อธิบายครบถ้วนทุกสาระ ทั้งภาคทฤษฎี และหลักปฏิบัติอย่างครบครัน

ระบบปฏิบัติการ
(Operating Systems)

- โครงสร้างของคอมพิวเตอร์ และระบบปฏิบัติการ
- การจัดการหน่วยความจำหลัก
- หน่วยความจำเสมือน
- การจัดการระบบแฟ้ม
- ระบบรับเข้าและส่งออก
- โครงสร้างระบบกระจาย
- การป้องกันและความปลอดภัย
- และอื่นๆ อีกมากมาย

วิเชษฐ พลายมาศ

ซีเอ็ด

ระบบปฏิบัติการ (Operating Systems)

โดย วิเชษฐ์ พลายมาศ

สงวนลิขสิทธิ์ในประเทศไทยตาม พ.ร.บ. ลิขสิทธิ์ © พ.ศ. 2556 โดย วิเชษฐ์ พลายมาศ
ห้ามคัดลอก ลอกเลียน ดัดแปลง ทำซ้ำ จัดพิมพ์ หรือกระทำอื่นใด โดยวิธีการใดๆ ในรูปแบบใดๆ
ไม่ว่าส่วนหนึ่งส่วนใดของหนังสือเล่มนี้ เพื่อเผยแพร่ในสื่อทุกประเภท หรือเพื่อวัตถุประสงค์ใดๆ
นอกจากจะได้รับอนุญาต

ข้อมูลทางบรรณานุกรมของหอสมุดแห่งชาติ

วิเชษฐ์ พลายมาศ.
ระบบปฏิบัติการ (Operating Systems). --กรุงเทพฯ : ซีเอ็ดยูเคชั่น, 2556.
1. การจัดการองค์กร. 2. ระบบสารสนเทศเพื่อการจัดการ.
I. ชื่อเรื่อง.
658

ISBN(e-book) : 978-616-08-0679-9

ผลิตและจัดจำหน่ายโดย

บริษัท ซีเอ็ดยูเคชั่น จำกัด (มหาชน)
SE-EDUCATION PUBLIC COMPANY LIMITED

อาคารทีซีไอเอฟ ทาวเวอร์ ชั้น 19 เลขที่ 1858/87-90 ถนนบางนา-ตราด แขวงบางนา
เขตบางนา กรุงเทพฯ 10260 โทรศัพท์ 0-2739-8000

[หากมีคำแนะนำหรือติชม สามารถติดต่อได้ที่ comment@se-ed.com]

คำนำ

ระบบปฏิบัติการ เป็นวิชาสำคัญวิชาหนึ่งที่ผู้ศึกษาในศาสตร์ด้านคอมพิวเตอร์และเทคโนโลยีสารสนเทศทุกคนจำเป็นต้องเรียนรู้ เพื่อให้มีความเข้าใจเกี่ยวกับซอฟต์แวร์ที่อยู่เบื้องหลังความทรงพลังของฮาร์ดแวร์ และความมหัศจรรย์ของซอฟต์แวร์ทั้งปวง

หนังสือระบบปฏิบัติการที่มีอยู่ในท้องตลาดส่วนใหญ่มักมุ่งเน้นเนื้อหาเชิงประยุกต์ของระบบปฏิบัติการ แต่ตำราที่อธิบายสาระสำคัญของทฤษฎีและหลักการด้านระบบปฏิบัติการได้อย่างสมบูรณ์ยังมีอยู่ไม่มากนัก ผมจึงอาศัยความรู้ที่ได้จากประสบการณ์ที่ได้บรรยายวิชาระบบปฏิบัติการแก่นักศึกษาทั้งระดับปริญญาตรีและโทหลายแห่ง ผสมผสานกับประสบการณ์ที่มีโอกาสได้ทำงานด้านไอซีทีในบริษัทข้ามชาติขนาดใหญ่ ตั้งแต่ระดับปฏิบัติการจนถึงระดับบริหาร รวมทั้งจากการศึกษาค้นคว้าเพิ่มจากตำราต่างประเทศและแหล่งข้อมูลต่างๆ มาเรียบเรียงเป็นเนื้อหาของหนังสือเล่มนี้ โดยหวังว่าจะใช้เป็นตำราภาษาไทยเกี่ยวกับระบบปฏิบัติการพื้นฐานสำหรับนักศึกษาระดับปริญญาตรี ปริญญาโท และผู้สนใจศึกษาศาสตร์ด้านคอมพิวเตอร์ทุกคน เพื่อจะได้มีความเข้าใจสาระหลักของหลักการ และทฤษฎีที่สำคัญของระบบปฏิบัติการได้อย่างเพียงพอที่จะสามารถไปศึกษาเพิ่มเติมจากตำราต่างประเทศได้ต่อไป

หนังสือเล่มนี้จะไม่มีวันเสร็จสมบูรณ์ได้เลย ถ้าปราศจากคำแนะนำที่เต็มไปด้วยความเอาใจใส่และน้ำใจอย่างเสมอต้นเสมอปลายของ คุณสมโภชน์ ชื่นเอี่ยม คุณสุรชัย รุ่งเรืองกุลวณิชย์ และกองบรรณาธิการบริษัท ซีเอ็ดยูเคชั่น จำกัด (มหาชน) ขอขอบคุณทุกท่านที่เกี่ยวข้อง และแหล่งข้อมูลทุกแหล่งที่ได้ให้ความรู้จนกระทั่งก่อให้เกิดเป็นงานชิ้นนี้ขึ้นมาได้

วิเชษฐ์ พลายมาศ
Wichet.P@gmail.com

สารบัญ

บทที่ 1 ความรู้พื้นฐานระบบปฏิบัติการ 11

1. ระบบปฏิบัติการ หรือ OS (Operating System : OS) 12
2. วิวัฒนาการและชนิดของระบบปฏิบัติการ 13
3. หน้าที่ของระบบปฏิบัติการ 30
4. การติดต่อระหว่างอุปกรณ์รอบข้างกับหน่วยประมวลผลกลาง 34
5. การเรียกระบบ (System Calls) 35
6. ระบบปฏิบัติการสำหรับไมโครคอมพิวเตอร์ 38

สรุป 42

แบบทดสอบท้ายบท 44

บทที่ 2 โครงสร้างของคอมพิวเตอร์และระบบปฏิบัติการ 45

1. ระบบคอมพิวเตอร์ 46
2. โครงสร้างของ I/O (I/O Structure) 49
3. โครงสร้างของหน่วยเก็บ (Storage Structure) 52
4. ลำดับชั้นของหน่วยเก็บ 58
5. การป้องกันระดับฮาร์ดแวร์ (Hardware Protection) 60
6. โปรแกรมระบบ (System Programs) 65
7. โครงสร้างระบบ (System Structure) 66
8. เครื่องเสมือน (Virtual Machines) 69

สรุป 71

แบบทดสอบท้ายบท 73

SE-ED

บทที่ 3 กระบวนการ 75
1. แนวคิดเกี่ยวกับกระบวนการ (Process Concept) 76
2. การจัดลำดับกระบวนการ (Process Scheduling) 80
3. การทำงานของกระบวนการ (Operation on Process) 85
4. การประสานกระบวนการ (Cooperating Process) 86
5. การสื่อสารระหว่างกระบวนการ (Interprocess Communication : IPC) 87
6. เทรด (Thread) 91
สรุป 98
แบบทดสอบท้ายบท 100

บทที่ 4 การจัดลำดับซีพียู 101
1. แนวคิดพื้นฐาน 102
2. เงื่อนไขของการจัดลำดับ 106
3. ขั้นตอนวิธีการจัดลำดับ (Scheduling Algorithms) 108
4. การจัดลำดับบนมัลติโปรเซสเซอร์ (Multi-Processor Scheduling) 117
5. การจัดลำดับแบบทำงานแบบทันที (Real-Time Scheduling) 118
สรุป 121
แบบทดสอบท้ายบท 122

บทที่ 5 การประสานเวลากระบวนการ 123
1. แนวคิดพื้นฐานของการประสานเวลากระบวนการ 124
2. ความเป็นมา 125
3. ส่วนวิกฤติ (Critical Section) 127
4. ฮาร์ดแวร์ประสานเวลา (Synchronization Hardware) 129
5. เซมาฟอร์ (Semaphore) 132
6. ปัญหาพื้นฐานของการประสานเวลา 135
7. ตัวเฝ้าสังเกต (Monitors) 141
8. ภาวะติดตาย (Deadlock) 144
สรุป 159
แบบทดสอบท้ายบท 161

SE-ED

บทที่ 6 การจัดการหน่วยความจำหลัก 163
1. แนวคิดพื้นฐาน 164
2. การสับเปลี่ยน 172
3. การจัดสรรหน่วยความจำแบบพื้นที่ติดกัน 174
4. การสลับหน้า 178
5. การแบ่งส่วนหรือเซกเมนต์ 191
6. การผสมผสานหน่วยความจำการสลับหน้ากับการแบ่งส่วน 197
สรุป 199
แบบทดสอบท้ายบท 201

บทที่ 7 หน่วยความจำเสมือน 203
1. ความเป็นมา 204
2. การสลับหน้าตามคำขอทันที 207
3. การแทนที่หน้า 216
4. ตัวอย่างการจัดการหน่วยความจำเสมือนของระบบปฏิบัติการ 231
สรุป 236
แบบทดสอบท้ายบท 237

บทที่ 8 การจัดการระบบแฟ้ม 239
1. แนวคิดเกี่ยวกับแฟ้ม 240
2. วิธีการเข้าถึง 246
3. โครงสร้างสารบบ 249
4. ประสิทธิภาพ 257
5. วิธีการจัดสรร 260
สรุป 268
แบบทดสอบท้ายบท 269

บทที่ 9 ระบบรับเข้าและส่งออก 271
1. อุปกรณ์นำเข้าและส่งออก 272
2. โมดูลของการรับส่งข้อมูล 276
3. โครงสร้างของโมดูลการรับเข้าและส่งออก 279
4. ช่องสื่อสารการรับเข้า/ส่งออก และโปรเซสเซอร์การรับเข้า/ส่งออก 284
5. นาฬิกาและเวลา (Clock and Timers) 289
6. การจัดการความต้องการรับเข้าและส่งออก 293

SE-ED

7. ประสิทธิภาพ 293
สรุป 296
แบบทดสอบท้ายบท 297

บทที่ 10 การจัดการหน่วยเก็บ 299
1. โครงสร้างจานแม่เหล็กหรือดิสก์ (Disk Structure) 300
2. การจัดตาราง (Scheduling) 302
3. การจัดการดิสก์ (Disk Management) 309
4. การจัดการพื้นที่ว่างที่ใช้ในการสับเปลี่ยน (Swap-Space Management) 312
5. ความเชื่อถือได้ของดิสก์ (Disk Reliability) 315
6. ระบบปฏิบัติการกับหน่วยเก็บ 325
สรุป 327
แบบทดสอบท้ายบท 329

บทที่ 11 โครงสร้างระบบกระจาย 331
1. แนวคิดพื้นฐาน 332
2. ประเภทของระบบปฏิบัติการแบบกระจาย 336
3. โทโปโลยี (Topology) 339
4. ประเภทเครือข่าย 341
5. การสื่อสาร 344
6. โปรโตคอลการสื่อสาร (Communication Protocol) 350
7. ข้อที่พึงพิจารณาในการออกแบบ 354
สรุป 359
แบบทดสอบท้ายบท 361

บทที่ 12 การประสานแบบกระจาย 363
1. เวลา (Time) และการจัดอันดับเหตุการณ์ (Event Ordering) 364
2. เวลาและนาฬิกาเชิงตรรกะ (Logical Time and Logical Clock) 367
3. การไม่เกิดร่วมแบบกระจาย (Distributed Mutual Exclusion) 369
4. ภาวะครบหน่วย (Atomicity) 376
5. การควบคุมภาวะพร้อมกัน 379
6. การควบคุมภาวะติดตาย (Deadlock Handling) 384
7. การเลือกตั้ง (Election) 394
สรุป 398
แบบทดสอบท้ายบท 400

SE-ED

บทที่ 13 การป้องกันและความปลอดภัย 401

1. การป้องกันและความปลอดภัย 402
2. โครงสร้างการป้องกัน 405
3. โดเมนของการป้องกัน 409
4. การให้อำนาจ 413
5. การใช้งานแมทริกซ์การเข้าถึง 418
6. การถอนสิทธิ์การเข้าถึง 420
7. การพิสูจน์ตัวจริง 422
8. โปรแกรมอันตราย 425
9. การเข้ารหัสลับ 428

สรุป 429

แบบทดสอบท้ายบท 431

บรรณานุกรม 433

SE-ED

ความรู้พื้นฐานระบบปฏิบัติการ 1

วัตถุประสงค์การเรียนรู้

- เพื่อให้รู้จักสาระสำคัญเกี่ยวกับระบบปฏิบัติการ วิวัฒนาการและชนิดของระบบปฏิบัติการ
- เพื่อให้เข้าใจหน้าที่ของระบบปฏิบัติการในการติดต่อระหว่างอุปกรณ์รอบข้างกับหน่วยประมวลผลกลาง
- เพื่อศึกษาเกี่ยวกับการเรียกระบบ และระบบปฏิบัติการสำหรับไมโครคอมพิวเตอร์

SE-ED

เนื้อหาในบทนี้จะนำเสนอภาพโดยรวมของระบบปฏิบัติการ โดยจะอธิบายสาระสำคัญเกี่ยวกับระบบปฏิบัติการ วิวัฒนาการ และชนิดหน้าที่ของระบบปฏิบัติการ การติดต่อระหว่างอุปกรณ์รอบข้างกับหน่วยประมวลผลกลาง การเรียกระบบ และระบบปฏิบัติการสำหรับไมโครคอมพิวเตอร์

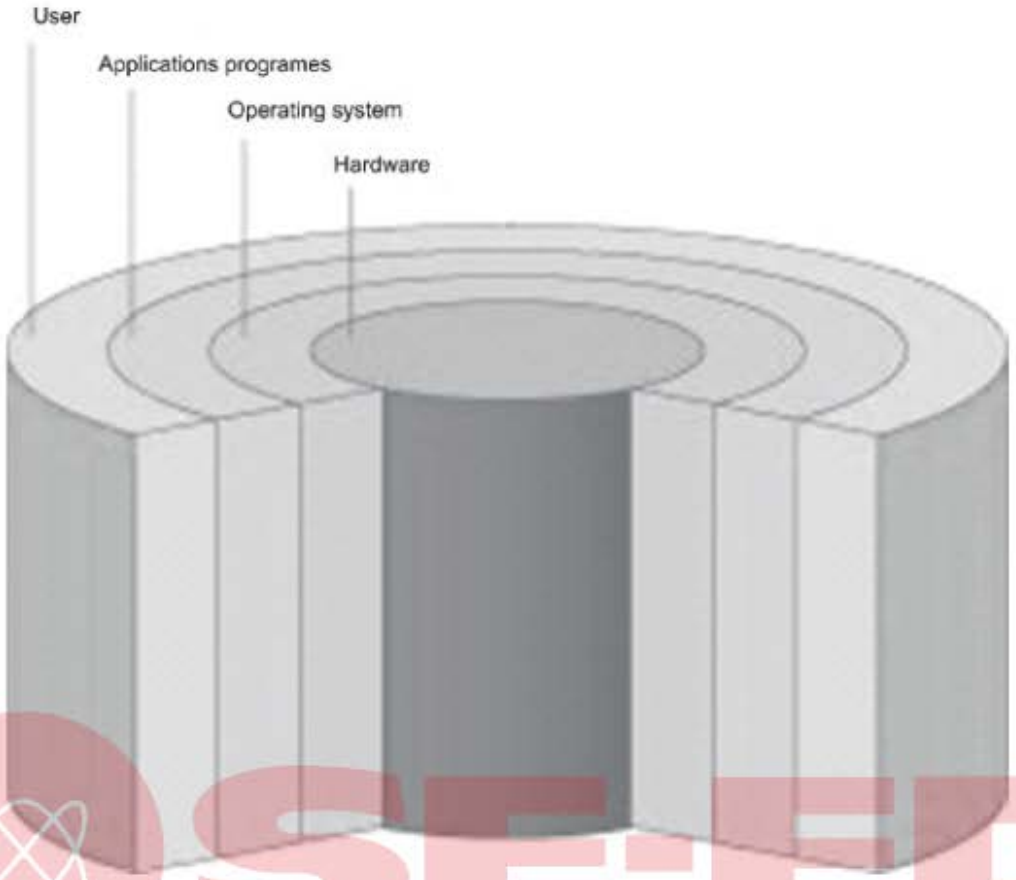
1. ระบบปฏิบัติการ หรือ OS (Operating Systems : OS)

ระบบปฏิบัติการ คือ กลุ่มโปรแกรมที่ทำหน้าที่เป็นตัวกลาง ระหว่างผู้ใช้กับเครื่องคอมพิวเตอร์ มีจุดมุ่งหมายเพื่อจัดหาสภาพแวดล้อมที่เหมาะสม เพื่อให้ผู้ใช้กระทำการกับโปรแกรม (Execute Programs) โดยมีเป้าหมายหลัก เพื่อให้การใช้ระบบคอมพิวเตอร์ของผู้ใช้เป็นไปอย่างสะดวก รวมทั้งการใช้ฮาร์ดแวร์เป็นไปอย่างมีประสิทธิภาพ

ระบบปฏิบัติการประกอบด้วยโปรแกรมหลักของระบบ ที่คอยจัดการเกี่ยวกับปฏิบัติการพื้นฐานของคอมพิวเตอร์ โปรแกรมเหล่านี้จะคอยจัดหาบริการเกี่ยวกับการจัดการทรัพยากรหลายประเภท ควบคุมกิจกรรมในการดำเนินการ จัดเก็บทั้งโปรแกรมและข้อมูลที่ประมวลผล ระบบปฏิบัติการจะทำงานคอยอำนวยความสะดวกเพื่อให้ผู้ใช้สามารถใช้โปรแกรมประยุกต์ได้อย่างสะดวก โดยไม่จำเป็นต้องเกี่ยวข้องกับรายละเอียดที่ซับซ้อนของฮาร์ดแวร์ ดังรูปที่ 1.1 ซึ่งเป็นภาพจำลองพื้นผิวแต่ละชั้นที่แสดงปฏิสัมพันธ์ระหว่างผู้ใช้ โปรแกรมประยุกต์ ระบบปฏิบัติการ และฮาร์ดแวร์ โดยที่ระบบปฏิบัติการจะคอยอำนวยความสะดวกต่อผู้ใช้ในการเข้าถึงฮาร์ดแวร์ผ่านโปรแกรมประยุกต์

ระบบปฏิบัติการ ถูกพัฒนาขึ้นเพื่อใช้สำหรับเครื่องคอมพิวเตอร์ในรุ่นและตระกูลที่เฉพาะเจาะจง เช่น ซูเปอร์คอมพิวเตอร์ยี่ห้อ Cray ใช้ระบบปฏิบัติการชื่อ UNICOS และ COS, IBM เมนเฟรมใช้ MVS และ VM, มินิคอมพิวเตอร์ของ Data General ใช้ AOS และ DG, DEC มินิคอมพิวเตอร์ใช้ VAX/VMS, เครื่องสถานีงาน Next ใช้ NextWare ส่วนคอมพิวเตอร์ประเภทปากกา (pen-based computer) ก็มีระบบปฏิบัติการของตน เช่น PenRight, PenPoint, PenDOS หรือ Windows CE เป็นต้น

ผู้ใช้เครื่องตระกูลแมคอินทอช (Macintosh) จะใช้ Mac OS ส่วนไอบีเอ็ม (IBM) และเครื่องแบบเดียวกับไอบีเอ็ม (IBM-compatible หรือ IBM clone เช่น Compaq, Dell, Zenith) จะใช้ DOS, Windows, NT, OS/2, UNIX หรือ Linux เป็นต้น หรืออาจใช้ NetWare สำหรับไมโครคอมพิวเตอร์ประเภทเครือข่าย



รูปที่ 1.1 แบบจำลองพื้นผิวแต่ละชั้นที่แสดงปฏิสัมพันธ์ระหว่างผู้ใช้ โปรแกรมประยุกต์ ระบบปฏิบัติการ และฮาร์ดแวร์
(ที่มา : H.L. Capron, 2000, p.144)

2. วิวัฒนาการและชนิดของระบบปฏิบัติการ

ระบบปฏิบัติการและสถาปัตยกรรมคอมพิวเตอร์มีอิทธิพลซึ่งกันและกันอย่างมาก ดังสามารถพบได้จากวิวัฒนาการของคอมพิวเตอร์ ตั้งแต่อดีตจากหลอดสุญญากาศที่ไม่มีระบบปฏิบัติการ ซึ่งเป็นสิ่งที่ทำให้ต้องมีการเพิ่มประสิทธิภาพของระบบปฏิบัติการให้มีโครงสร้างที่ทันสมัยเหมาะสำหรับการใช้งาน ระบบปฏิบัติการถูกพัฒนาและเติบโตมาจากระบบเครื่องเมนเฟรมที่ต้องการเพียงระบบปฏิบัติการอย่างง่ายๆ สำหรับทำงานกับแอปพลิเคชั่นเพียงอย่างเดียว จนกระทั่งถึงระบบแบ่งกันใช้เวลาที่ซับซ้อน จากนั้น จึงพัฒนาไปอยู่บนระบบคอมพิวเตอร์ขนาดเล็กอย่างคอมพิวเตอร์แบบตั้งโต๊ะ จนกระทั่งถึงระบบคอมพิวเตอร์แบบมือถือ

ระบบปฏิบัติการมีพัฒนาการที่สำคัญมาโดยลำดับดังนี้

2.1 ระบบเชิงกลุ่ม (Batch Systems)

เครื่องคอมพิวเตอร์ในยุคแรกๆ มีขนาดใหญ่มากดำเนินงานผ่านทางจอเฝ้าคุมหรือคอนโซล (console) มีอุปกรณ์ I/O (I/O devices) เป็นเครื่องอ่านบัตรกระดาษ (card readers) และเครื่องขับแถบบันทึก (tape drives) ส่วนอุปกรณ์แสดงผลเป็นเครื่องพิมพ์รายบรรทัด (line printer) เครื่องขับแถบบันทึก และเครื่องเจาะบัตรกระดาษ (card punches) ผู้ใช้ไม่ได้ติดต่อโดยตรงกับระบบคอมพิวเตอร์ แต่จะเตรียมงานซึ่งประกอบด้วยโปรแกรม ข้อมูล และข้อมูลควบคุมบางอย่างเกี่ยวกับงานนั้น เช่น บัตรควบคุม (control cards) และส่งให้กับพนักงานคุมเครื่อง (operator) หลังจากนั้น (อาจเป็นนาที ชั่วโมง หรือเป็นวัน) ก็จะได้ผลลัพธ์ออกมาซึ่งประกอบด้วยผลของโปรแกรม ตำแหน่งหน่วยความจำ (dump of memory) และรีจิสเตอร์ (registers) ในกรณีที่โปรแกรมมีข้อผิดพลาด

ระบบคอมพิวเตอร์ในยุคแรกๆ เป็นระบบแบบง่ายๆ มีหน้าที่หลักคือส่งการควบคุมโดยอัตโนมัติจากงานหนึ่งไปยังงานถัดไป ระบบปฏิบัติการต้องอยู่ในหน่วยความจำตลอดเวลา ดังรูปที่ 1.2 เพื่อทำให้การประมวลผลเร็วขึ้น ผู้ดูแลระบบทำการรวมกลุ่มงานที่มีความต้องการสิ่งที่เหมือนกันเข้าด้วยกัน และดำเนินงานงานเหล่านั้นผ่านทางคอมพิวเตอร์เป็นกลุ่ม ทำให้โปรแกรมเมอร์สามารถทิ้งโปรแกรมไว้กับพนักงานคุมเครื่องได้ ดังนั้นผู้ดูแลระบบจะเรียงโปรแกรมไว้เป็นกลุ่มที่ต้องการสิ่งที่เหมือนกัน เมื่อคอมพิวเตอร์ว่างก็จะดำเนินงานแต่ละโปรแกรมตามลำดับ ผลลัพธ์จากแต่ละงานจะถูกส่งกลับไปให้กับโปรแกรมเมอร์ที่เป็นเจ้าของโปรแกรม

รูปที่ 1.2 โครงร่างของหน่วยความจำสำหรับระบบเชิงกลุ่มอย่างง่าย

ในสภาพแวดล้อมของการดำเนินโปรแกรมนี้ ซีพียูมักจะว่างอยู่บ่อยๆ เนื่องจากความเร็วของอุปกรณ์ I/O เป็นอุปกรณ์ประเภทกลไก (mechanical I/O devices) มีความเร็วต่ำกว่าอุปกรณ์ที่เป็นแบบอิเล็กทรอนิกส์ แม้แต่ซีพียูที่ทำงานในช่วงความเร็วระดับไมโครวินาที ซึ่งคำสั่งถูกดำเนินการ 1000 คำสั่งต่อวินาที แต่ความเร็วของซีพียูเมื่อเปรียบเทียบกับอุปกรณ์ I/O ก็ยังมีความแตกต่างกันมาก

เมื่อเวลาผ่านไปการปรับปรุงทางด้านเทคโนโลยีและจานบันทึกรุ่นใหม่ๆ ทำให้อุปกรณ์ I/O มีความเร็วมากขึ้น อย่างไรก็ตามความเร็วของซีพียูก็เพิ่มขึ้นมากกว่า

เทคโนโลยีของแผ่นบันทึกรุ่นใหม่ ทำให้ระบบปฏิบัติการสามารถเก็บงานทั้งหมดไว้ในแผ่นบันทึก การเข้าถึงหลายๆ งานได้โดยตรงต้องมีการกำหนดจัดลำดับงาน เพื่อใช้ทรัพยากรและทำงานได้อย่างมีประสิทธิภาพ

2.2 ระบบเชิงกลุ่มแบบอัตโนมัติ (Automatic Batch Systems)

ต่อมาได้มีการพัฒนาระบบปฏิบัติการรุ่นแรกขึ้นมาเพื่อใช้กับเครื่อง IBM 701 ในห้องวิจัยของ General Motor Research Laboratories เรียกว่า การประมวลผลเชิงกลุ่มแบบอัตโนมัติ (automatic batch systems) ระบบปฏิบัติการรุ่นนี้เป็นเพียงโปรแกรมฝังตัวเล็กๆ (resident monitor) ซึ่งวิ่งอยู่ภายในเครื่องตลอดเวลา และทำการส่งมอบการควบคุมเครื่องให้กับโปรแกรมของผู้ใช้ทีละโปรแกรมตามลำดับ

เพื่อให้ระบบปฏิบัติการรู้จักเกี่ยวกับอุปกรณ์ที่ต้องใช้ในโปรแกรม ผู้ใช้จำเป็นต้องเขียนข้อมูลปะหัวปะท้ายโปรแกรม เพื่อจำแนกงานและระบุลักษณะงาน เช่น ตัวแปลภาษา ตู้เทป หมายเลขของม้วนเทป เป็นต้น ภาษานี้เรียกว่า **ภาษาควบคุมงาน** JCL (job control language) ในยุคนี้ความเกี่ยวข้องระหว่างผู้ใช้กับงานจึงหมดไป งานจะถูกตระเตรียมและส่งมอบ จนกระทั่งผลลัพธ์ปรากฏออกมา (ซึ่งอาจใช้เวลาเป็นนาที ชั่วโมง หรือวัน) ช่วงเวลารอคอย ตั้งแต่ส่งมอบงานเข้าเครื่อง (job submission) จนกระทั่งงานเสร็จสมบูรณ์ (job completion) เรียกว่า **ช่วงเวลารอคอย** (turnaround time)

ปัญหาในเรื่องความแตกต่างของความเร็วระหว่างอุปกรณ์ I/O กับซีพียู ก่อให้เกิดการพัฒนาเพื่อเพิ่มประโยชน์ใช้สอยซีพียู (CPU utilization) ในระยะต่อมาได้แก่ ระบบออฟไลน์ (offline) และสพูลิ่ง (spooling) เข้ามาช่วย

2.3 ระบบบัฟเฟอร์ (Buffering)

ระบบบัฟเฟอร์มีหลักการทำงานคือ ให้หน่วยรับและส่งข้อมูลทำงานขนานไปพร้อมกับหน่วยประมวลผลกลางให้มากที่สุด เพื่อลดเวลาที่หน่วยประมวลผลกลาง รออุปกรณ์ I/O ในขณะที่หน่วยประมวลผลกลางกำลังประมวลผล หน่วยรับข้อมูลจะอ่านข้อมูลถัดไป มาไว้ที่หน่วยความจำที่เรียกว่า **บัฟเฟอร์** (buffer)

บัฟเฟอร์ของหน่วยรับและส่งข้อมูล ได้แก่ เครื่องอ่านบัตร และเครื่องพิมพ์รายบรรทัด โดยหน่วยรับและส่งข้อมูล จะทำงานติดต่อกับหน่วยประมวลผลกลางโดยตรง ซึ่งเรียกว่า **ระบบต่อตรง** (On-Line) ถ้าการอ่าน/เขียนสำหรับข้อมูลแต่ละหน่วยใช้เวลาเท่ากับการประมวลผลแต่ละหน่วยพอดี I/O กับหน่วยประมวลผลกลางจะไม่มีการรอคอยซึ่งกันและกัน

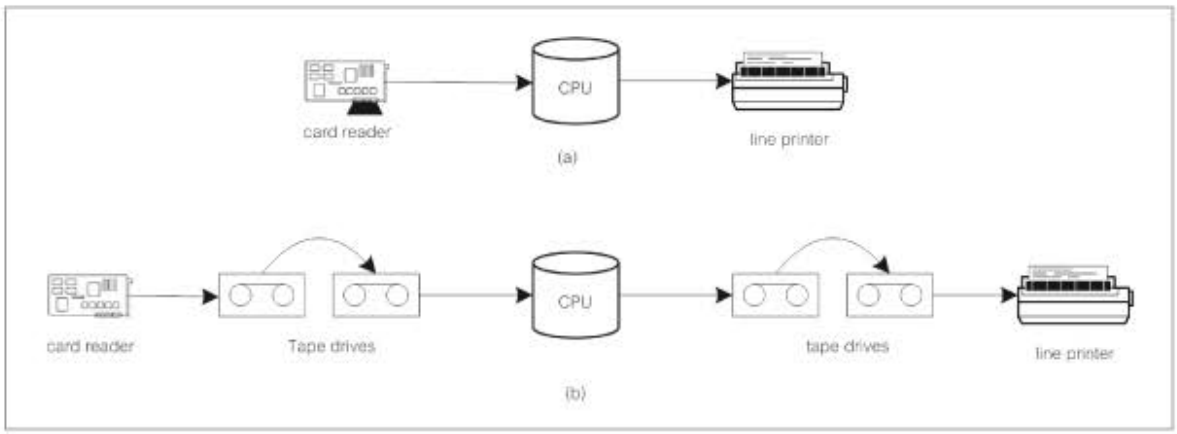
ปัญหาสำคัญของระบบนี้คือ ความเหลื่อมล้ำด้านเวลา หน่วยประมวลผลกลางจะมีความเร็วสูงกว่าอุปกรณ์ประเภท I/O มาก อันเนื่องมาจากข้อจำกัดเชิงกลไก แม้จะมีบัฟเฟอร์เข้ามาช่วย ซีพียูก็ต้องรอ I/O อยู่ดี หากเป็นงานประเภทต้องใช้ I/O มาก (I/O bounded) หน่วยประมวลผลกลางต้องรอ I/O ถ้าเป็นงานประเภทต้องใช้หน่วยประมวลผลกลางมาก (CPU bounded) ทำให้อุปกรณ์ I/O ต้องรอหน่วยประมวลผลกลาง

2.4 การประมวลผลออฟไลน์ (Off-Line Processing)

การประมวลผลออฟไลน์ หรือแบบไม่เชื่อมตรง มีความแตกต่างจากระบบเชื่อมโดยตรงหรือออนไลน์ ดังรูปที่ 1.3 ระบบออฟไลน์จะใช้เทปแม่เหล็ก (magnetic tape) แทนเครื่องอ่านบัตร และเครื่องพิมพ์ที่มีความเร็วต่ำมาก โดยคั่นระหว่างหน่วยรับเข้ากับซีพียู และระหว่างซีพียูกับหน่วยส่งออก การถ่ายเทข้อมูลผ่านเทปแม่เหล็กสามารถทำได้โดยใช้เครื่องอ่านบัตร และเครื่องพิมพ์ที่ได้รับการออกแบบเป็นพิเศษ ให้สามารถถ่ายเทข้อมูลโดยไม่ต้องผ่านซีพียู หรือเพิ่มหน่วยประมวลผลขนาดเล็กที่ทำหน้าที่เฉพาะด้านนี้ เรียกว่า I/O **โปรเซสเซอร์** (I/O Processor)

อย่างไรก็ตาม ระบบออฟไลน์ก็ยังมีข้อจำกัดดังนี้

- โปรแกรมต้องผ่านขั้นตอนมากขึ้น จะมีค่าใช้จ่ายอื่น (overhead) สูง
- ต้องการระบบปฏิบัติการที่ซับซ้อนมากขึ้น
- ในการเก็บข้อมูลลงเทป จำเป็นต้องรอให้มีหลายๆ โปรแกรมเสียก่อน จึงค่อยนำเข้าสู่เครื่องใหญ่เสียทีเดียว แม้จะทำให้ประโยชน์ใช้สอยของซีพียูดีขึ้น แต่ผู้ใช้ต้องมีเวลารอคอยมากขึ้น



รูปที่ 1.3 การทำงานของอุปกรณ์ I/O
(a) แบบออนไลน์ (b) แบบออฟไลน์

2.5 ระบบสพูลิ่ง (Spooling)

ในยุคนี้เทคโนโลยีสื่อบันทึกได้พัฒนาจากเทปแม่เหล็กเป็นดิสก์ซึ่งได้นำไปสู่การเปลี่ยนโฉมหน้าของการออกแบบระบบปฏิบัติการและแอปพลิเคชั่นในเวลาต่อมา ระบบสพูลิ่ง หรือระบบการเก็บพักได้ใช้ระบบดิสก์แทนเทปแม่เหล็ก

เนื่องจากเทปไม่สามารถทำการประมวลผลข้อมูลในเทป ขณะที่กำลังถ่ายเทข้อมูลจากเครื่องอ่านบัตรลงในเทปม้วนเดียวกันได้ ซึ่งเทปแม่เหล็กกับดิสก์มีข้อแตกต่างกันดังนี้

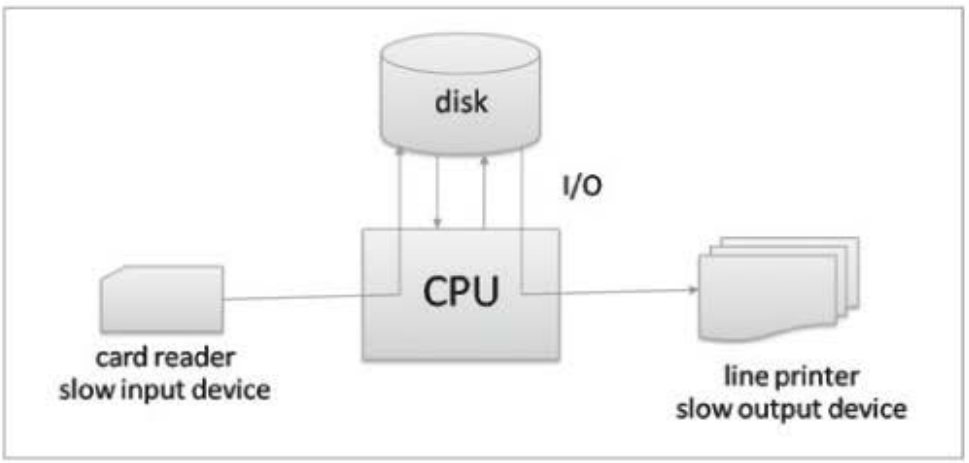
เทปแม่เหล็ก

- เทปเป็นอุปกรณ์ประเภทที่มีการเข้าถึงข้อมูลแบบเรียงลำดับ (sequential-access device)
- การประมวลผลโดยอาศัยเทปเป็นแบบออฟไลน์

ดิสก์

- ดิสก์เป็นอุปกรณ์ประเภทที่มีการเข้าถึงข้อมูลแบบสุ่ม (random-access device) สามารถเข้าถึงข้อมูลโดยตรง (direct access) ได้ จึงทำให้สามารถแยกงานออกจากกัน โดยสร้างตารางบ่งบอกว่าข้อมูลหรือผลลัพธ์อยู่ที่ส่วนใดของดิสก์
- การประมวลผลของดิสก์เป็นแบบเชื่อมโดยตรงกับซีพียู (on-line system) ดังนั้น หน่วยที่ใช้ในการถ่ายเทข้อมูลระหว่างดิสก์กับอุปกรณ์ I/O จึงต้องเป็นตัวเดียวกับหน่วยที่ใช้ประมวลผลงานของผู้ใช้ โดยอาศัยโปรแกรมพิเศษตัวหนึ่ง (เรียกว่า โปรแกรม spool) วิ่งคู่ขนานไปกับโปรแกรมของผู้ใช้ เพื่อทำการถ่ายเทข้อมูลกับดิสก์ จึงก่อให้เกิดระบบหลายโปรแกรม (multiprogramming) ขั้นพื้นฐาน

หลักการใช้ดิสก์แทนอุปกรณ์รับและแสดงผล เรียกว่า **การเก็บพัก หรือสพูลิ่ง** มาจากคำว่า simultaneous peripheral operating on-line : SPOOL ดังรูปที่ 1.4



รูปที่ 1.4 ระบบสพูลิ่ง

ข้อแตกต่างระหว่างระบบบัฟเฟอร์กับสพูลิ่ง

- ระบบบัฟเฟอร์ เป็นการเหลื่อมกันระหว่างการประมวลผลกับหน่วยนำเข้า/ส่งออก ของโปรแกรมเดียวกัน ซึ่งไม่อาจทำได้มากนัก ด้วยข้อจำกัดของขั้นตอนการทำงานของโปรแกรมนั้นๆ
- ระบบสพูลิ่ง เป็นการเหลื่อมกันของการประมวลผลกับการรับและแสดงผลของอีกงานหนึ่ง โดยผ่านโปรแกรมสพูล
- ระบบสพูลิ่ง สามารถจัดการงานที่ถูกป้อนเข้ามาแบบเรียงลำดับได้โดยอิสระ เกิดเป็นกองกลางงาน (job pool) ซึ่งระบบปฏิบัติการสามารถเลือกงานเข้าประมวลผลตามความเหมาะสม ก่อให้เกิดระบบการจัดลำดับงาน (job scheduling)

2.6 ระบบหลายโปรแกรม (Multiprogrammed Systems)

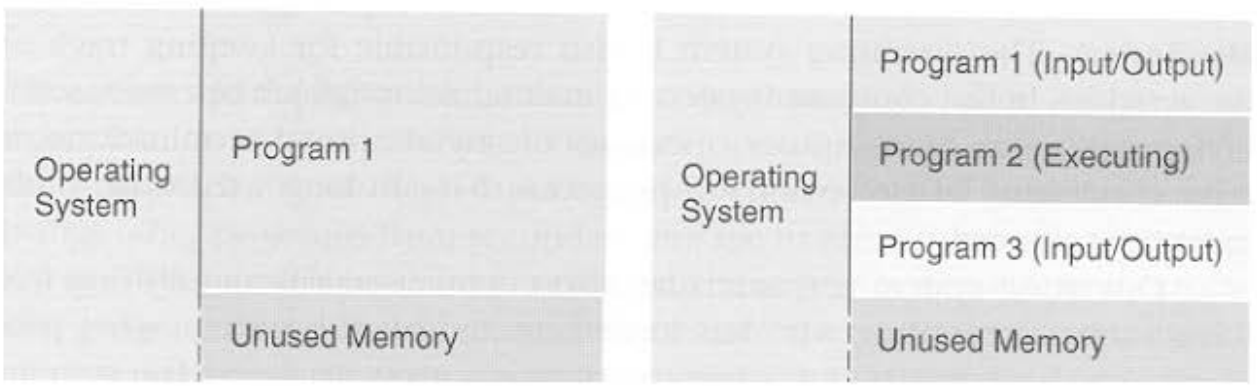
สิ่งสำคัญที่สุดของการจัดลำดับงาน คือความสามารถของการทำงานในลักษณะหลายโปรแกรม (multiprogramming) ผู้ใช้ไม่สามารถใช้ซีพียูหรืออุปกรณ์ I/O แต่เพียงผู้เดียวตลอดเวลา การทำงานแบบหลายโปรแกรมเป็นการเพิ่มการใช้งานซีพียูโดยการจัดงานให้กับซีพียูทำอยู่ตลอดเวลา

รูปที่ 1.5 โครงร่างหน่วยความจำสำหรับระบบหลายโปรแกรม

แนวคิดดังกล่าวเป็นดังนี้คือ ระบบปฏิบัติการจะเก็บงานหลายๆ งานไว้ในหน่วยความจำพร้อมๆ กัน ดังรูปที่ 1.5 งานกลุ่มนี้เป็นกลุ่มย่อยของงานที่เก็บอยู่ในงานกองกลาง ระบบปฏิบัติการจะหยิบหนึ่งงานในหน่วยความจำมาดำเนินการ จนกระทั่งงานนั้นอาจต้องรอให้งานบางอย่างเสร็จสมบูรณ์ เช่น การดำเนินการของอุปกรณ์ I/O ในระบบที่ไม่ใช่แบบหลายโปรแกรม ซีพียูต้องหยุดรออยู่ว่างๆ แต่สำหรับระบบแบบหลายโปรแกรม ระบบปฏิบัติการต้องสับเปลี่ยนไปทำงานอื่น และเมื่องานนั้นต้องหยุดรอบางอย่างอีก ซีพียูก็จะสลับไปทำงานอื่นอีก เป็นเช่นนี้ต่อไปเรื่อยๆ ในที่สุดงานแรกเสร็จสิ้นการรอและได้ซีพียูกลับมาใช้งานอีก ตราบใดที่ซีพียูมีอย่างน้อย 1 งานที่ต้องทำ ก็จะทำให้ซีพียูไม่มีเวลาว่างได้

ระบบปฏิบัติการแบบหลายโปรแกรมนั้น ทุกงานที่เข้ามาในระบบจะถูกเก็บไว้ในงานกองกลาง ซึ่งประกอบด้วยกระบวนการทั้งหมดที่เก็บอยู่บนดิสก์รอการใช้หน่วยความจำหลัก ถ้ามีหลายงานพร้อมที่จะถูกนำเข้าสู่หน่วยความจำพร้อมกัน แต่ไม่มีพื้นที่หน่วยความจำมากพอสำหรับงานทั้งหมดเหล่านั้น ระบบปฏิบัติการต้องเลือกระหว่างงานเหล่านั้นว่างานใดจะสามารถเข้าใช้หน่วยความจำได้ก่อน การตัดสินใจนี้เรียกว่า การสับเปลี่ยนงาน เมื่อระบบปฏิบัติการเลือกงานจากงานกองกลาง มันจะนำเอางานนั้นเข้ามาในหน่วยความจำเพื่อดำเนินการ หากมีหลายงานอยู่ในหน่วยความจำพร้อมกันต้องมีวิธีการจัดการหน่วยความจำ ถ้ามีหลายงานพร้อมดำเนินการในเวลาเดียวกัน ระบบปฏิบัติการต้องเลือกระหว่างงานเหล่านั้นว่าจะดำเนินการงานใดก่อน ซึ่งการตัดสินใจเลือกดังกล่าวเป็นการจัดลำดับการใช้ซีพียู (CPU scheduling) และสุดท้ายการที่มีหลายๆ งานดำเนินไปพร้อมๆ กัน ความสามารถของแต่ละงานที่จะมีผลกระทบต่องานอื่นต้องถูกจำกัดในทุกขั้นตอนของระบบปฏิบัติการ ซึ่งรวมถึงการจัดลำดับกระบวนการ (process scheduling) หน่วยเก็บข้อมูลแบบจานบันทึก (disk storage) และการจัดการหน่วยความจำ (memory management)

SE-ED



(a) Single Program Environment (b) Multiprogramming

รูปที่ 1.6 เปรียบเทียบการทำงานระหว่างสภาพแวดล้อมแบบโปรแกรมเดียว และแบบหลายโปรแกรม
*(**ที่มา** : Kenneth C. Laudon and Jane Price Laudon, 1998, p.324)*

ในรูปที่ 1.6 เปรียบเทียบการทำงานภายใต้สภาพแวดล้อมแบบโปรแกรมเดียวกับหลายโปรแกรม โดยที่แบบหลายโปรแกรมจะใช้ประโยชน์ซีพียูอย่างเต็มประสิทธิภาพ แบบ (a) เป็นการทำงานภายใต้สภาพแวดล้อมแบบโปรแกรมเดียว ในขณะกระทำการซีพียูต้องหยุดรอจนกว่า I/O จะเสร็จสิ้นกระบวนการ ส่วนแบบ (b) เป็นการทำงานแบบหลายโปรแกรม ระบบปฏิบัติการจะยอมให้ซีพียูทำงานได้หลายโปรแกรมในเวลาเดียวกัน เช่น ซีพียูกระทำการกับโปรแกรมที่ 1 จนกระทั่งโปรแกรมร้องขอ I/O ซีพียูจะหยุดโปรแกรมนั้นไว้แล้วไปกระทำการกับโปรแกรมที่ 2 จนกระทั่งโปรแกรมที่ 2 ร้องขอ I/O ซีพียูจะสลับกลับไปกระทำการกับโปรแกรมที่ 1 ต่อ ถ้าการใช้ I/O ของโปรแกรมที่ 1 ยังไม่แล้วเสร็จ ซีพียูจะไปกระทำการกับโปรแกรมที่ 3

2.7 ระบบการแบ่งกันใช้เวลา (Time-Sharing Systems)

การแบ่งกันใช้เวลา (time-sharing) เป็นเทคนิคที่ผู้ใช้หลายคนสามารถแบ่งปันการใช้ทรัพยากรคอมพิวเตอร์ร่วมกันในเวลาเดียวกัน โดยระบบปฏิบัติการจะทำการแบ่งเวลาออกเป็นช่วงสั้นๆ เรียกว่า **เสี้ยวเวลา (time slice)** ในหน่วยมิลลิวินาที (millisecond) หรือวินาที เวียนกระทำการกับโปรแกรมหรือกระบวนการของผู้ใช้เป็นลำดับไป เพื่อช่วยให้เวลาโดยรวมของงานทั้งหมดเสร็จเร็วขึ้น ช่วยให้เวลาการตอบสนอง (response time) ต่อผู้ใช้ทั้งหมดดีขึ้นและซีพียูทำงานได้เต็มประสิทธิภาพ เวลาการตอบสนอง หมายถึง ช่วงเวลาตั้งแต่ผู้ใช้ป้อนคำสั่งให้คอมพิวเตอร์จนกระทั่งคอมพิวเตอร์ตอบรับมา อย่างไรก็ตาม ในระบบการแบ่งกันใช้เวลานี้การเพิ่มขึ้นของจำนวนผู้ใช้ระบบจะมีผลกระทบต่ออัตราการตอบสนองด้วย

ระบบคอมพิวเตอร์แบบโต้ตอบ (interactive or hands-on computer system) ได้เตรียมการติดต่อโดยตรงระหว่างผู้ใช้และระบบคอมพิวเตอร์ให้ผู้ใช้สามารถป้อนคำสั่งไปยังระบบปฏิบัติการหรือโปรแกรมได้โดยตรงโดยการใช้แป้นพิมพ์หรือเมาส์ และรอผลลัพธ์กลับมา เวลาในการตอบสนองจะต้องเป็นระยะเวลาสั้นๆ โดยทั่วไปภายใน 1 วินาที หรือมากกว่านั้น

ระบบปฏิบัติการแบบแบ่งเวลา ทำให้ผู้ใช้สามารถใช้คอมพิวเตอร์พร้อมกันได้หลายคน เนื่องจากคำสั่งในระบบแบบแบ่งเวลามักจะมีขนาดสั้นๆ ซึ่งต้องการเวลาของซีพียูเพียงเล็กน้อยสำหรับแต่ละผู้ใช้ ดังนั้นการที่ระบบปฏิบัติการสามารถสลับสับเปลี่ยนจากผู้ใช้หนึ่งไปยังอีกผู้ใช้หนึ่งเกิดขึ้นอย่างรวดเร็ว ทำให้ผู้ใช้แต่ละคนมีความรู้สึกเหมือนกับว่าได้ใช้คอมพิวเตอร์ทั้งระบบแต่เพียงผู้เดียว แต่ความจริงแล้วคอมพิวเตอร์หนึ่งเครื่องถูกแบ่งปันกันใช้ระหว่างผู้ใช้หลายคน

ระบบปฏิบัติการแบบแบ่งเวลานั้นผู้ใช้แต่ละคนมีส่วนของโปรแกรมอยู่ในหน่วยความจำ โปรแกรมที่ถูกนำเข้าไปในหน่วยความจำและกำลังดำเนินการอยู่เรียกว่า **กระบวนการ (process)** ซึ่งโดยปกติจะดำเนินการเป็นระยะเวลาสั้นๆ ก่อนดำเนินการเสร็จหรือต้องปฏิบัติการเกี่ยวกับ I/O การโต้ตอบกับอุปกรณ์ I/O ทำงานด้วยความเร็วระดับที่คนสามารถโต้ตอบได้ แต่จะช้ามากเมื่อเทียบกับคอมพิวเตอร์ ดังนั้นระบบปฏิบัติการจะสลับสับเปลี่ยนไปยังโปรแกรมของผู้ใช้คนอื่น แทนที่จะรออยู่ว่างๆ เพื่อรอการดำเนินการของอุปกรณ์ I/O ของงานแรกเสร็จสมบูรณ์

ระบบปฏิบัติการแบบแบ่งเวลามีความซับซ้อนมากกว่าระบบปฏิบัติการแบบหลายโปรแกรม หลายๆ งานต้องถูกเก็บอยู่ในหน่วยความจำพร้อมกัน ดังนั้นระบบต้องมีการจัดการหน่วยความจำและการป้องกัน (protection) เพื่อให้ได้รับเวลาตอบสนองที่สมเหตุสมผล งานอาจต้องถูกสลับเข้าสลับออกระหว่างหน่วยความจำหลักและจานบันทึก โดยทั่วไปใช้หน่วยความจำเสมือน (virtual memory) ซึ่งเป็นเทคนิคที่อนุญาตให้ดำเนินงานที่ไม่ได้อยู่ในหน่วยความจำทั้งหมด ประโยชน์ที่เห็นได้ชัดของวิธีนี้ก็คือ โปรแกรมสามารถมีขนาดใหญ่กว่าหน่วยความจำทางกายภาพที่มีอยู่จริง การจัดการแบบนี้ทำให้โปรแกรมเมอร์ไม่ต้องกังวลเกี่ยวกับข้อจำกัดของหน่วยความจำ ระบบแบบแบ่งเวลาต้องเตรียมระบบแฟ้มข้อมูลซึ่งต้องมีการเตรียมเกี่ยวกับการจัดการแผ่นบันทึก (disk management) ระบบแบบแบ่งเวลาต้องเตรียมเกี่ยวกับกลไกสำหรับการดำเนินคำสั่งพร้อมๆ กัน ซึ่งต้องใช้การสลับสับเปลี่ยนกันใช้ซีพียู และเพื่อรับประกันว่าการดำเนินคำสั่งเป็นไปตามลำดับ ระบบต้องเตรียมเกี่ยวกับกลไกสำหรับการติดต่อสื่อสารและการประสานเวลาของงาน (job synchronization and communication) อีกทั้งต้องรับประกันว่าไม่มีงานใดติดค้างอยู่ในการรองานใดงานหนึ่งตลอดไป ที่เรียกว่า **ภาวะติดตาย** (deadlock)

ความคิดเกี่ยวกับระบบแบบแบ่งเวลาถูกแสดงในต้นปี 1960 แต่เนื่องจากระบบแบบแบ่งเวลานั้นสร้างยากและราคาแพง จึงไม่เป็นที่นิยมจนกระทั่งต้นปี 1970 ถึงแม้ว่าการประมวลผลเชิงกลุ่มยังคงใช้อยู่ แต่ระบบส่วนใหญ่ในปัจจุบันจะเป็นแบบแบ่งเวลา ระบบแบบหลายโปรแกรมและระบบแบบแบ่งเวลาเป็นแกนกลางของระบบปฏิบัติการสมัยใหม่

เพื่อความเข้าใจที่ชัดเจน ต่อไปนี้จะสรุปแนวคิดสำคัญของการออกแบบและพัฒนาระบบปฏิบัติการสมัยใหม่ ที่มีรากฐานมาจากระบบหลายโปรแกรม และการแบ่งเวลากันใช้เพื่อสนับสนุนให้การจัดการภารกิจหลายชนิด จากผู้ใช้เพียงคนเดียวหรือหลายคนในเวลาเดียวกันถูกดำเนินไปอย่างเหมาะสม และมีประสิทธิภาพ ระบบปฏิบัติการทั้งในเครื่องขนาดใหญ่และไมโครคอมพิวเตอร์จะใช้วิธีการทำงานในลักษณะต่อไปนี้ เพื่อขยายขีดความสามารถในการทำงานของระบบปฏิบัติการ ซึ่งได้แก่ การทำงานแบบหลายภารกิจ (multitasking) แบบมัลติเทรด (multithreading) และหน่วยเก็บเสมือน (virtual storage)

แบบหลายภารกิจ (Multitasking)

หมายถึง สมรรถนะการทำงานแบบหลายโปรแกรมของระบบปฏิบัติการสำหรับผู้ใช้คนเดียว อย่างเช่นในคอมพิวเตอร์ส่วนบุคคล โดยที่ผู้ใช้คนหนึ่งสามารถดำเนินการ (run) หลายโปรแกรมในเวลาเดียวกันบนเครื่องเดียวกัน เช่น โบรกเกอร์ขายหุ้นกำลังเขียนจดหมายถึงลูกค้าด้วยโปรแกรมประมวลผลคำ ในขณะที่อีกโปรแกรมหนึ่งกำลังบันทึกและปรับปรุงข้อมูลบัญชีลูกค้า ในระบบหลายภารกิจ โบรกเกอร์สามารถดูหน้าจอของทั้งสองโปรแกรมได้ในเวลาเดียวกัน โดยไม่จำเป็นต้องยุติจากโปรแกรมประมวลผลคำออกไปยังระบบปฏิบัติการ แล้วค่อยเรียกใช้โปรแกรมที่ควบคุมฐานข้อมูลลูกค้า

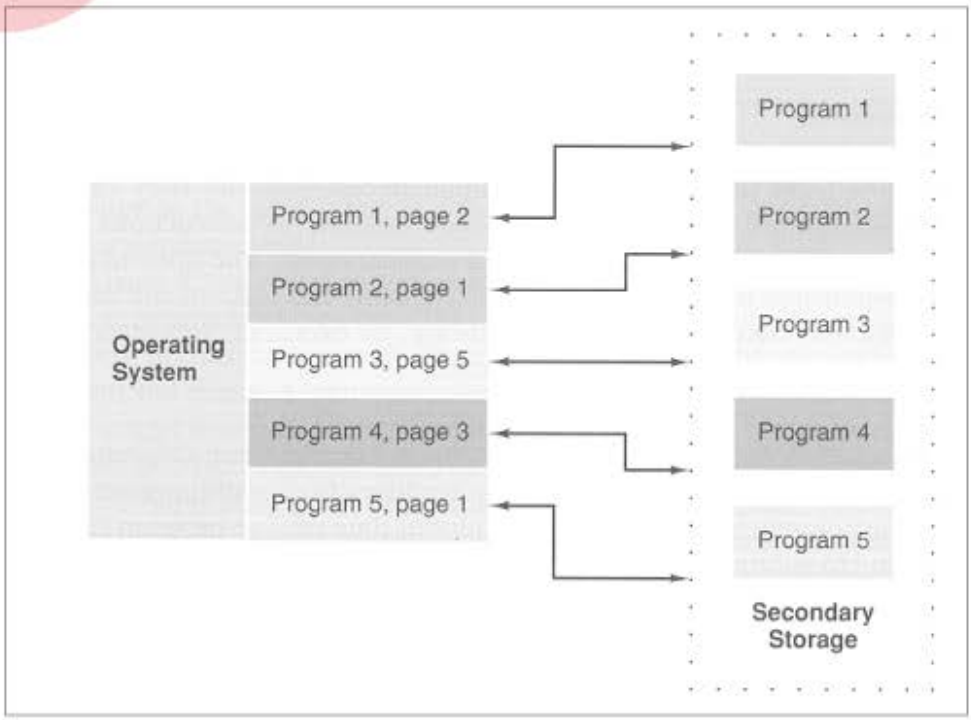
แบบมัลติเทรด (Multithreading)

ระบบปฏิบัติการบางรุ่น สามารถแบ่งโปรแกรมหรือกระจายกระบวนการออกเป็นกระบวนการย่อยๆ เรียกว่า เทรดหรือสายโยงใย (thread) ซึ่งเป็นหน่วยย่อยที่เล็กที่สุดของกระบวนการที่สามารถกระทำการได้ (execution unit of process หรือ executable entity) ในระบบมัลติเทรด ระบบปฏิบัติการจะซอยโปรแกรมหรือกระบวนการออกเป็นหน่วยย่อยๆ แล้วทำงานคู่ขนานกันไป ซึ่งจะช่วยให้โปรแกรมทำงานเสร็จเร็วขึ้น

หน่วยเก็บเสมือน (Virtual Storage)

เป็นเทคนิคในการแบ่งปันหน่วยความจำเพื่อให้สามารถกระทำการกับหลายโปรแกรมได้อย่างมีประสิทธิภาพ บางทีเรียกว่า **หน่วยความจำเสมือน** (virtual memory) โดยวิธีการแบ่งส่วนของโปรแกรมออกเป็นสองส่วน โดยส่วนแรกจะเก็บเฉพาะส่วนที่จำเป็นสำหรับการกระทำการไว้ในหน่วยความจำหลัก จึงเรียกส่วนนี้ว่า **หน่วยเก็บจริง** (real storage) และส่วนที่เหลือจะเก็บไว้ในหน่วยเก็บรอง เช่น ดิสก์ จึงเรียกส่วนนี้ว่า **หน่วยเก็บเสมือน** (virtual storage) เนื่องจากในขณะที่โปรแกรมกำลังถูกกระทำการ ซีพียูอาจต้องการบางส่วนของโปรแกรมเท่านั้นไม่ใช่ทั้งหมด จึงสามารถเลือกเก็บเฉพาะส่วนที่จำเป็นต้องใช้ไว้ในหน่วยความจำ ส่วนของโปรแกรมที่เหลือจะถูกแบ่งไปเก็บไว้ในหน่วยเก็บรอง โดยจำลองพื้นที่ในหน่วยเก็บให้มีลักษณะสอดคล้องกับหน่วยความจำที่มีขนาดคงที่ เรียกว่า **หน้า** (pages) หรือขนาดที่แปรเปลี่ยนได้เรียกว่า **ส่วน** (segments) เมื่อซีพียูต้องการโปรแกรมส่วนอื่นที่ไม่มีอยู่ในหน่วยความจำ ก็จะไปนำส่วนโปรแกรมที่เหลือจากหน่วยเก็บรองมายังหน่วยความจำ กิจกรรมเหล่านี้ดำเนินไปได้โดยอาศัยระบบปฏิบัติการ

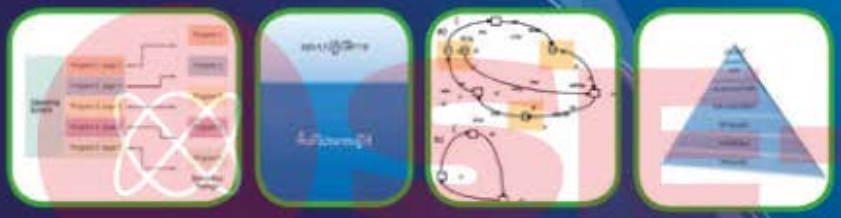
SE-ED



รูปที่ 1.7 หน่วยเก็บเสมือนเป็นเทคนิคที่ช่วยขยายหน่วยความจำหลักของคอมพิวเตอร์
*(**ที่มา** : Kenneth C. Laudon et. al., 1998, p. 165)*

ระบบปฏิบัติการ
(Operating Systems)

ในหนังสือเล่มนี้ กล่าวถึงวิธีการใช้งานระบบปฏิบัติการที่เริ่มตั้งแต่ความรู้ขั้นพื้นฐานอธิบายถึงสาระสำคัญของระบบปฏิบัติการ จนถึงวิวัฒนาการต่อๆ มา หน้าที่ การเรียกใช้งาน แนวคิดการจัดการเกี่ยวกับกระบวนการ การจัดการแฟ้ม ระบบรับเข้าและส่งออก ปัญหาและวิธีแก้ไขเกี่ยวกับการประสานเวลา รวมถึงการป้องกันและความปลอดภัย พร้อมแบบทดสอบท้ายบท เพื่อให้ผู้ศึกษาทดสอบความเข้าใจได้อีกด้วย



วิเชษฐ พลายมาศ

ผ่านงานภาคสนามด้าน ICT ทั้งในบริษัทมหาชนและบริษัทข้ามชาติ ตั้งแต่ระดับปฏิบัติการจนถึงระดับบริหาร มีความมุ่งมั่นในการยกระดับองค์ความรู้ด้าน ICT ให้แก่เด็กๆ และผู้ด้อยโอกาสในชนบท ปัจจุบันมีอาชีพเป็น ผู้บรรยาย (lecturer) และให้คำปรึกษา (consultant) อีกทั้งกำลังเรียนปริญญาเอก คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

หนังสือเล่มนี้สำหรับ

ผู้เริ่มต้น ▲ ระดับกลาง ▲ ระดับสูง

คอมพิวเตอร์/ระบบปฏิบัติการ

ISBN 978-974-212-985-9

9 789742 129859

199 บาท

www.se-ed.com