

ชื่อ-นามสกุล \_\_\_\_\_ รหัส \_\_\_\_\_ ตอน \_\_\_\_\_

คณะวิศวกรรมศาสตร์  
มหาวิทยาลัยสงขลานครินทร์

---

สอบปลายภาค : ภาคเรียนที่ 1	ปีการศึกษา : 2554
วันที่ : 3 กรกฎาคม 2554	เวลา : 9:00 – 12:00 น.
รหัสวิชา : 241-421	ห้องสอบ: S201
ชื่อวิชา : Client/Server Distributed Systems	ผู้สอน: อ.วรพต

---

คำสั่ง :

- อนุญาตเครื่องเขียน (ปากกาหรือดินสอ)
  - ไม่อนุญาตหนังสือ ตำราเรียนหรือเอกสารใดๆ และเครื่องคิดเลข
  - เขียนคำตอบลงในข้อสอบ หากไม่พอเขียนต่อด้านหลังได้ โดยต้องเขียนบอกให้เห็นได้อย่างชัดเจน
  - ข้อสอบมี 2 ตอน จำนวน 8 หน้า (ไม่รวมปก) ตอนที่ 1 มี 16 ข้อ ตอนที่ 2 มี 3 ข้อ
- คะแนนรวม 30 คะแนน

ทุจริตในการสอบ โทษขั้นต่ำคือปรับตกในรายวิชาที่ทุจริต และพักการเรียน 1 ภาคการศึกษา

**ตอนที่ 1** จงตอบคำถามต่อไปนี้ (16 คะแนน)

1. จงอธิบายความแตกต่างของโมเดลแบบ 2-tiers และ 3-tiers และข้อดีข้อเสียของแต่ละแบบ

.....

.....

.....

.....

.....

.....

.....

.....

2. ลักษณะของ Fat server กับ Fat client ต่างกันอย่างไร มีข้อดีข้อเสียอย่างไร

.....

.....

.....

.....

.....

3. จงอธิบาย distributed programming โดยยกตัวอย่างประกอบ

.....

.....

.....

.....

.....

.....

.....

4. จงอธิบายความแตกต่างของ Peer to Peer แบบ Reader Centric กับ Publisher Centric

.....

.....

.....

.....

.....

.....

.....

5. อธิบายการใช้ไฟล์ lock กับการเปิดไฟล์แบบ exclusive ในการควบคุมการเข้าถึง shared file

.....

.....

.....

.....

.....

6. จงเปรียบเทียบข้อดีข้อเสียในการใช้โปรโตคอล TCP และ UDP ในการติดต่อระหว่างเครื่องคอมพิวเตอร์ พร้อมทั้งยกตัวอย่างลักษณะการใช้งานที่เหมาะสมกับโปรโตคอลทั้งสอง

.....

.....

.....

.....

.....

7. จงอธิบายปัญหาที่อาจจะเกิดขึ้น 2 ปัญหาเมื่อทำการอ่านข้อมูลจากเครือข่ายโดยใช้ฟังก์ชัน read และบอกวิธีแก้ไข

.....

.....

.....

.....

.....

8. จงอธิบายหลักการเขียนโปรแกรม concurrent server โดยใช้ฟังก์ชัน fork และ select

.....

.....

.....

.....

.....

.....

.....

.....

9. จงอธิบายความจำเป็นในการส่งข้อมูลในเครือข่ายในรูปแบบ network byte order และอธิบายลักษณะของ network byte order มาพอสังเขป

.....  
.....  
.....  
.....

10. จงอธิบายขั้นตอนในการสร้าง Socket เพื่อรอรับการติดต่อบนเครื่อง server โดยใช้ฟังก์ชัน socket(), bind() และ listen()

.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....

11. ในกรณีที่มีการติดต่อจาก client จำนวน 3 เครื่องไปที่ server ซึ่งรันโปรแกรม hangman ในแบบ iterative ไว้ จงอธิบายว่าจะเกิดผลอย่างไรขึ้นที่ client ทั้งสาม

.....  
.....  
.....  
.....

12. จงบอกหน้าที่ของฟังก์ชัน accept และอธิบายการนำไปใช้

.....  
.....  
.....  
.....

13. ในการแก้ไขโปรแกรม Hangman server แบบ iterative ให้เป็นแบบ concurrent สามารถทำได้ง่ายที่สุดอย่างไร จงอธิบายมาพอสังเขป

.....

.....

.....

.....

14. Signal ชื่อ SIGCHLD เกิดขึ้นเมื่อใด การตรวจจับทำได้อย่างไร

.....

.....

.....

.....

15. จงอธิบายหลักการของ Processor Farm พร้อมทั้งยกตัวอย่างประกอบ

.....

.....

.....

.....

16. การคำนวณหาจำนวนตัวเลขจำนวนเฉพาะ ที่มีค่าในช่วงตั้งแต่ 1 – 24 ล้าน โดยแบ่งเป็นช่วงย่อย 3 ช่วงเท่ากันส่งไปคำนวณหาผลลัพธ์ของแต่ละช่วงย่อยบน server จำนวน 3 เครื่อง โดยส่งข้อมูลไปคำนวณพร้อมกัน จงอธิบายว่า

- จะได้ผลลัพธ์สุดท้ายสัมพันธ์กับความเร็วของเครื่องแต่ละเครื่องอย่างไร
- การสับเปลี่ยนช่วงย่อยของข้อมูลที่ส่งไปคำนวณบน server แต่ละเครื่อง มีผลต่อความเร็วในการหาผลลัพธ์สุดท้ายหรือไม่ อย่างไร

.....

.....

.....

.....



.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

2. จงออกแบบรูปแบบการสื่อสารและแสดงโค้ดในการจัดรูปแบบข้อมูลสื่อสารบนเครื่อง Client สำหรับติดต่อ Server เพื่อประมวลผลเมตริกซ์ของตัวเลขจำนวนเต็มขนาดไม่เกิน 10 แถว 10 หลัก ผลลัพธ์เป็น ทรานสโพสของเมตริกซ์ที่ส่งไป กำหนดตัวแปรเพื่อแสดงโค้ดดังนี้ (5 คะแนน)

```
int A[10][10]; // เมตริกซ์ข้อมูลที่ส่งให้ server
int B[10][10]; // เมตริกซ์ผลลัพธ์ที่ได้รับจาก server
int m, n; // ขนาดแถวและหลักของเมตริกซ์ข้อมูล A ตามลำดับ
```

นักศึกษาสามารถประกาศตัวแปรเพิ่มเติมได้ตามต้องการ

แนะนำ การทรานสโพสของเมตริกซ์ (Transpose of Matrix)

ทรานสโพสของเมตริกซ์ A เขียนแทนด้วย  $A^t$

ตัวอย่าง  $A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$  จะได้ว่า  $A^t = \begin{bmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{bmatrix}$

ซึ่งขนาดของ A เป็น 2 แถว 3 หลัก ส่วน  $A^t$  มีขนาด 3 แถว 2 หลัก

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

