ระบบปฏิบัติการ (Operating System)

- ความหมายของระบบปฏิบัติการ
- หน้าที่ของระบบปฏิบัติการ
- ส่วนประกอบของระบบปฏิบัติการ
- ประเภทของระบบปฏิบัติการ
- เริ่มต้นการทำงานของเครื่องคอมพิวเตอร์
- ส่วนประสานงานกับผู้ใช้
- การจัดการหน่วยความจำ
- การจัดการอุปกรณ์นำเข้าข้อมูลและอุปกรณ์แสดงผล
- การจัดการโปรเซสเซอร์

ผู้ช่วยศาสตราจารย์พัชรินทร์ บัวเย็น
(ประกอบตำราไมโครคอมพิวเตอร์และพีชคณิตบูลีน)

1

ความหมายและหน้าที่ของระบบปฏิบัติการ

ระบบปฏิบัติการ (OS : Operating System) คือกลุ่มของโปรแกรมที่ควบคุมการทำงานของอุปกรณ์คอมพิวเตอร์และช่วยจัดการให้การรันโปรแกรมเสร็จสิ้นสมบูรณ์ ซึ่งเป็นส่วนที่อำนวยความสะดวกแก่ผู้ใช้งานคอมพิวเตอร์ โดยที่ผู้ใช้ไม่จำเป็นต้องรู้กลไกการทำงานของระบบหรือฮาร์ดแวร์ของระบบ

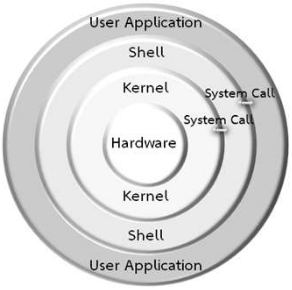
หน้าที่ของระบบปฏิบัติการ

- ทำหน้าที่ติดต่อประสานงานระหว่างผู้ใช้กับฮาร์ดแวร์ของคอมพิวเตอร์ (User Interface)
- ทำหน้าที่ควบคุมดูแลอุปกรณ์และการทำงานของเครื่องคอมพิวเตอร์ (Control Devices)
- ทำหน้าที่จัดสรรทรัพยากรต่างๆ ในระบบ (Resources Management)

2

ส่วนประกอบของระบบปฏิบัติการ

ระบบปฏิบัติการโดยทั่วไปจะประกอบด้วย 2 ส่วนประกอบหลักๆ คือ เคอร์เนล (Kernel) และเชลล์ (Shall) โดยมีคำสั่งเรียกระบบ หรือเรียกว่าซิสเต็มคอล (System Call) ในการติดต่อสื่อสารกัน



3

ส่วนประกอบของระบบปฏิบัติการ

- **เคอร์เนล (Kernel)**

  เคอร์เนลคือส่วนประกอบหลักหรือแกน (Core) ของระบบปฏิบัติการเป็นส่วนที่ทำหน้าที่สำคัญโดยรวมทั้งหมด เช่น การจัดลำดับงาน และการจัดสรรหน่วยความจำ เป็นต้น โดยเคอร์เนลจะเป็นส่วนที่ไม่อนุญาตให้ผู้ใช้เข้าถึงโดยตรง เป็นส่วนที่จะฝังตัวอยู่ในหน่วยความจำตลอดเวลา และเป็นส่วนที่ทำงานใกล้ชิดกับระบบฮาร์ดแวร์มากที่สุด

- **เชลล์ (Shall)**

  เชลล์ คือส่วนที่เป็นเสมือนตัวห่อหุ้มหรือเปลือกของเคอร์เนล ซึ่งเป็นส่วนที่อนุญาตให้ผู้ใช้ป้อนข้อมูลหรือส่งคำสั่งลงไปได้ โดยส่วนนี้จะจัดการงานบางส่วนเพื่ออำนวยความสะดวกแก่ผู้ใช้ เช่น ในระบบปฏิบัติการดอส จะมีโปรแกรม command.com เป็นเชลล์ที่คอยรับคำสั่งจากผู้ใช้เพื่อแปลความหมายของทำงานตามคำสั่งนั้น เช่น dir (แสดงไฟล์และไดเรกทอรี่ทั้งหมด) copy (คัดลอก) del (ลบ) และ ren (เปลี่ยนชื่อ) เป็นต้น โดยผ่านบรรทัดคำสั่ง (Command Line)

4

ส่วนประกอบของระบบปฏิบัติการ

- **คำสั่งเรียกระบบ (System Call)**

  คำสั่งเรียกระบบ เป็นส่วนหนึ่งของระบบปฏิบัติที่ทำหน้าที่จัดเตรียมการประสานงานระหว่างกระบวนการในระบบปฏิบัติการ โดยจะทำหน้าที่ติดต่อประสานงานระหว่างโปรแกรมของผู้ใช้กับเชลล์ และระหว่างเชลล์กับเคอร์เนล คำสั่งเรียกระบบมันจะเป็นคำสั่งภาษาแอสเซมบลี (Assembly) หรือในบางกรณีระบบอาจอนุญาตให้เรียกระบบได้โดยตรงจากคำสั่งโปรแกรมภาษาระดับสูงที่ผู้ใช้เขียนขึ้นได้

คำสั่งเรียกระบบสามารถแบ่งได้เป็น 5 กลุ่มใหญ่ๆ ได้แก่

- คำสั่งการควบคุมกระบวนการ (Process Control)
- คำสั่งการจัดการแฟ้ม (File Management)
- คำสั่งการจัดการอุปกรณ์ (Device Management)
- คำสั่งการบำรุงรักษาข้อมูลของระบบ (Information Maintenance)
- คำสั่งการติดต่อสื่อสาร (Communication)

5

ประเภทของระบบปฏิบัติการ

<u>การแบ่งประเภทตามระยะเวลาการตอบสนอง</u>

- ระบบปฏิบัติการแบบแบทช์ (Batch System) คือ ระบบปฏิบัติการที่จะไม่ตอบสนองต่อผู้ใช้ทันที แต่ผู้ใช้งานจะต้องส่งข้อมูลและคำสั่งที่จะประมวลผลเข้าสู่ระบบคอมพิวเตอร์ และรอเวลาจนกระทั่งระบบปฏิบัติการจัดสรรเวลาการทำงานให้ จึงจะเริ่มทำงานได้
- ระบบปฏิบัติการแบบตอบสนอง (Interactive System) หรือเรียกอีกชื่อว่าระบบไทม์แชร์ริ่ง (Time-Sharing System) คือระบบปฏิบัติการที่ยินยอมผู้ใช้งานสามารถติดต่อกับระบบปฏิบัติการได้โดยตรง โดยระบบปฏิบัติการจะตอบสนองต่อผู้ใช้งานในลักษณะโต้ตอบกันไปมา
- ระบบปฏิบัติการแบบเรียลไทม์ (Real-Time System) คือระบบปฏิบัติที่สามารถตอบสนองต่อผู้ใช้งานแบบทันทีทันใดได้ และทำงานได้ทันภายในเวลาที่กำหนด ซึ่งมักจะพบในระบบคอมพิวเตอร์ที่ใช้ในการควบคุมเครื่องจักรต่างๆ
- ระบบปฏิบัติการแบบไฮบริด (Hybrid System) คือระบบปฏิบัติการที่รวมเอาคุณสมบัติแบบแบทช์กับแบบตอบสนองเข้าไว้ด้วยกัน ซึ่งจะพบในระบบคอมพิวเตอร์ขนาดใหญ่เป็นส่วนมาก

6

➢ ประเภทของระบบปฏิบัติการ

การแบ่งประเภทตามจำนวนผู้ใช้งาน

- ระบบปฏิบัติการแบบผู้ใช้คนเดียว (Single User) คือ ระบบปฏิบัติการที่ ณ เวลาหนึ่งๆ จะมีผู้ใช้งานระบบปฏิบัติการได้เพียงคนเดียวเท่านั้น (แต่อาจใช้งานหลายโปรแกรมพร้อมกันได้) ได้แก่ ระบบปฏิบัติการวินโดวส์ (Windows OS) เช่น Windows 7, Windows 8 และ Windows 10 เป็นต้น และ MAC OS เป็นต้น

- ระบบปฏิบัติการแบบผู้ใช้หลายคน (Multi User) คือระบบปฏิบัติการที่ใช้ผู้ใช้งานหลายๆ คนทำงานพร้อมกันได้ในเวลาเดียวกัน อาทิ ระบบปฏิบัติการวินโดวส์เซอร์ฟเวอร์ (Windows Server) เช่น Windows Server 2012 และ OS X-Server สำหรับเครื่องตระกูล Apple และระบบปฏิบัติการ Linux เป็นต้น

7

➢ ประเภทของระบบปฏิบัติการ

การแบ่งประเภทตามจำนวนงานที่ประมวลผล

- ระบบปฏิบัติการประเภทประมวลผลทีละงาน (Single-Tasking) เป็นระบบปฏิบัติการที่สามารถประมวลผลโปรแกรมได้เพียง 1 โปรแกรมในเวลาหนึ่งๆ เท่านั้น ซึ่งในปัจจุบันไม่พบเห็นการใช้ระบบปฏิบัติการประเภทนี้แล้ว เนื่องจากไม่สามารถตอบสนองความต้องการในการใช้งานของผู้ใช้ได้

- ระบบปฏิบัติการประเภทประมวลผลหลายงาน (Multi-Tasking) เป็นระบบปฏิบัติการที่สามารถประมวลผลโปรแกรมหลายๆ โปรแกรมพร้อมกันได้ เช่น เมื่อผู้ใช้เปิดโปรแกรม Microsoft Word เพื่อพิมพ์งาน ผู้ใช้ยังสามารถเปิดโปรแกรม Windows Media Player เพื่อฟังเพลงไปพร้อมกันได้ ซึ่งระบบปฏิบัติการประเภทนี้จะพบเห็นโดยทั่วไปในปัจจุบัน

8

➢ เริ่มต้นการทำงานของเครื่องคอมพิวเตอร์

กระบวนการบู๊ตเครื่อง หรือเรียกว่าการบู๊ตอัพ (Boot Up) มีทั้งหมด 7 ขั้นตอน ดังนี้

- <u>ขั้นตอนที่ 1</u> การจ่ายไฟฟ้าเข้าสู่ระบบคอมพิวเตอร์ เป็นกระบวนการที่เริ่มตั้งแต่ผู้ใช้กดปุ่มสวิตช์เปิดเครื่อง (Power On) จากนั้นเพาเวอร์ซัพพลาย (Power Supply) จะแปลงสัญญาณไฟฟ้าทั่วไปให้เป็นสัญญาณสำหรับระบบคอมพิวเตอร์เพื่อจ่ายไปยังอุปกรณ์และชิ้นส่วนต่างๆ ภายในเครื่องคอมพิวเตอร์ ซึ่งเมื่ออุปกรณ์ต่างๆ เหล่านั้นได้รับกระแสไฟฟ้าหล่อเลี้ยงแล้ว ระบบจะส่งสัญญาณไปยังซีพียูเพื่อเริ่มการทำงานของเครื่องคอมพิวเตอร์ เรียกว่าสัญญาณเพาเวอร์กู๊ด (Power Good)
- <u>ขั้นตอนที่ 2</u> เริ่มการทำงานของไบออส (BIOS) โดยซีพียูจะเข้าถึงข้อมูลที่อยู่ในไบออสเพื่อประมวลผลตามชุดคำสั่งที่เก็บไว้
- <u>ขั้นตอนที่ 3</u> เริ่มกระบวนการโพสต์ (POST: Power-On Self-Test) ซึ่งการโพสต์ เป็นโปรแกรมส่วนหนึ่งในไบออสที่ทำหน้าที่ตรวจสอบความพร้อมของอุปกรณ์ที่ถูกติดตั้งไว้ในเครื่องคอมพิวเตอร์ทั้งหมด อาทิ ซีพียู เมนบอร์ด แรม ฮาร์ดดิสก์ รวมถึงอุปกรณ์ต่อพ่วงต่างๆ อาทิ คีย์บอร์ด และเม้าส์ เป็นต้น โดยผลการตรวจสอบที่ได้จะแสดงเป็นข้อความบนจอภาพในระหว่างการบู๊ต พร้อมกับมีสัญญาณเสียงปิ๊บ 1 ครั้ง เพื่อแสดงว่าอุปกรณ์หลักในการทำงานทั้งหมดครบและพร้อมใช้งาน

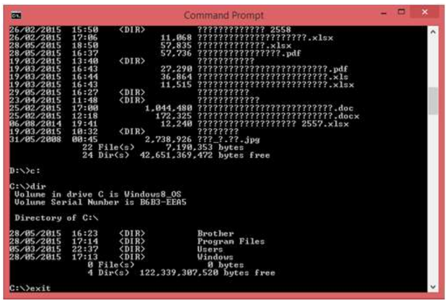
9

➢ เริ่มต้นการทำงานของเครื่องคอมพิวเตอร์

- <u>ขั้นตอนที่ 4</u> เปรียบเทียบรายการอุปกรณ์ที่ค้นพบจากกระบวนการโพสต์กับรายการในไบออส ซึ่งหากผลการเปรียบเทียบรายการอุปกรณ์ตรงกัน เครื่องคอมพิวเตอร์จะทำงานต่อไปตามปกติ แต่หากรายการของอุปกรณ์ที่ถูกติดตั้งในเครื่องคอมพิวเตอร์ไม่ตรงกับรายการในซีมอส ระบบจะแจ้งให้ผู้ใช้ทราบทันที
- <u>ขั้นตอนที่ 5</u> ไบออสจะค้นหาและอ่านค่าของระบบปฏิบัติการ ซึ่งระบบปฏิบัติการนี้ อาจจะถูกบรรจุอยู่ในฮาร์ดดิสก์ หรือซีดีรอม (CD-ROM) ก็ได้ (ลำดับการค้นหาอุปกรณ์ที่บรรจุระบบปฏิบัติการสามารถถูกกำหนดโดยผู้ใช้ได้)
- <u>ขั้นตอนที่ 6</u> โหลดเคอร์เนล (Kernel) ซึ่งเป็นส่วนสำคัญหลักของระบบปฏิบัติการ ไปไว้ในหน่วยความจำหลัก นั่นคือแรม
- <u>ขั้นตอนที่ 7</u> ระบบปฏิบัติการจะถูกประมวลผลภายใต้การควบคุมของเคอร์เนล โดยระบบปฏิบัติการจะเข้าควบคุมการทำงานของเครื่องคอมพิวเตอร์ทั้งหมด พร้อมทั้งแสดงสถานะและความพร้อมแก่ผู้ใช้ออกทางจอภาพ และเมื่อระบบปฏิบัติการได้ทำงานเสร็จสิ้นแล้ว จอภาพจะมีลักษณะที่พร้อมให้ผู้ใช้ใช้งาน ซึ่งเรียกส่วนของจอภาพที่พร้อมใช้งานนี้ว่า ส่วนประสานงานกับผู้ใช้ (User Interface) ซึ่งในปัจจุบันจะมีรูปแบบเป็นภาพกราฟิกร่วมกับข้อความเพื่อให้ผู้ใช้ใช้งานได้อย่างสะดวก (อ่านเพิ่มเติมในหัวข้อ ส่วนประสานงานกับผู้ใช้)

➢ ส่วนประสานงานกับผู้ใช้ (User Interface)

ส่วนประสานงานกับผู้ใช้ประเภทคอมมานด์ไลน์ (Command Line)



11

➢ ส่วนประสานงานกับผู้ใช้ (User Interface)

ส่วนประสานงานกับผู้ใช้ประเภทกราฟิก (GUI: Graphic User Interface)

12

การจัดการหน่วยความจำ (Memory Management)

เนื่องจากในความเป็นจริงของการประมวลผลจำเป็นจะต้องใช้ข้อมูลและคำสั่งในการประมวลผลเป็นจำนวนมากหรือในบางครั้งอาจจะต้องมีการทำงานของโปรแกรมหลายๆ โปรแกรมพร้อมกันซึ่งทำให้ต้องมีคำสั่งที่ใช้ในการประมวลผลเพิ่มมากขึ้นตามไปด้วย

ทั้งนี้ในการประมวลผลของระบบคอมพิวเตอร์ ระบบจำเป็นจะต้องเก็บคำสั่งหรือข้อมูลที่จะประมวลผลทั้งหมดไว้ในแรม ซึ่งแรมอาจจะมีเนื้อที่ไม่เพียงพอต่อการเก็บข้อมูลในขณะประมวลผลเหล่านั้นไว้ได้

จากปัญหาข้างต้น ระบบปฏิบัติการจะแก้ปัญหาโดยการสร้างหน่วยความจำเสมือน ซึ่งเป็นการจำลองเนื้อที่ในหน่วยความจำสำรอง นั่นคือฮาร์ดดิสก์ ให้ทำหน้าที่เสมือนแรม เพื่อใช้ในการเก็บข้อมูลทั้งหมดของโปรแกรมที่จะประมวลผลเอาไว้ โดยโปรแกรมที่ถูกจัดเก็บในหน่วยความจำเสมือนจะถูกเก็บในรูปของไฟล์ เรียกว่าสว๊อปไฟล์ (Swap File)

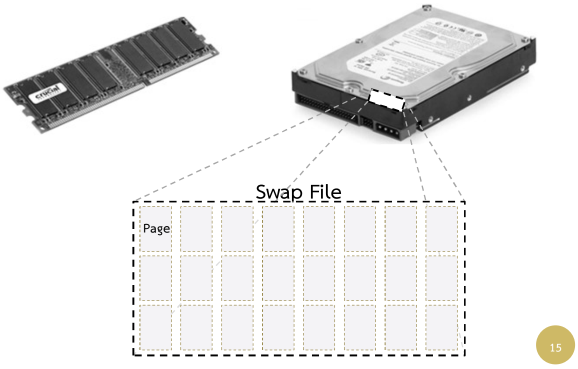
13

การจัดการหน่วยความจำ (Memory Management)

ระบบปฏิบัติการจะจัดการกับสว๊อปไฟล์โดยการแบ่งสว๊อปไฟล์นี้ออกเป็นส่วนๆ เรียกว่า เพจ (Page) โดยแต่ละเพจจะถูกกำหนดขนาดที่แน่นอนเอาไว้ และเมื่อระบบปฏิบัติการจะใช้งานโปรแกรมใดๆ ในเพจ ระบบปฏิบัติการจะเลือกเฉพาะเพจที่มีข้อมูลที่กำลังจะใช้ประมวลผลเท่านั้นเข้าสู่แรมจนกว่าจะเต็มเนื้อที่ของแรม แต่หากแรมถูกบรรจุจนเต็มแล้วและระบบยังคงต้องการใช้เนื้อที่ของแรมเพิ่ม ระบบปฏิบัติการจะถ่ายเทข้อมูลบางเพจในแรมที่ยังไม่ใช้ในขณะนั้นกลับไปไว้ในฮาร์ดดิสก์เหมือนเดิม เพื่อให้แรมมีเนื้อที่ว่างพอสำหรับนำเพจใหม่ที่มีข้อมูลที่จำเป็นต้องใช้ประมวลผลมาเก็บแทน ซึ่งจะทำให้ระบบสามารถประมวลผลโปรแกรมหรือข้อมูลเหล่านั้นได้ต่อไป

14

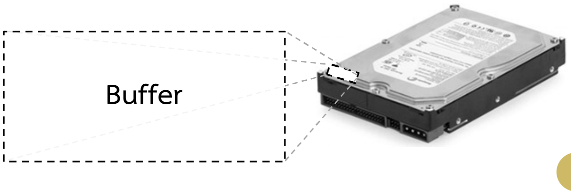
การจัดการหน่วยความจำ (Memory Management)



15

การจัดการอุปกรณ์นำเข้าและแสดงผล (I/O Device Management)

ในการทำงานของระบบคอมพิวเตอร์ อุปกรณ์นำเข้าข้อมูลมากกว่าหนึ่งชิ้นจะส่งข้อมูลเข้าสู่ระบบปฏิบัติการพร้อมๆ กัน โดยขณะเดียวกันนั้นระบบปฏิบัติการก็จะต้องส่งผลการทำงานของหลายโปรแกรมออกไปสู่อุปกรณ์แสดงผลหลายชิ้นด้วยเช่นกัน และเนื่องจากอุปกรณ์รับและแสดงผลเหล่านี้มีความเร็วน้อยกว่าซีพียูมาก ดังนั้นระบบปฏิบัติการจึงต้องจัดเตรียมพื้นที่ส่วนหนึ่งของหน่วยความจำแรมหรือฮาร์ดดิสก์ ซึ่งเรียกว่า บัฟเฟอร์ (Buffer) เพื่อเป็นที่พักรอของข้อมูลที่จะรับเข้ามาหรือเตรียมส่งออกไปยังอุปกรณ์แสดงผลต่อไป



16

การจัดการอุปกรณ์นำเข้าและแสดงผล (I/O Device Management)

เช่นในกรณีการสั่งพิมพ์งานออกทางเครื่องพิมพ์ ข้อมูลที่ถูกส่งไปยังเครื่องพิมพ์อาจมีขนาดใหญ่ หรือในกรณีสั่งพิมพ์หลายงานพร้อมกันจะทำให้มีปริมาณงานที่จะพิมพ์เพิ่มมากขึ้นซึ่งจะทำให้หน่วยความจำในเครื่องพิมพ์ไม่เพียงพอต่อการจัดเก็บไฟล์งาน อีกทั้งในการพิมพ์งานของเครื่องพิมพ์เครื่องเดียวจะต้องพิมพ์งานตามลำดับทีละงาน ไม่สามารถสลับงานได้ ระบบปฏิบัติการจะจัดการโดยการแบ่งเนื้อที่ส่วนหนึ่งในฮาร์ดดิสก์เพื่อสำรองข้อมูลที่จะพิมพ์เหล่านั้นเอาไว้

วิธีการสำรองข้อมูลที่จะสั่งออกทางเครื่องพิมพ์ไว้ในหน่วยความจำของเครื่องเรียกว่า สปูลลิ่ง (Spooling) โดยฮาร์ดดิสก์จะเก็บงานที่พิมพ์เหล่านั้นไว้และระบบปฏิบัติการจะทำหน้าที่จัดลำดับงานที่จะส่งไปยังเครื่องพิมพ์ตามลำดับการสั่งพิมพ์จากผู้ใช้เพื่อส่งไปยังเครื่องพิมพ์อีกครั้งตามลำดับ โดยในการสปูลลิ่งนี้ ผู้ใช้สามารถจัดการรายการพิมพ์ได้ อาทิ การยกเลิกการพิมพ์ได้ทั้งในสถานะงานนั้นยังคงรอพิมพ์อยู่ในคิว หรืองานนั้นกำลังถูกพิมพ์อยู่ก็ได้ หรือการหยุดการพิมพ์ชั่วคราว เป็นต้น

17

การจัดการอุปกรณ์นำเข้าและแสดงผล (I/O Device Management)

18

➢ การจัดการโปรเซสเซอร์ (Processor Management)

<u>โปรเซส (Process)</u> คือ โปรแกรมที่กำลังถูก Execute ซึ่งเป็นสิ่งที่หนึ่ง OS จะต้องทำการจัดการเพื่อให้การประมวลผลโปรเซสแต่ละโปรเซสเป็นไปได้อย่างมีประสิทธิภาพมากที่สุด

<u>ส่วนประกอบของโปรเซส</u> มีดังนี้

1. ชื่อและหมายเลขประจำตัวของโปรเซส (Process ID)
2. โค้ดของโปรแกรม (Program Code) เป็นคำสั่งที่สามารถ Execute ได้ทันที
3. ข้อมูล (Data)
4. บล็อกควบคุมโปรเซส (Process Control Block = PCB)
5. Program Status Words = PSW เป็นตัวเก็บสถานะของโปรเซสและเก็บแอดเดรสถัดไปที่จะต้องถูก Execute ซึ่งทำหน้าที่คล้าย PC (Program Counter) ในไมโครโปรเซสเซอร์

➢ การจัดการโปรเซสเซอร์ (Processor Management)

<u>PCB</u> คือ ส่วนหนึ่งของหน่วยความจำที่ถูกแบ่งออกมาเพื่อเก็บข้อมูลสำคัญของโปรเซส ได้แก่

- Process ID
- Process Status
- ลำดับความสำคัญของโปรเซส
- Pointer ที่ชี้ไปยังตำแหน่งที่อยู่ของโปรเซสในหน่วยความจำ
- Pointer ที่ชี้ไปยังทรัพยากรต่างๆ ที่โปรเซสนั้นครอบครอง
- พื้นที่ที่เก็บค่าของ Register

Identifier
State
Priority
Program Counter
Memory Pointers
Context Data
I/O Status Information
Accounting Information
⋮

➢ การจัดการโปรเซสเซอร์ (Processor Management)

<u>คุณสมบัติของโปรเซส</u>

- คุณสมบัติด้านการจัดลำดับค่าความสำคัญ (Priority) ซึ่งโปรเซสแต่ละตัวจะถูกกำหนดค่าความสำคัญขณะที่โปรเซสนั้นถูกสร้างขึ้น ซึ่งค่าความสำคัญนี้อาจถูกเปลี่ยนแปลงระหว่างการประมวลผลหรือไม่ถูกเปลี่ยนแปลงขึ้นอยู่กับระบบปฏิบัติการ ซึ่งระบบปฏิบัติการจะให้สิทธิพิเศษแก่โปรเซสที่มีค่าความสำคัญสูงมากกว่าโปรเซสที่มีความสำคัญต่ำกว่า
- คุณสมบัติด้านอำนาจหน้าที่ (Authority) เป็นสิ่งที่บ่งบอกว่าโปรเซสนั้นสามารถทำอะไรได้บ้าง และสามารถใช้อุปกรณ์ชิ้นไหนได้บ้าง เป็นต้น

➢ การจัดการโปรเซสเซอร์ (Processor Management)

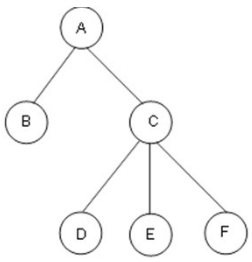
<u>การสร้างโปรเซส</u>

การสร้างโปรเซสเกิดจากการแตกตัวจากโปรเซสเริ่มต้น (Initial Process) และเมื่อได้โปรเซสสำหรับแต่ละงานแล้ว โปรเซสเหล่านั้นจะสร้างโปรเซสเพื่อประมวลผล ซึ่งโปรเซสนี้จะเรียกว่าโปรเซสพ่อแม่ (Parent Process) และโปรเซสที่ถูกสร้างขึ้นใหม่นั้นจะเรียกว่า โปรเซสลูก (Child Process)

การให้กำเนิดโปรเซสลูกต่อไปได้เรื่อยๆ จะทำให้เกิดเป็น “ลำดับชั้นของโปรเซส (Process Hierarchy)” ขึ้น ซึ่งลำดับชั้นของโปรเซสนี้มีลักษณะเป็น “โครงสร้างแบบต้นไม้ (Tree Structure)” โดยโปรเซสหนึ่งๆ จะมีโปรเซสพ่อแม่ได้เพียงโปรเซสเดียว แต่สามารถให้กำเนิดโปรเซสลูกได้หลายโปรเซส

➢ การจัดการโปรเซสเซอร์ (Processor Management)

จากรูปตัวอย่างลำดับชั้นของโปรเซส โปรเซส A มีโปรเซสลูก 2 โปรเซสคือ โปรเซส B และ C โปรเซส C ให้กำเนิดโปรเซสลูกออกมา 3 โปรเซสคือ D E และ F โดยโปรเซสต่างๆ ของผู้ใช้จะเป็นโปรเซสลูกของระบบปฏิบัติการเพราะถูกสร้างขึ้นโดยระบบปฏิบัติการ ข้อสำคัญประการหนึ่งเกี่ยวกับการให้กำเนิดโปรเซส คือ โปรเซสพ่อแม่ไม่ได้เป็นผู้สร้างโปรเซสลูก แต่เป็นผู้ให้กำเนิด (แตกงานออกเป็นงานย่อย) แล้วระบบปฏิบัติการจึงจะสร้างโปรเซสเพื่อนำงานย่อยเหล่านั้นเข้ามาทำงานในระบบ



➢ การจัดการโปรเซสเซอร์ (Processor Management)

<u>การทำลายโปรเซส</u>

การทำลายโปรเซสจะเกิดขึ้นเมื่อโปรเซสนั้นๆ ได้ทำงานเสร็จสิ้นแล้ว โดยโปรเซสจะแจ้งให้เคอร์เนลและโปรเซสพ่อแม่รับทราบว่าจะขอคืนทรัพยากรและหน่วยความจำที่ใช้งานอยู่ให้กับระบบ แล้วจึงจะสามารถทำลายโปรเซสนั้นและลบบล็อกควบคุมโปรเซสของโปรเซสนั้นออกจากระบบได้

ในกรณีที่โปรเซสพ่อแม่ถูกทำลายก่อนที่โปรเซสลูกจะทำงานเสร็จ เคอร์เนลจะให้โปรเซสลูกนั้นไปอยู่ในความดูแลของโปรเซสตั้งต้นแทน เมื่อโปรเซสลูกทำงานเสร็จแล้วก็จะแจ้งให้เคอร์เนลและโปรเซสตั้งต้นทราบแล้วทำลายโปรเซสลูกนั้น แต่อาจมีบางกรณีที่โปรเซสตั้งต้นทำงานผิดพลาดไม่สามารถจัดการกับโปรเซสลูกได้ ซึ่งจะก่อให้เกิดซอมบี้โปรเซส (Zombie Process) ได้

(ซอมบี้โปรเซส เป็นโปรเซสที่คืนทรัพยากรให้กับระบบแล้ว แต่ไม่สามารถกำจัดโปรเซสนั้นออกจากระบบได้ เนื่องจากโปรเซสพ่อแม่ของโปรเซสนั้นได้ถูกทำลายไปแล้ว และโปรเซสเริ่มต้นไม่สามารถจัดการกับโปรเซสนี้ได้)

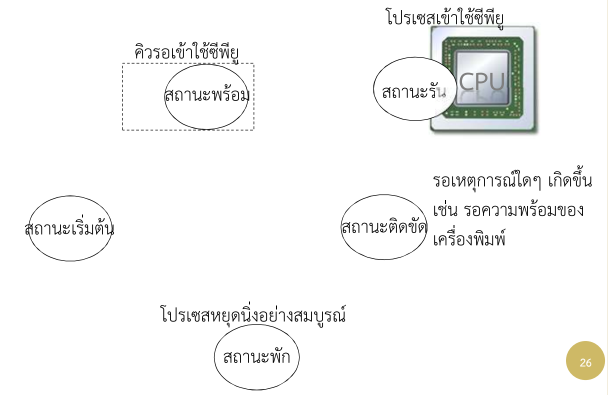
➢ การจัดการโปรเซสเซอร์ (Processor Management)

สถานะของโปรเซส

สถานะของโปรเซสมี 5 สถานะ ตั้งแต่เริ่มต้นจนสิ้นสุดการทำงาน ดังนี้

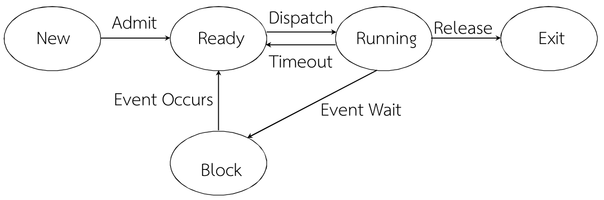
1. สถานะเริ่มต้น (Start State) คือ สถานะแรกเริ่มของโปรเซสที่ถูกสร้างขึ้นมาใหม่เพื่อนำเข้าสู่ขั้นตอนการประมวลผลต่อไป
2. สถานะพร้อม (Ready State) คือสถานะที่โปรเซสพร้อมที่จะใช้ซีพียูทันทีที่ระบบปฏิบัติการมอบหมายให้ ในสถานะนี้ไม่มีการรันของโปรเซส (โปรเซสหยุดนิ่ง)
3. สถานะรัน (Running State) คือสถานะที่โปรเซสกำลังครอบครองซีพียูอยู่ นั่นคือเป็นสถานะที่โปรเซสกำลังใช้ซีพียูเพื่อประมวลผลชุดคำสั่งหรือข้อมูลของโปรเซสนั้น
4. สถานะติดขัด (Blocked State) คือสถานะที่โปรเซสหยุดรอเหตุการณ์ใดเหตุการณ์หนึ่งให้เกิดขึ้น อาทิ รอเครื่องพิมพ์ให้พร้อมพิมพ์งาน โดยโปรเซสไม่จำเป็นต้องใช้ซีพียูและยังไม่พร้อมที่จะครอบครองทรัพยากรใดๆ ดังนั้นระบบปฏิบัติการจึงแยกสถานะนี้ออกมาต่างหาก
5. สถานะพัก (Suspend State) คือสถานะที่โปรเซสไม่มีการทำงานใดๆ นั่นคือเป็นสถานะที่โปรเซสหยุดนิ่งอย่างสมบูรณ์ ไม่มีการรอใช้ซีพียูหรือรอเหตุการณ์ใดๆ

➢ การจัดการโปรเซสเซอร์ (Processor Management)



➢ การจัดการโปรเซสเซอร์ (Processor Management)

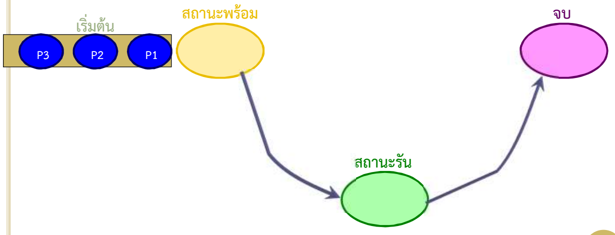
ความสัมพันธ์ของสถานะโปรเซส (Stalling, W., 2013, p. 300)



➢ การจัดการโปรเซสเซอร์ (Processor Management)

การเปลี่ยนสถานะของโปรเซส

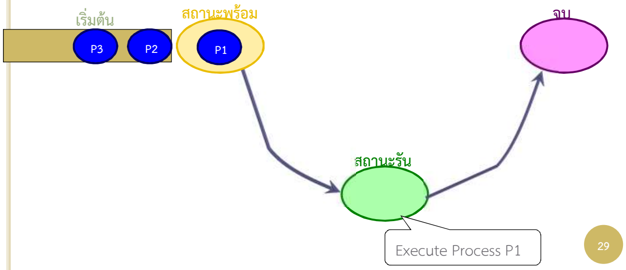
ไม่มีเวลาควอนตัม



➢ การจัดการโปรเซสเซอร์ (Processor Management)

การเปลี่ยนสถานะของโปรเซส (ต่อ)

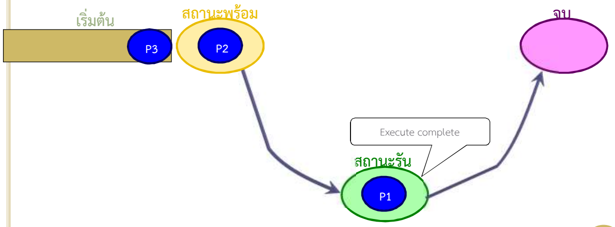
ไม่มีเวลาควอนตัม



➢ การจัดการโปรเซสเซอร์ (Processor Management)

การเปลี่ยนสถานะของโปรเซส (ต่อ)

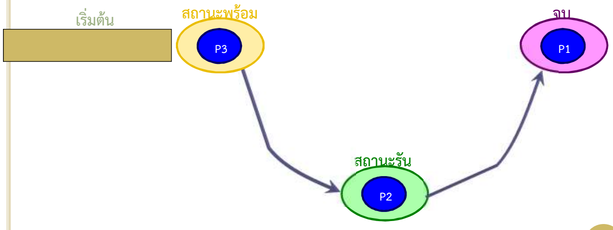
ไม่มีเวลาควอนตัม



➢ การจัดการโปรเซสเซอร์ (Processor Management)

<u>การเปลี่ยนสถานะของโปรเซส</u> (ต่อ)

ไม่มีเวลาควอนตัม

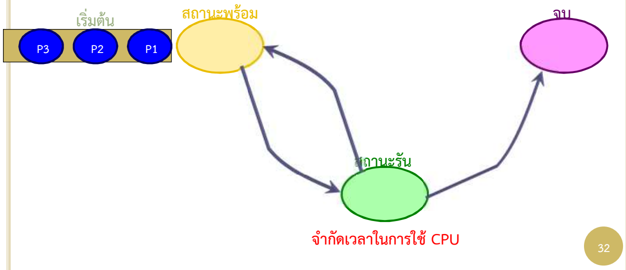


31

➢ การจัดการโปรเซสเซอร์ (Processor Management)

<u>การเปลี่ยนสถานะของโปรเซส</u> (ต่อ)

มีเวลาควอนตัม

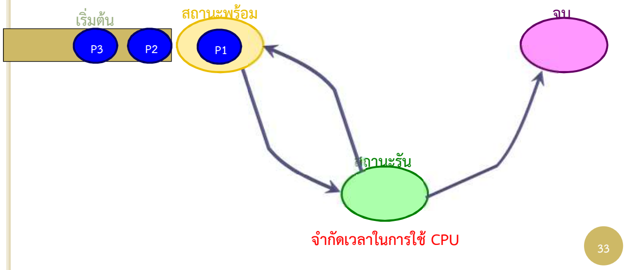


32

➢ การจัดการโปรเซสเซอร์ (Processor Management)

<u>การเปลี่ยนสถานะของโปรเซส</u> (ต่อ)

มีเวลาควอนตัม

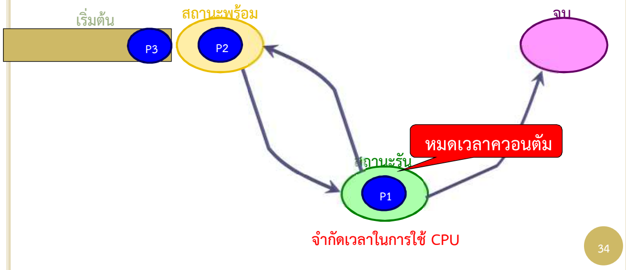


33

➢ การจัดการโปรเซสเซอร์ (Processor Management)

<u>การเปลี่ยนสถานะของโปรเซส</u> (ต่อ)

มีเวลาควอนตัม

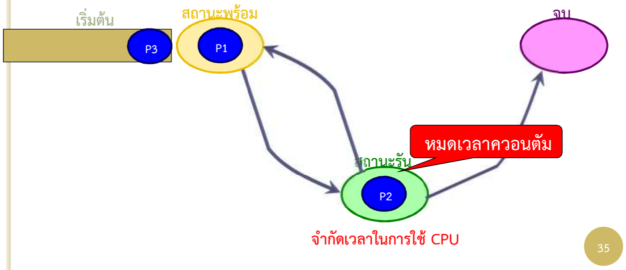


34

➢ การจัดการโปรเซสเซอร์ (Processor Management)

<u>การเปลี่ยนสถานะของโปรเซส</u> (ต่อ)

มีเวลาควอนตัม

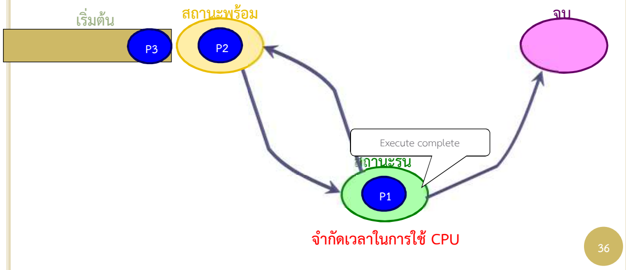


35

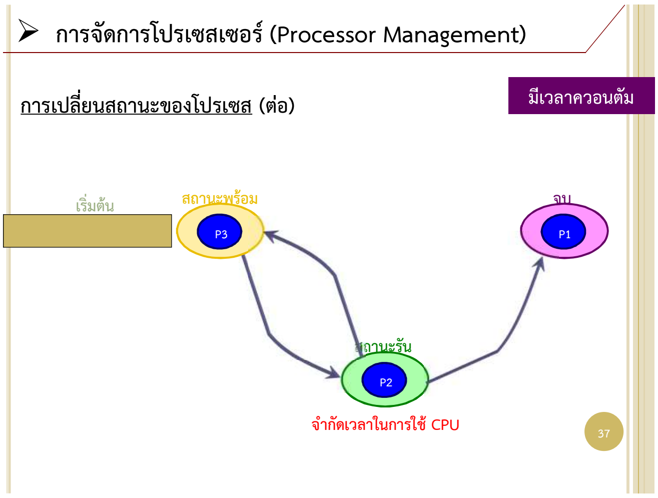
➢ การจัดการโปรเซสเซอร์ (Processor Management)

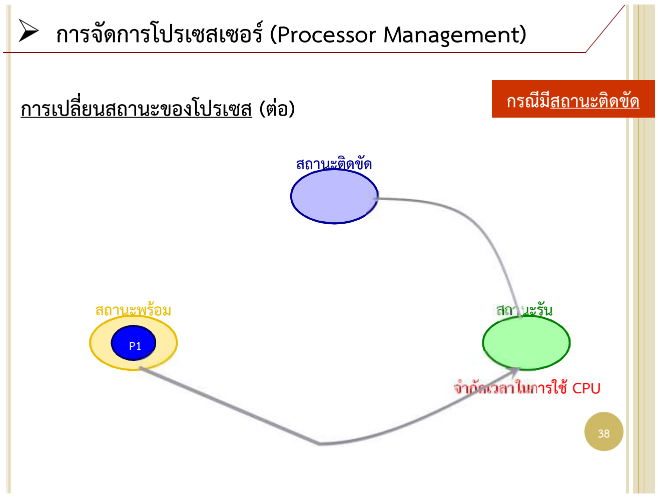
<u>การเปลี่ยนสถานะของโปรเซส</u> (ต่อ)

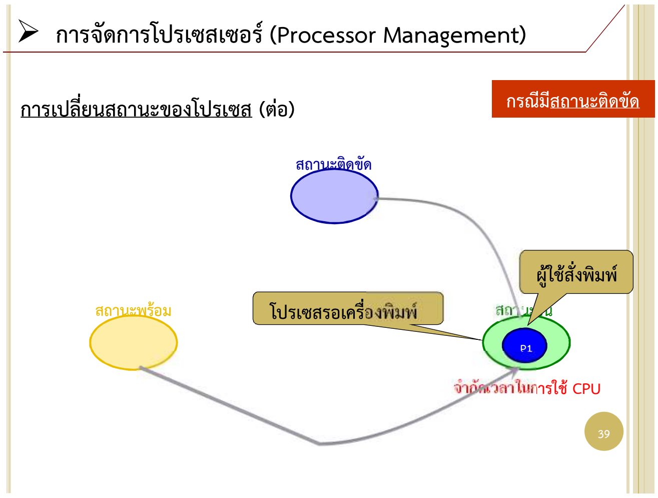
มีเวลาควอนตัม

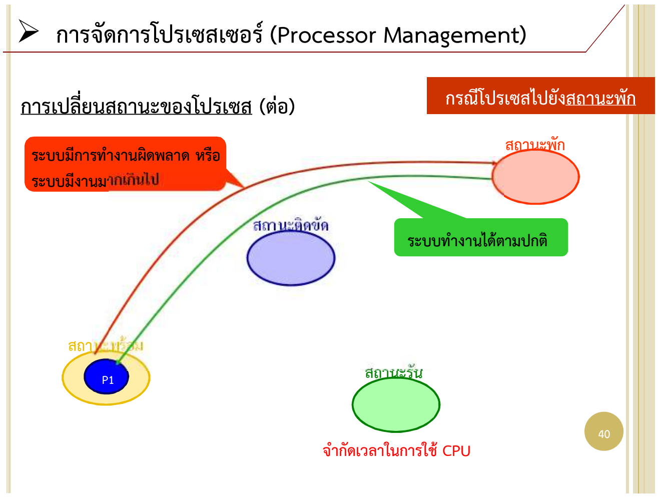


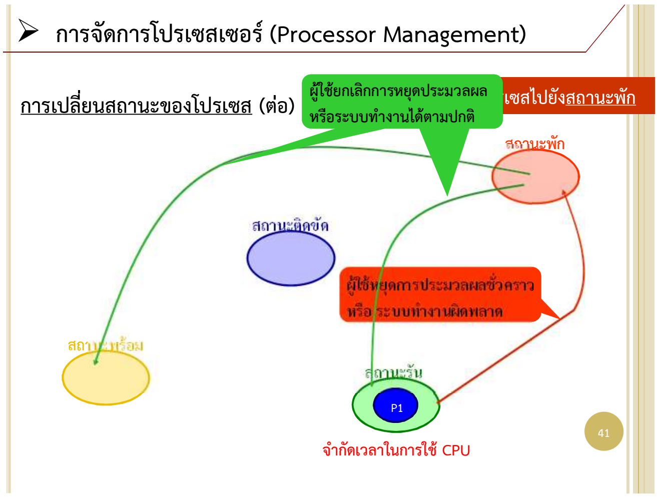
36

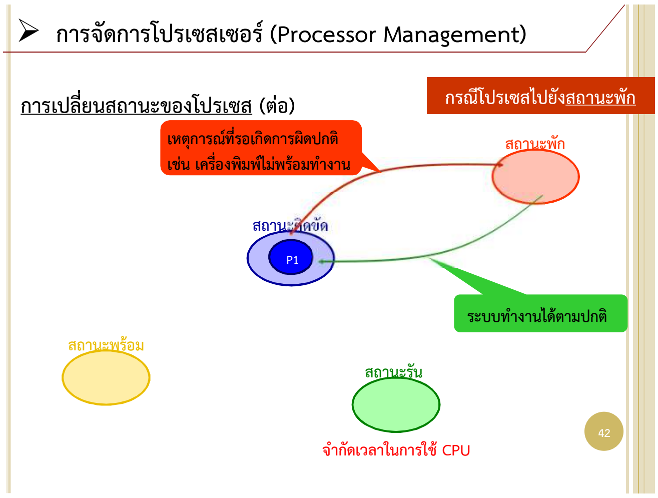

การจัดการโปรเซสเซอร์ (Processor Management)
การเปลี่ยนสถานะของโปรเซส (ต่อ)
มีเวลาควอนตัม
เริ่มต้น
สถานะพร้อม
P3
จบ
P1
สถานะรัน
P2
จำกัดเวลาในการใช้ CPU
37


การจัดการโปรเซสเซอร์ (Processor Management)
การเปลี่ยนสถานะของโปรเซส (ต่อ)
กรณีมีสถานะติดขัด
สถานะติดขัด
สถานะพร้อม
P1
สถานะรัน
จำกัดเวลาในการใช้ CPU
38


การจัดการโปรเซสเซอร์ (Processor Management)
การเปลี่ยนสถานะของโปรเซส (ต่อ)
กรณีมีสถานะติดขัด
สถานะติดขัด
ผู้ใช้สั่งพิมพ์
สถานะพร้อม
โปรเซสรอเครื่องพิมพ์
P1
จำกัดเวลาในการใช้ CPU
39


การจัดการโปรเซสเซอร์ (Processor Management)
การเปลี่ยนสถานะของโปรเซส (ต่อ)
กรณีโปรเซสไปยังสถานะพัก
ระบบมีการทำงานผิดพลาด หรือ
ระบบมีงานมากเกินไป
สถานะพัก
ระบบทำงานได้ตามปกติ
สถานะติดขัด
P1
สถานะรัน
จำกัดเวลาในการใช้ CPU
40

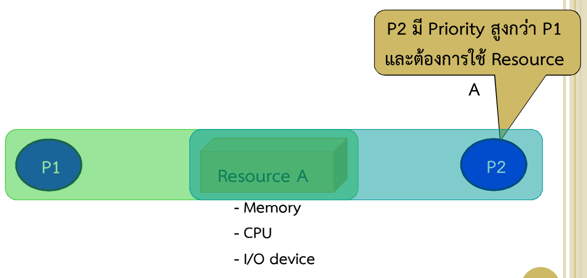

การจัดการโปรเซสเซอร์ (Processor Management)
การเปลี่ยนสถานะของโปรเซส (ต่อ)
ผู้ใช้ยกเลิกการหยุดประมวลผล
หรือระบบทำงานได้ตามปกติ
เซสไปยังสถานะพัก
สถานะพัก
สถานะติดขัด
ผู้ใช้หยุดการประมวลผลชั่วคราว
หรือ ระบบทำงานผิดพลาด
สถานะพร้อม
สถานะรัน
P1
จำกัดเวลาในการใช้ CPU
41


การจัดการโปรเซสเซอร์ (Processor Management)
การเปลี่ยนสถานะของโปรเซส (ต่อ)
กรณีโปรเซสไปยังสถานะพัก
เหตุการณ์ที่รอเกิดการผิดปกติ
เช่น เครื่องพิมพ์ไม่พร้อมทำงาน
สถานะพัก
สถานะติดขัด
P1
ระบบทำงานได้ตามปกติ
สถานะพร้อม
สถานะรัน
จำกัดเวลาในการใช้ CPU
42

➢ การจัดการโปรเซสเซอร์ (Processor Management)

ปัญหาการตัดตอน (Preemptive)

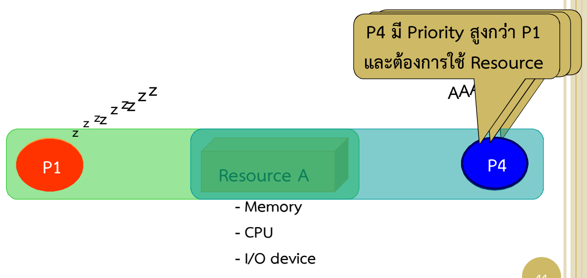
<u>ปัญหาการทำงานของโปรเซส</u>



➢ การจัดการโปรเซสเซอร์ (Processor Management)

ปัญหาการอดตาย (Starvation) หรือ Infinity Postponement

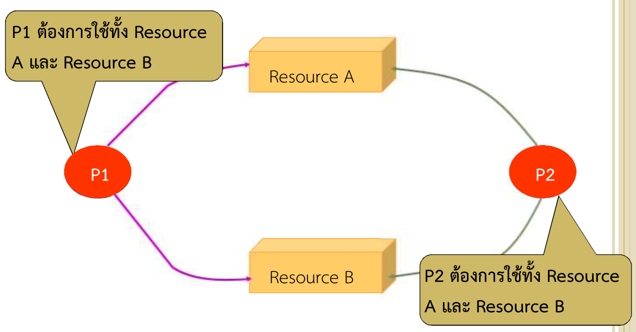
<u>ปัญหาการทำงานของโปรเซส</u>



➢ การจัดการโปรเซสเซอร์ (Processor Management)

ปัญหาการติดตาย (Deadlock)

<u>ปัญหาการทำงานของโปรเซส</u>



➢ การจัดการโปรเซสเซอร์ (Processor Management)

หน้าที่หนึ่งของ OS คือการจัดคิวและคัดสรรโปรเซสให้เข้าใช้โปรเซสเซอร์ ซึ่งเรียกว่า ตัวจัดคิวในระยะสั้น (Short-term Scheduler) โดยมี ตัวส่ง (Dispatcher) ทำหน้าที่คัดสรรและจัดลำดับโปรเซสในสถานะพร้อม เพื่อจัดส่งไปยัง สถานะรัน (เข้าใช้ซีพียู) ต่อไป

กลไกการคัดเลือกโปรเซสในคิวของสถานะพร้อม มีหลายวิธีได้แก่

1. FCFS (First-Come-First-Serve)
2. RR (Round-Robin)
3. Priority Qeueu
4. SJN (Shortest-Job-Next)
5. SRT (Shortest-Remaining-Time)

➢ การจัดการโปรเซสเซอร์ (Processor Management)

FCFS : <u>F</u>irst-<u>C</u>ome-<u>F</u>irst-<u>S</u>erve

การจัดคิวแบบ FCFS เป็นวิธีการจัดคิวที่ง่ายที่สุด คือ โปรเซสใดที่เข้ามาในคิวก่อนก็จะได้เข้าใช้ซีพียูก่อน นั่นคือ “มาก่อนได้ใช้ก่อน” และโปรเซสที่ครอบครองซีพียูจะใช้งานซีพียูจนกระทั่งงานเสร็จสิ้น ไม่มีการจำกัดเวลาควอนตัม

หากโปรเซสนั้นต้องการใช้ I/O Device โปรเซสก็จะปลดปล่อยซีพียูแล้วย้ายโปรเซสไปยังสถานะติดขัด และเมื่อโปรเซสนั่นใช้งาน I/O Device เสร็จก็จะวนกลับมาต่อคิวในสถานะพร้อม

➢ การจัดการโปรเซสเซอร์ (Processor Management)

FCFS : <u>F</u>irst-<u>C</u>ome-<u>F</u>irst-<u>S</u>erve

<u>การคำนวณเวลาและตารางเวลาของโปรเซส</u>

โปรเซส	ลำดับการเข้าคิว	เวลาที่ต้องใช้ซีพียู (วินาที)	เวลาที่ต้องรอในคิว (วินาที)	เวลาที่โปรเซสทำงานเสร็จสิ้น (วินาที)
P1	1	10	0	10
P2	2	4	10	14
P3	3	9	14	23

วินาทีที่ / โปรเซส	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
P1	R	R	R	R	R	R	R	R	R	R															
P2	o	o	o	o	o	o	o	o	o	o	R	R	R	R											
P3	o	o	o	o	o	o	o	o	o	o	o	o	o	o	R	R	R	R	R	R	R	R	R		

↑ P1 ↑ P2 ↑ P3

ให้ o แทนเวลาที่โปรเซสอยู่ในสถานะพร้อม

ให้ R แทนเวลาที่โปรเซสอยู่ในสถานะรัน

➢ การจัดการโปรเซสเซอร์ (Processor Management)

RR : Round-Robin

การจัดคิวแบบ FCFS อาจเรียกว่าเป็นการจัดคิวแบบวนรอบ โดยลักษณะการจัดจะคล้ายกับ FCFS คือโปรเซสใดมาก่อนได้เข้าใช้ซีพียูก่อน แต่ RR จะมีการกำหนดเวลาเข้าใช้ซีพียู หรือที่เรียกว่า เวลาควอนตัม (Quantum Time) ทำให้โปรเซสที่ต้องการเวลาในการประมวลผลนานจะต้องหยุดการทำงานเมื่อหมดเวลาควอนตัมและออกจากซีพียูกลับไปต่อคิวในสถานะพร้อมอีกครั้ง ซึ่งจะเป็นการวนรอบเช่นนี้จนกระทั่งโปรเซสนั้นทำงานเสร็จ

49

➢ การจัดการโปรเซสเซอร์ (Processor Management)

RR : Round-Robin

การคำนวณเวลาและตารางเวลาของโปรเซส

โปรเซส	ลำดับการเข้าคิว	เวลาที่ต้องใช้ซีพียู (วินาที)	เวลาที่ต้องรอในคิว (วินาที)	เวลาที่โปรเซสทำงานเสร็จสิ้น (วินาที)
P1	1	10	13	23
P2	2	4	7	11
P3	3	9	13	22

โปรเซส \ วินาทีที่	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
P1	R	o	o	R	o	o	R	o	o	R	o	o	R	o	R	o	R	o	R	o	R	o	R		
P2	o	R	o	o	R	o	o	R	o	o	R														
P3	o	o	R	o	o	R	o	o	R	o	o	R	o	R	o	R	o	R	o	R	o	R			

P2 ↑ (วินาทีที่ 12) P3 ↑ P2 ↑

ให้ o แทนเวลาที่โปรเซสอยู่ในสถานะพร้อม ให้ R แทนเวลาที่โปรเซสอยู่ในสถานะรัน

50

➢ การจัดการโปรเซสเซอร์ (Processor Management)

Priority Queue

การจัดคิวแบบ Priority Queue เป็นการจัดลำดับโปรเซสในสถานะพร้อมตามความสำคัญของโปรเซส โดยจะให้โปรเซสที่มีความสำคัญมากที่สุดเข้าใช้ซีพียูก่อน และให้โปรเซสที่มีความสำคัญน้อยที่สุดใช้ซีพียูสุดท้าย

การพิจารณาความสำคัญของโปรเซสมีดังนี้

1. ผู้ใช้ที่เป็นเจ้าของโปรเซส เช่นโปรเซสของผู้ควบคุมระบบจะมีความสำคัญมากกว่าโปรเซสของผู้ใช้ระบบทั่วไป
2. ประเภทของโปรเซส เช่น โปรเซสที่อยู่ในโหมด Batch จะมีความสำคัญน้อยกว่าโปรเซสในโหมด Interactive (โหมดโต้ตอบ)
3. ระยะเวลาที่โปรเซสเข้ามาในระบบ เช่น โปรเซสที่อยู่ในระบบนานจะมีความสำคัญมากกว่าโปรเซสที่เพิ่งเข้ามาในระบบ

51

➢ การจัดการโปรเซสเซอร์ (Processor Management)

SJN (Shortest-Job-Next)

การจัดคิวแบบ SJN เป็นการคัดเลือกเอาโปรเซสที่ต้องการเวลาในการทำงานน้อยที่สุดเข้าใช้ซีพียูก่อน เพื่อให้โปรเซสที่ใช้เวลาน้อยๆ ได้จบงานเร็วขึ้น ทำให้โปรเซสในระบบที่รออยู่ในคิวมีจำนวนลดลงเร็ว

แต่จะมีผลเสียต่อโปรเซสที่ต้องการเวลาทำงานมาก ซึ่งทำให้เกิดการอดตายได้

52

➢ การจัดการโปรเซสเซอร์ (Processor Management)

SRT (Shortest-Remaining-Time)

การจัดคิวแบบ SRT เป็นการคัดเลือกเอาโปรเซสที่มีเวลาทำงานเหลือน้อยที่สุด เข้าใช้ซีพียูก่อน เพื่อให้โปรเซสที่ต้องการเวลาในการทำงานนานแต่ใกล้จะจบงานแล้วสามารถจบออกไปจากระบบได้เร็วขึ้น

หากมีโปรเซสเข้ามาในระบบพร้อมกัน ก็จะทำการเลือกโปรเซสที่มาก่อนเข้าใช้โปรเซสเซอร์ก่อน แล้วจะเริ่มพิจารณาเวลาที่เหลือในรอบถัดไป

ซึ่งวิธีนี้จะต้องมีการบันทึกเวลาที่โปรเซสทำงานไปแล้ว และบันทึกเวลาที่เหลือของโปรเซสเพื่อนำเวลาที่เหลือนั้นมาจัดลำดับ

53

References

หนังสือ/ตำราอ้างอิง

George, K. J. (2004). *Operating System: Concepts and Principles (2 edition).* India: Shroff Publishers and Distributors Pvt.

Stallings, W. (2013). *Computer Organization and Architecture: Design for Performance.* England: Pearson.

พิเชษฐ์ ศิริรัตนไพศาลกุล. (2544). *ระบบปฏิบัติการ (Operating System).* กรุงเทพฯ: ซีเอ็ดยูเคชั่น.

พีระพนธ์ โสพัศสถิตย์. (2552). *ระบบปฏิบัติการ.* กรุงเทพฯ: จุฬาลงกรณ์มหาวิทยาลัย.

พัชรินทร์ บัวเย็น. (2556). *ไมโครคอมพิวเตอร์และพีชคณิตบูลีน:* บทที่ 4 ระบบปฏิบัติการเบื้องต้น [เอกสารอัดสำเนา]. จันทบุรี: สาขาวิชาคอมพิวเตอร์ธุรกิจ คณะวิทยาการจัดการ มหาวิทยาลัยราชภัฏรำไพพรรณี.

วิเชษฐ์ พลายมาศ. (2552). *ระบบปฏิบัติการ (Operating System).* กรุงเทพฯ: ซีเอ็ดยูเคชั่น.

อรพิน ประวัติบริสุทธิ์. (2551). *คู่มือระบบปฏิบัติการ (Operating System).* กรุงเทพฯ: โปรวิชั่น.

อ้างอิงรูปภาพ

Stallings, W. (2013). *Computer Organization and Architecture: Design for Performance.* England: Pearson.

54