

วิชา การวิเคราะห์และออกแบบเชิงวัตถุ

(Object Oriented Analysis and Design)



ผู้ช่วยศาสตราจารย์จุฑาวุฒิ จันทรมาลี

หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยสวนดุสิต

หน่วยที่ 2 หลักการพื้นฐานและแนวคิดเชิงวัตถุ

สาระการเรียนรู้

1. การเขียนโปรแกรมเชิงโครงสร้าง
2. การเขียนโปรแกรมเชิงวัตถุ
3. วัตถุหรืออ็อบเจกต์
4. คุณสมบัติทั่วไปของอ็อบเจกต์
5. องค์ประกอบของวัตถุหรืออ็อบเจกต์
6. ความสัมพันธ์ระหว่างอ็อบเจกต์

สาระการเรียนรู้

7. องค์ประกอบสำคัญของแนวคิดเชิงวัตถุ
8. กระบวนการพัฒนาเชิงวัตถุ
9. แนวคิดรวบยอดของอ็อบเจกต์
10. Abstraction และ Encapsulation
11. ภาษายูเอ็มแอล (UML)
12. ภาษาโปรแกรมเชิงวัตถุ

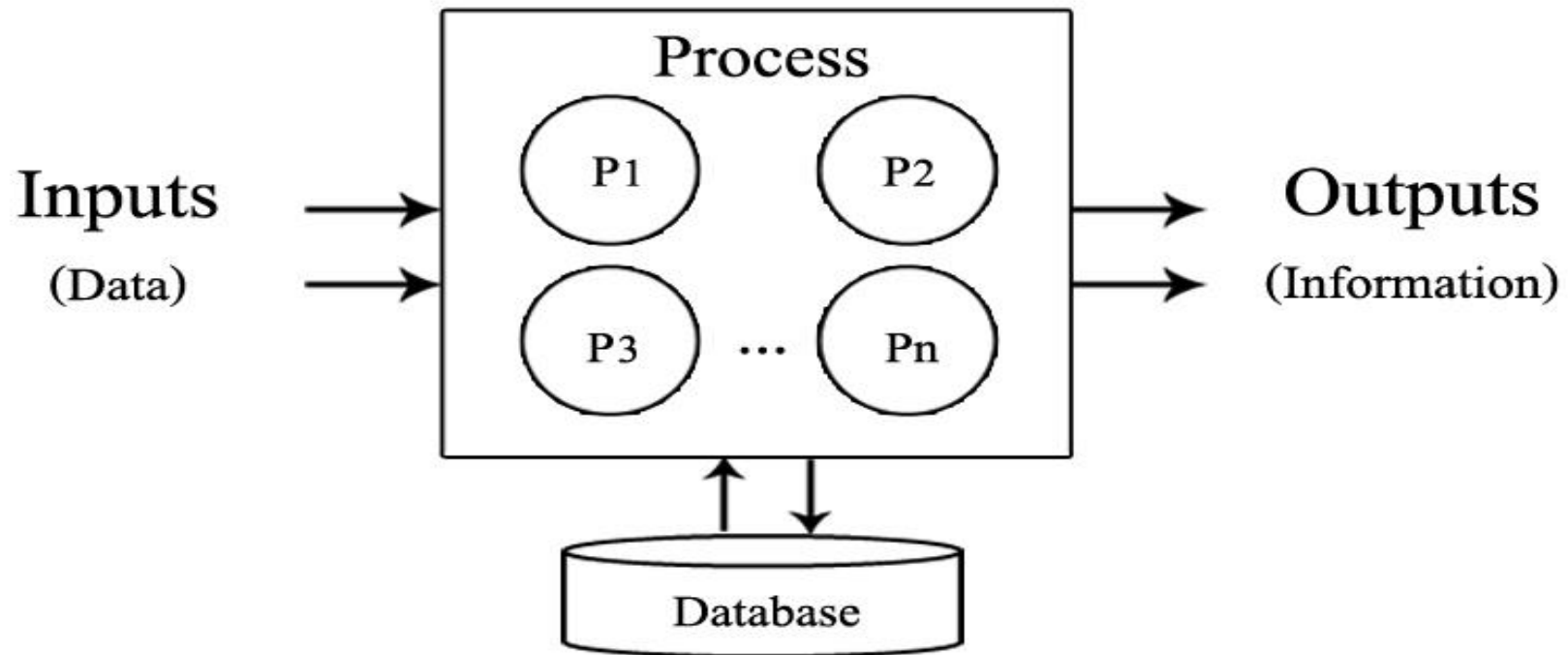
จุดประสงค์การเรียนรู้

1. อธิบายวิธีการเขียนโปรแกรมเชิงโครงสร้างและเชิงวัตถุได้
2. อธิบายองค์ประกอบของวัตถุและอ็อบเจกต์ได้
3. อธิบายกระบวนการพัฒนาเชิงวัตถุได้
4. อธิบายภาษาจูลีเอ็มแอลได้
5. ประยุกต์การเขียนภาษาโปรแกรมเชิงวัตถุได้

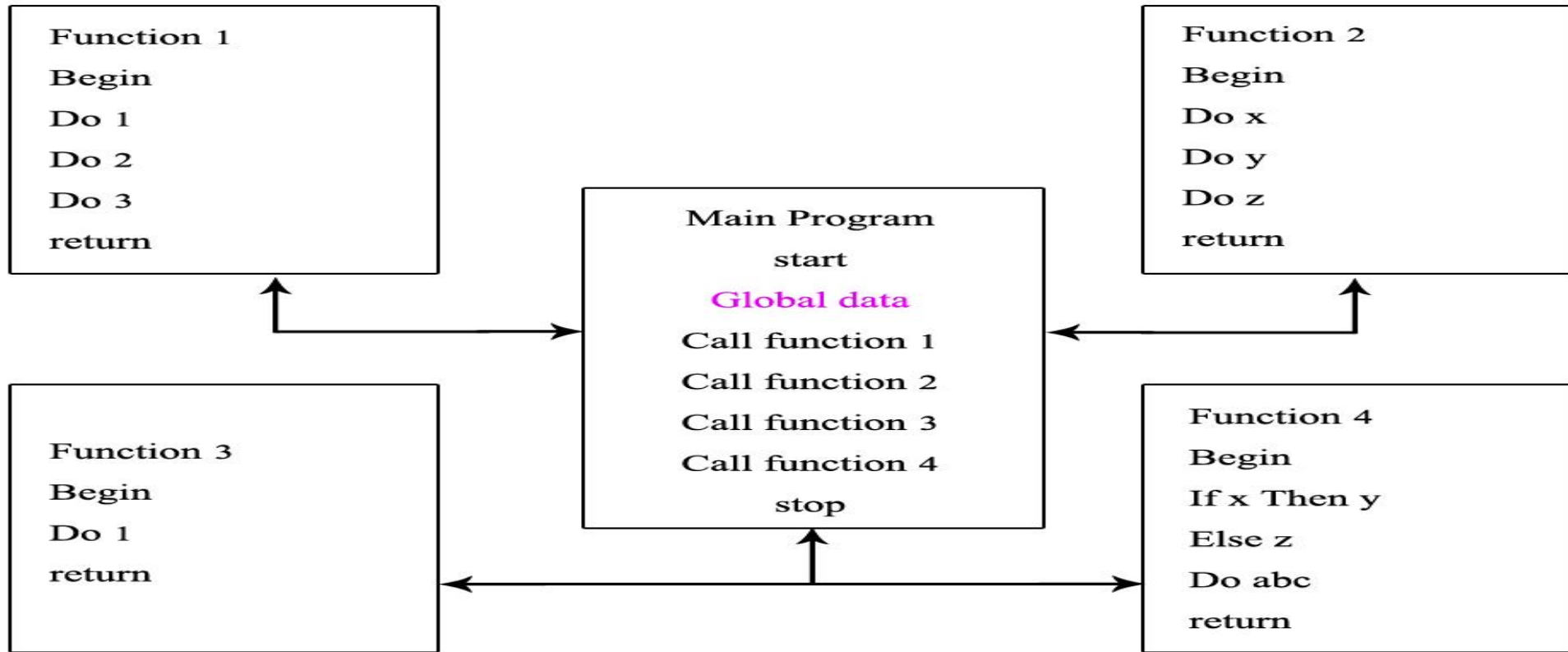
สมรรถนะการเรียนรู้

1. แสดงความรู้เกี่ยวกับหลักการพื้นฐานเชิงวัตถุ
2. แสดงความรู้เกี่ยวกับแนวคิดเชิงวัตถุ
3. ปฏิบัติการเขียนภาษาโปรแกรมเชิงวัตถุ

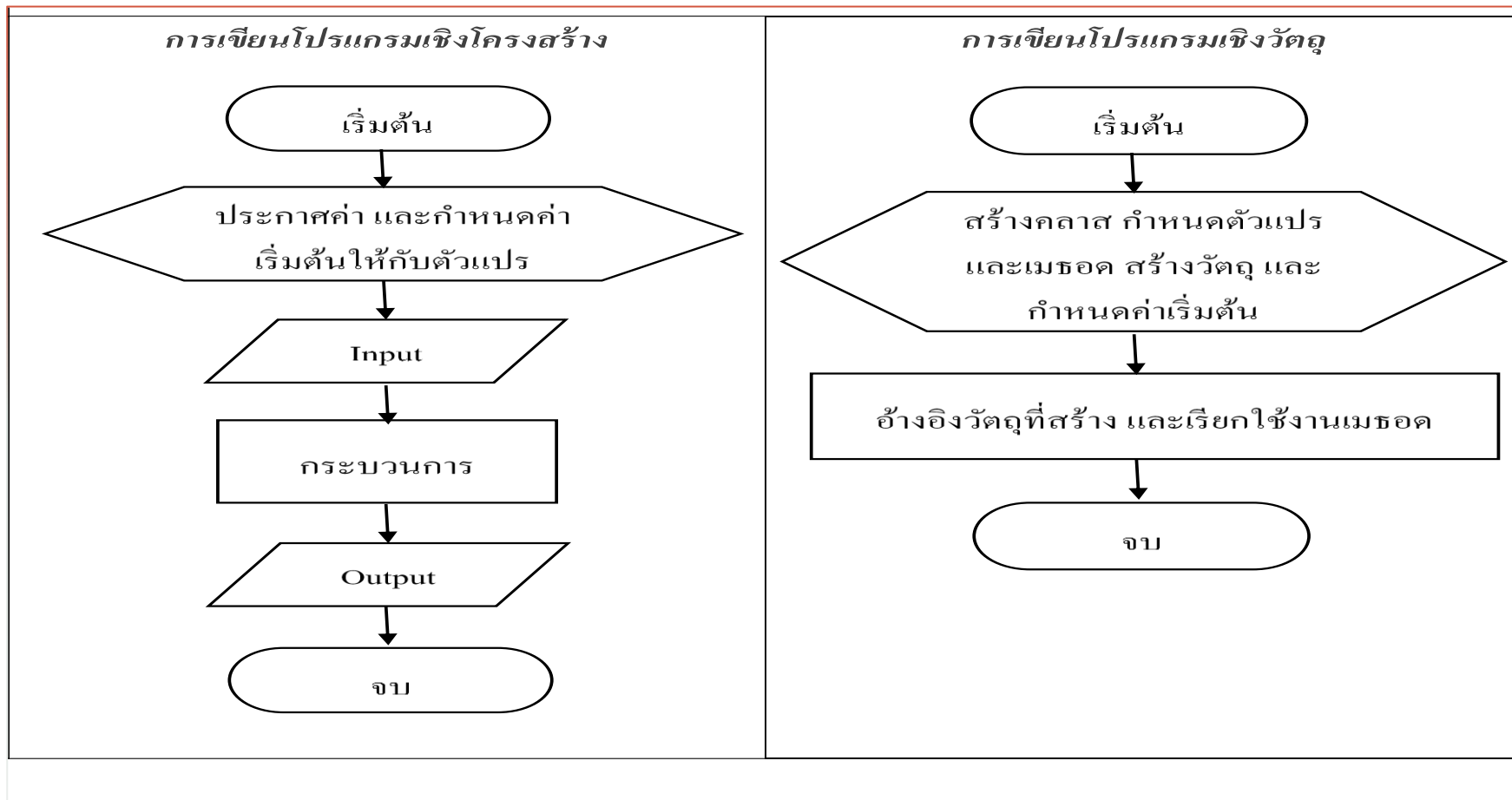
การเขียนโปรแกรมเชิงโครงสร้าง (Structured System Development)



การเขียนโปรแกรมเชิงวัตถุ (Object-Oriented System Development)



การเปรียบเทียบขั้นตอนการเขียนโปรแกรมเชิงโครงสร้าง และการเขียนโปรแกรมเชิงวัตถุ



วัตถุหรืออ็อบเจกต์ (object)

วัตถุ (Objects) คือ สิ่งใด ๆ ก็ตามซึ่งมีคุณลักษณะ (Attribute) ป่งบอกถึงความเป็นตัวของตัวเองในขณะนั้น และสามารถแสดงพฤติกรรม (Behavior) ของตัวเองออกมาได้ในชีวิตประจำวันเมื่อมองดูรอบตัว พบเห็นวัตถุต่าง ๆ วัตถุที่สามารถมองเห็นได้และจับต้องได้ (Tangible Objects)

คุณสมบัติทั่วไปของอ็อบเจกต์

การแบ่งโปรแกรมออกเป็นงานอ็อบเจกต์ต่างๆ โดยที่แต่ละอ็อบเจกต์มีความสมบูรณ์ในตัวเองทำให้ เพิ่มความสามารถของซอฟต์แวร์คือ การนำเอาโค้ดกลับมาใช้อีก หรือสามารถเรียกใช้งานอ็อบเจกต์ได้ หลาย ๆ ครั้ง ซึ่งโดยหลักการแล้วอ็อบเจกต์จะถูกเรียกใช้งานในช่วงไหนก็ได้ระหว่างที่โปรแกรมทำงานอยู่

องค์ประกอบของวัตถุหรืออ็อบเจกต์

1. การสืบทอด/สืบทอด (Inheritance)
2. การควบคุมการเข้าถึง (Encapsulation)
3. การกำหนดผลลัพธ์ของพฤติกรรมที่แตกต่างกันในสถานะที่ต่างกัน (Polymorphism)

องค์ประกอบของวัตถุหรืออ็อบเจกต์

1. การสืบทอด/สืบทอด (Inheritance)
2. การควบคุมการเข้าถึง (Encapsulation)
3. การกำหนดผลลัพธ์ของพฤติกรรมที่แตกต่างในสภาวะที่ต่างกัน (Polymorphism)



ความสัมพันธ์ระหว่างอ็อบเจกต์

1. ความสัมพันธ์แบบ Is-A Relationship
2. ความสัมพันธ์แบบ Has-A Relationship
3. ความสัมพันธ์แบบ Association
4. ความสัมพันธ์แบบ Composition
5. ความสัมพันธ์แบบ Aggregation

องค์ประกอบสำคัญของแนวคิดเชิงวัตถุ

- 1) การวิเคราะห์ ปัญหาโดยอาศัยแนวคิดเชิงวัตถุ (Object-Oriented Analysis: OOA)
- 2) การออกแบบเชิงวัตถุ (Object-oriented Design: OOD)

กระบวนการพัฒนาเชิงวัตถุ

1. การรวบรวมและศึกษาความต้องการของระบบ (Requirement Gathering)
2. การกำหนดคลาสเอนทิตี (Class Entity Identification)
3. เก็บข้อมูลพฤติกรรมของคลาส (Class Behavior)
4. สร้างความสัมพันธ์ระหว่างคลาส (Class Relationship)
5. สร้างโมเดลของคลาส (Class Modeling)

แนวคิดรวบยอดของอ็อบเจกต์

แนวคิดรวบยอดของอ็อบเจกต์ (Concept) หมายถึง แนวความคิดในข้อเท็จจริงต่าง ๆ ที่เราสามารถอธิบายลักษณะและพฤติกรรมของวัตถุนั้น ๆ ภายใต้อกรอบแนวคิดของโดเมนที่เรากำลังพิจารณาในการทำงาน โดยไม่รวมถึงความรู้สึก

Abstraction และ Encapsulation

Abstraction คือ การกำจัดข้อมูลที่ไม่เกี่ยวข้องทิ้งไปให้เหลือข้อมูลที่ต้องการเท่านั้น การแทนวัตถุที่มีอยู่ในโลกความเป็นจริงลงในระบบคอมพิวเตอร์ต้องพิจารณาถึงคุณสมบัติที่สำคัญของวัตถุเท่านั้น เป็นไปไม่ได้ที่จะพิจารณาคงสมบัติทุกข้อมาของวัตถุที่มีอยู่จริง

Encapsulation เป็นองค์ประกอบที่สำคัญอีกประการหนึ่งของ OOP ซึ่งหมายถึงการนำการปฏิบัติการ (Operation) รวมเป็นส่วนหนึ่งของวัตถุ เหตุที่เป็นเช่นนี้เพราะว่าต้องการซ่อนข้อมูลของวัตถุไว้ ผู้ใช้งานไม่จำเป็นต้องทราบถึงรายละเอียดต่าง ๆ ของวัตถุนั้น เพียงทราบถึงวิธีการจะใช้วัตถุนั้นก็เพียงพอ

ภาษายูเอ็มแอล (UML)

ยูเอ็มแอล (UML) ย่อมาจาก Unified Modeling Language เป็นภาษาที่ใช้อธิบายแบบจำลองต่าง ๆ หรือเป็นภาษาสัญลักษณ์รูปภาพมาตรฐาน สำหรับใช้ในการสร้างแบบจำลองเชิงวัตถุ โดยยูเอ็มแอล เป็นภาษามาตรฐานสำหรับสร้างแบบพิมพ์เขียวให้แก่ระบบงาน



ภาษาโปรแกรมเชิงวัตถุ

นักเขียนโปรแกรมบางคนคิดว่าการเขียนโปรแกรมขนาดใหญ่ บางครั้งก็เป็นงานที่หนักและเสียเวลามาก จึงได้พยายามคิดหาวิธีที่จะทำให้การเขียนโปรแกรมนั้นง่ายขึ้น และสามารถเขียนได้อย่างรวดเร็ว ทำให้เกิดเทคนิค การเขียนโปรแกรมเชิงวัตถุ (Object-Oriented Programming) หรือ OOP ขึ้นเพื่อช่วยลดความยุ่งยากของการเขียนโปรแกรม Object-Oriented Programming ต่างจากการเขียนโปรแกรมโดยทั่ว ๆ ไป

Visual Basic



ภาษา Visual Basic พัฒนาโดย Prof.Kemeny และ Kurtz ที่เมือง Dartmouth เกิดขึ้น ในปี ค.ศ. 1960 โดยมีจุดประสงค์สำหรับใช้สอนในห้องคอมพิวเตอร์ เมื่อมีการพัฒนาเครื่องไมโครคอมพิวเตอร์ขึ้นในยุคแรก ๆ จะมีหน่วยความจำไม่เพียงพอที่จะทำงานกับโปรแกรมภาษาอื่น



JAVA



Java[™]

จาวา เป็นภาษาใหม่ที่มาแรงมาก ภาษาจาวา ได้รับการพัฒนาขึ้นโดย บริษัท ซันไมโคร ซิสเต็มส์ ในปี ค.ศ. 1991 โดยมีเป้าหมายที่จะสร้างผลิตภัณฑ์ อิเล็กทรอนิกส์ สำหรับผู้บริโภคที่ใช้ง่าย มีค่าใช้จ่ายต่ำ ไม่มีข้อผิดพลาด และสามารถใช้กับเครื่องใด ๆ ก็ได้ ซึ่งสิ่งเหล่านี้ก็ได้กลายเป็นข้อดีของจาวา ที่เหนือกว่าภาษาอื่น ๆ โดยเฉพาะอย่างยิ่ง ในการที่โปรแกรมซึ่งเขียนขึ้นด้วยจาวาสามารถนำไปใช้กับเครื่องต่าง ๆ โดยไม่ต้องทำการคอมไพล์โปรแกรมใหม่

จบการนำเสนอ

