



เกม Pac-man

นายรุ่งสิทธิ์ ประทีปะเสน

รหัสนักศึกษา 6404062620150 ชั้นปีที่ 2

เสนอ

อาจารย์ สติത്യ ประสมพันธ์

รายงานฉบับนี้เป็นส่วนหนึ่งของการเรียนวิชา 040613204 OBJECT-ORIENTED PROGRAMMING

สาขาวิชาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยพระจอมเกล้าพระนครเหนือ

ภาคเรียนที่ 1 ปีการศึกษา 2565

บทที่ 1 บทนำ

ที่มาและความสำคัญ

ในยุคปัจจุบันนี้เกมแนว 8 บิต หรือเกมเก่าที่วัยรุ่นยุค 90's แม้แต่ปัจจุบันก็ยังคงความน่าสนใจและน่าเล่นอยู่ทั่วโลก ด้วยระบบเกมที่มีความคลาสสิกมีเอกลักษณ์เป็นของตัวเอง ทำให้เกม 8 บิต มีความโดดเด่นและน่าศึกษา

เนื่องจากผู้จัดทำมีความสนใจในการเล่นเกม 8 บิต เป็นทุนเดิมอยู่แล้วและได้มีการศึกษาการใช้โปรแกรม Net bean ในการสร้างเกม จึงมีจุดมุ่งหมายในการพัฒนาเกม Pac-man ที่สามารถพัฒนาสมองและผ่อนคลายไปพร้อม ๆ กันได้

ประเภทของโครงการ

โครงการประเภทพัฒนาเกม

ประโยชน์

- ฝึกไหวพริบ
- ฝึกการวางแผนล่วงหน้า
- เพื่อความสนุกสนาน
- ฝึกความอดทน

ขอบเขตของโครงการ

- รายละเอียดเกมส์

เกมแพ็คแมน เมื่อจู่ ๆ ตัวอักษรกลมสีเหลืองที่คล้ายกับ พืชชาต้องวิ่งผ่านเขาวงกตที่พยายามจับจุดและหนีมืด เพื่อที่จะสามารถผ่านด่านไปได้จำเป็นต้องจับจุดสีขาวที่เกิดอยู่ตามแม่พิมพ์ให้ครบจึงจะชนะเกม

- วิธีเล่น

ใช้ปุ่มลูกศรบนคีย์บอร์ดเพื่อบังคับทิศทางของ Pacman และคอยหลบหลีกผีที่ไล่ตามเรา

- Storyboard

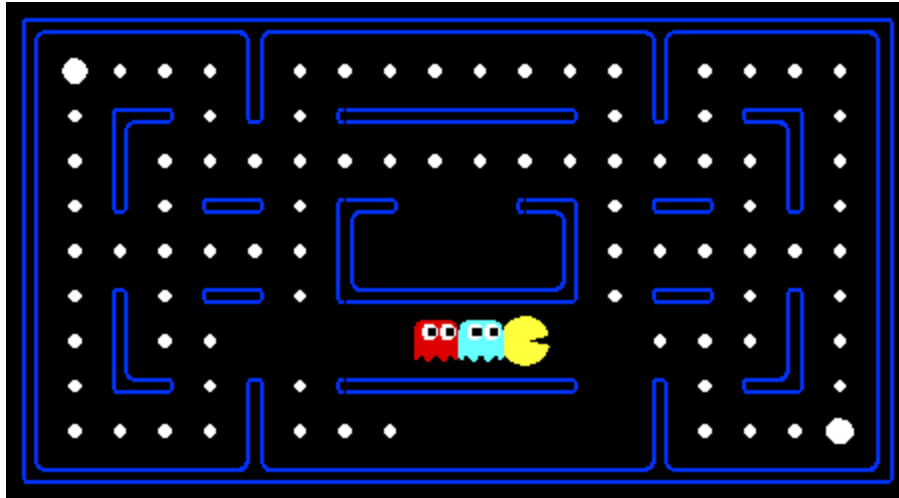
ตัวละคร



ฉาก

- เริ่มเกม

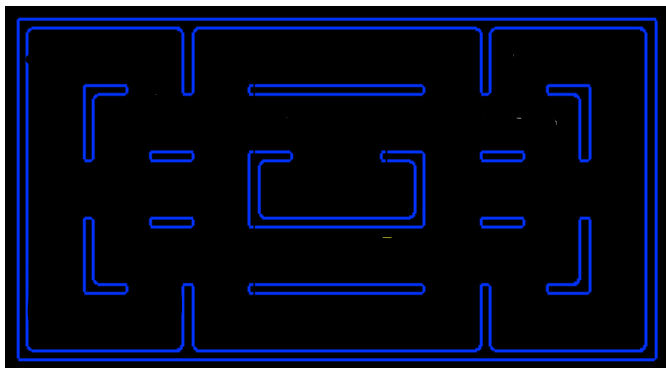




- เมื่อ Pacman เดินชนผีเกมชีวิตจะโดนหักไป 1 ดวง เมื่อชีวิตหมดเกมจะจบทันที และเมื่อผีกินจุดสีขาวลูกใหญ่จะไล่กินผีได้



- เมื่อสามารถกินจุดได้ทุกจุดบนแมพจะชนะเกม เมื่อกด R จะสามารถเริ่มเกมใหม่ได้



- ประโยชน์
 - ฝึกไหวพริบ
 - ฝึกการวางแผนล่วงหน้า
 - เพื่อความสนุกสนาน
 - ฝึกความอดทน
 - พัฒนาสมอง

- ตารางแผนการทำงาน

ลำดับ	งานที่ต้องทำ	1 เดือนก่อนส่งโปรเจค				
		1 - 4	5 - 10	11 - 15	16 - 25	26 - 31
1	วางแผนการทำงาน					
2	ออกแบบตัวละคร					
3	ศึกษาข้อมูล					
4	เขียนโปรแกรม					
5	ปรับปรุงและพัฒนา					

บทที่ 2 ส่วนการพัฒนา

เนื้อเรื่องย่อหรือวิธีการเล่น

เกมแพ็คแมน เมื่อจู่ ๆ ตัวอักษรกลมสีเหลืองที่คล้ายกับพิซซ่าที่หายไปเสียหนึ่ง ให้ผู้เล่นบังคับให้มันจับจุดไปตามเขาวงกตจนหมด โดยมีตัวมอนสเตอร์ทั้ง 4 ตัว 4 สี คอยขัดขวางตามไล่จับแพ็ค-แมน ซึ่งแต่ละตัวมีคาแรคเตอร์ที่ต่างกันไป ดังนี้

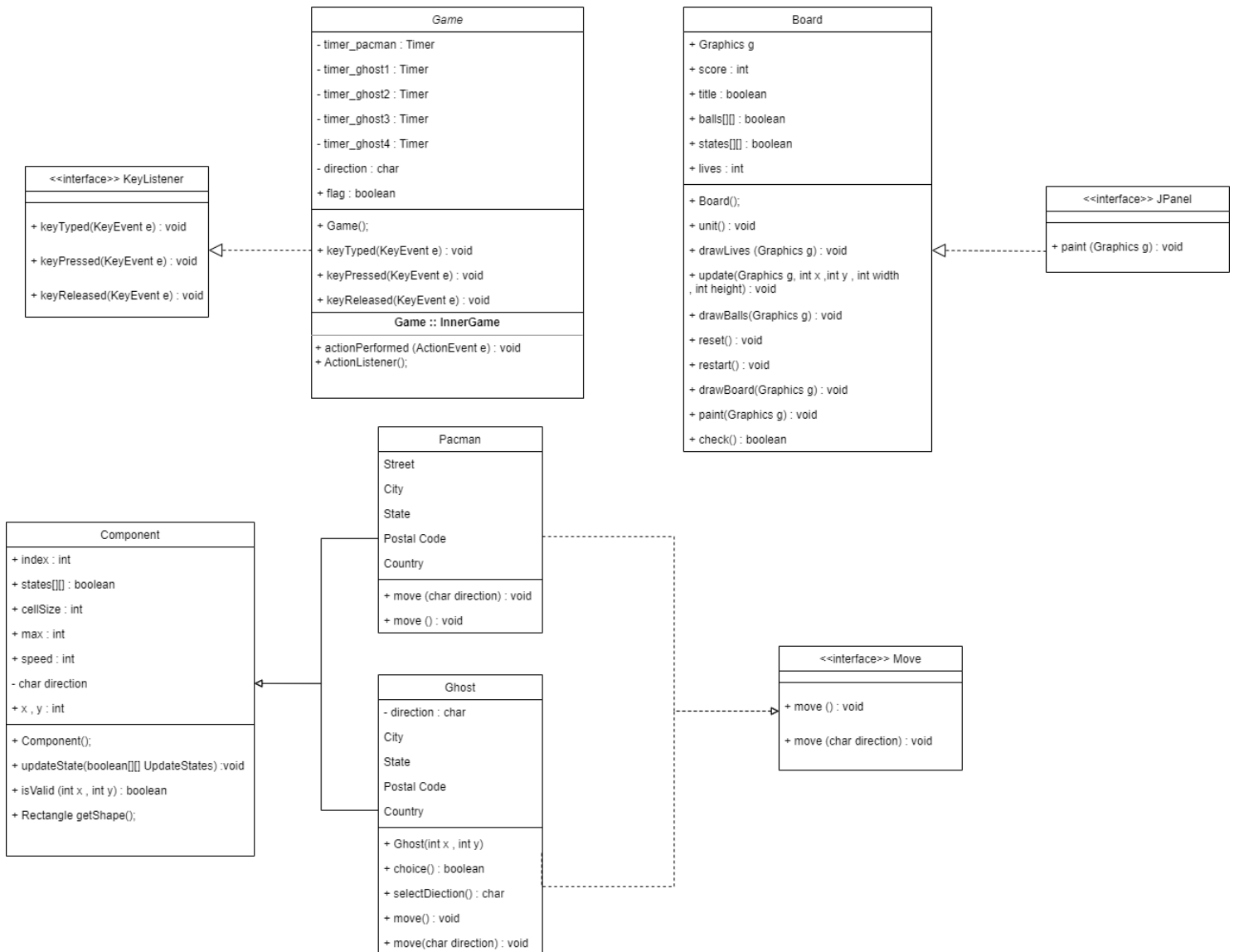
- ตัวสีแดง มีชื่อว่า “บลิงกี้” เป็นที่เคลื่อนที่ได้ไวที่สุด
- ตัวสีน้ำเงิน มีชื่อว่า “อิงกี้” เป็นตัวที่เคลื่อนที่รองลงมาจากสีแดง
- ตัวสีชมพู มีชื่อว่า “ฟิงค์กี้” เคลื่อนที่เร็วเท่าสีชมพู
- ตัวสีส้ม มีชื่อว่า “ไคลด์” เคลื่อนที่เร็วเท่าสีชมพู

วิธีเล่นก็ไม่ยาก เพียงแค่บังคับให้แพ็ค-แมนจับจุดที่มีอยู่ให้หมด โดยห้ามสัมผัสเจ้าผีทั้ง 4 ตัว หากเดินไปชนหรือปล่อยให้เจ้าผีมาชนถือว่าแพ้ในเกมนั้น โดยผู้เล่นจะมีชีวิตทั้งหมด 2 ชีวิตหากโดนผีครบ 2 ครั้งเกมจะจบทันที

รูปแบบการพัฒนา Application / Applet

รูปแบบ Java Application

คลาสไดอะแกรม



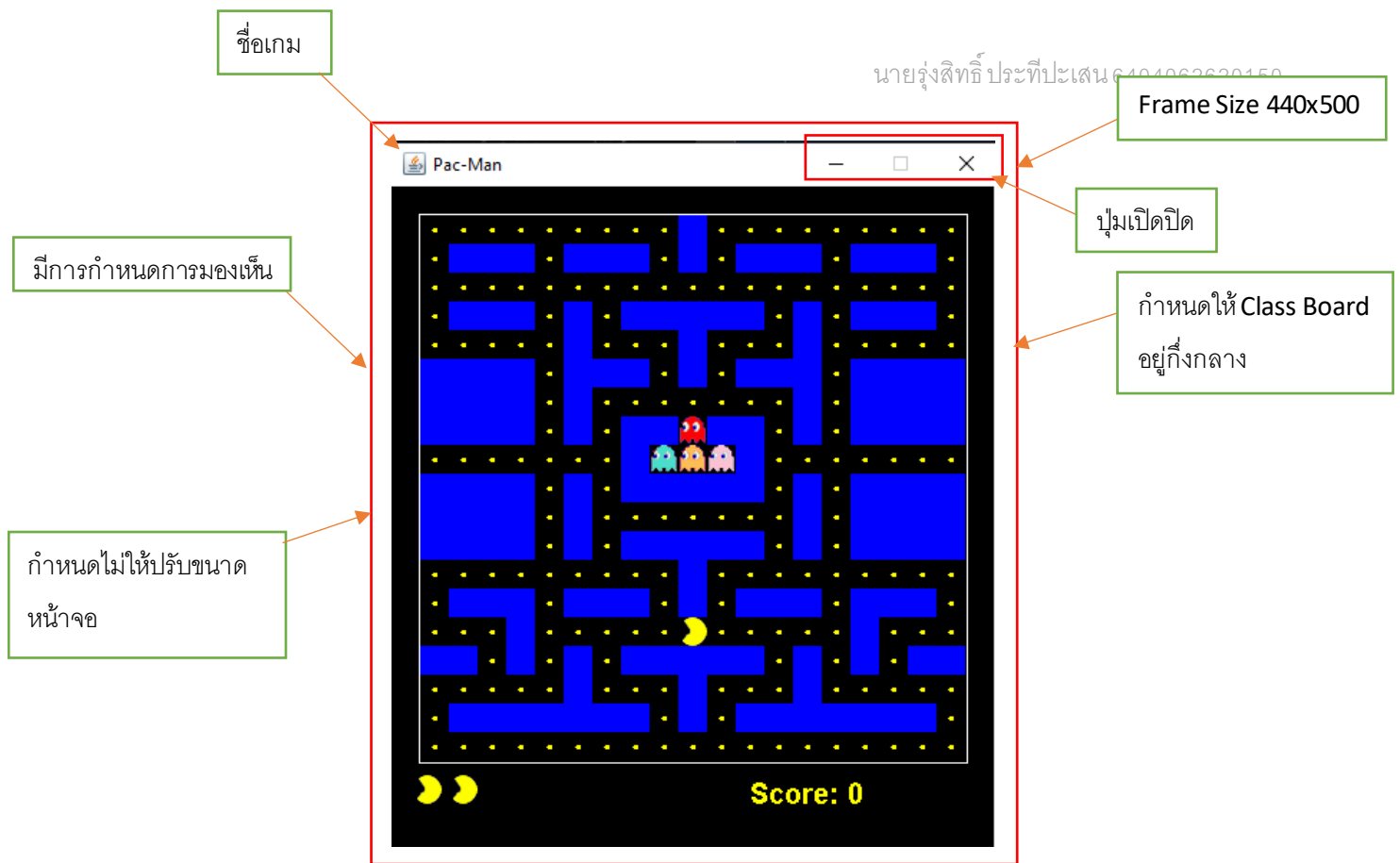
ในส่วนของคลาส Game มี inner class ที่มีการเรียก Action Performed และ ActionListener มาใช้ และเรียก interface Key Listener มาใช้ ส่วน Board มีการเรียก interface JPanel ที่มี method ของ paint มาใช้ในคลาส Pacman มีการเรียกใช้ interface Move และ extends คลาส Component มาใช้

อธิบายส่วนของโปรแกรม

1	Constructor	อธิบาย
	<pre> public class Board extends JPanel{ public Board() { title = true; balls = new boolean[Component.cellSize][Component.cellSize]; states = new boolean[Component.cellSize][Component.cellSize]; unit(); } } public class Pacman extends Component implements Move{ public Pacman(int x,int y){ this.x=x; this.y=y; } } public class Ghost extends Component implements Move{ private char direction; public Ghost(int x,int y) { this.x=x; this.y=y; direction='L'; } } </pre>	<p>มีการกำหนดค่าใน constructor Board ให้ title = true และมีการประกาศ balls , states เป็นแบบ boolean และมีการเรียกใช้ฟังก์ชัน unit(); เพื่อกำหนดค่าเริ่มต้น balls และ states ให้เป็น true</p> <p>มีการกำหนดค่าเริ่มต้นให้กับคลาส Pacman ให้ค่า x,y = ค่า x,y ที่รับเข้ามาผ่านพารามิเตอร์</p> <p>มีการกำหนดค่าเริ่มต้นให้กับคลาส Ghost ให้ค่า x,y มีค่าเท่ากับพารามิเตอร์ที่ส่งค่าเข้าไปและมีการกำหนด direction = 'L'</p>
2	Encapsulation	
	<pre> public class Game implements KeyListener{ Board board = new Board(); private Timer timer_pacman,timer_ghost1,timer_ghost2, timer_ghost3,timer_ghost4; private char direction = 'L'; } </pre>	<p>เป็นการประกาศตัวแปร ที่เป็นไพรเวท ซึ่งจะทำให้ตัวแปรเป็นตัวแปรที่ใช้ได้เฉพาะในคลาส ถ้าประกาศตัวแปรเป็น public จะสามารถใช้งานได้ในทุกคลาส</p>
3.	Composition	ไม่มีการใช้งานในส่วนนี้

4.	Polymorphism	อธิบาย
	<pre> @Override public void move(char direction){ switch (direction) { case 'L': if(isValid(x-speed,y) && isValid(x-speed, y+cellSize-5)){ x-=speed; } else if(y > 178 && y < 182 && x == 20){ x=380; } index = 0; break; case 'R': if(isValid(x+cellSize,y) && isValid(x+cellSize, y+cellSize- 5)){ x+=speed; } else if(y > 178 && y < 182 && x == 380){ x=20; } index = 1; break; case 'U': if(isValid(x,y-speed) && isValid(x+cellSize-5, y-speed)){ y-=speed; } index = 2; break; case 'D': if(isValid(x,y+cellSize) && isValid(x+cellSize-5, y+cellSize)){ y+=speed; } } } </pre>	<p>มีการเรียกใช้ method public void move ที่มีชื่อเหมือนกันแต่การทำงานต่างกัน</p>

	<pre> } index = 3; break; } } @Override public void move() { } </pre>	
5.	Abstract	
	<pre> public interface Move { abstract public void move(char direction); abstract public void move(); } </pre>	<p>ในส่วนของโปรแกรมไม่มีการใช้ method Abstract แต่มีการใช้งาน interface ที่กำหนดให้คลาส Pacman , Ghost มีการเรียกใช้ method abstract public void move(char direction); และ abstract public void move();</p>
6.	Inheritance	
	<pre> public class Board extends JPanel{ } public class Pacman extends Component implements Move{ } public class Ghost extends Component implements Move{ } </pre>	<p>มีการเรียกใช้ extends หรือสามารถเรียกใช้ Attribute และ Method ของคลาสที่สืบทอดมาได้</p>



Event Handle ที่ปรากฏในเกมมีการเรียกใช้ ActionListener ในการควบคุมการเคลื่อนไหวตัวละครผ่าน Timer และมีการเรียกใช้ KeyListener ผ่านการกดปุ่มโดยปุ่มที่กำหนด Event ไว้จะมี

- ลูกศรลง = จะทำการกำหนด direction ให้กับตัว Pacman เป็น “D”
- ลูกศรขึ้น = จะทำการกำหนด direction ให้กับตัว Pacman เป็น “U”
- ลูกศรซ้าย = จะทำการกำหนด direction ให้กับตัว Pacman เป็น “L”
- ลูกศรขวา = จะทำการกำหนด direction ให้กับตัว Pacman เป็น “R”
- ปุ่ม Enter = กดก่อนเริ่มเกมเพื่อเข้าสู่หน้าเริ่มเกมหลัก
- ปุ่ม R = เพื่อทำการ Restart เกม

อัลกอริทึมที่สำคัญในโปรแกรม

- เมื่อตัวแพ็ค-แมนกินปุ่มสีขาวย ปุ่มสีขาวยจะหายไปและคะแนนจะเพิ่มขึ้น
- เมื่อตัวแพ็ค-แมนโดนกับตัวผีชีวิตจะลดไปทันที 1 ดวง และเริ่มเกมใหม่ที่ตำแหน่งเดิมแต่จุดที่กินไปจะหายไป
- การเดินของตัวแพ็คแมนโดยใช้ Keylistener เป็นตัวกำหนดปุ่มโดยจะใช้ลูกศร ขึ้น ลง ซ้าย ขวา ในการควบคุมตัวละคร
- การเดินของตัวมอนสเตอร์มีการใช้ Keylistener แต่ User ไม่จำเป็นต้องควบคุมเอง แต่จะมีฟังก์ชันที่คอยกำหนดทิศทางของตัวมอนเตอร์แบบสุ่ม โดยใช้ Math.random

บทที่ 3 สรุป

ปัญหาที่พบบ่อยระหว่างการพัฒนา

- เกี่ยวกับตำแหน่งภายใน paint ที่ต้องนั่งคำนวณและกำหนดค่าเอง
- เวลาไม่เพียงพอต่อ scope ของงาน
- ปัญหาเกี่ยวกับการใช้ Thread

จุดเด่นของโปรแกรมที่ไม่เหมือนใคร

- มีการสร้างแมพโดยใช้ shape ที่กำหนดไว้ใน JPanel มาใช้งาน
- มอนสเตอร์มีความฉลาดได้ยาก

คำแนะนำสำหรับผู้สอนที่อยากให้อธิบาย หรือที่เรียนแล้วไม่เข้าใจ หรืออยากให้เพิ่มสำหรับน้อง ๆ รุ่นต่อไป

- อยากให้ลดภาระงานให้สัมพันธ์กับเวลาสอบและอ่านหนังสือ เนื่องด้วยเวลาที่มีให้ไม่เพียงพอต่อการศึกษาด้วยตนเองในเวลาว่าง
- และอยากให้ลดความยากของ task ลงครับ