บทที่ 2

ทฤษฎีแนวคิดที่สำคัญ

2.1 แนวความคิด

จากความสำคัญและความเป็นมาของปัญหาที่ได้กล่าวไปแล้วข้างต้น ประกอบกับแนวคิดและทฤษฎีที่เกี่ยวข้อง ทำให้เกิดแนวความคิดในการสร้างงานวิจัยชิ้นนี้ ซึ่งนับว่าจะเป็นประโยชน์ต่อสังคมและส่วนรวมเป็นอย่างมาก และแนวคิดของผู้เขียนบทความที่ถูกตีพิมพ์ในวารสารต่าง ๆ เช่น

สมเกียรติ เอกจริยาวัฒน์ (2531) ได้กล่าวถึง DIF ไฟล์ไว้ว่า "ในการใช้งานโปรแกรมสำเร็จรูปหลาย ๆ โปรแกรมมักพบปัญหาอย่างหนึ่งคือ การแลกเปลี่ยนข้อมูลระหว่างโปรแกรม เช่น เมื่อใช้งานเวิร์ดโปรเซสซิ่ง ดีเบส โลตัส หรือโปรแกรมกราฟิกอื่น ๆ ข้อมูลที่สร้างขึ้นในแต่ละโปรแกรมจะมีลักษณะเฉพาะของตัวเอง หรือที่เรียกว่าไฟล์ฟอร์แมต โปรแกรมต่าง ๆ ชนิดกันจะอ่านข้อมูลกันไม่ได้ เช่น โลตัสสร้างไฟล์แบบ .WK1 หรือ .WKS ดีเบสสร้างไฟล์แบบ .DBF ซูเปอร์แคลค์สร้างไฟล์แบบ .CAL ไมโครซอฟต์ชาร์ตสร้างไฟล์แบบ .MC การแลกเปลี่ยนข้อมูลซึ่งกันและกันจึงเป็นเรื่องยากมากที่จะทำสมบูรณ์หมด เพราะมีฉะนั้นแล้วทุกโปรแกรมจะต้องสร้างยูทิลิตี้ของตนเองในการแปลงคู่ไฟล์กัน ซึ่งเป็นเรื่องยุ่งยากและยากที่จะสนองตอบได้หมด"

สุชาย ธนวเสถียร (2536) ได้กล่าวถึงแหล่งบริการภายนอก (Outsourcing) ไว้ว่า "งานทางคอมพิวเตอร์เป็นงานที่ยาก ด้วยเหตุผลที่ว่า ต้องดึงข้อมูลดึกดำบรรพ์มาใช้ ในองค์กร IS ในปัจจุบันที่มีเครื่องที่ใช้ตั้งแต่ 20 ปีที่แล้วมาจนถึงปัจจุบัน เครื่องสมัยใหม่ก็ค่อย ๆ เข้ามา พอพูดถึงเรื่องการดึงข้อมูลมาใช้ เราขาดเจ้าหน้าที่ที่มีความชำนาญในการ access data ทุก ๆ ระบบงาน ทำให้ต้องใช้เวลามาก ข้อมูลที่ได้อาจไม่แน่นอนในเรื่อง data integrity คือว่าไม่สามารถระบุได้แน่นอนในเรื่องของความถูกต้องเมื่อได้ข้อมูลมาแล้วยังนำมาใช้กับระบบรวม (integrated system) สมัยใหม่ได้ยาก"

ทวีศักดิ์ กออนันตกูล และคณะ (2536) ได้กล่าวถึงคอมพิวเตอร์กับภาษาไทยไว้ว่า "เนื่องจากคอมพิวเตอร์ที่ใช้ในเมืองไทยต้องใช้ภาษาไทยซึ่งแตกต่างไปจากภาษาอังกฤษค่อนข้างมากดังนั้นการพัฒนาคอมพิวเตอร์ให้ใช้งานได้กับภาษาไทย จึงเป็นกระบวนการที่ต้องเกิดขึ้นโดยธรรมชาติ ไม่ว่าจะง่ายหรือยากเพียงใด หากจะเทียบกับภาษาอื่นก็จะพบว่า บางภาษาเช่นภาษามาเลย์ ซึ่งสูญเสียลักษณะตัวอักษรดั้งเดิมไปจนหมดสิ้น ประเทศเขาใช้ตัวอักษรภาษาอังกฤษแสดงข้อความภาษามาเลย์ และสามารถนำคอมพิวเตอร์มาใช้งานได้ทันที โดยไม่ต้องดัดแปลง แต่สำหรับภาษาจีน ญี่ปุ่น ไทย และอีกหลาย ๆ ประเทศที่ยังคงรักษาวัฒน-ธรรม

ของตนเองให้คงอยู่ต่อไป จำเป็นต้องมีการพัฒนาเครื่องคอมพิวเตอร์ให้ทำการประมวลผล รับข้อมูล และแสดงผลเป็นภาษานั้น ๆ ได้อย่างถูกต้อง"

จากแนวคิดของผู้เขียนบทความ และผลจากการสำรวจความต้องการของ ฝ่ายวางแผนและปฏิบัติ-การคอมพิวเตอร์, ฝ่ายตรวจสอบและวิเคราะห์ธนาคารพาณิชย์, ฝ่ายตรวจสอบและวิเคราะห์บริษัทเงินทุนและบริษัทเครดิตฟองซิเอร์, ฝ่ายตรวจสอบกิจการภายใน ของธนาคารแห่งประเทศไทย ทำให้เกิดความคิดในการพัฒนาเครื่องมือทางซอฟต์แวร์ (Software Tools) ขึ้นมาเพื่อช่วยขจัดปัญหาต่าง ๆ ที่ได้กล่าวไปแล้วในส่วนของความสำคัญและความเป็นมาของปัญหา ซึ่งซอฟต์แวร์ตัวนี้จะมีความสามารถต่าง ๆ ตามที่ระบุไว้ในวัตถุประสงค์และขอบเขตของการวิจัย

2.2 ทฤษฎีที่เกี่ยวข้อง

2.2.1 ทฤษฎีการชดเชยสระ

จากการศึกษาตัวอย่างแฟ้มข้อมูลรายงานที่ได้มาจากโปรแกรมโลตัส 1-2-3 โดยการใช้ คำสั่ง /Print File แล้วใส่ชื่อแฟ้มข้อมูลที่ต้องการ หลังจากนั้นจะได้แฟ้มข้อมูลรายงาน ดังรูปที่ 1.4 และรูปที่ 2.1 เกิดจากการถ่ายข้อมูล (dump) ข้อมูลผ่านโปรแกรมนอร์ตันยูทิลิตี้ (Norton Utility) จะเห็นได้ว่าการจัดระดับของข้อมูลภาษาไทยเป็นแบบ 1 ระดับ คือสระ,พยัญชนะและวรรณยุกต์ เรียงกันอยู่ในบรรทัดเดียวกัน

```
ษ sic.dat ออออออออออออออออออออออออออออออออออออออออออออออออออออออออออออออ Hex format อป
บ Cluster 1,602, Sector 1,633                                    File offset 0, hex 0 บ
บ20202020 2020BAC3 D4C9D1B7 20434326 4320A8D3 A1D1B420        บร.ษ.ท CC&C จำก.ด บ
บ20202020 20202020 20C3D2C2 A1D2C3CD D8BBA1C3 B3ECA1D2           รายการอ ปกรณ.กาบ
บC3CAD7E8 CDCAD2C3 20202020 20202020 202020CB B9E9D2B7 รส..อสาร              หน.าทบ
บD5E82031 0D0A2020 20202020 20202020 20202020 .. 1
                 บ
บ20202020 20202020 20202020 96969696 96969696 96969696                          บ
บ96969696 9696960D 0A202020 20A02020 20202020 20202020 ──────
              บ
บ20202020 20202020 20202020 20202020 28C7D1B9 B7D5E820                  (ว.นท.. บ
บ3120C12E A42E2032 35333829 0D0A2020 20202020 98969696 1 ม.ค. 2538)
     ┌──บ
บ96969696 96969D96 96969696 96969696 96969696 9D969696 ──────────────────────บ
บ96969696 96969696 96969696 96969696 96969696 96969696 ──────────────────────บ
บ9D969696 96969696 9696990D 0A202020 20202095 202020C5 ┌──────────┐
      |   ลบ
บEDD2B4D1 BA202095 20202020 20C3CBD1 CA202020 20209520 .าด.บ  |     รห.ส     | บ
บ20202020 20202020 C3D2C2C5 D0E0CDD5 C2B42020 20202020          รายละเอ.ยด      บ
บ20202095 202020C3 D2A4D220 20950D0A 20202020 20209C96    |   ราคา   |
```

รูปที่ 2.1 แสดงรายละเอียดแฟ้มข้อมูลรายงานที่ได้มาจากโปรแกรมโลตัส 1-2-3 โดยใช้โปรแกรมนอร์ตันยูทิลิตี้

จากรูปที่ 1.4 จะเห็นได้ว่าการที่ข้อมูลตัวเลขที่อยู่หลังข้อมูลภาษาไทยมีลักษณะของการเยื้องไปมา เนื่องมาจากมีการชดเชยสระของข้อมูลภาษาไทยที่อยู่ข้างหน้า ตัวอย่างเช่น ตัวเลข 41000 มีการเยื้องไปจากตำแหน่งที่ควรจะเป็น 5 ตัวอักษร ซึ่งเท่ากับผลบวกของจำนวนสระและวรรณยุกต์ของข้อมูลภาษาไทยที่อยู่หน้าตัวเลข ในที่นี้คือคำว่า "โทรศัพท์ระบบรังผึ้ง" ซึ่งมีจำนวนสระและวรรณยุกต์รวมกันแล้วเท่ากับ 5 ตัว ดังนั้นจึงเสนอวิธีการแก้ปัญหาตามขั้นตอนดังต่อไปนี้

ขั้นตอนที่ 1 ตรวจสอบข้อมูลภาษาไทยทั้งหมดว่า มีข้อมูลกลุ่มไหนที่มีจำนวนสระและวรรณยุกต์มากที่สุดแล้วจำตัวเลขนั้นไว้ ในกรณีนี้คำว่า "เครื่องดักฟังความถี่สูง" มีจำนวนสระและวรรณยุกต์รวมกันได้เท่ากับ 7 ตัว

ขั้นตอนที่ 2 ทำการลบช่องว่างข้างหลังข้อมูลภาษาไทยที่เกิดเนื่องจากการชดเชยสระ โดยจำนวนช่องว่างที่ลบเท่ากับจำนวนสระและวรรณยุกต์ของข้อมูลก่อนหน้า ซึ่งสามารถตรวจสอบได้ 2 รูปแบบคือ ในกรณีที่แฟ้มข้อมูลมีกรอบตาราง ดังรูปที่ 1.4 โปรแกรมจะยึดเส้นแนวตั้งของกรอบตารางเป็นหลัก คืออ่านทีละตัวอักษรจนกระทั่งเจอเส้นแนวตั้ง แล้วจึงทำการลบช่องว่างที่อยู่ก่อนหน้าเส้นแนวตั้งเท่ากับจำนวนสระและวรรณยุกต์ที่คำนวณไว้ก่อนแล้ว ในกรณีที่สองคือกรณีที่แฟ้มข้อมูลไม่มีกรอบตาราง ดังรูปที่ 1.5 โปรแกรมจะตรวจสอบว่าถ้ามีจำนวนช่องว่างคั่นระหว่างคอลัมน์ มากกว่าตัวเลขที่กำหนดโดยผู้ใช้ จะถือว่าใช้ช่องว่างจำนวนนั้นเป็นตัวแบ่งระหว่างคอลัมน์ เมื่อโปรแกรมอ่านเจอช่องว่างมากกว่าตัวเลขที่กำหนดแล้วอ่านเจอตัวอักษรที่ตามมาจะถือว่าเป็นเริ่มคอลัมน์ใหม่ จึงทำการลบช่องว่างที่อยู่ก่อนหน้าเท่ากับจำนวนสระและวรรณยุกต์ที่คำนวณไว้ก่อนแล้ว

ขั้นตอนที่ 3 เนื่องจากการลบช่องว่างเท่ากับจำนวนสระและวรรณยุกต์ ในบางกรณีอาจทำให้ข้อมูลบางส่วนถูกลบไปด้วย จากรูปที่ 1.4 จะเห็นได้ว่าข้อมูล "โปรแกรมเอเบิล คอนโทรลริวิเลอร์" ถ้าทำการลบช่องว่างที่อยู่ก่อนหน้าเส้นแนวตั้งไปเท่ากับจำนวนสระและวรรณยุกต์ ในที่นี้คือ 4 ตัว จะทำให้ข้อมูลตัวเองถูกลบไปด้วย จึงทำให้ต้องมีการคำนวณจำนวนสระและวรรณยุกต์ที่มากที่สุดเก็บไว้ก่อน หลังจากลบจำนวนช่องว่างออกไปแล้ว จะต้องบวกจำนวนช่องว่างกลับเข้าไปใหม่เท่ากับตัวเลขที่คำนวณไว้แล้วตามขั้นตอนที่ 1 ดังนั้นแฟ้มข้อมูลรับเข้าและแฟ้มข้อมูลผลลัพธ์อาจจะมีขนาดของเรคอร์ดไม่เท่ากัน เนื่องจากต้องมีการขยายขนาดของเรคอร์ดออกไปตามเหตุผลที่ได้กล่าวไปแล้วข้างต้น

2.2.2 ทฤษฎีการจัดระดับภาษาไทย

จากการศึกษาตัวอย่างแฟ้มข้อมูลรายงานที่ได้มาจากเครื่องคอมพิวเตอร์ขนาดใหญ่ ซึ่งอาจเกิดจากการสั่งพิมพ์รายงานผ่านโปรแกรมประยุกต์ (Application Program) แล้วถ่ายแฟ้มข้อมูลเหล่านี้จากระบบการเก็บพัก (spooling) มายังเครื่องคอมพิวเตอร์ขนาดเล็ก ตัวอย่างเช่น รูปที่ 1.6, 1.7 และรูปที่ 2.2 เกิดจากการถ่ายข้อมูล (dump) ข้อมูลผ่านโปรแกรมนอร์ตันยูทิลิตี้ (Norton Utility) จะเห็นได้ว่าการจัด

ภาษาไทย ถูกแบ่งออกเป็น 4 ระดับ คือระดับวรรณยุกต์บรรทัดที่ 1, ระดับสระบรรทัดที่ 2, ระดับพยัญชนะบรรทัดที่ 3 และระดับสระล่างบรรทัดที่ 4

```
ษ amdahl.dat ออออออออออออออออออออออออออออออออออออออออออออออออออออออออออ Hex format อป
บ Cluster 25,085, Sectors 201,099-201,1                    File offset 0, hex 0 บ
บ31202020 20202020 20202020 20202020 20202020 20202020 1                        บ
บ20202020 20202020 20202020 20202020 20202020 2020E820                          บ
บ20202020 20202020 20202020 2020E920 20202020 20202020                          บ
บ20202020 20202020 20202020 20202020 20202020 20202020                          บ
บ20202020 20202020 20E920E8 20202020 20202020 20202020                          บ
บ20202020 20202020 20202020 200D0A20 20202020 20202020
        บ
บ20202020 20202020 20202020 20202020 20202020 20202020                          บ
บ20202020 20202020 D120D5D7 20202020 D1202020 20D42020                          บ
บD7202020 20202020 20202020 2020D420 20202020 20202020                          บ
บ20202020 20202020 20202020 20202020 20202020 20202020                          บ
บ2020D520 20202020 20202020 20202020 20202020 20202020                          บ
บ20202020 0D0A2B20 20202020 20202020 20202020 20202020
+               บ
บ20202020 20202020 20202020 20202020 20202020 202020BA                        บบ
บADAAB6CD A8D2C2CD B5C3D2E0 A7B9E0B4 CDB9C5A1 A8D2A7BB ญชถอจายยอดราเงนเดอนลกจางปบ
บC3D0A8D3 E0A7B9A7 BABBC3D0 C1D2B320 20202020 20202020 ระจำเงนงบประมาณ          บ
บ20202020 20202020 20202020 2020CBB9 D2B72020 20312020                หนาท   1  บ
บ20202020 20202020 20202020 20202020 2020200D 0A202020
  บ
บ20202020 20202020 20202020 20202020 20202020 20202020                          บ
บ20202020 20202020 20202020 20202020 20202020 20202020                          บ
บ20202020 20202020 20D92020 20202020 20202020 20202020                          บ
บ20202020 20202020 20202020 20202020 20202020 20202020                          บ
บ20202020 20202020          Press Enter to continue                              บ
1Help   2Hex    3Text   4Dir    5FAT    6Partn  7       8Choose 9Undo   10QuitNU
```

รูปที่ 2.2 แสดงรายละเอียดแฟ้มข้อมูลรายงานที่ได้จากเครื่องคอมพิวเตอร์ขนาดใหญ่โดยใช้โปรแกรมนอร์ตันยูทิลิตี้

จากตัวอย่างแฟ้มข้อมูลข้างต้น จะเห็นได้ว่าข้อมูลที่ได้มาอยู่ในรูปแบบที่ไม่สามารถนำไปใช้งานต่อด้วยโปรแกรมประเภทต่าง ๆ ได้ ดังนั้นจึงเสนอวิธีการแก้ปัญหาตามขั้นตอนดังต่อไปนี้

ขั้นตอนที่ 1 คำนวณขนาดของหน้ารายงาน 1 หน้ามีขนาดกี่บรรทัด โดยใช้รหัสขึ้นหน้าใหม่ที่ติดมากับแฟ้มข้อมูลเป็นตัวตรวจสอบ ในกรณีนี้จะพบว่าตัวเลข 1 ที่คอลัมน์ที่ 1 แถวที่ 1 คือรหัสการขึ้นหน้าใหม่

ขั้นตอนที่ 2 เนื่องจากว่าแฟ้มข้อมูลรายงานที่ได้มา ไม่ได้มีขนาด 4 บรรทัดต่อข้อมูลภาษาไทย 1 ชุด เสมอไป ตัวอย่างเช่น รูปที่ 1.6 จะเห็นได้ว่าข้อมูลมีขนาด 4 บรรทัดทุกชุด ถึงแม้ว่าข้อมูลบางชุดจะไม่มีสระในระดับบรรทัดที่ 2 แต่จะมีบรรทัดว่าง ๆ 1 บรรทัด ส่วนรูปที่ 1.7 จะเห็นได้ว่าข้อมูลบางชุดมีขนาด 3 บรรทัด เนื่องจากไม่มีวรรณยุกต์ในระดับบรรทัดที่ 1 ดังนั้นจะต้องมีการจัดการข้อมูลให้อยู่ในรูปแบบ 4 ระดับ โดยเพิ่มบรรทัดว่าง ๆ เข้าไป เพื่อทำให้ข้อมูลมี 4 ระดับ ต่อ 1 บรรทัด โดยถือหลักว่าบรรทัดที่มีเครื่องหมายบวก(+) ที่คอลัมน์ที่ 1 เป็นบรรทัดที่อยู่ระดับเดียวกับพยัญชนะ

ขั้นตอนที่ 3 หลังจากจัดการข้อมูลให้อยู่ในรูปแบบ 4 ระดับเรียบร้อยแล้ว จึงทำการอ่านข้อมูลภาษาไทยทีละ 4 บรรทัดมาใส่ในแถวลำดับ (array) และทำการแปลงข้อมูลในแถวลำดับที่เก็บแบบ 4 ระดับให้เหลือเพียง 1 ระดับ

2.2.3 ทฤษฎีเกี่ยวกับประเภทของข้อมูลในแฟ้มข้อมูลต่าง ๆ

2.2.3.1 ประเภทของข้อมูลในแฟ้มข้อมูลดีเบส แบ่งออกเป็น 6 ประเภทหลัก ๆ ดังตารางที่ 2.1

ประเภทของข้อมูล	รูปแบบและตัวอย่าง
C ตัวอักษร (Character)	ตัวอักษรรหัสแอสกีทุกตัว
D วันที่ (Date)	ตัวเลขและตัวอักษรที่ใช้ในการแบ่งระหว่างเดือน,วัน,ปี มีรูปแบบ YYYYMMDD ใช้เนื้อที่ในการเก็บ 8 ตัว
F เลขจำนวนจริง (Floating point binary numeric)	ตัวเลขรับค่าต่อไปนี้ - . 0 1 2 3 4 5 6 7 8 9
L ตรรก (Logical)	ตรรกะรับค่าต่อไปนี้ ? Y y N n T t F f
M บันทึก (Memo)	ตัวอักษรรหัสแอสกีทุกตัวเก็บที่แฟ้มข้อมูล .dbt
N เลขฐานสิบเข้ารหัสฐานสอง (Binary coded decimal numeric)	ตัวเลขรับค่าต่อไปนี้ - . 0 1 2 3 4 5 6 7 8 9

ตารางที่ 2.1 ตารางแสดงประเภทของข้อมูลในแฟ้มข้อมูลดีเบส

2.2.3.2 ประเภทของข้อมูลในแฟ้มข้อมูลโลตัส แบ่งออกเป็น 6 ประเภทหลัก ๆ ดังตารางที่ 2.2

ประเภทของข้อมูล	รูปแบบและตัวอย่าง
ข้อความ (Label)	แสดงตัวอักษรชิดซ้าย,ชิดขวาหรือกึ่งกลางเซลล์
ตัวเลขกำหนดทศนิยม (Fixed)	แสดงตัวเลขกำหนดตำแหน่งทศนิยมได้ตั้งแต่ 0-15 เช่น 1.35
ตัวเลขทางวิทยาศาสตร์ (Scientific)	แสดงตัวเลขในรูปแบบของเอ็กซ์โปเนนเชียล เช่น 1.35E+01
ตัวเลขแบบมีเครื่องหมาย (Currency)	แสดงตัวเลขทางการเงิน โดยมีเครื่องหมาย $ และเครื่องหมาย , เช่น $1,352.25
ตัวเลขในรูปเปอร์เซ็นต์ (Percent)	แสดงตัวเลขในรูปของเปอร์เซ็นต์ โดยคูณค่าในเซลล์ด้วย 100 แล้วใส่เครื่องหมาย % ต่อท้าย เช่น 15.25%
ตัวเลขในรูปวันที่ (Date)	แสดงตัวเลขในรูปของวันที่มี 3 รูปแบบคือ 1. วันที่-เดือน-ปี เช่น 15-Jan-1986 2. วันที่-เดือน เช่น 15-Jan 3. เดือน-ปี เช่น Jan-1986

ตารางที่ 2.2 ตารางแสดงประเภทของข้อมูลในแฟ้มข้อมูลโลตัส 1-2-3

2.2.3.3 ประเภทของข้อมูลในแฟ้มข้อมูลซียูไรเตอร์ เนื่องจากแฟ้มข้อมูลที่เกิดจากโปรแกรมซียูไรเตอร์มีลักษณะการเก็บเป็นแฟ้มข้อมูลข้อความ (Text File) อยู่แล้ว จึงสามารถนำมาใช้งานได้ แต่อาจมีปัญหาในเรื่องของรหัสการพิมพ์ โดยการกำหนดรูปแบบตัวอักษรแบบต่าง ๆ ได้ดังตารางที่ 2.3

ตัวเอียง	คีย์คำสั่ง ^PW หรือ ALT-F2
ขีดเส้นใต้ 1 เส้น	คีย์คำสั่ง ^PS หรือ ALT-F3
ขีดเส้นใต้ 2 เส้น	คีย์คำสั่ง ^PR หรือ ALT-F4
ตัวหนา	คีย์คำสั่ง ^PB หรือ ALT-F5
ตัวใหญ่	คีย์คำสั่ง ^PE หรือ ALT-F6
ตัวยกขึ้น	คีย์คำสั่ง ^PT หรือ ALT-F7
ตัวห้อย	คีย์คำสั่ง ^PV หรือ ALT-F8

ตารางที่ 2.3 แสดงรูปแบบตัวอักษรที่ได้จากโปรแกรมซียูไรเตอร์

นอกจากนั้นอาจเจอปัญหาในเรื่องคำสั่งจุด (Dot Command) หมายถึงจุดที่ทำหน้าที่เป็นคำสั่ง เช่น .CW หมายถึงการกำหนดตัวอักษรให้เป็น 15 ตัว/นิ้ว โดยจุดที่จะเป็นคำสั่งได้นั้นจะต้องอยู่ที่ตำแหน่งแรกของบรรทัด ถ้าไม่อยู่ตำแหน่งแรก โปรแกรมจะถือว่าเป็นข้อความ

2.2.3.4 ประเภทของข้อมูลในแฟ้มข้อมูลเอ็กเซล แบ่งออกเป็น 6 ประเภทหลัก ๆ ดังตารางที่ 2.4

ประเภทของข้อมูล	รูปแบบและตัวอย่าง
ตัวอักษร (Text)	"a", "word", "w/punc", "", " " ค่าของตัวอักษรบรรจุได้ตั้งแต่ 0 ถึง 255
ตัวเลข (Numbers)	5.003, 0, 150.286, -30 มีความถูกต้องถึง 15 หลัก
ตรรก (Logical Values)	มีได้เพียง 2 ค่าคือจริง หรือ เท็จ
แถวลำดับ (Arrays)	เช่น {1,2,3,4,5,6}
ค่าผิดพลาด (Error Values)	เช่น #NUM!, #N/A, #DIV/0!
ค่าอ้างอิง (References)	A10, $A10, A10, A$10, R1C1, R[10]C[-10]

ตารางที่ 2.4 ตารางแสดงประเภทของข้อมูลในแฟ้มข้อมูลเอ็กเซล

2.2.3.5 ประเภทของข้อมูลในแฟ้มข้อมูลแอ็คเซส มีรายละเอียดดังตารางที่ 2.5

ประเภทของข้อมูล	รูปแบบและตัวอย่าง
ตัวอักษรปนตัวเลข (Text)	เก็บข้อมูลตัวอักษรและตัวเลขปนกัน (Alphanumeric) สามารถเก็บได้ถึง 255 ไบต์
ข้อมูลบันทึก (Memo)	เก็บข้อมูลตัวอักษรและตัวเลขปนกัน (Alphanumeric) ส่วนมากจะเก็บในรูปประโยคหรือพารากราฟ สามารถเก็บได้ถึง 32,000 ไบต์
ตัวเลข (Number)	เก็บข้อมูลตัวเลขจำนวนเต็มหรือเศษส่วน โดยใช้เนื้อที่ในการเก็บ 1,2,4 หรือ 8 ไบต์
วันที่และเวลา (Date/Time)	เก็บข้อมูลวันที่และเวลา โดยใช้เนื้อที่ในการเก็บ 8 ไบต์

ตารางที่ 2.5 ตารางแสดงประเภทของข้อมูลในแฟ้มข้อมูลแอ็คเซส

ประเภทของข้อมูล	รูปแบบและตัวอย่าง
ตัวเลขแบบมีเครื่องหมาย (Currency)	เก็บข้อมูลตัวเลขทางการเงิน โดยใช้เนื้อที่ในการเก็บ 8 ไบต์
ตัวนับ (Counter)	เก็บข้อมูลตัวเลขที่โปรแกรมจะทำการบวกเพิ่มให้ทีละ 1 โดยอัตโนมัติ ใช้เนื้อที่ในการเก็บ 4 ไบต์
ตรรกะ (Yes/No)	เก็บค่าบูลลีน (Boolean) ใช้เนื้อที่ในการเก็บ 1 บิต
โอแอลอี อ็อบเจ็ค (OLE Object)	เก็บข้อมูลกราฟฟิก (graphics) หรือข้อมูลไบนารี โดยสามารถเก็บได้ถึง 1 กิกาไบต์ (gigabyte)

ตารางที่ 2.5 ตารางแสดงประเภทของข้อมูลในแฟ้มข้อมูลแอ็คเซส (ต่อ)

2.2.4 ทฤษฎีเกี่ยวกับประเภทของข้อมูลในภาษาต่าง ๆ

2.2.4.1 ประเภทของข้อมูลในภาษาซี ที่ได้จากโปรแกรมเทอร์โบซี มีรายละเอียดดังตารางที่ 2.6

ชนิดตัวแปร	ชนิดของข้อมูล	ช่วงค่าของตัวแปร	จำนวนไบต์
char	ตัวอักษร (character)	0 to 255	1
int	เลขจำนวนเต็ม (integer)	-32768 to 32767	2
short int	เลขจำนวนเต็มแบบสั้น (short integer)	-32768 to 32767	2
long int	เลขจำนวนเต็มแบบยาว (long integer)	-4294967296 to 4294967295	4
unsigned int	เลขจำนวนเต็มแบบไม่คิดเครื่องหมาย (unsigned integer)	0 to 65535	2
float	เลขจำนวนจริง (floating point)	ความแม่นยำประมาณเลข 6 หลัก	4
double	เลขจำนวนจริงที่มีความแม่นยำเป็นสองเท่า (double-precision floating point)	ความแม่นยำประมาณเลข 12 หลัก	8

ตารางที่ 2.6 แสดงชนิดของตัวแปรในภาษาซี

2.2.4.2 ประเภทของข้อมูลในภาษาโคบอล ที่ได้จากโปรแกรมไมโครซอฟต์โคบอล มีรายละเอียดดังตารางที่ 2.7 และ 2.8

ชนิดของสัญลักษณ์	สัญลักษณ์	การใช้งาน
สัญลักษณ์แทนข้อมูล	9	ข้อมูลตัวเลข (Numeric field)
	A	ข้อมูลตัวอักษร (Alphabetic field)
	X	ข้อมูลตัวเลข,ตัวอักษรและเครื่องหมายปนกัน (Alphanumeric field)
สัญลักษณ์ที่ใช้ในการคำนวณ	V	แสดงตำแหน่งของจุดทศนิยม (Assumed decimal point)
	P	แสดงจำนวนทศนิยม (Decimal scaling)
	S	แสดงเครื่องหมาย (Operational sign included)
สัญลักษณ์ที่ใช้ในการพิมพ์	$	เครื่องหมายดอลลาร์ (Dollar sign)
	Z	เปลี่ยนเลขศูนย์หน้าเป็นช่องว่าง (Zero suppression)
	*	เครื่องหมายดอกจันทร์ (Check protection)
	.	เครื่องหมายจุดทศนิยม (Decimal point)
	,	เครื่องหมายคอมม่า (Comma)
	+	เครื่องหมายบวก (Plus sign)
	-	เครื่องหมายลบ (Minus sign)
	DB	เครื่องหมายเดบิต (Debit)
	CR	เครื่องหมายเครดิต (Credit)
	B	เครื่องหมายช่องว่างแทรกอยู่ (Blank insertion)
	0	ตัวเลขศูนย์แทรกอยู่ (Zero insertion)

ตารางที่ 2.7 แสดงชนิดของสัญลักษณ์ที่สามารถใช้ในพิคเจอร์คอส (PICTURE clauses)

ชนิดตัวแปร	คำอธิบาย
ข้อมูลตัวเลข	พิคเจอร์คอสอาจจะประกอบไปด้วยตัวอักษรต่อไปนี้ 9, V, P, S.
ข้อมูลตัวอักษร	พิคเจอร์คอสประกอบด้วยตัวอักษร A เท่านั้น
ข้อมูลตัวเลขปนตัวอักษร	พิคเจอร์คอสประกอบด้วยตัวอักษร A, 9, และ X แต่ไม่สามารถประกอบด้วย A ทั้งหมด หรือ 9 ทั้งหมด จะต้องประกอบด้วย A และ 9 ปนกัน

ตารางที่ 2.8 แสดงชนิดของตัวแปรในภาษาโคบอล

ชนิดตัวแปร	คำอธิบาย
การตรวจสอบข้อมูลตัวเลข	พิคเจอร์คอสสามารถประกอบด้วยตัวอักษรต่อไปนี้ B P V Z 0 9 , . * + - CR DB $
การตรวจสอบข้อมูลตัวอักษรปนตัวเลข	พิคเจอร์คอสสามารถประกอบด้วยตัวอักษรต่อไปนี้ A X 9 B 0

ตารางที่ 2.8 แสดงชนิดของตัวแปรในภาษาโคบอล (ต่อ)

2.2.4.3 ประเภทของข้อมูลในภาษาปาสกาล ที่ได้จากโปรแกรมเทอร์โบปาสกาล มี 5 ประเภทหลัก ๆ ดังนี้ - ตัวเลขจำนวนเต็ม (Integers) แบ่งออกเป็นชนิดต่าง ๆ ตามตารางที่ 2.9

ชนิดตัวแปร	ช่วงค่าของตัวแปร	จำนวนไบต์
byte	0 to 225	1
shortint	-128 to 127	1
integer	-32768 to 32767	2
word	0 to 65535	2
longint	-2147483648 to 2147483647	4

ตารางที่ 2.9 แสดงชนิดของข้อมูลตัวเลขจำนวนเต็มในภาษาปาสกาล

- ตัวเลขจำนวนจริง (Real Numbers) แบ่งออกเป็น 2 ส่วนคือเก็บในรูปเศษส่วน (Fractional) เช่น 3.14159 และเก็บในรูปยกกำลัง (Exponents) เช่น 2.579×10^{24} หรือที่รู้จักกันในชื่อเลขจำนวนจริง (floating-point numbers) แบ่งออกเป็นชนิดต่าง ๆ ตามตารางที่ 2.10

ชนิดตัวแปร	ช่วงค่าของตัวแปร	จำนวนไบต์
real	2.9x10E-39 to 1.7x10E38	6
single	1.5x10E-45 to 3.4x10E38	4
double	5.0x10E-324 to 1.7x10E308	8
extended	1.9x10E-4951 to 1.1x10E4932	10
comp*	-2E+63+1 to 2E+63-1	8

comp* ใช้กับตัวเลขจำนวนเต็มเท่านั้น

ตารางที่ 2.10 แสดงชนิดของข้อมูลตัวเลขจำนวนจริงในภาษาปาสกาล

- ตัวอักษรใด ๆ (Characters) อาจจะเป็นตัวเลข, ตัวอักษร หรือ สัญญลักษณ์ เช่น a, Z, !, 3 หรือประกอบกันเป็นชุดของตัวอักษร เช่น 'This is only a test.'
- ตรรกะแบบบูล (Boolean) ใช้สำหรับค่าที่เป็นไปได้ 2 ค่าคือจริง หรือ เท็จ
- พอยเตอร์ (Pointers) ใช้เก็บค่าตำแหน่งของที่อยู่ในหน่วยความจำ

2.2.4.4 ประเภทของข้อมูลในภาษาเบสิก ที่ได้จากโปรแกรมไมโครซอฟต์จีดับบลิวเบสิก แบ่งออกเป็น 2 ประเภทหลัก ๆ ดังนี้

- ตัวแปรสตริง (String Variable) ใช้ในการเก็บค่าที่เป็นตัวอักษรหรือตัวเลขที่ไม่ใช้ในการคำนวณเช่น รหัสประจำตัว โดยเขียนเครื่องหมาย $ ต่อท้ายชื่อตัวแปร เช่น A$ = "ABC"
- ตัวแปรตัวเลข (Numeric Variable) ใช้ในการเก็บค่าที่เป็นตัวเลขที่ใช้ในการคำนวณ โดยการกำหนดเครื่องหมายต่อท้ายชื่อตัวแปรได้ 3 แบบคือ

% - สำหรับตัวเลขจำนวนเต็ม (Integer Variable) ใช้เนื้อที่ 2 ไบท์ เช่น AMOUNT%

! - สำหรับเลขจำนวนจริงที่มีความแม่นยำหนึ่งเท่า (Single Precision) ใช้เนื้อที่ 4 ไบท์ เช่น MINIMUM!

- สำหรับเลขจำนวนจริงที่มีความแม่นยำเป็นสองเท่า (Double Precision) ใช้เนื้อที่ 8 ไบท์ เช่น PI#

2.2.5 ทฤษฎีเกี่ยวกับระบบจัดการแฟ้มข้อมูลบีทรีฟ

2.2.5.1 โครงสร้างแฟ้มข้อมูลบีทรีฟ

แฟ้มข้อมูลบีทรีฟจะแบ่งออกเป็นเพจโดยขนาดของเพจจะถูกระบุตั้งแต่เริ่มสร้างแฟ้มข้อมูลขนาดของเพจจะเป็นได้ตั้งแต่ 512 ไบต์ไปจนถึง 4096 ไบต์ โดยขนาดของเพจจะต้องหารด้วย 512 ลงตัวเสมอ นอกจากนั้นขนาดของเพจจะต้องใหญ่มากพอที่จะบรรจุข้อมูล 1 เรคอร์ดได้อีกด้วย (เฉพาะส่วนที่มีขนาดคงที่) ข้อมูลที่เก็บในแฟ้มข้อมูลบีทรีฟจะต้องอยู่ในรูปของเรคอร์ดเท่านั้น แต่เรคอร์ดของบีทรีฟนั้นไม่จำเป็นจะต้องมีความยาวเท่ากันเสมอไป สามารถแปรเปลี่ยนความยาวเรคอร์ดไปตามความยาวข้อมูลได้ จะทำให้ประหยัดเนื้อที่ในการเก็บ

เพจในบีทรีฟจะแบ่งออกเป็น 3 แบบด้วยกัน คือ เพจเฮดเดอร์ (Header Page), เพจอินเด็กซ์ (Index Page) และ เพจข้อมูล (Data Page) โดยแฟ้มข้อมูลทั่ว ๆ ไปของบีทรีฟจะต้องมีเพจทั้ง 3 แบบนี้รวมอยู่ด้วยเสมอ (แฟ้มข้อมูลบีทรีฟสามารถมีเฉพาะเพจอินเด็กซ์ หรือ เพจข้อมูลได้ด้วย) ข้อมูลเพจแรกของ บีทรีฟเป็นเพจเฮดเดอร์ซึ่งเป็นเรคอร์ดที่เก็บข้อมูลพารามิเตอร์ทั้งหมดของแฟ้มข้อมูล สำหรับเพจที่ 2 เก็บข้อมูลที่ใช้ในการเรียงข้อมูลในลักษณะอื่น ๆ นอกจากแอสกี เช่นลักษณะการเรียงข้อมูลตามที่กำหนดไว้ในแฟ้มข้อมูล UPPER.ALT ซึ่งเป็นแฟ้มข้อมูลที่ให้มาพร้อมกับชุดพัฒนาโปรแกรมบีทรีฟ เพจชนิดต่อไปคือ เพจ

หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

อินเด็กซ์จำนวนของเพจอินเด็กซ์ที่อยู่ในแฟ้มข้อมูลหนึ่ง ๆ จะมากหรือน้อยแค่ไหน ก็ขึ้นอยู่กับจำนวนอินเด็กซ์และจำนวนเรคอร์ดที่แฟ้มข้อมูลนั้นมีอยู่ โดยทั่วไปอินเด็กซ์แต่ละตัวจะใช้เนื้อที่เริ่มแรก 1 เพจเสมอ และเมื่อไม่พอจึงจะจองเนื้อที่เพิ่มทีละเพจ (บีทรีฟกำหนดว่าจะต้องมีอินเด็กซ์อย่างน้อย 1 คีย์เสมอ) เพจชนิดต่อไปเป็นเพจที่สำคัญที่สุด และมีจำนวนมากที่สุดในแฟ้มข้อมูลคือ เพจข้อมูล ซึ่งมีอยู่ 2 แบบคือ เพจข้อมูลแบบ "ธรรมดา" เพจข้อมูลแบบนี้แต่ละเรคอร์ดจะมีขนาดเท่ากันเสมอ และแบบที่ 2 คือเพจข้อมูลแบบ "แปรเปลี่ยน" (Variable Length) เพจข้อมูลแบบนี้แต่ละเรคอร์ดจะมีความยาวไม่คงที่ อย่างไรก็ตาม เนื่องจากบีทรีฟสามารถสร้างแฟ้มข้อมูลที่มีแต่คีย์หรืออินเด็กซ์อย่างเดียวได้ด้วย ดังนั้นแฟ้มข้อมูลแบบนี้ก็จะไม่มีเพจข้อมูล เพจข้อมูลนับว่าเป็นส่วนที่สำคัญที่สุดในแฟ้มข้อมูล เพราะในเพจข้อมูลแบบอื่น ๆ เช่นเพจเฮดเดอร์ หรือเพจอินเด็กซ์ แม้ว่าจะมีความเสียหายเกิดขึ้น ก็สามารถสร้างขึ้นมาใหม่ได้ แต่ไม่สามารถสร้างเพจข้อมูลขึ้นมาใหม่ได้หากเกิดความเสียหายขึ้น

2.2.5.2 ประเภทของข้อมูลบีทรีฟ มีรายละเอียดดังตารางที่ 2.11

ประเภทของข้อมูล	คำอธิบาย
String	เก็บข้อมูลตัวอักษรโดยเรียงลำดับจากซ้ายไปขวาและตัวอักษรแต่ละตัวแทนรหัสแอสกี 1 ตัวอักษร
Lstring	มีคุณสมบัติเหมือนกับประเภทสตริง ยกเว้นไบต์แรกของสตริงจะเก็บค่าความยาวของสตริงนั้น ๆ
Zstring	มีคุณสมบัติเหมือนกับประเภทสตริง ยกเว้นไบต์สุดท้ายจะเก็บค่า ไบนารี 0 ซึ่งจะเหมือนกับซีสตริง
Numeric	เป็นข้อมูลตัวเลขที่เก็บแบบสตริง แต่จะเก็บชิดขวาและเติม 0 หน้า แต่ละตำแหน่งใช้เนื้อที่ 1 ไบต์ โดยที่ตำแหน่งขวาสุดของตัวเลขจะรวมเครื่องหมายด้วย
Decimal	เป็นข้อมูลตัวเลขที่มีลักษณะการเก็บเช่นเดียวกับตัวเลขทศนิยมอัดแน่น (packed decimal number) โดยเก็บเลขฐานสิบ 2 ตัวใน 1 ไบต์
Money	มีคุณสมบัติเหมือนกับตัวเลขทศนิยม (Decimal type)
Float	เป็นชนิดข้อมูลที่ใช้มาตราฐานเดียวกับไอทริเพิลอี (IEEE) สำหรับ เลขจำนวนจริงที่มีความแม่นยำหนึ่งเท่า และ สองเท่า ใช้เนื้อที่ในการเก็บ 4 ไบต์
Integer	เป็นข้อมูลตัวเลขที่เก็บเครื่องหมาย

ตารางที่ 2.11 แสดงประเภทของข้อมูลบีทรีฟ

ประเภทของข้อมูล	คำอธิบาย
Date	เป็นข้อมูลวันที่ ใช้เนื้อที่ในการเก็บ 4 ไบต์ โดยที่วันที่ใช้เนื้อที่ในการเก็บ 1 ไบต์, เดือนใช้เนื้อที่ในการเก็บ 1 ไบต์, ปีใช้เนื้อที่ในการเก็บ 2 ไบต์
Time	เป็นข้อมูลเวลา ใช้เนื้อที่ในการเก็บ 4 ไบต์ โดยที่หนึ่งส่วนร้อยของวินาทีใช้เนื้อที่ในการเก็บ 1 ไบต์, วินาทีใช้เนื้อที่ในการเก็บ 1 ไบต์, นาทีใช้เนื้อที่ในการเก็บ 1 ไบต์ และ ชั่วโมงใช้เนื้อที่ในการเก็บ 1 ไบต์ ตามลำดับ
Logical	เปรียบได้กับข้อมูลสตริง ใช้ในเนื้อที่ในการเก็บ 1 หรือ 2 ไบต์ โดยขึ้นกับผู้ใช้ตัดสินใจว่าค่าใดเป็น จริง หรือ เท็จ

ตารางที่ 2.11 แสดงประเภทของข้อมูลบีทรีฟ (ต่อ)

2.2.6 <u>ทฤษฎีเกี่ยวกับระบบจัดการฐานข้อมูลออราเคิล</u>

ประเภทของข้อมูลออราเคิล มีรายละเอียดดังตารางที่ 2.12

ประเภทของข้อมูล	คำอธิบาย
Varchar2(size)	เก็บข้อมูลตัวอักษรแบบความยาวแปรได้ (Variable length) มีความยาวมากที่สุด 2000 ตัวอักษร การประกาศตัวแปรจะต้องระบุขนาด
Char(size)	เก็บข้อมูลตัวอักษรแบบความยาวคงที่ (Fixed length) มีความยาวมากที่สุด 255 ตัวอักษร การประกาศตัวแปรจะต้องระบุขนาด
Number(p,s)	เก็บข้อมูลตัวเลขที่มีความแม่นยำ p และมีสเกล s โดยที่ p มีค่าอยู่ในช่วง 1 ถึง 38 และ s มีค่าอยู่ในช่วง -84 ถึง 127
Long	เก็บข้อมูลตัวอักษรแบบความยาวแปรได้ (Variable length) ที่มีความยาวมากได้ถึง 2 กิกาไบต์
Date	เก็บข้อมูลวันที่ซึ่งอยู่ในช่วงวันที่ 1 มกราคม 4712 BC ถึง 31 ธันวาคม 4712 AD
Raw(size)	เก็บข้อมูลไบนารีแบบความยาวคงที่ (Fixed length)
Long Raw	เก็บข้อมูลไบนารีแบบความยาวแปรได้ (Variable length) ได้ถึง 2 กิกาไบต์

ตารางที่ 2.12 แสดงประเภทของข้อมูลออราเคิล