



บทที่ 2

การออกแบบและพัฒนาโปรแกรมด้วยรหัสเทียม (Pseudo code)

(วิธีการเขียนขั้นตอนแก้ปัญหารหัสเทียม(Pseudo code))

สื่อสไลด์นี้ใช้รูปแบบฟอนต์ สารบรรณ รุ่นปรับปรุงใหม่ “Sarabun New”

สามารถดาวน์โหลดฟอนต์ได้ที่ URL : <https://www.f0nt.com/release/th-sarabun-new/>

การวิเคราะห์ปัญหาสำหรับการออกแบบโปรแกรม

คอมพิวเตอร์กับการแก้ปัญหา

การทำงานของเครื่องคอมพิวเตอร์ จะทำตามโปรแกรมที่เขียนขึ้น ดังนั้นการนำเครื่องคอมพิวเตอร์มาช่วยสำหรับการแก้ปัญหา จึงต้องมีโปรแกรมสำหรับการแก้ปัญหาในงานนั้น ๆ

ขั้นตอนของการวิเคราะห์ปัญหา คือ การนำปัญหาของงาน มาสร้างเป็นขั้นตอนวิธี ในรูปแบบผังงาน หรือชุดโค้ด โดยเรียงลำดับการทำงาน เพื่อนำไปใช้ในการเขียนโปรแกรม ด้วยภาษาต่าง ๆ ต่อไป



การวิเคราะห์ปัญหาสำหรับการออกแบบโปรแกรม

ขั้นตอนของการแก้ปัญหา (Problem Solving) สำหรับเตรียมการก่อนลงมือเขียนโปรแกรมคอมพิวเตอร์ มี 5 ขั้นตอนดังนี้

ขั้นตอนการแก้ปัญหา



Problem Specification Phase
การทำความเข้าใจกับปัญหา



Input and Output Specification Phase
พิจารณาลักษณะของข้อมูลเข้าและข้อมูลออก



Hand Example Phase
การทดลองแก้ไขปัญหาด้วยตนเอง



Algorithm Development Phase
การเขียนขั้นตอนวิธีการแก้ปัญหา



Testing Phase
การทดสอบขั้นตอนวิธีการแก้ปัญหา

การวิเคราะห์ปัญหาสำหรับการออกแบบโปรแกรม

- การเขียนขั้นตอนวิธีการแก้ปัญหา (Algorithm Development Phase)

นำขั้นตอนของการทดลองการแก้ไขปัญหาด้วยตนเอง มาทำการเรียบเรียงเป็นลำดับขั้นตอนวิธีการทำงาน โดยเขียนเป็นข้อ ๆ ตั้งแต่ขั้นตอนแรกจนถึงขั้นตอนสุดท้าย

- เขียนลำดับขั้นตอนวิธีการทำงานทั้งหมดอย่างย่อ

- เขียนลำดับขั้นตอนวิธีการทำงานอย่างละเอียด

- วิธีการเขียนลำดับขั้นตอนวิธี

- Flowcharts เขียนขั้นตอนวิธีด้วยรูปภาพผังงาน

- Pseudo code (อ่านว่า ซูโดโค้ด) เขียนขั้นตอนวิธีด้วยภาษาที่ใกล้เคียงภาษาคอมพิวเตอร์



รหัสเทียม (Pseudo Code)

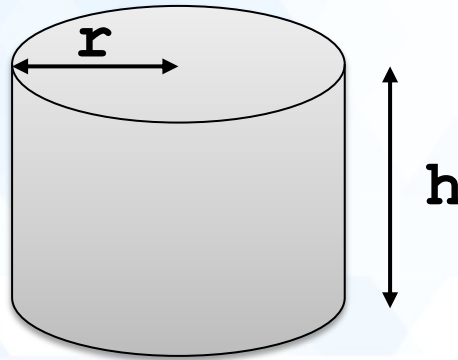
รหัสเทียม

- เป็นคำอธิบายการทำงานของขั้นตอนวิธี (Algorithm)
- ส่วนมากเขียนด้วยภาษาอังกฤษ
ที่ไม่คำนึงถึงไวยากรณ์ และไม่ขึ้นกับภาษาคอมพิวเตอร์ใด ๆ
- หลักการเขียนรหัสเทียม
 - ใช้ถ้อยคำที่เข้าใจง่าย
 - ในหนึ่งบรรทัด มีเพียงหนึ่งการกระทำ
 - เขียนคำสั่งการทำงานจากบนลงล่าง
 - ใช้ย่อหน้า เพื่อแสดงกลุ่มของคำสั่งที่ทำงานในระดับเดียวกัน

1 การวิเคราะห์ปัญหาสำหรับการออกแบบโปรแกรม

ตัวอย่างที่ 1 ขั้นตอนวิธีการแก้ปัญหาด้วยคอมพิวเตอร์

จงเขียนขั้นตอนวิธีการแก้ปัญหาสำหรับการหาค่าปริมาตรของรูปทรงกระบอก



สูตรการคำนวณ ปริมาตรทรงกระบอก = $\pi \times r^2 \times h$

1 การวิเคราะห์ปัญหาสำหรับการออกแบบโปรแกรม

- การพัฒนาลำดับขั้นตอนวิธีการแก้ปัญหา

วิธีการเขียนขั้นตอนแก้ปัญหา

1) รับข้อมูล

High และ Radius



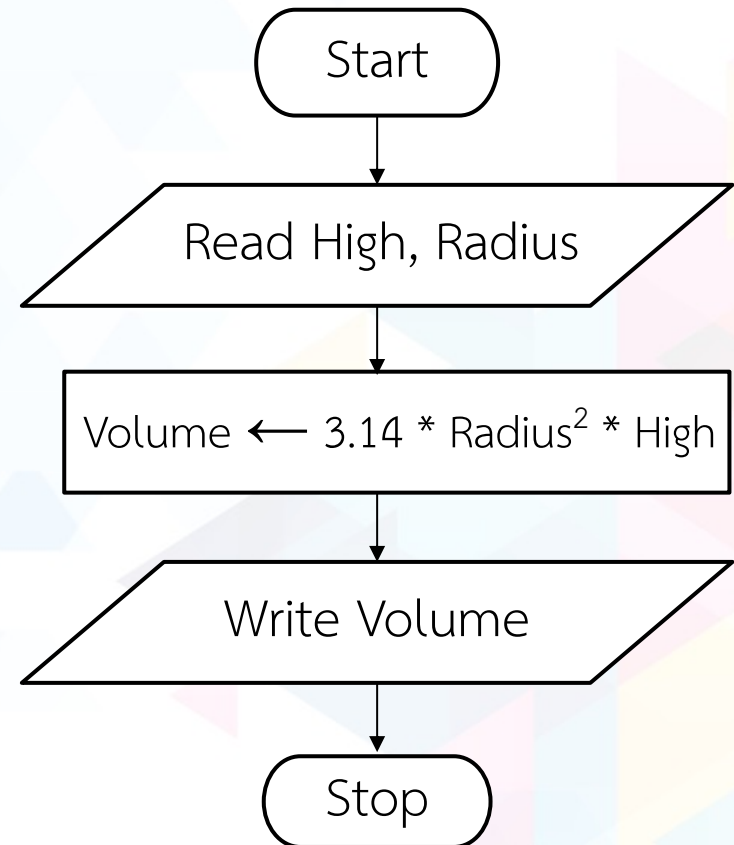
2) คำนวณ

$$\text{Volume} = 3.14 * \text{Radius}^2 * \text{High}$$

3) เขียนข้อมูล Volume



ผังงาน (Flowchart)



1 การวิเคราะห์ปัญหาสำหรับการออกแบบโปรแกรม

- การพัฒนาลำดับขั้นตอนวิธีการแก้ปัญหา

วิธีการเขียนขั้นตอนแก้ปัญหา

1) รับข้อมูล

High และ Radius



2) คำนวณ

$$\text{Volume} = 3.14 * \text{Radius}^2 * \text{High}$$

3) เขียนข้อมูล Volume



รหัสเทียม (Pseudo Code)

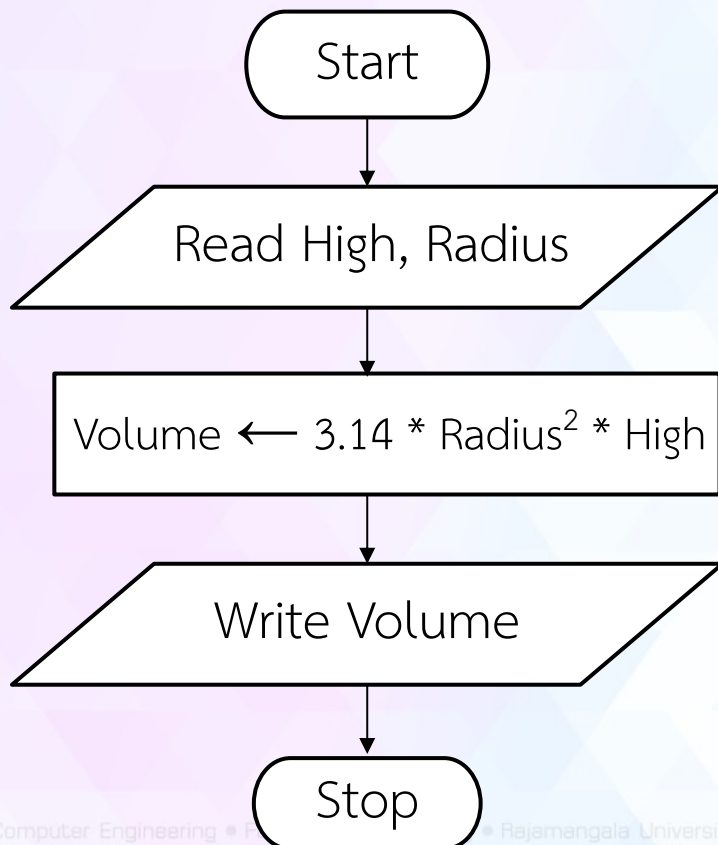
1. Begin
2. Read High, Radius
3. Volume \leftarrow 3.14 * Radius² * High
4. Write Volume
5. End

1 การวิเคราะห์ปัญหาสำหรับการออกแบบโปรแกรม

- การพัฒนาลำดับขั้นตอนวิธีการแก้ปัญหา

วิธีการเขียนขั้นตอนแก้ปัญหา

ผังงาน (Flowchart)



รหัสเทียม (Pseudo Code)

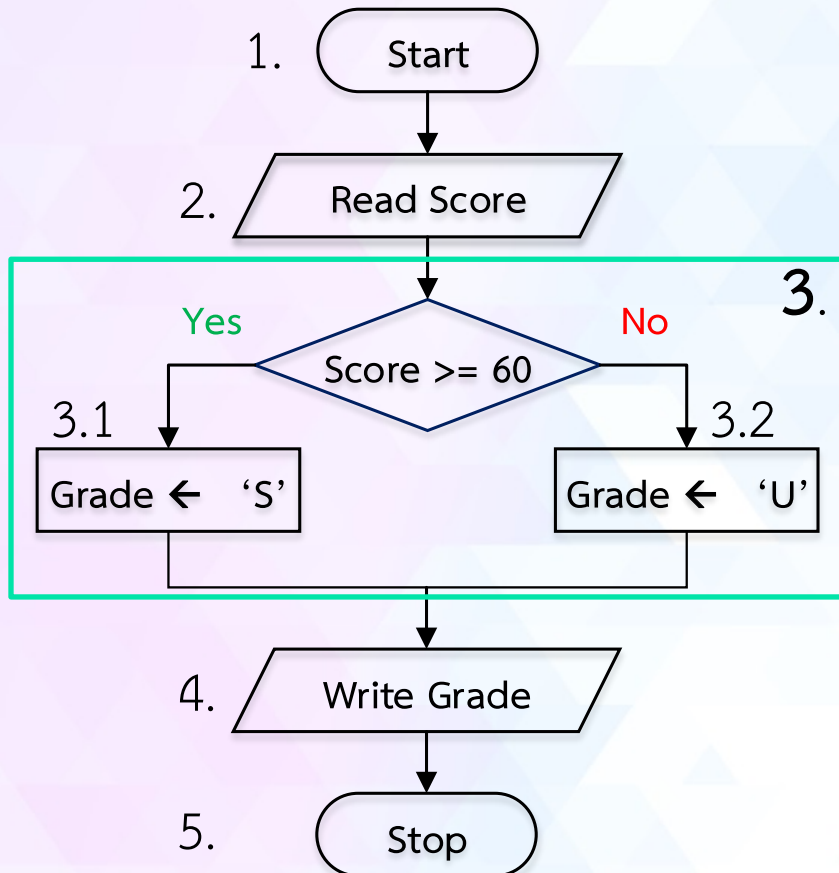
1. Begin
2. **Read** High, Radius
3. Volume ← 3.14 * Radius² * High
4. **Write** Volume
5. End

2 การวิเคราะห์ปัญหาสำหรับการออกแบบโปรแกรมแบบเลือกทำ

ตัวอย่างที่ 2 แสดงขั้นตอนวิธีการแก้ปัญหาด้วยคอมพิวเตอร์ Flowchart และ Pseudo

code

ผังงาน (Flowchart)



รหัสเทียม (Pseudo)

1. Begin
2. Read Score
3. If Score ≥ 60
 - 3.1 Grade \leftarrow 'S'Else
 - 3.2 Grade \leftarrow 'U' Grade
4. Write Grade
5. End

■ ขั้นตอนที่ 4 การพัฒนาลำดับขั้นตอนวิธีการแก้ปัญหา

รหัสเทียม (Pseudo Code)

1. Begin
2. **Read** Score เครื่องคอมพิวเตอร์จะรับข้อมูล 1 ค่า คือ คะแนนประเมิน (Score)
3. If Score \geq 60 Then คะแนนประเมินตั้งแต่ 60 ขึ้นไป
 - 3.1 Grade \leftarrow 'S' Grade ได้ตัวอักษรเกรด S
- Else นอกจากนั้น หรือ มิฉะนั้นแล้ว
 - 3.2 Grade \leftarrow 'U' Grade ได้ตัวอักษรเกรด U
4. **Write** Grade แสดงผลลัพธ์ (Grade) ของเกรดที่ได้
5. End

จำนวนคิดเกรด

3 การวิเคราะห์ปัญหาสำหรับการออกแบบโปรแกรม แบบทำซ้ำ

ตัวอย่างที่ 3 ขั้นตอนวิธีการแก้ปัญหาด้วยคอมพิวเตอร์

จงเขียนขั้นตอนวิธีการแก้ปัญหาด้วยการหาผลรวมของตัวเลขจำนวนเต็ม
ตั้งแต่เลข 1 ถึง ค่า n

สูตรการคำนวณ

$$\sum_{i=1}^n i = 1 + 2 + \dots + n$$

3 การวิเคราะห์ปัญหาสำหรับการออกแบบโปรแกรมแบบทำซ้ำ

- ขั้นตอนที่ 1 ทำความเข้าใจกับปัญหา

คำนวณผลรวมตัวเลขจำนวนเต็ม ตั้งแต่ 1 ถึง ค่า n

- ขั้นตอนที่ 2 ลักษณะของข้อมูลเข้าและข้อมูลออก

ข้อมูลเข้า คือ ตัวเลขค่าสุดท้ายที่จะคำนวณถึง เป็นข้อมูลชนิดตัวเลขจำนวนเต็ม

ข้อมูลออก คือ ตัวเลขผลรวมที่คำนวณได้ เป็นข้อมูลชนิดตัวเลขจำนวนเต็ม

- ขั้นตอนที่ 3 ทดลองแก้ไขปัญหาด้วยตนเอง

ถ้าให้ n มีค่าเท่ากับ 6

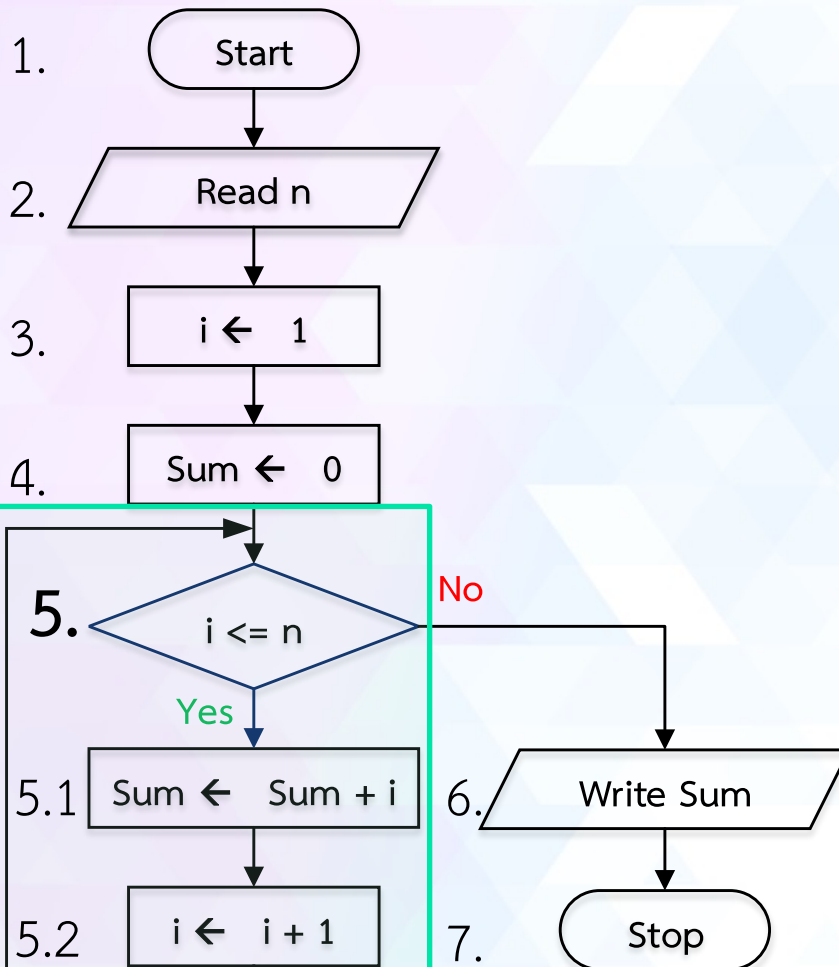
$$\text{ผลรวม} = 1 + 2 + 3 + 4 + 5 + 6 = 21$$

คำตอบ ผลรวม มีค่าเท่ากับ 21

3 การวิเคราะห์ปัญหาสำหรับการออกแบบโปรแกรม

- ขั้นตอนที่ 4 การพัฒนาลำดับขั้นตอนวิธีการแก้ปัญหา

ผังงาน (Flowchart)



1. Begin
2. Read n
3. $i = 1$
4. $Sum = 0$
5. **While** $i \leq n$ do
 - 5.1 $Sum \leftarrow Sum + i$
 - 5.2. $i \leftarrow i + 1$
6. Write Sum
7. End

3 การวิเคราะห์ปัญหาสำหรับการออกแบบโปรแกรม

■ ขั้นตอนที่ 4 การพัฒนาลำดับขั้นตอนวิธีการแก้ปัญหา

รหัสเทียม (Pseudo Code)

1. Begin
2. **Read** n เครื่องคอมพิวเตอร์จะรับข้อมูล 1 ค่า คือ ค่าสุดท้าย (n)
3. $i = 1$ กำหนดค่าตัวนับ (i) เริ่มต้นที่ 1
4. $Sum = 0$ กำหนดค่าผลรวม (Sum) เริ่มต้นที่ 0
5. While $i \leq n$ do ในขณะที่ ตัวนับ (i) น้อยกว่าหรือเท่ากับ ค่าสุดท้าย (n)
 - 5.1 $Sum \leftarrow Sum + i$ ให้ผลรวม มีค่าเท่ากับ ผลรวมเดิม + ตัวนับ
 - 5.2 $i \leftarrow i + 1$ ให้ตัวนับ มีค่าเท่ากับ ตัวนับเดิม + 1
6. **Write** Sum แสดงผลรวม (Sum) ที่ได้
7. End

3 การวิเคราะห์ปัญหาสำหรับการออกแบบโปรแกรม แบบทำซ้ำ(2)

ตัวอย่างที่4 ขั้นตอนวิธีการแก้ปัญหาด้วยคอมพิวเตอร์

ตรวจสอบการรับเลขเป็นจำนวนเต็มและมีค่าตั้งแต่ 2 ถึง 12 หรือไม่

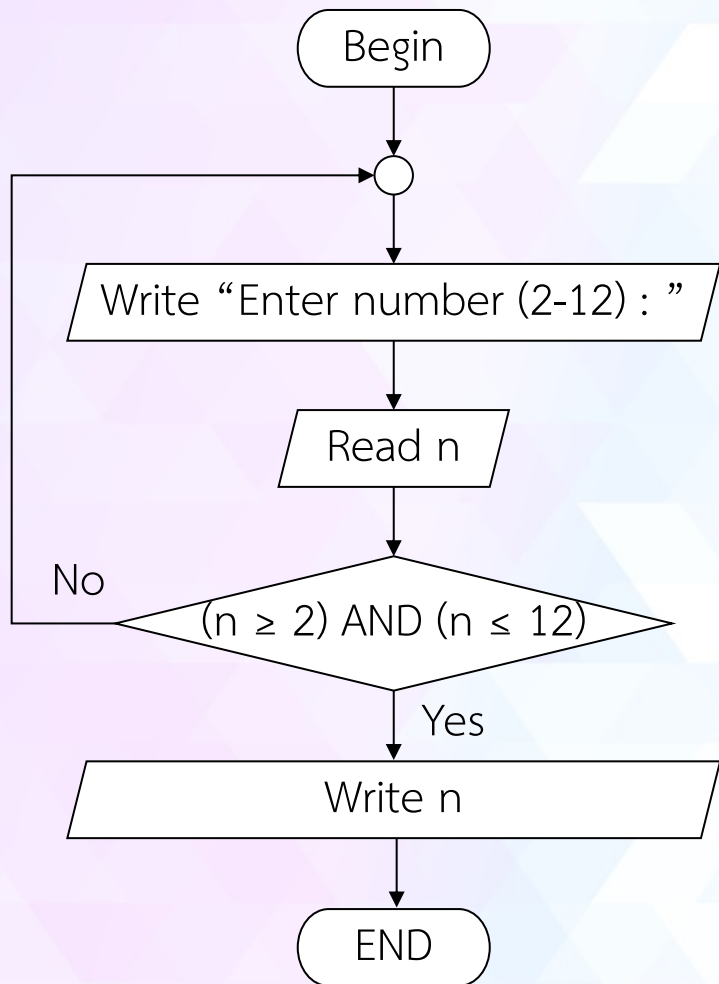
เป้าหมาย:	ตรวจสอบการรับเลขเป็นจำนวนเต็มและมีค่าตั้งแต่ 2 ถึง 12
ข้อมูลออก:	จำนวนเต็มที่มีค่าตั้งแต่ 2 ถึง 12
ข้อมูลเข้า:	จำนวนเต็มที่มีค่าตั้งแต่ 2 ถึง 12

ถ้าค่า n ไม่ได้อยู่ในช่วงที่ระบุ จะรับค่า n ไปเรื่อย ๆ

ข้อมูลเข้า:

แม่สูตรคูณ (n) ที่เป็นจำนวนเต็มที่มีค่าตั้งแต่ 2 ถึง 12

ถ้าค่า n ไม่ได้อยู่ในช่วงที่ระบุ จะรับค่า n ไปเรื่อย ๆ



1. Begin
2. REPEAT
 - 2.1 Write "Enter number (2-12) : "
 - 2.2 Read n
 - 2.3 UNTIL $(n \geq 2) \text{ AND } (n \leq 12)$
3. Write Sum
4. End

จบบทที่ 2

สื่อสไลด์นี้ใช้รูปแบบฟอนต์ สารบรรณ รุ่นปรับปรุงใหม่ “Sarabun New”

สามารถดาวน์โหลดฟอนต์ได้ที่ URL : <https://www.f0nt.com/release/th-sarabun-new/>