

CS

พื้นฐานการวิเคราะห์และออกแบบอัลกอริทึม Fundamentals of Algorithm Design

บทที่ 4

ตอนที่ 2 การโปรแกรมแบบทำซ้ำ Iteration Statement



ผู้ช่วยศาสตราจารย์ ดร.ณัฐภัทร แก้วรัตนภัทร
Asst.Prof.Dr.Nutthapat Kaewrattanapat
Suan Sunandha Rajabhat University

รายวิชาวิทยาการคำนวณ (Computational Science) 3(2-2-5) หน่วยกิต
บรรยายนักศึกษาสาขาวิชาเทคโนโลยีดิจิทัลเพื่อการศึกษา ภาคเรียนที่ 2 ปีการศึกษา 2566
ห้องบรรยาย 1121 คณะครุศาสตร์ มหาวิทยาลัยราชภัฏสวนสุนันทา



อาจารย์บรรยาย



ผู้ช่วยศาสตราจารย์ ดร.ณัฐภัทร แก้วรัตนภัทร

การศึกษา

- 2565 ปริญญาเอก ปรัชญาดุษฎีบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศและการสื่อสารเพื่อการศึกษา (GPA. 4.00)
- 2551 ปริญญาโท วิทยาศาสตร์มหาบัณฑิต สาขาวิชาระบบสารสนเทศเพื่อการจัดการ (GPA. 3.58)
- 2549 ปริญญาตรี วิทยาศาสตร์บัณฑิต เกียรตินิยมอันดับ 1 สาขาวิชาวิทยาการคอมพิวเตอร์ (GPA. 3.66)

คุณวุฒิวิชาชีพและประกาศนียบัตร

- คุณวุฒิวิชาชีพ จาก สถาบันคุณวุฒิวิชาชีพ (องค์การมหาชน) สาขาวิชาชีพอุตสาหกรรมดิจิทัล สาขาธุรกิจดิจิทัลและพาณิชย์อิเล็กทรอนิกส์ อาชีพนักจัดการเทคโนโลยีสารสนเทศสำหรับธุรกิจ ระดับ 6 เลขที่หนังสือรับรอง PQCEN-ICT-ECM-0-251100-B-64/000029
- วิทยาศาสตร์ข้อมูลด้วยภาษาไพทอน, มหาวิทยาลัยธรรมศาสตร์
- การโปรแกรมสำหรับนักภาษาศาสตร์, มหาวิทยาลัยมหิดล
- การโปรแกรมภาษาไพทอน, มหาวิทยาลัยเพนซิลวาเนีย, สหรัฐอเมริกา
- การโปรแกรมสำหรับทุกคน, มหาวิทยาลัยมิชซิกแกน, สหรัฐอเมริกา

ติดต่อ: nutthapat.ke@ssru.ac.th

Course Description

DTI1306 วิทยาการคำนวณ (Computational Science)

3(2-2-5) บรรยาย 2 ชม ปฏิบัติ 2 ชม ศึกษาด้วยตนเอง 5 ชม

วิเคราะห์ เทคนิค วิธีการขั้นตอนการแก้ปัญหา ทักษะการคิดเชิงคำนวณ เชิงนามธรรม ฝึกทักษะในการแก้ปัญหาโดยใช้ขั้นตอนการแก้ปัญหา การย่อยปัญหา การแสดงขั้นตอน การแก้ปัญหา โดยการเขียน บอกเล่า วาดภาพ หรือใช้สัญลักษณ์ ออกแบบและเขียนโปรแกรม โดยใช้ซอฟต์แวร์หรืออุปกรณ์ 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.

Reference: <https://edu.ssru.ac.th/useruploads/files/20230724/1772131ed638786bc8d19918b37249af72c36be4.pdf>

รายวิชาวิทยาการคำนวณ Computational Science
บทที่ 4 พื้นฐานการวิเคราะห์และออกแบบอัลกอริทึม
ผู้ช่วยศาสตราจารย์ ดร.ณัฐภัทร แก้วรัตนภัทร

System Theory

Computational Thinking

Decomposition

Abstraction

Pattern Recognition

Algorithm Design

Design Thinking

Standard of Flowchart Design

Flowgorithm

Computer Programming Language

Measurement and Evaluation

การวัดและประเมินผล

1. ระหว่างการจัดการเรียนรู้

- สอบ Pre-test
- การมอบหมายงาน
- สอบ Post-test
- การมีส่วนร่วมในชั้นเรียน

0%
24%
12%
4%

2. การสอบกลางภาค (Midterm Examination)

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

20%

3. โครงการประจำภาคเรียน (Term Project)

- บทความ และการนำเสนอ

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

การมีส่วนร่วมในชั้นเรียน - ส่ง Lecture Note

ครั้งที่ 1 วันพุธที่ 6 ธันวาคม 2566

ชื่อ-สกุล:
อีเมล:

รหัสนักศึกษา:

Pretest

- | | |
|----|-----|
| 1. | 6. |
| 2. | 7. |
| 3. | 8. |
| 4. | 9. |
| 5. | 10. |

Post-test

- | | |
|----|-----|
| 1. | 6. |
| 2. | 7. |
| 3. | 8. |
| 4. | 9. |
| 5. | 10. |

สรุปเนื้อหาบรรยาย

Course Outline

- บทที่ 1 – พื้นฐานทางวิทยาการคำนวณ
- บทที่ 2 – พื้นฐานทางด้านเทคโนโลยีดิจิทัล
- บทที่ 3 – พื้นฐานทางด้านความรู้เท่าทันสื่อและดิจิทัล
- **บทที่ 4 – พื้นฐานการวิเคราะห์และออกแบบอัลกอริทึม**
- บทที่ 5 – พื้นฐานการโปรแกรมแบบ Block-based ด้วย Scratch
- บทที่ 6 – พื้นฐานการโปรแกรมไมโครคอนโทรลเลอร์เบื้องต้นด้วย Microbit
- บทที่ 7 – พื้นฐานการโปรแกรมภาษาไพทอนเบื้องต้นด้วย Colab
- บทที่ 8 – การพัฒนาโครงงานทางเทคโนโลยีดิจิทัลเพื่อการศึกษา

Measurement and Evaluation

ครั้งที่/วันที่	บทเรียน/หัวข้อ	มอบหมายงาน (24%)	สอบ Post-test (12%)	การมีส่วนร่วมในชั้นเรียน (4%)
1 พุท 6 ธค 66 Onsite	แนะนำรายวิชา การวัดและการประเมินผล หัวข้อเรียนรู้ บทที่ 1 พื้นฐานทางวิทยาการคำนวณ	-	-	ขาด/ไม่ทันเช็คชื่อ -1%
2 พุท 13 ธค 66 Onsite	บทที่ 2 พื้นฐานทางด้านเทคโนโลยีดิจิทัล	2 คะแนน Minute Paper	2 คะแนน	ขาด/ไม่ทันเช็คชื่อ -1%
3 พุท 20 ธค 66 Onsite	บทที่ 3 พื้นฐานทางด้านความรู้เท่าทันสื่อและดิจิทัล	2 คะแนน Certificate TPQI	1 คะแนน	ขาด/ไม่ทันเช็คชื่อ -1%
4 พุท 27 ธค 66 Hybrid	บทที่ 4 พื้นฐานการวิเคราะห์และออกแบบอัลกอริทึม (Flowchart, Flowgorithm และโครงสร้างการควบคุมโปรแกรมแบบเรียงลำดับและโครงสร้างการควบคุมโปรแกรมแบบตัดสั่นใจ)	3 คะแนน โปรแกรม bmi	1 คะแนน สอบในระบบ	ขาด/ไม่ทันเช็คชื่อ -1%
5 พุท 3 มค 67 Onsite	บทที่ 4 พื้นฐานการวิเคราะห์และออกแบบอัลกอริทึม (โครงสร้างการควบคุมโปรแกรมแบบทำซ้ำ)	3 คะแนน	1 คะแนน	ขาด/ไม่ทันเช็คชื่อ -1%
6 พุท 10 มค 67	บทที่ 5 การโปรแกรมแบบบล็อก (พื้นฐานการโปรแกรมด้วย Scratch)	2 คะแนน	1 คะแนน	ขาด/ไม่ทันเช็คชื่อ -1%
7 พุท 17 มค 67	บทที่ 5 การโปรแกรมแบบบล็อก (การโปรแกรมด้วย Scratch เพื่อการประยุกต์ใช้)	2 คะแนน	1 คะแนน	ขาด/ไม่ทันเช็คชื่อ -1%
8 พุท 24 มค 67	สอบกลางภาค ปรนัย 35 ข้อ (35 คะแนน) และอัตนัย 1 ข้อ (5 คะแนน) 20%			

Measurement and Evaluation

ครั้งที่/วันที่	บทเรียน/หัวข้อ	มอบหมายงาน (24%)	สอบ Post-test (12%)	การมีส่วนร่วมในชั้นเรียน (4%)
9 พุท 7 กพ 67	บทที่ 6 การโปรแกรมโมโครคอนโทรลเลอร์เบื้องต้นด้วย Microbit	2 คะแนน	1 คะแนน	ขาด/ไม่ทันเช็คชื่อ -1%
10 พุท 14 กพ 67	บทที่ 6 การโปรแกรมโมโครคอนโทรลเลอร์เบื้องต้นด้วย Microbit (การเชื่อมต่อกับเซ็นเซอร์)	2 คะแนน	1 คะแนน	ขาด/ไม่ทันเช็คชื่อ -1%
12 พุท 21 กพ 67	บทที่ 7 การโปรแกรมภาษาไพทอนเบื้องต้นด้วย Colab (Basic Input/Output, Variables, Operation)	2 คะแนน	1 คะแนน	ขาด/ไม่ทันเช็คชื่อ -1%
13 พุท 28 กพ 67	บทที่ 7 การโปรแกรมภาษาไพทอนเบื้องต้นด้วย Colab (Decision and Iteration Statement)	2 คะแนน	1 คะแนน	ขาด/ไม่ทันเช็คชื่อ -1%
14 พุท 6 มีค 67	บทที่ 8 การพัฒนาโครงการทางเทคโนโลยีดิจิทัลเพื่อการศึกษา (การคิดเชิงออกแบบ, กระบวนการทางโครงการ)	2 คะแนน	1 คะแนน	ขาด/ไม่ทันเช็คชื่อ -1%
15 พุท 13 มีค 67	สอบปลายภาค ปรนัย 35 ข้อ (35 คะแนน) และอัตนัย 1 ข้อ (5 คะแนน) 20%			
16 พุท 20 มีค 67	ส่งบทความ (โครงการ) และนำเสนอ 20%			
17 พุท 27 มีค 67	แผนการจัดการเรียนรู้และสื่อการเรียนรู้ด้านวิทยาการคำนวณ			

Pretest

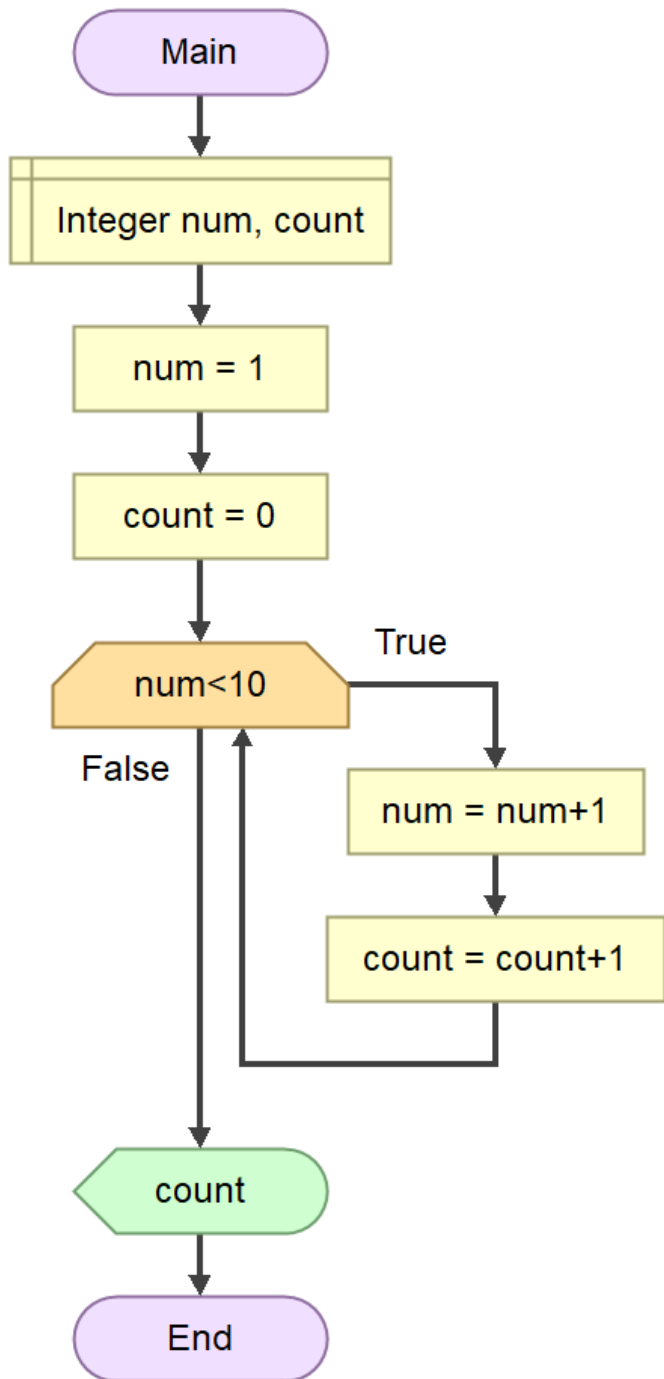


cutt.ly/cspretest5

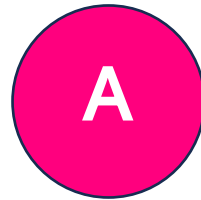
Post-test



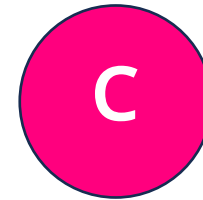
cutt.ly/csposttest5



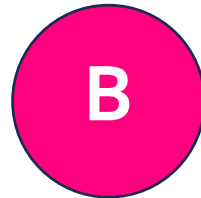
1. จาก Algorithm นี้ผลลัพธ์ที่แสดงออกมาคือข้อใด



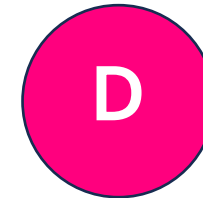
0



10

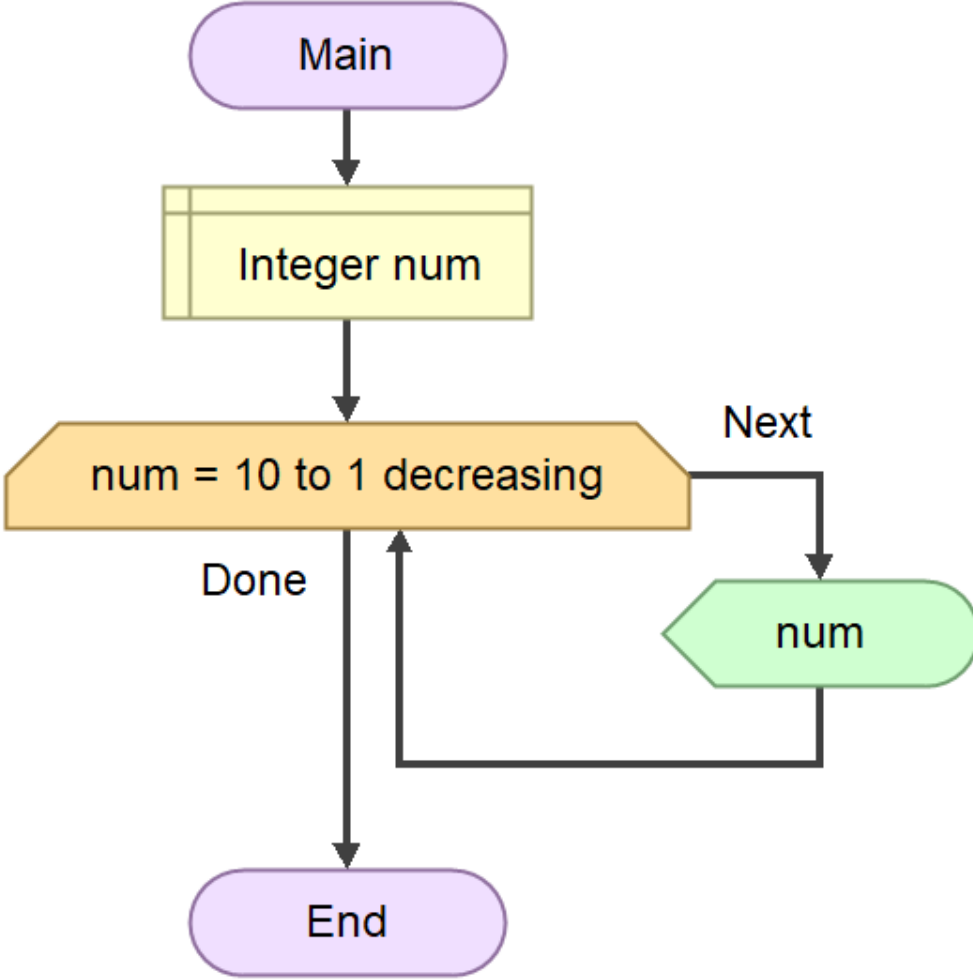


9



11

2. จาก Algorithm นี้ผลลัพธ์ที่แสดงออกมา คือข้อใด



A

0,...,10

C

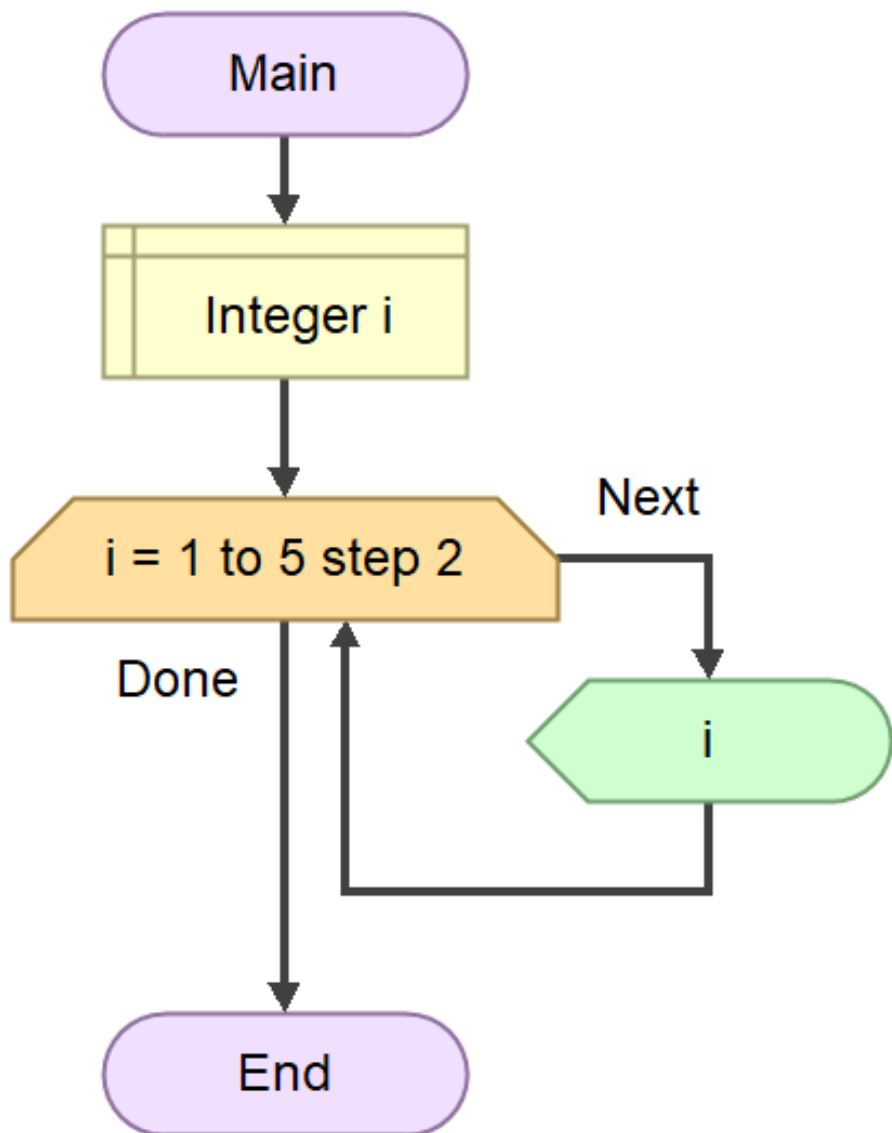
10,...,0

B

1,...,10

D

10,...,1



3. จาก Algorithm นี้ผลลัพธ์ที่แสดงออกมา คือข้อใด

A

1,2,3,4,5

C

1,3,5

B

2,3,4,5

D

1,2,4

4. จาก Algorithm นี้ผลลัพธ์ที่แสดงออกมา คือข้อใด

A

1,2,3,4,5

C

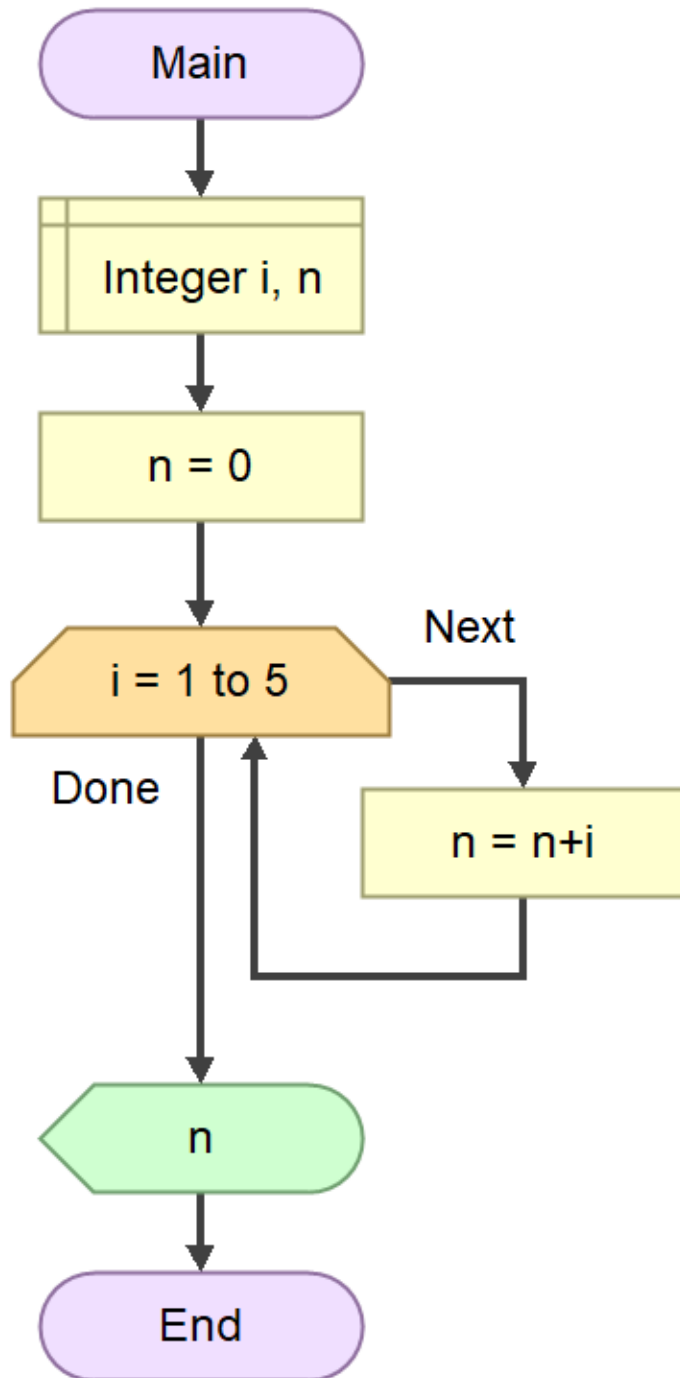
15

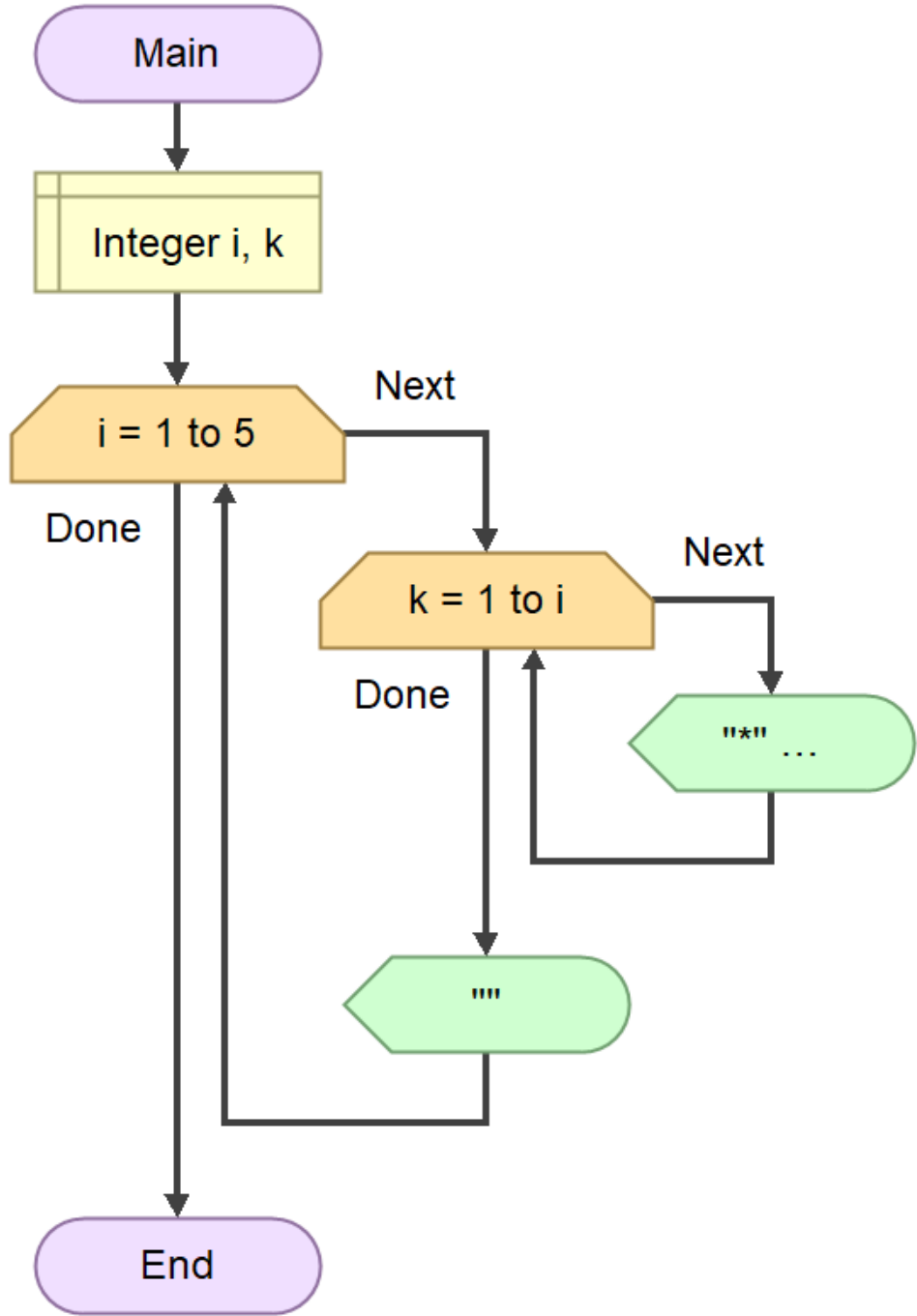
B

5

D

55





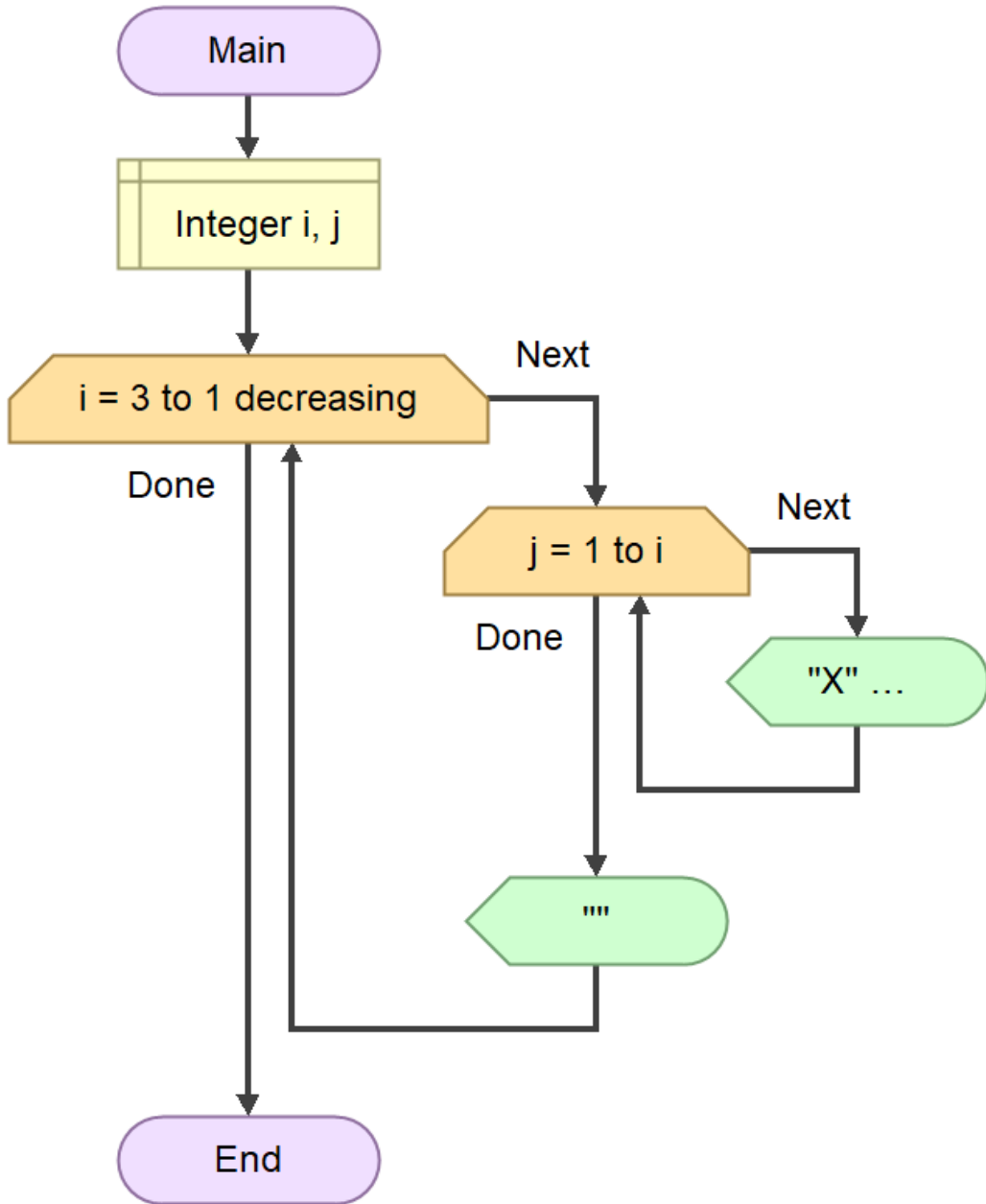
5. จาก Algorithm นี้ผลลัพธ์ที่แสดงออกมา คือข้อใด

A * , ** , ***

C ***
**
*

B *
**

D * ** ***



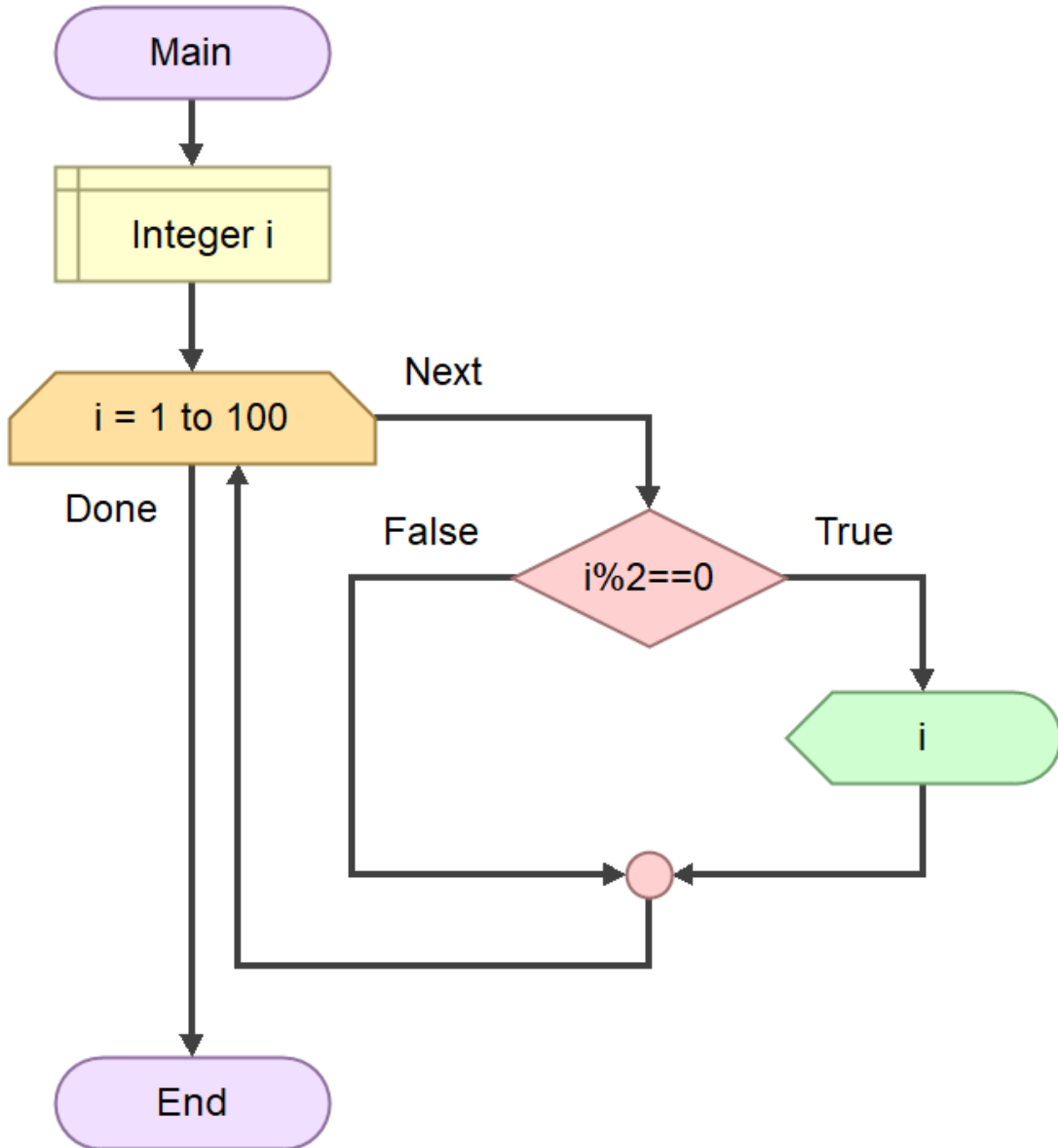
6. จาก Algorithm นี้ผลลัพธ์ที่แสดงออกมา คือข้อใด

A X, XX, XXX

C XXX
XX
X

B X
XX
XXX

D X XX XXX



7. จาก Algorithm นี้ผลลัพธ์ที่แสดงออกมา คือข้อใด

A

1,2,...,100

C

2,4,...,98,100

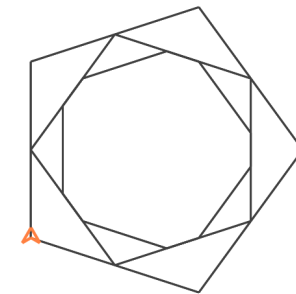
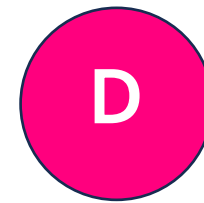
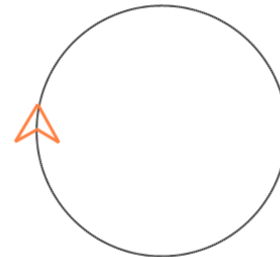
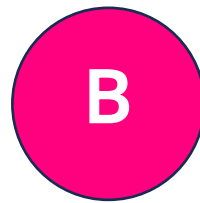
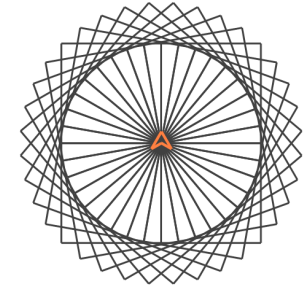
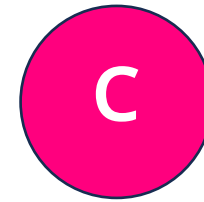
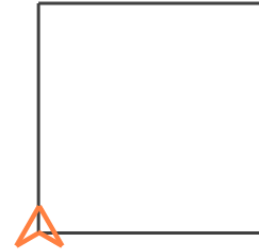
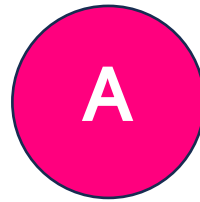
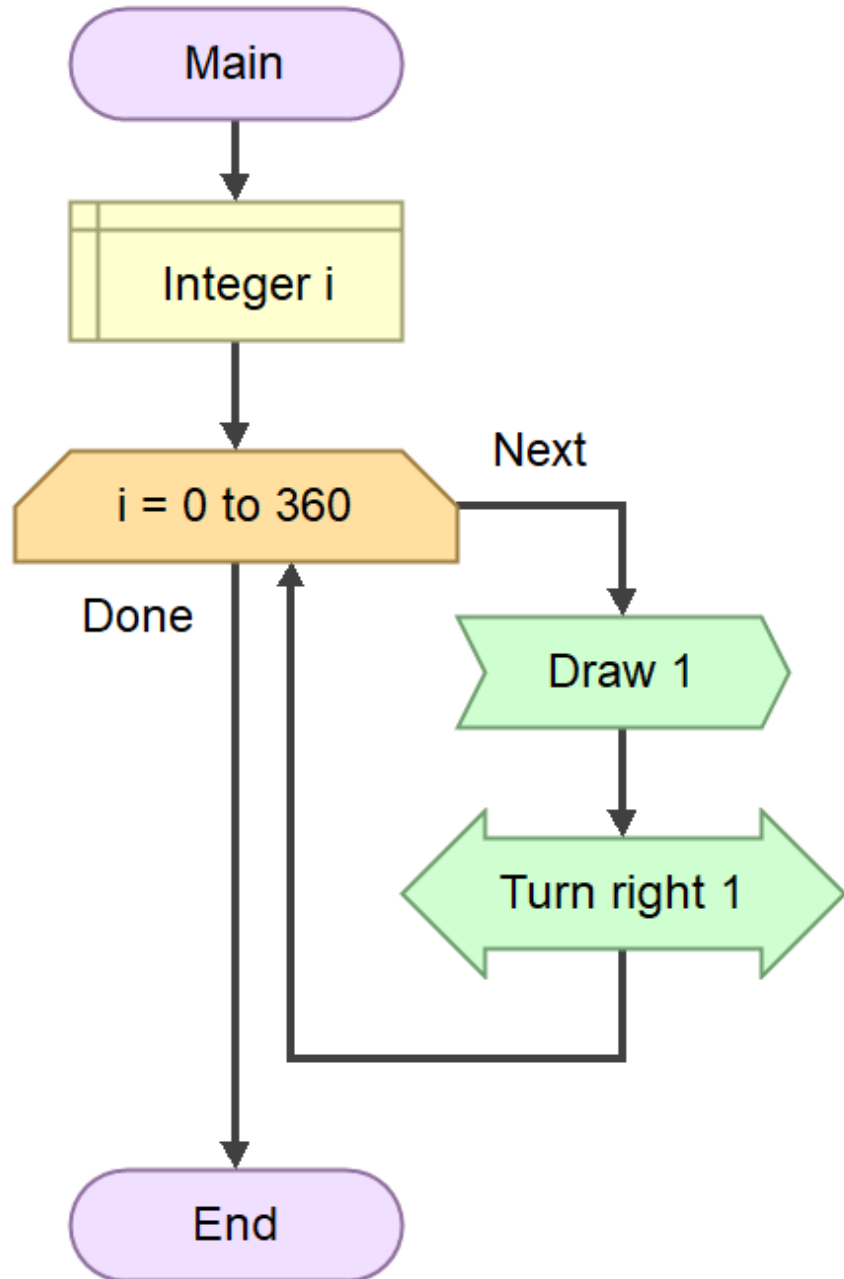
B

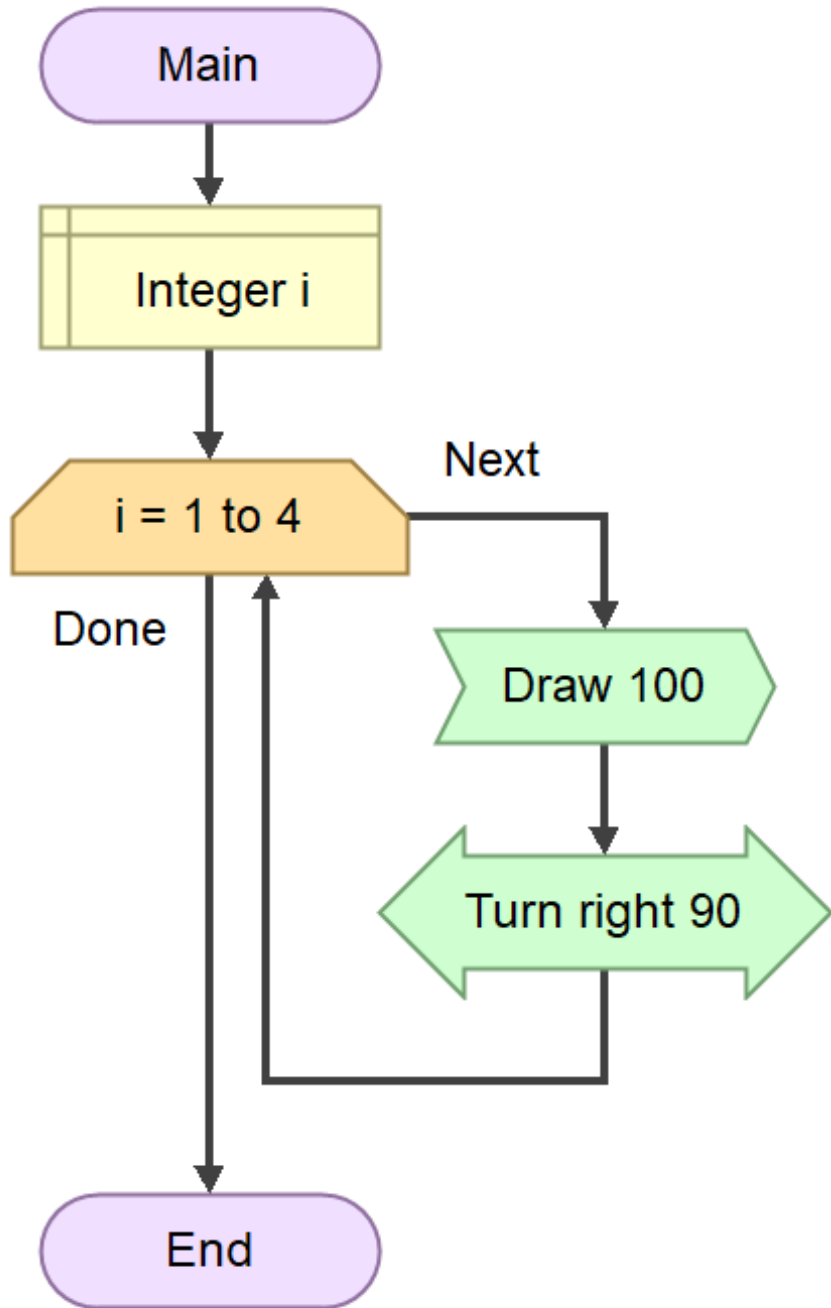
1,2,.....,99

D

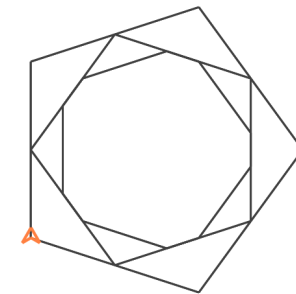
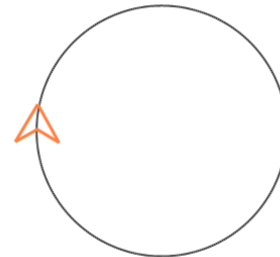
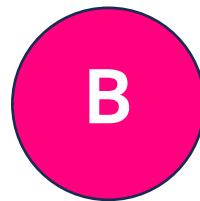
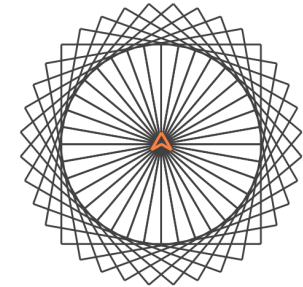
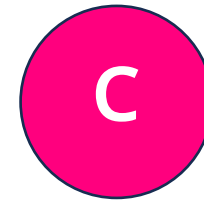
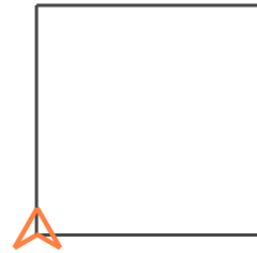
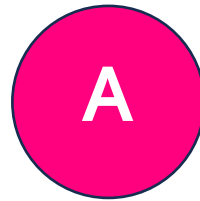
1,3,...,97,99

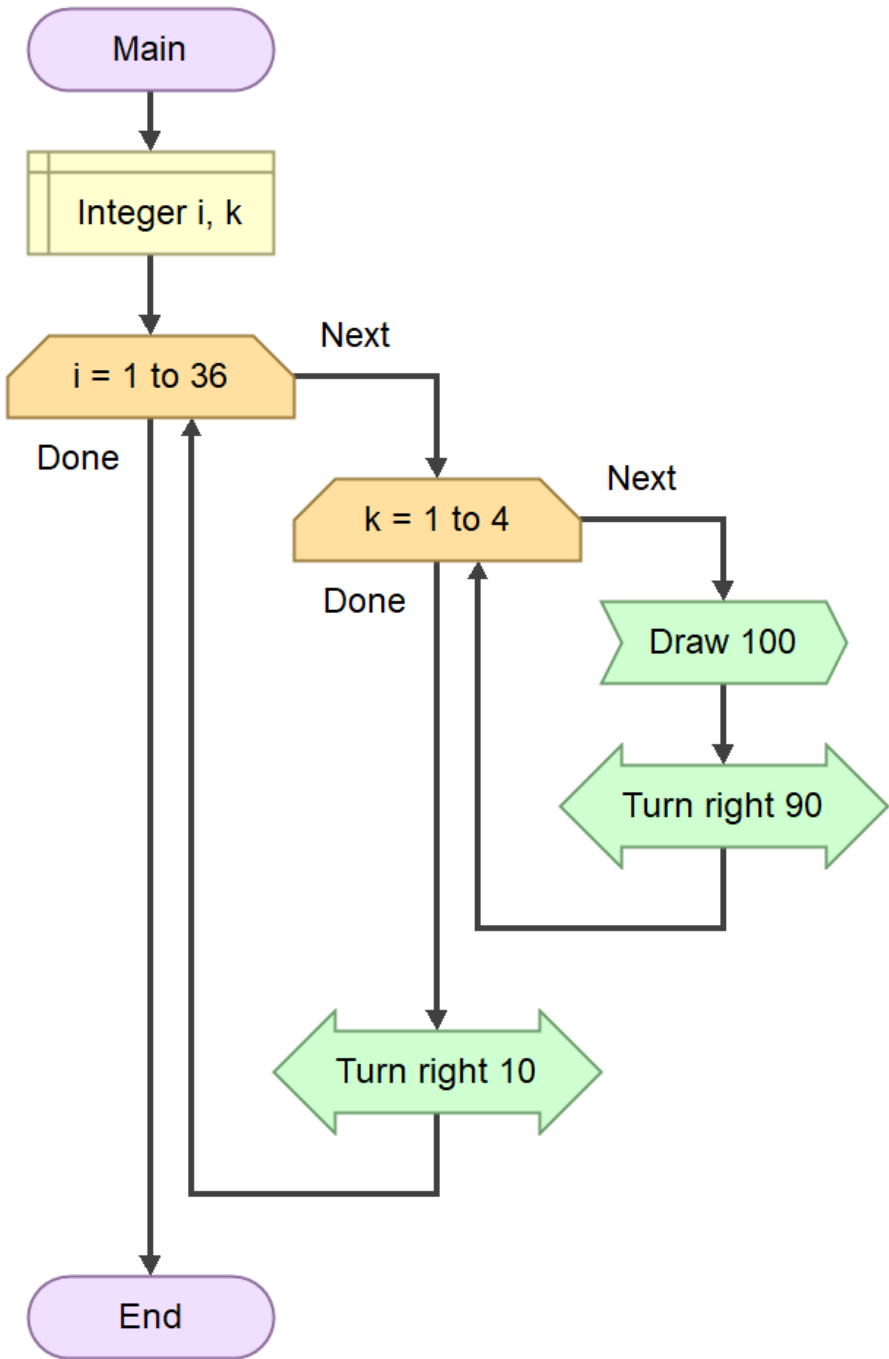
8. ข้อใดคือผลลัพธ์จาก Algorithm



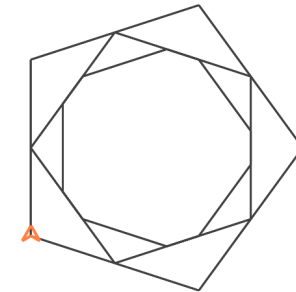
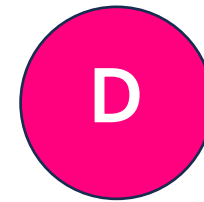
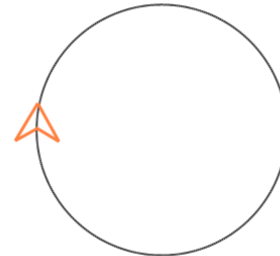
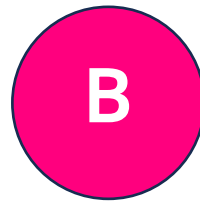
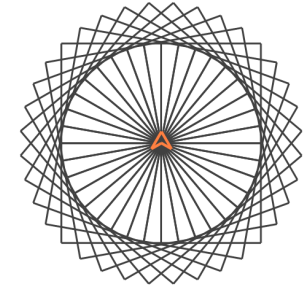
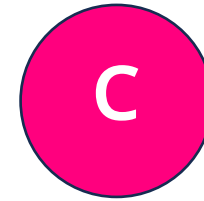
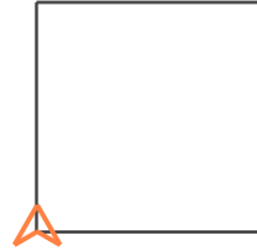
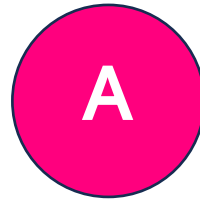


9. ข้อใดคือผลลัพธ์จาก Algorithm





10. ข้อใดคือผลลัพธ์จาก Algorithm



อัลกอริทึม (Algorithm)

อัลกอริทึม (Algorithm) คือ ชุดของขั้นตอนและกฎกติกา ที่ถูกกำหนดมาเพื่อแก้ปัญหาหรือดำเนิน กระบวนการที่สามารถทำงานได้ ด้วยมนุษย์ คอมพิวเตอร์หรือเครื่องจักรอัตโนมัติ

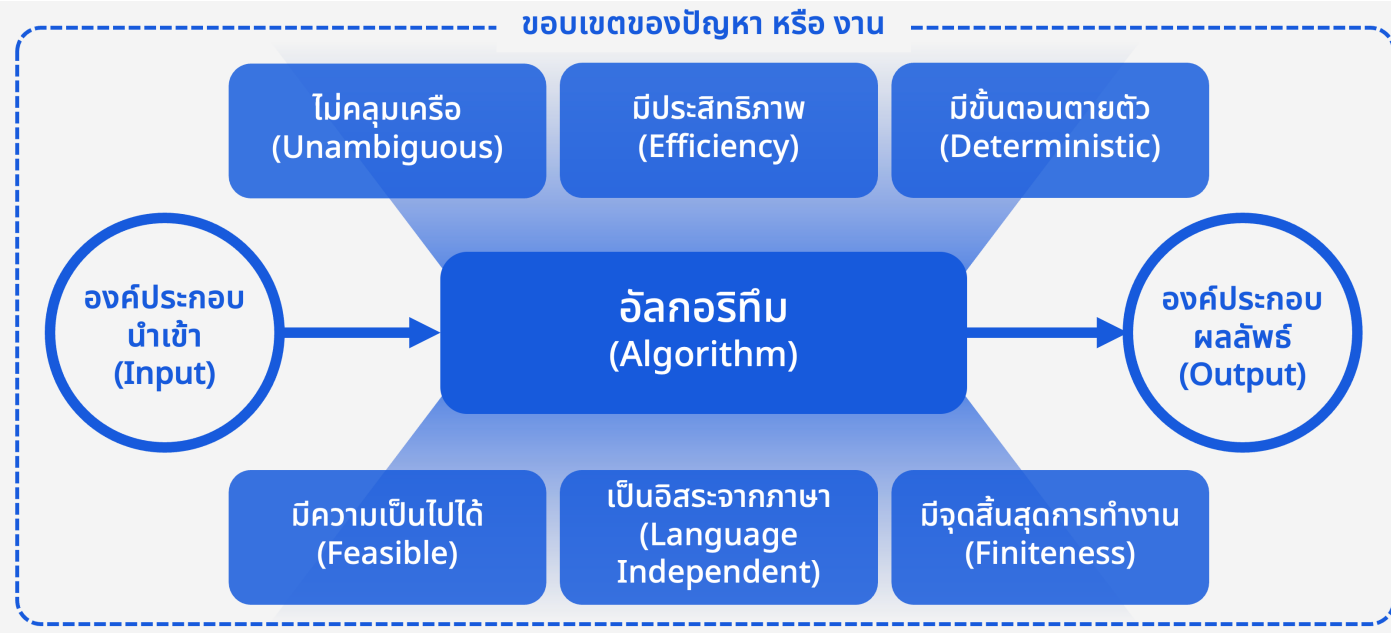
ซึ่งขั้นตอน (Algorithm) เป็นการปฏิบัติตามลำดับและหลักการที่แน่นอน เพื่อให้ได้ผลลัพธ์ที่ถูกต้องและเชื่อถือได้ทุกครั้งที่ทำซ้ำ หรือ ได้ผลลัพธ์ภายใต้เงื่อนไขที่เหมือนกันเสมอ

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. Sedgwick, 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)



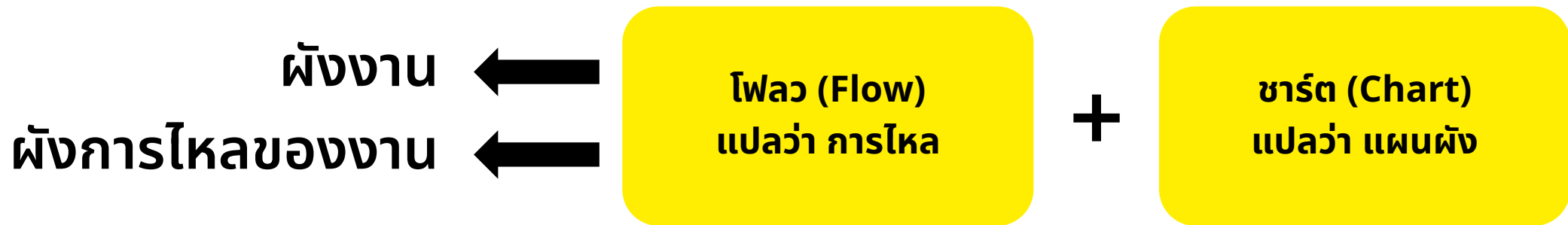
อ้างอิง

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.

- ลักษณะของอัลกอริทึมที่ดีที่ประกอบด้วยคุณสมบัติ ดังนี้
1. **ไม่คลุมเครือ (Unambiguous)** อัลกอริทึมต้องไม่มีความกำกวมที่ไม่ชัดเจน และต้องมีชุดคำสั่งที่เข้าใจง่ายและไม่ทำให้เกิดความสับสนในการประมวลผล
 2. **มีประสิทธิภาพ (Efficiency)** อัลกอริทึมควรมีประสิทธิภาพในการทำงาน คือต้องให้ผลลัพธ์ที่ถูกต้องและใช้เวลาน้อยที่สุดในการแก้ปัญหา
 3. **มีขั้นตอนตายตัว (Deterministic)** อัลกอริทึมต้องเป็น deterministic หมายความว่า จะต้องให้ผลลัพธ์ที่เหมือนกันสำหรับข้อมูลนำเข้าที่เหมือนกันทุกครั้งที่รับ
 4. **มีความเป็นไปได้ (Feasible)** อัลกอริทึมต้องเป็นไปได้ในการนำไปใช้งานจริง สามารถทำงานบนเครื่องคอมพิวเตอร์หรือสภาพแวดล้อมที่เหมาะสม
 5. **เป็นอิสระจากภาษา (Language Independent)** อัลกอริทึมควรไม่เกี่ยวข้องกับภาษาเฉพาะเสียเพียงอย่างเดียว และสามารถนำไปใช้งานในหลายภาษาโปรแกรมได้
 6. **มีจุดสิ้นสุดการทำงาน (Finiteness)** อัลกอริทึมต้องสามารถสิ้นสุดการทำงานหลังจากทำขั้นตอนทุกขั้นตอนเสร็จสมบูรณ์ และไม่ต้องเข้าสู่วงจรไม่สิ้นสุด (Infinity Loop)

อัลกอริทึม (Algorithm)

การแสดงอัลกอริทึมด้วยผังงาน หรือ โฟลวชาร์ต (Flowchart)

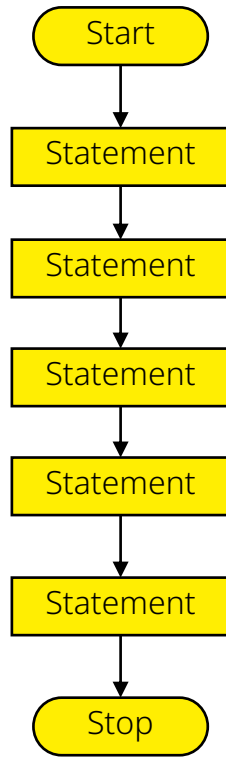


เมื่อรวมกันแล้ว จึงหมายถึง แผนผังที่แสดงการไหลของงาน หรือ แผนผังที่แสดงขั้นตอนการทำงานอย่างชัดเจน นั่นเอง ซึ่งสามารถนำผังงานมาใช้แสดงขั้นตอนการทำงานของโปรแกรมได้

โครงสร้างการควบคุมทิศทางอัลกอริทึม (Control Flow in Algorithm)

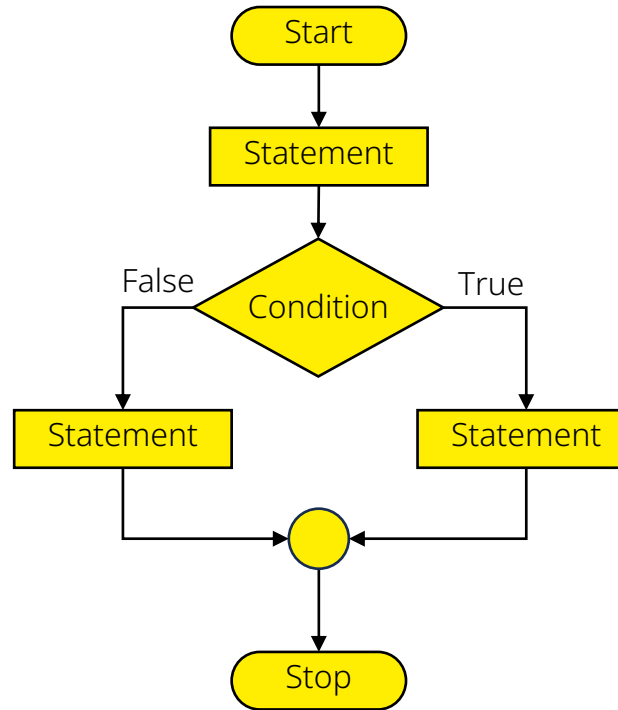
โครงสร้างการควบคุมแบบลำดับ Sequential Control Structure

โครงสร้างการควบคุมแบบบนลงล่าง
Top-Down Control Structure



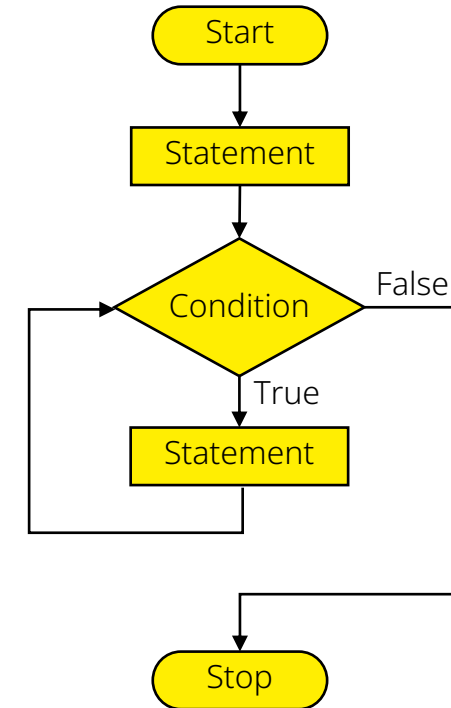
โครงสร้างการควบคุมแบบทางเลือก Selection Control Structure

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

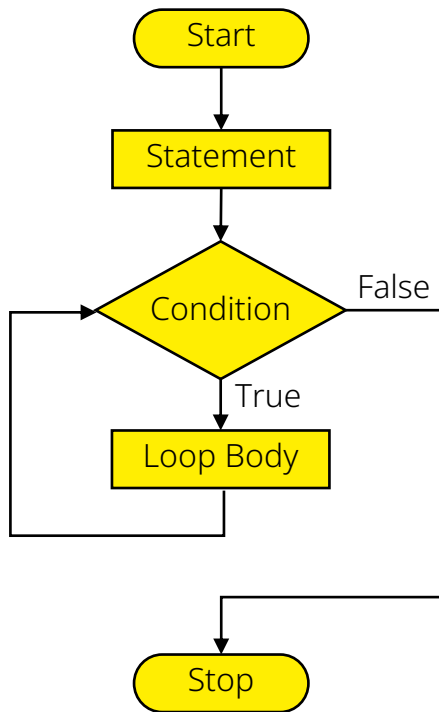
โครงสร้างนี้คล้ายกับโครงสร้างแบบ while แต่โปรแกรมจะวนซ้ำก่อนตรวจสอบเงื่อนไข หากเงื่อนไขเป็นเท็จ โปรแกรมจะวนซ้ำต่อไป หากเงื่อนไขเป็นจริง โปรแกรมจะออกจาก การวนซ้ำ

3. โครงสร้างแบบวนซ้ำแบบ for

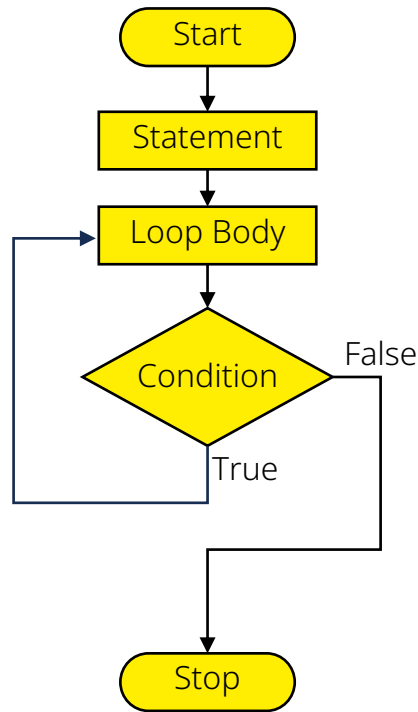
โครงสร้างนี้ใช้กำหนดจำนวนรอบในการวนซ้ำไว้ล่วงหน้า โดยระบุจำนวนรอบในตัวแปร

โครงสร้างการควบคุมแบบทำซ้ำ (Iteration Control Structure)

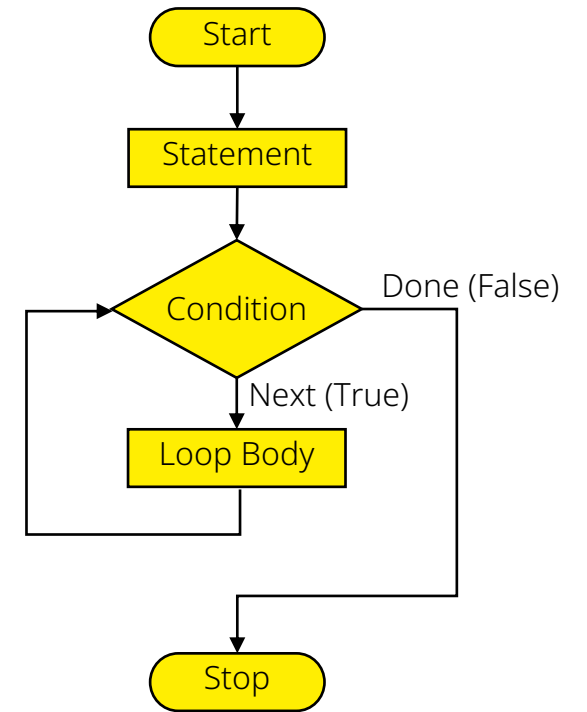
โครงสร้างการควบคุมทำซ้ำแบบ while
While Statement



โครงสร้างการควบคุมทำซ้ำแบบ do-while
Do-while Statement



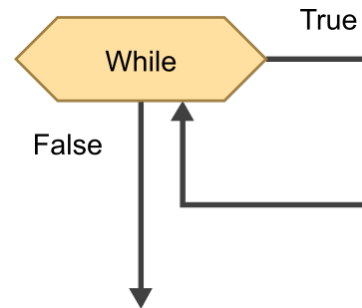
โครงสร้างการควบคุมทำซ้ำแบบ for
For Statement



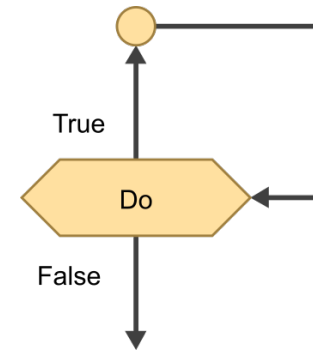
โครงสร้างการทำซ้ำใน Flowgorithm

Clipboard		Miscellaneous	
Statement			
Input / Output	Variables	Control	Looping
Turtle Graphics		Files	

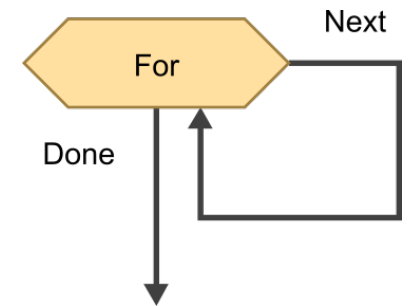
โครงสร้างการควบคุมทำซ้ำ
แบบ while
While Statement



โครงสร้างการควบคุมทำซ้ำ
แบบ do-while
Do-while Statement



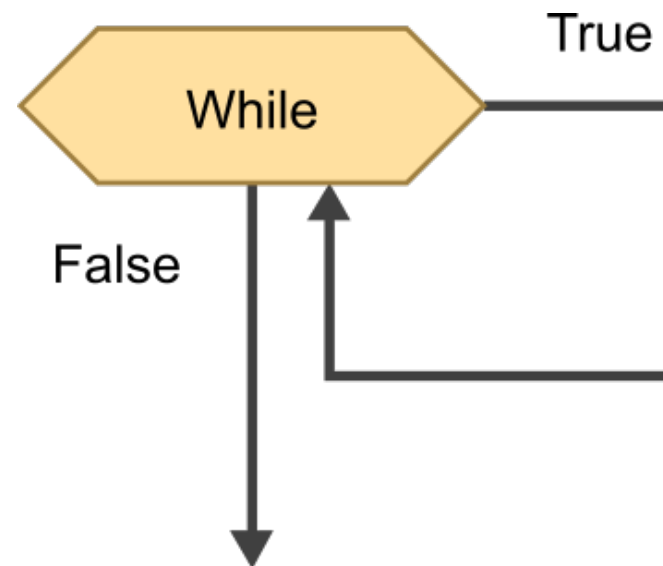
โครงสร้างการควบคุมทำซ้ำ
แบบ for
For Statement



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



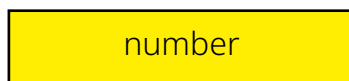
โครงสร้างการทำซ้ำแบบ while

สถานการณ์ปัญหา:

ต้องการแสดงค่า 1 ถึง 10 ออกทางหน้าจอ
ด้วยการวนซ้ำแบบ while

วิเคราะห์ปัจจัยนำเข้า (Input):

ตัวแปรที่ช่วยเก็บค่าตั้งแต่ 1 ถึง 10



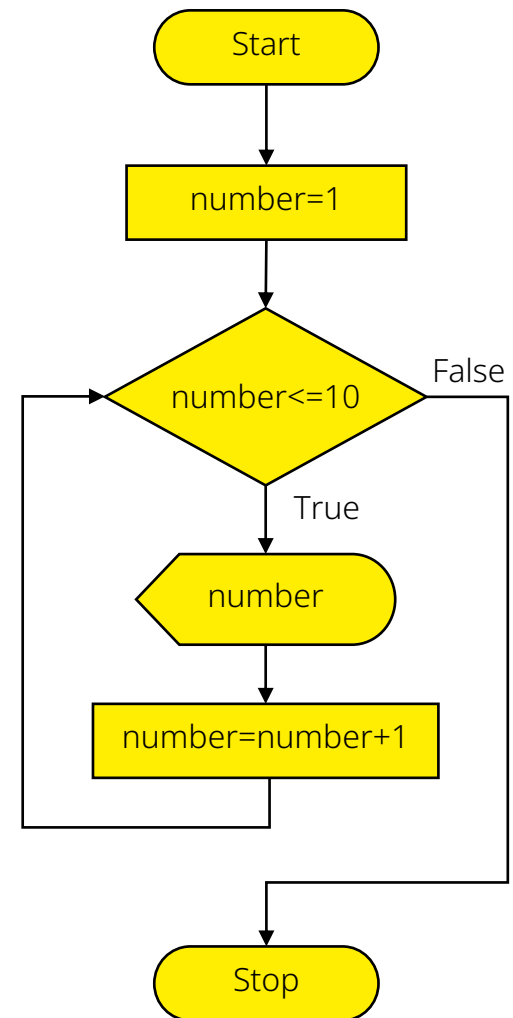
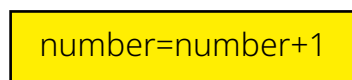
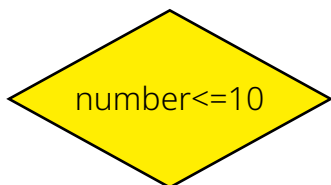
วิเคราะห์ปัจจัยผลลัพธ์ (Output):

แสดงค่าในตัวแปร



วิเคราะห์ปัจจัยประมวลผล (Process):

ทำการเพิ่มจำนวนในตัวแปรรอบละ +1



โครงสร้างการทำซ้ำแบบ while - code8

สถานการณ์ปัญหา:

ต้องการแสดงค่า 1 ถึง 10 ออกทางหน้าจอ
ด้วยการวนซ้ำแบบ while

วิเคราะห์ปัจจัยนำเข้า (Input):

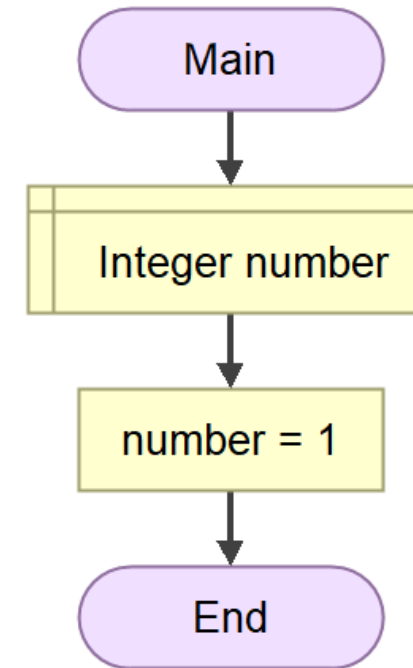
ตัวแปรที่ช่วยเก็บค่าตั้งแต่ 1 ถึง 10

วิเคราะห์ปัจจัยผลลัพธ์ (Output):

แสดงค่าในตัวแปร

วิเคราะห์ปัจจัยประมวลผล (Process):

ทำการเพิ่มจำนวนในตัวแปรรอบละ +1



โครงสร้างการทำซ้ำแบบ while - code8

สถานการณ์ปัญหา:

ต้องการแสดงค่า 1 ถึง 10 ออกทางหน้าจอ
ด้วยการวนซ้ำแบบ while

วิเคราะห์ปัจจัยนำเข้า (Input):

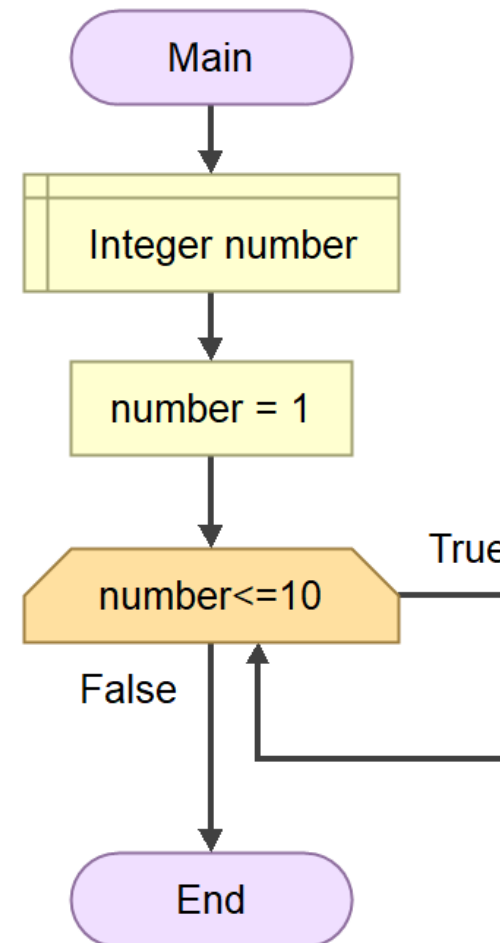
ตัวแปรที่ช่วยเก็บค่าตั้งแต่ 1 ถึง 10

วิเคราะห์ปัจจัยผลลัพธ์ (Output):

แสดงค่าในตัวแปร

วิเคราะห์ปัจจัยประมวลผล (Process):

ทำการเพิ่มจำนวนในตัวแปรรอบละ +1



โครงสร้างการทำซ้ำแบบ while - code8

สถานการณ์ปัญหา:

ต้องการแสดงค่า 1 ถึง 10 ออกทางหน้าจอ
ด้วยการวนซ้ำแบบ while

วิเคราะห์ปัจจัยนำเข้า (Input):

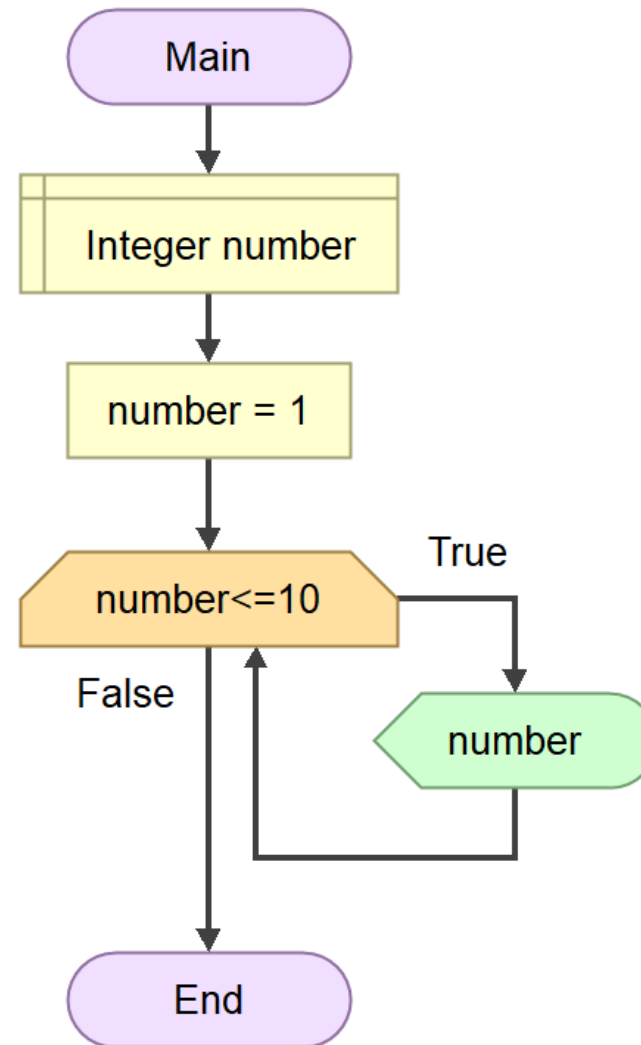
ตัวแปรที่ช่วยเก็บค่าตั้งแต่ 1 ถึง 10

วิเคราะห์ปัจจัยผลลัพธ์ (Output):

แสดงค่าในตัวแปร

วิเคราะห์ปัจจัยประมวลผล (Process):

ทำการเพิ่มจำนวนในตัวแปรรอบละ +1



โครงสร้างการทำซ้ำแบบ while - code8

สถานการณ์ปัญหา:

ต้องการแสดงค่า 1 ถึง 10 ออกทางหน้าจอ
ด้วยการวนซ้ำแบบ while

วิเคราะห์ปัจจัยนำเข้า (Input):

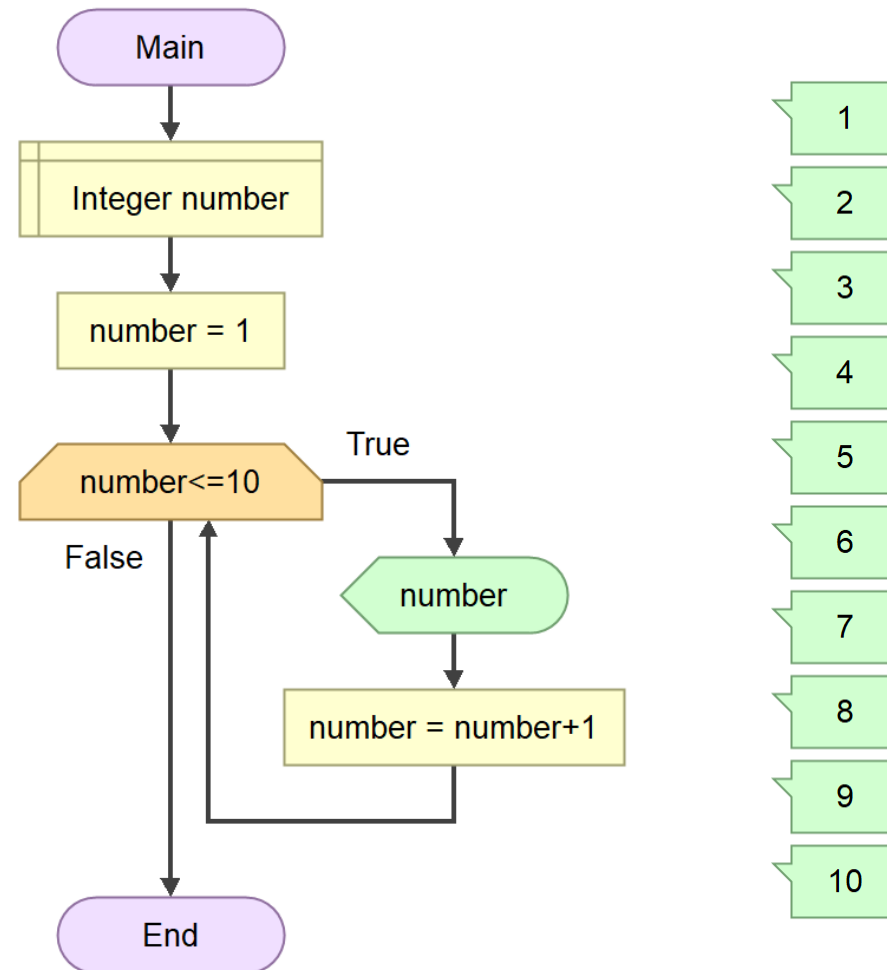
ตัวแปรที่ช่วยเก็บค่าตั้งแต่ 1 ถึง 10

วิเคราะห์ปัจจัยผลลัพธ์ (Output):

แสดงค่าในตัวแปร

วิเคราะห์ปัจจัยประมวลผล (Process):

ทำการเพิ่มจำนวนในตัวแปรรอบละ +1

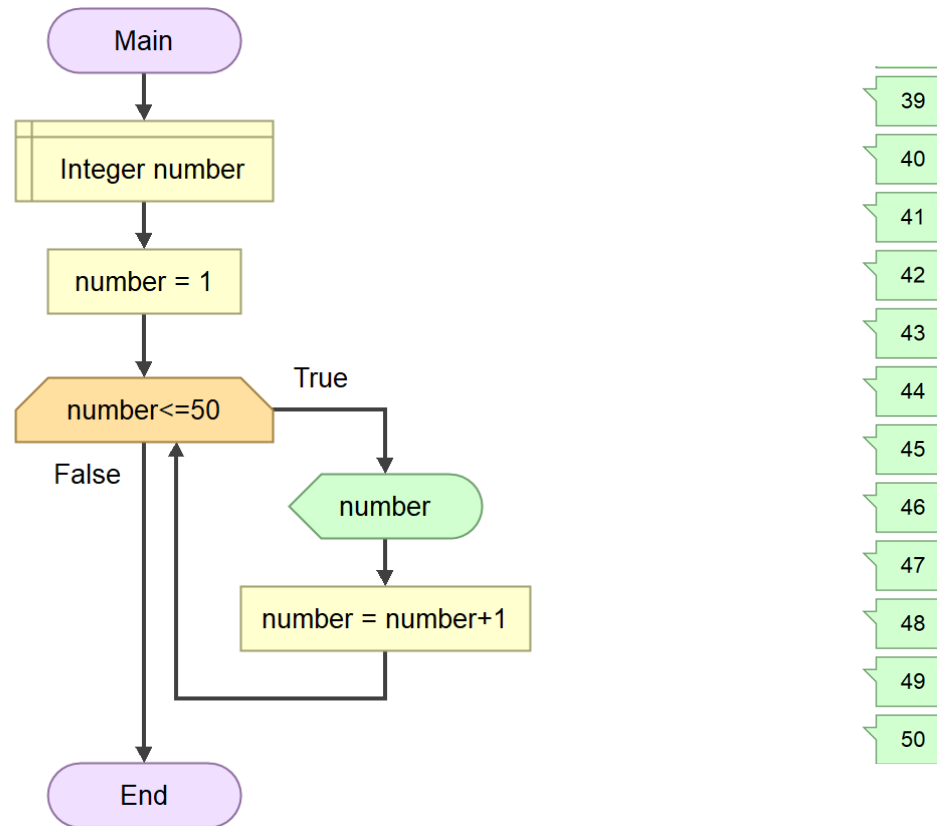


โครงสร้างการทำซ้ำแบบ while - code9

สถานการณ์ปัญหา: แสดงจำนวนตั้งแต่ 1 ไปเรื่อยๆ แต่ห้ามเกิน 50 โดยใช้ while loop

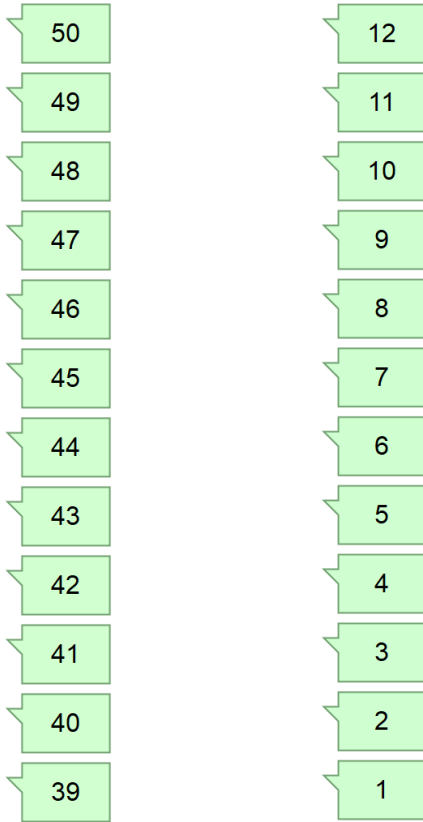
โครงสร้างการทำซ้ำแบบ while - code9

สถานการณ์ปัญหา: แสดงจำนวนตั้งแต่ 1 ไปเรื่อยๆ แต่ห้ามเกิน 50 โดยใช้ while loop



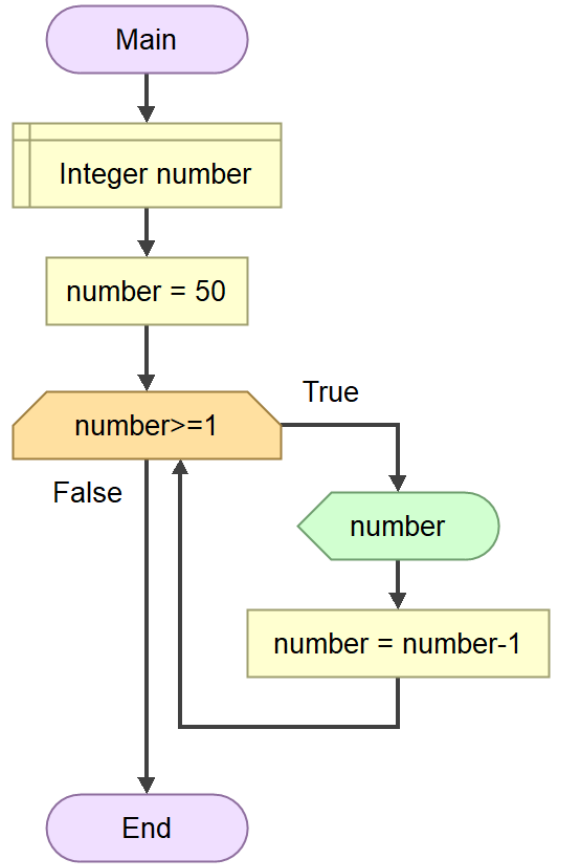
โครงสร้างการทำซ้ำแบบ while – code10

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



โครงสร้างการทำซ้ำแบบ while – code10

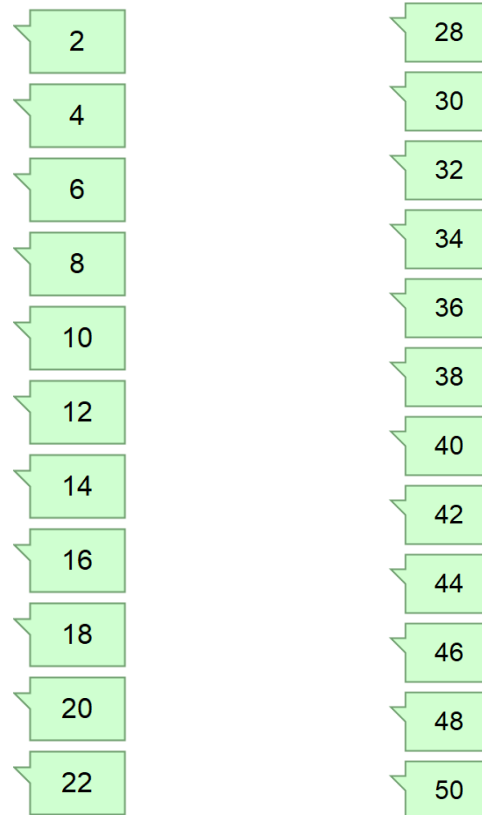
สถานการณ์ปัญหา: แสดงจำนวนตั้งแต่ 50, 49, 48 ไปเรื่อยๆ แต่ห้ามเกิน 1 โดยใช้ while loop



- | | |
|----|----|
| 50 | 12 |
| 49 | 11 |
| 48 | 10 |
| 47 | 9 |
| 46 | 8 |
| 45 | 7 |
| 44 | 6 |
| 43 | 5 |
| 42 | 4 |
| 41 | 3 |
| 40 | 2 |
| 39 | 1 |

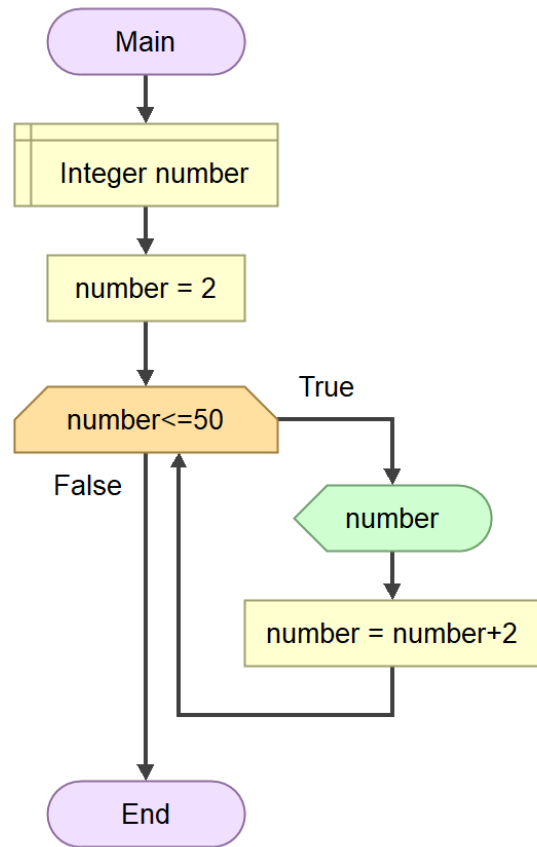
โครงสร้างการทำซ้ำแบบ while – code11

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



โครงสร้างการทำซ้ำแบบ while – code11

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

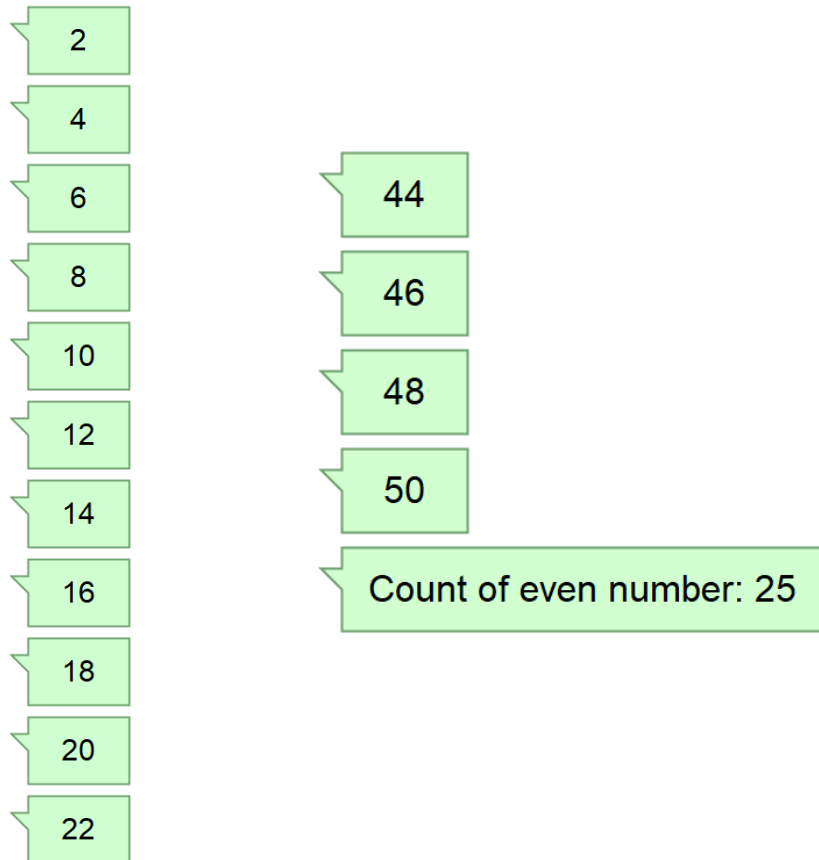


- 2
- 4
- 6
- 8
- 10
- 12
- 14
- 16
- 18
- 20
- 22

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

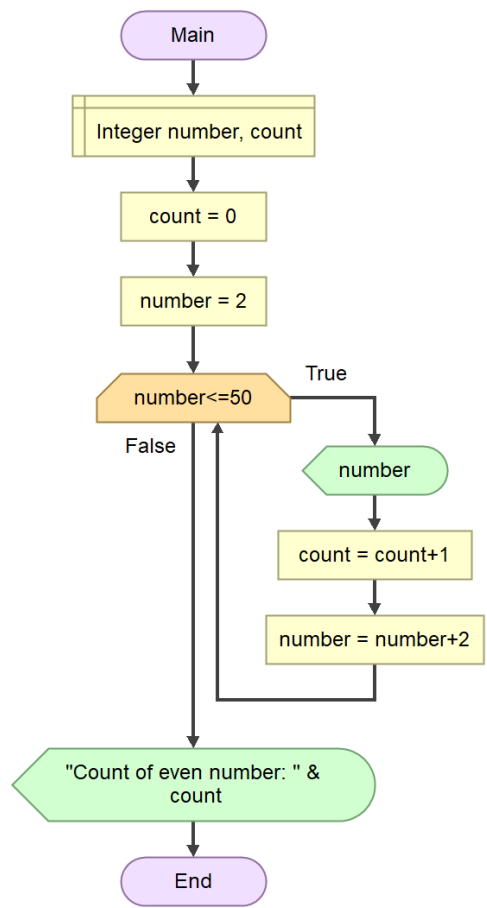
โครงสร้างการทำซ้ำแบบ while – code12

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



โครงสร้างการทำซ้ำแบบ while – code12

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 – code13

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

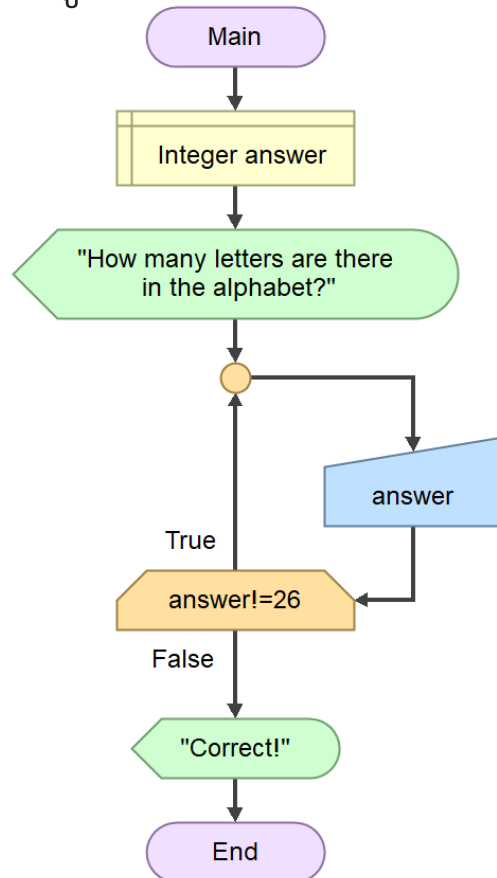
21

26

Correct!

โครงสร้างการทำซ้ำแบบ do-while – code13

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



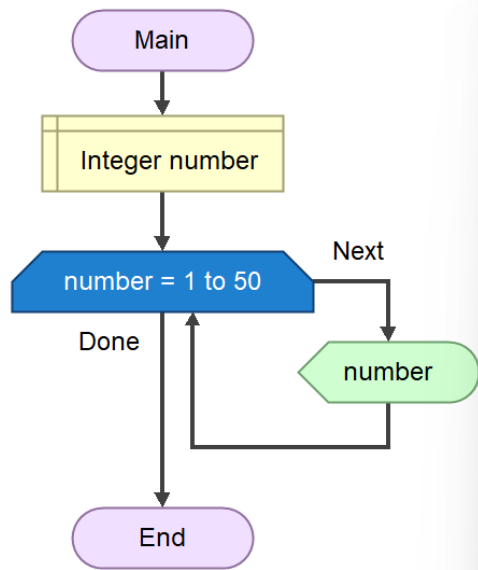
How many letters are there in the alphabet?

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

Correct!

โครงสร้างการทำซ้ำแบบ for – code14

สถานการณ์ปัญหา: แสดงจำนวนตั้งแต่ 1 ถึง 50 เท่านั้น โดยใช้ for loop



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

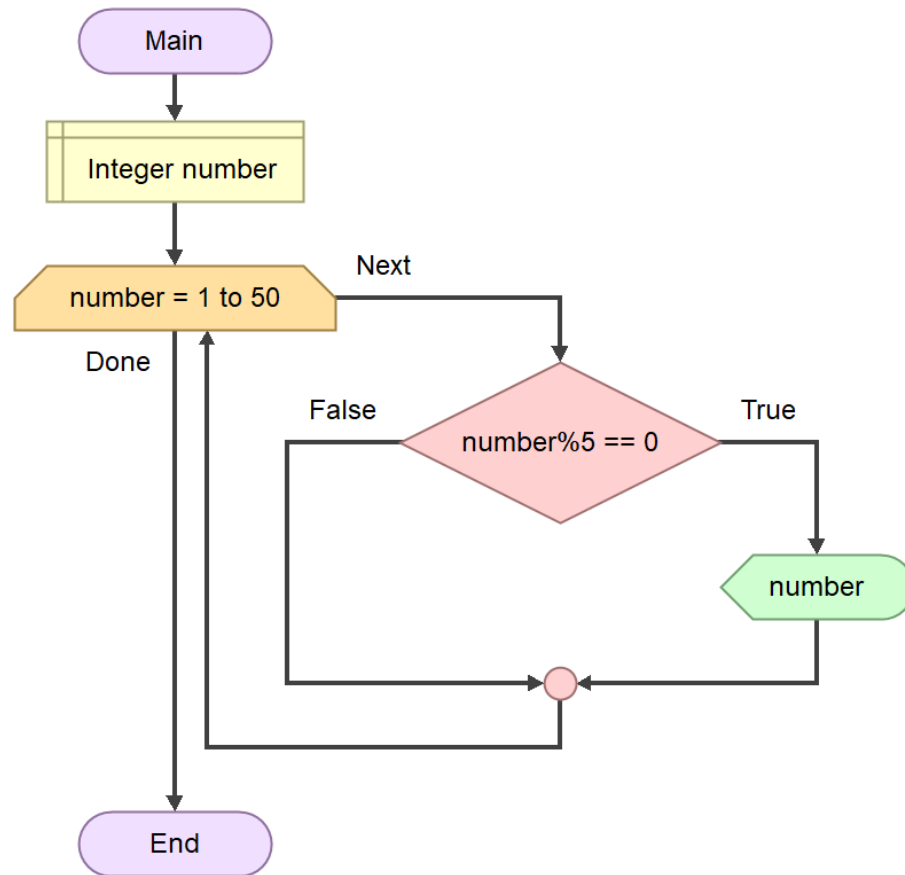
โครงสร้างการทำซ้ำแบบ for – code15

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

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

โครงสร้างการทำซ้ำแบบ for - code15

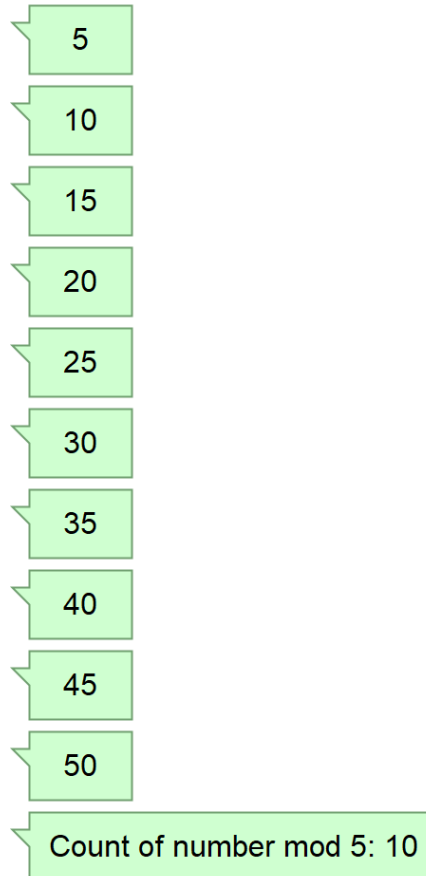
สถานการณ์ปัญหา: แสดงจำนวนตั้งแต่ 1 ถึง 50 ที่หาร 5 ลงตัวเท่านั้น โดยใช้ for loop



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

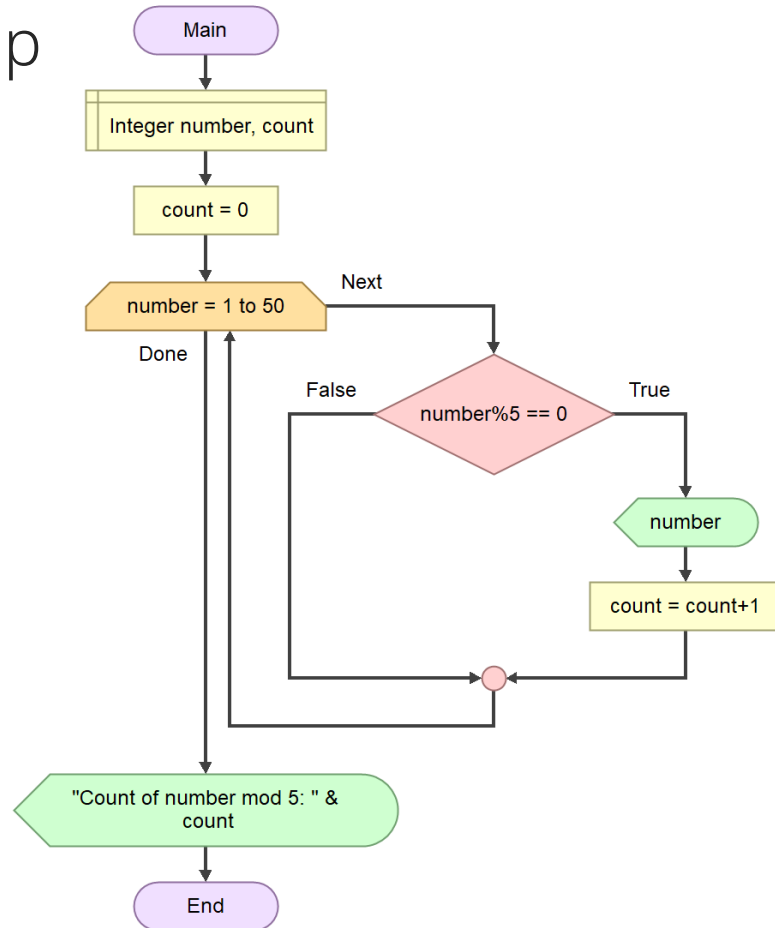
โครงสร้างการทำซ้ำแบบ for – code16

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



โครงสร้างการทำซ้ำแบบ for - code16

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



- 5
 - 10
 - 15
 - 20
 - 25
 - 30
 - 35
 - 40
 - 45
 - 50
- Count of number mod 5: 10

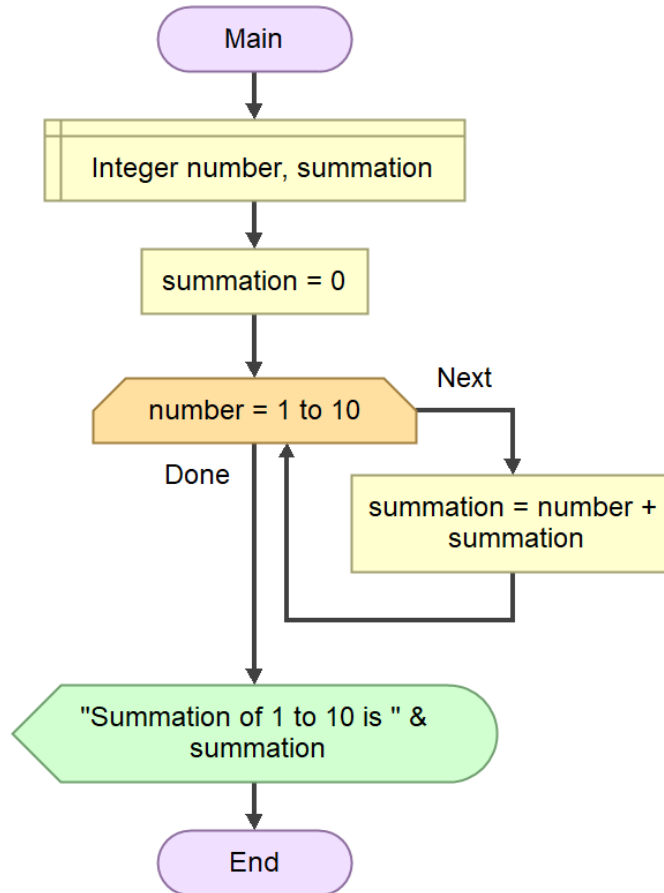
โครงสร้างการทำซ้ำแบบ for – code17

สถานการณ์ปัญหา: หาผลรวมของจำนวนตั้งแต่ 1 ถึง 10 โดยการใช้ for loop

Summation of 1 to 10 is 55

โครงสร้างการทำซ้ำแบบ for - code17

สถานการณ์ปัญหา: หาผลรวมของจำนวนตั้งแต่ 1 ถึง 10 โดยการใช้ for loop



Summation of 1 to 10 is 55

โครงสร้างการทำซ้ำแบบ for – code18

สถานการณ์ปัญหา: หาผลรวมและค่าเฉลี่ยของจำนวนตั้งแต่ 1 ถึง 10 โดยการใช้ for loop

- ค่าเฉลี่ย (average) ได้จาก ผลรวมของจำนวนทั้งหมด หารด้วย จำนวนทั้งหมด
- $\text{average} = \text{summation} / \text{count}$

Count of all number: 10

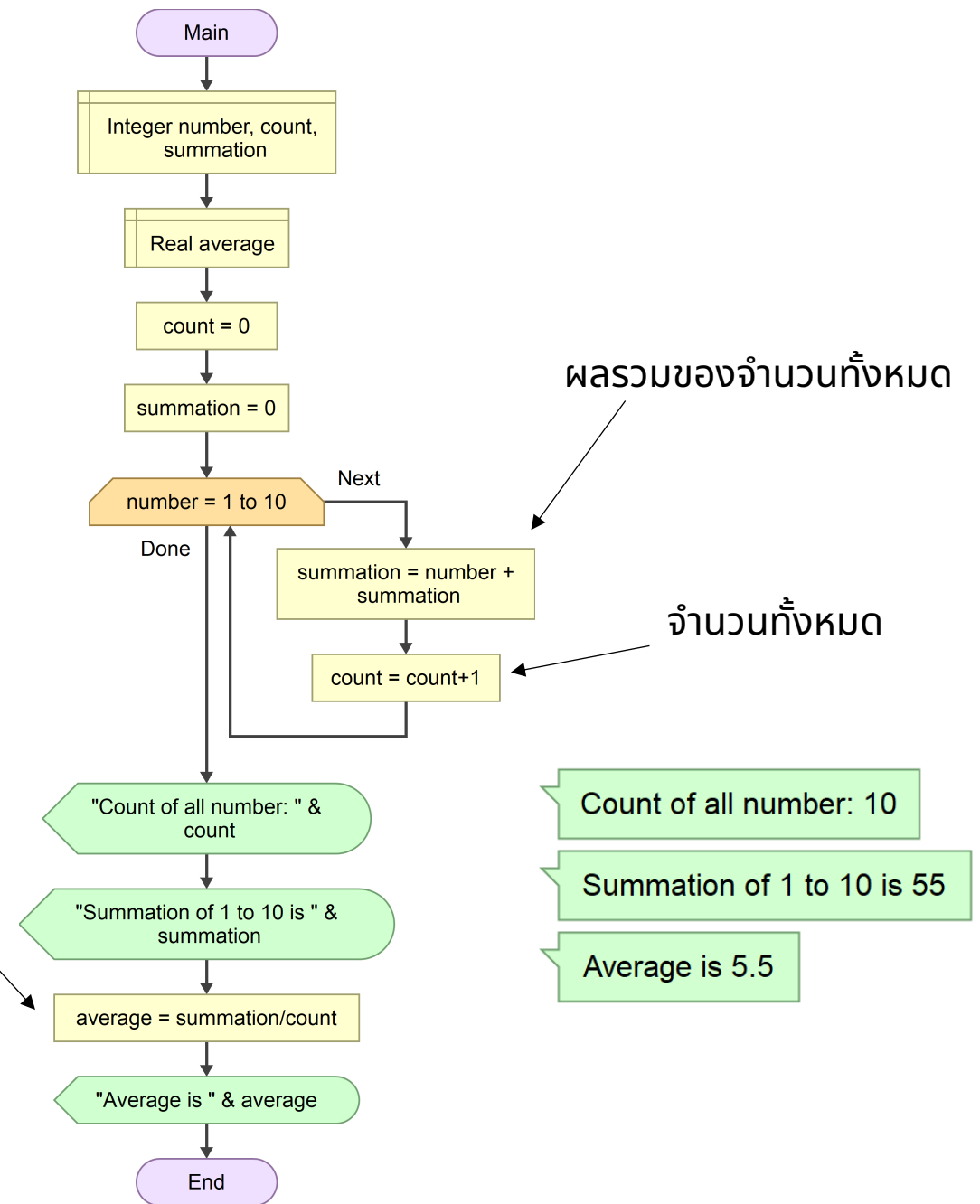
Summation of 1 to 10 is 55

Average is 5.5

โครงสร้างการทำซ้ำแบบ for - code18

สถานการณ์ปัญหา: หาผลรวมและค่าเฉลี่ยของจำนวนตั้งแต่ 1 ถึง 10 โดยการใช้ for loop

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



โครงสร้างการทำซ้ำแบบ for – code19

สถานการณ์ปัญหา: หาผลรวมและค่าเฉลี่ยของจำนวนตั้งแต่ 1 ถึง 1,000 โดยการใช้ for loop

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

Count of all number: 1000

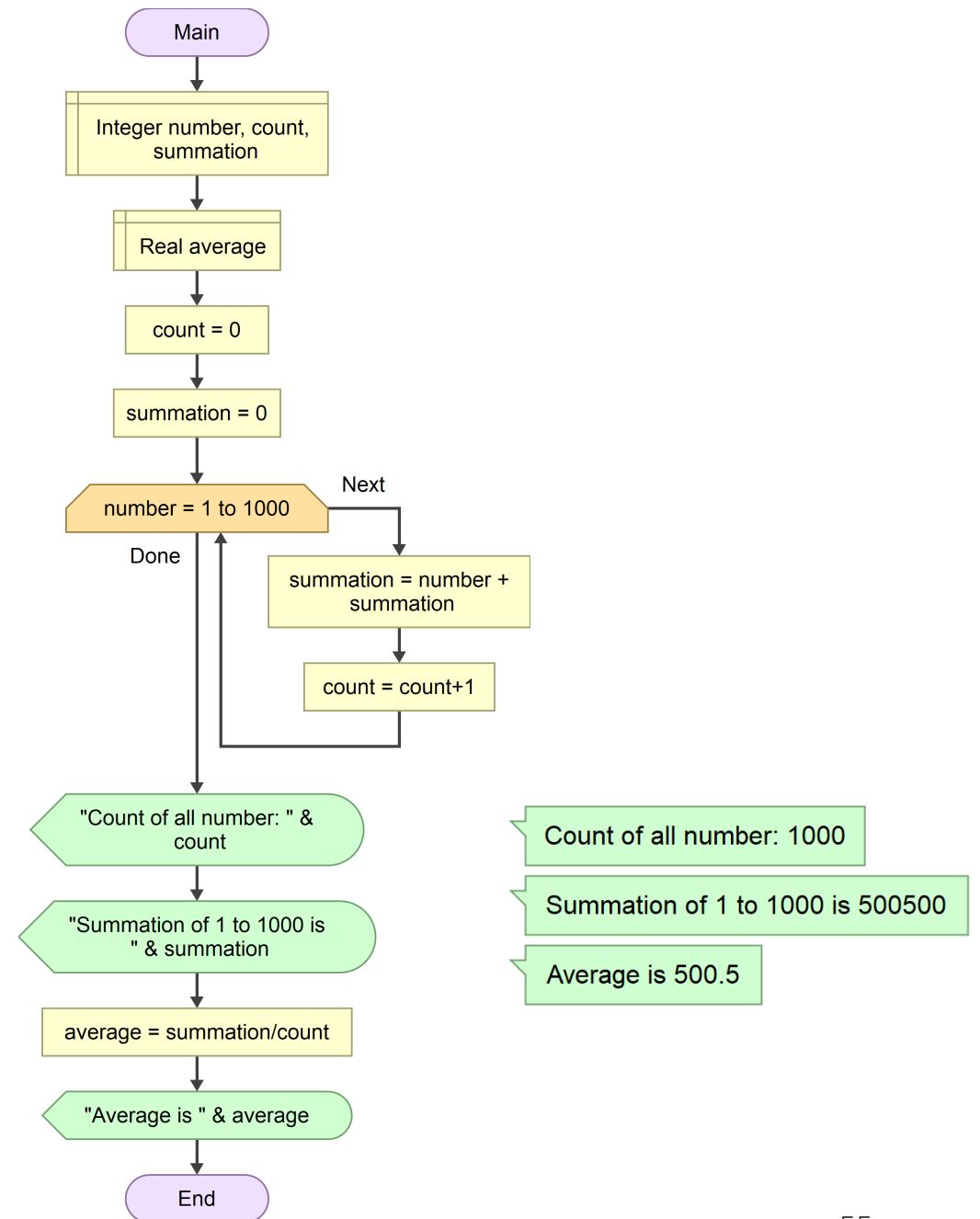
Summation of 1 to 1000 is 500500

Average is 500.5

โครงสร้างการทำซ้ำแบบ for – code19

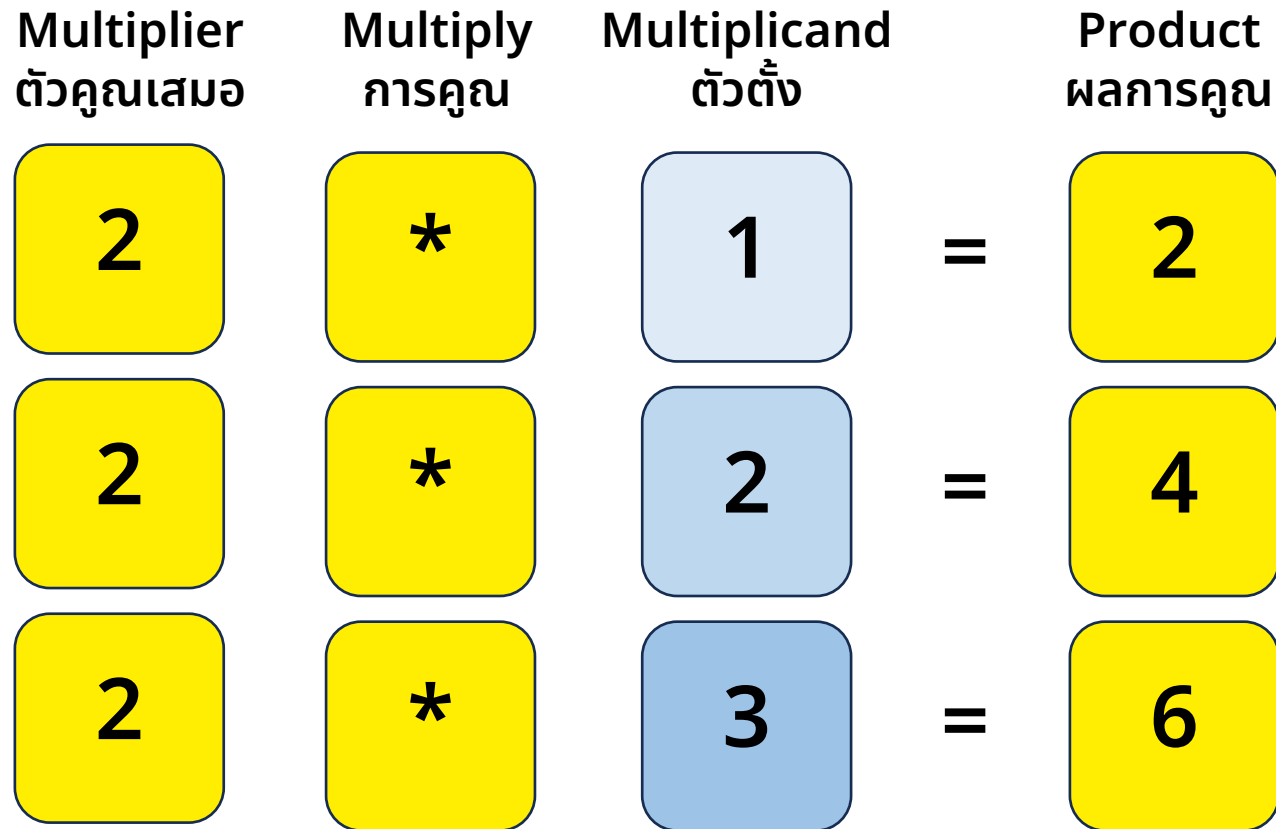
สถานการณ์ปัญหา: หาผลรวมและค่าเฉลี่ยของจำนวนตั้งแต่ 1 ถึง 1,000 โดยใช้ for loop

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



โครงสร้างการทำซ้ำแบบ for - code20

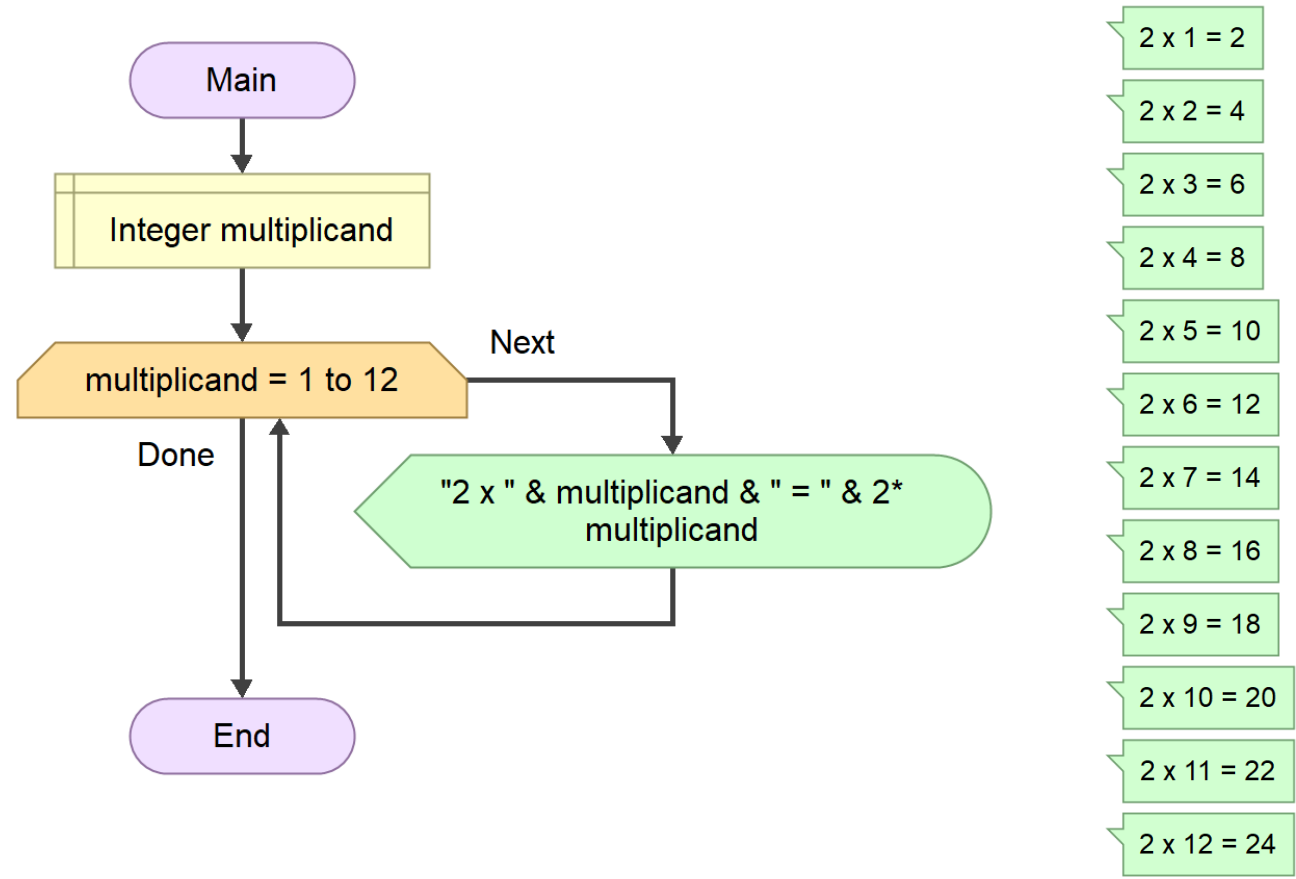
สถานการณ์ปัญหา: แสดงสูตรคูณแม่ 2 โดยเริ่มตั้งแต่ $2 \times 1 = 2$ ถึง $2 \times 12 = 24$ ด้วย for loop



- 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

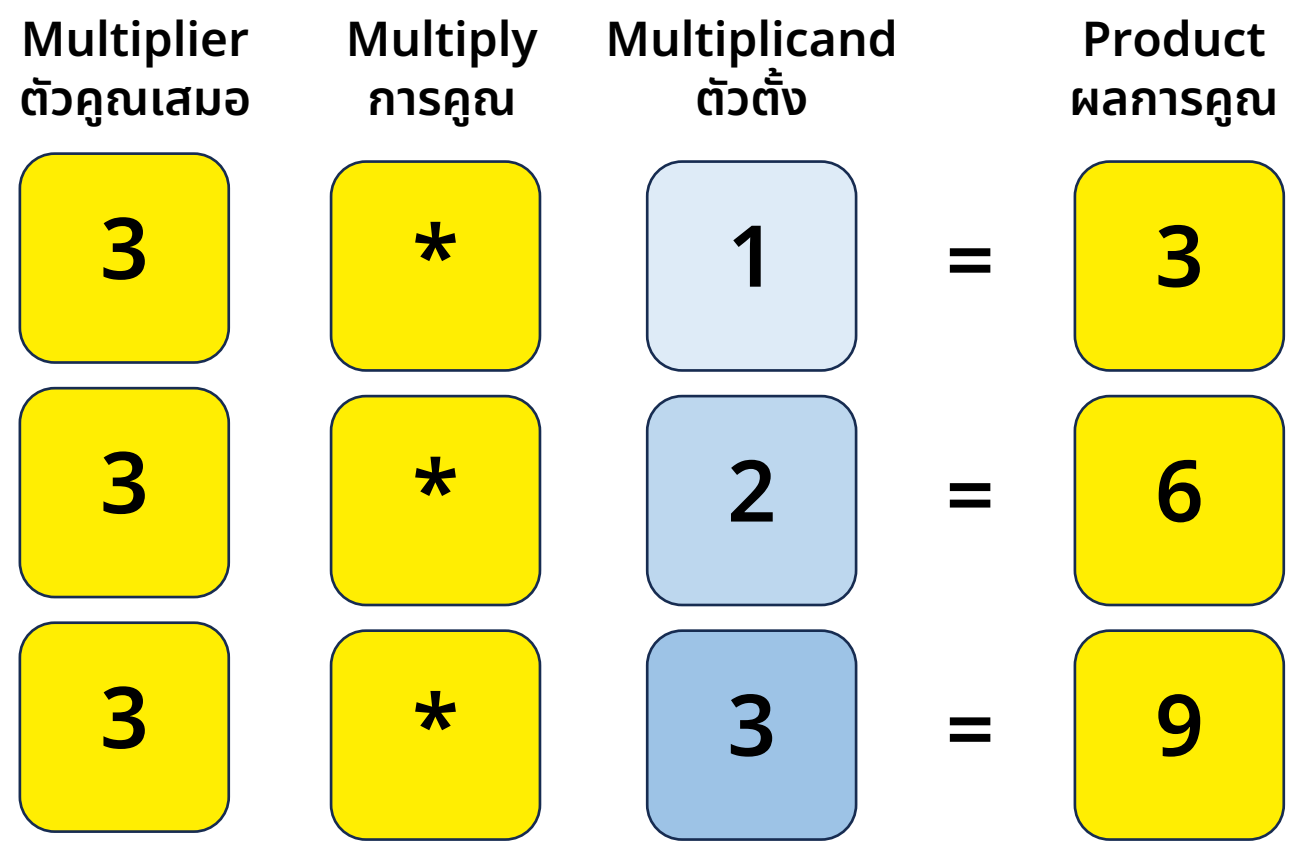
โครงสร้างการทำซ้ำแบบ for - code20

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



โครงสร้างการทำซ้ำแบบ for - code21

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



- 3 x 1 = 3
- 3 x 2 = 6
- 3 x 3 = 9
- 3 x 4 = 12
- 3 x 5 = 15
- 3 x 6 = 18
- 3 x 7 = 21
- 3 x 8 = 24
- 3 x 9 = 27
- 3 x 10 = 30
- 3 x 11 = 33
- 3 x 12 = 36

โครงสร้างการทำซ้ำแบบ for – code22

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

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



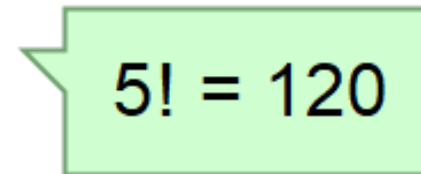
โครงสร้างการทำซ้ำแบบ for – code23

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

Factorial (n!) คือ ผลคูณของจำนวนเต็มบวกตั้งแต่ 1 ถึง n

เช่น

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



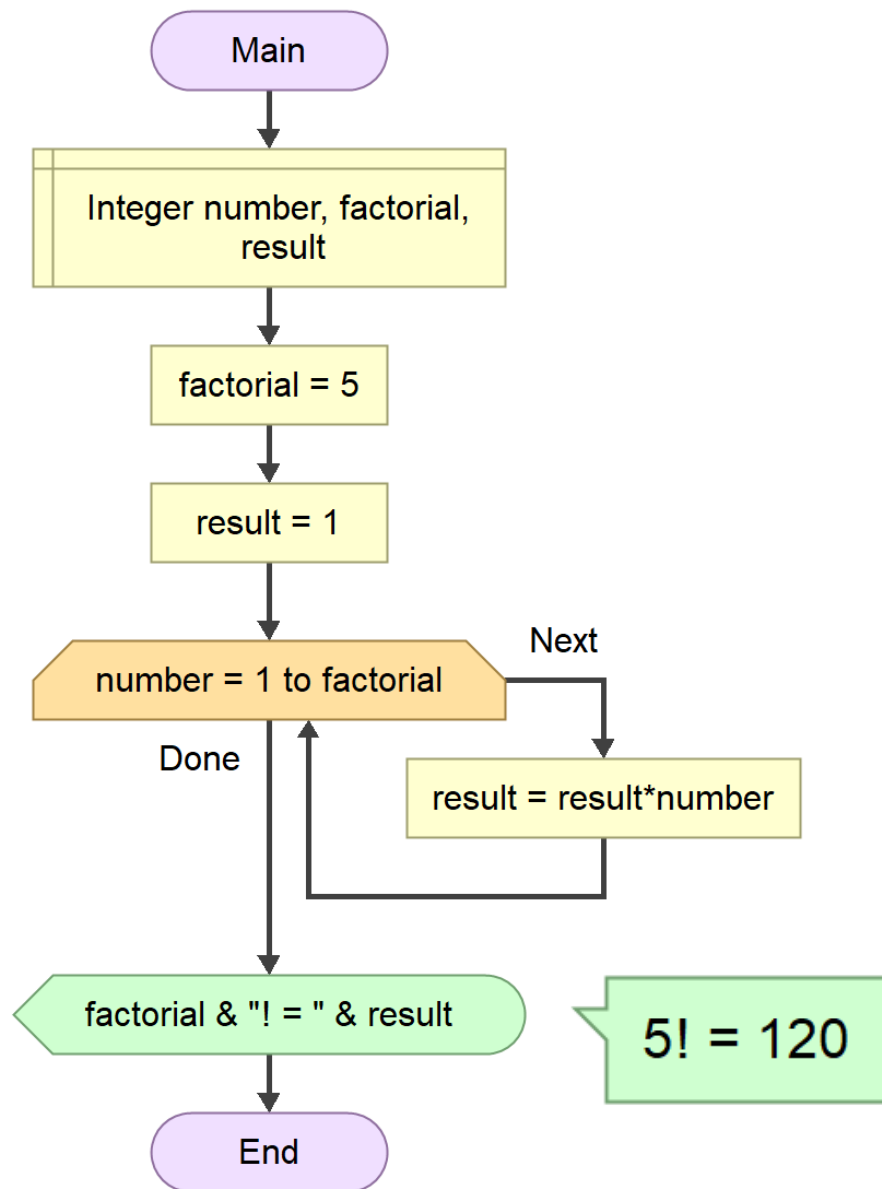
$5! = 120$

โครงสร้างการทำซ้ำแบบ for – code23

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

Factorial (n!) คือ ผลคูณของจำนวนเต็มบวกตั้งแต่ 1 ถึง n เช่น

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



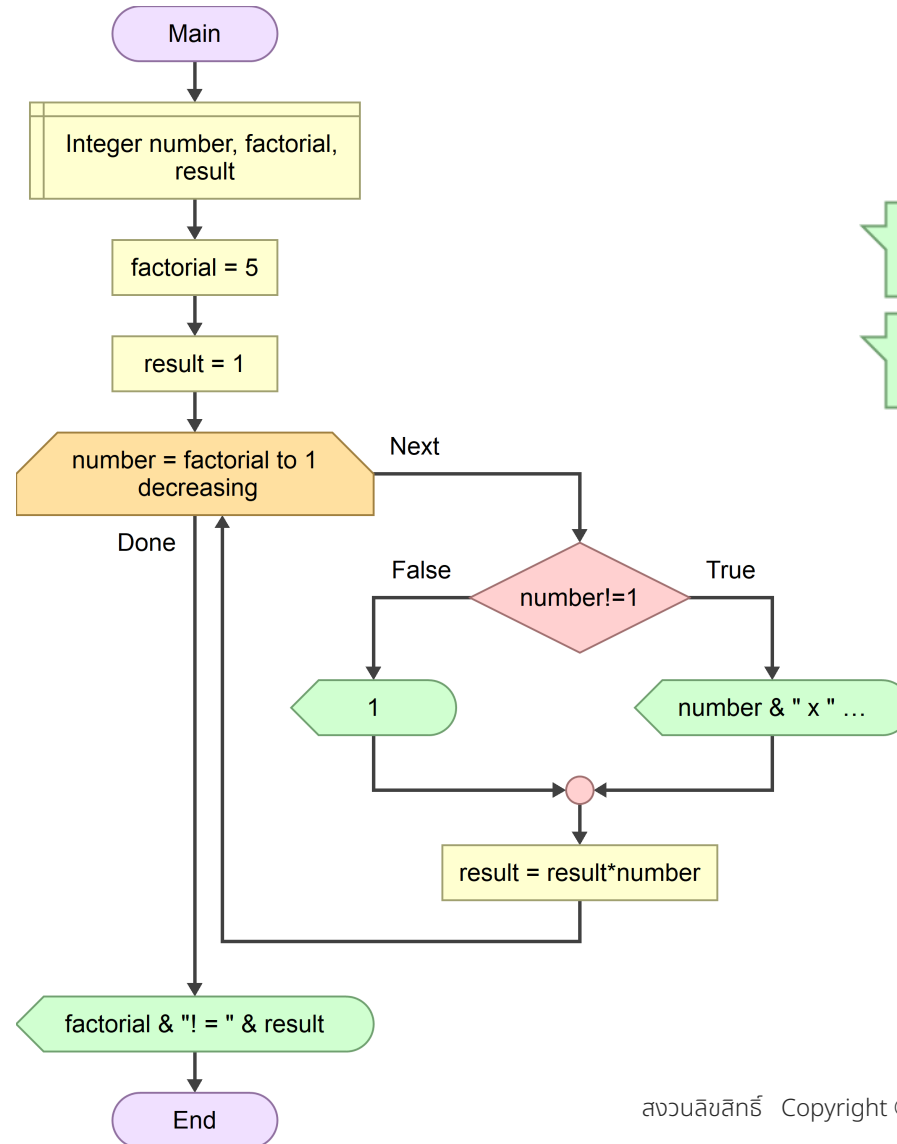
โครงสร้างการทำซ้ำแบบ for – code24

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

$$5 \times 4 \times 3 \times 2 \times 1$$
$$5! = 120$$

โครงสร้างการทำซ้ำแบบ for - code24

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



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

โครงสร้างการทำซ้ำแบบ for – code25

โจทย์สถานการณ์: ต้องการแสดงจำนวนทั้งหมดที่คูณกันจากค่า Factorial ที่รับจากผู้ใช้งาน โดยใช้ for loop

Please enter factorial number:

8

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

8! = 40320

โครงสร้างการทำซ้ำแบบ for - code25

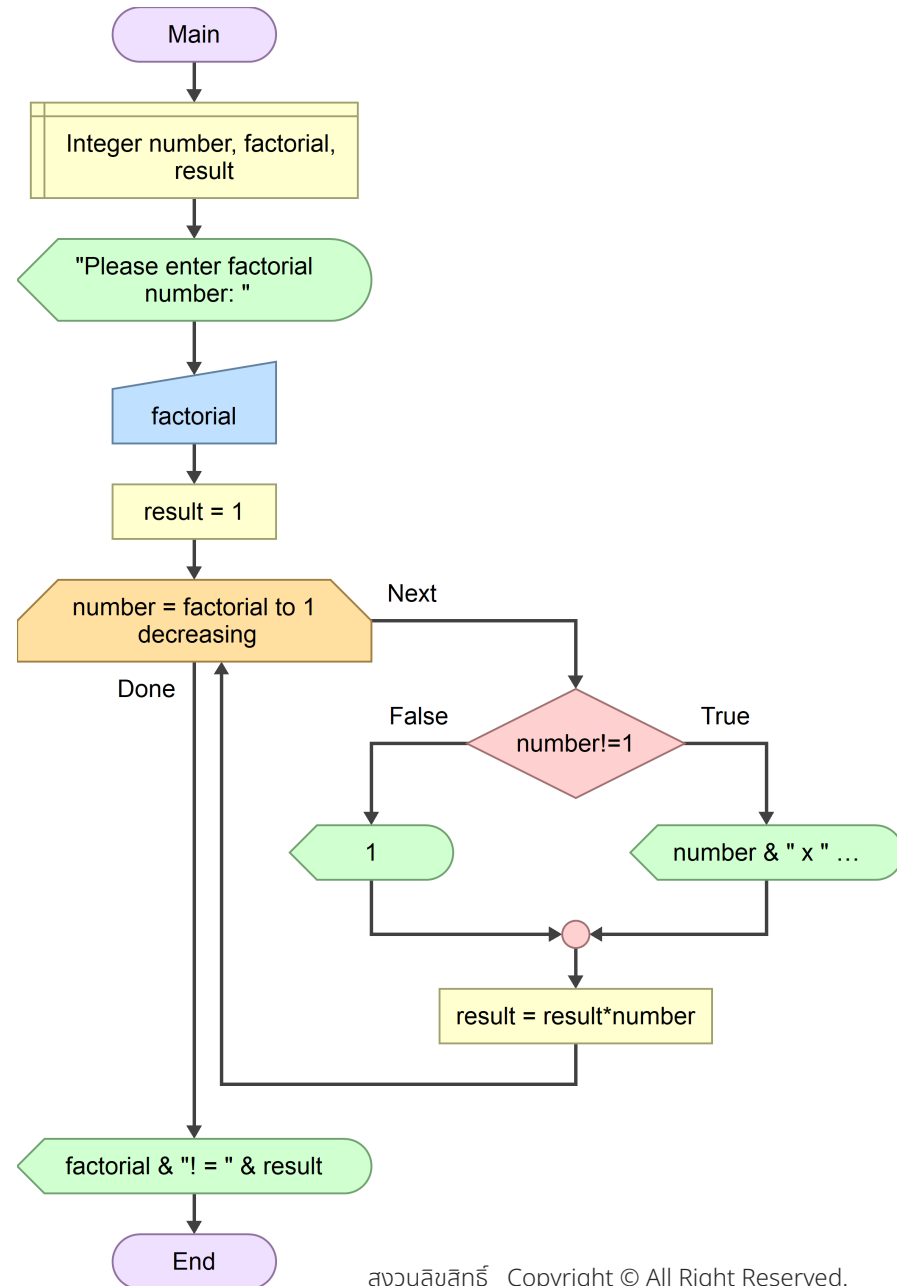
โจทย์สถานการณ์: ต้องการแสดงจำนวนทั้งหมดที่คุณค้นหาค่า Factorial ที่รับจากผู้ใช้ โดยใช้ for loop

Please enter factorial number:

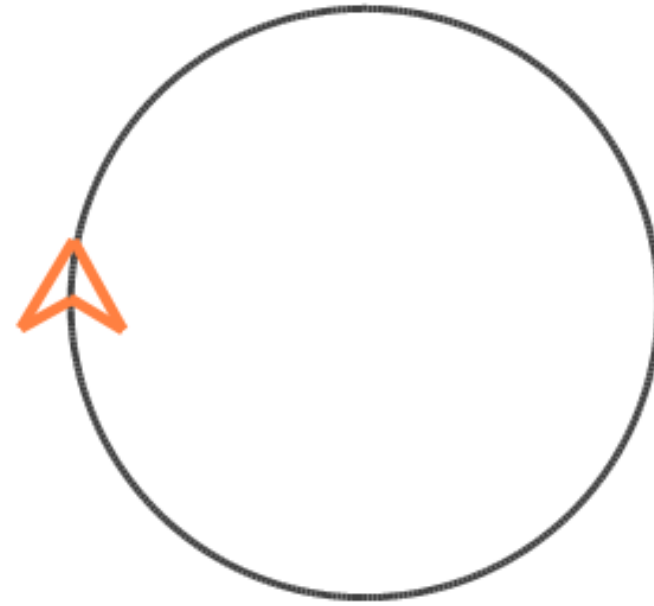
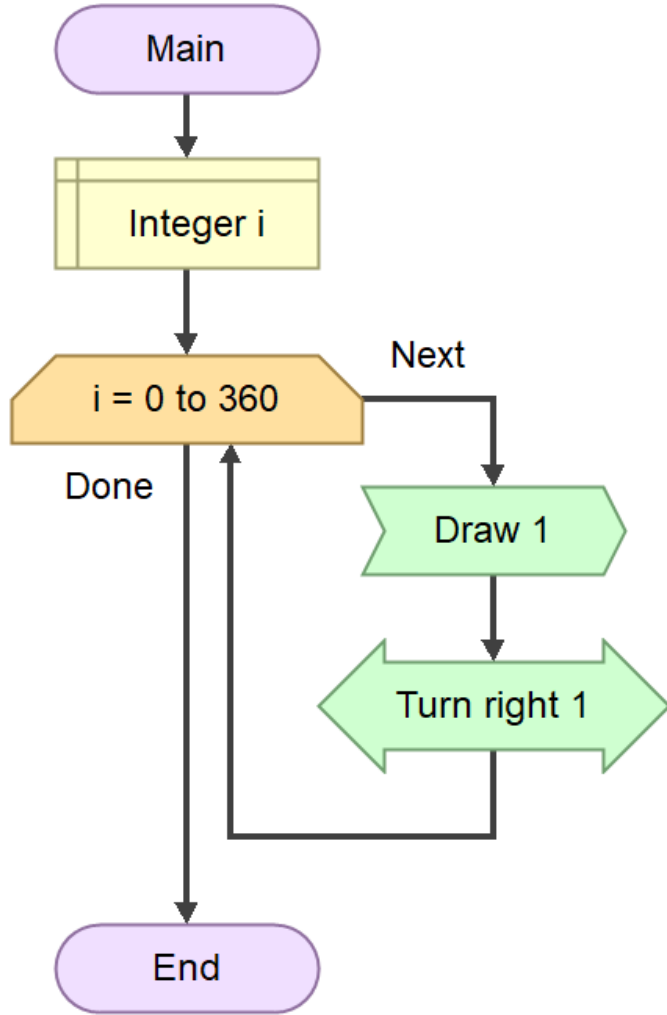
8 x 7 x 6 x 5 x 4 x 3 x 2 x 1

8! = 40320

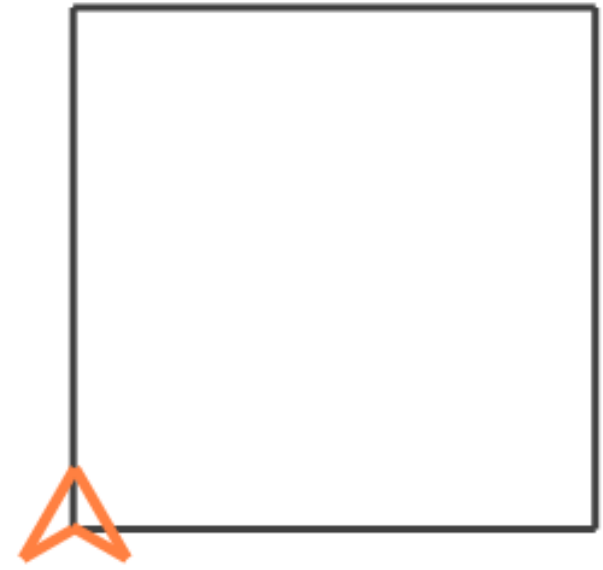
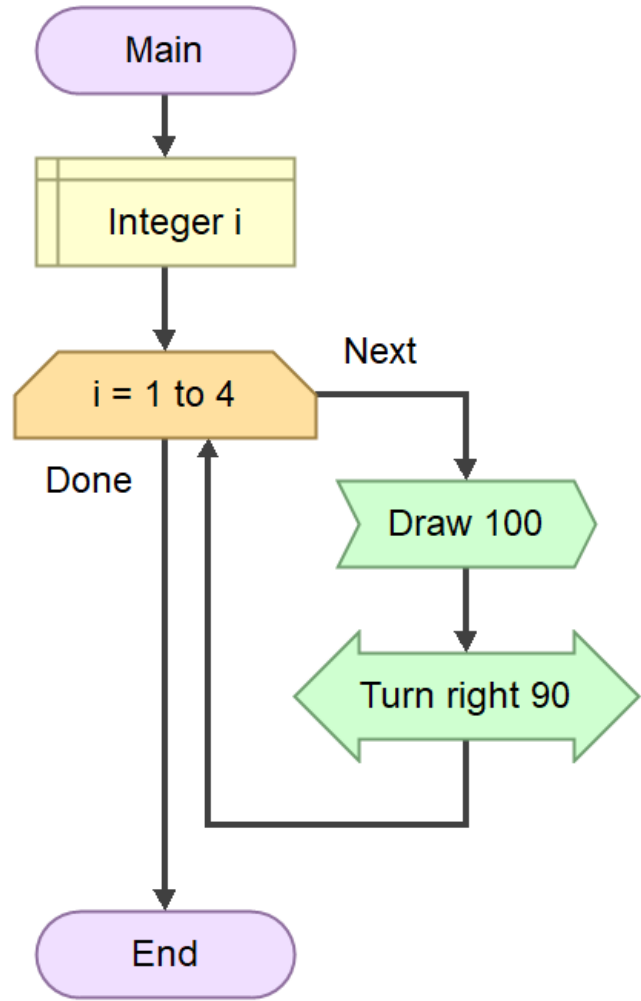
8



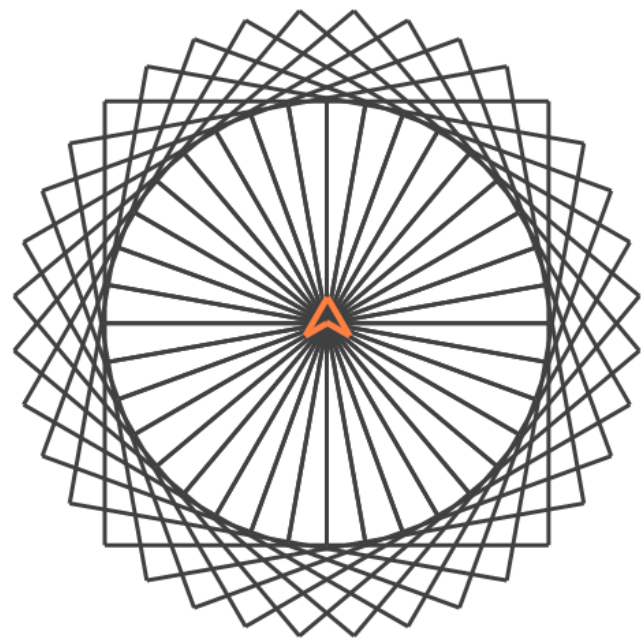
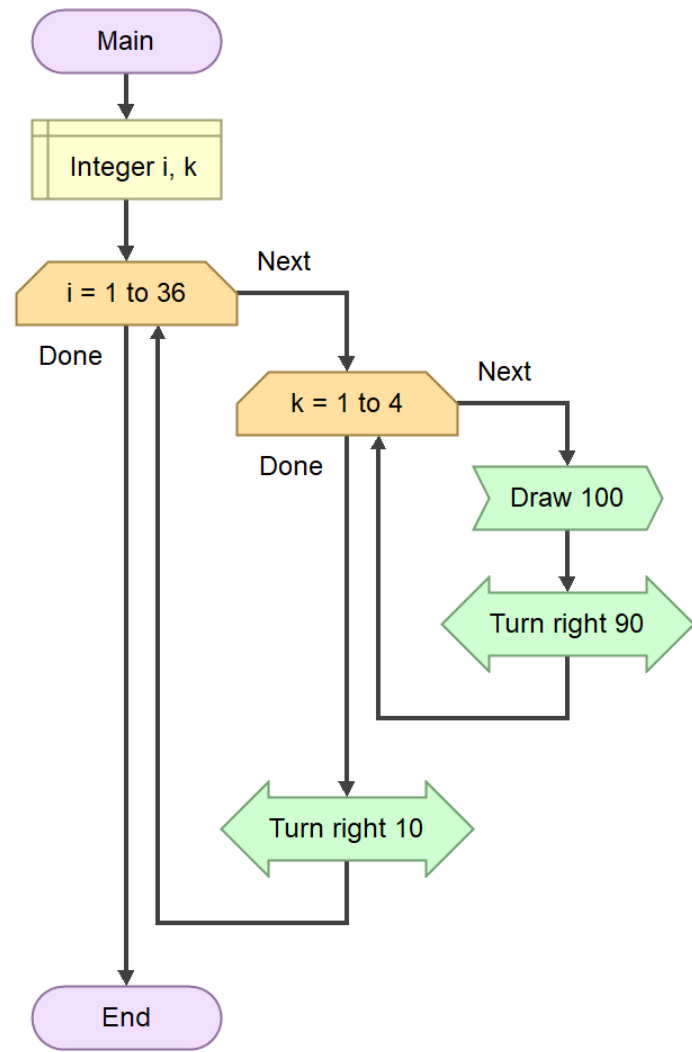
โครงสร้างการทำซ้ำแบบ for ร่วมกับการวาดภาพกราฟิก



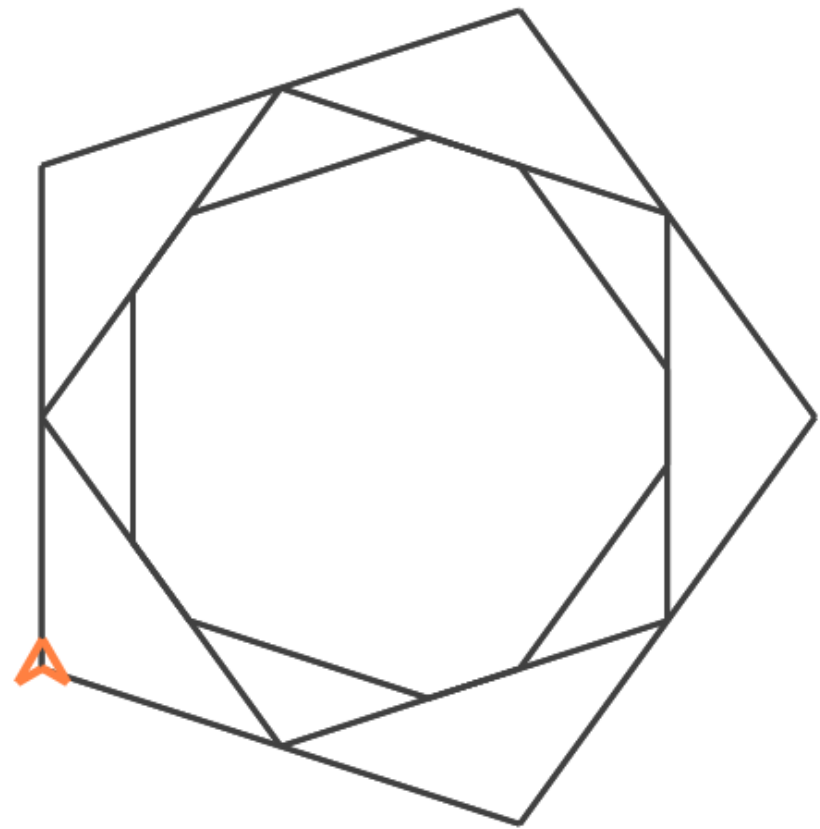
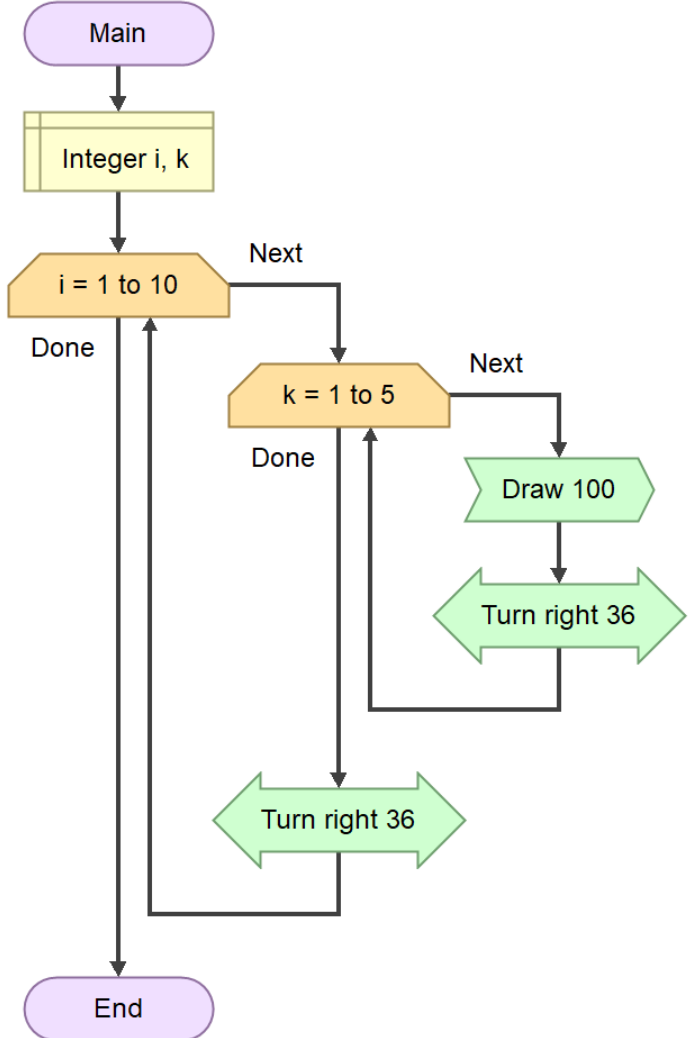
โครงสร้างการทำซ้ำแบบ for ร่วมกับการวาดภาพกราฟิก



โครงสร้างการทำซ้ำแบบ for ร่วมกับการวาดภาพกราฟิก



โครงสร้างการทำซ้ำแบบ for ร่วมกับการวาดภาพกราฟิก



Assignment บทที่ 4 ตอนที่ 2 Iteration Structure

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

เช่น

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



Assignment บทที่ 4 ตอนที่ 2 Iteration Structure

ข้อ 2: ให้เขียนโปรแกรมสุตรคูณแม่ 2 ถึงแม่ 50 โดยใช้ for loop เท่านั้น

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

Assignment บทที่ 4 ตอนที่ 2 Iteration Structure

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 *****

