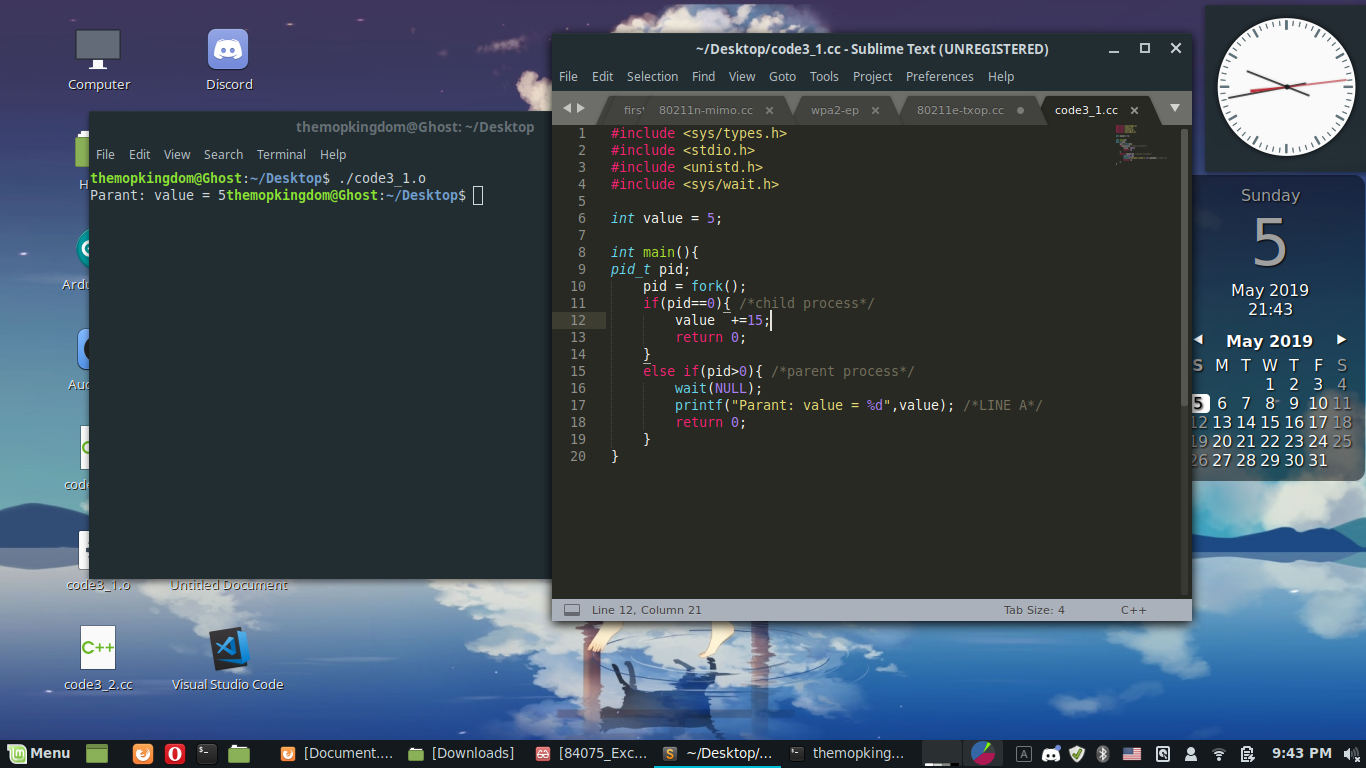
**Excercise\_3**

**ข้อที่ 1**

จากโค้ดภาษาซีด้านล่างต่อไปนี้ โปรแกรมที่บรรทัด Line A ให้เอาท์พุตเป็นค่าใด จงทดลองและอธิบายผลการทดลอง



การสร้าง ps ใหม่ทำให้มีการแชร์ทรัพยากรเป็นของตัวเองทำให้การรันคำสั่งต่าง แยกจากกันแบบอิสระแม้จะมีการสั่งเพิ่มค่าในอีก ps ก็ไม่เกิดผมในอีก ps เช่นกัน ดังจะเห็นได้จากที่มีการสั่ง เพิ่มค่า value ไปอีก 15 แต่เมื่อฝั่ง ps หลักรันออกมา ผลลัพธ์กลับเป็นค่าตั้งต้นที่ยังไม่ได้เพิ่มค่าขึ้นใน ps ลูก เพราะเมื่อสร้าง ps ใหม่จะทำงานแยกอิสระจาก ps เดิมโดยสิ้นเชิง  
และคำสั่งดังกล่าวมีการใช้ if ในการตรวจสอบว่า ps นั้นเป็นลูกหรือไม่ ทำให้ ps หลักไม่มีการทำคำสั่งในส่วนดังกล่าวแต่อย่างใด

**ข้อที่ 2**

จากโค้ ด ภาษาซี ต่อไปนี้

มีโปรเซสทั้งหมดกี่โปรเซส (นับรวมโปรเซสเริ่มต้นด้วย) จงอธิบาย

มีทั้งหมด 8 ps เพราะมีการ fork ทั้งสิน 3 ครั้ง

ครั้งแรก เพิ่มเป็น 2 ps

ครั้งสองเพิ่มเป็น 4 ps

ครั้งสามเพิ่มเป็น 8 ps สรุปรวมทั้งสิ้นมี 8 ps

เพราะทุกครั้งที่สร้าง ps ใหม่ ps ลูกจะได้ code ส่วนที่เหลือของแม่ไปด้วย ทำให้ ลูกก็สร้างเหลนขึ้นมาเรื่อย ๆ ตามจำจวนคำสั่ง fork() ที่เหลือไล่ลงมาจากคำสั่ง fork() ที่ใช้สร้างตังเอง ทำให้สุดท้ายแล้วมีทั้งสิ้น 8 ps นั่นเอง

**ข้อที่ 3**

เมื่อมีการสร้างโปรเซสใหม่โดยใช้คำสั่ง fork มีการแชร์สถานะ (State) ของสิ่งใดต่อไปนี้ Heap, Stack, และ Shared Memory Segment ระหว่างโปรเซสแม่กับโปรเซสลูก

มีการแชร์สถานะ Memory Segment เท่านั้น เพราะ ps แม่จะทำการส่ง code ส่วนที่เหลือที่ยังไม่ประมวลผลหลังใช้คำสั่ง fork ไปให้ลูกทำต่อไปด้วย

**ข้อที่ 4**

จงอธิบายความหมายของ Short-Term, Medium-Term และ Long-Term Scheduling

## Long Term Scheduler

เรีกอีกอย่างว่า **job scheduler[ตัวจักการงาน]** คอยพิจารณาว่าโปรแกรมใดบ้างที่ควรเข้ารับการประมวลผล โดยคัดเลือกจากคิวและทำการโหลดลงหน่วยความจำจุดเด่นหลัก ๆ คือการกำหนวตารางเวลาการใช้งาน ที่สมดุลกันเช่น ขอบเขตของการใช้งาน I/O และ cpu ว่าในขณะนั้น ๆ สามารถนำโปรแกรมไปประมวลผลโดยใช้งานส่วนใด ๆ ได้บ้าง

## Short Term Scheduler

เรียกอีกอย่างว่า CPU scheduler[ตัวจัดตารางงานของ cpu] ใช้ในการเพิ่มประสิทธิภาพของระบบตามเกณฑ์ที่เลือกใช้ เพื่อเปลี่ยนแปลงสถานะพร้อมเป็นสถานะทำงาน โดยเลือกจาก ps ที่พร้อมจะดำเนินการและจัดสรรให้ cpu นำ ps ดังกล่าวไปประมวลผล

## Medium Term Scheduler

หรือการจัดตารางงานระยะกลาง เป็นส่วนหนึ่งของการสับเปลี่ยนที่ทำการลบ ps ออกจากหน่วยความจำ และทำการลดระดับของการทำงานหลาย ๆ ps พร้อม ๆ กันลง และยังคอยจักการนำหรือสับเปลี่ยน ps ที่ต้องจัดการประมวลผลในช่วงเวลาหนึ่ง ๆ ด้วย กระบวนการประมวลผลใด ๆ อาจถูกระงับเมื่อมีการร้องของใช้งาน I/O หรือ cpu จาก ps อื่น ๆ เข้ามา และย้าย ps ที่กำลังประมวลผลในขณะนั้นไปเก็บไว้ในหน่วยความจำ แล้วนำมาประมวลผลใหม่เมื่อคิวเข้าใช้งานว่างลงแล้ว

**ข้อที่ 5**

จงอธิบายสิ่งที่ Kernel ทำเมื่อเกิด Context-Switch ระหว่างโปรเซส

-บันทึก context ของ process ที่กำลังทำงานอยู่บน CPU. อัพเดทค่า process control block และ fields ที่สำคัญอื่น ๆ

- ย้าย process control block ของกระบวนการดั่งกล่าวไปยัง relevant queue เช่น คิวพร้อมประมวลผล หรือ I/O คิว เป็นต้น

-เลือก process ใหม่ ที่จะนำมาทำการประมวลผล.

- อัพเดท process control block ของ process ที่เลือกมาประมวลผลและเปลี่ยนค่า state ของ process ดังกล่าวเป็น running

- อัพเดท ตัวจัดการข้อมูลบนหน่วยความจำตามต้องการ

- เรียกคืน กระบวนการก่อนหน้าที่เคยถูกประมวลผลอยู่บน cpu โดยการโหลดค่าก่อนหน้าของ process control block และ registers

**ข้อที่ 6**

โปรเซส init ในระบบปฏิบัติการ Unix หรือ Linxu คืออะไร มีหน้าที่อย่างไร จงอธิบายพอสังเขป

เป็นกระบวนการหรือ ps ตัวแรกที่จะทำการรันขึ้นมาครั้งแรกตั้งแต่เปิดใช้ระบบปฏิบัติการ โดยตัวโปรแกรมอื่น ๆ จะเป็นเหมือน ps ลูกของ ps ดังกล่าว และจะปิดตัวลง ก็ต่อเมื่อมีการปิดระบบปฏิบัติการ

**ข้อที่ 7**

Zombie Process และ Orphan Process คืออะไร เกิดขึ้นได้เมื่อใด จงอธิบาย

Zombie Process คือ การที่ child รันเสร็จเร็วเกิน terminate เร็วเกินไป Parent ยังไม่ waiting เลย พอถึง waiting ก็ไม่มีใครมาปลุกให้ทำงานต่อ ทำให้ Parent ต้อง waiting ตลอดไป เราเรียกอาการนี้โดยทั่วไปว่าอาการเครื่องแฮงค์

Orphan Process คือ เคสที่ Parent ทำการ terminate ไปแล้ว ในขณะที่ Child ยัง execute ไม่เสร็จ และไม่เกิดการ cascading (os อนุญาติให้ child มีชีวิตต่อ) กรณีนี้ เราเรียก child ว่า orphan

**ข้อที่ 8**

จากเรื่อง Interprocess Communication จงเขียนโปรแกรมเพื่อคำนวณ

Collatz Conjecture จากอินพุต n ที่มีค่าบวกใดๆ โดย n ตัวใหม่ คำนวณได้ ดังต่อไปนี้

จนกว่า n จะมีค่าเป็น 1

n = 3\*n + 1 if n is odd

n = n/2 if n is even

เช่น ถ้าเริ่มต้น n = 35 ผลลัพธ์ที่ได้คือ 35 , 106 , 53 , 160 , 80 , 40 , 20 , 10 , 5 , 16 , 8 , 4 ,2 , 1

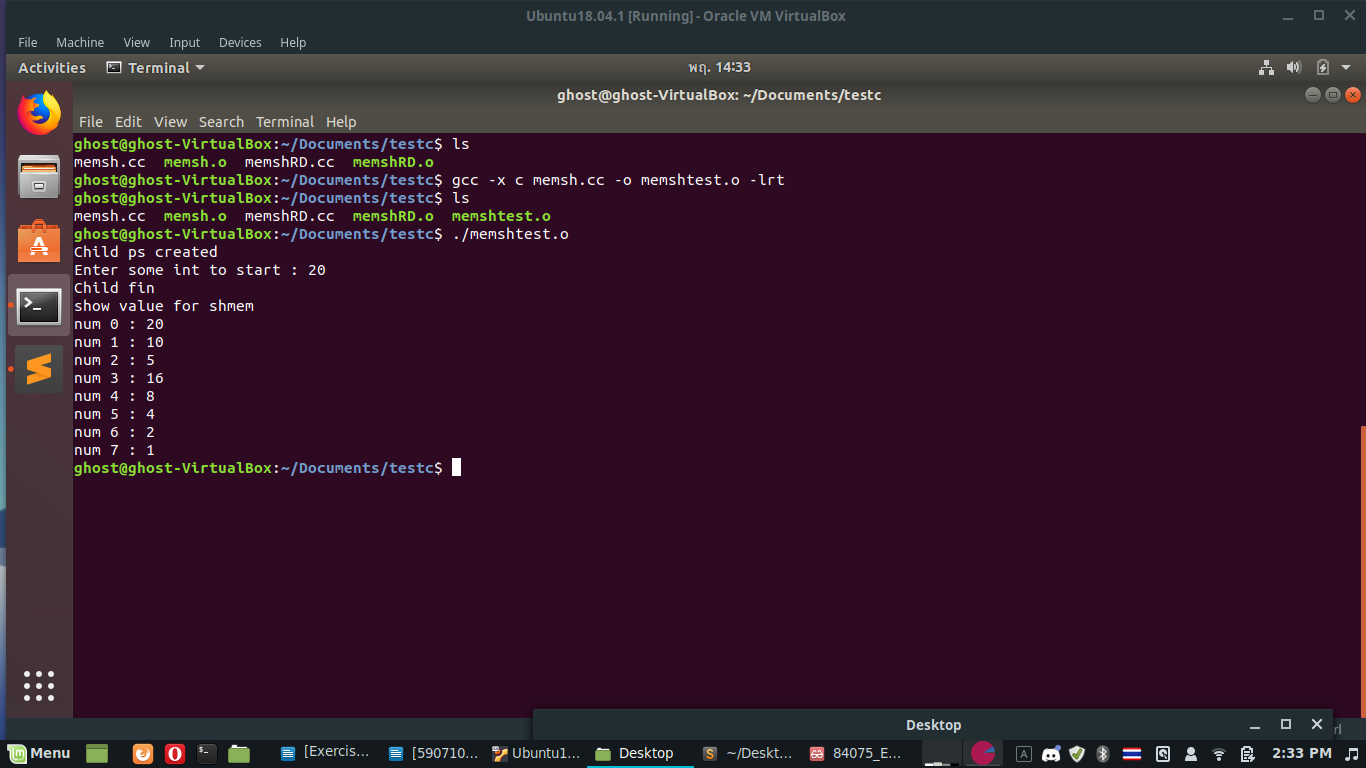
ให้เขียนโปรแกรมโดยใช้ภาษาซี และใช้ POSIX Shared Memory ในโปรเซสหลักมีขั้นตอนดังนี้

1. สร้าง Shared-Memory Object (shm open(), ftruncate(), and mmap())

2. สร้างโปรเซสลูกเพื่อคำนวณ n โดยที่โปรเซสลูกเขียนค่า n ที่คำนวณได้ในแต่ละรอบลงใน Shared-Memory Object

3.รอจนโปรเซสลูกเสร็จ แล้วแสดงผลข้อมูล จาก Shared-Memory Object ออกทางจอภาพ

4.ลบ Shared-Memory Object



**ไฟล์ code วางอยู่บน git ชื่อ memsh.cc**

**ข้อที่ 9**

จากเรื่อง

Interprocess Communication จงทดลองโปรแกรมภาษาจาวาต่อไปนี้ ( หมายเหตุ ให้รัน DateServer ก่อนแล้วจึงค่อยรัน DateClient)

a. โค้ดตัวอย่างข้างต้น มี Interprocess Communication ด้วยวิธีการใด

แบบ Socket

b. การรันโปรแกรมได้ผลลัพธ์อะไร จงอธิบายพฤติกรรมของโปรแกรมอย่างละเอียด

ผลลัพธ์ที่ได้คือวันที่และเวลาที่ อยู่ในเครื่อง server โดยผลลัพธ์ดังกล่าวมาจากคำสั่ง

java.util.Date()toString() ที่อยู่ทางฝั่ง server นั่นเอง

ไฟล์ฝั่ง server

ทำการสร้าง Socket ขึ้นมาที่ port 6013 แล้วรัน loop รอรับการเชื่อมต่อเข้ามาจาก port ุจๅ- fy’d]jk;

จากนั้นเมื่อฝั่งผู้ใช้เชื่อมต่อเข้ามาจำมีการส่ง outoptsteam จากฝั่ง server ออกไป ส่งค่ากลับไปเป็น วันที่และเวลาดั่งข้างต้น แล้วทำการปิดการเชื่อมต่อกับฝั่งผู้ใช้ และกลับไปรันคำสั่งบน oop ข้างต้นรอการเชื่อมต่ออีกครั้ง

โดย code ข้างต้นทั้งหมดมีการใช้ try catch ในการดักจับ error เอาไว้ด้วย

ไฟล์ฝั่ง client

ทำการเชื่อมต่อไปที่ ip ของ server ที่ port 6013 บน 127.0.0.1 ซึ่งคือ loopback ของเครื่องตนเอง

จากนั้นทำการรอรับไฟล์ outputsteam จากฝั่ง server ส่วนในการแสดงผลนั้นมีการทำดีเลย์โดยใช้ loop ครอบตัวแสดงผลค่า inputsteam ที่รับมาจาก outputsteam จากฝั่ง serve เอาไว้ แล้วทันทีที่แสดงผลเสร็จจะทำการปิดการเชื่อมต่อกับฝั่ง server ไป

โดย code ข้างต้นทั้งหมดมีการใช้ try catch ในการดักจับ error เอาไว้ด้วยเช่นกัน

c. จงอธิบายความหมายของ Socket

เป็นรูปแบบที่อยู่ในฝั่งที่ต้องการสื่อสารด้วยทั้งปลายทางและต้นทางโดยจะมาในรูปของ ip : port

และ socket ยังถือเป็น Interprocess Communication รูปแบบหนึ่งด้วย

d. หากรัน DateClient ก่อนการรัน DateServer ได้ผลลัพธ์อย่างไร

เกิด error จากการร้องขอการเชื่อมต่อเนื่องจากปลายทางดังกล่าว ไม่มีการเปิดโปรแกรมใด ๆ เพื่อรอฟังคำสั่งรอขอการเชื่อต่อเข้ามาอยู่ ซึ่งในที่นี้ก็คือ port 6013 นั่นเอง