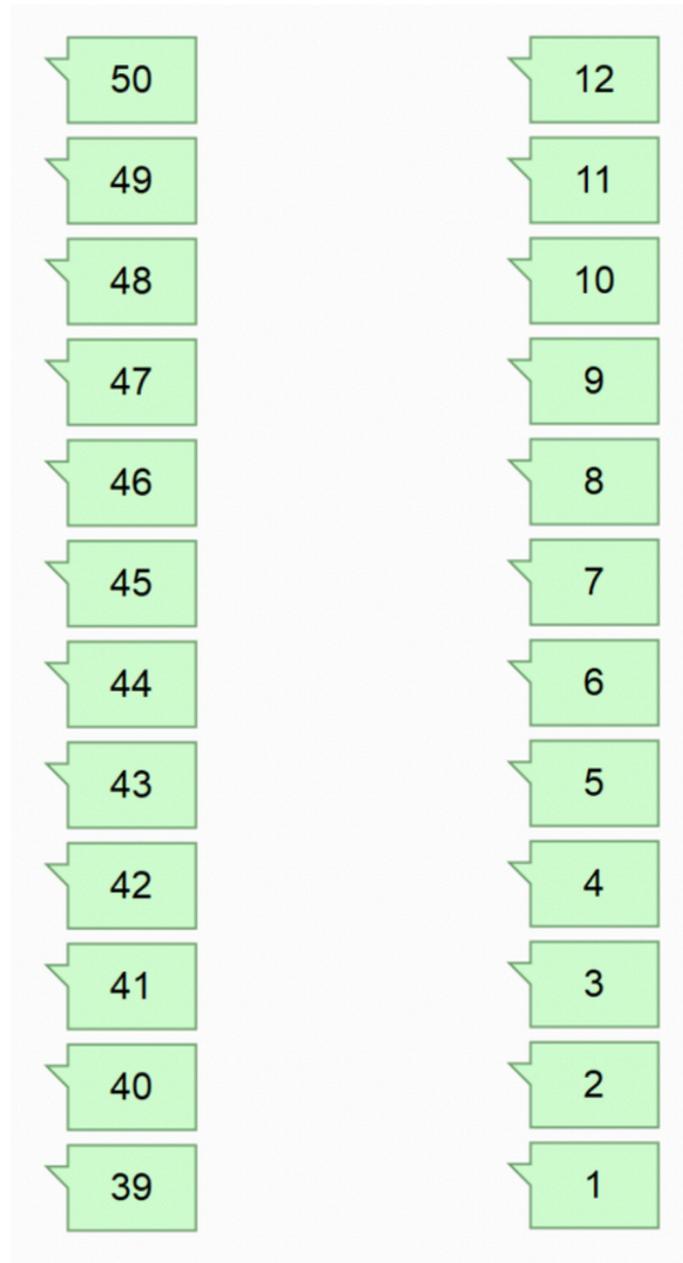
48

49

50

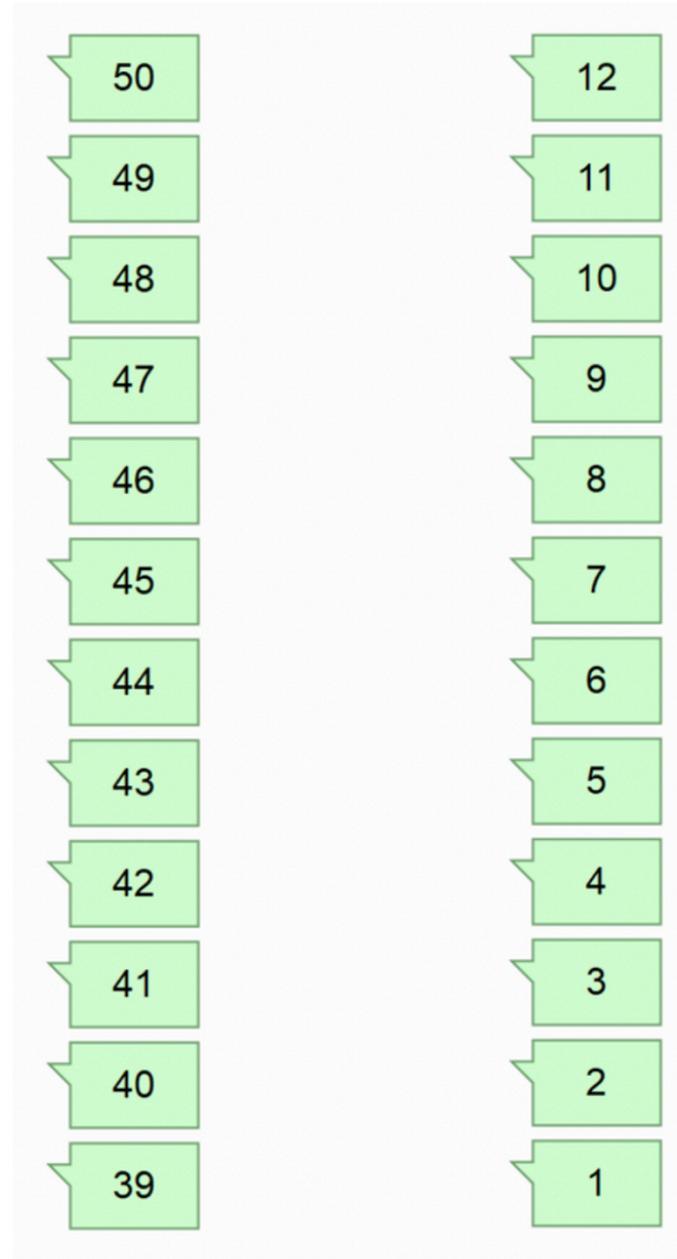
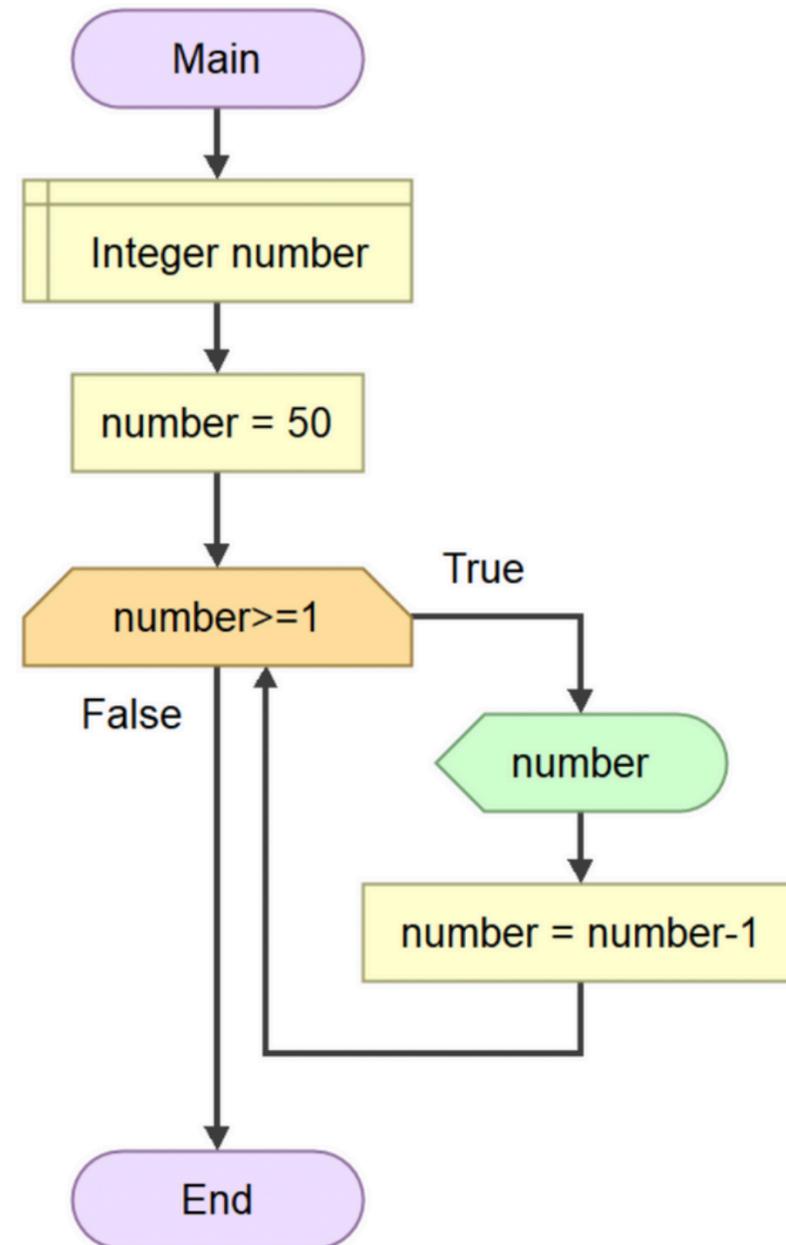
# โครงสร้างการทำซ้ำแบบ while - Code #10

สถานการณ์ปัญหา: แสดงจำนวนตั้งแต่ 50, 49, 48 ไปเรื่อยๆ แต่ห้ามเกิน 1 โดยใช้ while loop



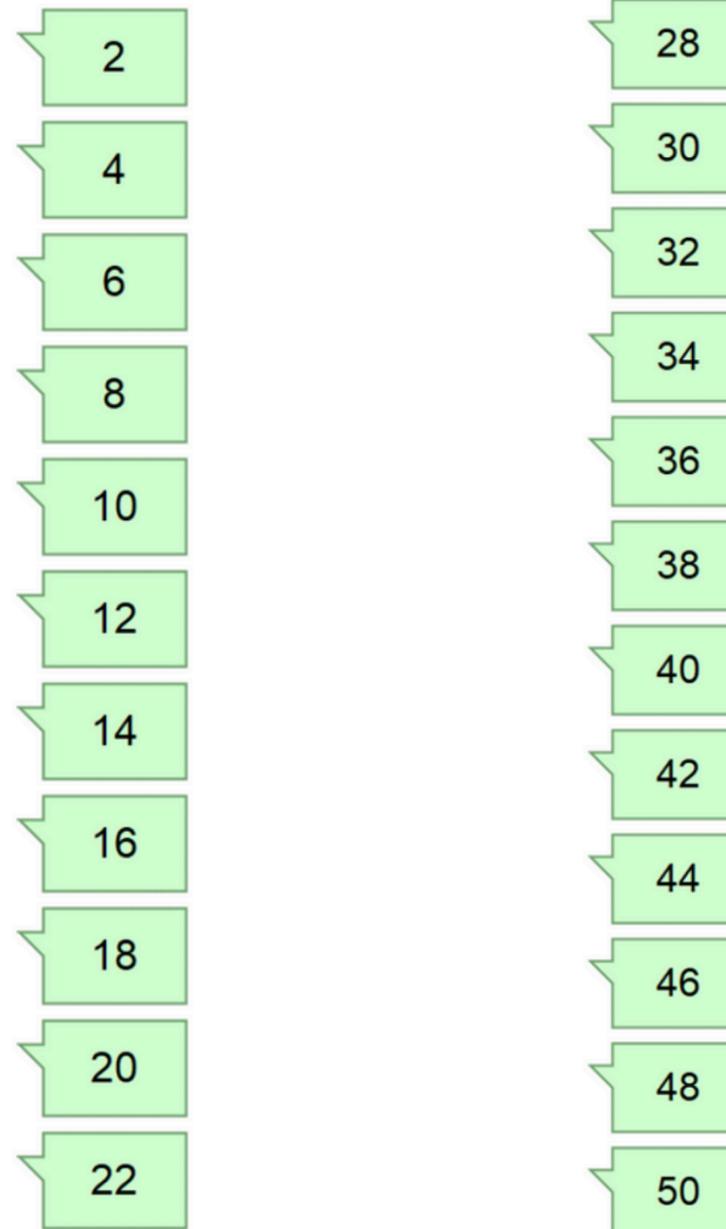
# โครงสร้างการทำซ้ำแบบ while - Code #10

สถานการณ์ปัญหา: แสดงจำนวนตั้งแต่ 50, 49, 48 ไปเรื่อยๆ แต่ห้ามเกิน 1 โดยใช้ while loop



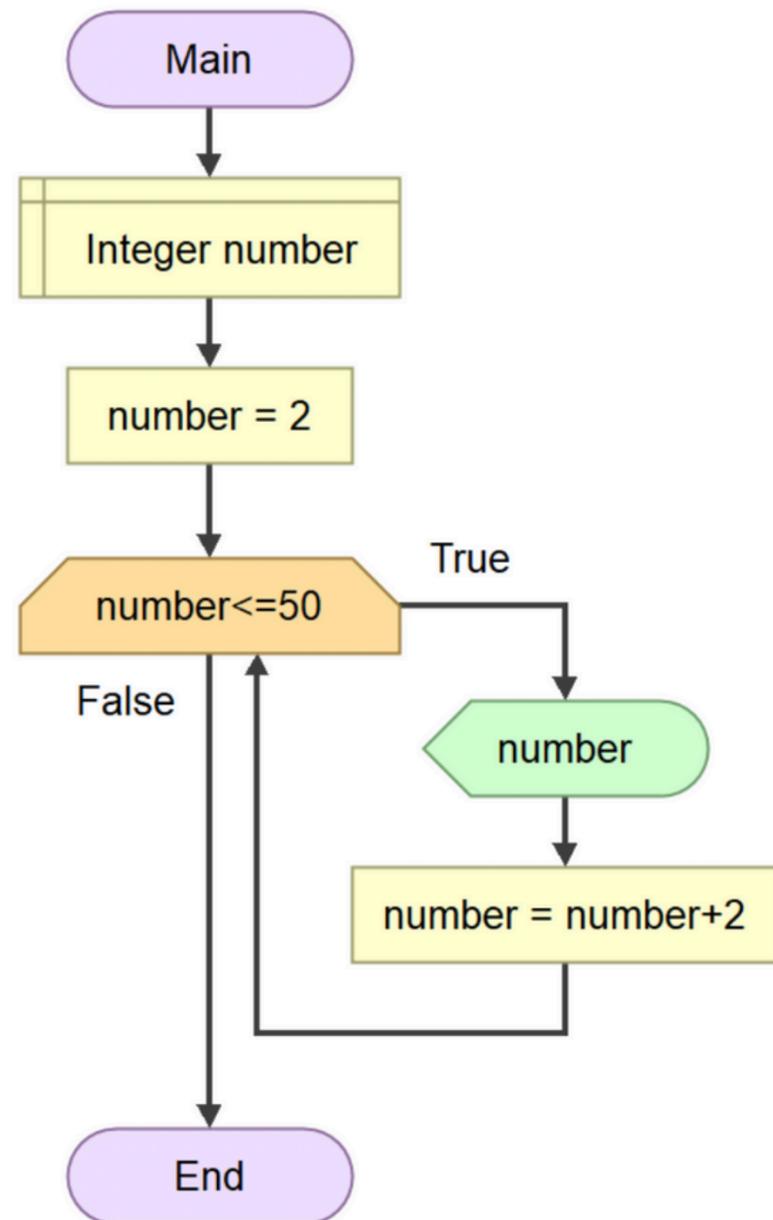
# โครงสร้างการทำซ้ำแบบ while - Code #11

สถานการณ์ปัญหา: แสดงจำนวนคู่ตั้งแต่ 2,4,6,..., ไปเรื่อยๆ แต่ห้ามเกิน 50 โดยใช้ while loop



# โครงสร้างการทำซ้ำแบบ while - Code #11

สถานการณ์ปัญหา: แสดงจำนวนคู่ตั้งแต่ 2,4,6,..., ไปเรื่อยๆ แต่ห้ามเกิน 50 โดยใช้ while loop

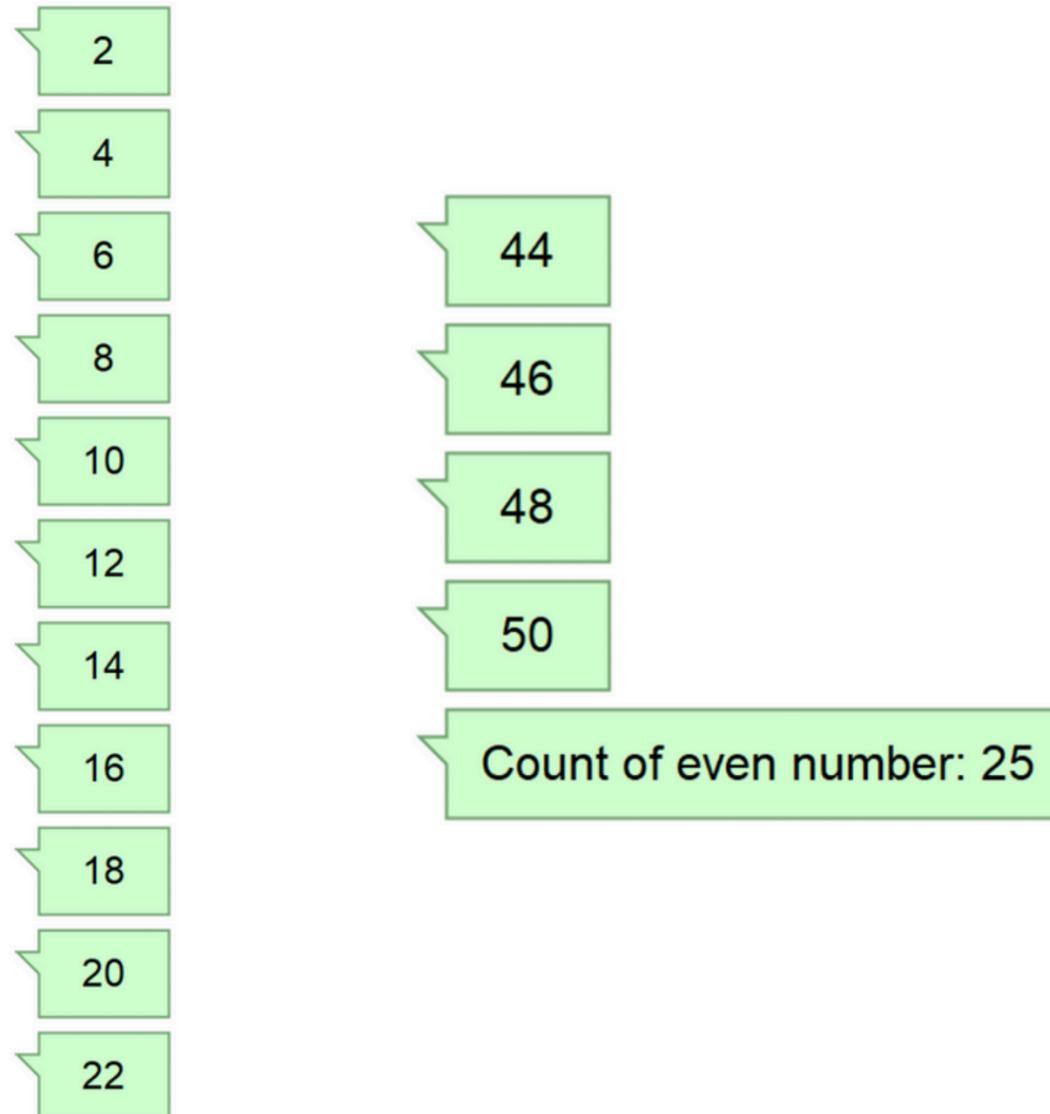


2  
4  
6  
8  
10  
12  
14  
16  
18  
20  
22

28  
30  
32  
34  
36  
38  
40  
42  
44  
46  
48  
50

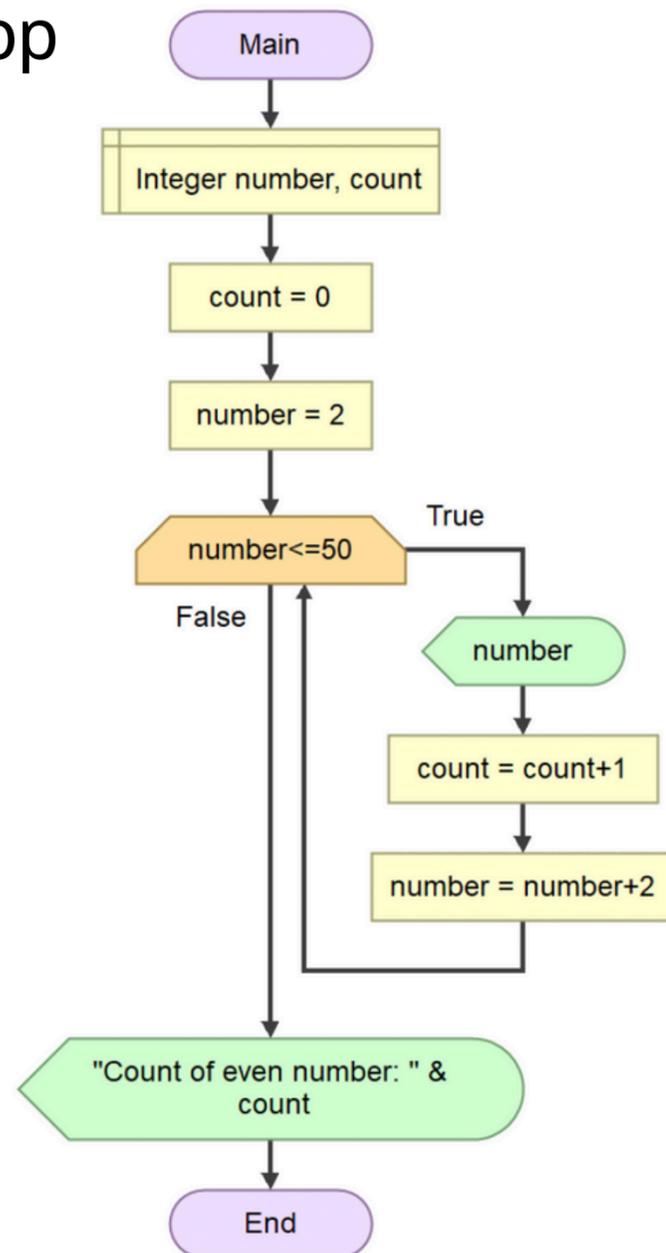
## โครงสร้างการทำซ้ำแบบ while - Code #12

**สถานการณ์ปัญหา:** แสดงจำนวนคู่ตั้งแต่ 2,4,6,..., ไปเรื่อยๆ แต่ห้ามเกิน 50 และสรุปว่ามีทั้งหมดกี่จำนวน โดยใช้ while loop



# โครงสร้างการทำซ้ำแบบ while - Code #12

**สถานการณ์ปัญหา:** แสดงจำนวนคู่ตั้งแต่ 2,4,6,..., ไปเรื่อยๆ แต่ห้ามเกิน 50 และสรุปว่ามีทั้งหมดกี่จำนวน โดยใช้ while loop



- 2
- 4
- 6
- 8
- 10
- 12
- 14
- 16
- 18
- 20
- 22

- 44
- 46
- 48
- 50
- Count of even number: 25

# โครงสร้างการทำซ้ำแบบ do-while - Code #13

**สถานการณ์ปัญหา:** ให้แสดงคำถาม “How many letters are there in the alphabet?” เพื่อให้ผู้ใช้ทายและป้อนจำนวน หากผู้ใช้ตอบไม่ตรงคำตอบจะทำการวนซ้ำให้ตอบใหม่ไปเรื่อยๆ จนกว่าจะตอบถูก

How many letters are there in the alphabet?

24

22

34

32

20

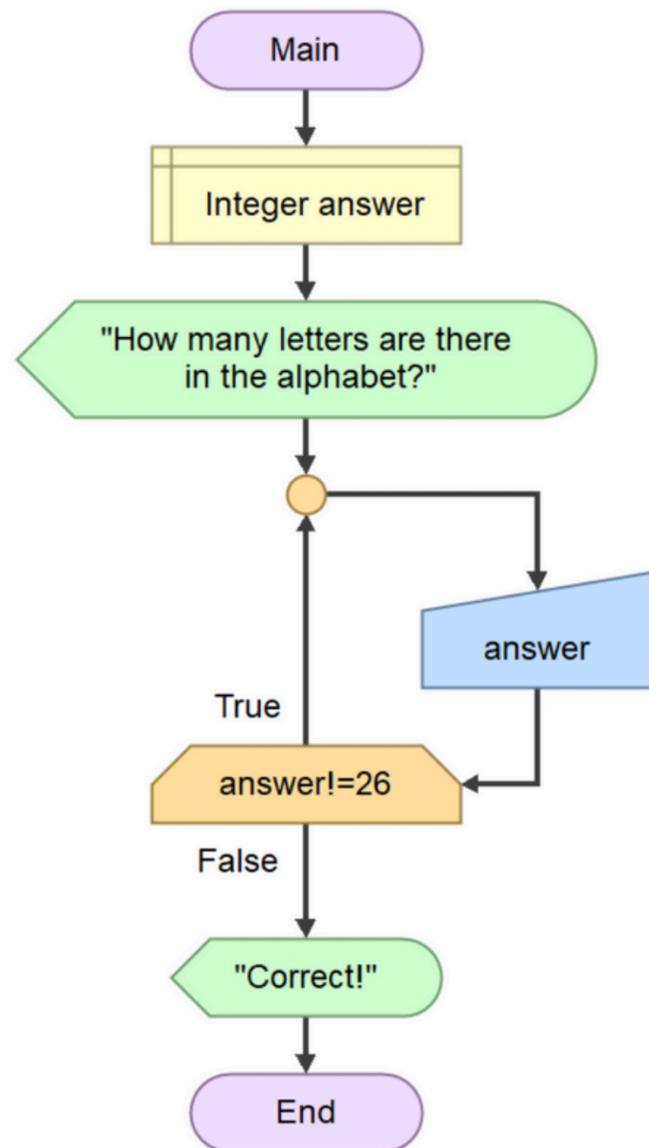
21

26

Correct!

# โครงสร้างการทำซ้ำแบบ do-while - Code #13

**สถานการณ์ปัญหา:** ให้แสดงคำถาม “How many letters are there in the alphabet?” เพื่อให้ผู้ใช้ทาย และป้อนจำนวน หากผู้ใช้ตอบไม่ตรงคำตอบจะทำการวนซ้ำให้ตอบใหม่ไปเรื่อยๆ จนกว่าจะตอบถูก



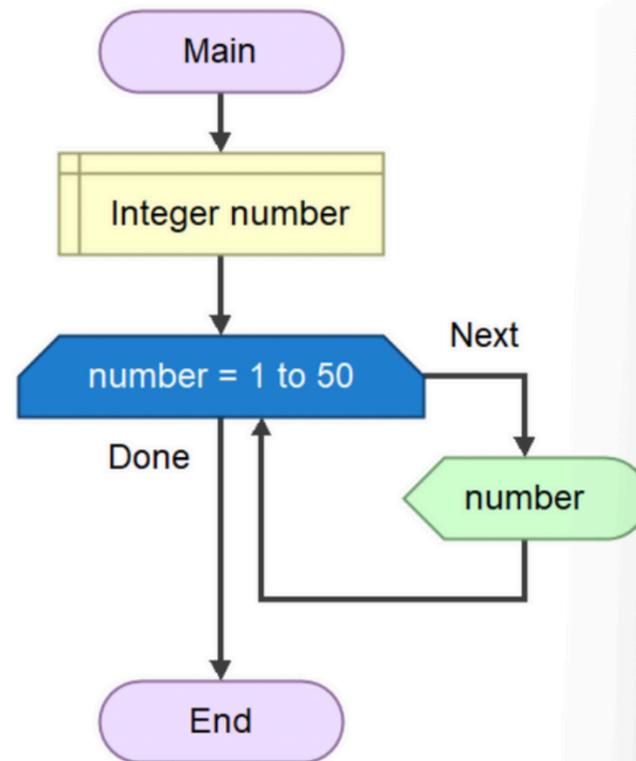
How many letters are there in the alphabet?

- 24
- 22
- 34
- 32
- 20
- 21
- 26

Correct!

# โครงสร้างการทำซ้ำแบบ for - Code #14

สถานการณ์ปัญหา: แสดงจำนวนตั้งแต่ 1 ถึง 50 เท่านั้น โดยใช้ for loop



The 'For Properties' dialog box is shown with the following settings:

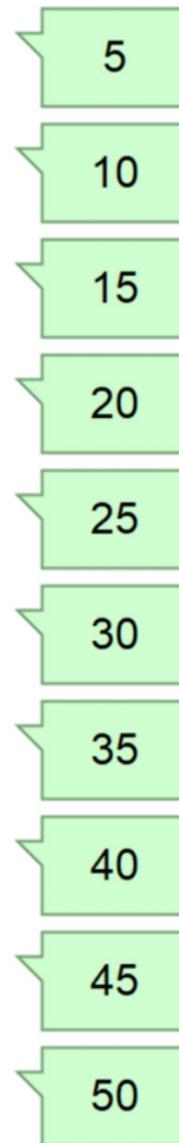
- Variable:** number
- Start Value:** 1
- End Value:** 50
- Direction:**  Increasing,  Decreasing
- Step By:** 1

Buttons: OK, Cancel

1	39
2	40
3	41
4	42
5	43
6	44
7	45
8	46
9	47
10	48
11	49
	50

# โครงสร้างการทำซ้ำแบบ for - Code #15

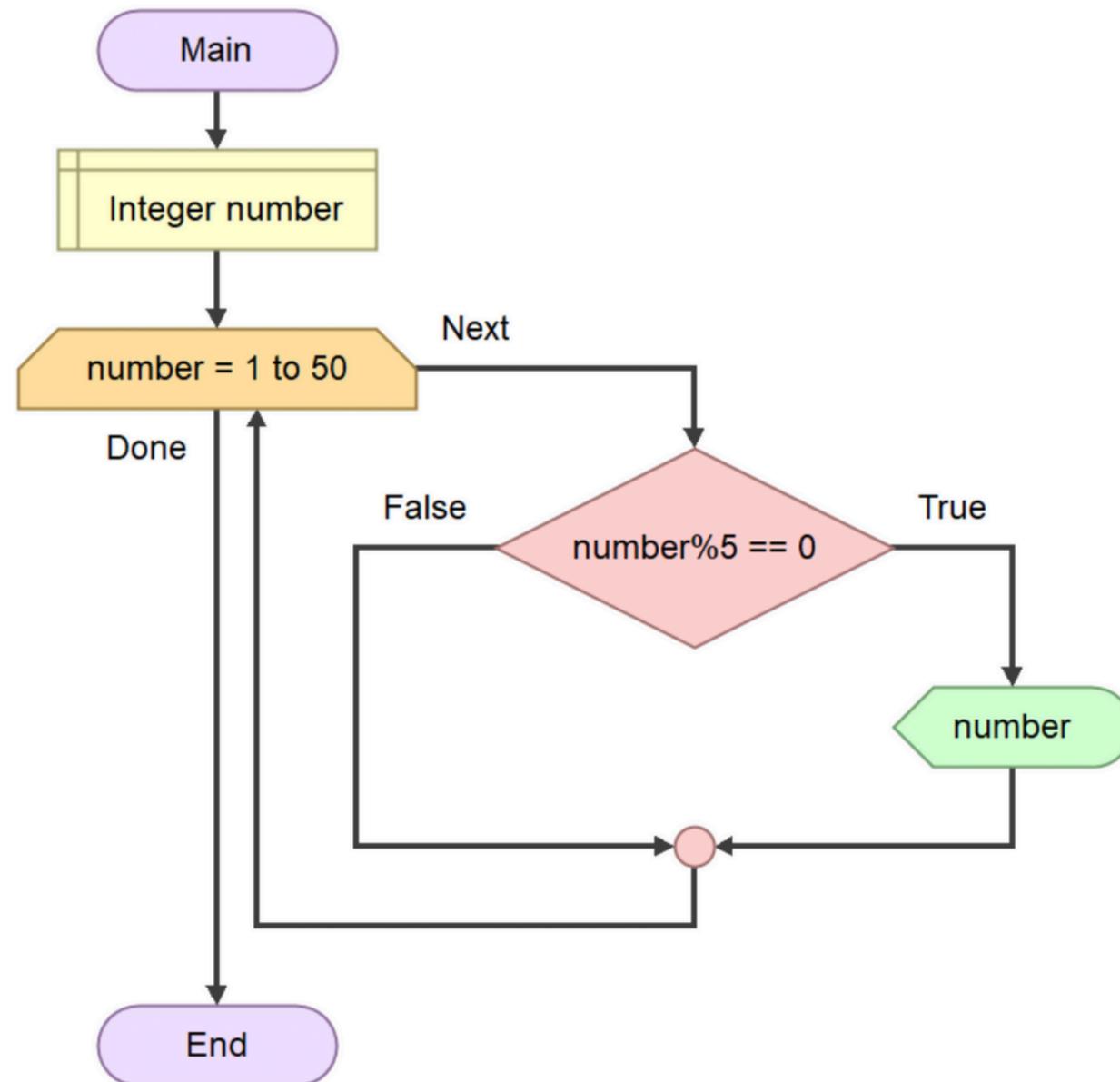
สถานการณ์ปัญหา: แสดงจำนวนตั้งแต่ 1 ถึง 50 ที่หาร 5 ลงตัวเท่านั้น โดยใช้ for loop



5  
10  
15  
20  
25  
30  
35  
40  
45  
50

# โครงสร้างการทำซ้ำแบบ for - Code #15

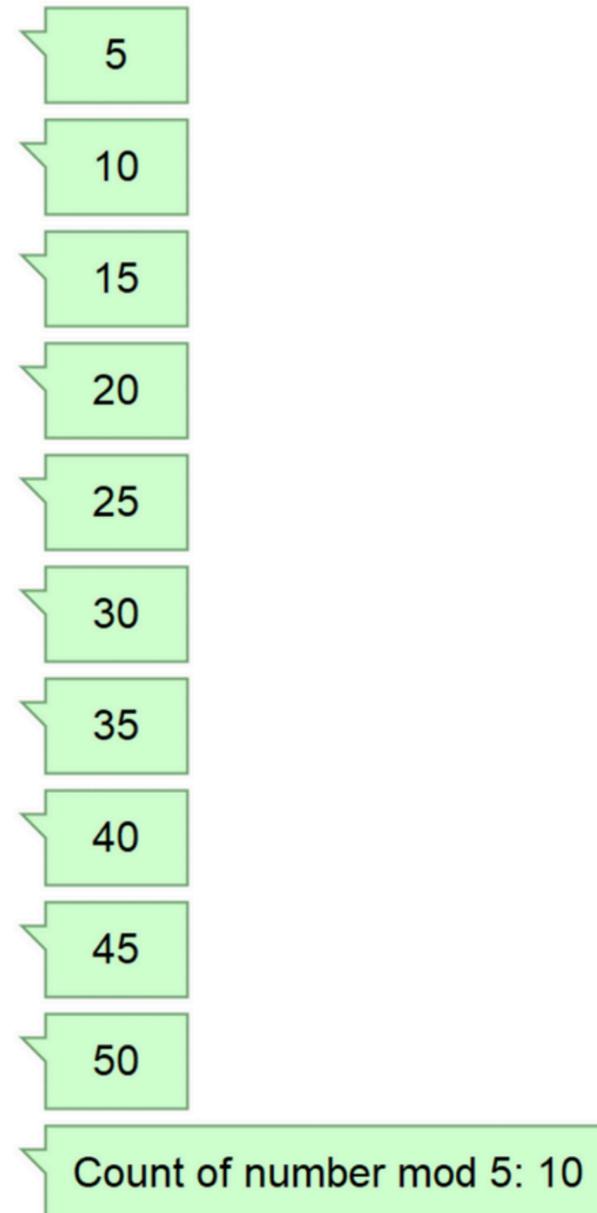
สถานการณ์ปัญหา: แสดงจำนวนตั้งแต่ 1 ถึง 50 ที่หาร 5 ลงตัวเท่านั้น โดยใช้ for loop



- 5
- 10
- 15
- 20
- 25
- 30
- 35
- 40
- 45
- 50

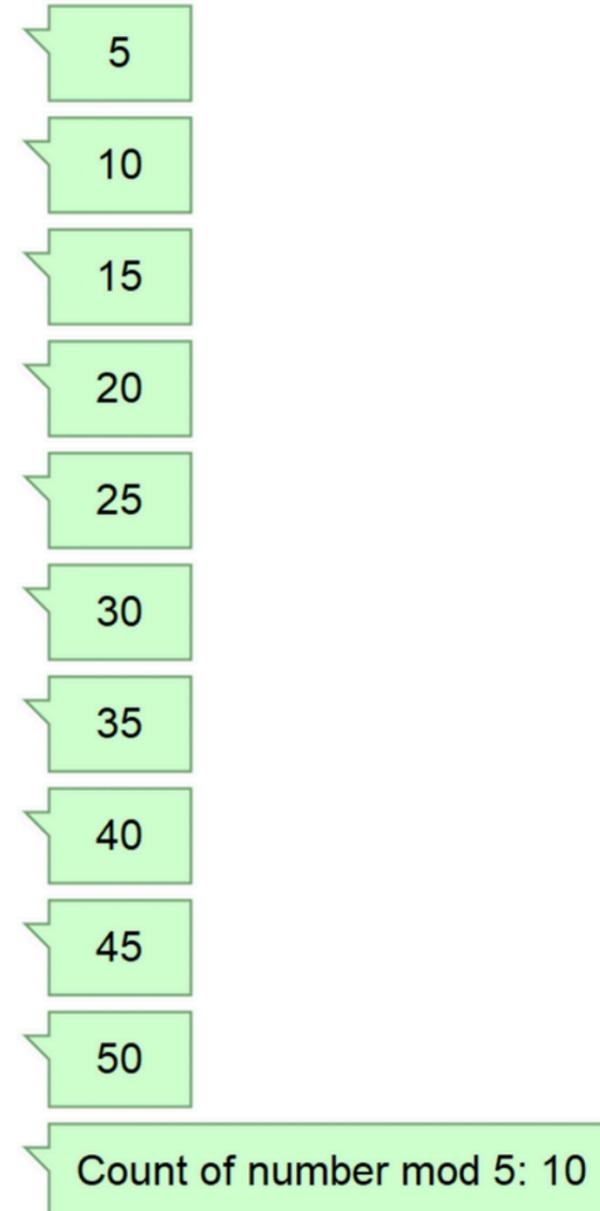
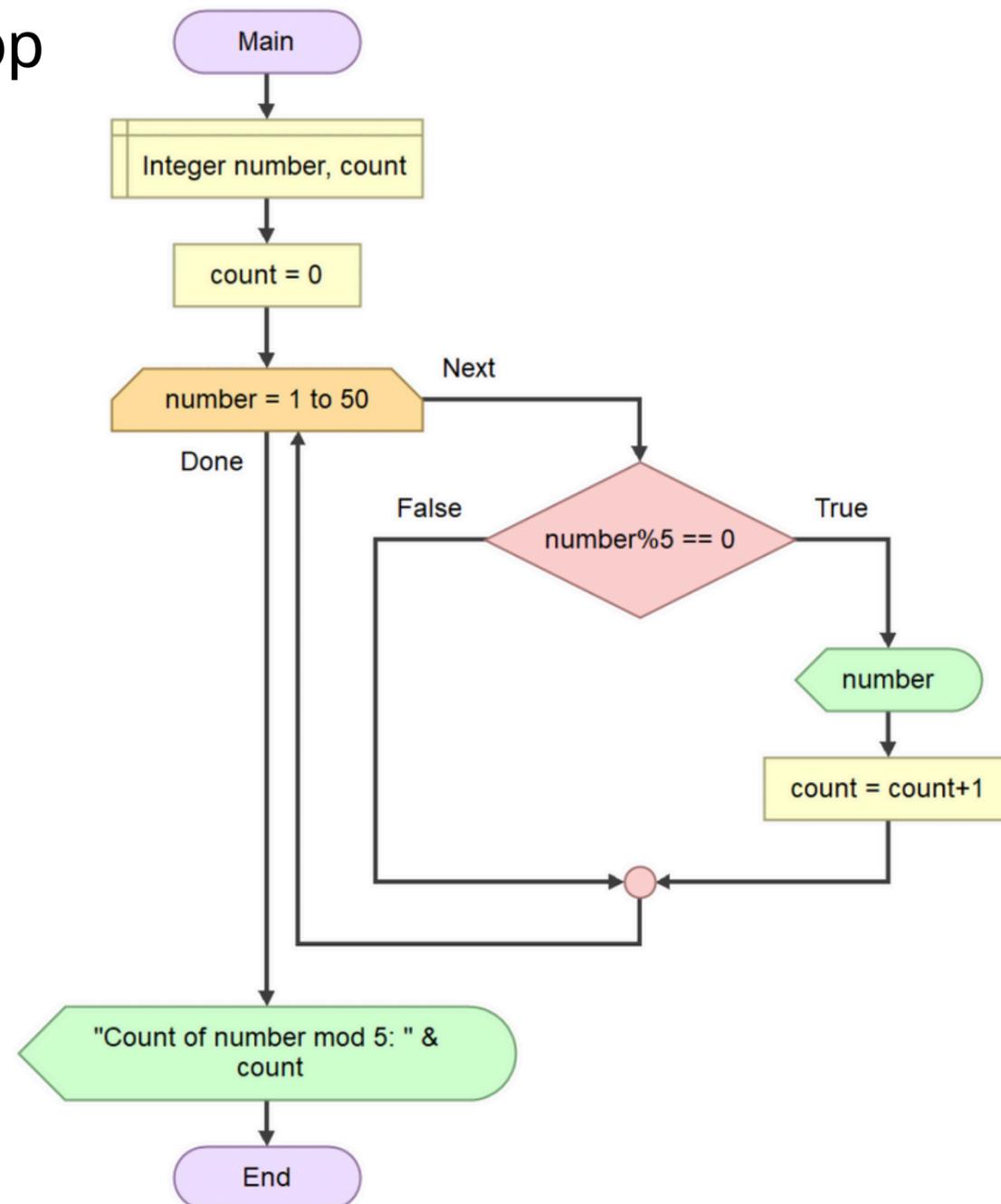
# โครงสร้างการทำซ้ำแบบ for - Code #16

**สถานการณ์ปัญหา:** แสดงจำนวนตั้งแต่ 1 ถึง 50 ที่หาร 5 ลงตัวเท่านั้น และสรุปว่ามีที่จำนวน โดยใช้ for loop



# โครงสร้างการทำซ้ำแบบ for - Code #16

สถานการณ์ปัญหา: แสดงจำนวนตั้งแต่ 1 ถึง 50 ที่หาร 5 ลงตัวเท่านั้น และสรุปว่ามีที่จำนวน โดยใช้ for loop



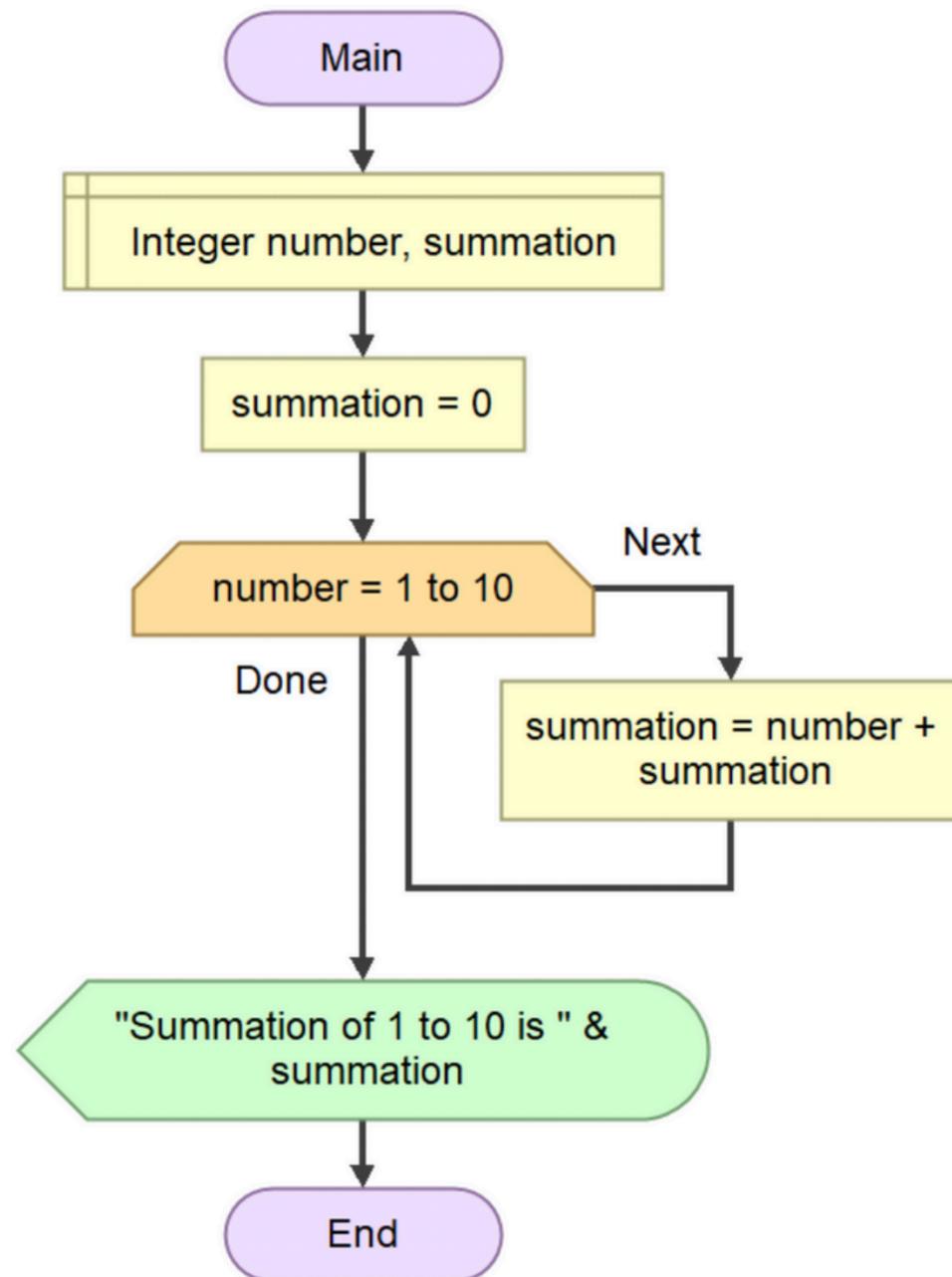
## โครงสร้างการทำซ้ำแบบ for - Code #17

สถานการณ์ปัญหา: หาผลรวมของจำนวนตั้งแต่ 1 ถึง 10 โดยการใช้ for loop

Summation of 1 to 10 is 55

# โครงสร้างการทำซ้ำแบบ for - Code #17

สถานการณ์ปัญหา: หาผลรวมของจำนวนตั้งแต่ 1 ถึง 10 โดยการใช้ for loop



Summation of 1 to 10 is 55

## โครงสร้างการทำซ้ำแบบ for - Code #18

**สถานการณ์ปัญหา:** หาผลรวมและค่าเฉลี่ยของจำนวนตั้งแต่ 1 ถึง 10 โดยการใช้ for loop

- ค่าเฉลี่ย (average) ได้จาก ผลรวมของจำนวนทั้งหมด หารด้วย จำนวนทั้งหมด
- $average = summation / count$

Count of all number: 10

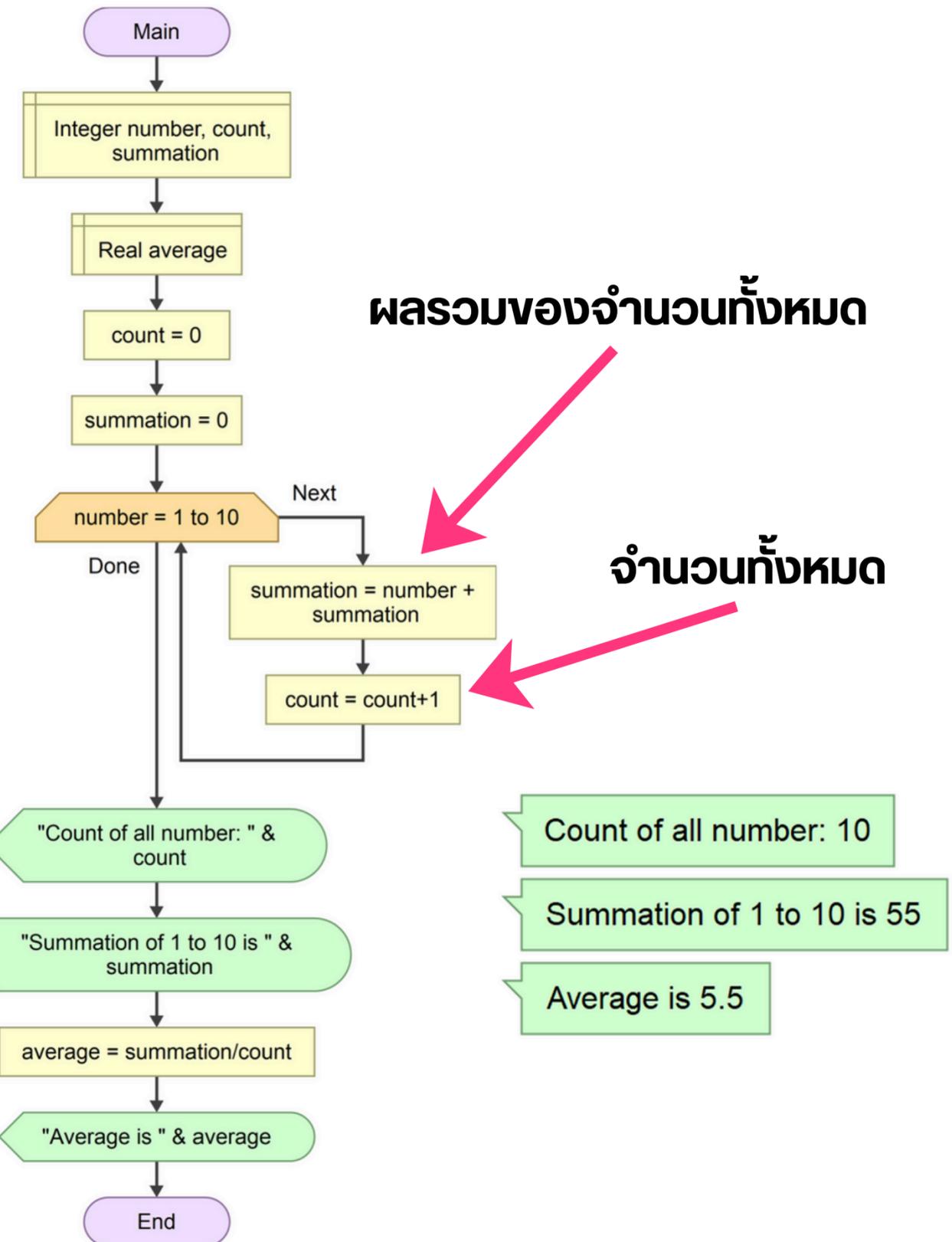
Summation of 1 to 10 is 55

Average is 5.5

# โครงสร้างการทำซ้ำแบบ for - Code #18

สถานการณ์ปัญหา: หาผลรวมและค่าเฉลี่ยของจำนวนตั้งแต่ 1 ถึง 10 โดยการใช้ for loop

- ค่าเฉลี่ย (average) ได้จาก ผลรวมของจำนวนทั้งหมดหารด้วยจำนวนทั้งหมด
- $average = summation / count$



## โครงสร้างการทำซ้ำแบบ for - Code #19

**สถานการณ์ปัญหา:** หาผลรวมและค่าเฉลี่ยของจำนวนตั้งแต่ 1 ถึง 1,000 โดยการใช้ for loop

- ค่าเฉลี่ย (average) ได้จาก ผลรวมของจำนวนทั้งหมดหารด้วย จำนวนทั้งหมด
- $average = summation / count$

Count of all number: 1000

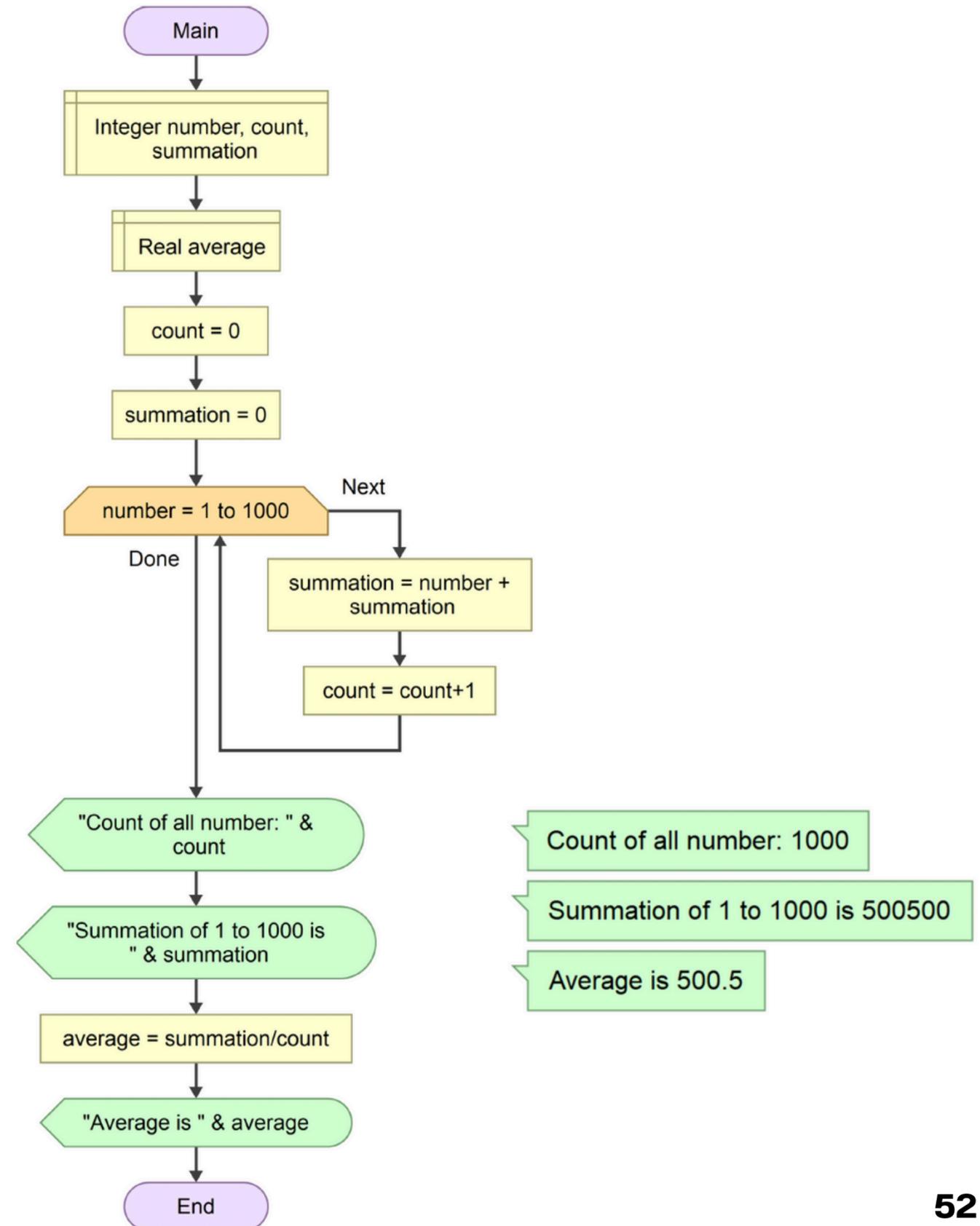
Summation of 1 to 1000 is 500500

Average is 500.5

# โครงสร้างการทำซ้ำแบบ for - Code #19

สถานการณ์ปัญหา: หาผลรวมและค่าเฉลี่ยของจำนวนตั้งแต่ 1 ถึง 1,000 โดยการใช้ for loop

- ค่าเฉลี่ย (average) ได้จาก ผลรวมของจำนวนทั้งหมด หารด้วย จำนวนทั้งหมด
- $average = summation / count$



# โครงสร้างการทำซ้ำแบบ for - Code #20

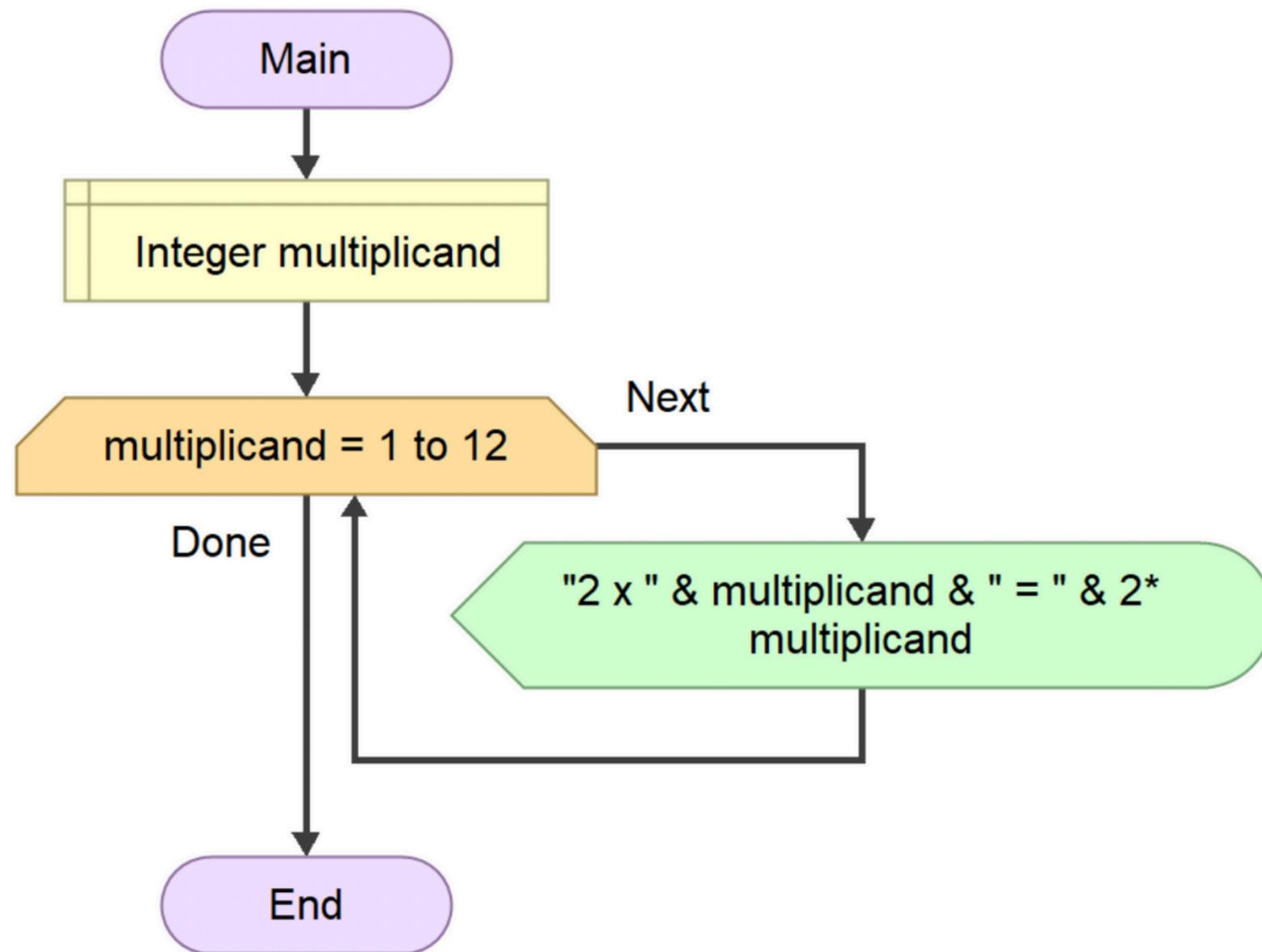
สถานการณ์ปัญหา: แสดงสูตรคูณแม่ 2 โดยเริ่มตั้งแต่  $2 \times 1 = 2$  ถึง  $2 \times 12 = 24$  ด้วย for loop

Multiplier ตัวคูณเสมอ	Multiply การคูณ	Multiplicand ตัวตั้ง	=	Product ผลการคูณ
2	*	1	=	2
2	*	2	=	4
2	*	3	=	6

- 2 x 1 = 2
- 2 x 2 = 4
- 2 x 3 = 6
- 2 x 4 = 8
- 2 x 5 = 10
- 2 x 6 = 12
- 2 x 7 = 14
- 2 x 8 = 16
- 2 x 9 = 18
- 2 x 10 = 20
- 2 x 11 = 22
- 2 x 12 = 24

# โครงสร้างการทำซ้ำแบบ for - Code #20

สถานการณ์ปัญหา: แสดงสูตรคูณแม่ 2 โดยเริ่มตั้งแต่  $2 \times 1 = 2$  ถึง  $2 \times 12 = 24$  ด้วย for loop



- 2 x 1 = 2
- 2 x 2 = 4
- 2 x 3 = 6
- 2 x 4 = 8
- 2 x 5 = 10
- 2 x 6 = 12
- 2 x 7 = 14
- 2 x 8 = 16
- 2 x 9 = 18
- 2 x 10 = 20
- 2 x 11 = 22
- 2 x 12 = 24

# โครงสร้างการทำซ้ำแบบ for - Code #21

สถานการณ์ปัญหา: แสดงสูตรคูณแม่ 3 โดยเริ่มตั้งแต่  $3 \times 1 = 3$  ถึง  $3 \times 12 = 36$  ด้วย for loop

Multiplier  
ตัวคูณเสมอ

3

3

3

Multiply  
การคูณ

\*

\*

\*

Multiplicand  
ตัวตั้ง

1

2

3

=

=

=

Product  
ผลการคูณ

3

6

9

$3 \times 1 = 3$

$3 \times 2 = 6$

$3 \times 3 = 9$

$3 \times 4 = 12$

$3 \times 5 = 15$

$3 \times 6 = 18$

$3 \times 7 = 21$

$3 \times 8 = 24$

$3 \times 9 = 27$

$3 \times 10 = 30$

$3 \times 11 = 33$

$3 \times 12 = 36$

# โครงสร้างการทำซ้ำแบบ for - Code #22

**สถานการณ์ปัญหา:** แสดง "\*" ตามจำนวนของตัวแปร number โดย number จะมีค่าตั้งแต่ 1 ถึง 10 เช่น

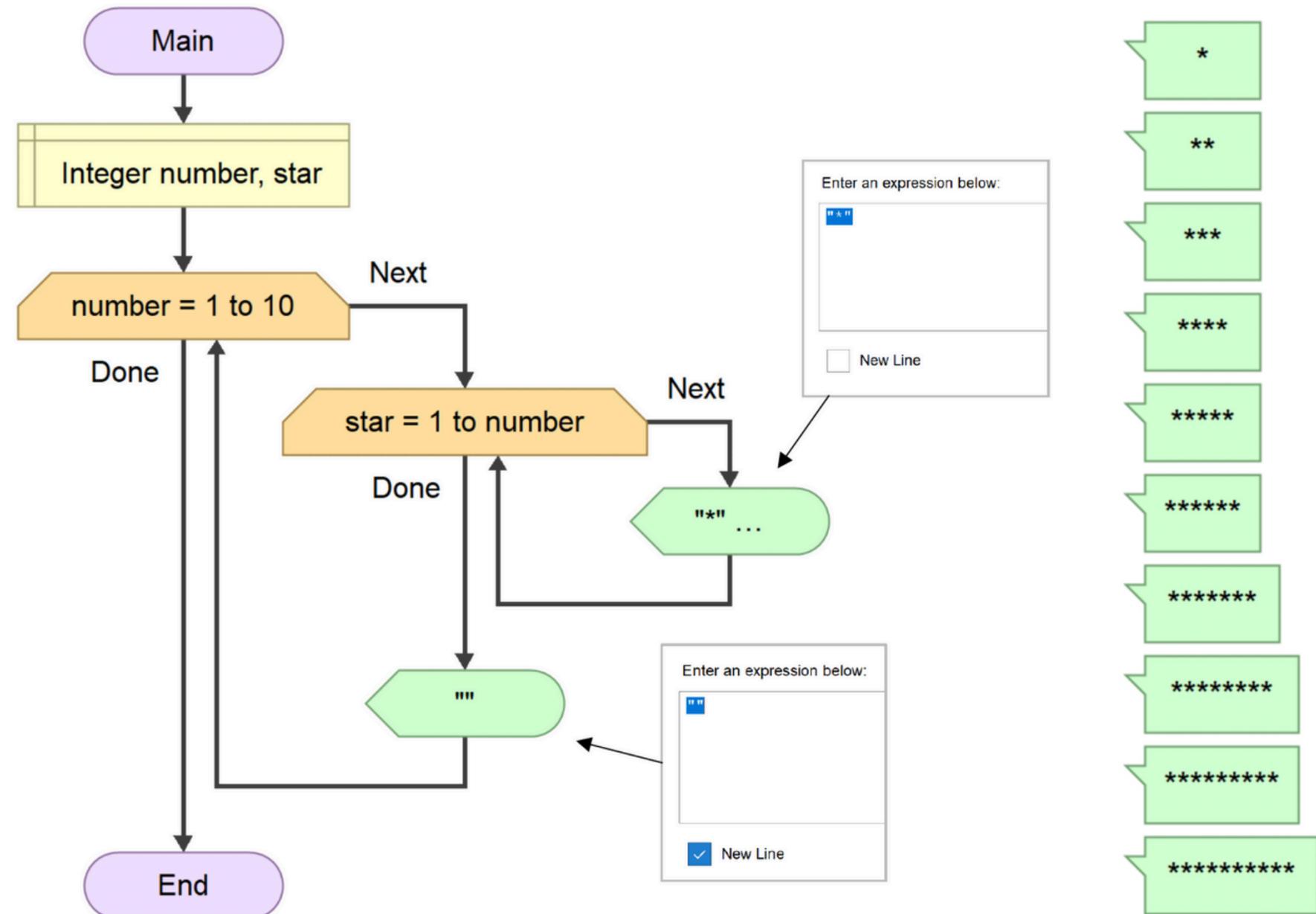
- number มีค่าเป็น 1 แสดง \*
- number มีค่าเป็น 2 แสดง \*\*
- number มีค่าเป็น 5 แสดง \*\*\*\*\*



# โครงสร้างการทำซ้ำแบบ for - Code #22

สถานการณ์ปัญหา: แสดง "\*" ตามจำนวนของตัวแปร number โดย number จะมีค่าตั้งแต่ 1 ถึง 10 เช่น

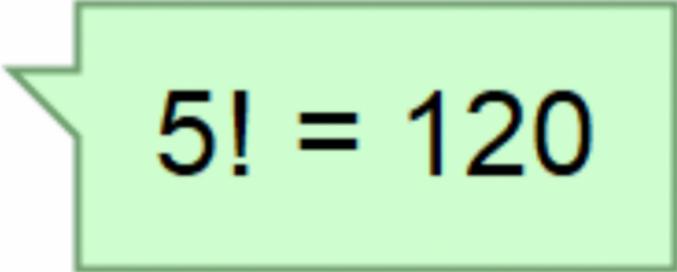
- number มีค่าเป็น 1 แสดง \*
- number มีค่าเป็น 2 แสดง \*\*
- number มีค่าเป็น 5 แสดง \*\*\*\*\*



## โครงสร้างการทำซ้ำแบบ for - Code #23

**โจทย์สถานการณ์:** ต้องการหาค่า Factorial ของ 5 หรือ 5! โดยใช้ for loop  
**Factorial (n!) คือ ผลคูณของจำนวนเต็มบวกตั้งแต่ 1 ถึง n**  
เช่น

- $1! = 1$
- $2! = 2 \times 1 = 2$
- $3! = 3 \times 2 \times 1 = 6$
- $4! = 4 \times 3 \times 2 \times 1 = 24$
- $5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$

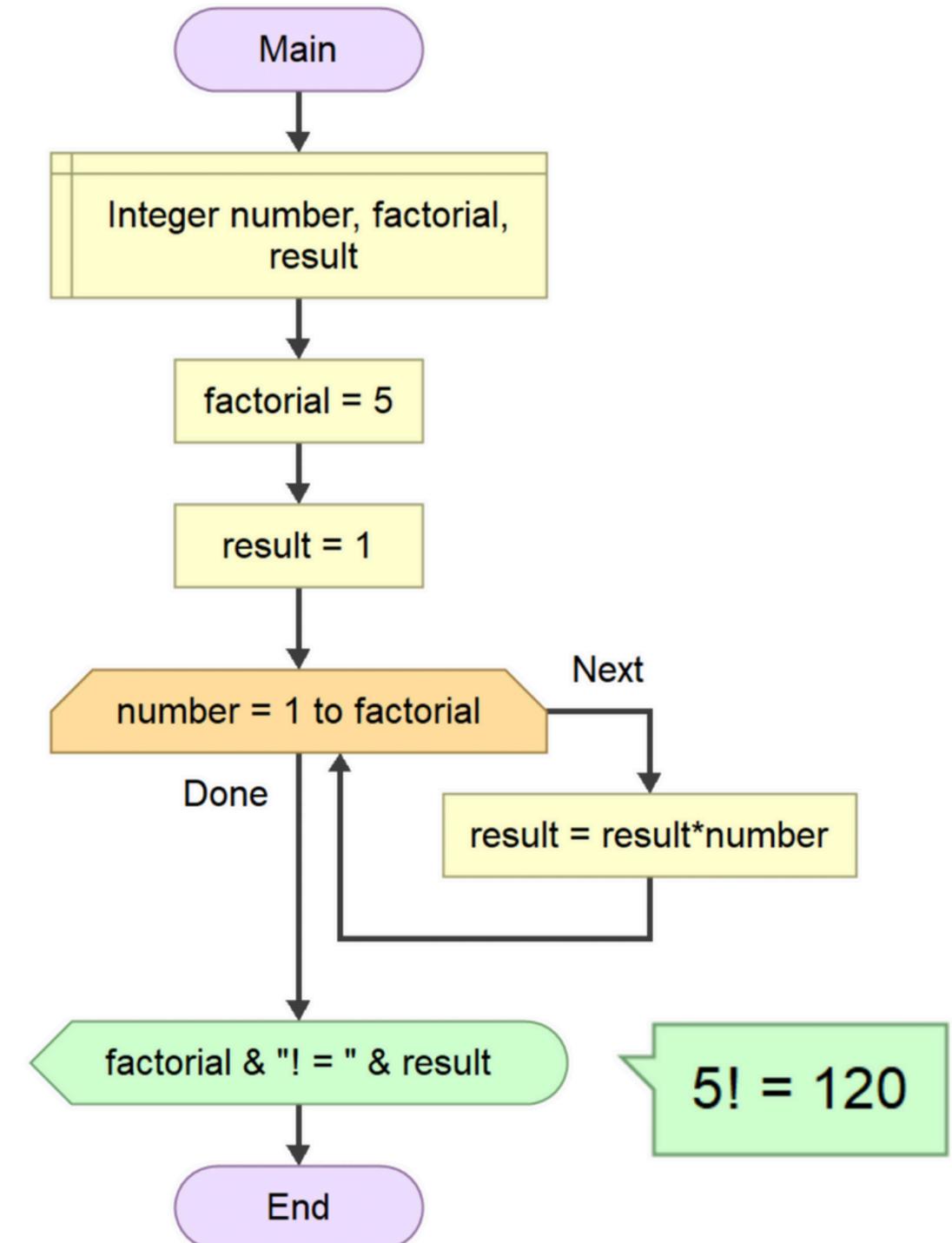

$$5! = 120$$

# โครงสร้างการทำซ้ำแบบ for - Code #23

**โจทย์สถานการณ์:** ต้องการหาค่า Factorial ของ 5 หรือ 5!  
โดยใช้ for loop

**Factorial (n!) คือ ผลคูณของจำนวนเต็มบวกตั้งแต่ 1 ถึง n**  
เช่น

- $1! = 1$
- $2! = 2 \times 1 = 2$
- $3! = 3 \times 2 \times 1 = 6$
- $4! = 4 \times 3 \times 2 \times 1 = 24$
- $5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$



## โครงสร้างการทำซ้ำแบบ for - Code #24

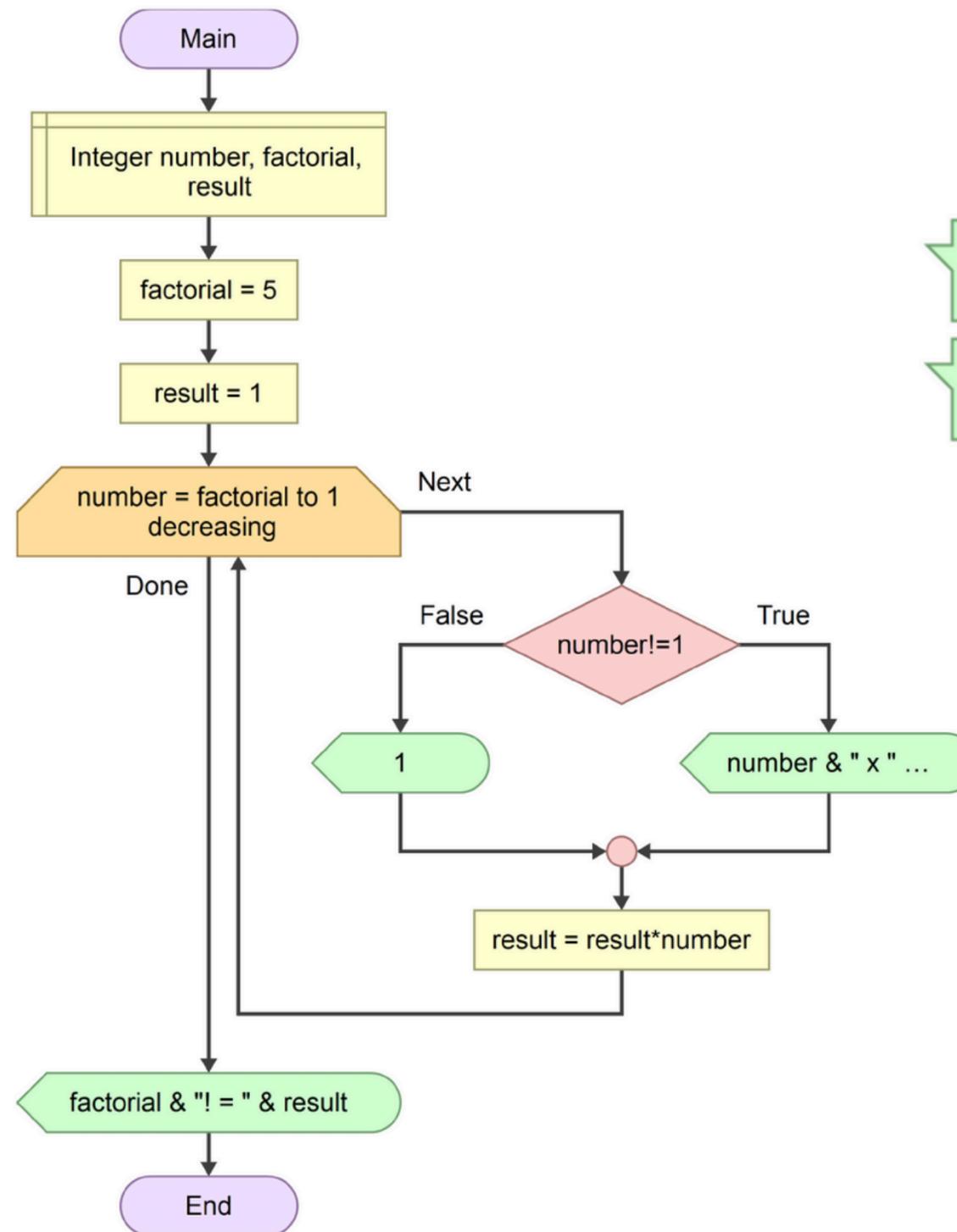
**โจทย์สถานการณ์:** ต้องการแสดงจำนวนทั้งหมดที่คูณกันจากค่า Factorial ของ 5 หรือ 5! โดยใช้ for loop

$$5 \times 4 \times 3 \times 2 \times 1$$

$$5! = 120$$

# โครงสร้างการทำซ้ำแบบ for - Code #24

โจทย์สถานการณ์: ต้องการแสดงจำนวนทั้งหมดที่คูณกันจากค่า Factorial ของ 5 หรือ 5! โดยใช้ for loop



5 x 4 x 3 x 2 x 1  
5! = 120

## โครงสร้างการทำซ้ำแบบ for - Code #25

**โจทย์สถานการณ์:** ต้องการแสดงจำนวนทั้งหมดที่คูณกันจากค่า Factorial ที่รับจากผู้ใช้ โดยใช้ for loop

Please enter factorial number:

8 x 7 x 6 x 5 x 4 x 3 x 2 x 1

8! = 40320

8

# โครงสร้างการทำซ้ำแบบ for - Code #25

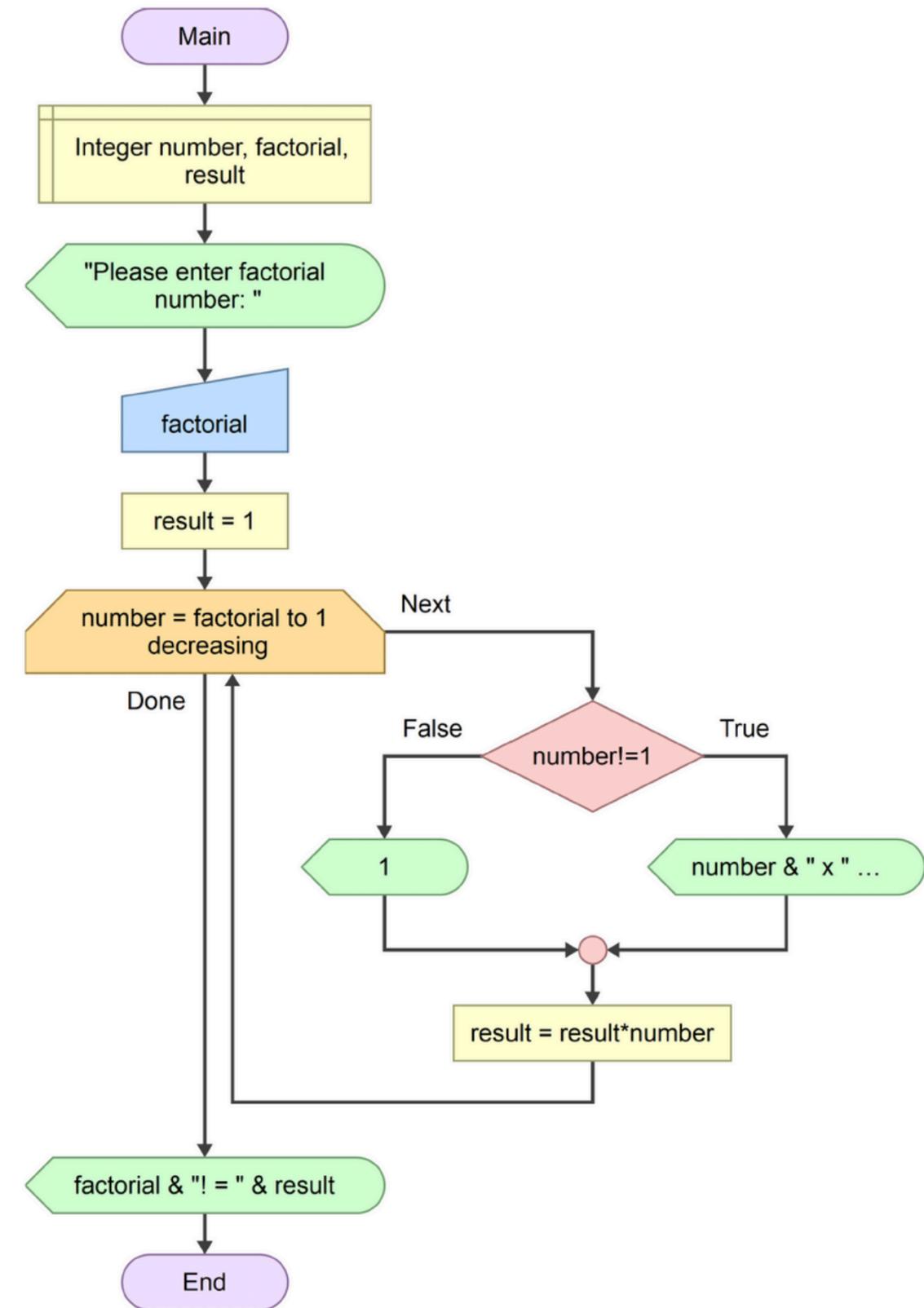
โจทย์สถานการณ์: ต้องการแสดงจำนวนทั้งหมดที่คุณก้นจากค่า Factorial ที่รับจากผู้ใช้ โดยใช้ for loop

Please enter factorial number:

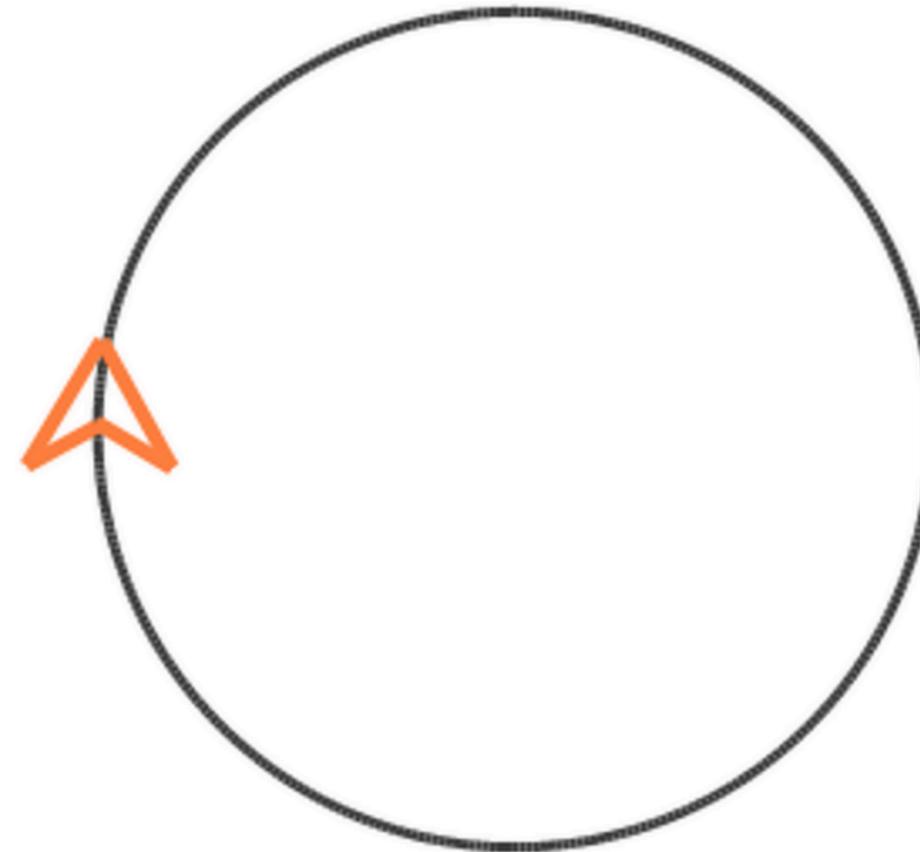
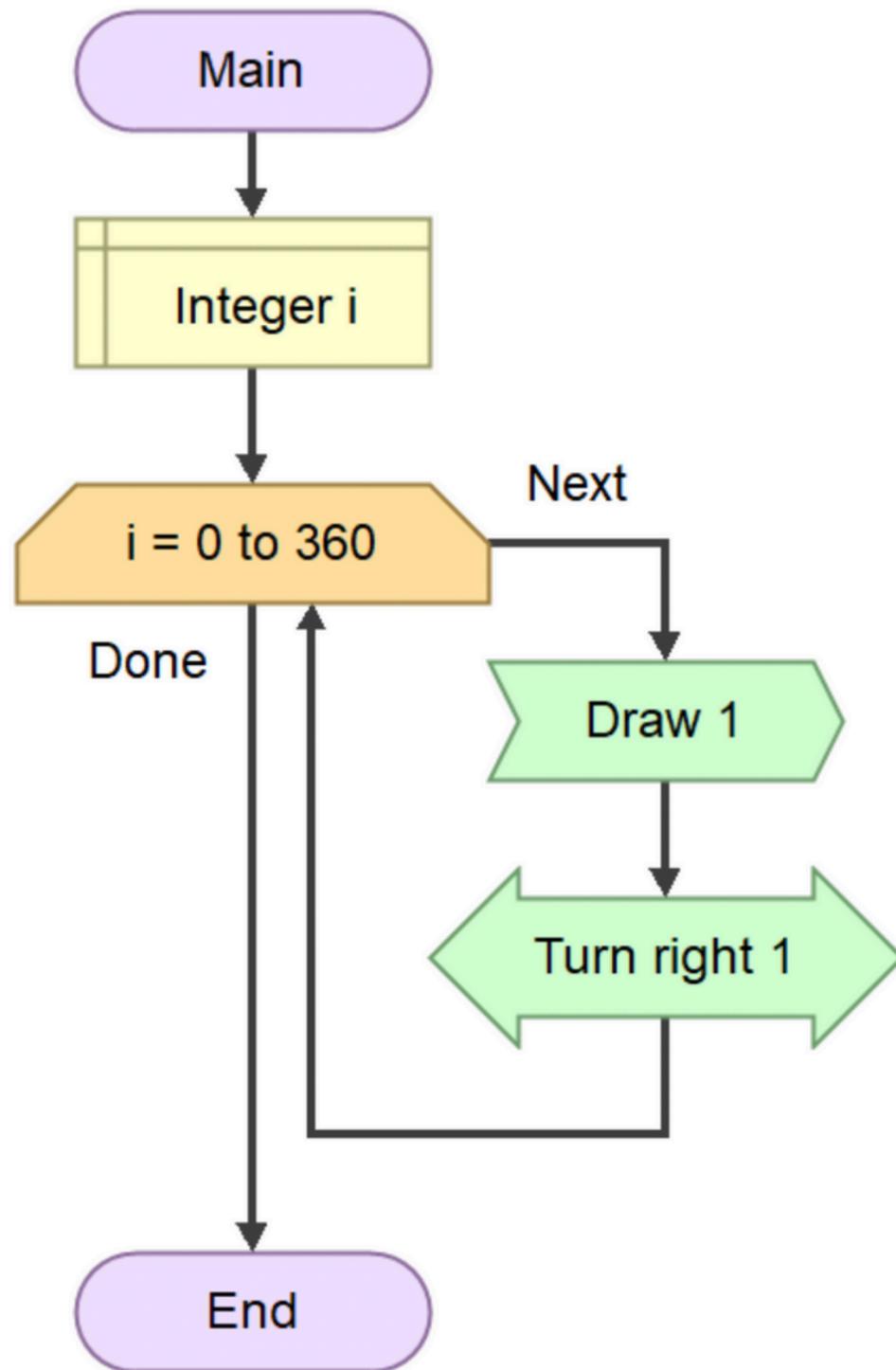
8 x 7 x 6 x 5 x 4 x 3 x 2 x 1

8! = 40320

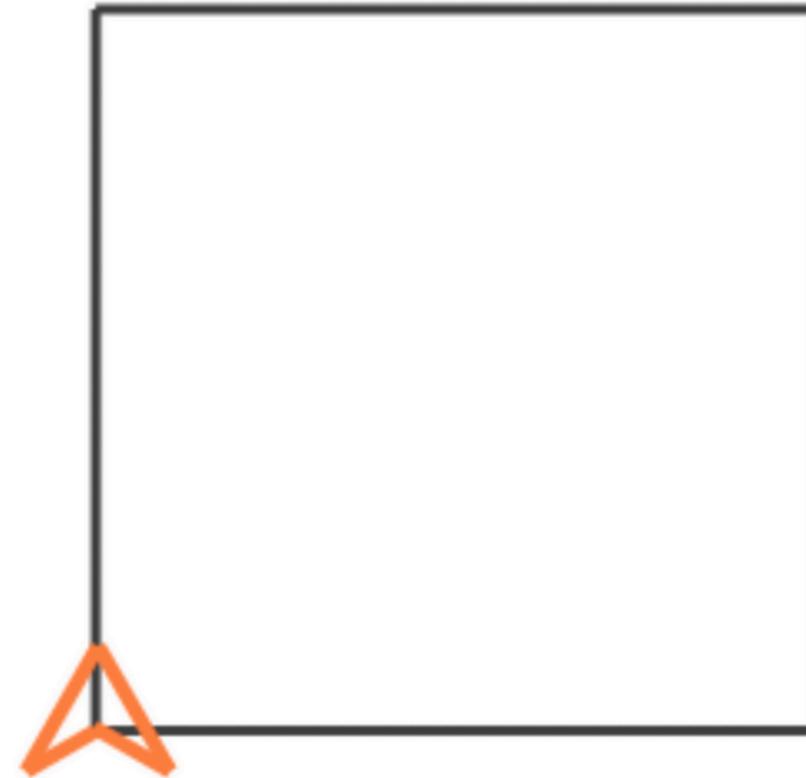
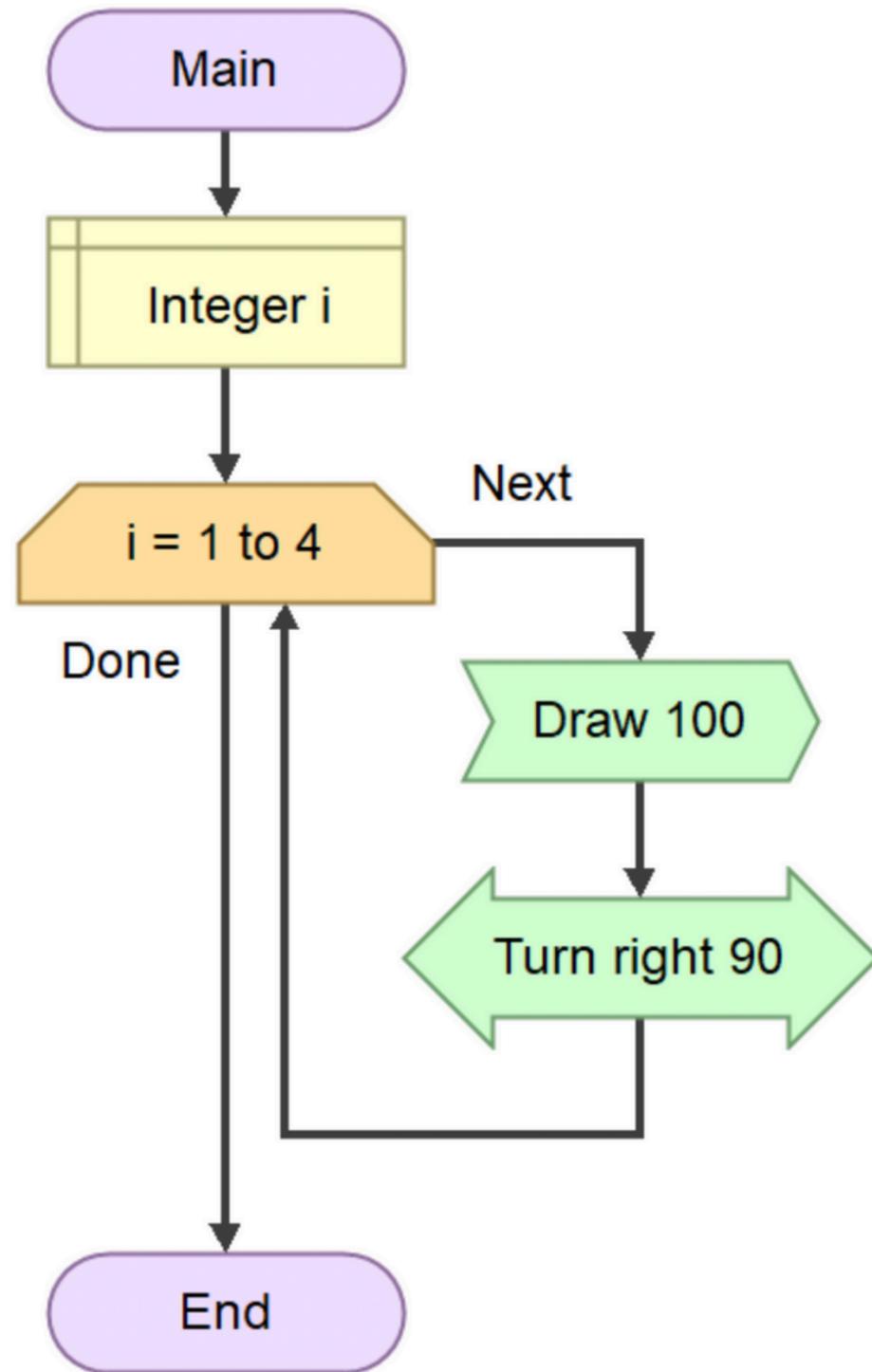
8



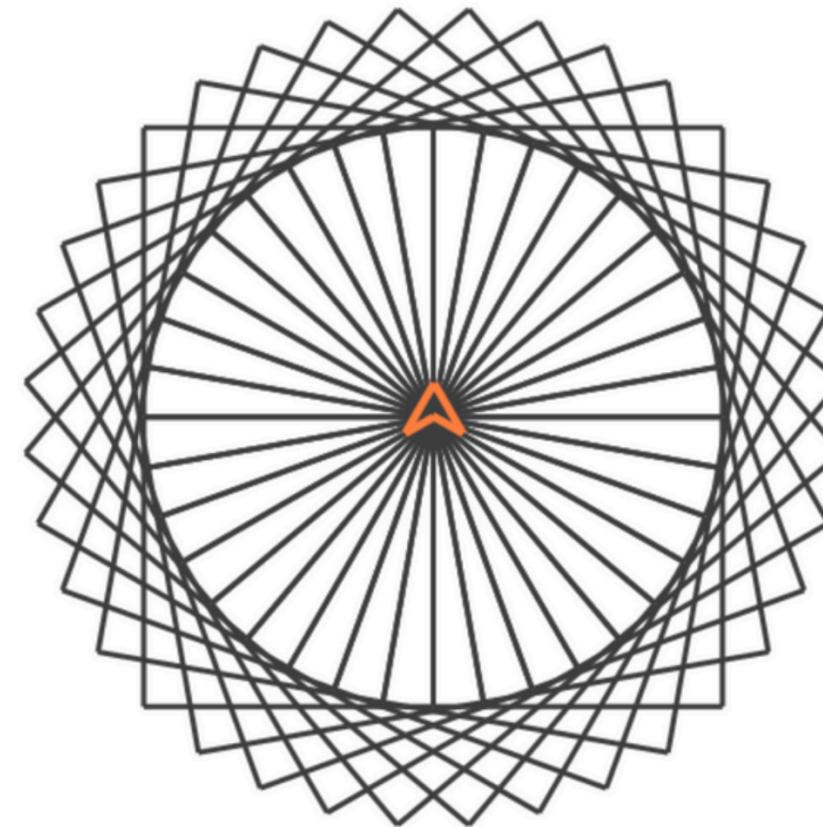
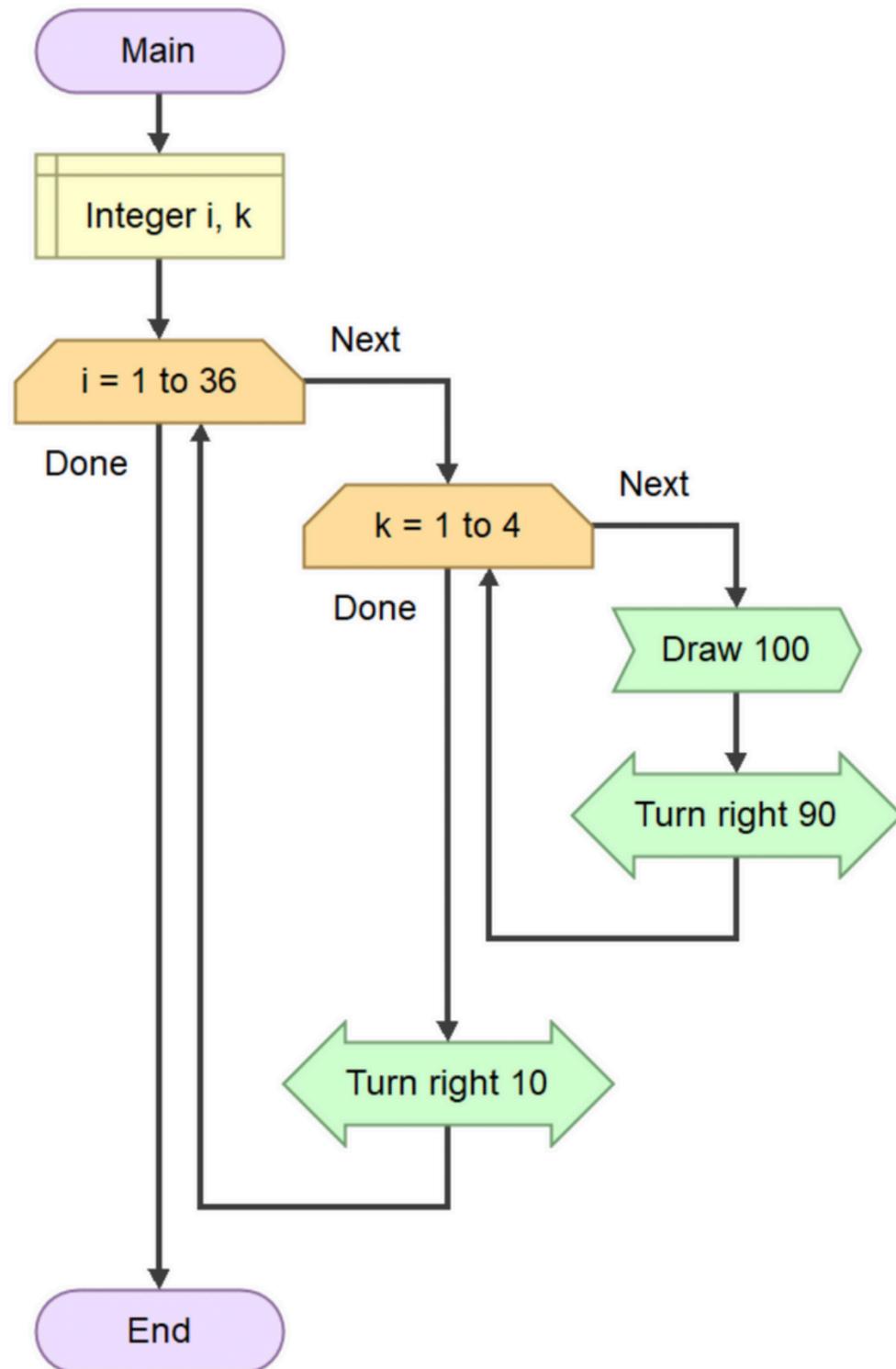
# โครงสร้างการทำซ้ำแบบ for ร่วมกับการวาดภาพกราฟิก



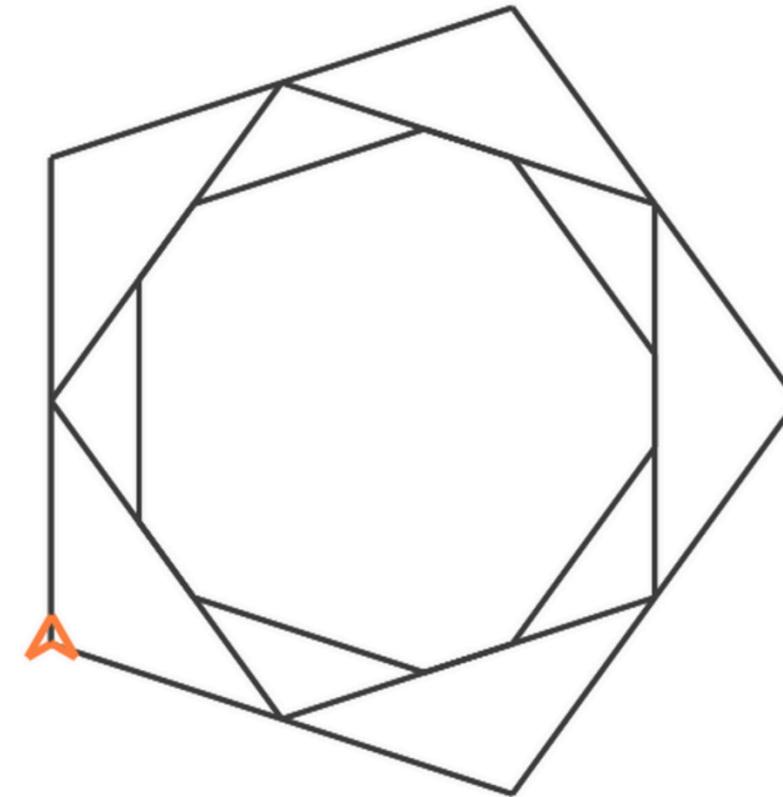
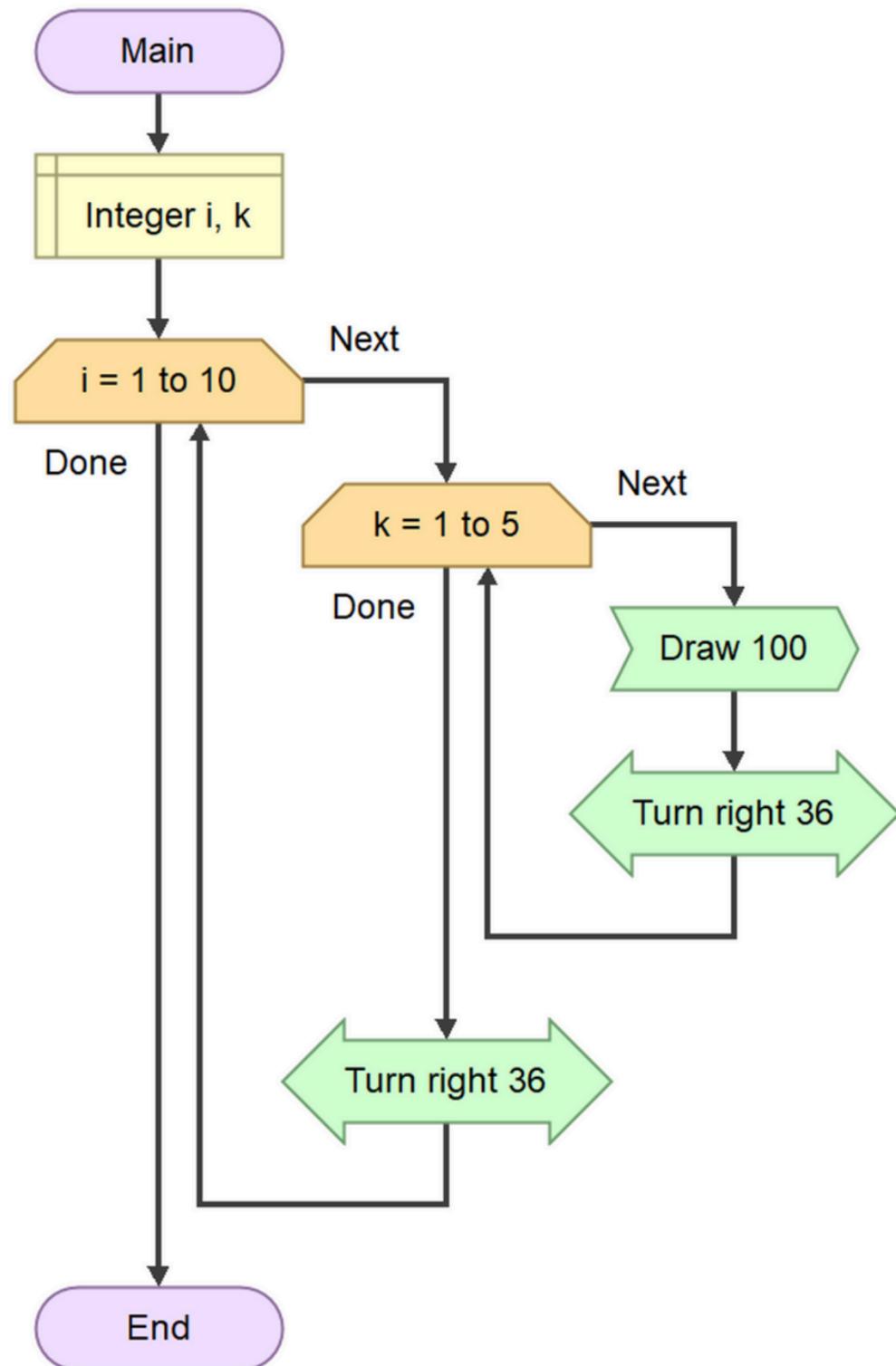
# โครงสร้างการทำซ้ำแบบ for ร่วมกับการวาดภาพกราฟิก



# โครงสร้างการทำซ้ำแบบ for ร่วมกับการวาดภาพกราฟิก



# โครงสร้างการทำซ้ำแบบ for ร่วมกับการวาดภาพกราฟิก



## มอบหมายงานครั้งที่ 4.2 - Assignment #4.2

ข้อ 1: แสดง “\*” ตามจำนวนของตัวแปร number โดย number จะมีค่าตั้งแต่ 10 ถึง 1 โดยใช้ for loop เท่านั้น

- number มีค่าเป็น 10 แสดง \*\*\*\*\*
- number มีค่าเป็น 9 แสดง \*\*\*\*\*
- number มีค่าเป็น 5 แสดง \*\*\*\*\*



## มอบหมายงานครั้งที่ 4.2 - Assignment #4.2

ข้อ 2: ให้เขียนโปรแกรมสุตรคูณแม่ 2 ถึงแม่ 50 โดยใช้ for loop เท่านั้น

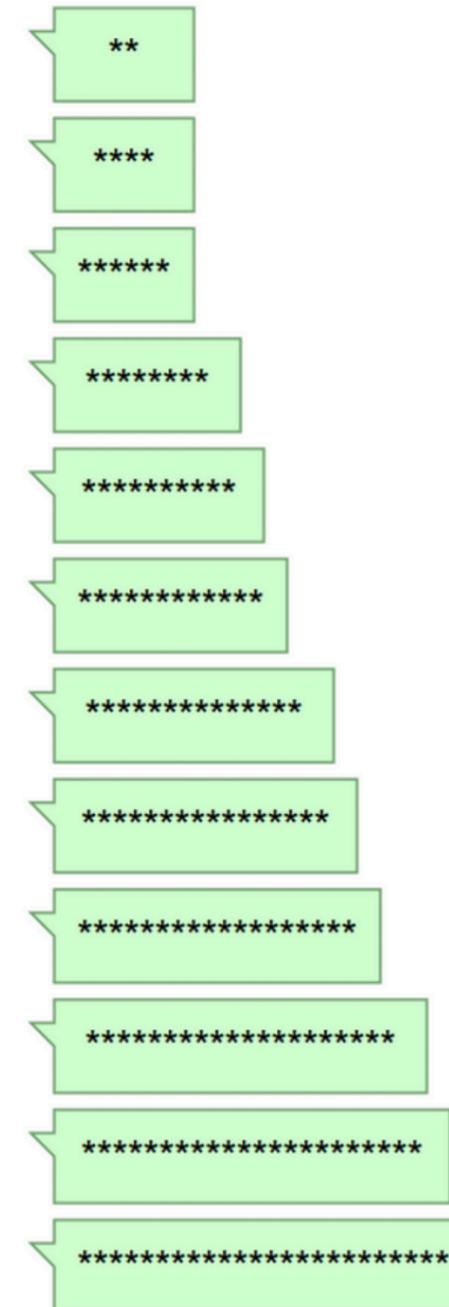
$2 \times 1 = 2$	$3 \times 1 = 3$	$49 \times 1 = 49$	$50 \times 1 = 50$
$2 \times 2 = 4$	$3 \times 2 = 6$	$49 \times 2 = 98$	$50 \times 2 = 100$
$2 \times 3 = 6$	$3 \times 3 = 9$	$49 \times 3 = 147$	$50 \times 3 = 150$
$2 \times 4 = 8$	$3 \times 4 = 12$	$49 \times 4 = 196$	$50 \times 4 = 200$
$2 \times 5 = 10$	$3 \times 5 = 15$	$49 \times 5 = 245$	$50 \times 5 = 250$
$2 \times 6 = 12$	$3 \times 6 = 18$	$49 \times 6 = 294$	$50 \times 6 = 300$
$2 \times 7 = 14$	$3 \times 7 = 21$	$49 \times 7 = 343$	$50 \times 7 = 350$
$2 \times 8 = 16$	$3 \times 8 = 24$	$49 \times 8 = 392$	$50 \times 8 = 400$
$2 \times 9 = 18$	$3 \times 9 = 27$	$49 \times 9 = 441$	$50 \times 9 = 450$
$2 \times 10 = 20$	$3 \times 10 = 30$	$49 \times 10 = 490$	$50 \times 10 = 500$
$2 \times 11 = 22$	$3 \times 11 = 33$	$49 \times 11 = 539$	$50 \times 11 = 550$
$2 \times 12 = 24$	$3 \times 12 = 36$	$49 \times 12 = 588$	$50 \times 12 = 600$

## มอบหมายงานครั้งที่ 4.2 - Assignment #4.2

ข้อ 3: แสดง "\*" ตามจำนวนของตัวแปร number ที่เป็นเลขคู่เท่านั้น โดย number จะมีค่าตั้งแต่ 1 ถึง 30 โดยใช้ for loop เท่านั้น

เช่น

- number มีค่าเป็น 1 ไม่แสดง
- number มีค่าเป็น 2 แสดง \*\*
- number มีค่าเป็น 3 ไม่แสดง
- number มีค่าเป็น 4 แสดง \*\*\*\*
- number มีค่าเป็น 5 ไม่แสดง
- number มีค่าเป็น 6 แสดง \*\*\*\*\*



# Post Test

**Question:**

**1. ข้อใดอธิบายความหมายของ “อัลกอริทึม (Algorithm)”  
ได้ถูกต้องที่สุด**

**A**

โปรแกรมที่เขียนด้วย  
ภาษา Python

**C**

ชุดขั้นตอนที่ชัดเจนเพื่อแก้  
ปัญหาหรือดำเนินงาน

**B**

ชุดคำสั่งที่ทำงานแบบสุ่ม

**D**

ผังงานที่ใช้วาดแทนโปรแกรม

**Question:**

## 2. ข้อใด ไม่ใช่ คุณลักษณะของอัลกอริทึมที่ดี

**A**

มีจุดสิ้นสุดการทำงาน

**C**

ขึ้นกับภาษาโปรแกรม

**B**

ไม่คลุมเครือ

**D**

มีขั้นตอนตายตัว

**Question:**

### 3. อัลกอริทึมหรือผังงานที่มีโครงสร้างแบบใด ที่ทำงานแบบเรียงลำดับ

**A** Sequential Structure

**C** Iteration Structure

**B** Conditional Structure

**D** Decision Structure

**Question:**

## 4. อัลกอริทึมหรือผังงานที่มีโครงสร้างแบบใด ที่ทำงานแบบทำซ้ำ

**A** Sequential Structure

**B** Conditional Structure

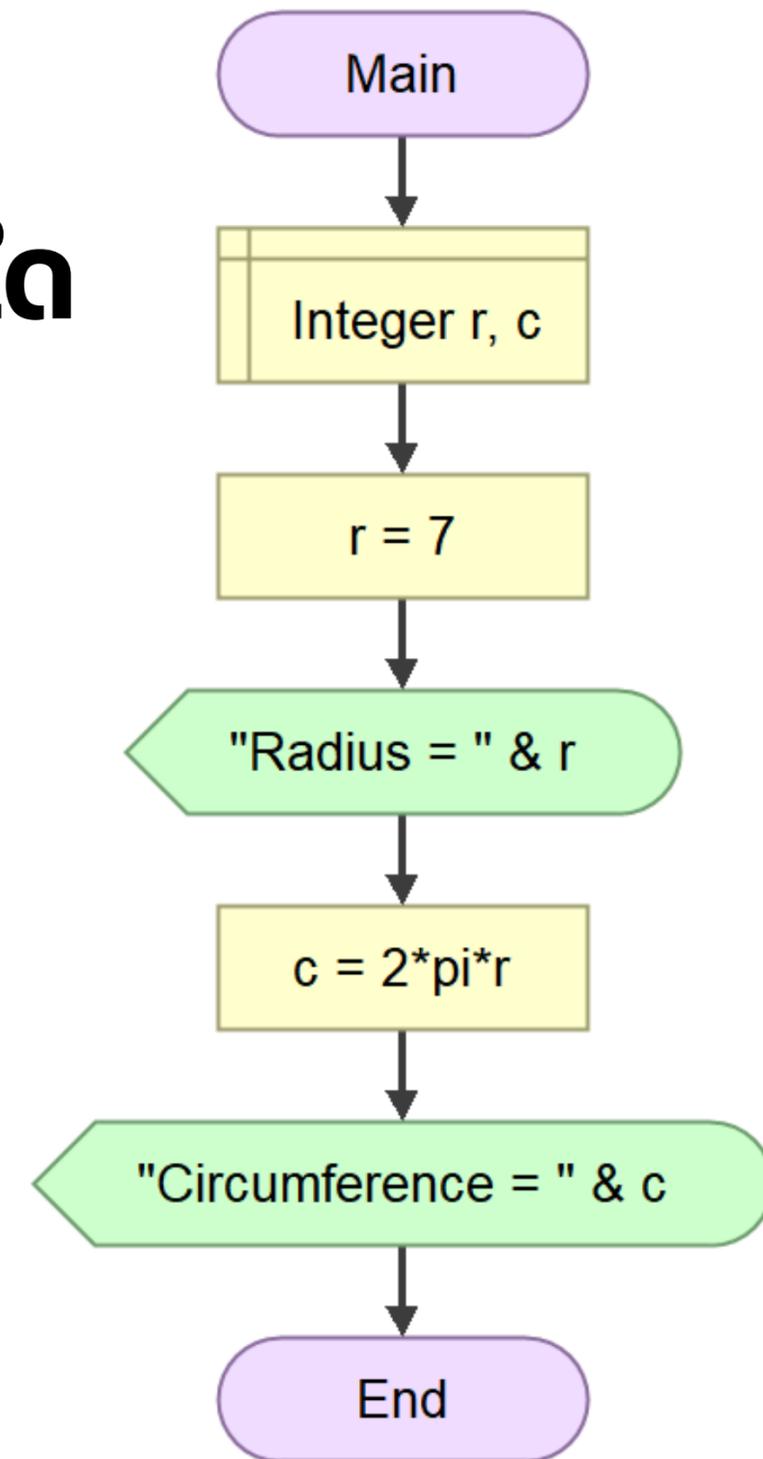
**C** Iteration Structure

**D** Decision Structure

**Question:**

## 5. จากภาพเป็นโครงสร้างผังงานแบบใด

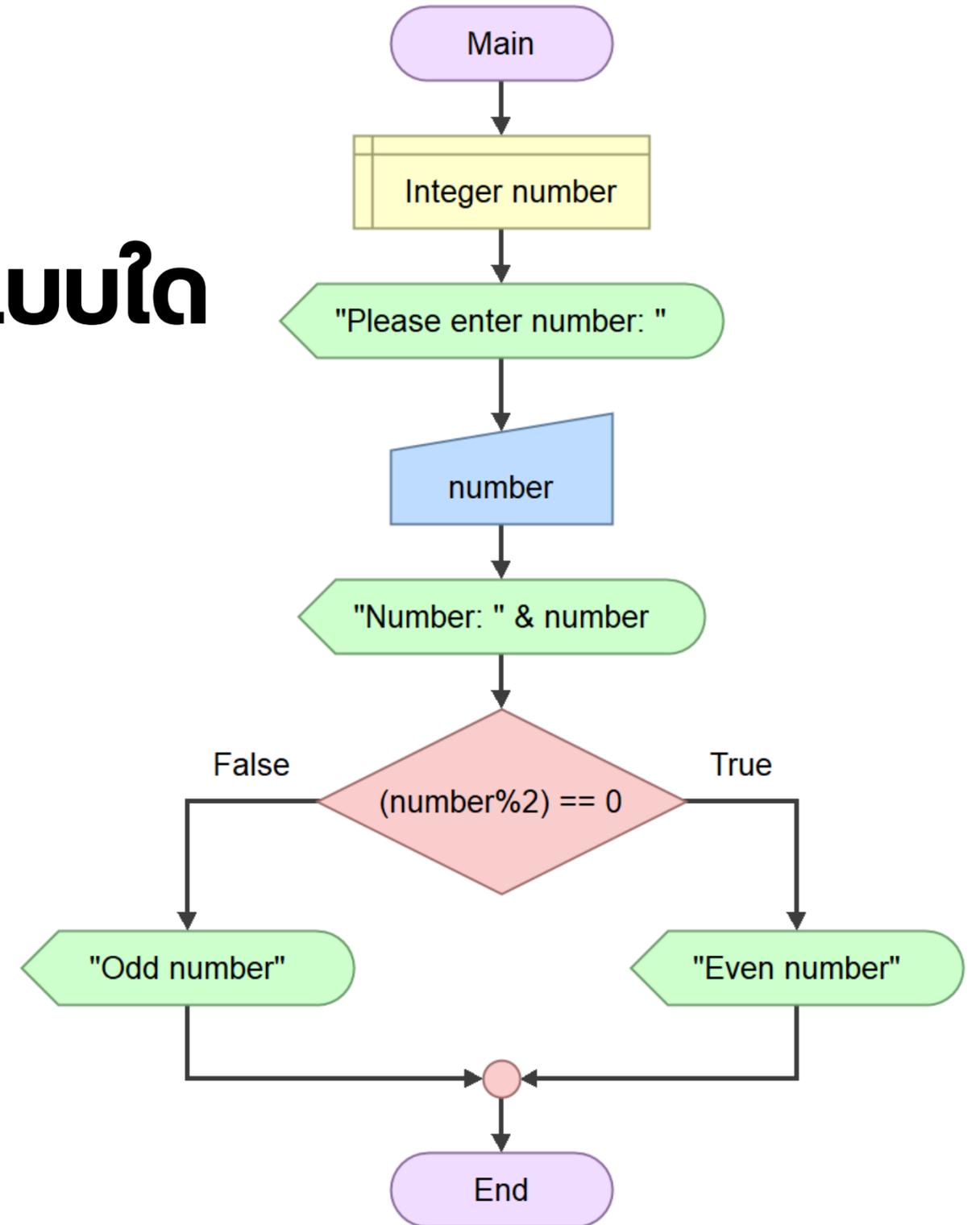
- A** Iteration Structure
- B** Sequential Structure
- C** Conditional Structure
- D** Decision Structure



**Question:**

# 6. จากภาพเป็นโครงสร้างผังงานแบบใด

- A** Iteration Structure
- B** Sequential Structure
- C** Conditional / Decision Structure
- D** ไม่มีข้อใดถูก



**Question:**

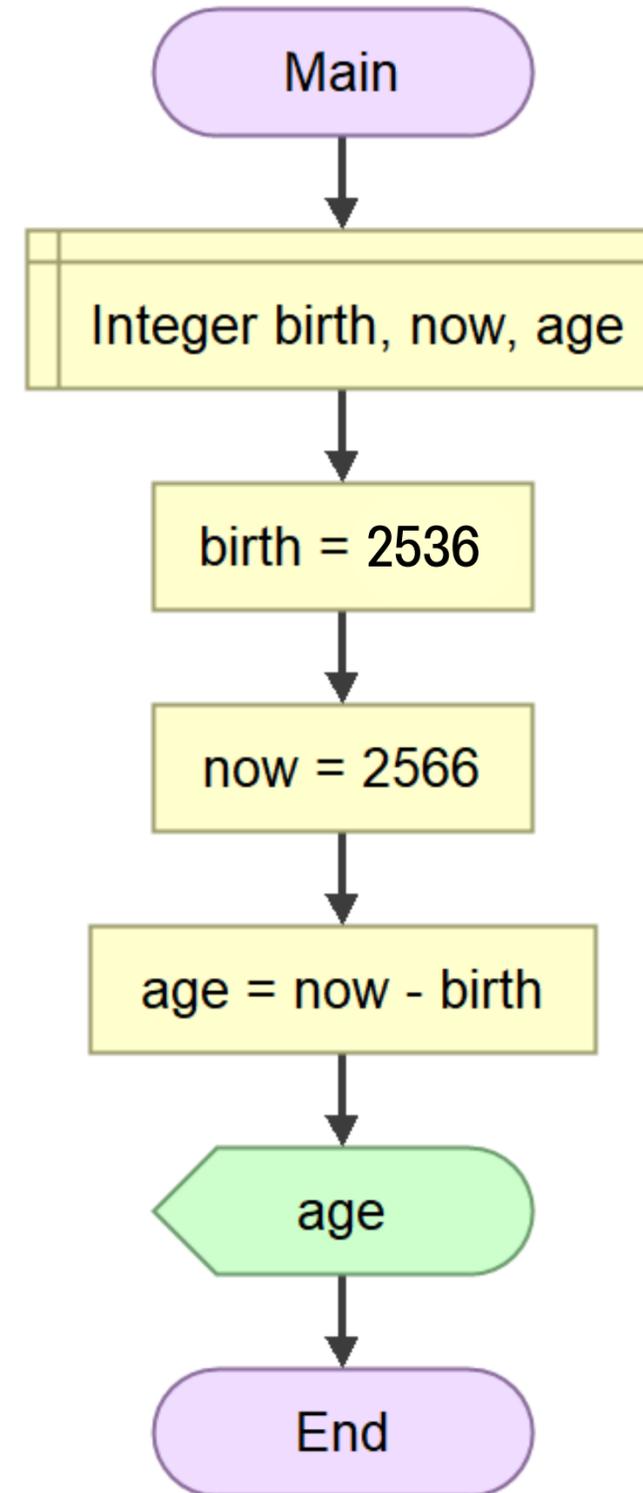
**7. จาก Algorithm นี้ผลลัพธ์ที่  
แสดงออกมา คือข้อใด**

**A** 25

**B** 27

**C** 30

**D** 33



**Question:**

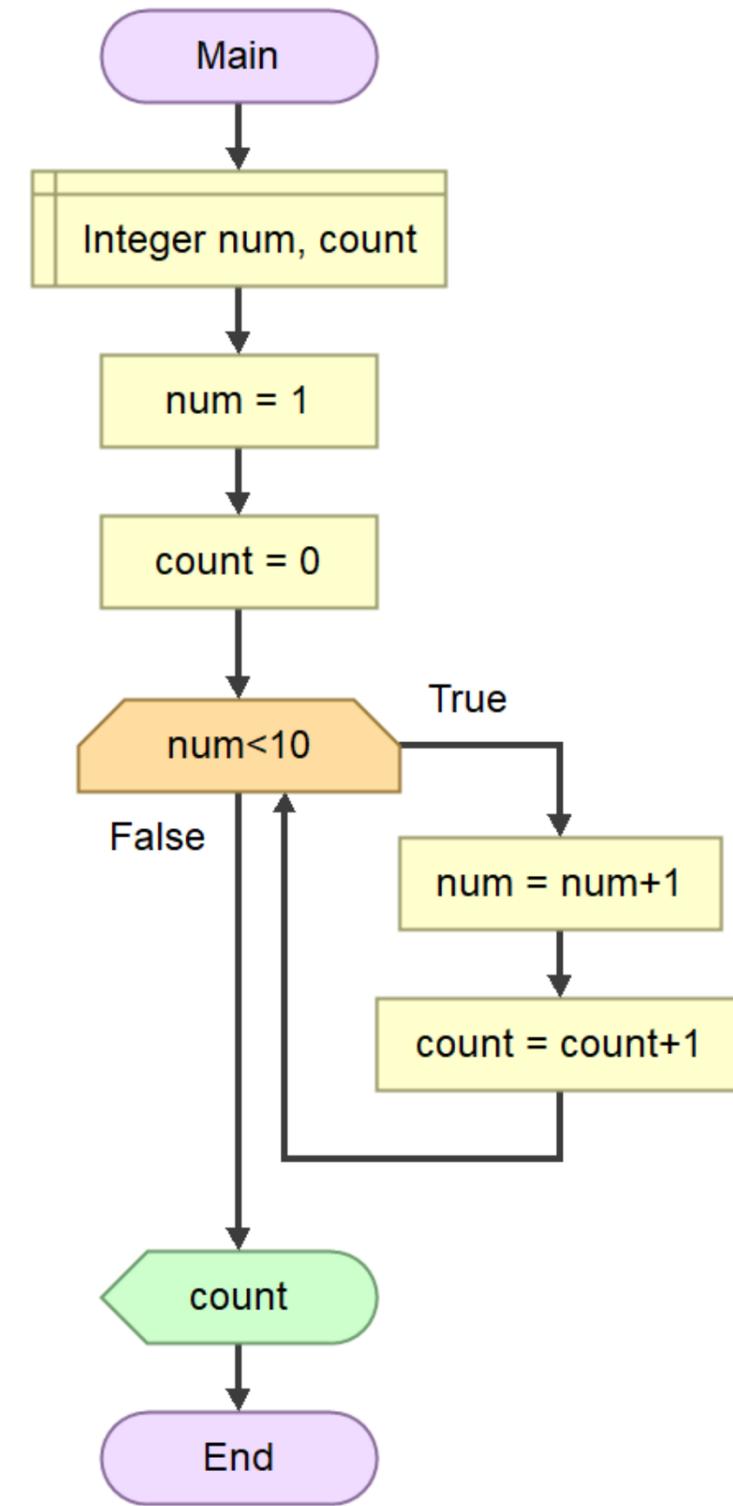
**8. จาก Algorithm นี้ผลลัพธ์ที่  
แสดงออกมา คือข้อใด**

**A** 0

**B** 9

**C** 10

**D** 11



**Question:**

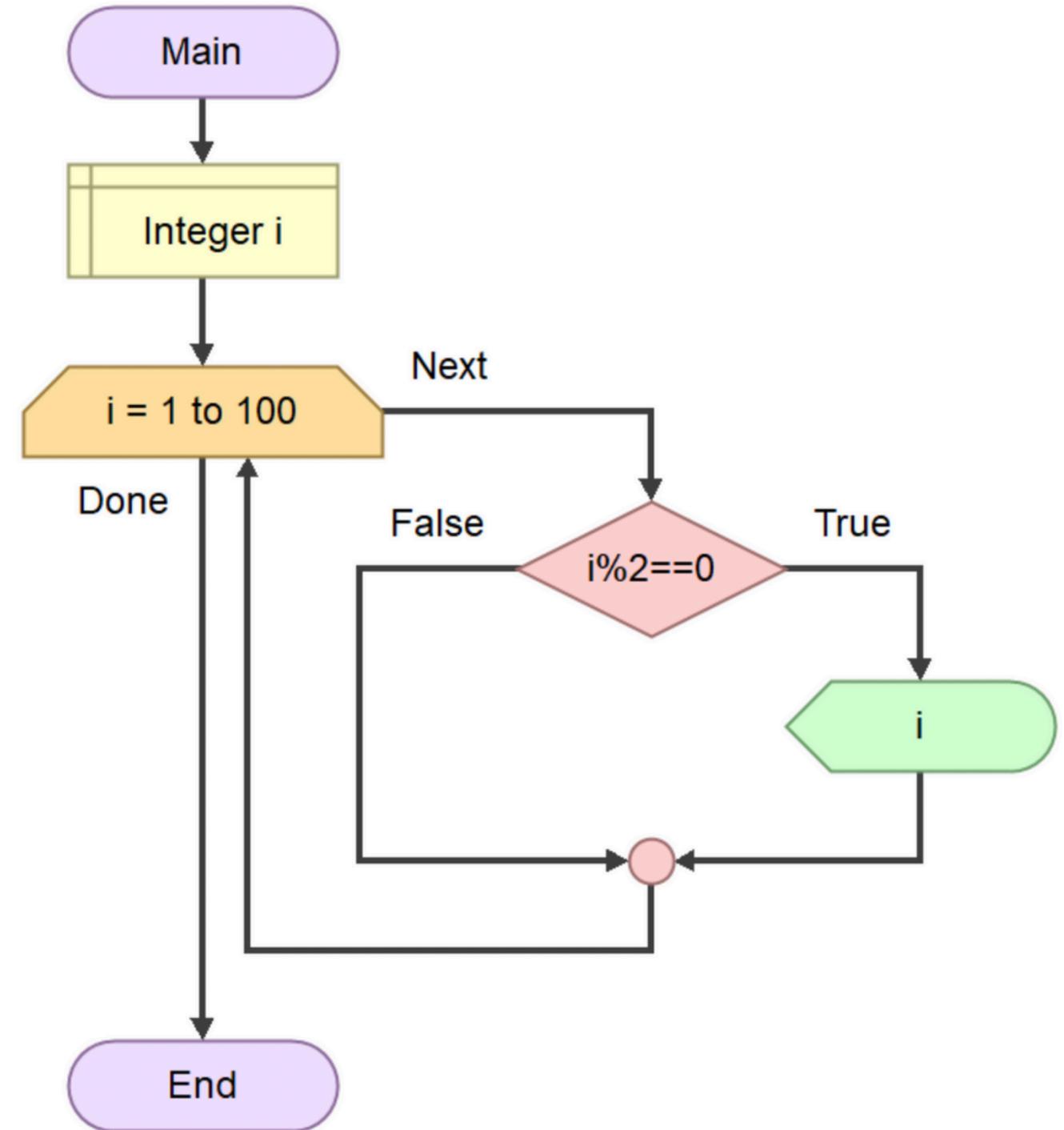
**9. จาก Algorithm นี้ผลลัพธ์ที่  
แสดงออกมา คือข้อใด**

**A** 1, 2, 3, ..., 100

**C** 2, 4, 6, ..., 98, 100

**B** 1, 2, 3, ..., 99

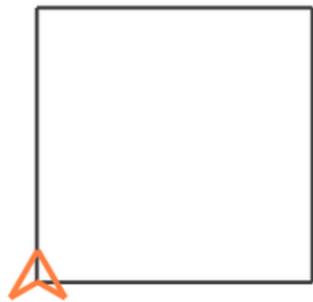
**D** 1, 3, 5, ..., 97, 99



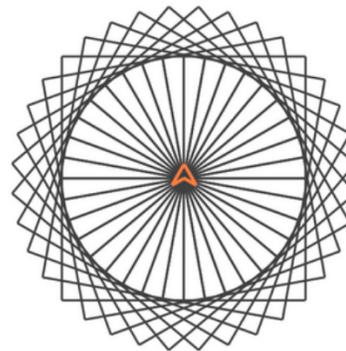
**Question:**

**10. จาก Algorithm นี้ผลลัพธ์ที่  
แสดงออกมา คือข้อใด**

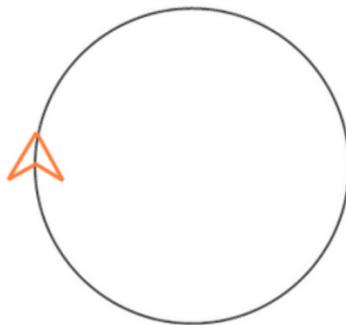
**A**



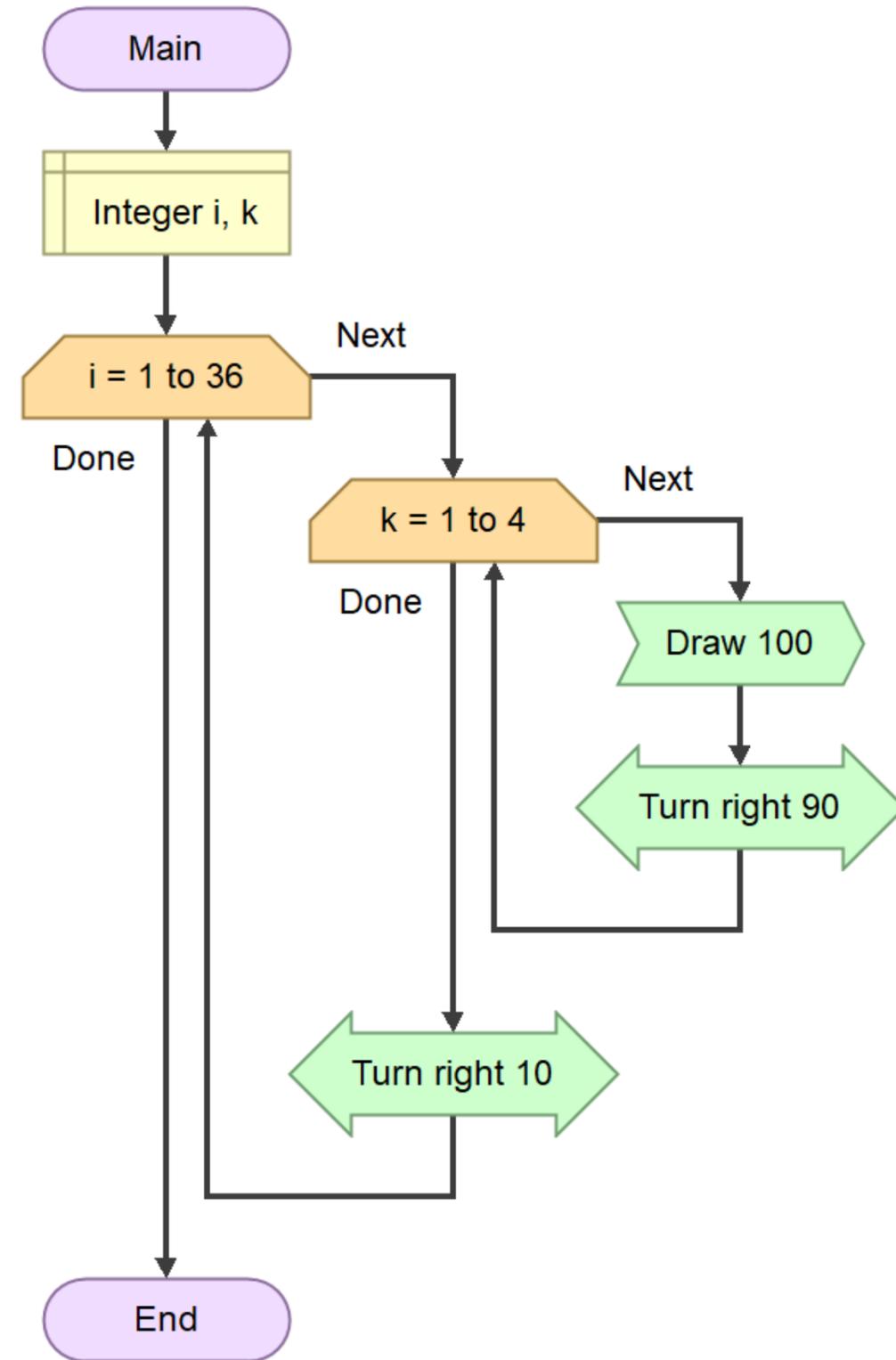
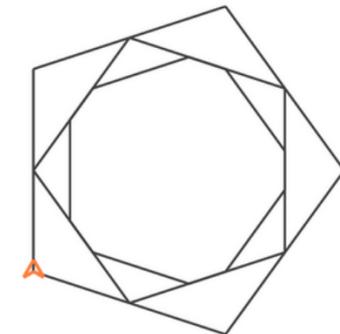
**C**



**B**



**D**



Computational Science

**Thank You**

**DTI1306 Computational Science**

Department of Digital Technology for Education  
Faculty of Education, Suan Sunandha Rajabhat University

**Content Credit By:** Asst.Prof.Nutthapat Kaewrattanapat, PhD.



**Pasawut Cheerapakorn**

Suan Sunandha Rajabhat University