แบบฝึกหัดที่ 4

1. เรื่อง User-Level Thread และ Kernel-Level Thread

• จงอธิบายความแตกต่างระหว่าง User-Level Thread และ Kernel-Level Thread

1.1

* **User-Level threads** จะเป็นแบบ **many-to-one** (เป็นรูปแบบที่ใช้เธรดสำหรับระบบปฏิบัติการ 1 หน่วย กับเธรดสำหรับผู้ใช้หลายหน่วย การจัดการเธรดจะอยู่ในพื้นที่ของผู้ใช้ซึ่งมีประสิทธิภาพ แต่ถ้าเธรดบล็อก System call โพรเซสทั้งหมดจะถูกบล็อกไปด้วย)
* **Kernel-level threads** จะเป็นแบบ **one-to-one** (เป็นรูปแบบที่แต่ละเธรดสำหรับผู้ใช้ จะจับคู่กับเธรดสำหรับระบบปฏิบัติการ ในลักษณะ 1 ต่อ 1 ทำให้สามารถทำงานพร้อมกันดีกว่าแบบ Many-to-One โดยยอมให้เธรดอื่นรันได้เมื่อเธรดบล็อก System call นอกจากนี้โมเดลนี้ยังยอมให้หลายๆเธรดทำงานแบบขนานกันได้ในระบบมัลติโพรเซสเซอร์ได้อีกด้วย)

1.2

- การใช้ User-Level Thread และ Kernel-Level Thread

เป็นรูปแบบของเธรดที่พยายามจะเชื่อมช่องว่างระหว่างการทำงานของเธรดในรูปแบบ Many-to-one กับ One-to-one เข้าด้วยกันโดยนำ User-level threads กับ Kernel-level threads มาพัฒนารวมกันเกิดเป็นการทำงานของเธรดที่เรียกว่า Many-to-many หลักการทำงานคือ จะ Mapping ตัว User-level threads หลายๆตัวเข้ากับชุดของ Kernel-level threads ตัวอย่างของรูปแบบนี้คือ ระบบปฏิบัติการ ยูนิกส์

1.3

- การใช้ Kernel-Level Thread

เหมาะสำหรับแอฟพิเคชั่นที่ต้องการความละเอียดสูง

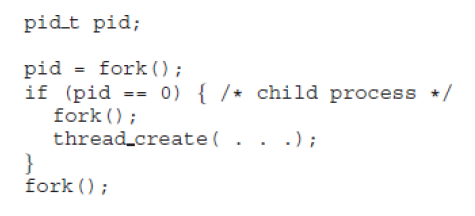
2. สถานการณ์ใดที่ Multithreaded Solution ที่ใช้ Multiple Kernel Threads

ทำให้ระบบมีประสิทธิภาพดีกว่าการใช้ Single-Threaded Solution บนระบบที่มีโพรเซสเซอร์เดียว จง

3. สิ่งใดต่อไปนี้ Register Values, Heap Memory, Global Variables, Stack Memory ที่ถูกแชร์ระหว่าง Multithreaded Process

เธรดที่ถูกแชร์คือ **Heap Memory** และ **Global Variables**

4. จากส่วนของโค้ดต่อไปนี้



• มีโปรเซสที่สร้างขึ้นกี่โปรเซส

มี 6 โปรเซส

• มี Thread ที่สร้างขึ้นกี่ Thread

มี 8 Thread

5. Thread Pool คืออะไร จงอธิบาย

Thread Pool นั้นเหมาะสำหรับที่จะใช้จัดการ Thread ที่ทำงานเล็กๆ ไม่ยุ่งยากซับซ้อนและใช้เวลาในการประมวลผลไม่นานนัก รวมทั้งไม่คำนึงถึงลำดับความสำคัญของ Thread อีกด้วยThread Pool เป็น Class ที่ .NET Framework เตรียมเอาไว้ให้สำหรับการทำงานแบบ Multi-thread ที่ง่ายๆ ไม่ซับซ้อน รวมถึงการจัดการกับการทำงานของ Method ที่ Run แบบ Asynchronous ด้วย Thread Pool นั้น Run แบบ Background Thread

6. จงหาตัวอย่างของโปรแกรมที่เขียนด้วย Java Thread และ OpenMP แสดงโค้ดพร้อมผลลัพธ์ของการรันที่ได้ และอธิบายพฤติกรรมของโปรแกรม

**public** **class** demo1 **extends** Thread{

**public** **static** **int** *count* = 0;

**public** **static** **synchronized** **void** count() {

*count*++;

}

**public** **static** **void** main(String[] args) {

Thread t1 = **new** Thread(**new** Runnable() {

**public** **void** run() {

**for**(**int** i=0; i<100; i++) {

*count*();

System.***out***.println(Thread.*currentThread*().getName());

}

}

});

Thread t2 = **new** Thread(**new** Runnable() {

**public** **void** run() {

**for**(**int** i=0; i<100; i++) {

*count*();

System.***out***.println(Thread.*currentThread*().getName());

}

}

});

t1.start();

t2.start();

**try** {

t1.join();

t2.join();

} **catch** (InterruptedException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

System.***out***.println("count= "+*count*);

}

}

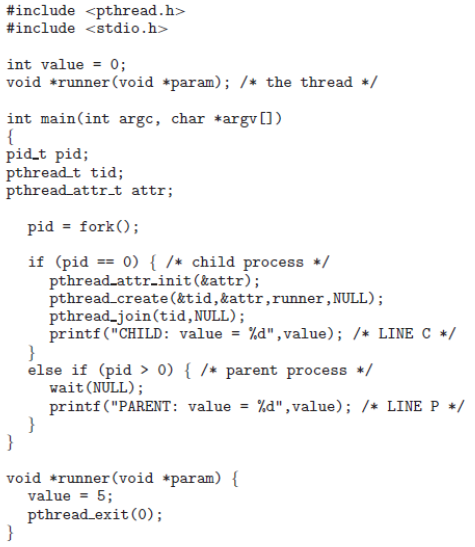
ผลที่ได้

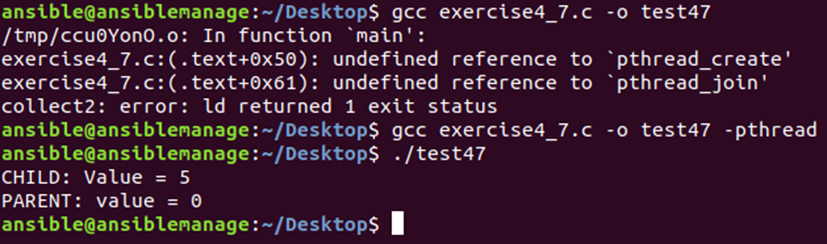


พฤติกรรมของโปรแกรม

โปรแกรมทำการสร้าง Thread ขึ้นมา 2 Thread มีการเรียกใช้ Thread ทั้ง 2 Thread โดยตรงจาก Class Reader1 และ 2 ตามลำดับ Thread ทั้ง 2 มีการทำงานเหมือนกันคืออ่านค่าจากไฟล์ text แต่เป็น text ที่ต่างกันทั้งคู่ทำงานไปในเวลาพร้อม ๆกันและทำให้ผลลัพธ์ที่จะแสดงผลค่าที่อ่านได้ออกมาเป็นบรรทัดจะเห็นได้ว่ามีการสลับไปสลับมาอยู่เกือบตลอดเนื่องจาก Thread

7. จงทดลองโค้ดภาษาซีต่อไปนี้ ผลลัพธ์ที่ได้จาก Line C และ Line P เป็นอย่างไร





Child process ถูก fork โดย parent process หลังจาก fork แล้ว parent process จะรอ child process รันจนเสร็จสมบูรณ์มี Thread ใหม่ที่ถูกสร้างขึ้นสำหรับ child process และเรียกใช้ฟังก์ชัน runner() ซึ่งตั้งค่าของ value เป็น5 ค่าของ value จะได้ค่าเป็น 0

8. จงเขียนโปรแกรมแบบ Multithread โดยมีการทำงานดังนี้

• Main Thread สร้างเลขจำนวนเต็มแบบสุ่ม 10 ค่า แล้วแสดงออกทางจอภาพ

• Main Thread เตรียมตัวแปร min, max, และ avg สำหรับเก็บค่าต่ำสุด, สูงสุด, และค่าเฉลี่ยตามลาดับ

• สร้าง Thread ย่อย 3 Thread มีหน้าที่ดังนี้ จากเลขจานวนเต็มที่สุ่มมาจาก Main Thread

i. Thread #1: ค้นหาค่าต่ำสุด เก็บลงใน min

ii. Thread #2: ค้นหาค่าสูงสุด เก็บลงใน max

iii. Thread #3: คำนวณค่าเฉลี่ย เก็บลงใน avg

• Main Thread แสดงผลลัพธ์ของ min, max, avg ออกทางจอภาพ

>>>>> ทำไม่ได้ครับ ผมไม่เข้าใจ <<<<<