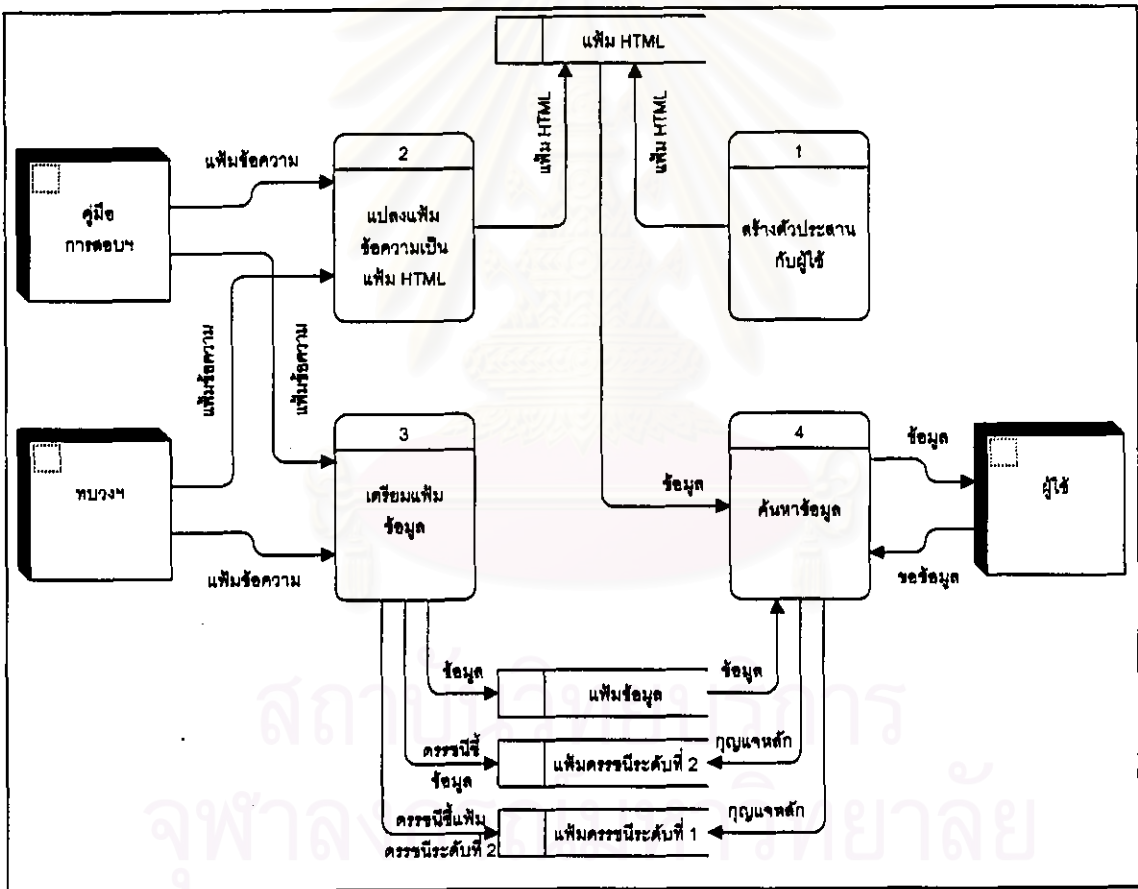




แผนภาพการทำงาน

กระบวนการทำงานเพื่อนำเสนอข้อมูลการสอบคัดเลือกฯ ผ่านเว็บไซต์เว็บสามารถแสดงเป็นแผนภาพได้ดังแสดงในรูป 4.1



รูปที่ 4.1 แผนภาพการทำงานโดยรวม

จากแผนภาพแบ่งออกได้เป็น 2 ส่วนใหญ่ๆ คือ ส่วนที่เป็นกระบวนการทำงานและส่วนที่เป็นข้อมูล

1. กระบวนการทำงาน

1.1 สร้างตัวประสานกับผู้ใช้

ตัวประสานกับผู้ใช้บนเว็บสร้างด้วย HTML ซึ่งเป็นภาษาพื้นฐานที่ช่วยให้ผู้ใช้เข้าถึงข้อมูลที่เก็บอยู่บนเว็บเซิร์ฟเวอร์ได้ ในที่นี้เป็นการสร้างเว็บเพจซึ่งมีตัวเชื่อมโยง (link) เชื่อมข้อมูลต่างๆ ที่เก็บอยู่บนเว็บเซิร์ฟเวอร์เข้าด้วยกัน ผู้ใช้จะทราบว่ามีข้อมูลอะไรบ้างที่สามารถสืบค้นดูได้จากเว็บเพจนี้ นอกจากตัวเชื่อมโยงซึ่งชี้ไปยังตำแหน่งที่เก็บข้อมูลแล้ว ยังมีตัวประสานที่เป็นฟอร์มซึ่งผู้ใช้สามารถป้อนข้อมูลลงไปได้

ตัวอย่างของ HTML ที่ใช้สร้างฟอร์ม

```
<FORM Name="form1" Method="GET" Action="http://www.car.chula.ac.th/cgi-bin/entrance/
  place/exam_plc.exe" onSubmit="return(chkValue(this))" Target="seatOutFrame">
<B>กรุณาป้อนเลขที่นั่งสอบ:</B>
<INPUT Type="text" Name="seat" Size=8 Maxlength=7>
<INPUT Type="submit" Value="OK">
</FORM>
```

จากตัวอย่างนี้จะได้ฟอร์มในรูปที่ 4.2

กรุณาป้อนเลขที่นั่งสอบ:

รูปที่ 4.2 ตัวอย่างฟอร์ม

เมื่อผู้ใช้กดปุ่ม OK ในรูปที่ 4.2 เลขที่นั่งสอบจะถูกตรวจสอบด้วยฟังก์ชัน `chkValue()` ก่อน หากเลขที่นั่งสอบที่ผู้ใช้ป้อนอยู่ในช่วงที่ถูกต้อง ค่านั้นจึงจะถูกส่งไปให้โปรแกรม `exam_plc.exe` ประมวลผลต่อไป

1.2 แปลงเพิ่มข้อความเป็นแท้ม HTML

ข้อมูลที่ได้จากการแปลงเพิ่มข้อความให้เป็นแท้ม HTML นั้นเป็นลักษณะของข้อมูลคงที่ โดยมากใช้รูปแบบของตารางช่วยในการนำเสนอเพราะดูเป็นระเบียบและอ่านง่าย

ตัวอย่างของ HTML ที่ใช้สร้างตาราง

```

<TABLE Cellpadding=5 Border>
  <TR Align=Center>
    <TH>รหัสคณะหรือประเภทวิชา</TH><TH>คณะหรือประเภทวิชา</TH>
    <TH>คะแนนสูงสุด</TH><TH>คะแนนต่ำสุด</TH>
  </TR>
  <TR Align=Center>
    <TH>0001</TH><TD>คณะแพทยศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย</TD>
    <TD>480</TD><TD>366</TD>
  </TR>
</TABLE>

```

จากตัวอย่างนี้จะได้ตารางในรูปที่ 4.3

รหัสคณะหรือประเภทวิชา	คณะหรือประเภทวิชา	คะแนนสูงสุด	คะแนนต่ำสุด
0001	คณะแพทยศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย	480	366

รูปที่ 4.3 ตัวอย่างตาราง

ในกรณีที่แฟ้มซึ่งจะนำมาแปลงเป็น HTML มีข้อมูลค่อนข้างมากและมีรูปแบบของข้อมูลแน่นอน มีลักษณะคล้ายตาราง คือ มีการแบ่งเป็นแถวและสดมภ์ชัดเจนด้วยช่องว่างหรือสัญลักษณ์บางอย่าง การแปลงแฟ้มข้อมูลนี้เป็นแฟ้ม HTML สามารถทำได้ง่ายโดยการใช้ไมโครซอฟต์เอ็กเซลช่วย การแทรกแก้ระหว่างเขตข้อมูลในทุกๆ แถวทำได้โดยการสร้างสดมภ์ใหม่ระหว่างเขตข้อมูลที่ต้องการแทรกแก้แล้วจึงพิมพ์แก้ที่ที่ต้องการแทรกลงไป จากนั้นจึงทำสำเนาของข้อมูลนี้ไปยังสดมภ์เดียวกันของทุกๆ แถว เมื่อเติมแก้ในส่วนของตารางเรียบร้อยแล้ว อาจเพิ่มเติมส่วนอื่นๆ เพื่อให้เป็นแฟ้ม HTML ที่สมบูรณ์ก่อนที่จะจัดเก็บแฟ้มนี้เป็นแฟ้มข้อความที่มีส่วนขยาย (extension) เป็น htm หรือ html (Microsoft Excel Internet Assistant สามารถแปลงตารางให้อยู่ในรูปแบบ HTML ได้ แต่อักษรภาษาไทยจะถูกแปลงให้เป็นรหัส ซึ่งทำให้แฟ้มข้อมูลมีขนาดใหญ่ขึ้น นอกจากนี้หากต้องการเปลี่ยนแปลงแก้ไขข้อมูลในแฟ้ม HTML นี้โดยใช้โปรแกรมอื่นก็ทำได้ยากเนื่องจากอักษรภาษาไทยถูกเปลี่ยนเป็นรหัสที่อ่านไม่ออกแล้ว)

1.3 เตรียมแฟ้มข้อมูล

การเตรียมแฟ้มข้อมูลนี้เป็นการเตรียมและจัดรูปแบบของแฟ้มให้เหมาะสมสำหรับใช้ในการค้นหา ข้อมูลแต่ละประเภทจะมีโปรแกรมเฉพาะสำหรับใช้จัดการกับข้อมูลประเภทนั้นๆ ข้อมูลดังกล่าวได้แก่ ข้อมูลสำหรับ ประกาศสถานที่สอบ ผู้สอบผ่านข้อเขียน คะแนนสูงสุดต่ำสุดและจำนวนผู้สอบได้แยกตามคณะหรือสาขาวิชา ข้อมูลเหล่านี้จัดเก็บเป็นแฟ้มข้อความเรียงลำดับตามกฎแจนหลัก (เลขที่นั่งสอบหรือรหัสคณะ ขึ้นอยู่กับประเภทของข้อมูล) เนื่องจากแฟ้มข้อมูลนี้มีขนาดใหญ่จึงได้สร้างแฟ้มดรรชนีเพื่อให้สามารถค้นหาข้อมูลจากการระบุเลขที่อยู่ (address) ได้ ซึ่งช่วยให้การค้นหาทำได้รวดเร็วขึ้น โปรแกรมสร้างแฟ้มดรรชนีรับแฟ้มข้อมูลที่เรียงลำดับตามกฎแจนหลักแล้วเป็นแฟ้มข้อมูลเข้าและมีแฟ้มดรรชนีเป็นแฟ้มข้อมูลออก ในกรณีที่แฟ้มข้อมูลมีขนาดใหญ่มากๆ เช่น แฟ้มข้อมูลสำหรับประกาศผู้สอบผ่านข้อเขียน จะใช้แฟ้มดรรชนีสองระดับ โดยแฟ้มดรรชนีระดับที่สองเก็บดรรชนีที่ชี้ไปยังตัวข้อมูล ส่วนแฟ้มดรรชนีระดับที่หนึ่งเก็บดรรชนีที่ชี้ไปยังข้อมูลในแฟ้มดรรชนีระดับที่สอง

1.4 ค้นหาข้อมูล

การค้นหาข้อมูลจะค้นด้วยกฎแจนหลักซึ่งเป็นข้อมูลเข้าที่ผู้ใช้ป้อนลงไปในรูปแบบ โดยโปรแกรมที่ทำหน้าที่ค้นหาจะนำกฎแจนหลักไปหาเลขที่อยู่ของข้อมูลจากแฟ้มดรรชนีก่อน แล้วจึงไปดึงข้อมูลจากเลขที่อยู่บนแฟ้มข้อมูล ข้อมูลนี้จะถูกจัดให้อยู่ในรูปแบบ HTML ก่อนที่จะส่งไปให้ผู้ใช้เพื่อแสดงผล ในกรณีที่ไม่มีข้อมูล โปรแกรมจะส่งข้อความแจ้งให้ทราบเช่นกัน

2. ข้อมูลเข้าและข้อมูลออกที่เกี่ยวข้อง

2.1 ข้อมูลจากคู่มือการสอบคัดเลือกฯ

ข้อมูลจากคู่มือการสอบคัดเลือกฯ เป็นรายละเอียดต่างๆ ในการสอบคัดเลือกฯ ที่ผู้สมัครทุกคนควรทราบ การนำเสนอข้อมูลเหล่านี้ผ่านเว็บเป็นอีกทางเลือกหนึ่งให้ผู้สนใจสามารถเลือกดูได้ในขั้นแรกต้องทำการจัดเก็บข้อมูลเหล่านี้ให้อยู่ในรูปแบบอิเล็กทรอนิกส์ก่อนโดยใช้โปรแกรมประมวลผลคำ (word processor) ใดก็ได้ และจัดเก็บเป็นแฟ้มข้อความ (text file) เพื่อให้ง่ายในการทำงานขั้นตอนต่อไป

2.2 ข้อมูลจากทบวงมหาวิทยาลัย

ข้อมูลจากทบวงมหาวิทยาลัยจะมาในรูปแบบอิเล็กทรอนิกส์ ส่วนมากมีการจัดเก็บเป็นแฟ้มข้อความอยู่แล้ว ข้อมูลในส่วนนี้เป็นข้อมูลที่เกิดขึ้นภายหลังการรับสมัครจนกระทั่งการสอบข้อเขียนดำเนินการเสร็จเรียบร้อยแล้ว โดยมากเป็นสถิติต่างๆ เช่น จำนวนผู้สมัครสอบ จำนวนผู้สอบผ่านข้อเขียน

2.3 แฟ้ม HTML

แฟ้ม HTML เป็นแฟ้มข้อความที่มีส่วนขยายเป็น htm หรือ html ภายในเก็บข้อมูลและสารสนเทศที่ต้องการนำเสนอ แท็กที่แทรกอยู่เป็นตัวกำหนดว่าข้อมูลหรือสารสนเทศจะแสดงบนบราวเซอร์ในลักษณะใด

2.4 แฟ้มข้อมูล

แฟ้มข้อมูลเป็นเสมือนฐานข้อมูลซึ่งเก็บข้อมูลทั้งหมดไว้ เช่น แฟ้มสำหรับประกาศสถานที่สอบจะเก็บระเบียบของผู้สมัครสอบในสนามสอบกรุงเทพมหานครทั้งหมดว่าใครสอบวิชาอะไร ที่ไหนบ้าง ส่วนแฟ้มสำหรับประกาศผลผู้สอบผ่านข้อเขียนจะเก็บระเบียบของผู้สมัครสอบทั้งหมดว่าใครสอบผ่านหรือสอบไม่ผ่าน ถ้าสอบผ่านแล้วได้คะแนนอะไร เป็นต้น

2.5 แฟ้มตวรรษในระดับที่สอง

แฟ้มนี้เปรียบเสมือนตู้บัตรรายการในห้องสมุดที่บอกให้ทราบว่าหนังสือที่ต้องการอยู่ตรงส่วนใดในห้องสมุด แฟ้มนี้จะมีขนาดเล็กกว่าแฟ้มข้อมูลมากและมีการจัดรูปแบบให้เหมาะสำหรับการค้นหา ภายในแฟ้มมีเพียงกุญแจหลักและเลขที่อยู่ของข้อมูลบนแฟ้มข้อมูลซึ่งมีกุญแจหลักตรงกัน

2.6 แฟ้มตวรรษในระดับที่หนึ่ง

หากแฟ้มข้อมูลมีขนาดใหญ่มากๆ แฟ้มตวรรษนี้ได้ก็จะมีขนาดใหญ่ตามไปด้วย เพื่อให้การค้นหามีประสิทธิภาพและรวดเร็วมากยิ่งขึ้นจึงใช้ตวรรษที่สองระดับ (two-level index) ช่วย โดยตวรรษในแฟ้มตวรรษระดับที่หนึ่งจะชี้ไปยังข้อมูลในแฟ้มตวรรษระดับที่สอง ตวรรษในแฟ้มนี้ได้มาจากการแบ่งตวรรษในแฟ้มตวรรษระดับที่สองเป็นกลุ่มๆ แล้วตั้งกุญแจหลักและเลขที่อยู่ของตวรรษแรกของแต่ละกลุ่มมาเก็บไว้ในแฟ้มตวรรษระดับที่หนึ่ง

การตรวจสอบความถูกต้องของข้อมูล

ผู้ใช้ส่งคำร้องขอผ่านฟอร์มไปยังเว็บเซิร์ฟเวอร์เพื่อขอข้อมูล แต่ถ้าหากข้อมูลที่ผู้ใช้ป้อนผ่านฟอร์มเป็นข้อมูลที่ไม่ถูกต้อง เว็บเซิร์ฟเวอร์ต้องเสียเวลาประมวลผลโดยไม่ได้ให้คำตอบตามที่ผู้ใช้ต้องการเพื่อลดภาระในส่วนนี้ของเว็บเซิร์ฟเวอร์และให้เว็บเซิร์ฟเวอร์ทำงานอย่างมีประสิทธิภาพยิ่งขึ้น จึงให้มีการตรวจสอบความถูกต้องของข้อมูลก่อนที่จะส่งไปประมวลผลที่เว็บเซิร์ฟเวอร์ จาวาสคริปต์สามารถช่วยในจุดนี้ได้โดยการเพิ่มสคริปต์ที่ทำหน้าที่ตรวจสอบไว้ในแฟ้ม HTML ซึ่งสร้างฟอร์มสำหรับรับข้อมูลเข้า ด้วยวิธีนี้ การตรวจสอบความถูกต้องของข้อมูลขั้นต้นจะทำให้เว็บบราวเซอร์ก่อน จนกระทั่งถึงที่ผู้ใช้ป้อนเข้าไปนั้นถูกต้องจึงจะส่งไปประมวลผลที่เว็บเซิร์ฟเวอร์ต่อไป

ในกรณีศึกษา นี้ ผู้ใช้ต้องป้อนเลขที่นั่งรถเพื่อขอดูสถานที่สอบหรือผลสอบข้อเขียน และป้อนรหัสคณะ/ประเภทวิชาเพื่อขอดูคะแนนสูงสุดต่ำสุดและจำนวนผู้สอบผ่านข้อเขียน

ตัวอย่างจาวาสคริปต์ที่ใช้ตรวจสอบเลขที่นั่งรถ

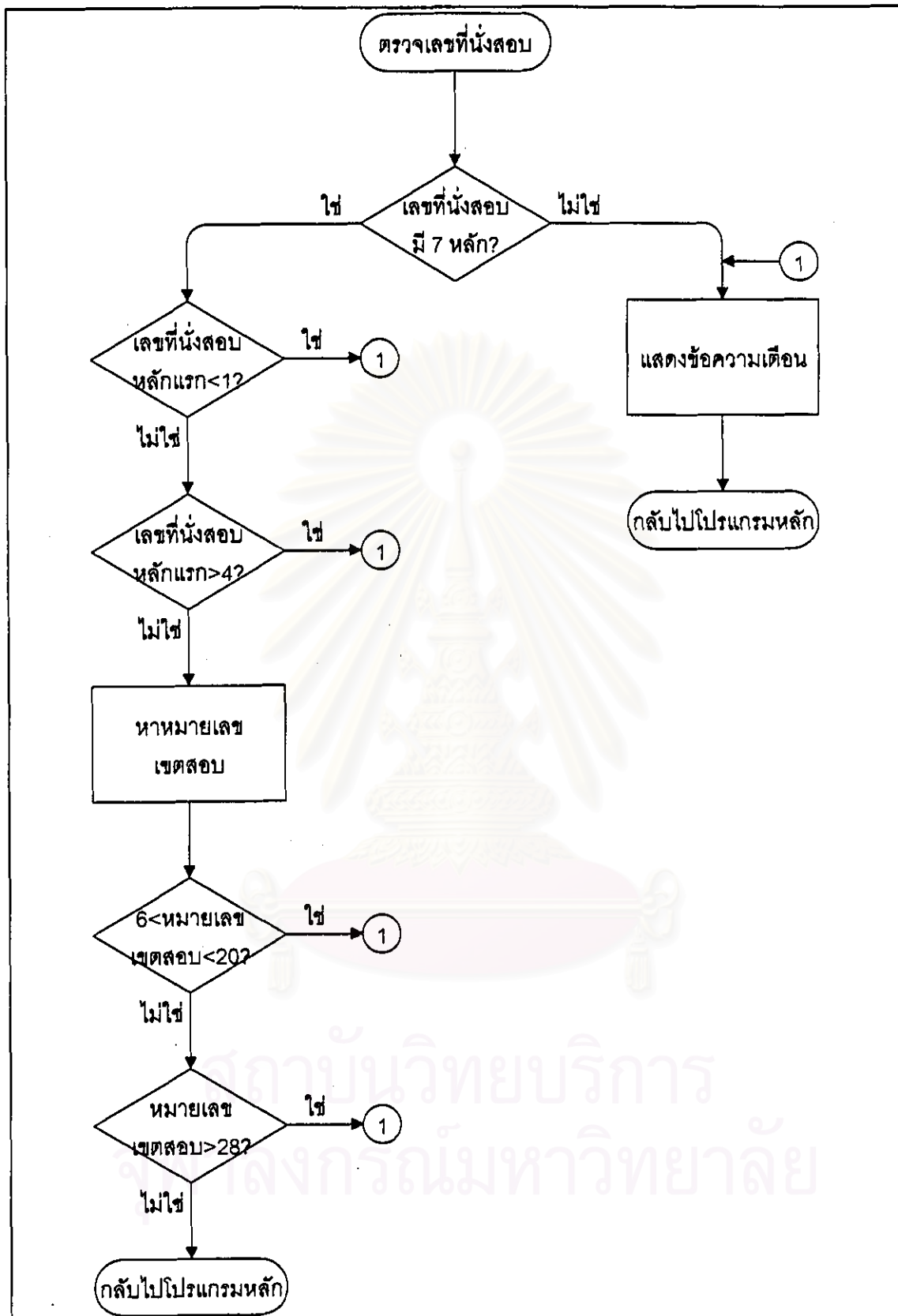
```
function chkValue(form) {
    Ctrl = form.seat;
    if ((Ctrl.value.length != 7) || (Ctrl.value.charAt(0) < "1") || (Ctrl.value.charAt(0) > "4")) {
        validatePrompt(Ctrl, "เลขที่นั่งรถไม่ถูกต้อง กรุณาป้อนเลขที่นั่งรถใหม่");
        return(false);
    }
    bound = Math.floor(Ctrl.value.substring(1,7) / 25000);
    if (((bound > 6) && (bound < 20)) || (bound > 28)) {
        validatePrompt(Ctrl, "เลขที่นั่งรถไม่ถูกต้อง กรุณาป้อนเลขที่นั่งรถใหม่");
        return(false);
    }
}
```

จากตัวอย่างจาวาสคริปต์นี้สามารถแสดงเงื่อนไขที่ใช้ในการตรวจสอบเลขที่นั่งรถด้วยผังงานในรูปที่ 4.4

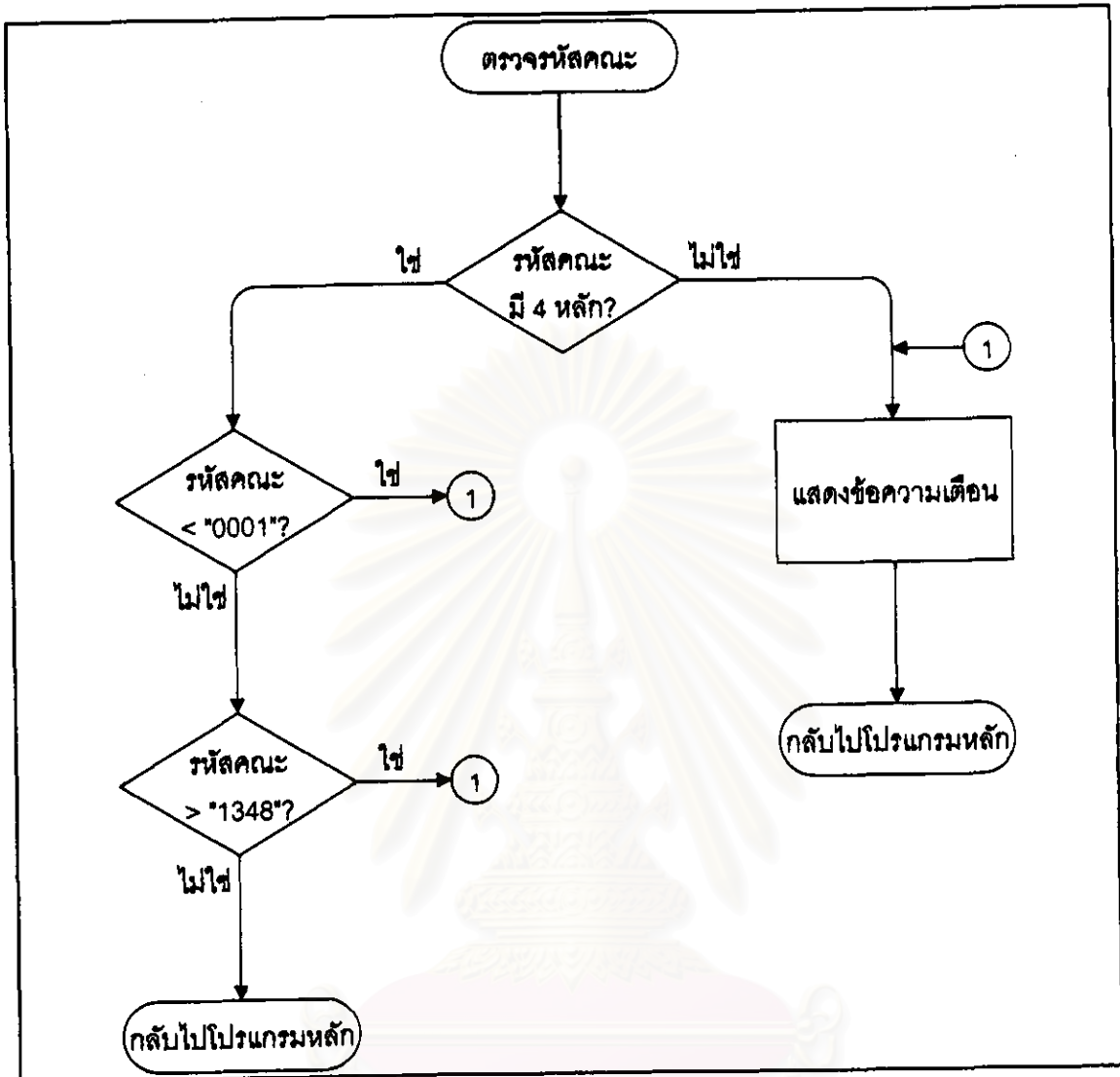
ตัวอย่างจาวาสคริปต์ที่ใช้ตรวจสอบรหัสคณะ

```
function chkValue(form) {
    Ctrl = form.fac;
    if ((Ctrl.value.length != 4) || (Ctrl.value < "0001") || (Ctrl.value > "1348")) {
        validatePrompt(Ctrl, "รหัสคณะหรือประเภทวิชาไม่ถูกต้อง กรุณาป้อนใหม่");
        return(false);
    }
}
```

ผังงานในรูปที่ 4.5 แสดงเงื่อนไขที่ใช้ในการตรวจสอบรหัสคณะ



รูปที่ 4.4 ผังงานการตรวจสอบเลขที่นั่งสอบ



รูปที่ 4.5 ผังงานการตรวจสอบรหัสดนนะ

โครงสร้างแฟ้มข้อมูล

แฟ้มข้อมูลที่ใช้ในการค้นหาสามารถแยกได้เป็น 2 ประเภท คือ

1. แฟ้มข้อมูลหลัก

แฟ้มข้อมูลหลักสำหรับการประกาศสถานที่สอบและผู้สอบผ่านข้อเขียนที่ใช้ในงานวิจัย เป็นแฟ้มที่ทบวงมหาวิทยาลัยใช้สำหรับพิมพ์และส่งให้ผู้สมัครสอบทางไปรษณีย์ โครงสร้างแฟ้มข้อมูลจะถูกกำหนดมาจากทบวงมหาวิทยาลัย แฟ้มข้อมูลนี้มีการจัดเก็บเป็นแฟ้มข้อความเรียงลำดับตามกฎแฉหลัก ข้อมูลของผู้สมัครแต่ละคนมีขนาดไม่เท่ากัน ถ้าให้ 1 บรรทัดของแฟ้มเป็น 1 ระเบียบ ผู้สมัครแต่ละคนจะมีข้อมูลได้หลายระเบียบ แต่ระเบียบมีอักขระ 9 ตัวแรกเป็นกฎแฉหลัก ซึ่งประกอบด้วย

- อักขระที่ 1 - 7 เป็นเลขที่นั่งสอบ
 อักขระที่ 8 - 9 เป็นเลขวิ่งตั้งแต่ 00 ถึง 99 บอกให้ทราบว่า เป็นข้อมูลบรรทัดที่เท่าไร
 ของเลขที่นั่งสอบนั้น บรรทัดสุดท้ายบอกด้วยอักขระ 99

อักขระตั้งแต่ตัวที่ 10 เป็นต้นไปเป็นส่วน of ข้อมูล

เลขวิ่งแต่ละค่าให้ข้อมูลแตกต่างกันไป เลขวิ่งหลักๆ ที่แสดงข้อมูล เช่น

1.1 เพิ่มข้อมูลสำหรับการประกาศสถานที่สอบ

- 02 แสดงเลขที่นั่งสอบ ชื่อ และเขตการสอบของผู้สมัคร
 04 แสดงอันดับการเลือก และรหัสวิชาที่สอบ
 06 แสดงรายละเอียดของสถานที่สอบ เช่น รหัสวิชา ห้องสอบ ชั้น
 อาคาร หน่วยสอบ ถ้ามีสถานที่สอบมากกว่าหนึ่งที่ จะแสดงราย
 ละเอียดของสถานที่สอบสำหรับวิชานั้นๆ ในระเบียบถัดไปโดยเพิ่ม
 ค่าเลขวิ่งครั้งละหนึ่ง
 99 หมดรายการของผู้สมัครหนึ่งคน

1.2 เพิ่มข้อมูลสำหรับการประกาศผลสอบข้อเขียน

- 01 แสดงเลขที่นั่งสอบ ชื่อ และผลการสอบข้อเขียน ถ้าสอบผ่านจะ
 แสดงรหัสคณะที่สอบผ่านด้วย
 02 ถ้าสอบผ่านจะแสดงรายละเอียดเกี่ยวกับการสอบสัมภาษณ์และ
 ตรวจร่างกายไว้ในระเบียบซึ่งมีเลขวิ่งตั้งแต่ 02 เป็นต้นไป
 99 หมดรายการของผู้สมัครหนึ่งคน

ตัวอย่างเพิ่มข้อมูลสำหรับการประกาศสถานที่สอบ

100000100	แบบแจ้งสถานที่สอบ
100000101	การสอบคัดเลือกเข้าศึกษาในสถานอุดมศึกษาทบวงปีการศึกษา 2540
100000102	เลขที่นั่งสอบ 1000001 ชื่อเกรียงไกร ทองกร เขตการสอบ เกษตรศาสตร์
100000103	อันดับการเลือก รหัสวิชาที่สอบ
100000104	0848 0126 0862 0578 01,02,03,04,05,06
100000105	รหัสวิชา ห้องสอบ ชั้น อาคาร หน่วยสอบ สถานที่
100000106	01-06 1307 03 อาคาร 1 มหาวิทยาลัยเกริก เขตการสอบเกษตรศาสตร์
100000199	

ตัวอย่างเพิ่มข้อมูลสำหรับการประกาศผลสอบ (สอบไม่ผ่าน)

100000100	ใบแจ้งผลสอบข้อเขียนทบวงมหาวิทยาลัย	
1000001011000001	นายเกรียงไกร ทองกร	สอบไม่ผ่าน
100000199		

ตัวอย่างเพิ่มข้อมูลสำหรับการประกาศผลสอบ (สอบผ่าน)

100000800	ใบแจ้งผลสอบข้อเขียนทบวงมหาวิทยาลัย	
1000008011000008	น.ส.กรรณก สิริวงศ์	สอบผ่าน คณะ 0873 เลขลำดับ 0001
100000802	สอบสัมภาษณ์ :	
100000803	13 พ.ค. 2540 09.00-12.00 น. ชั้น 7 ภาควิชาเทคโนโลยีอุตสาหกรรมเกษตร	
100000804	ตึกคณะวิทยาศาสตร์ประยุกต์	
100000805	ตรวจร่างกาย :	
100000806	13 พ.ค. 2540 09.00-16.00 น. งานสวัสดิการและอนามัย ชั้น 4 อาคารกิจกรรมนักศึกษา	
100000899	ท่านสามารถกู้เงินกองทุนให้กู้ยืมเงินเพื่อการศึกษาได้ที่สถาบันอุดมศึกษาทุกแห่ง	

2. เพิ่มดรรชนี

สำหรับเพิ่มข้อมูลหลักที่มีขนาดใหญ่ เช่น เพิ่มสำหรับการประกาศสถานที่สอบหรือผู้สอบผ่านข้อเขียน จะใช้ดรรชนีสองระดับช่วย โดยจัดเก็บในเพิ่มดรรชนี 2 เพิ่มด้วยกัน คือ

2.1 เพิ่มดรรชนีระดับที่สอง เก็บดรรชนีที่ชี้ไปยังเพิ่มข้อมูลหลัก มีขนาด 12 ไบต์ มีโครงสร้างดังนี้

```
char seat_no[8]
```

```
int offset
```

seat_no เป็นเลขที่นั่งสอบ ประกอบด้วยตัวเลข 7 หลัก ค่าของเลขที่นั่งสอบที่เป็นไปได้ตั้งแต่แสดงไว้ในภาคผนวก ก

offset บอกเลขที่อยู่เริ่มต้นของระเบียบบนเพิ่มข้อมูลหลัก โดยค่า offset นี้เริ่มนับจากต้นเพิ่มข้อมูลหลัก

2.2 แฟ้มดรรชนีระดับที่หนึ่ง เก็บดรรชนีที่ชี้ไปยังแฟ้มดรรชนีระดับที่สอง มีขนาด 8 ไบต์ มีโครงสร้างดังนี้

```
int    grp
int    offset
```

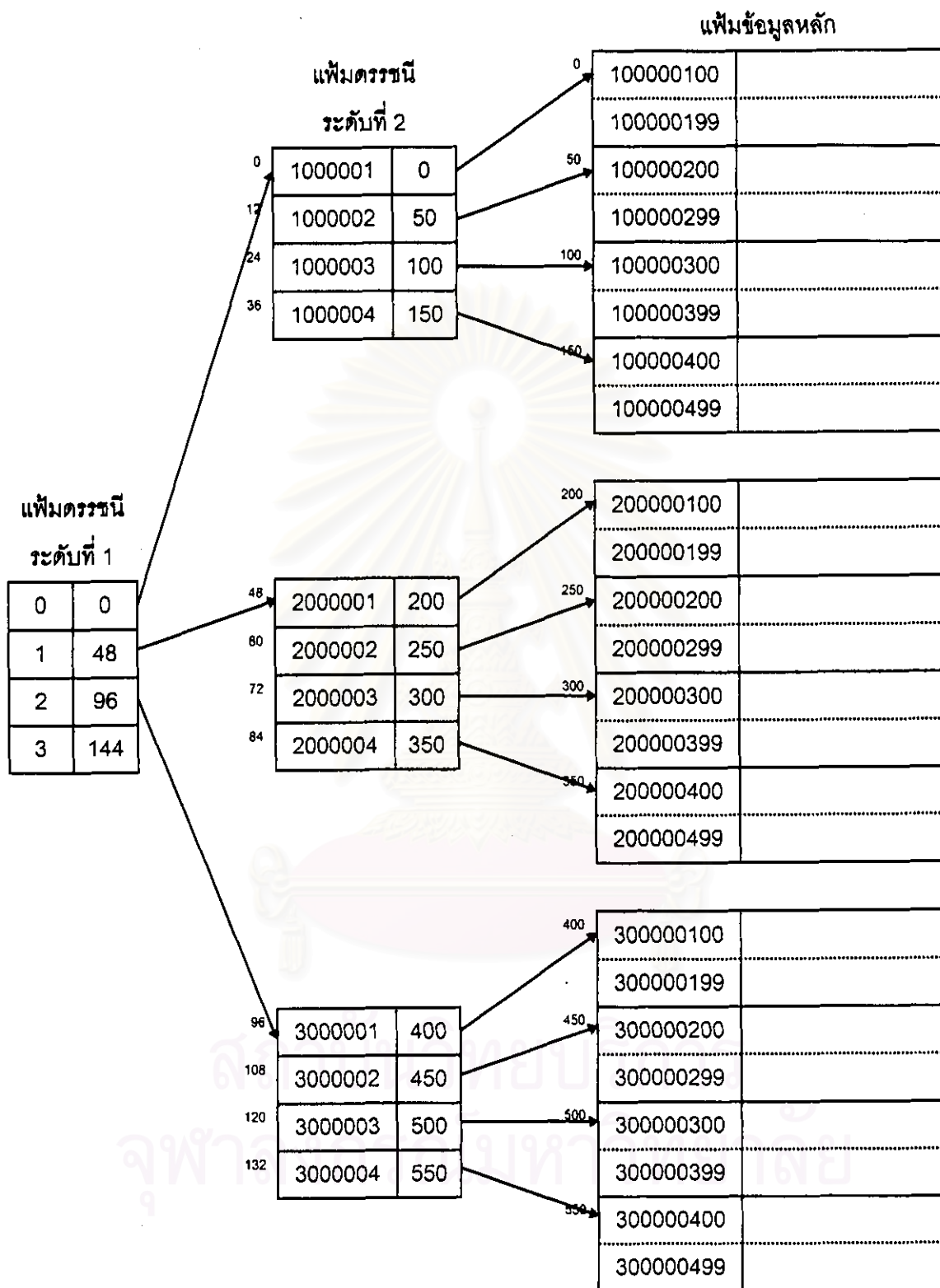
grp เป็นตัวเลขซึ่งบอกให้รู้ว่าเลขที่นั่งสอบนั้นอยู่ในกลุ่มไหน grp คำนวณได้ดังนี้

$$\text{grp} = \text{seat_no} / 25000 - 40$$

โดย 25000 คือ จำนวนหมายเลขที่เก็บเรียงกันอยู่ในแต่ละกลุ่ม (ตามที่จัดแบ่งไว้ดังแสดงในภาคผนวก ก) และเพื่อให้ grp เริ่มจาก 0 จึงต้องลบออกด้วย 40 (หมายเลขแรกของกลุ่มแรก คือ 1000000 เมื่อนำมาหารด้วย 25000 แล้วจะได้ 40 จึงลบออกด้วย 40 เพื่อให้ได้ค่าเป็น 0) ดังนั้นค่า grp ที่เป็นไปได้จะอยู่ระหว่าง 0 ถึง 148 โดยมีหมายเลขแรกของกลุ่ม 0 คือ 1000000 และหมายเลขสุดท้ายของกลุ่ม 148 คือ 4724999 (หมายเลขแรกของแต่ละกลุ่มจะไม่นำมาใช้เป็นเลขที่นั่งสอ เพราะฉะนั้นเลขที่นั่งสอแรกที่ใช้ได้ คือ 1000001 ส่วนหมายเลขสุดท้ายขึ้นอยู่กับจำนวนผู้สมัครในแต่ละเขตการสอบ)

offset บอกเลขที่อยู่เริ่มต้นของระเบียบแรกที่อยู่ในกลุ่ม โดยเลขที่อยู่นี้เริ่มนับจากต้นแฟ้มดรรชนีระดับที่สอง

หมายเหตุ ประเภทข้อมูล int ในภาษาซีบนระบบปฏิบัติการ Microsoft Windows NT มีขนาด 4 ไบต์



รูปที่ 4.6 ความสัมพันธ์ระหว่างเพิ่มดรรชนีระดับที่ 1 เพิ่มดรรชนีระดับที่ 2 และเพิ่มข้อมูลหลัก

ขั้นตอนการค้นหา

1. หากกลุ่มของเลขที่นั่งสอบ และหาเลขที่อยู่ซึ่งชี้ไปแฟ้มตวรรษในระดับที่สอง

นำเลขที่นั่งสอบมาคำนวณว่าอยู่ในกลุ่มไหน ($grp = seat_no / 25000 - 40$) จากนั้นคำนวณเลขที่อยู่ของระเบียบบนแฟ้มตวรรษระดับที่หนึ่ง โดยการคูณหมายเลขกลุ่มด้วยขนาดระเบียบของแฟ้มตวรรษระดับที่หนึ่ง ($addr = grp * 8$) เขตข้อมูลที่สองของระเบียบในแฟ้มตวรรษระดับที่หนึ่งเป็นเลขที่อยู่ซึ่งชี้ไปยังเลขที่นั่งสอบแรกในกลุ่มบนแฟ้มตวรรษระดับที่สอง

2. หาเลขที่อยู่ซึ่งชี้ไปแฟ้มข้อมูลหลัก

เขตข้อมูลแรกของระเบียบในแฟ้มตวรรษระดับที่สองเป็นเลขที่นั่งสอบเรียงลำดับกัน และเลขที่นั่งสอบที่อยู่ในกลุ่มเดียวกันจะเรียงลำดับกันอย่างต่อเนื่อง (เลขที่นั่งสอบในระเบียบถัดไปจะมีค่ามากกว่าเลขที่นั่งสอบในระเบียบปัจจุบันอยู่หนึ่งเสมอ ไม่มีการกระโดดข้าม) ดังนั้นการคำนวณเลขที่อยู่ของระเบียบที่มีเลขที่นั่งสอบที่ต้องการค้นหา สามารถทำได้โดยการหาผลต่างระหว่างเลขที่นั่งสอบที่ต้องการค้นหากับเลขที่นั่งสอบแรกในกลุ่ม ($rec_no = seat_no - index.seat_no$) แล้วนำผลต่างที่ได้นั้นคูณกับขนาดระเบียบของแฟ้มตวรรษระดับที่สอง ($addr = rec_no * 12$) เขตข้อมูลที่สองของระเบียบที่ได้เป็นเลขที่อยู่ซึ่งชี้ไปยังระเบียบบนแฟ้มข้อมูลหลัก

3. ดึงข้อมูลและจัดรูปแบบข้อมูลเป็น HTML

เนื่องจากข้อมูลของแต่ละเลขที่นั่งสอบมีรายละเอียดมาก จึงจัดให้แต่ละเลขที่นั่งสอบมีข้อมูลหลายระเบียบ แต่ละระเบียบจะมีกฎแฉกหลักเป็นเขตข้อมูลแรกบอกให้ทราบว่าระเบียบนั้นเป็นระเบียบที่เท่าไรของเลขที่นั่งสอบไหน ข้อมูลที่จัดเก็บในแฟ้มข้อมูลหลักเรียงลำดับกันตามกฎแฉกหลัก (กฎแฉกหลักประกอบด้วยเลขที่นั่งสอบ 7 อักขระและเลขวง 2 อักขระเริ่มจาก 00 ถ้าเป็นระเบียบสุดท้ายของเลขที่นั่งสอบจะระบุเลขวงเป็น 99)

การดึงข้อมูลทำได้โดยการอ่านข้อมูลที่ระเบียบ ตรวจสอบว่าเลขวงของระเบียบนั้นเป็นค่าอะไรเพื่อให้ทราบรูปแบบของข้อมูล เลือกข้อมูลส่วนที่ต้องการเก็บไว้ในตัวแปร เมื่อหมดระเบียบจึงส่งแสดงข้อความในรูปแบบ HTML จากนั้นจึงนำระเบียบถัดไปมาประมวลผล ทำเช่นนี้จนกว่าเลขวงเป็น 99

สำหรับการประกาศผลสอบข้อเขียนนั้น ในกรณีที่สอบผ่าน ทบวงมหาวิทยาลัยจะแจ้งคณะที่สอบผ่านเป็นรหัสคณะ ส่วนการนำเสนอม่านเว็บจะแจ้งชื่อคณะให้ทราบด้วย โดยใช้รหัสคณะเป็นกฎแฉกหลักไปค้นหาชื่อคณะจากแฟ้มข้อมูลซึ่งเก็บรหัสและชื่อคณะไว้ ก่อนการค้นหาต้องตรวจสอบว่าสอบผ่านหรือไม่ โดยทำการตรวจกวาด (scan) ระเบียบที่เก็บผลการสอบ (ในที่นี้คือระเบียบที่มีเลขวง

เป็น 01) ถ้าตรวจพบคำว่า “สอบผ่าน” จึงทำการตรวจกวาดระเบียบนั้นต่อจนพบรหัสคณะ แล้วจึงนำรหัสคณะนั้นไปค้นหาชื่อคณะต่อไป

3.1 เพิ่มข้อมูลที่ใช้ในการค้นหาชื่อคณะ

เพิ่มข้อมูลที่เก็บรหัสและชื่อคณะมีระเบียบซึ่งประกอบด้วย 2 เขตข้อมูล เขตข้อมูลแรกเป็นรหัสคณะมี 4 อักขระเริ่มจาก 0001 เรียงกันไปจนถึง 1348 (รหัสคณะ/ประเภทวิชาที่เปิดสอบในปี 2540) เขตข้อมูลที่สองเป็นชื่อคณะ แต่ละระเบียบมีขนาดคงที่เป็น 180 ไบต์

ตัวอย่างเพิ่มข้อมูลซึ่งเก็บรหัสและชื่อคณะ

0001	คณะแพทยศาสตร์	จุฬาลงกรณ์มหาวิทยาลัย
0002	คณะสหเวชศาสตร์ สาขาวิชาเทคนิคการแพทย์	จุฬาลงกรณ์มหาวิทยาลัย
0003	คณะสหเวชศาสตร์ สาขาวิชากายภาพบำบัด	จุฬาลงกรณ์มหาวิทยาลัย
0004	คณะเภสัชศาสตร์	จุฬาลงกรณ์มหาวิทยาลัย
0005	คณะสัตวแพทยศาสตร์	จุฬาลงกรณ์มหาวิทยาลัย
0006	คณะทันตแพทยศาสตร์	จุฬาลงกรณ์มหาวิทยาลัย

3.2 การค้นหาชื่อคณะ

เนื่องจากรหัสคณะเรียงลำดับกันอย่างต่อเนื่องและแต่ละระเบียบมีขนาดเท่ากัน การหาเลขที่อยู่ของระเบียบจึงสามารถคำนวณได้จากรหัสคณะและขนาดของระเบียบ ($addr = (fac_code - 1) * 180$) เมื่อได้เลขที่อยู่ของระเบียบที่ต้องการแล้วจึงดึงส่วนที่เป็นชื่อคณะของรหัสคณะที่ต้องการมาเก็บไว้ในตัวแปรเพื่อนำไปใช้ต่อไป

เช่นเดียวกับข้อมูลที่ได้จากระเบียบอื่นๆ เมื่อได้ชื่อคณะแล้วก็นำชื่อคณะที่ค้นหาได้นั้นจัดรูปแบบเป็น HTML เพื่อให้สามารถแสดงผลผ่านเว็บเบราว์เซอร์ในรูปแบบที่ผู้ใช้สามารถอ่านได้ง่าย

สรุป

ในบทนี้กล่าวถึงการจัดการกับข้อมูลด้วยวิธีการต่างๆ เพื่อนำข้อมูลไปใช้ในแต่ละขั้นตอนการทำงาน นอกจากการใช้โปรแกรมประมวลผลคำเพื่อสร้างแฟ้ม HTML แล้ว ยังสามารถนำโปรแกรมอื่น เช่น ไมโครซอฟต์เอ็กเซล มาประยุกต์ใช้ในการแปลงแฟ้มข้อมูลให้เป็นแฟ้ม HTML ได้ ส่วนการสืบค้นข้อมูลที่มีในงานวิจัยนี้ใช้หลักการของดรรชนีเพื่อให้การสืบค้นทำได้อย่างรวดเร็ว โดยใช้ดรรชนีสองระดับสำหรับการสืบค้นแฟ้มข้อมูลที่มีขนาดใหญ่