

โปรแกรมพัฒนาการเรียนรู้อีกก่อนวัยเรียน

A DEVELOPMENT OF PRE-SCHOOL AGE PROGRAM

นางสาวพรศิริ สุนทรธรรมกุล รหัส 41360389  
นายสมชาติ คงเกต รหัส 41360496  
นายสมภพ มิ่งกลิ่น รหัส 41360504

ห้องสมุดคณะวิศวกรรมศาสตร์  
วันที่รับ... 30 พ.ย. 2544  
เลขทะเบียน... รศ. 4400608  
เลขเรียกหนังสือ... LB  
มหาวิทยาลัยนเรศวร 1028.5  
นเรศวร

15094077. e.2  
รศ.  
นเรศวร  
2544

๒๕๔๔ ค. ๒

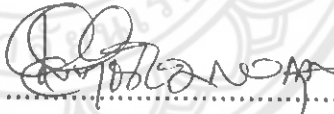
ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์  
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร  
ปีการศึกษา ๒๕๔๔

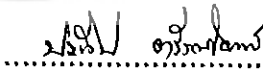


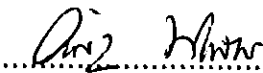
## ใบรับรองโครงการวิจัย


หัวข้อโครงการ      โปรแกรมพัฒนาการเรียนรู้เด็กก่อนวัยเรียน  
ผู้ดำเนินโครงการ      นางสาวพรศิริ      สุนทรธรรมกุล รหัส 41360389  
   นายสมชาติ      คงเกตุ      รหัส 41360496  
   นายสมภพ      มิ่งกลิ่น      รหัส 41360504  
อาจารย์ที่ปรึกษา      อาจารย์สถิธิโชค      เขาวกุล  
สาขา      วิศวกรรมคอมพิวเตอร์  
ภาควิชา      วิศวกรรมไฟฟ้าและคอมพิวเตอร์  
ปีการศึกษา      2544

.....  
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยรัตนนคร อนุมัติให้โครงการฉบับนี้เป็นส่วนหนึ่งของ  
การศึกษาดำเนินหลักสูตร วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์  
คณะกรรมการสอบโครงการวิจัย

  
.....ประธานกรรมการ  
(อาจารย์สถิธิโชค เขาวกุล)

  
.....กรรมการ  
(อาจารย์ประทีป ศิริธรรโสภาส)

  
.....กรรมการ  
(อาจารย์วัชรวิทย์ พิษพันธ์)

  
.....กรรมการ  
(อาจารย์ศิริพร เดชะศิริรักษ์)

หัวข้อโครงการ	โปรแกรมพัฒนาการเรียนรู้เด็กก่อนวัยเรียน		
ผู้ดำเนินโครงการ	นางสาวพรศิริ	สุนทรธรรมกุล	รหัส 41360389
	นายสมชาติ	คงเกตุ	รหัส 41360496
	นายสมภพ	มีงกลิ่น	รหัส 41360504
อาจารย์ที่ปรึกษา	อาจารย์สิทธิโชค เชาวกุล		
สาขา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2544		

### บทคัดย่อ

โครงการนี้เป็นการศึกษาและพัฒนาการเรียน โปรแกรมเพื่อส่งเสริมพัฒนาการของเด็กก่อนวัยเรียน โดยนำเสนอในรูปแบบของเกมการศึกษา และมีแบบทดสอบพัฒนาการของเด็ก เพื่อให้เด็กได้เกิดการเรียนรู้อย่างอิสระ ก้าวหน้าไปตามอัตราการเรียนรู้ของตน พัฒนาการที่เลือกมาพิจารณาและสร้างให้อยู่ในรูปแบบของเกมและแบบทดสอบคือ พัฒนาการทางด้านสติปัญญา (IQ) ซึ่งเป็นพัฒนาการที่สำคัญและจำเป็นเป็นอย่างยิ่งก่อนที่จะเข้าสู่ชั้นประถมศึกษา โดยจะทำการวัดสมรรถภาพทางด้านทักษะพื้นฐานคณิตศาสตร์ของเด็กอายุ 3-6 ปี และมีการประเมินผลการทดสอบพัฒนาการของเด็ก การพัฒนาโปรแกรมนี้ทดสอบและรันบนระบบปฏิบัติการไมโครซอฟท์วินโดวส์ 98 ใช้ MS Visual Basic version 6.0 ในการพัฒนา และอ้างอิงการใช้งานฟังก์ชัน API และ GDI ในการสร้างภาพกราฟิกและภาพเคลื่อนไหวต่าง ๆ บนวินโดวส์

ผลที่ได้จากการทำโครงการนี้ คือ ได้โปรแกรมพัฒนาการเรียนรู้เด็กก่อนวัยเรียน ในลักษณะของเกมการศึกษา ที่มีรูปลักษณะสวยงามน่าติดตาม เหมาะสมกับวัยเด็ก ตัวโปรแกรมจะประกอบไปด้วย ส่วนของการเก็บข้อมูลส่วนตัว ข้อมูลในการเล่น เกมฯ ในการประเมิน ผลการประเมิน โดยสามารถนำโปรแกรมนี้ไปใช้ประกอบการเรียนการสอน และเป็นกิจกรรมเสริมในโรงเรียนอนุบาลได้

**Project Title** A Development of Pre-School Age Program  
**Name** Miss Pornsiri Suntornammakul ID. 41360389  
Mr. Somchat Konggate ID. 41360496  
Mr. Sompob Mingklin ID. 41360504  
**Project Advisor** Mr. Sitthichock Chaovakul  
**Major** Computer Engineering  
**Department** Electrical and Computer Engineering  
**Academic Year** 2001

.....

### ABSTRACT

This project is the study and the development of programming for pre-school children. This program is in a form of game which has the test to measure a development of children. For children has a free learning and has an advance rate of their learning. The development that choose and create are from I.Q.(Intelligence Quautant) teat that is the neccessary and very important for children before study in school. The measurement will use the basic of mathemetics devide for 3-6 years old and there are the assessment of the development of children. This program run and test on Microsoft Windows 98 operating system , use Microsoft Visual basic version 6.0 and use the function of API and GDI for create graphic design and animation on windows.

The result from this project is a development of pre-school children program. This program is in form of education game that has many feature suitable for children. The program consist of private data saving , game operating data , standard measuring , result of measurement. It able to use this program for activity in nursary.

## กิตติกรรมประกาศ

ขอขอบพระคุณทุกๆท่านที่ทำให้โครงการชิ้นนี้ ประกอบกันจนเสร็จสิ้น โดยเฉพาะ อาจารย์สิทธิโชค เชาวกุล ที่เป็นผู้ให้คำปรึกษาระหว่างทำโครงการชิ้นนี้ และขอบใจเพื่อนๆ และน้องๆ ทุกคนที่มาช่วยงานให้สำเร็จลุล่วงไปได้ด้วยดี



นางสาวพรศิริ สุนทรธรรมกุล

นายสมชาติ กงเกตุ

นายสมภพ มิ่งกลิ่น

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ก
บทคัดย่อภาษาอังกฤษ.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ฉ
สารบัญรูป.....	ช
<b>บทที่ 1 บทนำ</b>	
1.1 ที่มาและความสำคัญของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	2
1.3 ขอบข่ายของโครงการ.....	2
1.4 ขั้นตอนการดำเนินงาน.....	3
1.5 ผลที่คาดว่าจะได้รับ.....	3
1.6 งบประมาณที่ใช้.....	3
<b>บทที่ 2 ทฤษฎีพัฒนาการและ โปรแกรมที่เกี่ยวข้อง</b>	
2.1 ทฤษฎีพัฒนาการ.....	4
2.2 หลักการของ Windows API (Application Programming Interface).....	5
2.2.1 ไฟล์ไลบรารี .dll ของวินโดวส์ 32 บิต.....	5
2.2.2 การประกาศฟังก์ชัน Windows API 32 บิต.....	5
2.3 หลักการของ GDI (Graphic Device Interface).....	8
2.3.1 การใช้งาน Device Context ของวินโดวส์.....	8
2.3.2 การสร้างและใช้งาน Device Context.....	9
2.3.3 การสร้าง DC คอมแพททิเบิลกับฟอรัมหรือคอนโทรล PictureBox.....	10
2.3.4 ระบบพิกัดของหน้าจอคอมพิวเตอร์.....	13
2.3.5 การใช้งานคำสั่ง PaintPicture.....	13
2.3.6 Raster Operation.....	15
2.3.7 สไปรต์และภาพมาสก์ (Sprite & Mask).....	16

## สารบัญ (ต่อ)

	หน้า
<b>บทที่ 3</b> การสร้างโปรแกรมพัฒนาการเรียนรู้เด็กก่อนวัยเรียน	
3.1 ซอฟต์แวร์ที่ใช้ในการพัฒนาโปรแกรม.....	17
3.2 คุณลักษณะของซอฟต์แวร์.....	17
3.3 โครงสร้างของโปรแกรมพัฒนาการเรียนรู้เด็กก่อนวัยเรียน.....	18
<b>บทที่ 4</b> การใช้งานโปรแกรมพัฒนาการเรียนรู้เด็กก่อนวัยเรียน	
4.1 เมนูหลัก.....	21
4.2 เกมส์.....	22
4.3 แบบทดสอบ.....	23
4.4 ช่วยเหลือ.....	24
<b>บทที่ 5</b> บทสรุป	
5.1 สรุปผล.....	25
5.2 ปัญหาและแนวทางแก้ไข.....	25
5.3 แนวทางในการพัฒนาต่อ.....	25
เอกสารอ้างอิง.....	26
ภาคผนวก.....	27
ประวัติผู้เขียน.....	31

# สารบัญตาราง

ตารางที่	หน้า
1.1 ขั้นตอนและระยะเวลาการดำเนินงาน.....	3
2.1 ค่าตัวแปรคงที่สำหรับการทำ Raster Operation ของ Visual Basic.....	15





# สารบัญรูป

รูปที่	หน้า
2.1 แสดงกล่องคำ Device Context (DC).....	9
3.1 โครงสร้างโดยรวมของโปรแกรม.....	18
3.2 โครงสร้างของโปรแกรมพัฒนาการเรียนรู้เด็กก่อนวัยเรียน.....	19
4.1 หน้าจอหลักของโปรแกรม.....	21
4.2 หน้าจอเกมส์.....	22
4.3 หน้าจอแบบทดสอบ.....	23
4.4 หน้าจอรับข้อมูลของผู้เล่น.....	23
4.5 แสดงผลการประเมินการทดสอบ.....	24
4.6 หน้าจอระบบช่วยเหลือ.....	24



## บทที่ 1

### บทนำ

#### 1.1 ที่มาและความสำคัญของโครงการ

การจัดการเรียนการสอนเด็กปฐมวัยหรือเด็กก่อนวัยเรียน (เด็กที่อยู่ในช่วงอายุ 3-6 ปี) เป็น การสอนเพื่อเตรียมความพร้อมของเด็ก ซึ่งจำเป็นต้องมีการจัดกิจกรรมในกระบวนการสอน การจัดกิจกรรมต่าง ๆ นั้นล้วนมีจุดมุ่งหมายเพื่อพัฒนาเด็กให้มีความเจริญเติบโตทางด้านร่างกาย อารมณ์ จิตใจ สังคม และพัฒนาสติปัญญา โดยเฉพาะกิจกรรมการเล่นทางสร้างสรรค์ ทั้งนี้ เพื่อให้เด็กก่อนวัยเรียนได้เตรียมความพร้อมก่อนเข้าเรียนในชั้นประถมศึกษาปีที่ 1

ทางผู้เสนอโครงการจึงมีความประสงค์ที่จะจัดทำโปรแกรมพัฒนาการเรียนรู้เด็กก่อน วัยเรียนขึ้นมา อาศัยแนวคิดและหลักการเกี่ยวกับพัฒนาการ การเรียนรู้ การเล่น วัฒนธรรม และ สังคม พัฒนาการที่เราสนใจก็คือพัฒนาการทางด้านสติปัญญา (IQ) โดยใช้ทักษะพื้นฐานทาง คณิตศาสตร์ ซึ่งเป็นพื้นฐานที่สำคัญอย่างยิ่งในการพร้อมที่จะเข้าเรียนร่วมกับผู้อื่นในโรงเรียน ชั้นประถมศึกษาได้ และจากกิจกรรมการเล่นของเด็ก พบว่ากิจกรรมที่ทำให้เด็กเกิดความสุขใจ มีความสุขเล่นได้ยาวนาน ซึ่งจะเชื่อมโยงไปสู่การเล่นอย่างอื่น ๆ ซึ่งจะพัฒนาความคิด ความรู้และ ความเข้าใจของเด็กได้ ด้วยเหตุนี้จึงนำมาเป็นแนวทางในการสร้างเกมคอมพิวเตอร์ ที่มีภาพลักษณ์ สวยงามน่าติดตามเหมาะกับช่วงอายุของเด็ก เพื่อให้เด็กได้มีการเรียนรู้อย่างอิสระ ก้าวหน้าไปตาม อัตราการเรียนรู้ของตน

ลักษณะของโปรแกรมที่สร้างขึ้นแบ่งออกเป็น 2 ส่วนคือ ส่วนของเกม และส่วนของแบบ ทดสอบ ซึ่งในส่วนของเกมจะมีบทเรียนแทรกอยู่ด้วย ในส่วนของแบบทดสอบจะมีการเก็บผลการ ทดสอบแต่ละครั้งไว้ เมื่อเด็กเข้ามาทดสอบอีกครั้ง ก็จะสามารถประเมินได้ว่าเด็กมีพัฒนาการไป ในทางที่ดีขึ้นหรือแย่ลง ทั้งนี้เพื่อเป็นประโยชน์ต่อครูผู้สอน ได้นำผลที่ได้จากโปรแกรมนี้อำนาจ ในการประเมินพัฒนาการทางด้านสติปัญญาของเด็กแต่ละคนว่ามีปัญหาอย่างไรบ้าง หรือสำหรับผู้ ปกครอง ก็สามารถนำโปรแกรมนี้ไปใช้กับลูกหลานของตนได้ โดยสามารถนำข้อมูลนี้ไปช่วยใน การพัฒนาเด็กให้เต็มตามความสามารถหรือแก้ไขได้ทันที่

## 1.2 วัตถุประสงค์ของโครงการ

- 1.2.1 เพื่อความรู้ความเข้าใจ และพัฒนาทักษะด้านการเขียน โปรแกรมให้แก่ผู้พัฒนา
- 1.2.2 เพื่อเป็นซอฟต์แวร์ประยุกต์ที่สามารถนำไปใช้ประโยชน์ในการศึกษา
- 1.2.3 เพื่อช่วยให้เด็กเกิดทักษะ พัฒนาการเรียนรู้ และ เกิดความเพลิดเพลิน
- 1.2.4 เพื่อช่วยให้ครู ผู้ปกครองสามารถติดตาม และใช้ประเมินผลการเปลี่ยนแปลงทางพัฒนาการของเด็กก่อนวัยเรียนได้โดยง่าย

## 1.3 ขอบข่ายของโครงการ

โปรแกรมนี้ทางผู้จัดทำได้ออกแบบเป็น 3 ส่วน คือ เกม แบบทดสอบ และส่วนช่วยเหลือ โดยเนื้อหาจะมุ่งเน้นไปที่พัฒนาการทางคณิตศาสตร์ และพัฒนาการพื้นฐานของเด็กก่อนวัยเรียน

1.3.1 ส่วนของเกม จะประกอบไปด้วย 4 เกม เป็นเกมเกี่ยวกับพัฒนาการต่างๆ 4 เกม

- ความสามารถในการนับ รู้ค่าจำนวน 1-10
- ความสามารถในการรู้จักสีต่างๆ
- รู้จักรูปทรงเรขาคณิต
- ความสามารถในการแบ่งขนาดของวัตถุ และสิ่งของต่างๆ

ลักษณะของเกมในแต่ละเกมก็จะมีรูปแบบแตกต่างกันไป โดยจะทำการสุ่มขึ้นมาเพื่อให้ผู้เล่นเกิดความรู้สึกลากหลาย ไม่เบื่อหน่าย

1.3.2 ส่วนของแบบทดสอบ จะเป็นการนำเกมข้างต้นมาเป็นแบบทดสอบ โดยเด็กต้องเข้าเล่นเกมตามลำดับที่กำหนดไว้ และต้องเล่นจนจบเกม โดยจะมีการเก็บข้อมูลและคะแนนของเด็กไว้ทุกครั้ง และรายงานผลพัฒนาการเทียบกับเกณฑ์พัฒนาการปกติ และสำหรับเด็กที่เคยเข้ามาทำแบบทดสอบแล้ว ก็จะมีการนำผลมาเปรียบเทียบกับเพื่อน เพื่อที่จะสามารถบอกได้ว่ามีการพัฒนาไปในทางที่ดีขึ้น หรือแย่ลงอย่างไร

## 1.4 ขั้นตอนการดำเนินงาน

ตารางที่ 1.1 ขั้นตอนและระยะเวลาการดำเนินงาน

กิจกรรม	เดือน - ปี						
	มี.ค 44	เม.ย 44	พ.ค 44	มิ.ย 44	ก.ค 44	ส.ค 44	ก.ย 44
1. ออกแบบในส่วน ของโครงสร้างของ โปรแกรม	←→						
2. รวบรวมข้อมูล ภาพ เสียงที่เกี่ยวข้อง			←→				
3. เขียนโปรแกรม				←→			
4. แก้ไขข้อผิดพลาด						←→	
5. สรุปผล							←→
6. จัดทำรูปเล่ม							←→

## 1.5 ผลที่คาดว่าจะได้รับ

- 1.5.1 ได้รับความรู้ความเข้าใจ และพัฒนาทักษะด้านการเขียนโปรแกรม โดยใช้ MS Visual Basic ให้แก่ผู้พัฒนา
- 1.5.2 ได้ซอฟต์แวร์ประยุกต์ที่สามารถนำไปใช้ประโยชน์ในการศึกษา
- 1.5.3 เพื่อช่วยให้เด็กเกิดทักษะ พัฒนาการเรียนรู้ และ เกิดความเพลิดเพลิน

## 1.6 งบประมาณที่ใช้ นิติต : คน : 1,000 บาท

1.6.1	กระดาษ A4	3 รีม	ราคา 500 บาท
1.6.2	ดรัมหมึกปริ้นเตอร์	1 กล่อง	ราคา 1,000 บาท
1.6.3	หนังสือการเขียนโปรแกรม	2 เล่ม	ราคา 1,000 บาท
1.6.4	แผ่น CD-R	5 แผ่น	ราคา 200 บาท
1.6.5	ค่าถ่ายเอกสาร		ราคา 300 บาท
รวมเป็นเงินทั้งสิ้น			3,000 บาท

## บทที่ 2

# ทฤษฎีพัฒนาการ และ โปรแกรมที่เกี่ยวข้อง

### 2.1 ทฤษฎีพัฒนาการ [1]

ทฤษฎีพัฒนาการของเด็กก่อนวัยเรียนตามหลักสูตรก่อนประถมศึกษา ปี พ.ศ. 2540 ที่ได้ศึกษามาพบว่า พัฒนาการปกติที่สำคัญของเด็กก่อนวัยเรียนมีดังนี้

พัฒนาการทางด้านร่างกาย สามารถเคลื่อนไหว เดิน วิ่งได้อย่างคล่องแคล่ว มีความกระฉับกระเฉง ไม่อยู่เฉย สามารถวาดภาพและระบายสีได้ คัดกระดาษ และใช้ก้ำมือน้อยเด็กได้ดี สามารถช่วยเหลือตนเองได้เช่น ตัดกระดาษ ผูกเชือกทรงเท้า

พัฒนาการทางด้านสติปัญญา มีความสนใจอยากรู้หรืออยากเห็นเกี่ยวกับเรื่องราวต่างๆรอบตัว สนใจนิทาน เรื่องราวหรือเหตุการณ์ต่างๆ สามารถเรียงลำดับเหตุการณ์ก่อนหลังได้ สามารถพูดเป็นประโยคได้ ท่องจำคำคล้องจองได้ รู้คำศัพท์ที่ใช้ในชีวิตประจำวัน สามารถนับเลข 1-20 ได้ สามารถเปรียบเทียบ จำแนกและจัดหมวดหมู่สิ่งของได้ บอกความแตกต่างของสี เสียง และรูปทรงเรขาคณิตเบื้องต้นได้

พัฒนาการทางอารมณ์และจิตใจ เด็กจะชอบเล่น ชอบแสดงออก และสามารถช่วยเหลือตัวเองได้

จากพัฒนาการพื้นฐานเหล่านี้ พัฒนาการที่สนใจนำมาสร้างเป็นเกม และแบบทดสอบก็คือ พัฒนาการทางด้านสติปัญญา (IQ) โดยใช้ทักษะพื้นฐานทางคณิตศาสตร์ ซึ่งเป็นพื้นฐานที่สำคัญอย่างยิ่งในการพร้อมที่จะเข้าเรียนร่วมกับผู้อื่นในโรงเรียนชั้นประถมศึกษาได้ ซึ่งประกอบไปด้วยทักษะต่อไปนี้

- ความสามารถในการนับ รู้ค่าจำนวน 1-10
- ความสามารถในการรู้จักสีต่างๆ
- รู้จักรูปทรงเรขาคณิต
- ความสามารถในการแบ่งขนาดของวัตถุ และสิ่งของต่างๆ

เกณฑ์ที่ใช้ในการวัดความสามารถของเด็กมีอยู่ 3 ระดับ คือ สูงกว่าเกณฑ์ (ยอดเยี่ยม) ตามเกณฑ์ (ปกติ) ต่ำกว่าเกณฑ์ (ต้องแก้ไข) โดยในแต่ละเกมจะมีเกณฑ์แตกต่างกันออกไปตามลักษณะ เช่น ความสามารถในการนับ เด็กก็จะต้องสามารถนับจำนวนระหว่าง 1-10 ได้ หรือในเรื่องของสี เด็กก็จะต้องรู้สีอย่างน้อย 5 สี เป็นต้น และถ้าทำแบบทดสอบแล้วได้ผลว่าต่ำกว่าเกณฑ์ทางด้านใด ก็จะเป็นตัวชี้บอกครูอนุบาลและผู้ปกครอง ได้เสริมสร้างและเพิ่มพูนความสามารถในสมรรถภาพด้านนั้น ๆ เพื่อให้เกิดความมั่นใจว่าเด็กจะมีสมรรถภาพด้านทักษะพื้นฐานคณิตศาสตร์ครบถ้วนสำหรับที่จะเรียนในขั้นสูงต่อไป

## 2.2 หลักการของ Windows API (Application Programming Interface)

### Windows API [2-3]

API เป็นฟังก์ชันที่จัดเตรียมไว้สำหรับให้ Application ติดต่อเข้าไปเรียกใช้ เพื่อทำงานต่างๆ ได้ เช่น การวาดรูป การบันทึกข้อมูลลงในไฟล์ การเตรียมพื้นที่ในหน่วยความจำ ภายใน API จะประกอบไปด้วยฟังก์ชัน และ Procedure ต่างๆ มากมาย โดยจัดแบ่งออกเป็นกลุ่ม เช่น กลุ่มฟังก์ชันและ Procedure ที่เกี่ยวข้องกับการจัดการภาพกราฟิก กลุ่มฟังก์ชันและ Procedure ที่เกี่ยวข้องกับการจัดการด้านระบบเครือข่าย กลุ่มฟังก์ชันและ Procedure ที่เกี่ยวข้องกับการจัดการด้านมัลติมีเดีย โดยฟังก์ชันต่างๆ นี้จะอยู่ในแฟ้มข้อมูลนามสกุล \*.DLL (Dynamic Link Library)

#### 2.2.1 ไฟล์ไลบรารี .dll ของวินโดวส์ 32 บิต

ไฟล์ไลบรารี .dll หรือที่เรียกว่า Dynamic Link Library เป็นไฟล์ที่โดยปกติจะถูกกำหนดให้มีนามสกุล .dll และจะถูกจัดเก็บเอาไว้ในไดเรกทอรีย่อย \System ของวินโดวส์ และในความเป็นจริงการทำงานของวินโดวส์ก็ต้องอาศัยฟังก์ชันวินโดวส์ API ที่มีในแต่ละไฟล์ไลบรารี .dll นั้นเอง ในที่นี้ เราจะกล่าวเฉพาะไฟล์ Gdi32.dll โดยไฟล์ Gdi32.dll เป็นไลบรารีสำหรับการติดต่อสื่อสารกับดีไวซ์กราฟิก (Graphic Device Interface) ซึ่งประกอบด้วยฟังก์ชันต่าง ๆ สำหรับงานด้านอุปกรณ์แสดงผล การวาดกราฟิก ไฟล์เมต้า (.wmf) ฟอนต์ และการกำหนดตำแหน่งโคออร์ดิเนต

โดยปกติไฟล์ Dynamic Link Library ของวินโดวส์จะมีนามสกุล .dll แต่สำหรับ Visual Basic 6.0 ก็สนับสนุนไฟล์ไลบรารีชนิดโคเนนามิกส์เช่นกันแต่จะมีรูปแบบของไฟล์ที่แตกต่างไปจากไฟล์ไลบรารีโคเนนามิกส์มาตรฐาน และจะถูกกำหนดให้มีนามสกุล .ocx

#### 2.2.2 การประกาศฟังก์ชัน Windows API

ก่อนที่เราจะสามารถใช้งานฟังก์ชันวินโดวส์ API ในการเขียนโปรแกรม เราจะต้องทำการประกาศฟังก์ชันนั้น ๆ เสียก่อน เพื่อเป็นการกำหนดโปรโตไทป์ (prototype) ในการติดต่อระหว่างฟังก์ชันวินโดวส์ API กับ Visual Basic 6.0 การประกาศฟังก์ชันใน Dynamic Link Library นี้ Procedure จะต้องกระทำในระดับโมดูล (Module level) สำหรับการประกาศฟังก์ชันวินโดวส์ API โดยมีหลักการประกาศได้ดังนี้ [3-4]

**[Public/Private] Declare Function *functionname* Lib "*libname*" [Alias] "*aliasname*"**  
**(arguments) [As type]**

**Public/Private** เป็นคำกำหนดขอบเขตของ Procedure ว่าเป็นแบบ Public หรือ Private

<b>Declare Function</b>	เป็นคีย์เวิร์ดใน Visual Basic เพื่อกำหนดว่าเป็นการประกาศ Procedure ใน Dynamic Link Library
<i>functionname</i>	คือชื่อของฟังก์ชันที่ต้องการประกาศ
<b>Lib</b>	เป็นคีย์เวิร์ดในการประกาศ Procedure ใน Dynamic Link Library เพื่อกำหนดชื่อไลบรารี
<i>libname</i>	เป็นชื่อของไลบรารีที่บรรจุ Procedure ที่ต้องการประกาศ โดยไม่ต้องระบุนามสกุลของไฟล์ เพราะ Visual Basic จะรับรู้โดยอัตโนมัติ
<b>Alias</b>	เป็นคีย์เวิร์ดในการประกาศ Procedure ใน Dynamic Link Library เพื่อกำหนดชื่อเสมือนในการเรียกใช้งาน
<i>aliasname</i>	เป็นชื่อเสมือนที่ Visual Basic จะมองว่าเป็นชื่อเดียวกันกับชื่อฟังก์ชัน
<i>arguments</i>	เป็นรายชื่อและรายละเอียดของอาร์กิวเมนต์แต่ละตัวของ Procedure
<b>As type</b>	เป็นตัวกำหนดชนิดข้อมูลของค่าคืนกลับของ Procedure

ซึ่งฟังก์ชันและ Procedure ในแต่ละกลุ่มนี้จะถูก Application ต่างๆ เรียกไปใช้งาน จะขึ้นอยู่กับว่า Application นั้นทำงานเกี่ยวข้องกับอะไร ต้องการใช้งานฟังก์ชันและ Procedure ใดบ้าง ในส่วนของโครงงานนี้ได้ใช้ฟังก์ชัน Windows API คือ

- การจัดการเกี่ยวกับระบบ เคอร์เซอร์ เมาส์
- การจัดการภาพกราฟฟิบบนวินโดวส์
- การเรียกใช้มัดคิมิเดีย

**การจัดการเกี่ยวกับระบบ** ฟังก์ชันและ Procedure ที่เรียกใช้ได้แก่

#### **DrawIcon**

<b>การประกาศ</b>	Declare Function DrawIcon Lib "user32" (ByVal hdc As Long, ByVal x As Long, ByVal y As Long, ByVal hIcon As Long) As Long
<b>ลักษณะการทำงาน</b>	ใช้วาด icon ลงบนตำแหน่งที่กำหนด

#### **ReleaseCapture**

<b>การประกาศ</b>	Declare Function ReleaseCapture Lib "user32" Alias "ReleaseCapture" () As Long
<b>ลักษณะการทำงาน</b>	ยกเลิกการจับตำแหน่งเมาส์ของโปรแกรมประยุกต์ที่กำลังทำงาน

**SetCapture****การประกาศ**

Declare Function SetCapture Lib "user32" Alias "SetCapture" (ByVal hwnd As Long) As Long

**ลักษณะการทำงาน**

ดำเนินการจับตำแหน่งเมาส์ของวินโดว์ที่กำหนด วินโดว์ดังกล่าวนี้จะเป็นตัวรับอินพุตจากเมาส์ของโปรแกรมประยุกต์นั้นๆ หรือถ้าปุ่มเมาส์ถูกกดค้างเอาไว้ ระบบทั้งระบบจะเป็นตัวรับอินพุตจากเมาส์

**ShowCursor****การประกาศ**

Declare Function ShowCursor Lib "user32" Alias "ShowCursor" (ByVal bShow As Long) As Long

**ลักษณะการทำงาน**

ใช้สำหรับควบคุมการแสดงตัวของตัวชี้เมาส์ (cursor)

**การเรียกใช้มัลติมีเดีย** ในโปรแกรมประเภทมัลติมีเดีย เช่น โปรแกรมช่วยสอน , เกมส์ , โปรแกรมประเภทชาวค้ต่างๆ ฟังก์ชันและ Procedure ที่เรียกใช้ได้แก่

**SndPlaySound****การประกาศ**

Public Declare Function sndPlaySound Lib "winmm.dll" Alias "sndPlaySound" (ByVal lpzSoundName As String, ByVal uFlags As Long) As Long

**ลักษณะการทำงาน**

ฟังก์ชันนี้อาจใช้เล่นไฟล์ WAV ต้องประกาศไว้ในโมดูลไฟล์ของ VB

**mciSendString**

Public Declare Function mciSendString Lib "winmm.dll" Alias "mciSendStringA" (ByVal lpstrCommand As String, ByVal lpstrReturnString As String, ByVal uReturnLength As Long, ByVal hwndCallback As Long) As Long

**ลักษณะการทำงาน**

ใช้เปิดไฟล์ MIDI โดยมากในไฟล์เพลงประเภทคาราโอเกะจะใช้กันคือไฟล์ที่ได้จากการสังเคราะห์เสียงมาจากซินติไซเซอร์หรือคอมพิวเตอร์ ที่มีโปรแกรมสร้างเสียงประเภทใช้แต่งเพลง โดยจะมีขนาดไฟล์ไม่ใหญ่มาก โดยเราจะสร้างซับโปรแกรมตัวนี้ชื่อ PlayMIDI



## 2.3 หลักการของ GDI (Graphic Device Interface) [3-4]

GDI คือ ไฟล์ไลบรารีสำหรับการสร้างภาพกราฟิกบนวินโดวส์ สามารถจัดการกับภาพกราฟิกที่จะนำไปแสดงยังหน้าจอคอมพิวเตอร์ โดยเรียกใช้งานจาก gdi.exe ซึ่งเป็น Dynamic Link Library อันหนึ่งที่มาพร้อมกับวินโดวส์ทุกเวอร์ชัน

### 2.3.1 การใช้งาน Device Context ของวินโดวส์

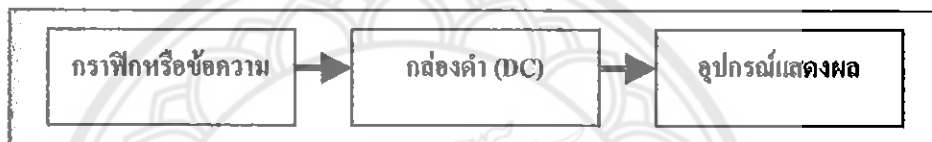
ในส่วนนี้จะกล่าวถึงออบเจกต์ของวินโดวส์ที่มีความสำคัญในด้านของการแสดงผล เนื่องจากวินโดวส์มีการแสดงผลทุกอย่างในรูปแบบของกราฟิก ดังนั้น วินโดวส์จึงต้องมีการออกแบบส่วนที่จะช่วยในการแสดงผลกราฟิกอย่างมีประสิทธิภาพ และสามารถใช้งานได้กับฮาร์ดแวร์ที่มีความละเอียดแตกต่างกันได้ โดยไม่จำเป็นต้องมีการเขียนโค้ดเฉพาะสำหรับฮาร์ดแวร์ชนิดเดียวกัน แต่มีความละเอียดแตกต่างกัน ซึ่งออบเจกต์ของวินโดวส์ดังกล่าวถูกเรียกว่า Device Context (DC) ซึ่งการเขียนโปรแกรมด้วย Visual Basic 6.0 ซึ่งจะสามารถอาศัยฟังก์ชันวินโดวส์ API ในกลุ่มของ DC เพื่อช่วยในการเพิ่มความเร็วหรือเพิ่มศักยภาพด้านกราฟิก และสามารถที่จะเข้าถึง DC ทั้งหมดที่วินโดวส์สนับสนุนได้โดยไม่มีขีดจำกัด

การวาดรูปหรือแสดงผลตัวอักษรใด ๆ ก็ตามภายใต้ระบบปฏิบัติการวินโดวส์ จะต้องถูกกระทำผ่านทาง Device Context (DC) ซึ่งเป็นออบเจกต์ตัวหนึ่งของวินโดวส์ที่ช่วยในการจัดการในด้านการจัดการกราฟิก เช่น การจัดขนาด การกำหนดขอบเขตพื้นที่การแสดงผล การกำหนดขนาดของเส้นหรือกราฟิก เป็นต้น เพื่อให้การเขียนโค้ดด้านกราฟิกสำหรับวินโดวส์มีความสะดวกมากขึ้น เนื่องจากวินโดวส์สนับสนุนอุปกรณ์แสดงผล (output device) ที่มีความละเอียดแตกต่างกัน ดังนั้น ถ้าหากไม่มีการออกแบบส่วนหรือองค์ประกอบของวินโดวส์ที่จะช่วยในการแปลงระบบโคออร์ดิเนต (ขนาดความกว้าง หรือ ความยาวของอุปกรณ์แสดงผล) ให้อย่างอัตโนมัติ ก็จะทำให้การเขียนโค้ดมีความยุ่งยากมากขึ้น

ออบเจกต์ DC ของวินโดวส์มีความสามารถที่สำคัญ 2 ประการ ที่ช่วยให้การแสดงผลกราฟิกไม่เกิดความยุ่งยาก คือ การตัดขอบ (clipping) และการแปลงระบบโคออร์ดิเนต (coordinate transformation) ดังนี้

**การตัดขอบ (clipping)** เป็นกระบวนการที่ถูกวินโดวส์เรียกใช้ทุกครั้งที่มีการแสดงผลกราฟิกหรือข้อความออกทางอุปกรณ์แสดงผลต่างๆ เพื่อตรวจสอบส่วนของการแสดงผลเทียบกับพื้นที่แสดงผล โดยถ้าหากส่วนของการแสดงผลอยู่ภายในพื้นที่แสดงผล วินโดวส์ก็จะแสดงผลแบบจุดต่อจุด (pixel - by - pixel) พื้นที่ แต่ถ้าหากส่วนของการแสดงผลอยู่นอกพื้นที่แสดงผล วินโดวส์ก็จะไม่ทำการแสดงผลส่วนดังกล่าว จึงทำให้การแสดงผลของแต่ละพื้นที่ที่ถูกจำกัดขนาด จะไม่ทับกับการแสดงผลของพื้นที่อื่นๆ โดยจะเห็นได้จากการที่วินโดวส์แสดงผลในแต่ละหน้าต่างอย่างเอกเทศ ถึงแม้ว่าหน้าต่างจะมีส่วนซ้อนทับกัน แต่การแสดงผลในแต่ละหน้าต่างจะไม่ทับกัน

การแปลงระบบโคออร์ดิเนต (coordinate transformation) หมายถึง การคำนวณด้านคณิตศาสตร์ เพื่อแปลงตำแหน่งการแสดงผลกราฟิก ระหว่างระบบโคออร์ดิเนตของอุปกรณ์แสดงผล และระบบโคออร์ดิเนตที่ถูกเลือก (วินโดวส์สามารถสนับสนุนระบบโคออร์ดิเนตที่หลากหลาย ซึ่งเรียกว่า Logical coordinate) คุณสมบัติการแปลงระบบโคออร์ดิเนตของ DC นี้เองที่ถือได้ว่าเป็นหัวใจของการแสดงผลกราฟิกออกทางอุปกรณ์แสดงผลที่มีความละเอียดต่างกันของวินโดวส์ ดังนั้น DC จึงมีลักษณะคล้ายๆกับกล่องดำ (black box) สำหรับการแสดงผลด้านกราฟิก โดยที่กราฟิกที่ต้องการแสดงผลก็จะถูกส่งมายังกล่องดำ DC เพื่อทำการประมวลผล เช่น การตัดขอบหรือการแปลงระบบโคออร์ดิเนต เป็นต้น เพื่อให้ตรงกับอุปกรณ์แสดงผลที่กำหนด จากนั้นจึงนำผลลัพธ์ที่ได้ไปแสดงผลออกทางอุปกรณ์แสดงผลที่กำหนด ดังแสดงรูปที่ 2.1



รูปที่ 2.1 แสดงกล่องดำ Device Context (DC)

### 2.3.2 การสร้างและใช้งาน Device Context

วินโดวส์ได้เตรียม DC มาตรฐานอยู่ภายใน (build-in) หน่วยความจำที่เรียกว่า pool เพื่อให้สามารถใช้ในการวาดกราฟิกลงในพื้นที่แสดงผลของหน้าต่างหรือดีไวซ์ต่าง ๆ ได้ โดยถ้าหากเราต้องการวาดกราฟิกลงในพื้นที่ของหน้าต่างหรือดีไวซ์ เราก็ต้องอ้างอิงถึง DC ตัวใดตัวหนึ่งจาก pool และกำหนดค่าแอตทริบิวต์ตามที่ต้องการ เช่น ขนาดหรือสีของเส้น เป็นต้น ให้กับ DC จากนั้นจึงเรียกใช้ฟังก์ชันวินโดวส์ API ด้านกราฟิกเพื่อวาดกราฟิก และเมื่อเราไม่ต้องการใช้ DC อีกต่อไป ก็ต้องทำการยกเลิกการถือครอง DC ดังกล่าว เพื่อให้ DC ที่ถูกยกเลิกการถือครอง ได้ถูกส่งกลับไปยัง pool และเป็น DC ที่ว่างสำหรับถูกเรียกใช้งาน โดยโปรแกรมประยุกต์ตัวอื่น ๆ ต่อไป และนอกจากนี้ ถ้าหากเราไม่พอใจกับ DC มาตรฐานที่วินโดวส์ได้เตรียมไว้ให้ เราก็สามารถที่จะสร้าง DC ขึ้นมาใหม่ ให้เข้ากันได้กับ DC มาตรฐาน เพื่อนำมาใช้ส่วนตัวสำหรับแต่ละโปรแกรมประยุกต์หนึ่ง ๆ ก็ได้เช่นกัน

ออบเจกต์ DC ไม่ว่าจะ เป็น DC มาตรฐานหรือที่ถูกสร้างขึ้นมาโดยฟังก์ชันวินโดวส์ CreateDC ก็ตาม จะต้องมีหมายเลข handle ประจำตัวเสมอ เพื่อให้สามารถใช้ในการจัดการเรียกใช้ DC โดยฟังก์ชันวินโดวส์ API ได้อย่างถูกต้อง สำหรับตัวของออบเจกต์ DC ก็เป็นข้อมูลโครงสร้าง

ภายในของวินโดวส์ที่มีขนาดประมาณ 800 ไบต์ โดยที่วินโดวส์ได้เตรียมวิธีการที่หลากหลายในการสร้าง และ เข้าถึง DC โดยขึ้นกับลักษณะของ DC ดังนี้

1. Private Device Context
2. Cached Device Context
3. Created Device Context

Created Device Context เป็น DC ที่ถูกสร้างขึ้นใหม่โดยโค้ดของโปรแกรม ซึ่งเราสามารถสร้าง DC สำหรับโปรแกรมประยุกต์ต่างๆ ได้โดยใช้ฟังก์ชันวินโดวส์ CreatDC หรือ CreatCompatibleDC โดยในทางทฤษฎีเราสามารถสร้าง DC ประเภทนี้ได้มากเท่าที่ต้องการหรือไม่มีข้อจำกัด แต่ในทางปฏิบัติจำนวนของ Created Device Context จะถูกจำกัดโดยทรัพยากรของระบบ โดยเฉพาะภายใต้วินโดวส์ 95 และ 98 ที่ยังคงมีข้อจำกัดในด้านของการจัดการทรัพยากรของระบบ สำหรับหมายเลขแฮนเคิล ของ Created Device Context จะถูกรายงานโดยฟังก์ชันวินโดวส์ CreatDC หรือ CreatCompatibleDC เมื่อการสร้าง Created Device Context ประสบผลสำเร็จ

### 2.3.3 การสร้าง DC คอมแพททิเบิลกับฟอร์มหรือคอนโทรล PictureBox

การสร้าง DC ที่คอมแพททิเบิลกับฟอร์มหรือคอนโทรล PictureBox ของ Visual Basic 6.0 โดยฟังก์ชันวินโดวส์ CreateCompatibleDC เพื่อทำการแสดงผลบิตแมปและไอคอนลงในคอนโทรล PictureBox หรือฟอร์ม ซึ่งจะมีฟังก์ชันวินโดวส์ API ที่เกี่ยวข้องดังนี้ [3-4]

#### BitBlt

##### การประกาศ

Declare Function BitBlt Lib "gdi32" Alias "BitBlt" (ByVal hDestDC As Long, ByVal x As Long, ByVal y As Long, ByVal nWidth As Long, ByVal nHeight As Long, ByVal hSrcDC As Long, ByVal xSrc As Long, ByVal ySrc As Long, ByVal dwRop As Long) As Long

##### ลักษณะการทำงาน

จะทำการ copy บิตแมปจากดีไวซ์คอนเท็กซ์ (DC) หนึ่ง ไปยังดีไวซ์คอนเท็กซ์ (DC) อื่นๆ โดยดีไวซ์คอนเท็กซ์ทั้งคู่จะต้องคอมแพทิเบิลกัน

##### อาร์กิวเมนต์

hDestDC      ข้อมูลชนิด Long – ค่าแฮนเคิล (ค่าตัวเลขที่กำหนดขึ้นโดยโปรแกรมวินโดวส์ สำหรับใช้อ้างอิงถึงคอนเท็กซ์สำหรับการแสดงผลต่าง ๆ) ของดีไวซ์คอนเท็กซ์เป้าหมาย

- X,Y** ข้อมูลชนิด Long – ตำแหน่งมุมบนซ้ายที่ต้องการจะวาดภาพบนดีไวซ์คอนเท็กซ์ เป้าหมาย
- Nwidth,nHeight** ข้อมูลชนิด Long – ความกว้างและความสูงของพื้นที่ในการวาดที่จะทำลงบนดีไวซ์คอนเท็กซ์เป้าหมาย
- HSrcDC** ข้อมูลชนิด Long – ค่าแฮนเคิล (ค่าตัวเลขที่กำหนดขึ้นโดยโปรแกรมวินโดวส์ สำหรับใช้อ้างอิงถึงคอนเท็กซ์ สำหรับการแสดงผลต่าง ๆ) ของดีไวซ์คอนเท็กซ์ แหล่งกำเนิด
- XSrc,ySrc** ข้อมูลชนิด Long – ตำแหน่งมุมบนซ้ายที่ต้องการจะวาดภาพบนดีไวซ์คอนเท็กซ์ แหล่งกำเนิด
- DwROP** ข้อมูลชนิด Long - ค่า Raster Operations ในการถ่ายเทข้อมูล
- ค่าคืนกลับ**  
ข้อมูลชนิด Long - หากฟังก์ชันทำงานสำเร็จจะคืนค่าที่ไม่ใช่ 0 แต่หากเกิดข้อผิดพลาดจะคืนค่า 0

### CreateCompatibleDC

#### การประกาศ

Declare Function CreateCompatibleDC Lib "gdi32" Alias "CreateCompatibleDC" (ByVal hdc As Long) As Long

#### ลักษณะการทำงาน

สร้างดีไวซ์คอนเท็กซ์ในหน่วยความจำ โดยดีไวซ์คอนเท็กซ์ที่สร้างขึ้นจะคอมแพททิเบิลกับดีไวซ์ที่กำหนด

#### จาร์กิวเมนต์

- hDC** ข้อมูลชนิด Long – ค่าแฮนเคิล (ค่าตัวเลขที่ถูกระบุขึ้นโดยโปรแกรมวินโดวส์ สำหรับใช้อ้างอิงถึงคอนเท็กซ์สำหรับการแสดงผลต่าง ๆ) ของดีไวซ์คอนเท็กซ์ที่ต้องการให้ดีไวซ์คอนเท็กซ์ที่สร้างขึ้นมีความคอมแพททิเบิลด้วยหากส่งค่า 0 จะหมายถึงดีไวซ์คอนเท็กซ์ของหน้าจอ

#### ค่าคืนกลับ

ข้อมูลชนิด Long – หากฟังก์ชันทำงานสำเร็จ จะคืนค่าแฮนเคิล (ค่าตัวเลขที่กำหนดขึ้นโดยโปรแกรมวินโดวส์ สำหรับใช้อ้างอิงถึงคอนเท็กซ์สำหรับการแสดงผลต่าง ๆ) ของดีไวซ์คอนเท็กซ์ในหน่วยความจำที่สร้างขึ้นใหม่ และจะคืนค่า 0 ถ้าเกิดข้อผิดพลาด

**หมายเหตุ** ก่อนที่จะทำการวาดภาพใด ๆ ลงบนดีไวซ์คอนเท็กซ์ จะต้องทำการเลือกข้อมูลเข้าสู่ดีไวซ์คอนเท็กซ์เสียก่อน และควรลบดีไวซ์ที่สร้างขึ้นทิ้งเมื่อไม่จำเป็นต้องใช้งานแล้ว

**DeleteDC****การประกาศ**

Declare Function DeleteDC Lib "gdi32" Alias "DeleteDC" (ByVal hdc AS Long) As Long

**ลักษณะการทำงาน**

จัดการลบดีไวซ์คอนเท็กซ์ตามค่าแฮนเดิล (ค่าตัวเลขที่กำหนดขึ้นโดยโปรแกรมวินโดวส์ สำหรับใช้อ้างอิงถึงคอนเท็กซ์สำหรับการแสดงผลต่าง ๆ) ที่ส่งเป็นอาร์กิวเมนต์ ห้ามใช้กับดีไวซ์คอนเท็กซ์ที่ได้จาก ฟังก์ชัน GetDC()

**อาร์กิวเมนต์**

**hDC** ข้อมูลชนิด Long – ค่าแฮนเดิล (ค่าตัวเลขที่กำหนดขึ้นโดยโปรแกรมวินโดวส์ สำหรับใช้อ้างอิงถึงคอนเท็กซ์สำหรับการแสดงผลต่าง ๆ) ของดีไวซ์คอนเท็กซ์ที่ต้องการจะลบออก

**ค่าคืนกลับ**

ข้อมูลชนิด Long - หากฟังก์ชันทำงานสำเร็จจะคืนค่า 1 หากเกิดความผิดพลาดก็จะคืนค่า 0

**SelectObject****การประกาศ**

Declare Function SelectObject Lib "gdi32" Alias "SelectObject" (ByVal hdc As Long, ByVal hObject As Long) As Long

**ลักษณะการทำงาน**

โดยปกติแล้วจะมี GDI Object อย่างเช่น pen, brush, font, regions, palletes หรือ bitmap ใดๆ อย่างหนึ่งถูกเลือกเข้าสู่ดีไวซ์คอนเท็กซ์เสมอ ซึ่งออบเจกต์แต่ละชนิดจะสามารถถูกเลือกเข้าสู่ดีไวซ์คอนเท็กซ์ได้เพียงชนิดละ 1 ออบเจกต์ เพื่อใช้สำหรับการวาดภาพบนดีไวซ์คอนเท็กซ์

**อาร์กิวเมนต์**

**hDC** ข้อมูลชนิด Long – ค่าแฮนเดิล (ซึ่งเป็นค่าตัวเลขที่กำหนดขึ้นโดยโปรแกรมวินโดวส์ สำหรับใช้อ้างอิงถึงคอนเท็กซ์สำหรับการแสดงผลต่าง ๆ) ของดีไวซ์คอนเท็กซ์ที่ต้องการจะเลือกออบเจกต์เข้าสู่ดีไวซ์คอนเท็กซ์

**hObject** ข้อมูลชนิด Long – ค่าแฮนเดิล (ซึ่งเป็นค่าตัวเลขที่กำหนดขึ้นโดยโปรแกรมวินโดวส์ สำหรับใช้อ้างอิงถึงคอนเท็กซ์สำหรับการแสดงผลต่าง ๆ) ของ GDI ออบเจกต์ที่จะเลือก

**ค่าคืนกลับ**

ข้อมูลชนิด Long – หากฟังก์ชันทำงานสำเร็จจะคืนค่าแฮนเดิล (ซึ่งเป็นค่าตัวเลขที่กำหนดขึ้นโดยโปรแกรมวินโดวส์ สำหรับใช้อ้างอิงถึงคอนเท็กซ์สำหรับการแสดงผลต่าง ๆ) ของ

ขอบเข็ทที่เป็นชนิดเดียวกันกับขอบเข็ทที่ถูกเลือกเข้าสู่ติวซ์คอนเท็ทซ์เมื่อก่อนหน้านี้ หากเกิดข้อผิดพลาดจะคืนค่า 0

#### 2.3.4 ระบบพิกคัของหน้าจอกอมพิวเตอร้ [4]

ในระบบกราฟฟิคนคอมพิวเตอร้ หน้าจอกอมพิวเตอร้จะถูกมองเป็นแนวระนาบที่มีแกน X และแกน Y เป็นตัวกำหนดระยะแนวนอนและแนวตั้งตามลำดับ เราจะสามารถอ้างอิงถึงตำแหน่งด้วยพิกคัหรือตำแหน่งใดๆ บนหน้าจอกอมพิวเตอร้ด้วยค่า (X,Y) โดยมีจุดมูบนซ้ายของหน้าจอกอเป็นจุดกำเนิดมีพิกคัเป็น (0,0) ระยะจากจุดกำเนิดไปตามแนวแกน X ค่า X จะเพิ่มขึ้นตามลำดับ เช่นเดียวกันระยะจากจุดกำเนิดไปตามแนวแกน Y ค่า Y จะมีค่าเพิ่มขึ้นเรื่อยๆ

สิ่งทีสำคัญในการทำความเข้าใจระบบพิกคัก็คือ หน่วย (Unit) ของพิกคั เนื่องจากว่าระบบปฏิบัติการวินโดวส์ จะต้องมีความยืดหยุ่นกับฮาร์ดแวร์ทุกชนิด ดังนั้นหน่วยพิกคัของวินโดวส์ 95/98 จึงเป็นแบบ Twip ซึ่ง 1440 Twip จะมีค่าเท่ากับความยาว 1 นิ้วบนหน้าจอกอมพิวเตอร้ หมายความว่าหากเราสั่งให้ วินโดวส์ วาดเส้นตรงยาว 1440 Twip ขึ้นบนหน้าจอกอมพิวเตอร้ ผลลัพธ์ทีเกิดขึ้นก็คือจะเกิดเส้นตรงยาว 1 นิ้วขึ้นบนหน้าจอกอเสมอ ไม่ว่าขณะนั้นความละเอียดของหน้าจอกอมพิวเตอร้จะถูกตั้งไว้เท่าใด

หน่วยพิกคัอีกชนิดหนึ่งทีมีความสำคัญไม่น้อยไปกว่า Twip ก็คือ หน่วยพิกเซล (Pixel) ซึ่งหมายถึงจุดหนึ่งจุดทีคอมพิวเตอร้จะสามารถแสดงผลขึ้นบนหน้าจอกอได้ ค่าความละเอียดในการแสดงผลของหน้าจอกอสามารถถูกปรับเปลี่ยนได้ตามความต้องการของผู้ใช้งานและความสามารถของฮาร์ดแวร์ในการแสดงผลของเครื่องนั้นๆ ยกตัวอย่างเช่น ความละเอียด 640 \* 480 จุดก็หมายความว่า มีจุดตามแนวแกน X ทีสามารถแสดงผลได้อยู่ 640 จุด และตามแนวแกน Y อยู่ 480 จุด

สิ่งทีนักเขียนเกมด้วย Visual Basic ควรจะทราบคือ

- ทุกวินโดว้ทีเราเห็นบนหน้าจอกอเป็นวินโดว้รอง (Child Windows) ของระบบปฏิบัติการวินโดวส์ 95/98
- ตำแหน่งพิกคัของวินโดว้รองจะใช้หน่วยพิกคัของวินโดว้หลัก (Parent Windows) แม้ว่าหน่วยพิกคัในวินโดว้รองจะมีหน่วยทีแตกต่างจากวินโดว้หลักก็ตาม

#### 2.3.5 การใช้งานคำสั่ง PaintPicture [4]

PaintPicture เป็นคำสั่งสำหรับใช้ในการวาดภาพลงบนฟอร์ม คอนโทรล PictureBox และเครื่องพิมพ์ โดยมีรูปแบบการเรียกใช้งานดังต่อไปนี้

`Object.PaintPicture picture, x1, y1, width1, height1, x2, y2, width2, height2, opcode`

Object	เป็นชื่อของออบเจกต์เป้าหมาย (ฟอร์ม คอนโทรล PictureBox หรือเครื่องพิมพ์) ที่เราต้องการจะวาดภาพลงไป ถ้าหากเราไม่ได้ระบุชื่อออบเจกต์ จะถือว่าออบเจกต์เป็นฟอร์มที่บรรจุโค้ดโปรแกรมดังกล่าว
Picture	เป็นภาพที่ใช้เป็นต้นฉบับในการวาด คุณสมบัติของภาพก็คือต้องเป็นภาพในรูปแบบที่ Visual Basic รู้จักและสามารถกำหนดเป็นค่าคุณลักษณะ Picture ของฟอร์ม หรือคอนโทรล PictureBox ได้
x1, y1	เป็นค่าพิกัดมุมบนซ้ายของตำแหน่งบนออบเจกต์เป้าหมายที่จะวาดภาพลงไป มีหน่วยเป็นไปตามที่กำหนดไว้ในคุณลักษณะ ScaleMode
width1	เป็นค่าความกว้างของภาพที่ต้องการให้ปรากฏบนออบเจกต์เป้าหมาย มีหน่วยเป็นไปตามที่กำหนดไว้ในคุณลักษณะ ScaleMode ถ้าหากว่าความกว้างที่ระบุนี้มีขนาดเล็กหรือใหญ่กว่าความกว้างของภาพต้นฉบับที่ (width2) ภาพดังกล่าวจะถูกปรับขนาดให้พอดีกับค่าความกว้างนี้ หรือหากว่าผู้เขียนโปรแกรมไม่ได้กำหนดค่านี้โดยเว้นช่องว่างไว้ จะถือเอาความกว้างของภาพต้นฉบับเป็นหลัก
height1	เช่นเดียวกับกับความกว้าง แต่เป็นค่าความสูงของภาพบนออบเจกต์เป้าหมาย
x2, y2	เป็นค่าพิกัดมุมบนซ้ายของตำแหน่งบนออบเจกต์ต้นฉบับ มีหน่วยเป็นไปตามที่กำหนดไว้ในคุณลักษณะ ScaleMode ถ้าหากผู้เขียนโปรแกรมละเอาไว้ จะถือเอาค่า 0 เป็นหลัก
width2	เป็นค่าความกว้างของภาพที่จะใช้เป็นส่วนภาพต้นฉบับ (อาจจะมีขนาดเล็กกว่าความกว้างของภาพต้นฉบับก็ได้) มีหน่วยเป็นไปตามที่กำหนดไว้ในคุณลักษณะ ScaleMode ถ้าหากละเอาไว้จะถือเอาความกว้างของภาพต้นฉบับเป็นหลัก
height2	เช่นเดียวกับกับความกว้าง แต่เป็นค่าความสูงของภาพที่ต้องการจะใช้เป็นส่วนภาพต้นฉบับ
opcode	เป็นโค้ดพิเศษที่ใช้เฉพาะกับการจัดการภาพบิตแมป เพื่อกำหนดรูปแบบของการวาดบนออบเจกต์เป้าหมายด้วยภาพต้นฉบับ โค้ดดังกล่าวนี้เป็นการจัดการระดับบิต (Bit-wise Operation) ของคอมพิวเตอร์ ที่จะกระทำกันระหว่าง 2 ออบเจกต์ดังกล่าว

ในการใช้งาน PaintPicture เราสามารถจะกลับภาพ (Flip) ตามแนวตั้งและตามแนวนอนในการวาดของเราได้ด้วยการกำหนดค่าความกว้างและ/หรือความสูงของภาพที่จะปรากฏบนออบเจกต์เป้าหมาย (width1/height1) ให้มีค่าติดลบ

**2.3.6 Raster Operation [4]**

ภาพบิตแมปเป็นอาร์เรย์ของข้อมูลที่เก็บค่าเป็น 0 หรือ 1 ในการนำภาพบิตแมป 2 ภาพมากระทำกัน จะมีตัวแปรที่ส่งผลอยู่ 3 ตัวคือ เป้าหมาย (Destination) แหล่งกำเนิด (Source) และรูปแบบ (Pattern) ซึ่งหมายถึงรูปแบบของแปรงสี (Brush ) ในการรวมตัวกันของบิตแมป 2 ภาพสามารถมีได้หลายรูปแบบ โดยมีพื้นฐานอยู่ 4 ประการ ดังนี้

NOT คือการเปลี่ยนค่าบิตให้เป็นตรงข้าม เช่น NOT 0 = 1 ,NOT 1 = 0

AND คือการเทียบค่าบิตสองตัว และจะให้ค่า 1 ก็ต่อเมื่อทั้งสองตัวมีค่าเป็น 1

OR คือการเทียบค่าบิตสองตัว จะให้ค่าเป็น 1 ก็ต่อเมื่อตัวใดตัวหนึ่งมีค่าเป็น 1

XOR คือการเทียบค่าบิตสองตัว จะให้ค่าเป็น 1 ก็ต่อเมื่อบิตสองตัวมีค่าไม่เหมือนกัน

จะเห็นได้ว่ารูปแบบของการกระทำระหว่างบิตแมปที่เป็นไปได้มีอยู่ด้วยกันมากมาย(อย่างน้อยก็ 20 วิธี) แต่ Visual Basic ได้จัดสร้างค่าตัวแปรคงที่ที่ใช้บ่อยๆ ไว้สำหรับกรณีแล้ว เช่นดังตัวอย่างต่อไปนี้

ตารางที่ 2.1 ค่าตัวแปรคงที่ สำหรับการทำ Raster Operation ของ Visual Basic

ค่าคงที่	ค่าคงที่เป็นเลขฐาน 16	ความหมาย
VbMergePaint	&H00BB0226	NOT บิตแมปแหล่งกำเนิดและรวมเข้ากับบิตแมปเป้าหมายด้วยรูปแบบ OR
VbSrcAND	&H008800C6	รวมบิตของบิตแมปของบิตแมปแหล่งกำเนิดกับบิตแมปเป้าหมายด้วยรูปแบบ AND
VbSrcCopy	&H00CC0020	Copy บิตแมปแหล่งกำเนิดกับบิตแมปเป้าหมาย
VbSrcInvert	&H00660046	รวมบิตของบิตแมปแหล่งกำเนิดกับบิตแมปเป้าหมายด้วยรูปแบบ XOR
VbSrcPaint	&H00EE0086	รวมบิตของบิตแมปแหล่งกำเนิดกับบิตแมปเป้าหมายด้วยรูปแบบ OR



### 2.3.7 สไปรต์และภาพมาสก์ (Sprite & Mask) [4]

คำว่าสไปรต์ (Sprite) หมายถึง ภาพของวัตถุที่เคลื่อนไหวอยู่ไปมาและตอบโต้กับผู้เล่นหรือวัตถุอื่นๆ ที่ปรากฏในเกมได้ ภาพสไปรต์เหล่านี้จะเป็นภาพที่มีรูปทรงอิสระ ซึ่งในทางเทคนิคแล้วภาพเหล่านี้เป็นภาพรูปทรงสี่เหลี่ยมมุมฉากที่ถูกทำให้จางหายไป ทำให้เรามองเห็นราวกับว่าภาพวัตถุนั้นปรากฏบนฉากหลัง เมื่อนำภาพสไปรต์หลายๆ ภาพมาต่อเนื่องกันเราก็จะมองเห็นเป็นภาพเคลื่อนไหวของสไปรต์

ส่วนภาพฉากหลังของภาพบิตแมปทั่วไปที่มีลักษณะเป็นสี่เหลี่ยมมุมฉากหายไปนั้น มีหลักการคือ ในเชิงเทคนิคนั้น ภาพบิตแมปก็คือข้อมูลที่เรียงตัวกันอยู่เป็นอาร์เรย์ในหน่วยความจำ มันเรียงตัวกันเป็นเส้นตรง หมายความว่าหากภาพมีขนาด 10 คูณ 10 พิกเซลถ้าหากว่าแต่ละพิกเซลแทนด้วยข้อมูลขนาด 1 ไบต์ ก็จะต้องอาศัยหน่วยความจำขนาด 10 ไบต์เรียงกัน 10 ชุด ซึ่งเท่ากับ 100 ไบต์ ในการเก็บภาพดังกล่าวนี้ เมื่อเราต้องการให้ข้อมูลของภาพนี้ไปอยู่ ณ ตำแหน่งอื่นของหน่วยความจำ เราก็จะทำการ Blitting (ในที่นี้หมายถึงการใช้งานฟังก์ชัน BitBlt() ) ทั้ง 100 ไบต์นี้ไปยังตำแหน่งอื่น ทำให้เราตระหนักข้อเท็จจริงได้ว่าฟังก์ชันวาดภาพโดยพื้นฐานแล้วจะทำงานกับภาพในลักษณะที่เป็นสี่เหลี่ยมมุมฉากเท่านั้น (เพราะไม่สามารถเลือก Bit เฉพาะข้อมูลที่เราต้องการให้ไปยังตำแหน่งอื่นได้) เมื่อเป็นดังนี้วิธีการที่เราจะสามารถสร้างภาพสไปรต์รูปทรงอิสระขึ้นมาได้ก็ต้องอาศัยภาพบิตแมปอีกภาพหนึ่งเข้ามาช่วย โดยมีหลักการว่าจะต้องวาดภาพอีกภาพหนึ่งทับส่วนที่เป็นฉากหลังของวัตถุที่เราสนใจ และการกระทำเช่นนั้นจะต้องนำไปสู่การที่ฉากหลังของภาพวัตถุถูกมองทะลุไปได้ และทำให้ผลลัพธ์ที่ได้คือภาพวัตถุที่เราสนใจมีรูปทรงอิสระตามต้องการ คำตอบของวิธีการนี้คือการกำหนด Raster Operation ของการ Blitting()

## บทที่ 3

### การสร้างโปรแกรม

### พัฒนาการเรียนรู้เด็กก่อนวันเรียน

ในช่วงแรกจะเป็นการศึกษา และทำการรวบรวมข้อมูลต่าง ๆ ที่ใช้ในการพัฒนาโปรแกรม โดยได้ทำการศึกษาการใช้งานฟังก์ชันของ Visual Basic และการทำงานของฟังก์ชัน Windows API แล้ว จากนั้นจะเป็นขั้นตอนการออกแบบโปรแกรมว่าในโปรแกรมนั้นจะประกอบด้วยอะไรบ้าง ซึ่งสามารถสรุปได้ดังนี้

#### 3.1 ซอฟต์แวร์ที่ใช้ในการพัฒนาโปรแกรม

3.1.1 Microsoft Visual Basic Version 6.0

3.1.2 Adobe Photoshop 6.0

3.1.3 การอ้างอิง Function Windows API

#### 3.2 คุณลักษณะของซอฟต์แวร์ (Software Specification)

3.2.1 โปรแกรมพัฒนาการเรียนรู้เด็กก่อนวัยเรียนนี้ สำหรับเด็กก่อนวัยเรียนอายุ 3-6 ปี โดยมีครู หรือ ผู้ปกครองคอยแนะนำ

3.2.2 Input Specification ของโปรแกรม จะใช้เมาส์ และ คีย์บอร์ดเป็นอุปกรณ์ในการ input ข้อมูล

3.2.3 Output Specification ของโปรแกรม เป็นภาพ 2 มิติแสดงออกทางจอภาพ

### 3.3 โครงสร้างของโปรแกรมพัฒนาการเรียนรู้เด็กก่อนวัยเรียน

โปรแกรมพัฒนาการเรียนรู้เด็กก่อนวัยเรียนนี้ ประกอบไปด้วย 3 ส่วนหลัก ๆ แสดงโครงสร้างโดยรวมของโปรแกรมพัฒนาการเรียนรู้เด็กก่อนวัยเรียนดังแสดงรูปที่ 3.1

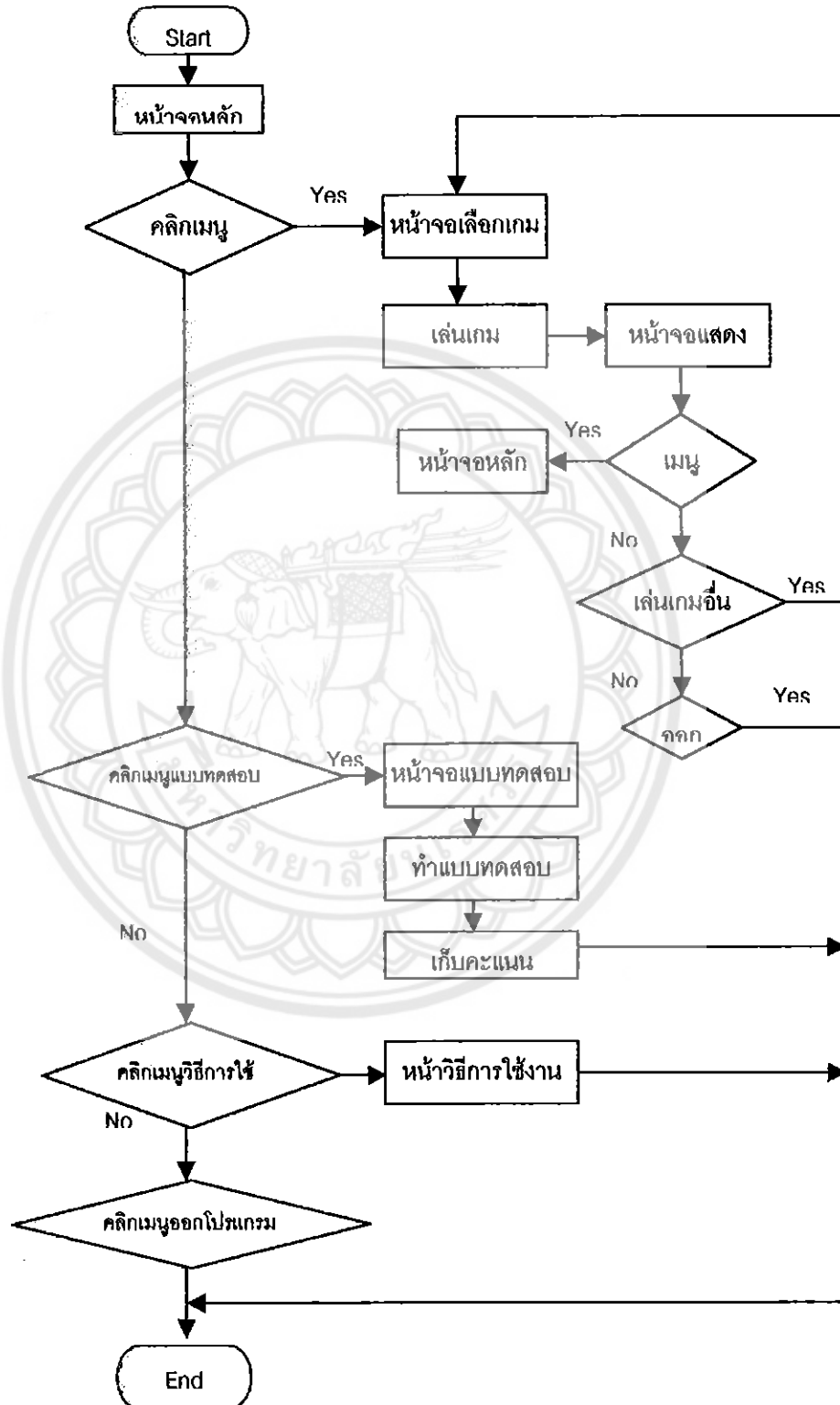


รูปที่ 3.1 โครงสร้างโดยรวมของโปรแกรม

- **Games**- ส่วนของเกม ในส่วนนี้จะประกอบไปด้วยเกมที่นำมาใช้ในการวัดพัฒนาการเรียนรู้ 4 เกม โดยภายในแต่ละเกมจะมีรูปแบบที่ต่างกัน ในส่วนนี้ผู้เล่นสามารถ เลือกเล่นเกมได้ทุกเกม
- **Test** – ส่วนของแบบทดสอบ จะเป็นกรนำเกมต่างๆ มาทำเป็นแบบทดสอบ โดยที่ผู้เล่นจะต้องเข้าเล่นเกมตามลำดับที่จัดไว้ให้ และจะมีการนำผลคะแนนที่ผู้เล่นทำได้ไปประเมินเทียบกับเกณฑ์ปกติ ในส่วนนี้จะมีการเก็บข้อมูลของผู้เล่นเอาไว้ เพื่อที่ผู้เล่นกลับมาเล่นใหม่ ก็จะทำให้การนำผลนั้นมาเปรียบเทียบกับพัฒนาการเรียนรู้เปลี่ยนแปลงไปในทางใด
- **Help** – ส่วนช่วยเหลือ จะแนะนำวิธีการใช้งานโปรแกรมต่างๆ

### 3.3.1 หลักการทำงานของโปรแกรม

โครงสร้างการทำงานทั้งหมดของโปรแกรมสามารถแสดงได้ดังรูปที่ 3.2



รูปที่ 3.2 โครงสร้างของโปรแกรมพัฒนาการเรียนรู้อีก่อนวัยเรียน

### 3.3.2 โครงสร้างการทำงานในส่วนของเกม

1. จะมีเกมให้ผู้เล่นเลือกทั้งหมด 4 เกม คือ
  - ความสามารถในการนับ รู้ค่าจำนวน 1-10
  - ความสามารถในการรู้จักสีต่างๆ
  - รู้จักรูปทรงเรขาคณิต
  - ความสามารถในการแบ่งขนาดของวัตถุ และสิ่งของต่างๆ
2. ในแต่ละเกมจะมีรูปแบบที่แตกต่างกัน ซึ่งในการเข้าเล่นเกมแต่ละครั้งจะทำการสุ่มรูปแบบเกมให้แตกต่างกันไป
3. ขณะที่เล่นเกม ถ้าผู้เล่นไม่เข้าใจในการเล่น เกม ครูหรือผู้ปกครองสามารถเลือกส่วนของบทเรียนให้ผู้เล่นได้ทำความเข้าใจก่อนได้

### 3.3.3 โครงสร้างการทำงานในส่วนของแบบทดสอบ

1. ในส่วนของเนื้อหาของแบบทดสอบ จะมีเกมให้เล่นทั้งหมด 4 เกม ในลักษณะเป็นลำดับไป และจะมีการเก็บคะแนนของผู้เล่นในแต่ละเกม โดยที่ผู้เล่นจะต้องเล่นจนครบทุกเกม
2. เมื่อผู้เล่นทำแบบทดสอบเสร็จ จะมีการสรุปผลลัพท์ออกมา และนำผลลัพท์ที่ได้ไปเทียบกับเกณฑ์ของแต่ละเกมที่ตั้งไว้
3. สำหรับผู้เล่นที่เคยเข้ามาทำแบบทดสอบแล้ว โปรแกรมจะนำผลลัพท์ไปเปรียบเทียบกับของเก่าด้วย เพื่อที่จะบอกได้ว่ามีการพัฒนาการเรียนรู้ไปในทางที่ดีขึ้นหรือแย่ลงอย่างไร

### 3.3.4 โครงสร้างการทำงานในส่วนของความช่วยเหลือ

เป็นระบบความช่วยเหลือ จะแสดงรายละเอียดต่างๆ ของการใช้งานโปรแกรมและบทเรียนช่วยในการเล่น

## บทที่ 4

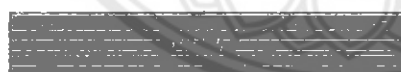
### การใช้งานโปรแกรม พัฒนาการเรียนรู้เด็กก่อนวัยเรียน

#### 4.1 เมนูหลัก

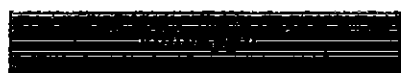
เมื่อเริ่มต้น โปรแกรมจะปรากฏหน้าจอเมนูหลักดังรูปที่ 4.1



รูปที่ 4.1 หน้าจอหลักของโปรแกรม



- คลิกเพื่อเข้า เล่นเกมส์



- คลิกเพื่อเข้าสู่ แบบทดสอบ



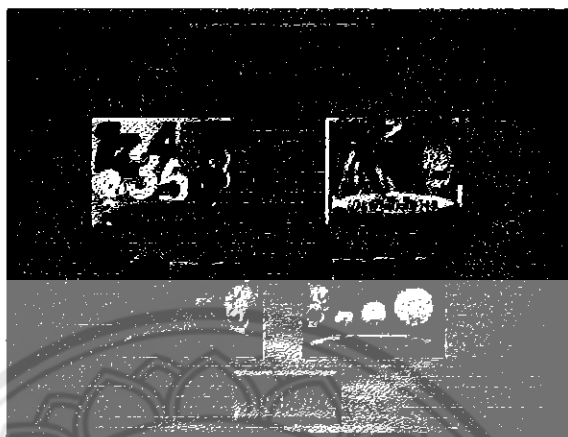
- คลิกเพื่อเข้าสู่ วิธีการใช้งาน



- ออกจากโปรแกรม

### 4.2 เกมส์

เมื่อกดคลิก 'เล่นเกมส์' จะปรากฏหน้าจอดังรูปที่ 4.2

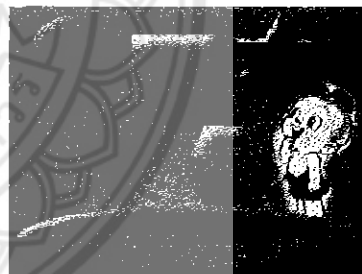


รูปที่ 4.2 หน้าจอเกมส์

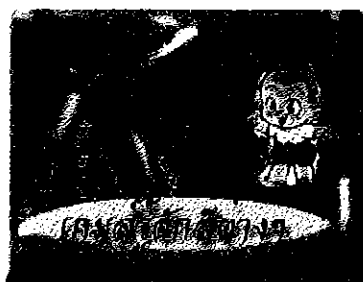
ประกอบไปด้วยเมนูต่างๆ ดังนี้



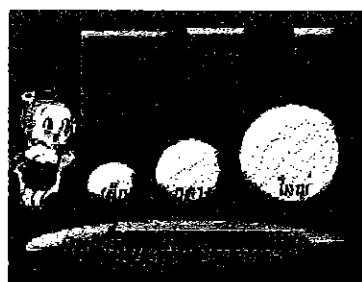
- กดคลิกเพื่อเข้าสู่ 'เกมส์ตัวเลข'



- กดคลิกเพื่อเข้าสู่ 'เกมส์รูปภาพชนิด'



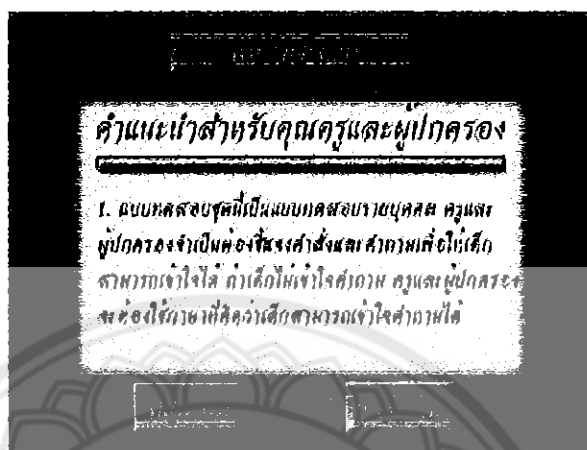
- กดคลิกเพื่อเข้าสู่ 'เกมส์รู้จักสีต่างๆ'



- กดคลิกเพื่อเข้าสู่ 'เกมส์เปรียบเทียบขนาด'

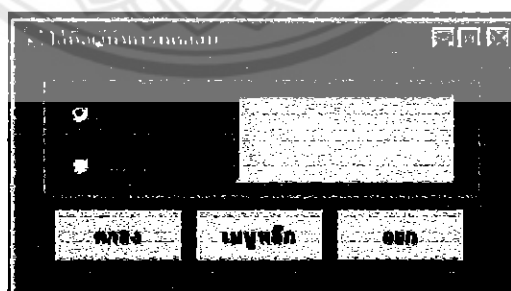
### 4.3 แบบทดสอบ

เมื่อคลิก 'แบบทดสอบ' จะปรากฏหน้าจอ ดังรูปที่ 4.3



รูปที่ 4.3 หน้าจอแบบทดสอบ

เมื่ออ่านคำแนะนำเสร็จแล้ว จะเข้าสู่การทดสอบ โดยจะต้องมีการกรอกข้อมูล คือ ชื่อและเพศของผู้เล่น ซึ่งถ้าเคยทดสอบแล้วก็จะนำข้อมูลเก่ามาเปรียบเทียบ แต่ถ้ายังไม่เคยมาทดสอบ ก็ จะทำการสร้างใหม่ให้ หน้าจอการรับข้อมูล แสดงดังรูปที่ 4.4



รูปที่ 4.4 หน้าจอรับข้อมูลของผู้เล่น



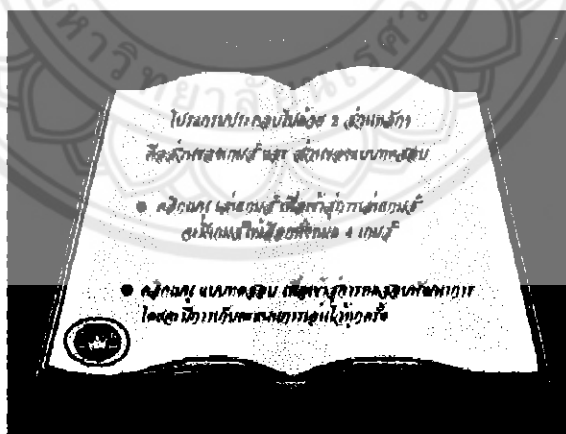
เมื่อทำการทดสอบเสร็จแล้วจะแสดงผลการประเมินดังรูปที่ 4.5



รูปที่ 4.5 แสดงผลการประเมินการทดสอบ

#### 4.4 ช่วยเหลือ

เมื่อคลิก 'ช่วยเหลือ' จะปรากฏหน้าจอดังรูปที่ 4.5



รูปที่ 4.5 หน้าจอระบบช่วยเหลือ

## บทที่ 5

## บทสรุป

ในบทนี้จะกล่าวถึงการสรุปผลที่ได้จากการทำโครงการโปรแกรมพัฒนาการเรียนรู้เด็กก่อนวัยเรียน การศึกษารวมทั้งปัญหาที่เกิดขึ้นพร้อมทั้งแนวทางในการแก้ปัญหา ตลอดจนข้อดีข้อเสีย และแนวทางในการพัฒนาในอนาคตต่อไป

## 5.1 สรุปผล

โปรแกรมนี้ได้พัฒนาโดยใช้โปรแกรม Visual Basic version 6.0 นำเสนอ โดยมีการแบ่งโปรแกรมออกเป็น 4 ส่วน คือ บทเรียนเสริม เกม แบบทดสอบ เพื่อช่วยในการพัฒนาความรู้ความสามารถให้เด็กก่อนวัยเรียนได้รับความรู้ ความเพลิดเพลิน และยังเป็นเตรียมความพร้อมก่อนเข้าเรียนในชั้นประถมศึกษาต่อไป และมีระบบช่วยเหลือสำหรับคอยแนะนำวิธีการเล่นต่างๆ

## 5.2 ปัญหาและแนวทางการแก้ไข

จากการทำโครงการเรื่องนี้ ปัญหาที่ทางผู้จัดทำพบเป็นปัญหาในเรื่องของการเขียนโปรแกรมซึ่งจะเกิดขึ้นได้ตลอดเวลา ซึ่งเราสามารถแก้ไขปัญหาโดยการปรึกษาอาจารย์ที่ปรึกษา และค้นคว้าด้วยตัวเองจากตำราคู่มือการเขียนโปรแกรมต่างๆ

## 5.3 แนวทางการพัฒนาต่อ

1. เพิ่มเติมในส่วนของเกมแบบทดสอบเพื่อให้ครอบคลุมพัฒนาการต่างๆของเด็กให้มากยิ่งขึ้น
2. นำหลักการมาประยุกต์ในการสร้างโปรแกรมวัดพัฒนาการของเด็กในช่วงวัยอื่นๆที่มีการเปลี่ยนแปลงทางพัฒนาการอย่างเห็นได้ชัด

4400608

LB

10 28-5

N 282 ✓

๒๕๔๔ C.2

## เอกสารอ้างอิง

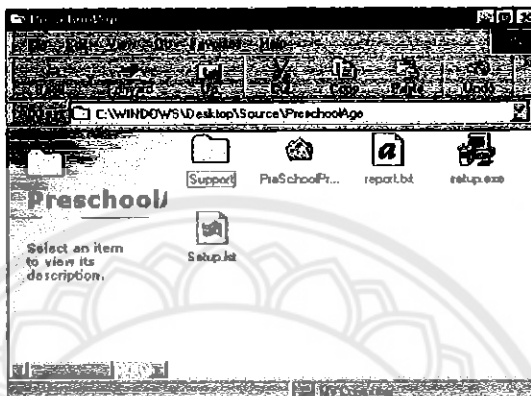
- [1] กรมวิชาการ. คู่มือ หลักสูตรก่อนประถมศึกษา พุทธศักราช 2540 (อายุ 3-6). กรุงเทพฯ : กรมวิชาการ กระทรวงศึกษาธิการ.
- [2] กิตติ ภัคดีวัฒนะกุล, จำลอง ครูอุสาหะ. Visual Basic 6 ฉบับโปรแกรมเมอร์. กรุงเทพฯ : หจก. ไทยเจริญการพิมพ์.
- [3] สุทธิศักดิ์ พงศ์ธนาพาณิชย์. การเขียนโปรแกรมด้วย Visual Basic 6.0 ระดับสูง การใช้งานฟังก์ชันวินโดวส์ API – 32 บิต. กรุงเทพฯ : Primaed Co.,Ltd.
- [4] พุฒิพงษ์ นาคะปัท. การเขียนคอมบนวินโดวส์ด้วย Visual Basic. กรุงเทพฯ : ซีเอ็ดดูเคชั่น. 2542.



## ภาคผนวก

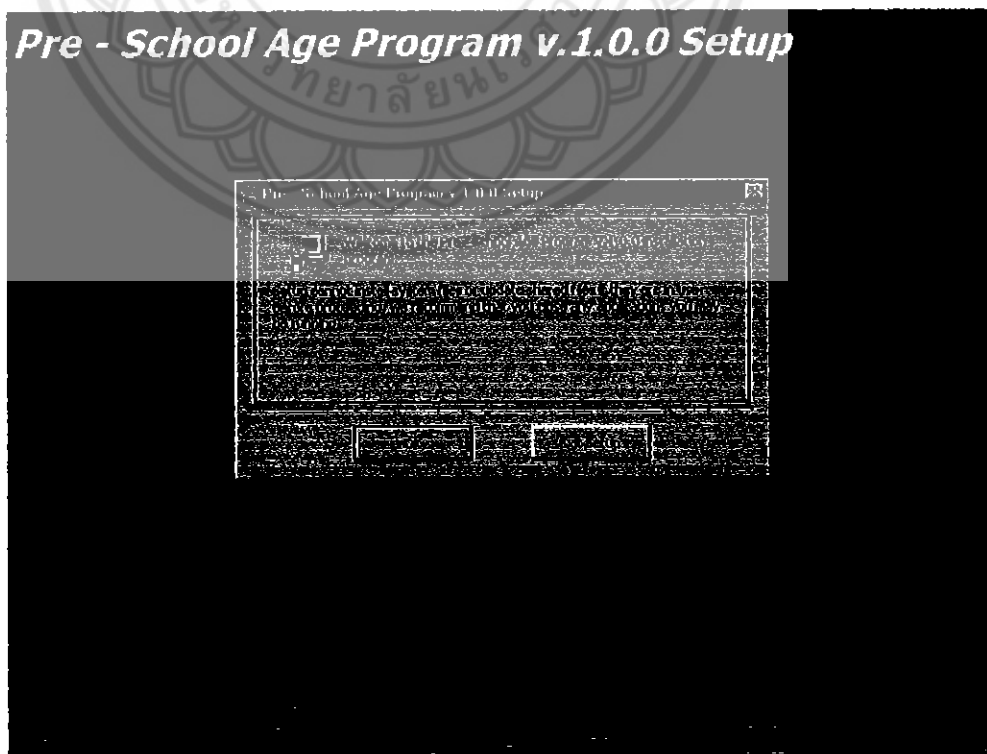
วิธีการติดตั้งใช้งาน โปรแกรมพัฒนาการเรียนรู้เด็กก่อนวัยเรียน

1. โปรแกรมจะประกอบไปด้วยไฟล์และไฟล์เคอร์ต่างๆดังรูป



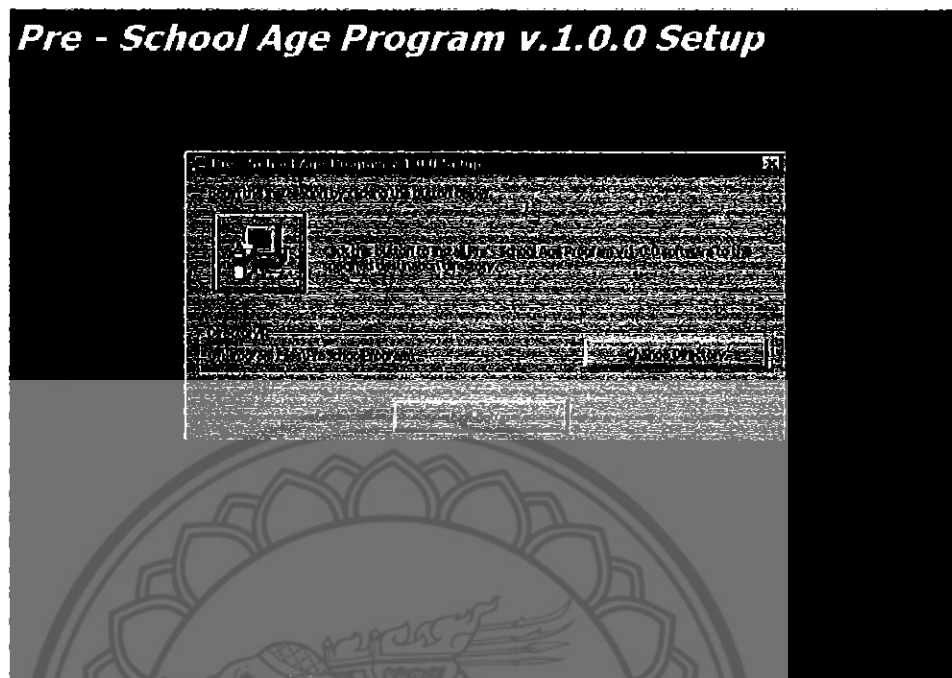
รูปที่ 1 ไฟล์และไฟล์เคอร์ของ โปรแกรม

2. จากนั้นทำการติดตั้ง โปรแกรม โดยดับเบิลคลิกที่ไอคอน setup.exe จะปรากฏหน้าจอดังรูปที่ 2



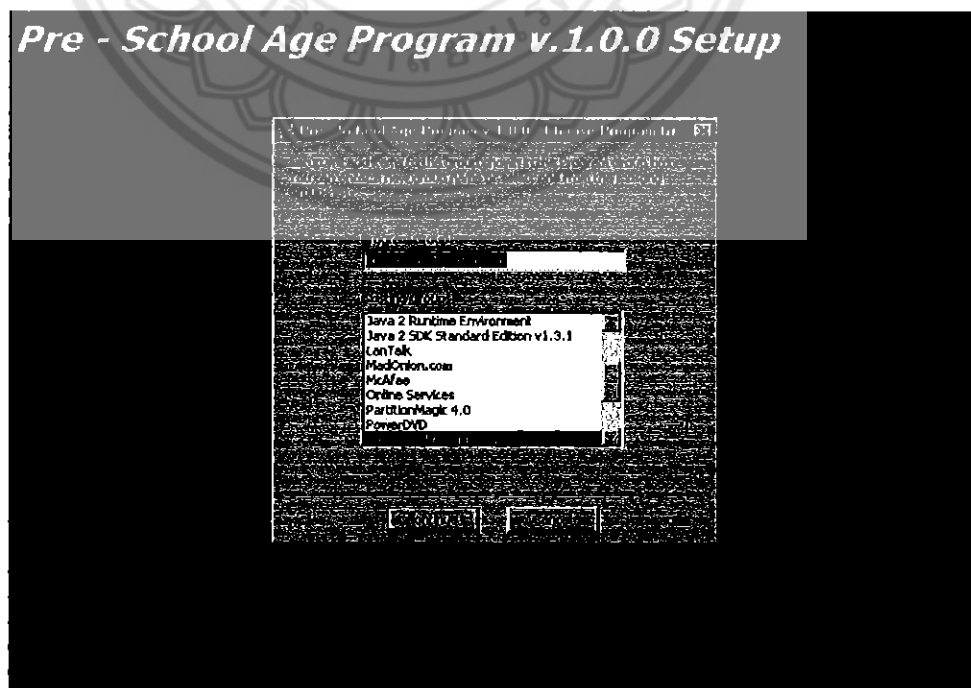
รูปที่ 2 หน้าจอการติดตั้ง โปรแกรม

3. เลือก OK เพื่อทำการยืนยันการติดตั้ง โปรแกรม และเลือกไดเรกทอรีที่ต้องการจะติดตั้ง โดยคลิกที่ปุ่ม Chang Directory หรือคลิกที่รูปคอมพิวเตอร์ได้เลย เพราะระบบจะเลือกไว้ให้อยู่แล้ว



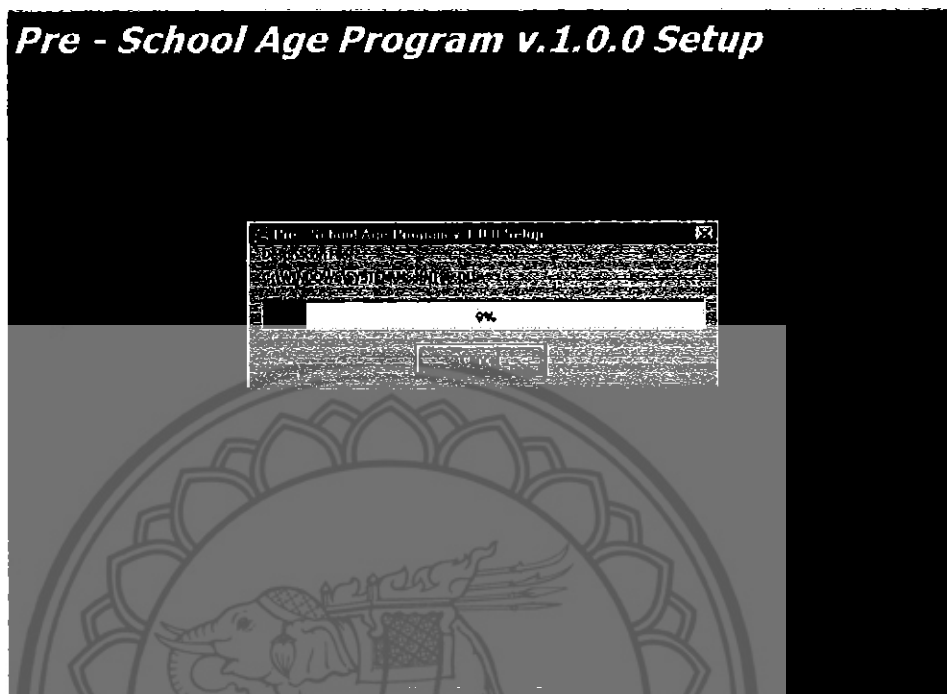
รูปที่ 3 หน้าจอการเลือกไดเรกทอรีในการติดตั้งโปรแกรม

4. จากนั้นโปรแกรมจะทำการเลือก Program Group ถ้าไม่ต้องการเปลี่ยนแปลงก็คลิกเลือก Continue เพื่อทำการติดตั้งโปรแกรมต่อไป



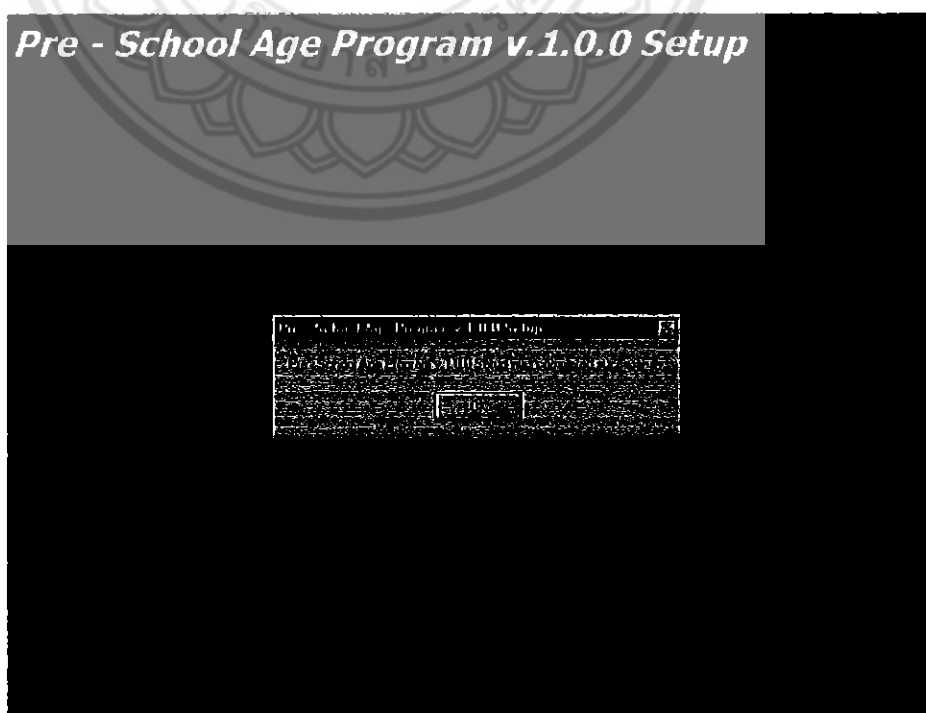
รูปที่ 4 หน้าจอการเลือก Program Group

5. จากนั้นโปรแกรมจะทำการ copy ไฟล์ต่างๆและทำการติดตั้งลงเครื่อง โดยจะต้องรอนแถบสีน้ำเงินเต็ม โดยจะมีตัวเลขแสดงเป็นเปอร์เซ็นต์ เมื่อครบ 100 % เป็นอันว่าเสร็จขั้นตอนการติดตั้ง



รูปที่ 5 หน้าจอการติดตั้งไฟล์ลงเครื่อง

6. เมื่อการติดตั้งเสร็จสมบูรณ์ก็จะปรากฏหน้าต่างรูปที่ 6



รูปที่ 6 หน้าจอการติดตั้งเสร็จสมบูรณ์

7. เมื่อทำการติดตั้งโปรแกรมเสร็จเรียบร้อยแล้วให้ทำการ copy ไฟล์ db.mdb มาไว้ใน c:\ เนื่องจากโปรแกรมจะต้องดึงข้อมูลจากไฟล์นี้มาใช้งาน



รูปที่ 7 หน้าจอแสดงไฟล์ที่ได้จากการติดตั้ง

8. สิ้นสุดการติดตั้ง

## ประวัติผู้ทำโครงการ

1. นางสาว พรศิริ สุนทรธรรมกุล รหัส 41360389  
เกิดวันที่ 2 มีนาคม 2523 ภูมิลำเนา 210/7-8 ถ.มิตรภาพ อ.เมือง จ.พิษณุโลก  
จบการศึกษาระดับมัธยมศึกษาจากโรงเรียนเฉลิมขวัญสตรี
2. นาย สมชาติ คงเกตุ รหัส 41360496  
เกิดวันที่ 24 พฤษภาคม 2521 ภูมิลำเนา 1711 หมู่ 1 ต.เขาทราช อ.ทับคล้อ  
จ.พิจิตร จบการศึกษาระดับมัธยมศึกษาจากศูนย์บริการการศึกษานอกโรงเรียน  
อ.ตะพานหิน
3. นาย สมภพ มิ่งกลิ่น รหัส 41360504  
เกิดวันที่ 19 กันยายน 2522 ภูมิลำเนา 71/5 หมู่ 4 ต.บางซ้าย อ.เมือง จ.สุโขทัย  
จบการศึกษาระดับมัธยมศึกษาจากโรงเรียนสุโขทัยวิทยาคม

