

บทที่ 4

การสร้างและพัฒนาโปรแกรม

ในบทนี้จะกล่าวถึงลำดับขั้นตอน ในการสร้างและทดสอบโปรแกรม ซึ่งประกอบไปด้วย การรวบรวมข้อมูลเพื่อนำมาใช้ใน การออกแบบฐานข้อมูล การเขียนผังงานเพื่อใช้ในการเขียนโปรแกรม การเข้ารหัสข้อมูลที่ใช้ในการป้องกันรหัสผ่าน และการทดสอบโปรแกรม

4.1 การออกแบบฐานข้อมูล

จากข้อมูลที่รวบรวมได้เรานำมาใช้ในการออกแบบฐานข้อมูลโดยการเขียน ER-Diagram (Entity Relationship Diagram) เพื่อแสดงความสัมพันธ์ของข้อมูล ได้ดังนี้

ER-Diagram ประกอบไปด้วย 3 เอนทิตี คือ

นิสิต (Student) , กลุ่มเรียน (Section) , รายวิชาที่เรียน (Course)

แต่ละเอนทิตีมีแอททริบิวต์ประกอบดังนี้

1. นิสิต (Student) ประกอบไปด้วย

รหัสประจำตัว (ID) , ชื่อ-นามสกุล (Name) , ชั้นปี (Class) , วิชาเอก (Major) , รหัสประจำชั้นปี (Curri code) และ รหัสผ่าน(Password)

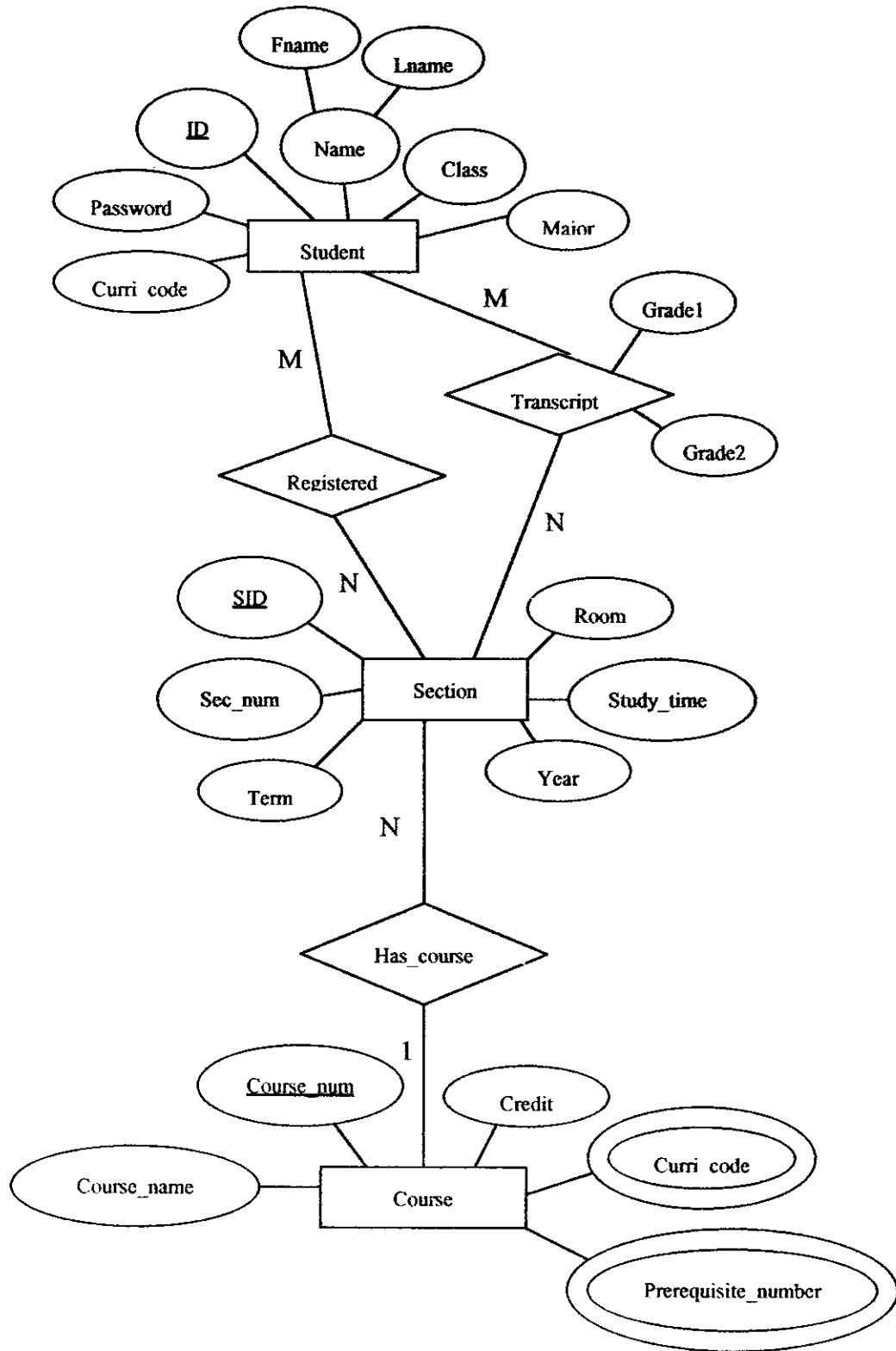
2. กลุ่มเรียน(Section) ประกอบไปด้วย

เลขลำดับกลุ่มเรียนที่กำหนดเพื่อเป็นคีย์หลัก (SID) , หมายเลขประจำกลุ่มเรียน (Sec number) , เวลาเรียน (Strdy time) , ปีการศึกษา (Year) , ภาคเรียน (Term) , ห้องเรียน (Room)

3. รายวิชาที่เรียน (Course) ประกอบไปด้วย

รหัสวิชา (Course number) , ชื่อวิชา (Course name) , หน่วยกิต (Credit) , รหัสวิชา บังคับก่อนหน้า (Prerequisite number) , รหัสประจำชั้นปี (Curri code)

จากนั้นกำหนดความสัมพันธ์ระหว่างแต่ละเอนทิตี และเขียน ER-Diagram แสดงได้ดังรูป



รูปที่ 4.1 ER-Diagram

จาก ER-Diagram นำมาเขียนเป็นตารางได้ดังนี้

Student

ID	Fname	Lname	Class	Major	Curri_code	Password
----	-------	-------	-------	-------	------------	----------

Course

Course_number	Course_name	Credit	Prerequisite_num	Curri_code
---------------	-------------	--------	------------------	------------

Section

Course_number	SID	Sec_num	Study_time	Year	Term	Room
---------------	-----	---------	------------	------	------	------

Transcript

ID	SID	Grade1	Grade2
----	-----	--------	--------

Register

<u>ID</u>	<u>SID</u>
-----------	------------

แปลงการออกแบบทางตรรกที่ ได้ให้เป็นรูปแบบนอร์มัล โดย

รูปแบบนอร์มัลที่หนึ่ง

คือทำให้ไม่มีคอลัมน์ (แอทริบิวท์) ที่เป็นคอลัมน์ที่มีหลายค่า (multivalue) หรือกลุ่มซ้ำกัน (repeating groups) พุคอีกอย่างหนึ่งแต่ละคอลัมน์ในแถวหนึ่งสามารถมีได้เพียงค่าเดียว
นำมานอร์มัลไหลครั้งที่ 1 จะได้ตาราง และรูปแบบความสัมพันธ์ดังนี้

Student

ID	Fname	Lname	Class	Major	Curri_code	Password
↑						
↑						

Course

Course_number	Course_name	Credit
↑		
↑		

Prerequisite

<u>Course number</u>	<u>Prerequisite num</u>
----------------------	-------------------------

Curri_code

<u>Course number</u>	<u>Curri code</u>
----------------------	-------------------

Section

<u>Course_number</u>	<u>SID</u>	<u>Sec_num</u>	<u>Study_time</u>	<u>Year</u>	<u>Term</u>	<u>Room</u>
----------------------	------------	----------------	-------------------	-------------	-------------	-------------

Transcript

<u>ID</u>	<u>SID</u>	<u>Grade1</u>	<u>Grade2</u>
-----------	------------	---------------	---------------

Register

<u>ID</u>	<u>SID</u>
-----------	------------

รูปแบบฟอร์มัลที่สอง

เมื่อคีย์หลักประกอบด้วยมากกว่าหนึ่งคอลัมน์ (แอทริบิวต์) คอลัมน์ (แอทริบิวต์) ที่ไม่ได้เป็นคีย์หลักทั้งหมดต้องขึ้นกับทั้งคีย์หลัก และ ไม่ใช่ส่วนหนึ่งของคีย์หลัก
ข้อสังเกตคีย์หลักที่สร้างขึ้นจากมากกว่าหนึ่งแอทริบิวต์เรียกว่าคีย์ผสม
ตารางข้างบนเป็นรูปแบบฟอร์มัลที่ 2 แล้วจึงไม่ต้องทำอะไรอีก

รูปแบบฟอร์มัลที่สาม

แอทริบิวต์ที่ไม่ใช่คีย์ต้อง ไม่ขึ้นกับแอทริบิวต์อื่นๆ ที่ไม่ใช่คีย์
จากตาราง Course ไม่ผ่านรูปแบบฟอร์มัลที่ 3 จึงต้องแยกตาราง Course ออกเป็น 2 ตารางคือ

Course

<u>Course Number</u>	<u>Course name</u>
----------------------	--------------------

Credit

<u>Course name</u>	<u>Credit</u>
--------------------	---------------

4.2 การออกแบบโปรแกรม

โปรแกรมจัดการระเบียนแสดงผลการศึกษาตาม CD-ROM ที่แนบมามีขั้นตอนในการออกแบบดังนี้

4.2.1 การวิเคราะห์งาน**1. วิเคราะห์สิ่งที่ต้องการ**

- ❖ ส่วนที่หนึ่งตรวจสอบผลการเรียน
 - ผลการเรียนเฉลี่ยสะสม (GPA : Grade point average)
 - หน่วยการเรียนสะสม (Total credit earned)
 - ผลการเรียนรายวิชาในแต่ละเทอม
 - ผลการเรียนเฉลี่ยในแต่ละเทอม
- ❖ ส่วนที่สองตรวจสอบรายวิชาที่สามารถลงทะเบียนในเทอมถัดไป
 - รายชื่อวิชาในเทอมถัดไป
 - รายชื่อวิชาในเทอมถัดไปที่สามารถลงทะเบียนได้
 - รายชื่อวิชาในเทอมถัดไปที่ไม่สามารถลงทะเบียนได้
 - เหตุผลที่ไม่สามารถลงทะเบียนรายวิชานั้นได้
- ❖ ส่วนที่สามตารางเรียนในเทอมถัดไป
 - ตารางเรียนในเทอมถัดไปแบ่งสีตามรายวิชา

2. วิเคราะห์ลักษณะของรายงาน

ลักษณะของรายงานที่ต้องการให้เครื่องคอมพิวเตอร์แสดงผลเป็นเอกสารเอชทีเอ็มแอล (Html : Hypertext Markup Language) ซึ่งสามารถแสดงผลได้ในบราวเซอร์ (Browser) ทั่วไป เช่น Netscape Navigator หรือ Internet Explorer

❖ ส่วนที่หนึ่งตรวจสอบผลการเรียน

- แสดงหัวรายงานซึ่งประกอบไปด้วย ชื่อ, นามสกุล, สาขา, ชั้นปี, ผลการเรียนเฉลี่ยสะสม และ หน่วยการเรียนสะสมบนสุดของรายงาน
- แสดงผลการเรียนประจำแต่ละปีการศึกษาแต่ละภาคการเรียนเรียงตามเวลาให้ภาคการศึกษาที่เรียนที่หลังแสดงก่อนในรูปแบบตาราง โดยมีหัวสคมภ์ดังนี้ รหัสวิชา, ชื่อวิชา, เกรดคิด, เกรด
- แสดงผลการเรียนเฉลี่ยประจำแต่ละปีการศึกษาแต่ละภาคการเรียนท้ายตารางด้านบนแต่ละตาราง

❖ ส่วนที่สองตรวจสอบรายวิชาที่สามารถลงทะเบียนในเทอมถัดไป

- แสดงหัวรายงานซึ่งประกอบไปด้วย ชื่อ, นามสกุล, สาขา และ ชั้นปีบนสุดของรายงาน
- แสดงตารางรายวิชาในเทอมถัดไป โดยมีหัวสคมภ์คือ รหัสวิชา, ชื่อวิชา และหน่วยกิต
- แสดงรายวิชาในเทอมถัดไปที่สามารถลงทะเบียนได้ โดยมีหัวสคมภ์คือ รหัสวิชา, ชื่อวิชาและหน่วยกิต
- แสดงรายวิชาในเทอมถัดไปที่ไม่สามารถลงทะเบียนได้ โดยมีหัวสคมภ์คือ รหัสวิชา, ชื่อวิชา, หน่วยกิตและหมายเหตุ

❖ ส่วนที่สามตารางเรียนในเทอมถัดไป

- แสดงหัวรายงานซึ่งประกอบไปด้วย ชื่อ, นามสกุล, สาขา และ ชั้นปีบนสุดของรายงาน
- แสดงตารางเรียน โดยตารางเรียนที่แสดงผลนั้นประกอบไปด้วย วัน, เวลา, รายชื่อวิชา, ชื่อห้องเรียนของวิชาที่ผู้ใช้มีสิทธิ์ลงทะเบียนแบ่งตามสี

3. การวิเคราะห์ลักษณะของข้อมูล

❖ ลักษณะของข้อมูลที่จะต้องป้อนเข้าไปในเครื่องคอมพิวเตอร์

- มี 2 ลักษณะคือเป็นตัวหนังสือกับตัวเลข รหัสนิสิตเป็นตัวเลข รหัสผ่านเป็นตัวหนังสือกับตัวเลขผสมกัน
- เป็นข้อมูลภายนอกที่รับจากผู้ใช้
- มีแหล่งกำเนิดโดยเป็นข้อมูลส่วนตัว

❖ ลักษณะของข้อมูลที่เก็บในฐานข้อมูล

- มี 2 ลักษณะคือเป็นตัวหนังสือกับตัวเลข
- เป็นข้อมูลภายในเก็บไว้ในเครื่องคอมพิวเตอร์
- มีแหล่งกำเนิด โดยเป็นข้อมูลจากหน่วยงานคือจากฝ่ายวิชาการ

มหาวิทยาลัยนเรศวร

❖ พิจารณาจากรูปแบบลักษณะรายงานแล้วพบว่า

- ค่าบางค่าที่นำมาแสดงผลได้จากข้อมูลที่เก็บในฐานข้อมูล
- ค่าบางค่าที่นำมาแสดงผลได้มาจากการคำนวณ
- ข้อมูลที่จะทำให้ได้ผลลัพธ์ในแต่ละส่วนเป็นข้อมูลที่ได้จากฐานข้อมูล

นิสิต

- ข้อมูลที่ต้องป้อนเข้าเครื่องคือรหัสนิสิตกับรหัสผ่านเท่านั้น

4. วิเคราะห์การประมวลผล

❖ ส่วนที่หนึ่งตรวจสอบผลการเรียน

- การคำนวณผลการเรียนเฉลี่ยในแต่ละภาคการศึกษาได้จากสูตร

ผลการเรียนเฉลี่ย = (ผลรวมของ (หน่วยการเรียน X ค่าประจำผลการเรียน)) / หน่วยการเรียนรวม

● วิธีคำนวณ

- อ่านค่าหน่วยการเรียนและผลการเรียนประจำภาคการศึกษา
- แปลงผลการเรียนมาเปลี่ยนเป็นค่าประจำผลการเรียน
- นำค่าประจำผลการเรียนมาคูณกับหน่วยการเรียนทุกวิชา
- นำค่าประจำผลการเรียนที่คูณกับหน่วยการเรียนทุกวิชากัน แต่ละค่ามารวมกัน
- หาหน่วยการเรียนรวมจากหน่วยการเรียนในแต่ละวิชา
- หาผลการเรียนเฉลี่ยตามสูตรคำนวณ

❖ ส่วนที่สองตรวจสอบรายวิชาที่สามารถลงทะเบียนในเทอมถัดไป

- รายวิชาในเทอมถัดไปไม่ต้องคำนวณนำค่าจากฐานข้อมูลมาแสดงได้

เลข

➤ **คำนวณรายวิชาในเทอมถัดไปที่สามารถลงทะเบียนได้**

● **วิธีคำนวณ**

- อ่านค่ารายวิชาในเทอมถัดไป
- อ่านค่ารายวิชาก่อนหน้าของแต่ละรายวิชาที่อ่านได้จากข้อ
คำนวณ
- อ่านค่าผลการเรียนของรายวิชาก่อนหน้า
- ตรวจสอบผลการเรียนของรายวิชาก่อนหน้า

ถ้าได้ผลการเรียน = “E”

รายวิชาในเทอมถัดไปที่มีรายวิชาก่อนหน้านี้นี้ไม่ต้อง
แสดงยังไม่มีผลการเรียน (ยังไม่ได้เรียน)

รายวิชาในเทอมถัดไปที่มีรายวิชาก่อนหน้านี้นี้ไม่ต้อง
แสดง

ถ้าได้ผลการเรียน “A” หรือ “B” หรือ “C” หรือ “D”

รายวิชาในเทอมถัดไปที่มีรายวิชาก่อนหน้านี้นี้ที่ไม่ใช่วิชา
ใน 2 เงื่อนไขข้างบนแสดงได้

- แสดงผลลัพธ์

คำนวณรายวิชาในเทอมถัดไปที่ไม่สามารถลงทะเบียน
ได้จากการคำนวณข้างบน โดยเอาวิชาที่ไม่ได้แสดงข้าง
บนมาแสดงในส่วนนี้และแสดงเหตุผลที่ไม่สามารถลง
ทะเบียนได้เพิ่มเข้าไป

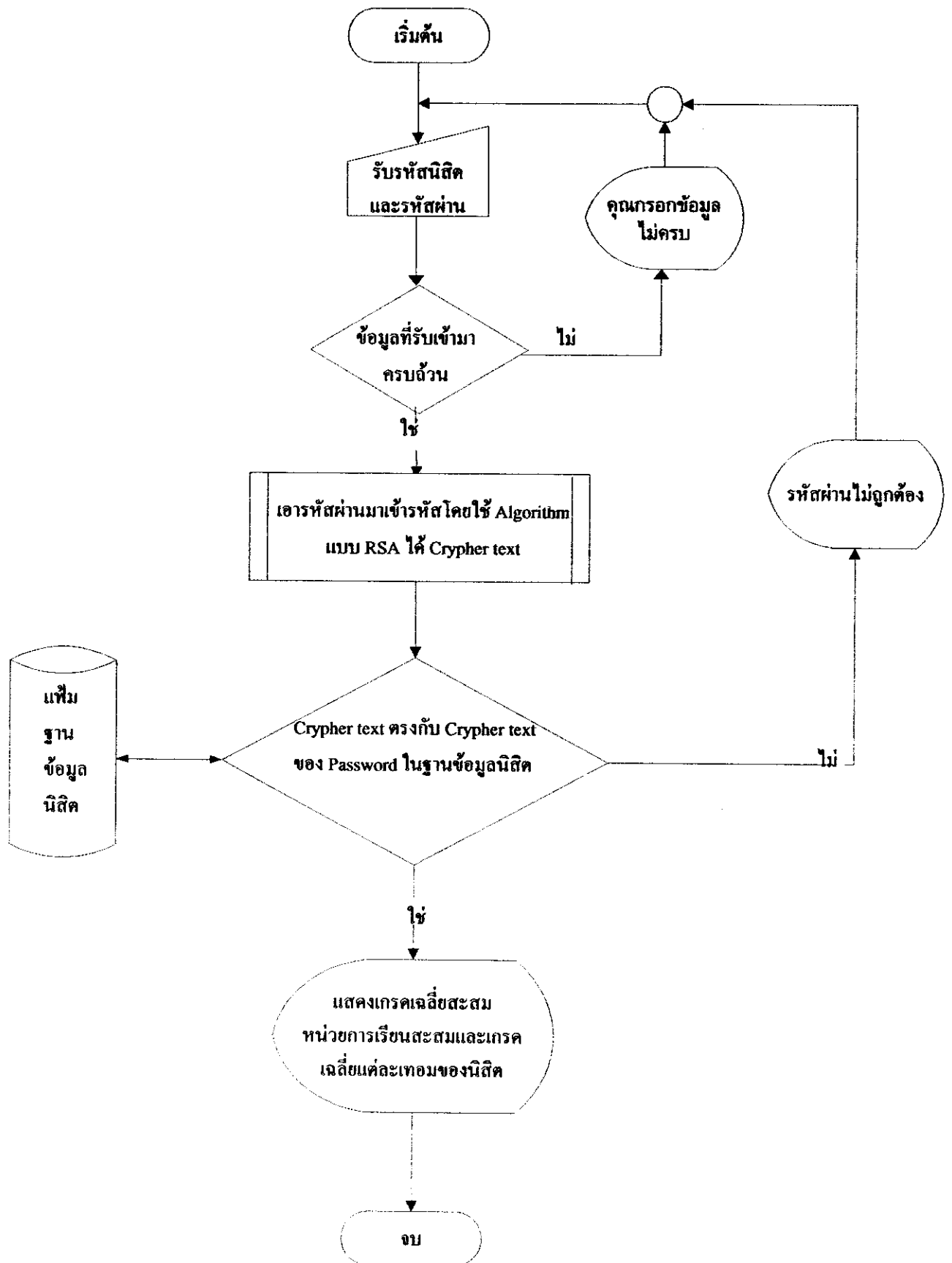
❖ **ส่วนที่สามตารางเรียนในเทอมถัดไป**

➤ **คำนวณตารางเรียนมีวิธีดังนี้**

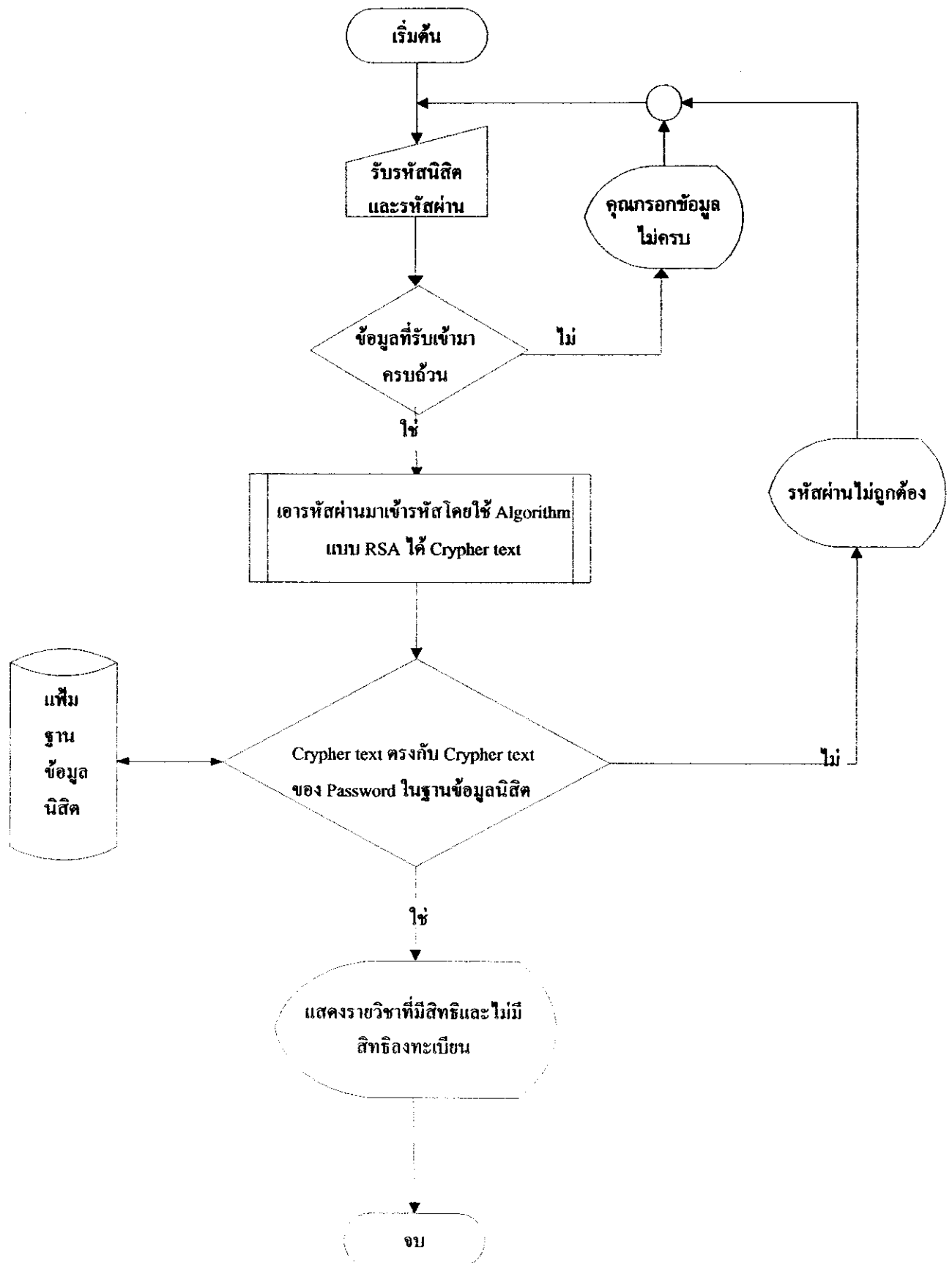
- อ่านค่าเวลาเรียนของรายวิชาที่ต้องเรียนในเทอมถัดไปที่มี curri
code ตรงกับคนที่ป้อนข้อมูลเข้ามา
- แสดงตารางเรียนแต่ละรายวิชาในตารางแบ่งแยกตามสีและแสดง
ห้องเรียน

4.2.2 การเขียนผังงาน

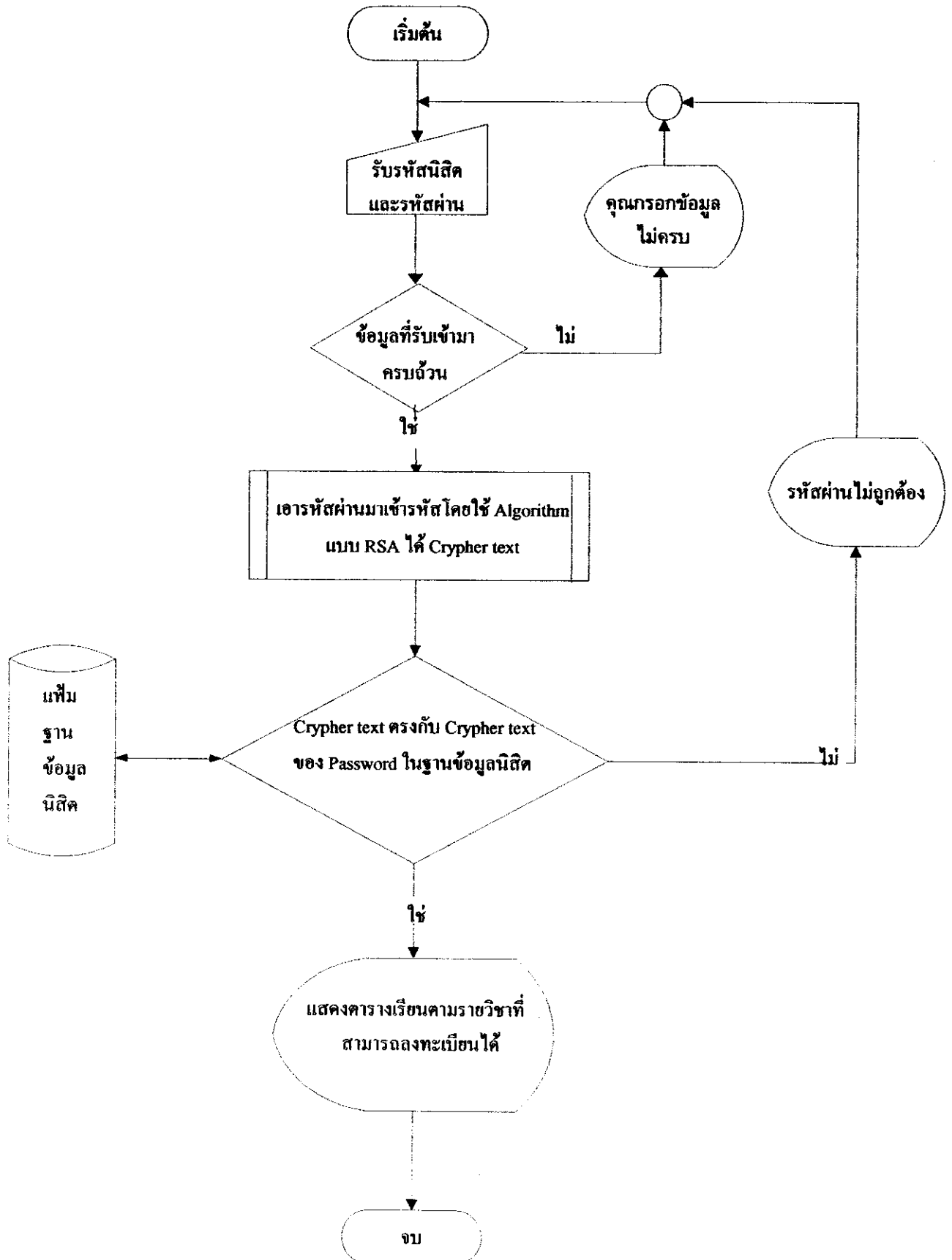
จากขั้นตอนการทำงานของโปรแกรมสามารถเขียนผังงานได้ดังนี้



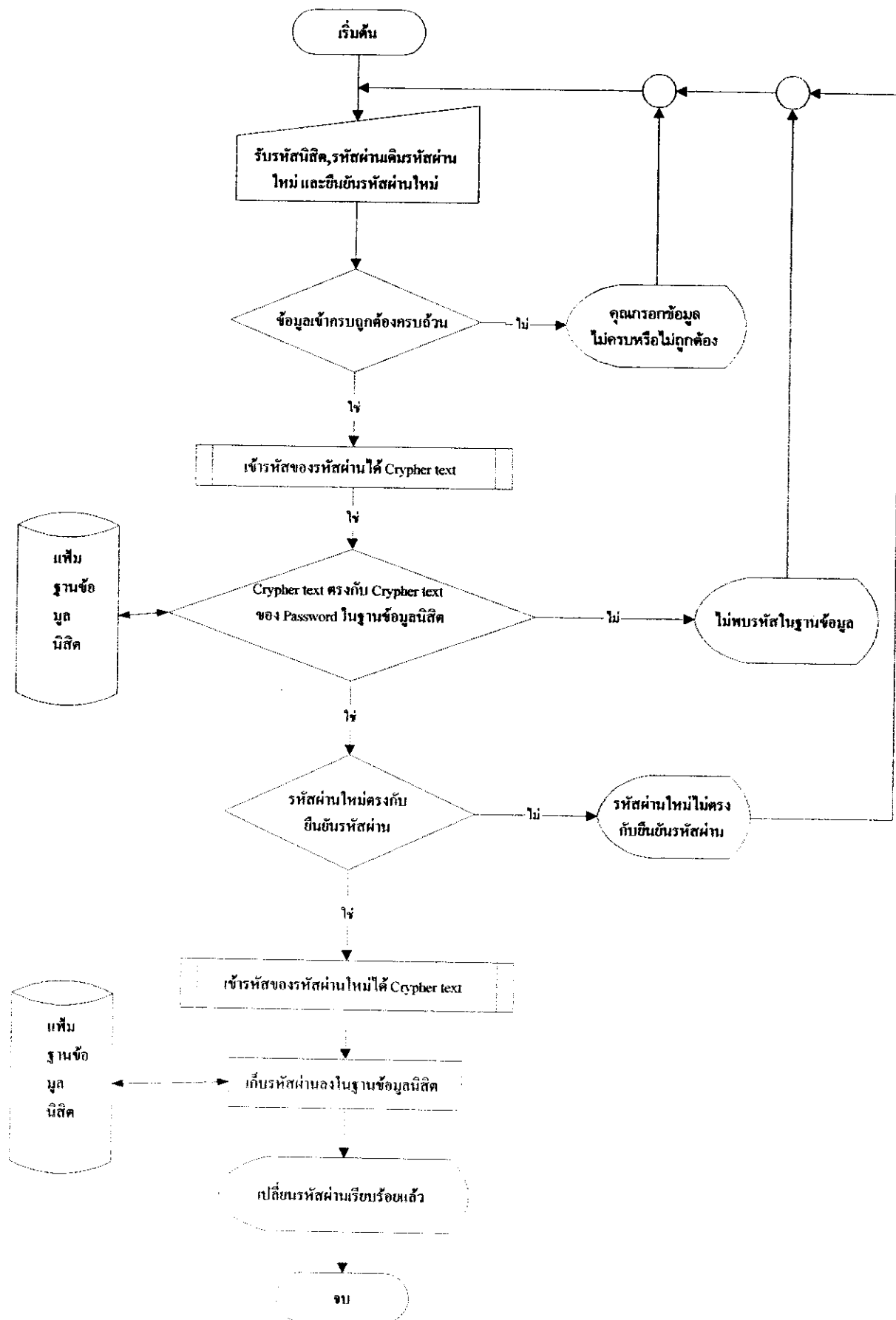
รูปที่ 4.2 แผนผังงานของโปรแกรมส่วนของการตรวจสอบผลการเรียน



รูปที่ 4.2 แผนผังงานของโปรแกรมส่วนของการตรวจสอบรายวิชาที่มีสิทธิและไม่มีสิทธิลงทะเบียน



รูปที่ 4.2 แผนผังงานของโปรแกรมส่วนของการตรวจสอบตารางเรียน



รูปที่ 4.5 แผนผังงานของ โปรแกรมส่วนของการเปลี่ยนรหัสผ่าน

4.3 การเข้ารหัสด้วยวิธีการแบบ RSA เพื่อให้ในการเก็บรหัสผ่าน

จากทฤษฎีการเข้ารหัสแบบ RSA นั้นเราได้นำมาประยุกต์ใช้ในส่วนของการเก็บรหัสผ่านของผู้ใช้งานเพื่อให้เกิดความปลอดภัยมากขึ้น โดยก่อนที่จะนำรหัสผ่านของผู้ใช้งานมาเก็บลงในฐานข้อมูลโดยตรงนั้นเราจะทำการแปลงรหัสผ่านด้วยการเข้ารหัสแบบ RSA ก่อน

ตัวอย่างในการแปลงรหัสผ่านของผู้ใช้ก่อนเก็บลงในฐานข้อมูลแสดงได้ดังนี้ เมื่อผู้ใช้ตั้งรหัสผ่านของตนเองว่า “STOP” แทนที่เราจะเก็บคำว่า “STOP” โดยตรงในฐานข้อมูลรหัสผ่านของผู้ใช้งาน

เราจะต้องทำการแปลงให้อยู่ในรูปของรหัสแบบ RSA ด้วยวิธีการดังนี้

- ❖ นำรหัสผ่านที่ผู้ใช้งานตั้งมาแปลงให้อยู่ในรูปของตัวเลขเพื่อนำไปคำนวณ การแปลงรหัสผ่านให้เป็นตัวเลขนั้นจะใช้วิธีการแทนค่าของรหัสผ่าน โดยกำหนดค่าที่แทนดังนี้

A-Z	แทนด้วย	00-25
a-z	แทนด้วย	26-51
0-9	แทนด้วย	52-61

จากคำว่า “STOP” แทนได้ดังนี้ S = 18, T = 19, O = 14 และ P = 15 เพราะฉะนั้นคำว่า STOP จะมีค่าเท่ากับ “18191415”

- ❖ ในการเข้ารหัสนี้จากสูตร $c = m^e \pmod n$; $n = pq$

เรากำหนดค่า $p = 43$, $q = 59$ จากการคำนวณได้ค่า $n = pq = 2537$

และใช้ค่า $e = 13$ เพราะว่า $\text{gcd}(13, (42 * 58)) = 1$

- ❖ การเข้ารหัสจะนำเลขที่ได้จากการแปลงมาทำการเข้ารหัสโดยจะใช้ทีละคู่ จากตัวอย่างรหัสผ่านเท่ากับ “STOP” เพราะฉะนั้น

ชุดแรกใช้ “ST” = “1819”

จาก $c = m^e \pmod n$

จะได้ $c = 1819^{13} \pmod 2537$

จากการคำนวณได้ $c = 2081$ เป็นชุดที่หนึ่ง

ชุดที่สอง “OP” = “1415”

จาก $c = m^e \pmod n$

จะได้ $c = 1415^{13} \pmod 2537$

จากการคำนวณได้ $c = 2182$ เป็นชุดที่สอง

- ❖ เพราะฉะนั้นจากการเข้ารหัสเราจะได้ค่าของรหัสผ่าน “STOP” เป็น “20812182” ซึ่งเป็นค่าที่เราใช้เก็บลงในฐานข้อมูล