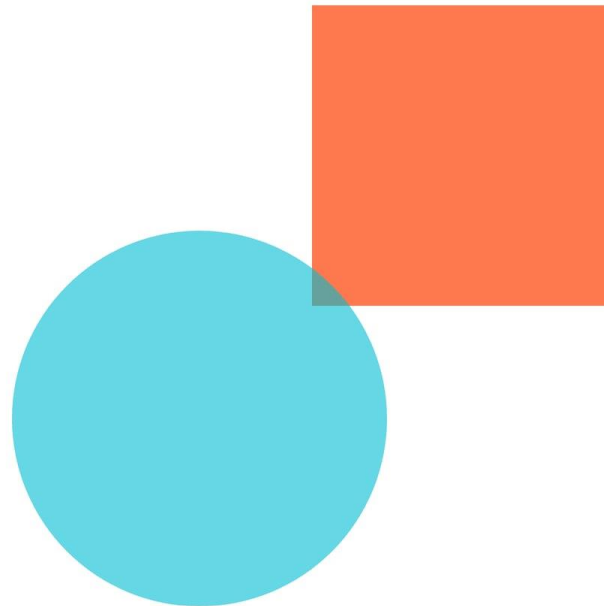


CPE1201

OBJECT-ORIENTED PROGRAMMING

ผศ.ดร. รวี อุตตมธนนินทร์

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์และเทคโนโลยีอุตสาหกรรม
มหาวิทยาลัยราชภัฏสวนสุนันทา



ผศ.ดร. รวิ อุตตมธนินทร์

Asst. Prof. Dr. Ravi Uttamathanin



สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์และเทคโนโลยีอุตสาหกรรม

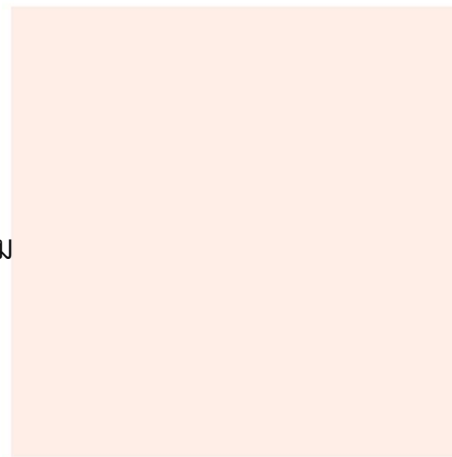


ravi.ut@ssru.ac.th





elfit.ssru.ac.th/ravi_ut


INSTRUCTOR PROFILE



COURSE DESCRIPTION



 ศึกษาหลักการและแนวคิดของการโปรแกรมเชิงวัตถุ (Object-Oriented Programming Paradigm) เพื่อการพัฒนาซอฟต์แวร์ที่มีประสิทธิภาพ

 การวิเคราะห์และออกแบบระบบโดยใช้แนวคิดเชิงวัตถุ การใช้งานคลาส (Classes) และวัตถุ (Objects)

คุณสมบัติหลัก 4 ประการของ OOP:

1. Encapsulation

2. Inheritance

3. Polymorphism

4. Abstraction

 การจัดการข้อผิดพลาด (Exception Handling) และการเขียนโปรแกรมเชิงโครงสร้างร่วมกับวัตถุ

01 Conceptual Understanding
เข้าใจหลักการพื้นฐานและแนวคิดของการโปรแกรมเชิงวัตถุ (OOP Paradigm) อย่างลึกซึ้ง

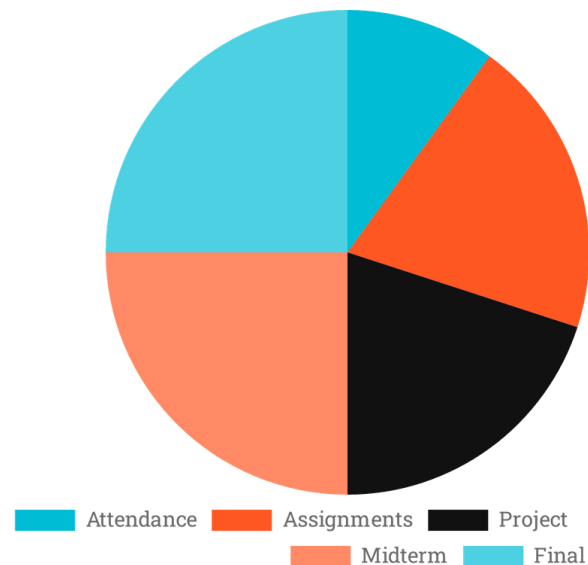
03 Engineering Application
ประยุกต์ใช้แนวคิด OOP ในการวิเคราะห์และแก้ปัญหาทางวิศวกรรมคอมพิวเตอร์ได้อย่างมีประสิทธิภาพ

02 Practical Development
สามารถออกแบบและพัฒนาโปรแกรมโดยใช้ภาษาที่รองรับ OOP เช่น Java หรือ C++ ได้อย่างคล่องแคล่ว

04 Advanced Readiness
เตรียมความพร้อมสำหรับการเรียนในวิชาขั้นสูง เช่น การออกแบบซอฟต์แวร์ และโครงสร้างข้อมูล

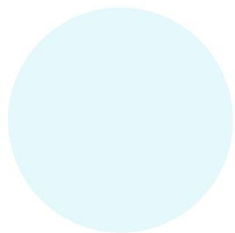
LEARNING OBJECTIVES

การเข้าชั้นเรียนและส่วนร่วม	10%
งานมอบหมายและแบบฝึกหัด	20%
โครงการกลุ่ม (Project)	20%
สอบกลางภาค	25%
สอบปลายภาค	25%



* นักศึกษาต้องมีเวลาเข้าเรียนไม่น้อยกว่า 80% จึงจะมีสิทธิ์สอบ

ASSESSMENT & GRADING



INTRODUCTION TO OOP

OOP คืออะไร?

เป็นรูปแบบการเขียนโปรแกรม (Programming Paradigm) ที่มองทุกอย่างเป็น "วัตถุ" (Object) ซึ่งประกอบด้วยข้อมูล (Data) และพฤติกรรม (Behavior)

ทำไมต้อง OOP?



Reusability

ช่วยให้โค้ดนำกลับมาใช้ใหม่ได้ง่าย ลดความซ้ำซ้อน



Maintainability

ง่ายต่อการบำรุงรักษาและแก้ไขในระยะยาว



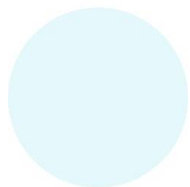
Scalability

เหมาะสำหรับการพัฒนาซอฟต์แวร์ขนาดใหญ่และซับซ้อน



Real-world Modeling

สะท้อนภาพความเป็นจริงของโลกผ่านการออกแบบวัตถุ



CLASS & OBJECT

CLASS

เปรียบเสมือน "พิมพ์เขียว" (Blueprint) หรือแม่แบบที่กำหนดโครงสร้างและพฤติกรรม

OBJECT

เปรียบเสมือน "สิ่งที่สร้างขึ้นจริง" (Instance) จากพิมพ์เขียวนั้น มีตัวตนและข้อมูลจริง

ตัวอย่าง: คลาสรถยนต์ (Car Class)

คุณลักษณะ (Attributes)

- ยี่ห้อ (Brand)
- รุ่น (Model)
- สี (Color)
- ความเร็วสูงสุด (Max Speed)

พฤติกรรม (Behaviors)

- วิ่ง (Drive)
- หยุด (Stop)
- เลี้ยว (Turn)
- เร่งความเร็ว (Accelerate)

Objects: รถ Toyota สีแดง ทะเบียน กข-123, รถ Honda สีน้ำเงิน ทะเบียน คม-456