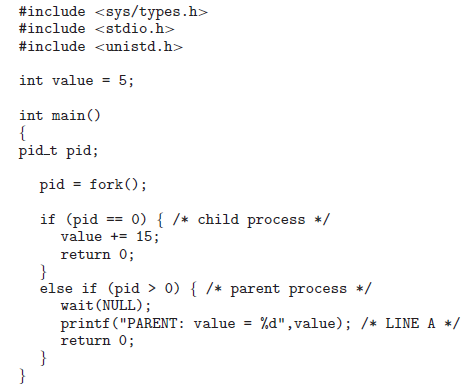
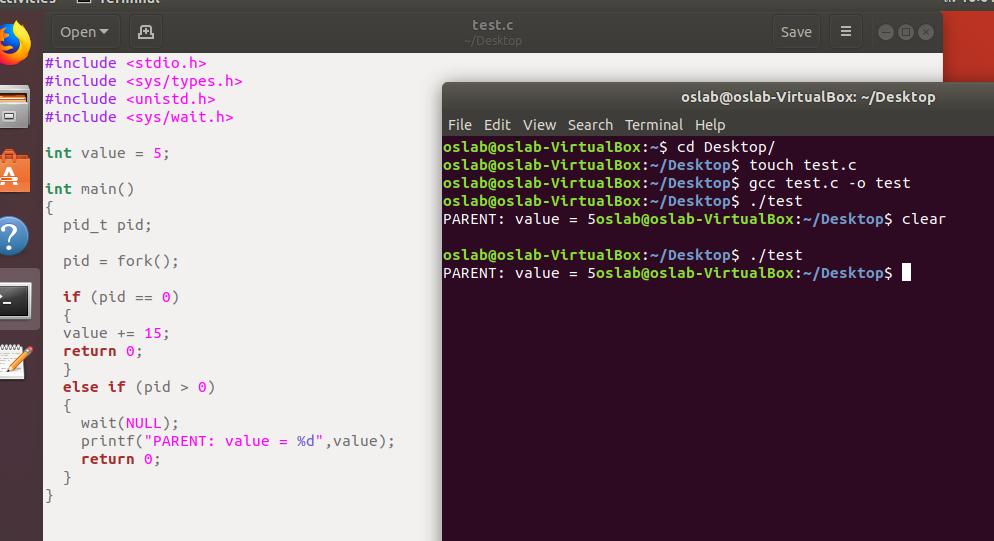
**แบบฝึกหัด บทที่ 3**

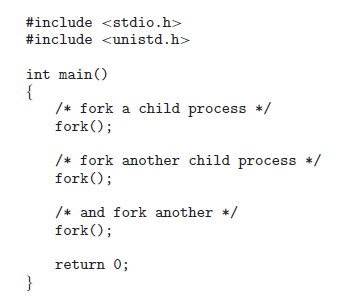
1. จากโค้ดภาษาซีด้านล่างต่อไปนี้ โปรแกรมที่บรรทัด Line A ให้เอาท์พุตเป็นค่าใด จงทดลองและอธิบายผลการทดลอง



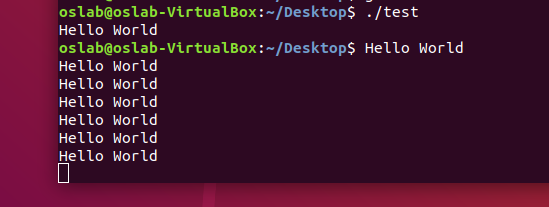
ตอบ หาก pid มีค่าเท่ากับ 0 จะทำที่ child process ถ้ามากกว่า 0 จะทำที่ child process ก่อนแล้วจึงทำ parent processที่บรรทัด LINE A Parent process มีการ print แสดงผลค่าของ Parent ออกมามีค่าเท่ากับ 5 เนื่องจาก value ที่เข้ามามีการตั้งค่าไว้ที่ 5 ส่วนของ child นั้นจะมีค่า 20 เนื่องจากการ fork เพียง 1 ครั้งแต่ในส่วนของ child process ไม่ได้มีการใช้คำสั่ง print ออกมากเพื่อแสดงผล



1. จากโค้ดภาษาซีต่อไปนี้ มีโปรเซสทั้งหมดกี่โปรเซส ( นับรวมโปรเซสเริ่มต้นด้วย ) จงอธิบาย



ตอบ จำนวนของ process ทั้งหมดสามารถหาได้จากจำนวนการ fork หากจำนวน fork = n จะมีจำนวน process เท่ากับ 2n ในกรณีตัวอย่างนั้นมีการ fork 3 ครั้งจะทำให้เกิด process ทั้งหมดคือ 8 process ( 2\*2\*2 = 8)



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

ตอบ มีการแชร์เฉพาะส่วนของ Memory Segment ระหว่าง parent process กับ child parent

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

ตอบ Short-term คือ ตัวจัดตารางระยะสั้น คือ เลือก Process ที่อยู่ในสถานะรอเพื่อนำมาใช้เป็นหน่วยประมวลผลกลาง

Medium-term คือ ตัวจัดการตารางระยะกลาง คือ ย้าย Process ออกจากหน่วยความจำหลักเพื่อลดจำนวน Process ที่มีมากเกินไปในหน่วยความจำ เพื่อทำให้ CPU ทำงานได้ดีและเร็วขึ้น

Long-term คือ ตัวจัดตารางระยะยาว คือ เลือก Process จากหน่วยเก็บข้อมูลเพื่อเข้าสู่หน่วยความจำ

1. จงอธิบายสิ่งที่ Kernel ทำเมื่อเกิด Context - Switch ระหว่างโปรเซส

ตอบ เมื่อเกิด Context - Switch จะทำการบันทึก save Process ที่ประมวลผลปัจจุบันที่อยู่สถานะ Running เอาไว้ และต่อมาจะทำการเรียก load Process อื่น ๆหรือ Process ที่ถูกบันทึกอยู่สถานะ Ready ไว้กลับขึ้นมาทำประมวลผลต่ออีกครั้งเมื่อและเมื่อ ประมวลผล Process ที่ทำอยู่เสร็จก็จะสลับไปกลับยัง Process อื่น ๆที่อยู่ถัดไปในคิว

1. โปรเซส init ในระบบปฏิบัติการ Unix หรือ Linx คืออะไร มีหน้าที่อย่างไรจงอธิบายพอสังเขป

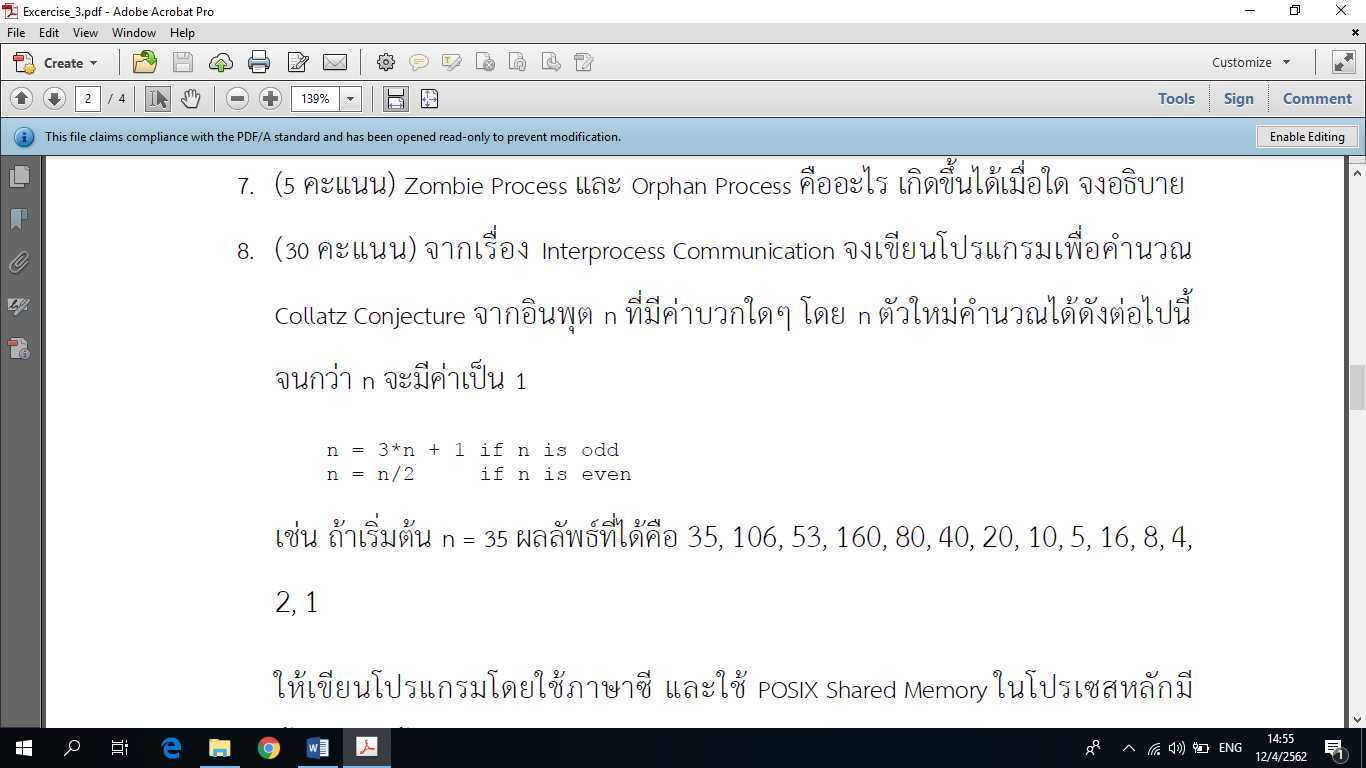
ตอบ init (ย่อมาจาก initialization) เป็นกระบวนการแรกที่เริ่มต้นในระหว่างการบูทระบบคอมพิวเตอร์ Init นั้นเป็น (a daemon process) ที่ยังทำงานต่อไปจนกว่าระบบจะปิด

1. Zombie Process และ Orphan Process คืออะไร เกิดขึ้นเมื่อใด จงอธิบาย

ตอบ Zombie process คือ โปรเซสที่คืนทรัพยากรและหน่วยความจำให้กับระบบแล้ว แต่ไม่สามารถกำจัดโปรเซสนี้ออกจากระบบได้ เนื่องจากโปรเซสพ่อแม่ของโปรเซสนี้ได้ถูกทำลาย  และ Init process ไม่สามารถจัดการกับโปรเซสนี้ได้ ทำให้ไม่สามารถทำลายโปรเซสออกจากระบบ กลายเป็นเสมือนผีดิบที่ฆ่าไม่ตาย

Orphan Process หรือ process กําพร้า คือ process ที่ child process อยู่ในระหว่างการ execute ยังไม่สามารถเข้าสู่สถานะ terminate แต่ในส่วนของ Parent process กลับเข้าสู่สถานะ terminate ก่อนแม้จะมีความผิดพลาดเกิดขึ้นแต่ process นี้ยังคงสมารถกำจัดอยู่ในระบบต่อไปและสามารถกำจัดได้

1. จากเรื่อง Interprocess Communication จงเขียนโปรแกรมเพื่อคำนวณ Collatz Conjecture จากอินพุต n ที่มีค่าบวกใด ๆ โดย n ตัวใหม่คำนวณได้ดังต่อไปนี้ จนกว่า n จะมีค่าเป็น 1

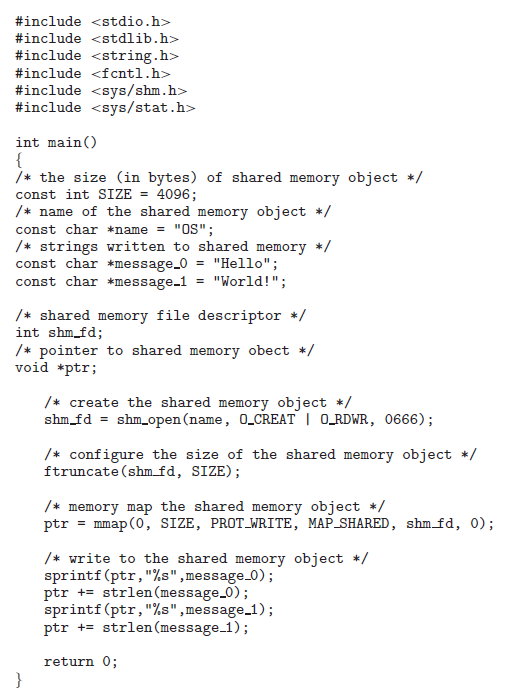


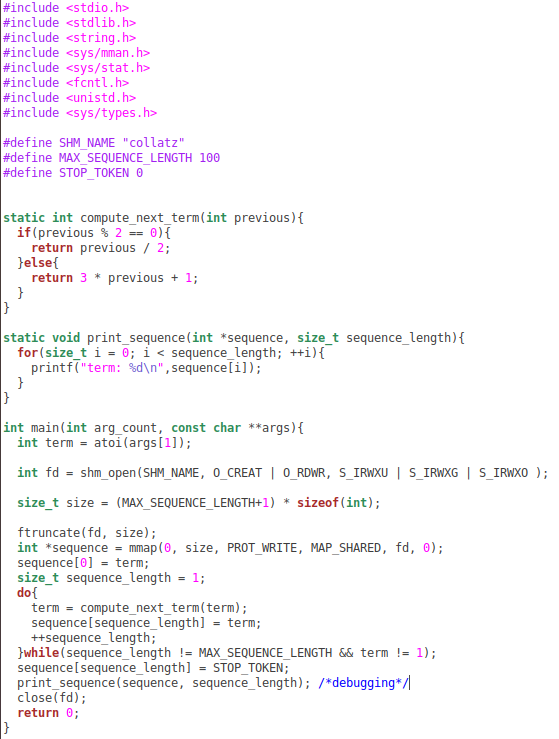
เช่น ถ้าเริ่ม n = 35 ผลลัพธ์ที่ได้คือ 35 , 106 , 53 , 160 , 80 , 40 , 20 , 10 , 5 , 16 , 8 , 4 , 2 , 1

ให้เขียนโปรแกรมโดยใช้ภาษาซี และใช้ POSIX Shared Memory ในโปรเซสหลักมีขั้นตอนดังนี้

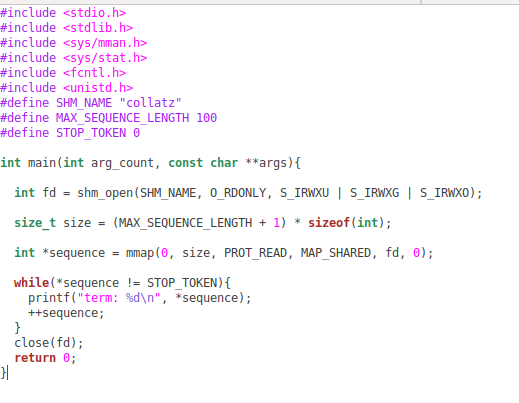
1. สร้าง Shared-Memory Object ( shm open () , ftruncate () , and mmap () )
2. สร้างโปรเซสลูกเพื่อคำนวณ n โดยโปรเซสลูกเขียนค่า n ที่คำนวณได้ในแต่ละรอบลงใน Shared-Memory Object
3. รอจนโปรเซสลูกเสร็จ แล้วแสดงผลขอมูลจาก Shared-Memory Object ออกทางจอภาพ
4. ลบ Shared-Memory Object

ตัวอย่างการใช้ Shared-Memory Object

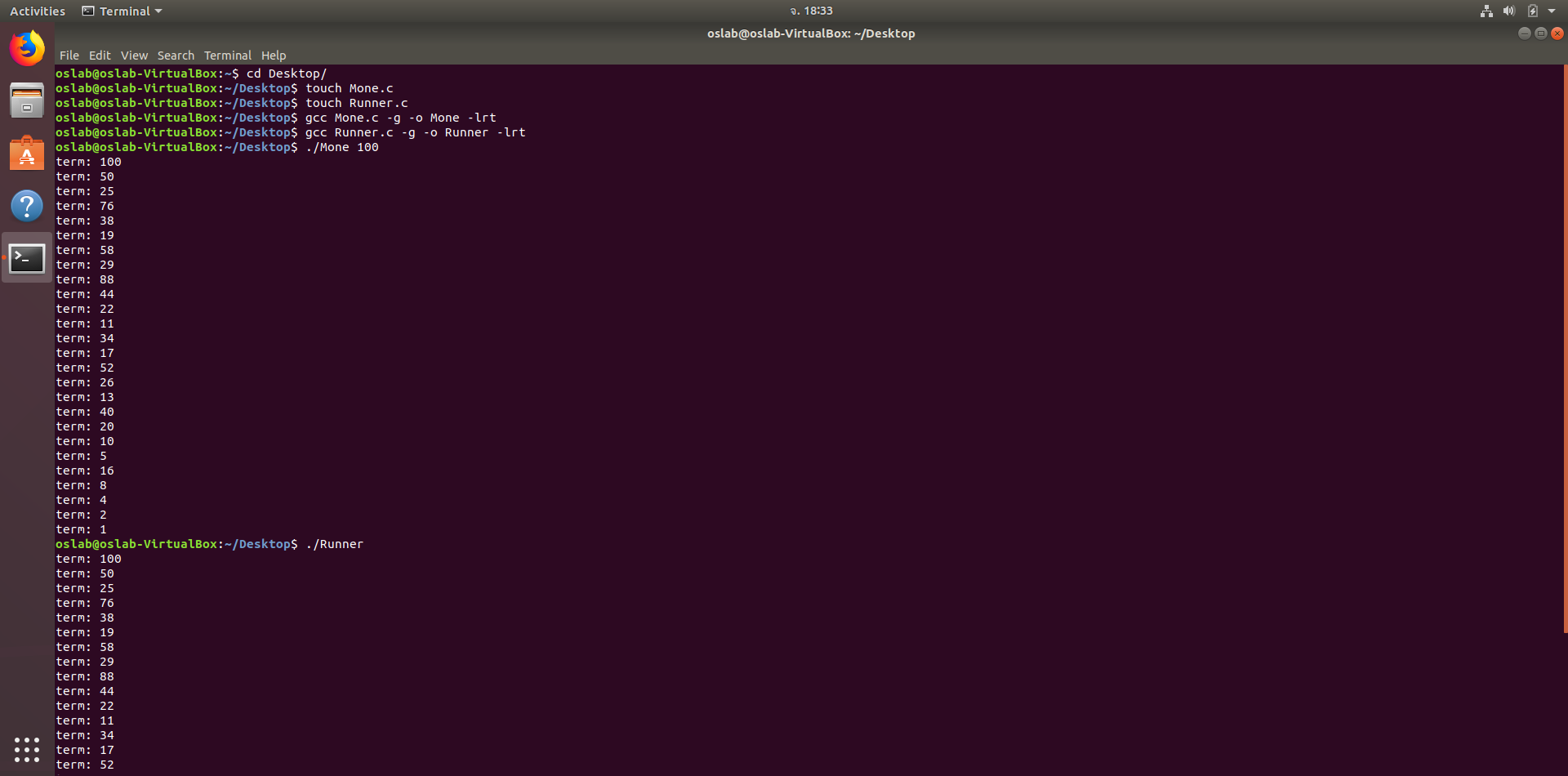


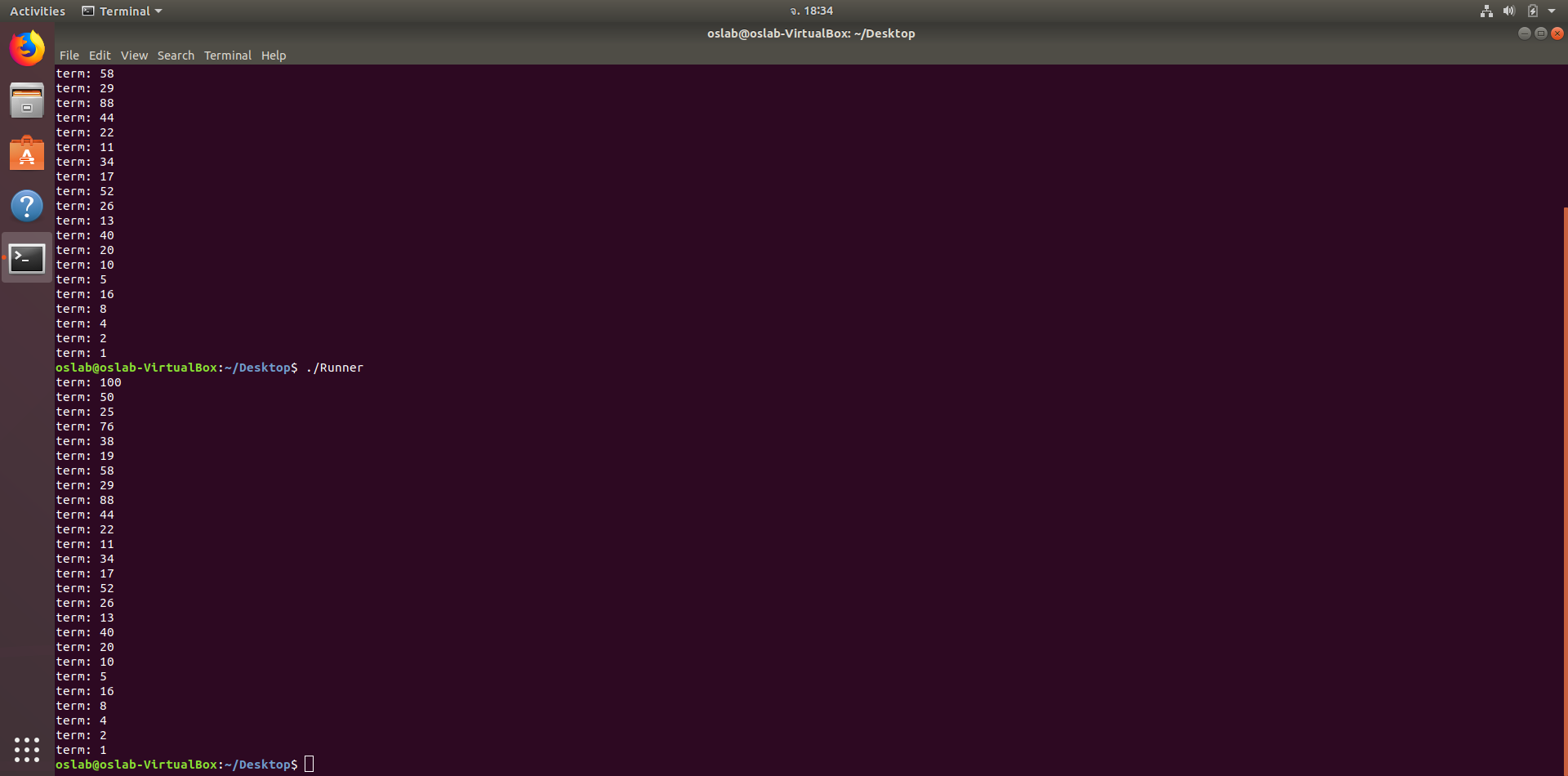
ตอบ

( Sample code write to memory )



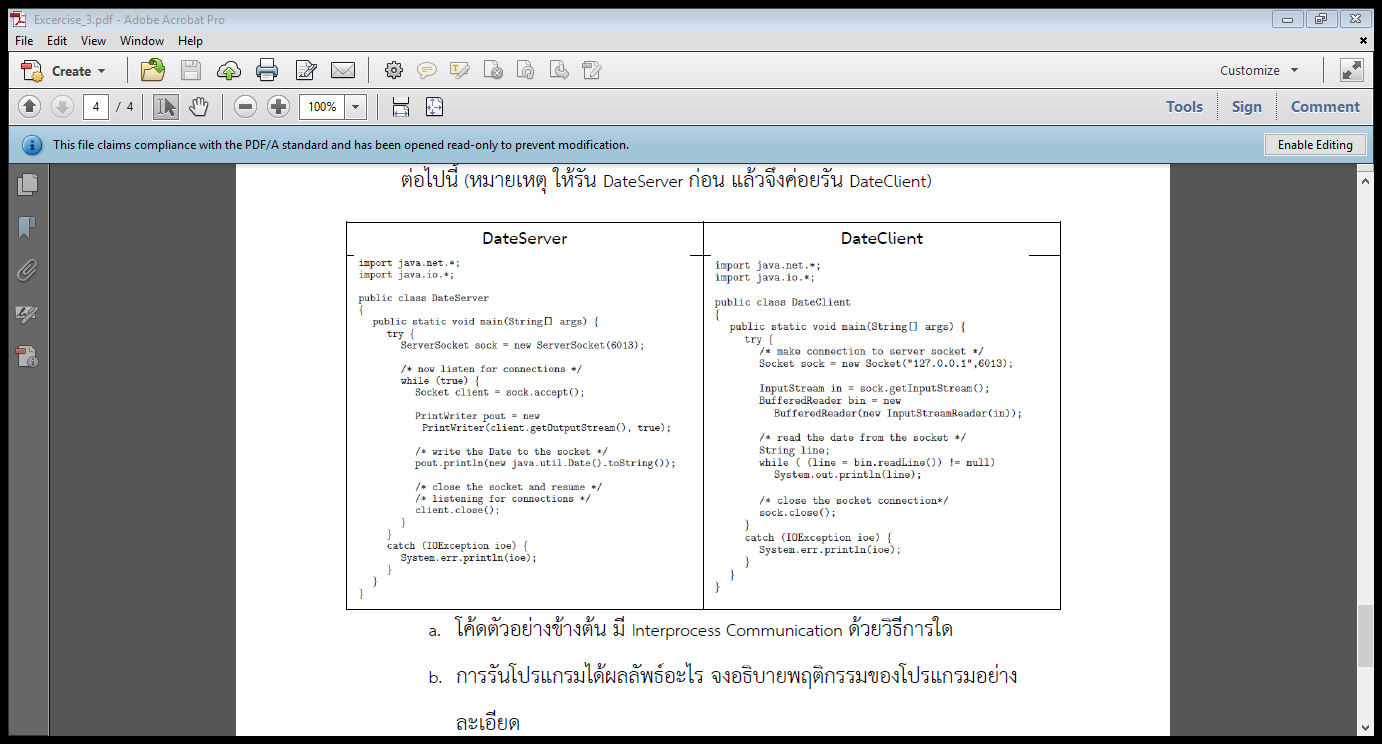
( Sample code read from memory )





( ภาพประกอบจากผลการทดลอง )

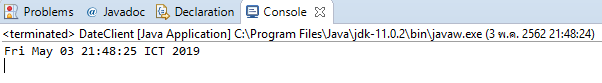
1. จากเรื่อง Interprocess Communication จงทดลองโปรแกรมภาษาจาวาต่อไปนี้ ( หมายเหตุ ให้รัน DateSever ก่อน แล้วจึงค่อยรัน DateClient )



1. โค้ดตัวอย่างข้างต้นมี Interprocess Communication ด้วยวิธีการใด

ตอบ Code ข้างต้นมี Interprocess Communication ด้วยวิธีการแบบ Socket

1. การรันโปรแกรมได้ผลลัพธ์อะไร จงอธิบายพฤติกรรมของโปนแกรมอย่างละเอียด

ตอบ การ Run ได้ผลลัพธ์ดังภาพต่อไปนี้

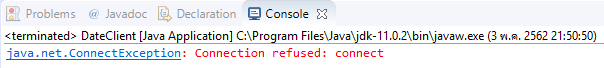
* พฤติกรรมของโปรแกรม DateServer จะทำการเปิด socket หรือ port 6013 ไว้เพื่อรอการเชื่อมจากภายนอกและรอที่จะส่งข้อมูลวันที่ในรูปแบบของ String ออกไปยัง Client ที่เชื่อมต่อ DateClient นั้น
* จำลองเป็นผู้เชื่อมต่อซึ่งมาเชื่อมต่อใน socket หรือ port เดียวที่ตรงกับของ DateServer จึงได้รับข้อมูลวันที่จาก DateServer มาและได้ทำการปริ้นแสดงผลออกมา
* ที่ class DateServer
* มีการสร้าง socket ขึ้นมาโดนระบุ port ที่ใช้คือ 6013
* มี Loop while ตรวจสอบว่ามีการเชื่อมต่อที่ socket นี้หรือไม่หากเป็นจริงก็จะทำงานต่อไป
* มีการสร้าง Object PrintWriter pout เพื่อทำการ get ค่า OutputStream จาก client ที่เชื่อมต่อ
* มีการใช้ Object pout เพื่อ เขียนข้อมูลวันที่ในรูปแบบของ String
* หลังจากนั้นทำการตัดการเชื่อมต่อกับ client ที่เข้ามาเชื่อต่อ port 6013 และทำการรอคอยการเชื่อมต่อใหม่อีกครั้ง
* มีการใช้ try catch เพื่อดักจับ ERROR
* ที่ class DateClient
* มีการสร้าง Socket ขึ้นมา IP 127.0.0.1 Port 6013
* มีการสร้าง Object InputStream in เป็น get ค่า InputStream
* มีการสร้าง Object BufferedReader มี parameter เป็น Object InputStream(in)
* มีการสร้างตัวแปร Line ชนิด String
* มี Loop เพื่อเช็คว่าถ้า มีการเขียนค่า String ที่มาจาก BufferedReader ลงในตัวแปร Line ให้ทำการปริ้นข้อความนั้นออกมาแสดงผล
* หลังจากนั้นทำการตัดการเชื่อมต่อกับ socket นั้น
* มีการใช้ try catch เพื่อดักจับ ERROR

1. จงอธิบายความหมายของ Socket

ตอบ Socket คือ กลุ่มของหมายเลข Port และ หมายเลข IP ซึ่งจะเป็นตัวบ่งชี้ที่เฉพาะเจาะจงสำหรับ Network process หนึ่งเดียวที่มีอยู่ในทั้งระบบ Internet คู่ของ Socket ที่ประกอบด้วย Socket หนึ่งตัว สำหรับต้นทาง และอีกตัว สำหรับปลายทาง สามารถใช้บรรยายถึงคุณลักษณะของ Connection oriented protocols เช่น

ถ้าผู้ใช้คนที่ 2 ต้องการใช้ Service Telnet จากเครื่องปลายทางเครื่องเดียวกัน ผู้ใช้นั้นก็จะได้รับการ assign หมายเลข Port ต้นทางที่แตกต่างกันออกไป โดยมีหมายเลข Port ปลายทางเหมือนกันกับผู้ใช้คนแรก จะเห็นได้ว่าการจับคู่ของหมายเลข Port และหมายเลข IP ทั้งต้นทางและปลายทางสามารถทำให้แยกความแตกต่างของ Internet connection ระหว่างเครื่องต้นทางและเครื่องปลายทางได้

1. หากรัน DateClient ก่อนการรัน DateServer ได้ผลลัพธ์อย่างไร

ตอบ หากรัน Client ก่อน จะทำให้เกิดผลลัพธ์ดังภาพต่อไปนี้

DateClient มีความพยายามที่จะเชื่อมต่อที่ port 6013 แต่ที่ port นั้นไม่ได้มีการถูกเปิดไว้ไม่มี process ที่ port 6013และหลังจากรัน DateServer ข้อความ ConectionException จะหายไปและถ้าหากทำการรันที่ DateClient อีกครั้งก็จะมีการ print ข้อความออกมา การทำงานสมบูรณ์เป็นปกติ DateServer จะต้อง listen connection ก่อนจะมีการ make connection จาก DateClient