

## เค้าโครงการศึกษา(Course Syllabus)

ชื่อวิชา: คพ.300 สัมมนาการโปรแกรมภาคปฏิบัติ

CS 300 Seminar in Practical Programming

ผู้ประสานงาน: อาจารย์สิริกัญญา นิลพานิช ([sirikanya\\_n@sci.tu.ac.th](mailto:sirikanya_n@sci.tu.ac.th))

คณาจารย์ผู้รับผิดชอบการสัมมนาและการทดสอบ	ผู้ช่วยดูแลการสอบ
ผู้ช่วยศาสตราจารย์. ดร.วนิดา พุทธิวิทยา	คุณนรินทร์ ทรัพย์สมบูรณ์
อาจารย์ นุชชากร งามเสาวรส	คุณประณต ผาสุกชีวิน
ผู้ช่วยศาสตราจารย์. ดร.เสาวลักษณ์ วรรณภา	คุณกรรณิการ์ ทรัพย์สมบูรณ์
อาจารย์ ดร. ธาปนา บุญชู	คุณณลินี ทรัพย์สมบูรณ์
ผู้ช่วยศาสตราจารย์. ดร. กษิตศ ชาญเขียว	คุณสมสุข ล้อมอัม
อาจารย์ สิริกัญญา นิลพานิช	คุณพิมพ์วรรณ อินฟ้าแสง

เว็บไซต์รายวิชา: <http://courses.cs.tu.ac.th>

วิชาบังคับก่อน: สอบได้ คพ.111

คำอธิบายรายวิชา:

สัมมนาการออกแบบและการโปรแกรมคอมพิวเตอร์เชิงโครงสร้างและเชิงวัตถุเพื่อการแก้ไขปัญหาให้ปฏิบัติได้จริง สำหรับปัญหาพิเศษทางวิทยาการคอมพิวเตอร์ การวัดผลแบ่งเป็น 2 ระดับ คือ ระดับS (ใช้ได้) และ ระดับU (ใช้ไม่ได้)

Seminar of design and practical programming with structural and object-oriented paradigms as qualification for special topics in Computer Science. Measurement and evaluation in this course can results in two levels: “S” (Satisfactory) and “U” (Unsatisfactory).

วิธีการศึกษา:1 หน่วยกิต (1-0-3) ในการศึกษารายวิชานี้ นักศึกษาจะต้อง

- o เข้าสัมมนา 3 ชั่วโมง
- o ศึกษาด้วยตนเองฝึกทำโจทย์สัปดาห์ละ3 ชั่วโมงเพื่อเตรียมทักษะสำหรับสอบปฏิบัติเขียนโปรแกรม
- o สอบปฏิบัติเขียนโปรแกรม (สอบข้อเขียน และ สอบปากเปล่า)

การวัดผล: เก็บคะแนนตามแผนการประเมินผลการเรียนดังนี้

สอบปฏิบัติการเขียนโปรแกรม (Programming Exit Examination) 100%

1. คะแนนสอบปฏิบัติส่วนข้อเขียน น้อยกว่า 80% ได้เกรด U
2. คะแนนสอบปฏิบัติส่วนข้อเขียนอยู่ระหว่าง 80 – 100% ลงสอบปากเปล่าเพื่อทดสอบความเข้าใจจากส่วนข้อเขียน
  - a. ไม่ผ่านการสอบปากเปล่า ได้เกรด U
  - b. ผ่านการสอบปากเปล่า ได้เกรด S

หนังสืออ้างอิง:

- [1] Spraul, A. V., *Think Like a Programmer: An Introduction to Creative Problem Solving*. 2012: No Starch Press.
- [2] Dale, N. and C. Weems, *Programming and Problem Solving with C++: Brief Edition*, 5th Edition. 2010: Jones & Bartlett Learning.
- [3] Lindsey, C. S., et al., *JavaTech: An Introduction to Scientific and Technical Computing with Java*. 2005: CAMBRIDGE University Press.
- [4] Horstmann, C., *Java Concepts 6th Edition*. 2010: John Wiley & Sons.
- [5] Garrido, J., *Introduction to Elementary Computational Modeling: Essential Concepts, Principles, and Problem Solving*. 2011: Chapman and Hall/CRC.

### วัตถุประสงค์ของข้อสอบปฏิบัติการเขียนโปรแกรม

1. สามารถทำความเข้าใจโจทย์วิเคราะห์เพื่อหาขั้นตอนวิธี (อัลกอริทึม) ในการผลิตผลลัพธ์ ตามข้อกำหนดที่โจทย์ระบุได้
2. สามารถเขียนโปรแกรมด้วยภาษาC, C++ หรือJava ตามขั้นตอนที่ออกแบบไว้จนได้ผลลัพธ์ตามที่โจทย์กำหนดได้อย่างถูกต้อง

## แผนการศึกษา CS300 (2/2564)

\*\*\*\* เนื่องจากสถานการณ์ COVID-19 ในเทอมนี้รูปแบบ/ตารางเวลาการทดสอบระบบและการสอบอาจมีการปรับเปลี่ยนตามสถานการณ์ \*\*\*\*

วันที่	รายละเอียดกิจกรรม	หมายเหตุ
7 กุมภาพันธ์ 2565	เข้ารับฟังการชี้แจงการสอบ ผ่าน <u>online meeting</u>	นักศึกษาที่ลงทะเบียนวิชานี้ต้องเข้าร่วมทุกคน เวลา 16.30 น.
21 กุมภาพันธ์ 2565	ทดสอบระบบ ผ่าน online (ภาคพิเศษ ช่วงเช้า, ภาคปกติ ช่วงบ่าย)	นักศึกษาที่ลงทะเบียนวิชานี้ต้องเข้าร่วมทุกคน ผู้ที่ไม่เข้าทดสอบระบบ จะไม่มีสิทธิ์หักทวง เรื่องผลการสอบใดๆทั้งสิ้น
7-9 มีนาคม 2565	ลงทะเบียนขอสอบในรอบที่ 1	
11 มีนาคม 2565	ประกาศรายชื่อผู้มีสิทธิ์สอบในรอบ 1	
13 มีนาคม 2565	<u>สอบข้อเขียนรอบที่ 1 ***</u>	สอบพร้อมกันทุกกลุ่ม เวลา 13.00 น.
4-6 เมษายน 2565	ลงทะเบียนขอสอบในรอบที่ 2	
8 เมษายน 2565	ประกาศรายชื่อผู้มีสิทธิ์สอบในรอบ 2	
10 เมษายน 2565	<u>สอบข้อเขียนรอบที่ 2 ***</u>	สอบพร้อมกันทุกกลุ่ม เวลา 13.00 น.
2-4 พฤษภาคม 2565	ลงทะเบียนขอสอบในรอบที่ 3	
6 พฤษภาคม 2565	ประกาศรายชื่อผู้มีสิทธิ์สอบในรอบ 3	
8 พฤษภาคม 2565	<u>สอบข้อเขียนรอบที่ 3 ***</u>	สอบพร้อมกันทุกกลุ่ม เวลา 13.00 น.
	<u>สอบปากเปล่า ***</u>	เฉพาะผู้ที่ได้คะแนนสอบข้อเขียนในรอบ 1-3 ตั้งแต่ 80% ขึ้นไป (วัน-เวลาตามนัดหมายในแต่ละรอบ)

### หมายเหตุ

1. แผนการศึกษาและกำหนดการอาจถูกปรับ/เปลี่ยนเพื่อส่งเสริมให้การดำเนินการเรียนการสอนเกิดผลสัมฤทธิ์ทางการศึกษาสูงสุด
2. หากนักศึกษาไม่เข้าสอบตามรอบการสอบที่ลงทะเบียนไว้ จะถือว่านักศึกษาขาดสอบ และไม่ผ่านการสอบในรอบนั้น

## ขอบเขตเนื้อหาที่ใช้ออกข้อสอบ

1. ชนิดข้อมูล (Data types)
  - 1.1. จำนวนเต็ม (Integer) และชนิดข้อมูลแบบบูลีน(Boolean)
  - 1.2. จำนวนจริง(Floating point numbers)
  - 1.3. ตัวอักษร (Characters)
2. การกำหนดค่าคงที่
  - 2.1. การกำหนดค่าคงที่ซึ่งเป็นจำนวนเต็ม จำนวนจริง และตัวอักษรโดยใช้ตัวเลขฐานสิบ
  - 2.2. การกำหนดค่าคงที่จำนวนเต็มจำนวนจริง โดยใช้ค่าบิตของข้อมูลฐานสอง,ฐานแปดหรือฐานสิบหก
  - 2.3. การกำหนดค่าคงที่ตัวอักษร โดยใช้รหัสแอสกี (ASCII code)
  - 2.4. การกำหนดค่าคงที่โดยใช้ชนิดข้อมูลแบบเรียงลำดับ (Enumerated type)
3. คำสั่งคำนวณ
  - 3.1. การคำนวณแบบพีชคณิต (Arithmetic operations) บนข้อมูลจำนวนเต็ม,จำนวนจริงโดยใช้โอเปอเรเตอร์ทาง พีชคณิต (Arithmetic operators) ได้แก่ บวก ลบ คูณ หาร หารเอาเศษ
  - 3.2. การคำนวณแบบพีชคณิต บนข้อมูลแบบบิต โดยใช้โอเปอเรเตอร์แบบบิต (Bitwise Arithmetic Operators) ได้แก่ AND, OR, NOT, Exclusive OR, Shift Left, Shift Right
4. โครงสร้างควบคุมลำดับการประมวลผลโปรแกรม (program control structures)
  - 4.1. การประมวลผลแบบลำดับ (Sequence structures) : การกำหนดค่าให้ตัวแปร (assignment) การรับข้อมูลจาก คีย์บอร์ด หรือแฟ้มข้อมูล,การแสดงผลข้อมูลไปที่จอภาพหรือแฟ้มข้อมูล
  - 4.2. การเลือกเส้นทางประมวลผลตามเงื่อนไข (Selection structures) : การใช้ If/then และ If/then/else ทั้งแบบเดี่ยว แบบซ้อน (nested if) และแบบเรียงต่อ (cascaded if), การใช้ Case หรือ Switch
  - 4.3. การประมวลผลแบบทำซ้ำ (Iteration structures): การใช้โครงสร้างรูปแบบทดสอบเงื่อนไขก่อน (For, while, for each), การใช้โครงสร้างรูปแบบทดสอบเงื่อนไขภายหลัง (do..while หรือ repeat...until)
5. การแบ่งโปรแกรมเป็นส่วนที่มีความอิสระต่อกันโดยใช้โมดูลเช่น เมธอด (Methods) หรือฟังก์ชัน (Functions)
6. การเขียนโปรแกรมโดยสร้างโครงสร้างแบบซับซ้อน หรือผู้ใช้กำหนดเอง (Complex Data Type or User Defined Type)
  - 6.1. อาร์เรย์ (Array) หนึ่งมิติและหลายมิติ
  - 6.2. สตรัค (Struct)
7. การประมวลผลข้อความ (String)
  - 7.1. กำหนดค่าใช้ตัวแปรเก็บข้อความ
  - 7.2. การเปรียบเทียบ,การค้นหาข้อความย่อย (Substring Search), การแทนที่, การต่อ, การตัดข้อความ
8. การจัดการแฟ้มข้อมูลแบบข้อความ(Text file)