

CP352-353 (1/2567): กิจกรรม 04 (30 คะแนน)

วันที่ 6 พฤศจิกายน 2567/กำหนดส่ง 12 พฤศจิกายน 2567 ก่อน 23:55น

1. เขียนอธิบายเกณฑ์ในการวัดคุณภาพของซอฟต์แวร์ (Problem-domain class diagram) ดังต่อไปนี้ โดยใช้คำพูดตามความเข้าใจของนิสิต (คะแนนรวม 11 คะแนน)

* (a) Coupling ตามความเข้าใจของนิสิตมันใช้วัดคุณภาพด้านไหน? (1 คะแนน)

- อธิบาย Interaction coupling เป็น coupling เกี่ยวกับอะไร ยกตัวอย่าง coupling ที่อยู่ในชนิดนี้ควรมีลักษณะอย่างไร จึงมีคุณภาพในระดับดีที่สุด และในระดับที่แย่ที่สุด (2 คะแนน)
- Inheritance coupling เป็น coupling เกี่ยวกับอะไร ยกตัวอย่าง coupling ที่อยู่ในชนิดนี้ควรมีลักษณะอย่างไร จึงมีคุณภาพในระดับดีที่สุด และในระดับที่แย่ที่สุด (2 คะแนน)

* (b) Cohesion ตามความเข้าใจของนิสิตมันใช้วัดคุณภาพด้านไหน? (1 คะแนน)

- Method cohesion เป็น cohesion เกี่ยวกับอะไร ยกตัวอย่าง cohesion ที่อยู่ในชนิดนี้ควรมีลักษณะอย่างไร จึงมีคุณภาพในระดับดีที่สุด และในระดับที่แย่ที่สุด (2 คะแนน)
- Class cohesion เป็น cohesion เกี่ยวกับอะไร ยกตัวอย่าง cohesion ที่อยู่ในชนิดนี้ควรมีลักษณะอย่างไร จึงมีคุณภาพในระดับดีที่สุด และในระดับที่แย่ที่สุด (2 คะแนน)

* a.) ໄວ້ໃຫ້ຄວາມສົມຜັນຮະບູກວ່າງ class / ສ່ວນປະກອບຂອງ software ກ່າວກຳຈິງກັນວຸກໝາຍແດ້ໄຟ

i.) ເປີແກວເຊື່ອມໄລ ຂະວະນັງ class ຕັ້ນທີ່ສ້ອງຕັ້ງຂໍ້ຕົວສ່ວນຮະບູກ

↳ ຕັ້ນທີ່ຈະການເຊື່ອມໄລໃຈແຈ້ງຕົວທີ່ສ້ອງຕົວ ເພື່ອ class ຜ່ານມາຂອງຄາຍເນວັບທີ່ສູດ ດຣ. ຕັ້ນ interface

ເກົ່າດີໃນຮະບູກທີ່ສ້ອງຕົວ ດັ່ງ class ລັດຖະບູກຂອງລາຍງານໂຄງໂລດ ນວຍອີ່ນ method ພາບນອກທີ່ສູບຮັບຮັບ

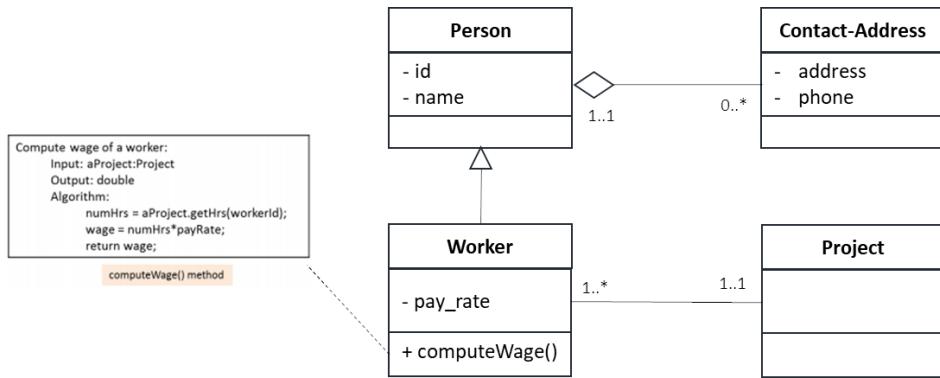
ii.) ເກົ່າຊ່ວຍເຫຼົາການ Inheritance class ອອນທີ່ຄວບສັບກຳດັກຈາກ class ມີລັດຖະບູກທີ່ສ້ອງຕົວໃຈ ໂດຍເຊື່ອມໄລໃຈແຈ້ງຕົວທີ່ສູດ ພະຍາກົດສິນທີ່ໄວ້ດັກຈາກ class ອັດໃຈ code ດາວໂຫຼວດ ແລ້ວ ກຳເຊີນ / ປຸ່ມແກ້ວດ

b.) ກ່າວກຳທີ່ກຳຕົວຢ່າງໃນ class / method ກ່າວບັນຫຼັກທີ່ກຳຕົວຢ່າງໃນ ຕັ້ນທີ່ມີ Cohesion ສູງ ທີ່ໄຟເຫັນໄດ້ແລ້ວຢັງວັກຫາຕ່ອງ

i.) ກ່າວວັນດີອາຍານັ້ນທີ່ກຳຕົວຢ່າງໃນ method ສ້ອງຄວາມກຳເນັ້ນຍົງ ອ່າງເລື່ອງແກະຕ້ອງຮັດເຢັນຕອນບັດ ສົງເຫຼົາການ Single Responsibility ທີ່ກຳຕົວຢ່າງໃນ code ສົບຮັບນັດແລະເຫັນຢັງຍຸດ

ii.) ກ່າວວັນດີທີ່ຂອງ method ແລ້ວ property ກຳຕົວຢ່າງໃນ class ສ້ອງຄວາມເກື່ອງຂັງແລະສົກສົນກັນ class ທີ່ຕີ້ອງຢັ້ງຢູ່ກຳຕົວຢ່າງໃນ ກົດຕັ້ງຂໍ້ມູນ ສ່ວນ class ທີ່ມີລາຍລັບທີ່ ແລ້ວ method ທີ່ໄໝເກື່ອງຂັງກົດຕັ້ງກົດຕັ້ງ

2. เขียน method contract และ specification ในรูปที่ 2 และ 3 ให้สมบูรณ์สำหรับ computeWage() method ของคลาสชื่อ Worker ในรูปที่ 1 โดยสมมติว่า computeWage() method ถูกเรียกใช้ภายใน method หนึ่งของคลาส Project และ use case ที่เกี่ยวข้องคือ Create project cost report สำหรับค่าอื่นๆ ที่เหลือ สำหรับ method contract เช่น description, arguments, returned type, pre-conditions และ post-conditions ให้นิยิตลอง สมมติขึ้นเอง รวมถึงค่าต่างๆใน method specification เช่นกัน (คะแนนรวม 10 คะแนน)



รูปที่ 1: Structured english for computeWage()

(a) Method contract (5 คะแนน)

Method Name: ComputeWage()	Class Name: Worker	ID: 999
Clients (Consumers): Project		
Associated Use Cases: Create project cost report		
Description of Responsibilities: คำนวณเงินเดือนของพนักงานตามจำนวนชั่วโมงทำงานและอัตราค่าจ้างที่กำหนดไว้ สำหรับเชิงบัญชีการสร้างรายงานต้นทุนโครงการ (Create project cost report use case)		
Arguments Received: workHours : อัตราเดือนชั่วโมงการทำงานที่ได้รับมา (ประเภท float) wageRate : อัตราค่าจ้าง / อัตราเดือนของพนักงาน (ประเภท float)		
Type of Value Returned: Float		
Pre-conditions: workHours ≥ 0 wageRate ≥ 0		
Post-conditions: result ต้องเป็น ≥ 0 (ผลลัพธ์ต้องเป็น正值)		

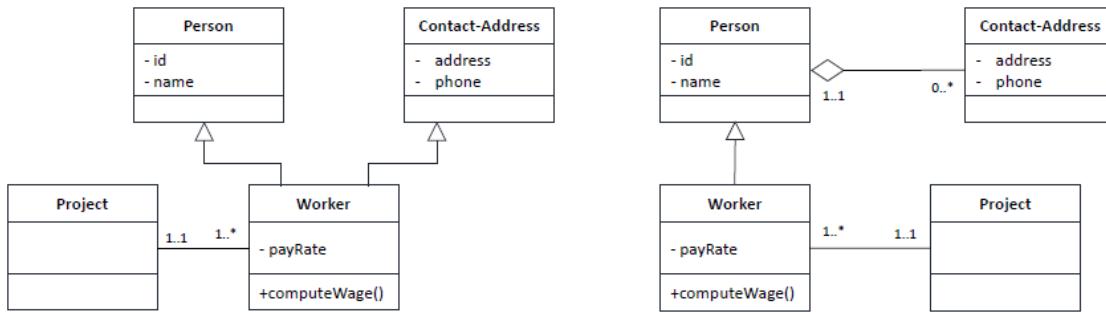
รูปที่ 2: Method contract

(b) Method specification (5 คะแนน)

Method Name: <u>ComputeWage()</u>	Class Name: <u>Worker</u>	ID: <u>999</u>
Contract ID: <u>999</u>	Programmer: <u>Thatchaya Sirivasesee</u>	Date Due: <u>19/11/2567</u>
Programming Language:	<input type="checkbox"/> Visual Basic <input type="checkbox"/> Smalltalk <input type="checkbox"/> C++ <input checked="" type="checkbox"/> Java	
Triggers/Events: <u>Create project cost report</u>		
Arguments Received: Data Types:	Notes:	
<u>wageRate : float</u>	<u>หักห้าสิบต่อหนึ่งชั่วโมง</u>	
<u>workHours : float</u>	<u>จำนวนชั่วโมงที่ทำงานต่อหนึ่งวัน</u>	
Messages Sent & Arguments Passed: ClassName.MethodName:	Data Type:	Notes:
<u>Worker computeWage (wageRate, workHours)</u>	<u>wageRate : float</u>	<u>หักห้าสิบต่อหนึ่งชั่วโมง</u>
	<u>workHours : float</u>	<u>จำนวนชั่วโมงที่ทำงานต่อหนึ่งวัน</u>
Arguments Returned: Data Type:	Notes:	
<u>totalWage : float</u>	<u>จำนวนเงินที่ได้จากการทำงาน</u>	
Algorithm Specification:	<ol style="list-style-type: none"> 1. ต้องรับ input WorkHours ค่า จำนวนชั่วโมง + 2. ต้องรับ input WageRate ค่า จำนวน + 3. คำนวณเงินเดือนโดย formula $WorkHours \times WageRate$ 4. คืนค่า totalWage 	
Misc. Notes:		

รูปที่ 3: Method specification

3. เขียนอธิบายความแตกต่างของ class diagram ในรูปที่ 4 พร้อมทั้งบ่งชี้ว่า class diagram ได้ออกแบบได้ดีและถูกต้องกว่า ในเรื่องของ inheritance และ aggregation (4 คะแนน)

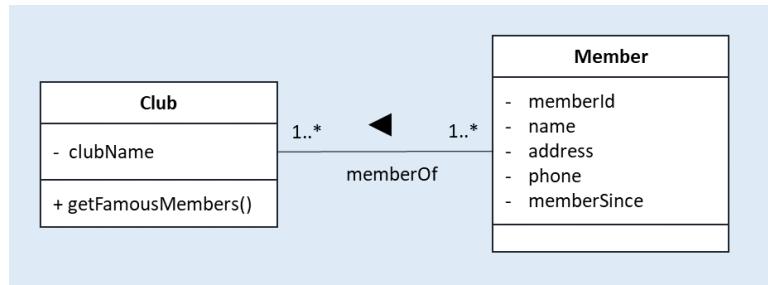


รูปที่ 4: Class diagrams

Inheritance: ตรวจสอบว่า ความสืบทอดนั้นแบบ inheritance ใน diagram class ถูกใช้อย่างถูกต้องหรือไม่ Ex. class ลูก ควรสืบทอด คุณสมบัติจาก class แม่ ที่ไม่เหมาะสม แล้วมีซ้ำกันในแม่ (เช่น ชื่อบน) หรือต้องก่อตัวท่องเที่ยมส่วนตัว ตรวจสอบว่าใช้คำอ่านชื่อเดียวกันที่ใช้ใน diagram class หรือไม่ Inheritance ที่จะแสดงถึงการสืบทอดเดียว ก็ต้องมีชื่อเดียวกันที่ใช้ในสอง class อีกด้วย เช่น ให้ class ทำงาน เว็บไซต์ และ class ลูก ทำงานเว็บไซต์ ก็ต้องใช้ชื่อเดียวกันใน diagram class

Aggregation: ตรวจสอบว่า aggregation ถูกใช้ในกรณี class ที่มีความสัมพันธ์กันอย่างถูกต้องหรือไม่ โดยทั่วไป aggregation ควรแสดงถึงเป็นราก ความสัมพันธ์ของ aggregation ควรแสดงถึงเป็นราก ความสัมพันธ์ของ aggregation ที่มีความสัมพันธ์กันอย่างถูกต้อง เช่น class A เป็นรากของ class B ใน diagram class A สามารถแสดง aggregation ที่สัมภพันธ์กันในตัวของ自身 ความสัมพันธ์นี้จะถูกใช้ประโยชน์ เช่น การจัดเก็บและจัดหางาน ฉันขอรับใช้ใน aggregation ที่มีความสัมพันธ์กันใน aggregation

4. ออกแบบฐานข้อมูลเชิงสัมพันธ์ (RDBMS) และชุดของคลาสสำหรับเข้าถึงข้อมูล (DAM class) สำหรับไดอะแกรมคลาสที่ ประกอบไปด้วยคลาส Member และคลาส Club โดยทั้งสองคลาสเป็น concrete class ดังแสดงในรูปที่ 5 (5 คะแนน)



รูปที่ 5: Club-Member class diagram

RDBMS

ตาราง Club

- ClubID : เป็น Primary Key ที่ไม่ซ้ำกันในแต่ละ club
- ClubName : ชื่อของ club
- Location : สถานที่ตั้ง club

ตาราง Member

- MemberID : เป็น Primary Key ที่ไม่ซ้ำกันในแต่ละ member
- Name : ชื่อ ของ member
- JoinDate : วันที่เข้าร่วม club
- ClubID : เป็น Foreign Key ที่โยง เชื่อม ระหว่าง ตาราง club และ member และ club ในฐาน

DAM Classes

ClubDAM : จัดการข้อมูล Club

- addClub (ClubID, ClubName, Location) : เพิ่ม club ใหม่
- updateClub (ClubID, ClubName, Location) : แก้ไขเดาข้อมูลของ club
- deleteClub (ClubID) : ลบ club ตาม ClubID
- getClub (ClubID) : ดึงข้อมูลของ club ตาม ClubID

MemberDAM : จัดการข้อมูล Member

- addMember (MemberID, Name, JoinDate, ClubID) : เพิ่ม member ใหม่
- updateMember (MemberID, Name, JoinDate, ClubID) : แก้ไขเดาข้อมูล member
- deleteMember (MemberID) : ลบ member ตาม MemberID
- getMember (MemberID) : ดึงข้อมูล member ตาม MemberID
- getMemberByClub (ClubID) : ดึงข้อมูล member ที่อยู่同一个 club ตาม ClubID