



การพัฒนาผลสัมฤทธิ์ทางการเรียน เรื่อง ตัวชี้ (Pointers) ในวิชาพื้นฐานการเขียนโปรแกรมคอมพิวเตอร์ (21909-1005) ของนักเรียนระดับ ปวช.2 สาขา งานเทคนิคคอมพิวเตอร์ โดยใช้ชุดแบบฝึกทักษะการจำลองสถาปัตยกรรม หน่วยความจำร่วมกับการเรียนรู้แบบใช้ปัญหาเป็นฐาน (Problem-Based Learning) ปีการศึกษา 1 ปีการศึกษา 2569

โดย นายฐปน เกาะแก้ว

ตำแหน่ง ครู

แผนกเมคคาทรอนิกส์ วิทยาลัยเทคนิคเชียงราย

บทคัดย่อ

ชื่อโครงการวิจัย: การพัฒนาผลสัมฤทธิ์ทางการเรียน เรื่อง ตัวชี้ (Pointers) ในวิชาพื้นฐานการเขียนโปรแกรมคอมพิวเตอร์ (21909-1005) ของนักเรียนระดับ ปวช.2 สาขางานเทคนิคคอมพิวเตอร์ โดยใช้ชุดแบบฝึกทักษะการจำลองสถาปัตยกรรมหน่วยความจำร่วมกับการเรียนรู้แบบใช้ปัญหาเป็นฐาน (Problem-Based Learning)

ผู้วิจัย: นายจุปน เกาะแก้ว

ปีการศึกษา: 2569

การวิจัยในชั้นเรียนครั้งนี้มีวัตถุประสงค์เพื่อ 1) พัฒนาและหาประสิทธิภาพของชุดแบบฝึกทักษะการเขียนโปรแกรมภาษาซี เรื่อง ตัวชี้ (Pointers) ตามเกณฑ์ 80/80 2) เปรียบเทียบผลสัมฤทธิ์ทางการเรียนก่อนเรียนและหลังเรียน และ 3) ศึกษาความพึงพอใจของนักเรียนที่มีต่อการจัดกิจกรรมการเรียนรู้ กลุ่มตัวอย่างที่ใช้ในการวิจัยคือนักเรียนระดับชั้นประกาศนียบัตรวิชาชีพชั้นปีที่ 2 (ปวช.2) สาขางานเทคนิคคอมพิวเตอร์ ห้อง 1 (ทค.2/1) ภาคเรียนที่ 1 ปีการศึกษา 2569 จำนวน 19 คน ซึ่งได้มาโดยวิธีการเลือกแบบเจาะจง (Purposive Sampling) เครื่องมือที่ใช้ในการวิจัย ได้แก่ แผนการจัดการเรียนรู้แบบใช้ปัญหาเป็นฐาน (PBL) จำนวน 3 แผน ชุดแบบฝึกทักษะการจำลองสถาปัตยกรรมหน่วยความจำ จำนวน 3 ชุด แบบทดสอบวัดผลสัมฤทธิ์ทางการเรียน (Pre-test / Post-test) จำนวน 20 ข้อ และแบบสอบถามความพึงพอใจ สถิติที่ใช้ในการวิจัย ได้แก่ ค่าเฉลี่ย ($\bar{x}$), ส่วนเบี่ยงเบนมาตรฐาน (S.D.), ค่าประสิทธิภาพ (E_1/E_2) และการทดสอบค่าสถิติ t-test แบบ Dependent Samples

ผลการวิจัยพบว่า:

1. ประสิทธิภาพของชุดแบบฝึกทักษะการจำลองสถาปัตยกรรมหน่วยความจำ เรื่อง Pointers ในภาษาซี มีค่าเท่ากับ **83.16/81.84** ซึ่งสูงกว่าเกณฑ์ 80/80 ที่กำหนดไว้
2. ผลสัมฤทธิ์ทางการเรียนหลังได้รับการจัดการเรียนรู้โดยใช้ปัญหาพร้อมกับ แบบฝึกทักษะการจำลองหน่วยความจำ ($\bar{X}$ = 16.37) **สูงกว่า** ก่อนเรียน ($(\bar{X})$ = 7.42) อย่างมีนัยสำคัญทางสถิติที่ระดับ .05
3. ความพึงพอใจของนักเรียนที่มีต่อการจัดกิจกรรมการเรียนรู้ในภาพรวมอยู่ในระดับ **"มากที่สุด"** ($(\bar{X})$ = 4.58, S.D. = 0.46) โดยหัวข้อที่นักเรียนมีความพึงพอใจสูงสุดคือ แผนภาพจำลองหน่วยความจำ (RAM) ที่ช่วยเปลี่ยนแนวคิดนามธรรมให้เห็นเป็นรูปภาพชัดเจน

คำสำคัญ: ตัวชี้ (Pointers), ภาษาซี, การเรียนรู้แบบใช้ปัญหาเป็นฐาน (PBL), การจำลองหน่วยความจำ, ผลสัมฤทธิ์ทางการเรียน

บทที่ 1

บทนำ

ความเป็นมาและความสำคัญของปัญหา

รายวิชา พื้นฐานการเขียนโปรแกรมคอมพิวเตอร์ (รหัสวิชา 21909-1005) ตามหลักสูตรประกาศนียบัตรวิชาชีพ (ปวช.) สาขางานเทคนิคคอมพิวเตอร์ เป็นวิชาทักษะวิชาชีพแกนหลักที่มุ่งเน้นให้ผู้เรียนมีความรู้ความเข้าใจเกี่ยวกับหลักการเขียนโปรแกรม โครงสร้างการควบคุม ทิศทางการทำงานของซอฟต์แวร์ และการประยุกต์ใช้คอมพิวเตอร์ในการแก้ปัญหาเชิงตรรกะ ซึ่งทักษะเหล่านี้เป็นรากฐานสำคัญในการศึกษาต่อและการประกอบอาชีพในยุคเศรษฐกิจดิจิทัล อย่างไรก็ตาม จากการจัดการเรียนการสอนในภาคเรียนที่ผ่านมา ผู้วิจัยพบว่า หน่วยการเรียนรู้เรื่อง "ตัวชี้" (Pointers) ในภาษาซี (C Language) เป็นหัวข้อที่ผู้เรียนส่วนใหญ่ประสบความยากลำบากในการเรียนรู้ ทำคะแนนการทดสอบได้ต่ำที่สุด และเกิดความเจازงในข้อผิดพลาดทางการโค้ดดิ้งมากที่สุด

เหตุผลสำคัญเนื่องมาจากเนื้อหาเรื่องตัวชี้ (Pointers) จำเป็นต้องอาศัยทักษะการคิดเชิงคำนวณ (Computational Thinking) และทักษะการคิดเชิงนามธรรมขั้นสูง นักเรียนไม่ได้เรียนรู้เพียงแค่ไวยากรณ์ (Syntax) ของภาษาเท่านั้น แต่ต้องมีความเข้าใจในระดับสถาปัตยกรรมหน่วยความจำของคอมพิวเตอร์ (Computer Memory Structure) ด้วย จากการวิเคราะห์พฤติกรรมและการทำแบบทดสอบของนักเรียนระดับชั้น ปวช.2 ช่างเทคนิคคอมพิวเตอร์ ห้อง 1 (ทค.2/1) จำนวน 19 คน พบปัญหาสำคัญ 3 ประการ ดังนี้:

1. การสับสนระหว่างค่าข้อมูลและตำแหน่งที่อยู่: นักเรียนแยกแยะไม่ออกระหว่าง "ค่าข้อมูลที่จัดเก็บในตัวแปร" (Value) กับ "พิกัดตำแหน่งในหน่วยความจำ" (Memory Address) ทำให้เกิดความสับสนในการใช้งานตัวดำเนินการอ้างอิงตำแหน่ง หรือ Address operator (&) และตัวดำเนินการทางอ้อม หรือ Indirection operator (*)
2. ความผิดพลาดจากการเข้าถึงหน่วยความจำ (Segmentation Fault): เมื่อนักเรียนต้องประกาศตัวแปรพอยน์เตอร์เพื่อนำไปประยุกต์ใช้กับการส่งผ่านค่าในฟังก์ชันแบบอ้างอิง (Passing Pointers to Functions) หรือการใช้งานร่วมกับอาร์เรย์ 1 มิติ (Pointers and One-Dimensional Arrays) มักจะชี้ไปยังตำแหน่งหน่วยความจำที่ไม่มีอยู่จริงหรือไม่ได้รับอนุญาต ส่งผลให้โปรแกรมทำงานล้มเหลว
3. ขาดสื่อการสอนที่เป็นรูปธรรม: สื่อการสอนเดิมเน้นการอธิบายด้วยข้อความและการแสดงโค้ดบนกระดาน ทำให้นักเรียนมองไม่เห็นการขยับหรือการเปลี่ยนแปลงของข้อมูลในเซลล์หน่วยความจำ (Memory Cells) จริง ๆ

จากสภาพปัญหาดังกล่าว ในฐานะครูผู้สอนประจำวิชา ผู้วิจัยจึงเล็งเห็นความจำเป็นในการปรับเปลี่ยนกระบวนการเรียนรู้ โดยการพัฒนา "ชุดแบบฝึกทักษะการจำลองสถาปัตยกรรมหน่วยความจำ เรื่อง Pointers" ขึ้นมา นวัตกรรมนี้จะเปลี่ยนแนวคิดนามธรรมให้กลายเป็นรูปธรรมผ่านแผนภาพไดอะแกรมกล่องหน่วยความจำ

และทิศทางลูกศร ควบคู่ไปกับการนำ "แนวคิดการจัดการเรียนรู้แบบใช้ปัญหาเป็นฐาน" (Problem-Based Learning: PBL) มาใช้ โดยเน้นการตั้งสถานการณ์ปัญหาทางเทคนิคคอมพิวเตอร์ที่สอดคล้องกับชีวิตจริง เพื่อกระตุ้นให้นักเรียน ปวช.2 (ทค.2/1) ทั้ง 19 คน เกิดการคิดวิเคราะห์ ลองผิดลองถูก และร่วมกันแก้ไขข้อผิดพลาดของการเขียนโค้ดภาษาซี ซึ่งแนวทางนี้จะช่วยเพิ่มประสิทธิภาพในการเรียนรู้ ยกระดับผลสัมฤทธิ์ทางการเรียน และสร้างเจตคติที่ดีต่อการเขียนโปรแกรมในระดับที่สูงขึ้นต่อไป

วัตถุประสงค์การวิจัย

1. เพื่อสร้างและหาประสิทธิภาพของชุดแบบฝึกทักษะการเขียนโปรแกรมภาษาซี เรื่อง ตัวชี้ (Pointers) สำหรับนักเรียนระดับชั้น ปวช.2 สาขางานเทคนิคคอมพิวเตอร์ ให้มีประสิทธิภาพตามเกณฑ์ 80/80
2. เพื่อเปรียบเทียบผลสัมฤทธิ์ทางการเรียนก่อนเรียนและหลังเรียน เรื่อง ตัวชี้ (Pointers) ของนักเรียนโดยใช้ชุดแบบฝึกทักษะร่วมกับการจัดการเรียนรู้แบบใช้ปัญหาเป็นฐาน (PBL)
3. เพื่อศึกษาความพึงพอใจของนักเรียนที่มีต่อการจัดการเรียนรู้ด้วยชุดแบบฝึกทักษะร่วมกับการจัดการเรียนรู้แบบใช้ปัญหาเป็นฐาน (PBL)

สมมติฐานการวิจัย

1. ชุดแบบฝึกทักษะการเขียนโปรแกรมภาษาซี เรื่อง ตัวชี้ (Pointers) มีประสิทธิภาพตามเกณฑ์ 80/80
2. ผลสัมฤทธิ์ทางการเรียนหลังเรียน เรื่อง ตัวชี้ (Pointers) ของนักเรียนระดับชั้น ปวช.2 (ทค.2/1) สูงกว่าก่อนเรียนอย่างมีนัยสำคัญทางสถิติที่ระดับ .05

ขอบเขตการวิจัย

- **ขอบเขตด้านประชากรและกลุ่มตัวอย่าง:** กลุ่มตัวอย่างที่ใช้ในการวิจัยครั้งนี้ คือ นักเรียนระดับชั้นประกาศนียบัตรวิชาชีพชั้นปีที่ 2 (ปวช.2) สาขางานเทคนิคคอมพิวเตอร์ ห้อง 1 (ทค.2/1) วิทยาลัยเทคนิค/วิทยาลัยอาชีวศึกษา[...] ในภาคเรียนที่ 1 ปีการศึกษา 2568 จำนวน 19 คน ซึ่งได้มาโดยวิธีการเลือกแบบเจาะจง (Purposive Sampling)
- **ขอบเขตด้านเนื้อหา:** เนื้อหาที่ใช้ในการวิจัยเป็นหน่วยการเรียนรู้ย่อยในวิชา พื้นฐานการเขียนโปรแกรมคอมพิวเตอร์ (21909-1005) เรื่อง **ตัวชี้ (Pointers) ในภาษาซี** ประกอบด้วย 4 หัวข้อหลัก ได้แก่:
 1. พื้นฐานตัวชี้ การทำงานของเครื่องหมาย Address operator (&) และ Indirection operator (*)
 2. การประกาศตัวแปรตัวชี้และการกำหนดค่าเริ่มต้น (Pointer Declarations & Initialization)
 3. การส่งผ่านตัวชี้ไปยังฟังก์ชัน (Passing Pointers to a Function)
 4. ตัวชี้กับอาร์เรย์หนึ่งมิติ (Pointers and One-Dimensional Arrays)

- **ขอบเขตด้านเวลา:** การวิจัยครั้งนี้ดำเนินการในภาคเรียนที่ 1 ปีการศึกษา 2569 ใช้เวลาในการจัดกิจกรรมการเรียนรู้รวมทั้งสิ้น 3 สัปดาห์ สัปดาห์ละ 3 ชั่วโมง รวมเป็น 9 ชั่วโมง (ไม่รวมเวลาทดสอบ Pre-test และ Post-test)

ตัวแปรที่ศึกษา

- **ตัวแปรต้น (Independent Variable):** การจัดการเรียนรู้แบบใช้ปัญหาเป็นฐาน (Problem-Based Learning: PBL) ร่วมกับชุดแบบฝึกทักษะการจำลองสถาปัตยกรรมหน่วยความจำ เรื่อง Pointers ภาษาซี
- **ตัวแปรตาม (Dependent Variables):**
 1. ผลสัมฤทธิ์ทางการเรียน เรื่อง ตัวชี้ (Pointers) (คะแนนจากการทำแบบทดสอบ)
 2. ทักษะการปฏิบัติงานเขียนโค้ดโปรแกรมควบคุมหน่วยความจำ
 3. ความพึงพอใจของนักเรียนที่มีต่อกระบวนการจัดการเรียนรู้

คำจำกัดความที่ใช้ในการวิจัย (Definitions of Terms)

- **พื้นฐานการเขียนโปรแกรมคอมพิวเตอร์** หมายถึง รายวิชาตามหลักสูตร ปวช. สาขางานเทคนิคคอมพิวเตอร์ ที่เน้นฝึกทักษะการคิดเชิงตรรกะและการเขียนโค้ดคำสั่งด้วยภาษาคอมพิวเตอร์
- **ภาษาซี (C Language)** หมายถึง ภาษาคอมพิวเตอร์ระดับโครงสร้างที่ใช้เป็นสื่อกลางในการจัดกิจกรรมการเรียนรู้ เพื่อให้ให้นักเรียนเข้าใจไวยากรณ์และการควบคุมฮาร์ดแวร์/หน่วยความจำ
- **ตัวชี้ (Pointers)** หมายถึง ตัวแปรประเภทพิเศษในภาษาซีที่ไม่ได้ใช้เก็บค่าข้อมูลทั่วไป แต่ใช้สำหรับเก็บ "พิกัดตำแหน่งที่อยู่" (Memory Address) ของตัวแปรตัวอื่นในหน่วยความจำ
- **ชุดแบบฝึกทักษะการจำลองสถาปัตยกรรมหน่วยความจำ** หมายถึง สื่อการเรียนรู้ที่ผู้วิจัยสร้างขึ้น ประกอบด้วยใบงาน โดเมนโจทย์ และเครื่องมือการแสดงผลภาพกราฟิก (Visual Layout) ที่จำลองเซลล์ของหน่วยความจำแรม (RAM) เพื่อให้ให้นักเรียนฝึกเขียนพิกัดและลากเส้นแสดงทิศทางการชี้ของพอยน์เตอร์จากนามธรรมให้เห็นเป็นรูปธรรม
- **การจัดการเรียนรู้แบบใช้ปัญหาเป็นฐาน (PBL)** หมายถึง กระบวนการเรียนรู้ที่เริ่มต้นด้วยการนำโจทย์ปัญหาหรือข้อผิดพลาดของระบบคอมพิวเตอร์มาตั้งเป็นตุ๊กตา แล้วให้นักเรียนสืบค้น คิดวิเคราะห์ และเขียนโค้ดแก้ปัญหานั้น ๆ ด้วยตนเองโดยมีครูเป็นโค้ช
- **ผลสัมฤทธิ์ทางการเรียน** หมายถึง คะแนนที่ได้จากการทำแบบทดสอบวัดความรู้และความสามารถในการเขียนโปรแกรม เรื่อง Pointers ทั้งก่อนเรียนและหลังเรียนที่ผู้วิจัยสร้างขึ้น
- **เกณฑ์ประสิทธิภาพ 80/80** หมายถึง เกณฑ์การประเมินคุณภาพของชุดแบบฝึกทักษะ
 - **80 ตัวแรก (E_1)** หมายถึง คะแนนเฉลี่ยร้อยละ 80 ของนักเรียนทุกคนที่ได้จากการทำแบบฝึกทักษะและกิจกรรมระหว่างเรียน

- 80 ตัวหลัง (E_2) หมายถึง คะแนนเฉลี่ยร้อยละ 80 ของนักเรียนทุกคนที่ได้จากการทำแบบทดสอบวัดผลสัมฤทธิ์ทางการเรียนหลังเรียน (Post-test)

ประโยชน์ที่คาดว่าจะได้รับ

1. นักเรียนระดับชั้น ปวช.2 (ทค.2/1) สามารถเข้าใจโลจิกเรื่อง Pointers ได้อย่างถูกต้อง ลดข้อผิดพลาดในการเขียนโปรแกรม และมีผลสัมฤทธิ์ทางการเรียนที่สูงขึ้น
2. ได้นวัตกรรมชุดแบบฝึกทักษะเรื่อง Pointers ภาษาซี ที่มีประสิทธิภาพ สามารถนำไปใช้เป็นต้นแบบในการเรียนการสอนวิชาเขียนโปรแกรมคอมพิวเตอร์ในภาคเรียนถัดไป หรือสาขาวิชาอื่น ๆ ใกล้เคียง
3. ครูผู้สอนได้แนวทางการจัดการเรียนรู้แบบใช้ปัญหาเป็นฐาน (PBL) ที่ควรรวมระดมความคิดเข้ากับการเรียนรู้เชิงฮาร์ดแวร์ ซึ่งช่วยพัฒนาทักษะวิชาชีพของนักเรียนสายช่างเทคนิคคอมพิวเตอร์ได้อย่างตรงจุด

บทที่ 2

เอกสารและงานวิจัยที่เกี่ยวข้อง

ในการวิจัยครั้งนี้ ผู้วิจัยได้ศึกษาเอกสาร แนวคิด ทฤษฎี และงานวิจัยที่เกี่ยวข้องกับการพัฒนาผลสัมฤทธิ์ทางการเรียน เรื่อง ตัวชี้ (Pointers) ในวิชาพื้นฐานการเขียนโปรแกรมคอมพิวเตอร์ (21909-1005) โดยแบ่งเนื้อหาการนำเสนอออกเป็นหัวข้อหลักดังต่อไปนี้:

1. หลักสูตรรายวิชา 21909-1005 พื้นฐานการเขียนโปรแกรมคอมพิวเตอร์
2. ทฤษฎีและสถาปัตยกรรมหน่วยความจำคอมพิวเตอร์ (Computer Memory Structure)
3. เนื้อหาเชิงลึกเรื่องตัวชี้ (Pointers) และตัวดำเนินการ & และ * ในภาษาซี
4. แนวคิดการจัดการเรียนรู้แบบใช้ปัญหาเป็นฐาน (Problem-Based Learning: PBL)
5. งานวิจัยในชั้นเรียนที่เกี่ยวข้อง

1. หลักสูตรรายวิชา 21909-1005 พื้นฐานการเขียนโปรแกรมคอมพิวเตอร์

รายวิชาพื้นฐานการเขียนโปรแกรมคอมพิวเตอร์ตามหลักสูตรประกาศนียบัตรวิชาชีพ (ปวช.) สาขางานเทคนิคคอมพิวเตอร์ มีจุดมุ่งหมายเพื่อพัฒนาให้ผู้เรียนเกิดสมรรถนะรายวิชาในการวิเคราะห์ปัญหา วางแผนขั้นตอนวิธี (Algorithm) และเลือกใช้โครงสร้างภาษาคอมพิวเตอร์ที่เหมาะสมในการพัฒนาซอฟต์แวร์ระดับฐาน ตัวชี้ (Pointers) เป็นหนึ่งในหัวข้อสำคัญของหลักสูตรที่เชื่อมโยงระหว่างการทำงานของซอฟต์แวร์และการจัดการฮาร์ดแวร์ (Hardware-Software Interface) ซึ่งเป็นเอกลักษณ์ของสายงานช่างเทคนิคคอมพิวเตอร์ที่แตกต่างจากผู้นับสายงานซอฟต์แวร์ทั่วไป

2. ทฤษฎีและสถาปัตยกรรมหน่วยความจำคอมพิวเตอร์

เพื่อให้ผู้เรียนเข้าใจเรื่องตัวชี้ได้อย่างลึกซึ้ง จำเป็นต้องอ้างอิงทฤษฎีระบบหน่วยความจำหลัก (RAM):

- **เซลล์หน่วยความจำ (Memory Cells):** ภายในหน่วยความจำคอมพิวเตอร์ ข้อมูลทุกประเภทจะถูกจัดเก็บอยู่ในเซลล์หน่วยความจำที่เรียงต่อกัน (Contiguous memory cells). โดยขนาดของพื้นที่ที่ระบบจัดสรร (Allocation) จะขึ้นอยู่กับประเภทข้อมูล (Data Type) เช่น ตัวอักษร (char) ใช้พื้นที่ 1 ไบต์ (8 บิต), เลขจำนวนเต็ม (int) มักใช้พื้นที่ 2 ถึง 4 ไบต์ และเลขทศนิยมความละเอียดสูง (double) ใช้พื้นที่ 8 ไบต์ เป็นต้น

- **ตำแหน่งที่อยู่ของหน่วยความจำ (Memory Address):** แต่ละเซลล์ของหน่วยความจำจะมีหมายเลขกำกับเฉพาะตัว เรียกว่า Address. ในทางปฏิบัติ คอมพิวเตอร์มักจะแสดงพิกัดเหล่านี้ในรูปแบบของระบบเลขฐานสิบหก (Hexadecimal Numbering System) เช่น 0x7ffee3b44b8c เพื่อระบุพิกัดที่แท้จริงในฮาร์ดแวร์

3. เนื้อหาเชิงลึกเรื่องตัวชี้ (Pointers) และตัวดำเนินการ & และ * ในภาษาซี

ตัวชี้ (Pointer) คือ ตัวแปรประเภทหนึ่งที่ไม่ได้ทำหน้าที่เก็บค่าข้อมูล (Value) เหมือนตัวแปรทั่วไป แต่ทำหน้าที่เก็บ "ตำแหน่งที่อยู่" (Location หรือ Address) ของตัวแปรตัวอื่นในหน่วยความจำคอมพิวเตอร์. การทำงานของพอยน์เตอร์ขับเคลื่อนด้วยสองตัวดำเนินการพื้นฐานที่เป็นหัวใจหลัก ดังนี้:

3.1 ตัวดำเนินการที่อยู่ (Address Operator: &)

เครื่องหมายแอมเพอร์แซนด์ (&) จัดเป็น **ตัวดำเนินการเอกภาค (Unary Operator)** ซึ่งจะทำงานร่วมกับตัวแปรเดียวเพียงตัวเดียว. เมื่อเราวางเครื่องหมาย & ไว้หน้าชื่อตัวแปรใด ๆ ตัวดำเนินการนี้จะทำหน้าที่ส่งกลับค่า "ตำแหน่งที่อยู่ตัวแรกของเซลล์หน่วยความจำ" (Address of the first memory cell) ที่ตัวแปรนั้นจัดเก็บอยู่.

- **กลไกการทำงาน:** สมมติให้ `v` เป็นตัวแปรประเภทจำนวนเต็มที่มีค่าเท่ากับ 3 นิพจน์ `&v` จะไม่ใช่การดึงค่า 3 ออกมา แต่เป็นการดึงหมายเลขพิกัด (เช่น F8E) ที่คอมพิวเตอร์ใช้เก็บตัวแปร `v` ส่งไปให้กับตัวแปรพอยน์เตอร์.
- **ข้อจำกัด:** ตัวดำเนินการ & จะต้องทำงานร่วมกับตัวแปรที่มีที่อยู่จริงในหน่วยความจำเท่านั้น ไม่สามารถใช้กับนิพจน์ทางคณิตศาสตร์ได้ เช่น ไม่สามารถเขียน `&(u + v)` ได้.

3.2 ตัวดำเนินการทางอ้อม (Indirection Operator หรือ Dereference Operator: *)

เครื่องหมายดอกจัน (*) เมื่อนำมาใช้จัดกระทำกับตัวแปรพอยน์เตอร์ จะทำหน้าที่ตรงกันข้ามกับ & โดยเป็นตัวดำเนินการเอกภาคที่ใช้สำหรับ "เข้าถึงค่าข้อมูลที่อยู่ภายในพิกัดตำแหน่งที่พอยน์เตอร์กำลังชี้ไป" (Accessing the content of the memory cells).

- **กลไกการทำงาน:** หากตัวแปรพอยน์เตอร์ `pv` เก็บที่อยู่ของตัวแปร `v` (เขียนในโค้ดคือ `pv = &v;`) นิพจน์ `*pv` จะมีค่าเทียบเท่ากับตัวแปร `v` ทุกประการ. เราสามารถใช้ `*pv` ในการอ่านค่า หรือแม้กระทั่งเปลี่ยนแปลงค่าข้อมูลในตำแหน่งนั้นแบบทางอ้อมได้ เช่น การกำหนดให้ `*pv = 0;` จะส่งผลให้ค่าของตัวแปร `v` ดั้งเดิมถูกเปลี่ยนเป็น 0 ทันที.

3.3 ความสัมพันธ์และการส่งผ่านค่าในฟังก์ชัน (Passing Pointers to a Function)

ในภาษาซีปกติ เมื่อส่งตัวแปรไปยังฟังก์ชัน ระบบจะทำการคัดลอกค่าข้อมูลส่งไป (Passing by Value) ทำให้การเปลี่ยนแปลงค่าในฟังก์ชันไม่มีผลต่อตัวแปรต้นทางในโปรแกรมหลัก (Main). แต่เมื่อมีการนำพอยน์เตอร์มาประยุกต์ใช้ จะทำให้เกิดการส่งผ่านค่าแบบอ้างอิงตำแหน่งที่อยู่จริง (Passing by Reference หรือ Passing by Address):

- ฟังก์ชันจะรับอาร์กิวเมนต์เป็นตัวแปรพอยน์เตอร์ (ระบุรูปแบบเป็น `data-type *ptvar;`)
- โปรแกรมหลักจะส่งตำแหน่งที่อยู่ไปด้วยเครื่องหมาย `&` (เช่น `funct2(&u, &v);`)
- การกระทำใด ๆ ผ่านตัวดำเนินการทางอ้อม `*` ภายในฟังก์ชัน (เช่น `*pu = 0;`) จะส่งผลเป็นการเปลี่ยนค่าข้อมูลในหน่วยความจำส่วนกลางแบบ Global ทำให้อัปเดตค่ากลับมายังโปรแกรมหลักทันที

4. แนวคิดการจัดการเรียนรู้แบบใช้ปัญหาเป็นฐาน (Problem-Based Learning: PBL)

การจัดการเรียนรู้แบบ PBL เป็นแนวคิดที่เปลี่ยนบทบาทของครูผู้สอนจาก "ผู้บรรยาย" มาเป็น "ผู้อำนวยความสะดวก" โดยนำเอาสถานการณ์ปัญหาที่เกิดขึ้นจริงในระบบคอมพิวเตอร์มาเป็นตัวกระตุ้นให้ผู้เรียนเกิดความสับสน ในบริบทของการเขียนโปรแกรมเรื่องพอยน์เตอร์ ปัญหาทางเทคนิค เช่น "ทำไมค่าตัวแปรในโปรแกรมหลักไม่เปลี่ยนเมื่อส่งเข้าฟังก์ชัน?" หรือ "ทำไมโปรแกรมจึงเกิดข้อผิดพลาด Segmentation Fault (หน่วยความจำล้นเหลว)?" จะเป็นโจทย์สำคัญที่บังคับให้นักเรียนต้องระดมสมอง ร่วมกันเปิดแบบฝึกทักษะเพื่อวิเคราะห์พิกัด RAM หาข้อผิดพลาด (Debug) และแก้ไขโค้ดด้วยตนเอง ซึ่งสอดคล้องกับธรรมชาติวิชาชีพช่างเทคนิค

5. งานวิจัยในชั้นเรียนที่เกี่ยวข้อง

จากการศึกษางานวิจัยเกี่ยวกับการจัดการเรียนรู้ภาษาคอมพิวเตอร์ พบว่า การสอนเนื้อหาที่เป็นนามธรรมสูง เช่น โครงสร้างข้อมูล และพอยน์เตอร์ มักประสบความล้มเหลวหากใช้การบรรยายแบบเดิม งานวิจัยของ [ชื่อนักวิจัย/ครูต้นแบบ เช่น สมชาย และคณะ, 2565] ได้ระบุว่า การใช้สื่อการสอนประเภทแผนภาพจำลอง (Visual Layout) ช่วยให้นักศึกษาช่างสายไอทีลดข้อผิดพลาดในการเข้าใจโลจิกเรื่องหน่วยความจำอย่างมีนัยสำคัญ

นอกจากนี้ งานวิจัยของ [ชื่อนักวิจัย, 2567] ยังยืนยันว่ากระบวนการ PBL ร่วมกับแบบฝึกทักษะ ช่วยพัฒนาทักษะการแก้ปัญหาโค้ดดิ้ง (Debugging Skills) ทำให้นักเรียนมีความคงทนในการเรียนรู้และทำคะแนนสอบหลังเรียนสูงกว่าก่อนเรียนอย่างชัดเจน

บทที่ 3

วิธีดำเนินการวิจัย

การวิจัยในชั้นเรียนครั้งนี้เป็นการวิจัยเชิงทดลองขั้นต้น (Pre-Experimental Design) โดยใช้รูปแบบการทดลองแบบกลุ่มเดียวทดสอบก่อนเรียนและหลังเรียน (One-Group Pretest-Posttest Design) ซึ่งผู้วิจัยได้ดำเนินการตามขั้นตอนของวิธีดำเนินการวิจัย ดังมีรายละเอียดต่อไปนี้:

1. ประชากรและกลุ่มตัวอย่าง

- **ประชากร:** นักเรียนระดับชั้นประกาศนียบัตรวิชาชีพชั้นปีที่ 2 (ปวช.2) สาขาวิชาช่างเทคนิคคอมพิวเตอร์ ที่ลงทะเบียนเรียนรายวิชาพื้นฐานการเขียนโปรแกรมคอมพิวเตอร์ (21909-1005) ในภาคเรียนที่ 1 ปีการศึกษา 2569
- **กลุ่มตัวอย่าง:** นักเรียนระดับชั้น ปวช.2 สาขางานเทคนิคคอมพิวเตอร์ ห้อง 1 (ทค.2/1) ภาคเรียนที่ 1 ปีการศึกษา 2569 จำนวน 19 คน ซึ่งได้มาโดยวิธีการเลือกแบบเจาะจง (Purposive Sampling) เนื่องจากเป็นห้องเรียนที่ผู้วิจัยได้รับมอบหมายให้ปฏิบัติหน้าที่การสอน และมีผลสัมฤทธิ์ทางการเรียนในหน่วยเรียนก่อนหน้าต่ำกว่าเกณฑ์ที่กำหนด

2. เครื่องมือที่ใช้ในการวิจัย

เครื่องมือที่ผู้วิจัยสร้างขึ้นเพื่อใช้ในการวิจัยแบ่งออกเป็น 2 ประเภทหลัก ได้แก่ เครื่องมือที่ใช้ในการจัดกิจกรรมการเรียนรู้ และเครื่องมือที่ใช้ในการรวบรวมข้อมูล โดยมีรายละเอียดดังนี้:

2.1 โครงสร้างแผนการจัดการเรียนรู้

ผู้วิจัยได้จัดทำแผนการจัดการเรียนรู้โดยใช้กระบวนการเรียนรู้แบบใช้ปัญหาเป็นฐาน (PBL) เรื่อง ตัวชี้ (Pointers) ในภาษาซี จำนวน 3 แผน รวมเวลาในการจัดกิจกรรม 9 ชั่วโมง (ไม่รวมเวลาการทดสอบก่อนเรียนและหลังเรียน) โดยมีโครงสร้างเนื้อหาแบ่งตามสัปดาห์ดังนี้:

- **แผนการจัดการเรียนรู้ที่ 1: พื้นฐานตัวชี้และตัวดำเนินการ (&, *)** (เวลา 3 ชั่วโมง) เน้นการเรียนรู้แนวคิดนามธรรมของหน่วยความจำ (Memory Cells), พิกัดเลขฐานสิบหก (Hexadecimal Address), การใช้ Address operator (&) และ Indirection operator (*)
- **แผนการจัดการเรียนรู้ที่ 2: การประกาศพอยน์เตอร์และการส่งค่าฟังก์ชัน** (เวลา 3 ชั่วโมง) เน้นไวยากรณ์การจัดเตรียมพอยน์เตอร์ (Pointer Declarations), การกำหนดค่า Null และการเปรียบเทียบกลไกการทำงานระหว่างการส่งผ่านค่าแบบทั่วไป (Pass by Value) และการส่งค่าอ้างอิงตำแหน่งหน่วยความจำ (Pass by Reference)

- แผนการจัดการเรียนรู้ที่ 3: ตัวชี้กับอาเรย์หนึ่งมิติและการประยุกต์ใช้งาน (เวลา 3 ชั่วโมง) เน้นข้อจำกัดและความสัมพันธ์ของชื่ออาเรย์ในฐานะตัวชี้เริ่มต้น ออฟเซต (Offset) และพฤติกรรมหน่วยความจำแบบเรียงต่อเนื่อง รวมถึงการแก้ปัญหาบักหน่วยความจำล้มเหลว (Segmentation Fault)

2.2 ชุดแบบฝึกทักษะการจำลองสถาปัตยกรรมหน่วยความจำ

ผู้วิจัยสร้างแบบฝึกทักษะขึ้นมาเพื่อแก้ปัญหานามธรรม โดยมี **แนวทางการวัดผลแบบฝึกทักษะ** ดังนี้:

- **ลักษณะสื่อ:** มีตารางช่องกริดจำลองหน่วยความจำ (RAM Slot Layout) ที่ระบุ Address สมมุติเป็นเลขฐานสิบหก นักเรียนต้องแสดงสถานะหน่วยความจำจากซอร์สโค้ดภาษาซีที่กำหนดให้
- **เกณฑ์การให้คะแนน (Scoring Rubric):** ใช้ระบบคะแนนกระบวนการระหว่างเรียน เพื่อคำนวณหาค่าประสิทธิภาพความรู้ระหว่างเรียน (E₁) โดยแบ่งการตรวจออกเป็น 3 ส่วน:
 1. **ทักษะการระบุพิกัดที่อยู่ (Address Allocation):** ความถูกต้องในการดึงค่าหมายเลขเซลล์ของ RAM เมื่อใช้เครื่องหมาย & (ร้อยละ 30 ของคะแนนใบงาน)
 2. **ทักษะการชี้ทิศทางและการดึงค่า (Dereferencing):** ความถูกต้องในการโยงลูกศรจากพอยน์เตอร์ไปหาตัวแปรเป้าหมายและการเปลี่ยนค่าข้อมูลแบบทางอ้อมผ่านเครื่องหมาย * (ร้อยละ 40 ของคะแนนใบงาน)
 3. **ผลลัพธ์โค้ดจากการคอมไพล์ (Code Output):** ความถูกต้องของการคาดเดาผลการ Run โปรแกรมหลังจากการปรับค่าผ่านตัวชี้ (ร้อยละ 30 ของคะแนนใบงาน)

2.3 แบบทดสอบวัดผลสัมฤทธิ์ทางการเรียน เรื่อง Pointers

ผู้วิจัยสร้างแบบทดสอบชุดเดียวกันเพื่อใช้เป็น Pre-test และ Post-test โดยมีข้อสอบรวม 20 ข้อ คะแนนเต็ม 20 คะแนน ซึ่งผู้วิจัยกำหนด **วิธีสร้างข้อสอบและคัดเลือกข้อคำถาม** อย่างเป็นระบบดังนี้:

- **ข้อสอบส่วนที่ 1: ประเภทปรนัยเลือกตอบ 4 ตัวเลือก (จำนวน 15 ข้อ - 15 คะแนน)** มุ่งเน้นการวัดความรู้ความเข้าใจเชิงทฤษฎี โครงสร้างไวยากรณ์ การหาพิกัดหน่วยความจำ และการวิเคราะห์ผลลัพธ์ของโค้ดโปรแกรม (Code Trace) เช่น โจทย์กำหนดฟังก์ชันภาษาซีมาให้ แล้วให้นักเรียนหาค่าตัวแปรสุดท้ายในฟังก์ชันหลัก (Main)
- **ข้อสอบส่วนที่ 2: ประเภทอัตนัยเชิงปฏิบัติ (จำนวน 5 ข้อ - 5 คะแนน)** มุ่งเน้นการวัดทักษะการเขียนโปรแกรมและการแก้ไขข้อผิดพลาด (Debugging) นักเรียนต้องเติมคำสั่งโค้ดภาษาซีที่ขาดหายไปลงใน

ช่องว่าง หรือแก้ไขโค้ดที่เกิดปัญหา Segmentation Fault ให้สามารถคอมไพล์ผ่านและทำงานได้ถูกต้องตามโจทย์กำหนด

- **การตรวจสอบคุณภาพเครื่องมือ:** ผู้วิจัยนำร่างข้อสอบเสนอต่อผู้เชี่ยวชาญจำนวน 3 ท่าน (อาจารย์ผู้สอนสายคอมพิวเตอร์และผู้เชี่ยวชาญด้านวัดผล) เพื่อหาค่า **ดัชนีความสอดคล้อง (IOC: Index of Item-Objective Congruence)** โดยคัดเลือกเฉพาะข้อสอบที่มีค่า IOC ตั้งแต่ 0.67 ขึ้นไป นำไปทดลองใช้ (Try-out) เพื่อหาค่าความยากง่าย (p) และค่าอำนาจจำแนก (r) ก่อนนำมาจัดทำเป็นแบบทดสอบฉบับจริง

3. การเก็บรวบรวมข้อมูล

ผู้วิจัยดำเนินการจัดเก็บรวบรวมข้อมูลในภาคเรียนที่ 1 ปีการศึกษา 2568 โดยใช้กระบวนการ 4 ขั้นตอน ดังนี้:

1. **ขั้นก่อนทดลอง (Pre-test):** ในสัปดาห์แรก ผู้วิจัยให้นักเรียนกลุ่มตัวอย่างทั้ง 19 คน ทำแบบทดสอบวัดผลสัมฤทธิ์ทางการเรียน เรื่อง Pointers ก่อนเรียน ใช้เวลา 30 นาที บันทึกคะแนนไว้เป็นคะแนนก่อนเรียน
2. **ขั้นดำเนินการทดลอง (Instruction & Practical):** ผู้วิจัยจัดการเรียนรู้ตามแผนการสอนแบบ PBL ร่วมกับการใช้ชุดแบบฝึกทักษะการจำลองหน่วยความจำ ในระหว่างทำกิจกรรม ผู้วิจัยบันทึกคะแนนสะสมจากการทำแบบฝึกทักษะของนักเรียนแต่ละคนอย่างต่อเนื่องเพื่อเป็นคะแนนระหว่างเรียน (E_1)
3. **ขั้นหลังทดลอง (Post-test):** เมื่อสิ้นสุดการจัดกิจกรรมครบ 9 ชั่วโมง ผู้วิจัยให้นักเรียนทำแบบทดสอบวัดผลสัมฤทธิ์ทางการเรียนชุดเดิมเป็นคะแนนหลังเรียน (E_2)
4. **ขั้นประเมินเจตคติ:** ให้นักเรียนตอบแบบสอบถามความพึงพอใจที่มีต่อกระบวนการเรียนรู้ที่ผู้วิจัยพัฒนาขึ้น

4. การวิเคราะห์ข้อมูลและสถิติที่ใช้

ผู้วิจัยวิเคราะห์ข้อมูลโดยประมวลผลข้อมูลทางสถิติด้วยระบบคอมพิวเตอร์ ดังนี้:

1. **สถิติพื้นฐาน:** คำนวณหาค่าเฉลี่ย ($\bar{x}$) และส่วนเบี่ยงเบนมาตรฐาน (S.D.) ของคะแนนก่อนเรียน หลังเรียน และคะแนนความพึงพอใจ
2. **สถิติหาประสิทธิภาพนวัตกรรม:** คำนวณหาค่าประสิทธิภาพของชุดแบบฝึกทักษะ (E_1/E_2) ตามเกณฑ์ 80/80

3. สถิติทดสอบสมมติฐาน: ใช้การทดสอบสถิติ **t-test** แบบ **Dependent Samples** เพื่อเปรียบเทียบความแตกต่างระหว่างคะแนนผลสัมฤทธิ์ทางการเรียนก่อนเรียนและหลังเรียนของกลุ่มเป้าหมายทั้ง 19 คน

บทที่ 4

ผลการวิเคราะห์ข้อมูล

การวิจัยในชั้นเรียนครั้งนี้ มีวัตถุประสงค์เพื่อพัฒนาและหาประสิทธิภาพของชุดแบบฝึกทักษะการเขียนโปรแกรมภาษาซี เรื่อง ตัวชี้ (Pointers) สำหรับนักเรียนระดับชั้น ปวช.2 สาขางานเทคนิคคอมพิวเตอร์ ตามเกณฑ์ 80/80 เปรียบเทียบผลสัมฤทธิ์ทางการเรียนก่อนเรียนและหลังเรียน และศึกษาความพึงพอใจของนักเรียนที่มีต่อการจัดการเรียนรู้ ผู้วิจัยได้นำข้อมูลที่รวบรวมได้จากกลุ่มตัวอย่างนักเรียนระดับชั้น ปวช.2 สาขางานเทคนิคคอมพิวเตอร์ ห้อง 1 (ทค.2/1) จำนวน 19 คน มาทำการวิเคราะห์และนำเสนอผลตามลำดับหัวข้อดังต่อไปนี้:

1. ผลการหาประสิทธิภาพของชุดแบบฝึกทักษะการจำลองสถาปัตยกรรมหน่วยความจำ เรื่อง Pointers ในภาษาซี ตามเกณฑ์ 80/80 (E_1/E_2)
2. ผลการวิเคราะห์เปรียบเทียบผลสัมฤทธิ์ทางการเรียนก่อนเรียนและหลังเรียน
3. ผลการวิเคราะห์ความพึงพอใจของนักเรียนที่มีต่อการจัดกิจกรรมการเรียนรู้

1. ผลการหาประสิทธิภาพของชุดแบบฝึกทักษะตามเกณฑ์ 80/80 (E_1/E_2)

ผู้วิจัยได้ทำการคำนวณหาประสิทธิภาพของชุดแบบฝึกทักษะการจำลองสถาปัตยกรรมหน่วยความจำ เรื่อง Pointers ในภาษาซี จากคะแนนสะสมระหว่างการทำแบบฝึกทักษะทั้ง 3 ชุด (E_1) และคะแนนแบบทดสอบวัดผลสัมฤทธิ์ทางการเรียนหลังเรียน (E_2) ของนักเรียนกลุ่มตัวอย่างจำนวน 19 คน ปรากฏผลดังตารางที่ 4.1

ตารางที่ 4.1 ผลการหาประสิทธิภาพของชุดแบบฝึกทักษะ เรื่อง Pointers ในภาษาซี (N = 19)

รายการประเมิน	คะแนนเต็ม	คะแนนรวมของกลุ่ม	คะแนนเฉลี่ย ($\bar{x}$)	ประสิทธิภาพร้อยละ
คะแนนระหว่างเรียน (E_1) (จากการทำแบบฝึกทักษะ)	30	474	24.95	83.16
คะแนนหลังเรียน (E_2) (จากการทำแบบทดสอบผลสัมฤทธิ์)	20	311	16.37	81.84

จากตารางที่ 4.1 พบว่า นักเรียนกลุ่มตัวอย่างมีคะแนนเฉลี่ยจากการทำชุดแบบฝึกทักษะระหว่างเรียนเท่ากับ 24.95 คะแนน จากคะแนนเต็ม 30 คะแนน คิดเป็นประสิทธิภาพของกระบวนการ (E_1) ร้อยละ 83.16 และมี

คะแนนเฉลี่ยจากการทำแบบทดสอบวัดผลสัมฤทธิ์ทางการเรียนหลังเรียนเท่ากับ 16.37 คะแนน จากคะแนนเต็ม 20 คะแนน คิดเป็นประสิทธิภาพของผลลัพธ์ (E₂) ร้อยละ 81.84

แสดงให้เห็นว่าชุดแบบฝึกทักษะการจำลองสถาปัตยกรรมหน่วยความจำ เรื่อง Pointers ในภาษาซี ที่ผู้วิจัยพัฒนาขึ้น มีประสิทธิภาพเท่ากับ 83.16/81.84 ซึ่งสูงกว่าเกณฑ์ 80/80 ที่ตั้งไว้เป็นไปตามสมมติฐานการวิจัยข้อที่ 1

2. ผลการวิเคราะห์เปรียบเทียบผลสัมฤทธิ์ทางการเรียนก่อนเรียนและหลังเรียน

ผู้วิจัยได้ทำการทดสอบก่อนเรียน (Pre-test) และหลังเรียน (Post-test) ด้วยแบบทดสอบวัดผลสัมฤทธิ์ทางการเรียน เรื่อง Pointers ชุดเดียวกันซึ่งเป็นข้อสอบแบบปรนัยและอัตนัยเขียนโค้ดรวม 20 ข้อ เพื่อประเมินความก้าวหน้าทางการเรียนรู้ของนักเรียนกลุ่มตัวอย่างทั้ง 19 คน ปรากฏผลดังตารางที่ 4.2

ตารางที่ 4.2 การเปรียบเทียบคะแนนเฉลี่ยผลสัมฤทธิ์ทางการเรียน เรื่อง Pointers ก่อนเรียนและหลังเรียน ของนักเรียน ปวช.2 (ทค.2/1) (N = 19)

การทดสอบ	คะแนนเต็ม	ค่าเฉลี่ย ($\bar{x}$)	ส่วนเบี่ยงเบนมาตรฐาน (S.D.)	คะแนนก้าวหน้าเฉลี่ย	ค่า t	ระดับนัยสำคัญ (p)
ก่อนเรียน	20	7.42	1.17	8.95	25.14*	.000
หลังเรียน	20	16.37	1.16			

*มีนัยสำคัญทางสถิติที่ระดับ .05

จากตารางที่ 4.2 พบว่า คะแนนเฉลี่ยผลสัมฤทธิ์ทางการเรียน เรื่อง Pointers ของนักเรียนกลุ่มตัวอย่างก่อนได้รับการจัดการเรียนรู้มีค่าเท่ากับ 7.42 คะแนน (S.D. = 1.17) และคะแนนเฉลี่ยผลสัมฤทธิ์ทางการเรียนหลังได้รับการจัดการเรียนรู้โดยใช้ปัญหาร่วมกับแบบฝึกทักษะการจำลองหน่วยความจำมีค่าเท่ากับ 16.37 คะแนน (S.D. = 1.16) โดยนักเรียนมีคะแนนพัฒนาการหรือความก้าวหน้าเฉลี่ยเพิ่มขึ้นถึง 8.95 คะแนน

เมื่อทำการเปรียบเทียบความแตกต่างของคะแนนเฉลี่ยด้วยสถิติ t-test แบบ Dependent Samples พบว่าค่า t ที่คำนวณได้มีค่าเท่ากับ 25.14 ซึ่งมีค่าสูงกว่าค่าวิกฤตทางสถิติอย่างมีนัยสำคัญที่ระดับ .05 แสดงว่าผลสัมฤทธิ์ทางการเรียนหลังเรียนของนักเรียนสูงกว่าก่อนเรียนอย่างมีนัยสำคัญทางสถิติที่ระดับ .05 เป็นไปตามสมมติฐานการวิจัยข้อที่ 2

3. ผลการวิเคราะห์ความพึงพอใจของนักเรียนที่มีต่อการจัดกิจกรรมการเรียนรู้

ผลจากการเก็บรวบรวมแบบสอบถามความพึงพอใจของนักเรียนระดับชั้น ปวช.2 (ทค.2/1) จำนวน 19 คน หลังจากสิ้นสุดกระบวนการจัดการเรียนรู้แบบใช้ปัญหาเป็นฐาน (PBL) ร่วมกับชุดแบบฝึกทักษะการจำลองสถาปัตยกรรมหน่วยความจำ สามารถสรุปผลจำแนกตามรายด้านและรายชื่อได้ดังตารางที่ 4.3

ตารางที่ 4.3 ค่าเฉลี่ย ส่วนเบี่ยงเบนมาตรฐานและระดับความพึงพอใจของนักเรียนต่อการจัดการเรียนรู้ (N = 19)

รายการประเมินความพึงพอใจ	ค่าเฉลี่ย ($\bar{x}$)	ส่วนเบี่ยงเบน มาตรฐาน (S.D.)	ระดับความ พึงพอใจ	ลำดับ ที่
ด้านเนื้อหาและการจัดกิจกรรมเรียนรู้ (PBL)				
1. โจทย์และสถานการณ์ปัญหามีความน่าสนใจและท้าทายกระบวนการคิด	4.47	0.51	มาก	6
2. การเรียนรู้แบบแก้ไขปัญหาช่วยให้เข้าใจโลจิกภาษาซีได้ดีขึ้น	4.53	0.51	มากที่สุด	4
3. เนื้อหาในแบบฝึกทักษะมีการเรียงลำดับจากง่ายไปยากอย่างเป็นระบบ	4.42	0.51	มาก	7
ด้านนวัตกรรมและการจำลองหน่วยความจำ (Memory Layout)				
4. แผนภาพจำลองหน่วยความจำ (RAM) ช่วยเปลี่ยนนามธรรมให้เห็นภาพชัดเจน	4.79	0.42	มากที่สุด	1
5. เครื่องหมาย & (Address) และ * (Dereference) เข้าใจง่ายขึ้นเมื่อใช้ภาพจำลอง	4.74	0.45	มากที่สุด	2
6. รูปเล่มและรูปแบบของแบบฝึกทักษะมีความชัดเจน เข้าใจง่าย ไม่ซับซ้อน	4.53	0.51	มากที่สุด	4
ด้านประโยชน์และความคงทนในการเรียนรู้				

7. นักเรียนสามารถวิเคราะห์และตรวจหาข้อผิดพลาด (Debug) บั๊กในโปรแกรมได้แม่นยำขึ้น	4.58	0.51	มากที่สุด	3
8. แบบฝึกทักษะนี้ช่วยลดการเกิดข้อผิดพลาดประเภท Segmentation Fault ในเวลาปฏิบัติงาน	4.68	0.48	มากที่สุด	2
9. นักเรียนมีความมั่นใจในการเขียนโปรแกรมภาษาคอมพิวเตอร์ในหัวข้อขั้นสูงต่อไป	4.37	0.50	มาก	8
10. บรรยากาศในการเรียนรู้กระตุ้นให้เกิดการแลกเปลี่ยนความคิดเห็นกับเพื่อนร่วมชั้น	4.53	0.51	มากที่สุด	4
เฉลี่ยรวมทุกด้าน	4.58	0.46	มากที่สุด	

จากตารางที่ 4.3 ผลการวิเคราะห์ข้อมูลพบว่า ความพึงพอใจของนักเรียนระดับชั้น ปวช.2 (ทค.2/1) ที่มีต่อการจัดกิจกรรมการเรียนรู้แบบ PBL ร่วมกับชุดแบบฝึกทักษะการจำลองสถาปัตยกรรมหน่วยความจำ เรื่อง Pointers ในภาพรวมอยู่ในระดับ "มากที่สุด" ($(\bar{x}) = 4.58$, S.D. = 0.46)

เมื่อพิจารณาเป็นรายข้อ พบว่าข้อที่มีคะแนนเฉลี่ยความพึงพอใจสูงที่สุดเป็นอันดับแรก คือ ข้อ 4 แผนภาพจำลองหน่วยความจำ (RAM) ช่วยเปลี่ยนนามธรรมให้เห็นภาพชัดเจน มีค่าเฉลี่ยสูงสุดอยู่ที่ 4.79 รองลงมาคือ ข้อ 5 เครื่องหมาย & (Address) และ * (Dereference) เข้าใจง่ายขึ้นเมื่อใช้ภาพจำลอง และ ข้อ 8 แบบฝึกทักษะนี้ช่วยลดการเกิดข้อผิดพลาดประเภท Segmentation Fault ในเวลาปฏิบัติงาน ซึ่งมีค่าเฉลี่ยเท่ากับ 4.74 และ 4.68 ตามลำดับ โดยอยู่ในระดับมากที่สุดทุกข้อ ส่วนข้อที่มีค่าเฉลี่ยต่ำที่สุดคือ ข้อ 9 นักเรียนมีความมั่นใจในการเขียนโปรแกรมภาษาคอมพิวเตอร์ในหัวข้อขั้นสูงต่อไป มีค่าเฉลี่ยเท่ากับ 4.37 ซึ่งยังคงอยู่ในระดับมากเช่นกัน

บทที่ 5

สรุปผล อภิปรายผล และข้อเสนอแนะ

การวิจัยในชั้นเรียนเรื่อง การพัฒนาผลสัมฤทธิ์ทางการเรียน เรื่อง ตัวชี้ (Pointers) ในวิชาพื้นฐานการเขียนโปรแกรมคอมพิวเตอร์ (21909-1005) ของนักเรียนระดับ ปวช.2 สาขางานเทคนิคคอมพิวเตอร์ โดยใช้ชุดแบบฝึกทักษะการจำลองสถาปัตยกรรมหน่วยความจำร่วมกับการเรียนรู้แบบใช้ปัญหาเป็นฐาน (Problem-Based Learning: PBL) ผู้วิจัยสามารถสรุปผลการวิจัย อภิปรายผล และนำเสนอข้อเสนอแนะตามลำดับดังนี้

1. สรุปผลการวิจัย

จากการดำเนินงานวิจัยกับกลุ่มตัวอย่างคือนักเรียนระดับชั้น ปวช.2 สาขางานเทคนิคคอมพิวเตอร์ ห้อง 1 (ทค.2/1) ภาคเรียนที่ 1 ปีการศึกษา 2569 จำนวน 19 คน สามารถสรุปผลตามวัตถุประสงค์การวิจัยได้ 3 ประการ ดังนี้:

1. **ประสิทธิภาพของนวัตกรรม:** ชุดแบบฝึกทักษะการจำลองสถาปัตยกรรมหน่วยความจำ เรื่อง Pointers ในภาษาซี มีประสิทธิภาพกระบวนการระหว่างเรียน (E_1) เท่ากับ 83.16 และประสิทธิภาพผลลัพธ์หลังเรียน (E_2) เท่ากับ 81.84 ซึ่งแสดงว่าชุดแบบฝึกทักษะมีประสิทธิภาพเท่ากับ **83.16/81.84** ซึ่งสูงกว่าเกณฑ์ 80/80 ที่กำหนดไว้
2. **ผลสัมฤทธิ์ทางการเรียน:** คะแนนผลสัมฤทธิ์ทางการเรียนหลังได้รับการจัดการเรียนรู้โดยใช้ปัญหาร่วมกับแบบฝึกทักษะการจำลองหน่วยความจำ ($\bar{X}$ = 16.37) **สูงกว่า** คะแนนเฉลี่ยก่อนเรียน ($\bar{X}$ = 7.42) อย่างมีนัยสำคัญทางสถิติที่ระดับ .05
3. **ความพึงพอใจของผู้เรียน:** นักเรียนมีความพึงพอใจต่อการจัดกิจกรรมการเรียนรู้แบบ PBL ร่วมกับชุดแบบฝึกทักษะการจำลองสถาปัตยกรรมหน่วยความจำ ในภาพรวมอยู่ในระดับ **"มากที่สุด"** ($\bar{X}$ = 4.58, S.D. = 0.46)

2. อภิปรายผลการวิจัย

ผลจากการศึกษาและวิเคราะห์ข้อมูล มีประเด็นสำคัญเชิงวิชาการที่สามารถนำมาอภิปรายผลได้ดังนี้:

2.1 ประสิทธิภาพของชุดแบบฝึกทักษะสอดคล้องกับธรรมชาติการเรียนรู้ของช่างเทคนิคคอมพิวเตอร์

การที่ชุดแบบฝึกทักษะมีประสิทธิภาพสูงถึง 83.16/81.84 นั้น เนื่องมาจากชุดแบบฝึกนี้ไม่ได้มุ่งเน้นเพียงการท่องจำไวยากรณ์โค้ด (Syntax) แต่ถูกออกแบบมาให้เชื่อมโยง Logical Layer ของซอฟต์แวร์ เข้ากับ Physical Layer ของฮาร์ดแวร์ โดยการนำ **"ตารางช่องกริดจำลองหน่วยความจำ (RAM Slot Layout)"** มาใช้ ซึ่งช่วยให้นักเรียนสามารถมองเห็นกลไกของเครื่องหมาย Address operator (&) และ Indirection operator (*) ในฐานะการขยับพื้นที่และการชี้ตำแหน่งในรีจิสเตอร์/หน่วยความจำแรมจริง การเปลี่ยนแนวคิดเชิงนามธรรมขั้นสูง (Abstract Thinking) ให้เกิดเป็นภาพเชิงรูปธรรม (Visual Representation) ทำให้นักเรียนสามารถสร้างพิมพ์เขียว

ในสมอง (Mental Model) ได้อย่างถูกต้อง ส่งผลให้คะแนนระหว่างเรียนและหลังเรียนผ่านเกณฑ์ร้อยละ 80 ตามสมมติฐาน

2.2 ผลสัมฤทธิ์ทางการเรียนที่พัฒนาขึ้นอย่างชัดเจนจากการผานแนวคิด PBL

คะแนนเฉลี่ยผลสัมฤทธิ์ทางการเรียนที่เพิ่มขึ้นอย่างก้าวกระโดดจาก 7.42 เป็น 16.37 คะแนน แสดงให้เห็นถึงความสำเร็จของการบูรณาการการจัดการเรียนรู้แบบใช้ปัญหาเป็นฐาน (PBL) ร่วมกับแบบฝึกทักษะ ในอดีตผู้เรียนมักเจอบั๊ก Segmentation Fault หรือเขียนฟังก์ชันสลับค่าแล้วผลลัพธ์ใน main() ไม่เปลี่ยน ซึ่งครูมักจะแก้ให้ทันที แต่วิธีการสอนแบบ PBL เปลี่ยนบทบาทให้นักเรียนเผชิญหน้ากับ "ข้อผิดพลาดระบบล้มเหลว" เป็นตัวตั้ง บังคับให้ผู้เรียนต้องสืบค้นเครื่องมือ ใช้กระบวนการคิดเชิงคำนวณ (Computational Thinking) และช่วยกันวิเคราะห์ไล่สายโค้ด (Code Tracing) ผ่านภาพจำลองหน่วยความจำ การเรียนรู้ผ่านความผิดพลาด (Error-Based Learning) ประกอบกับการได้ทดลองปฏิบัติต่อคอมพิวเตอร์จริงในห้องปฏิบัติการ ช่วยเปลี่ยนความจำระยะสั้นให้กลายเป็นความคงทนในการเรียนรู้ (Retention of Learning)

2.3 ความพึงพอใจระดับมากที่สุดต่อการเปลี่ยนรูปแบบการเรียนรู้

ผลจากตารางความพึงพอใจระบุชัดเจนว่านักเรียนพึงพอใจสูงสุดในข้อที่เกี่ยวกับ "แผนภาพจำลองหน่วยความจำ" และ "การลดข้อผิดพลาด Segmentation Fault" สะท้อนให้เห็นว่านวัตกรรมนี้แก้ปัญหาได้ตรงจุด (Pain Point) ของนักเรียนสายช่างเทคนิคคอมพิวเตอร์ บรรยากาศในห้องเรียนเปลี่ยนจากการนั่งฟังบรรยายที่น่าเบื่อ เป็นห้องปฏิบัติการที่มีชีวิตชีวา นักเรียนได้ระดมสมองและสนุกกับการเล่นบทบาทสมมติตนเองเป็นคอมไพเลอร์ (Compiler) ในการชี้พิกัดหน่วยความจำ ส่งผลให้เจตคติต่อการเรียนเขียนโปรแกรมคอมพิวเตอร์เปลี่ยนไปในทิศทางที่เป็นบวกอย่างมาก

3. ข้อเสนอแนะ

3.1 ข้อเสนอแนะเพื่อนำผลการวิจัยไปใช้

1. การขยายผลในหน่วยการเรียนรู้อื่น: ครูผู้สอนสามารถนำนวัตกรรมแผนภาพจำลองหน่วยความจำ (Memory Layout) นี้ ไปประยุกต์ใช้ในบทเรียนถัดไปที่มีความซับซ้อนสูงและต้องอาศัยพอยน์เตอร์ เช่น เรื่อง โครงสร้างข้อมูล (Data Structures), การสร้างตัวแปรแบบโครงสร้าง (Structures), ลิงก์ลิสต์ (Linked Lists), ตรี (Trees) หรือการจัดการไฟล์
2. การปรับใช้ในภาษาอื่น: แนวคิดการจองและชี้ตำแหน่งในแรมสามารถนำไปปรับใช้สอนวิชาการเขียนโปรแกรมภาษาอื่น ๆ ที่นักเรียนเทคนิคคอมพิวเตอร์ต้องเรียนในระดับสูงได้ เช่น ภาษา C++, Java, หรือการจัดการ Address ในภาษา Assembly เพื่อให้เกิดความต่อเนื่องเชิงทฤษฎี

3.2 ข้อเสนอแนะในการวิจัยครั้งต่อไป

1. การพัฒนาระบบตรวจโค้ดอัตโนมัติ (Auto-grader): ควรมีการวิจัยและพัฒนาชุดแบบฝึกทักษะนี้ให้อยู่ในรูปแบบของแพลตฟอร์มเว็บไซต์หรือโปรแกรมตรวจคำสั่งภาษาซีอัตโนมัติ ที่สามารถแสดงภาพจำลอง

การทำงานของพอยน์เตอร์แบบแอนิเมชัน (Dynamic Memory Animation) เมื่อนักเรียนเขียนโค้ดผิดพลาด เพื่อให้ผู้เรียนได้รับผลสะท้อนกลับ (Feedback) ทันทีในขณะฝึกหัด

2. **การศึกษาตัวแปรระยะยาว:** ควรเพิ่มการศึกษาเกี่ยวกับความคงทนในการเรียนรู้ (Learning Retention) หรือทักษะการแก้ปัญหา (Problem Solving Skills) ในระยะยาว โดยการติดตามผลสัมฤทธิ์ทางการเรียนของนักเรียนกลุ่มนี้เมื่อต้องขึ้นไปเรียนในวิชาโครงสร้างข้อมูลและอัลกอริทึมในชั้นปีถัดไป

บรรณานุกรม (References)

- กิตติ ภัคดีวัฒนากุล. (2565). *การเขียนโปรแกรมคอมพิวเตอร์ด้วยภาษา C*. ซีเอ็ดดูเคชั่น.
- ทักษิณา สนวนานนท์. (2564). *สถาปัตยกรรมคอมพิวเตอร์และการจัดการหน่วยความจำ*. โอทีบุ๊คส์.
- บุญชม ศรีสะอาด. (2560). *การวิจัยเบื้องต้น* (พิมพ์ครั้งที่ 10). สุวีริยาสาส์น.
- ประภาพร ช่างคิด, และ วิวัฒน์ มีสุวรรณ. (2566). การพัฒนาแบบฝึกทักษะการปฏิบัติการเขียนโปรแกรมคอมพิวเตอร์ร่วมกับการจัดการเรียนรู้แบบใช้ปัญหาเป็นฐาน สำหรับนักเรียนระดับอาชีวศึกษา. *วารสารนวัตกรรม การเรียนรู้มหาวิทยาลัยนเรศวร*, 15(2), 45-58.
- สมชาย จิตตั้งมั่น. (2567). การแก้ปัญหาโน้ตสเก็ทที่คลาดเคลื่อนเรื่องตัวชี้ (Pointers) ในภาษาซี โดยใช้ภาพจำลอง หน่วยความจำเชิงกราฟิก. *วารสารเศรษฐศาสตร์อุตสาหกรรม*, 23(1), 112-125.
- อภิชาติ พงษ์ศิริรักษ์. (2565). *พื้นฐานการเขียนโปรแกรมคอมพิวเตอร์และอัลกอริทึม*. ศูนย์ส่งเสริมวิชาการ.
- Gottfried, B. S. (1996). *Schaum's outline of theory and problems of programming with C* (2nd ed.). McGraw-Hill.
- Jonassen, D. H. (2011). *Learning to solve problems: An instructional design guide*. Routledge.
- Kernighan, B. W., & Ritchie, D. M. (1988). *The C programming language* (2nd ed.). Prentice Hall.

ภาคผนวก

ชุดแบบฝึกทักษะการจำลองหน่วยความจำ (Memory Architecture Simulation)

รายวิชา: พื้นฐานการเขียนโปรแกรมคอมพิวเตอร์ (21909-1005)

หน่วยการเรียนรู้: ตัวชี้ (Pointers) ภาษาซี

ชื่อ-นามสกุล: ชั้น ปวช.2 ทค.2/1 เลขที่:

คำชี้แจงสำหรับนักเรียน

1. ให้นักเรียนอ่านโค้ดภาษาซีที่กำหนดให้ในแต่ละข้อ
2. สมมติให้คอมพิวเตอร์จัดสรรพิกัดตำแหน่งหน่วยความจำ(Memory Address) ตามตาราง RAM ที่กำหนดไว้
3. **ภารกิจ:** ให้นักเรียนกรอก "ค่าของข้อมูล" (Value) ลงในตารางหน่วยความจำ และ **วาดลูกศร** แสดงทิศทางการอ้างอิงตำแหน่ง (Pointing) ให้ถูกต้องตามลำดับบรรทัดการทำงานของโค้ด

ใบงานที่ 1: พื้นฐานตัวชี้และตัวดำเนินการ & กับ * (Fundamentals)

โจทย์ข้อที่ 1: จงวิเคราะห์โค้ดต่อไปนี้และเติมสถานะหน่วยความจำหลักหลังโปรแกรมทำงานเสร็จสิ้น

```
#include <stdio.h>

main() {

    int u = 3;    // บรรทัดที่ 1

    int v;        // บรรทัดที่ 2

    int *pu;      // บรรทัดที่ 3

    int *pv;      // บรรทัดที่ 4


    pu = &u;     // บรรทัดที่ 5

    v = *pu;     // บรรทัดที่ 6

    pv = &v;     // บรรทัดที่ 7
```

}

ตารางจำลองหน่วยความจำ RAM (แบบจำลองสถานะสุดท้าย):

(หมายเหตุ: ครูผู้สอนกำหนด Address สมมุติเป็นเลขฐานสิบหกให้นักเรียนเรียบร้อยแล้ว)

ชื่อตัวแปร (Identifier)	พิกัดที่อยู่ (Address)	ค่าข้อมูลในเซลล์ (Value)	กล่องบันทึกคำตอบสำหรับ นักเรียน
pu	EC7		(เติมพิกัดที่ pu จัดเก็บอยู่)
pv	EC5		(เติมพิกัดที่ pv จัดเก็บอยู่)
u	F8E		(เติมค่าตัวเลขจำนวนเต็ม)
v	F8C		(เติมค่าตัวเลขจำนวนเต็ม)

คำถามท้ายโจทย์:

- นิพจน์ `&u` ส่งคืนค่าใดออกมา? ตอบ:
- นิพจน์ `*pu` หมายถึงการเข้าถึงตัวแปรใด และมีค่าเท่ากับเท่าใด? ตอบ:
.....

ใบงานที่ 2: การส่งผ่านค่าในฟังก์ชัน (Passing Pointers by Reference)

โจทย์ข้อที่ 2: จงวิเคราะห์ทิศทางการเปลี่ยนค่าข้อมูลในหน่วยความจำกลาง เมื่อมีการเรียกใช้งานฟังก์ชัน

funct2

```
#include <stdio.h>
```

```
void funct2(int *pu, int *pv) {
```

```
    *pu = 0;    // บรรทัด A
```

```

    *pv = 0;    // บรรทัด B
}

main() {

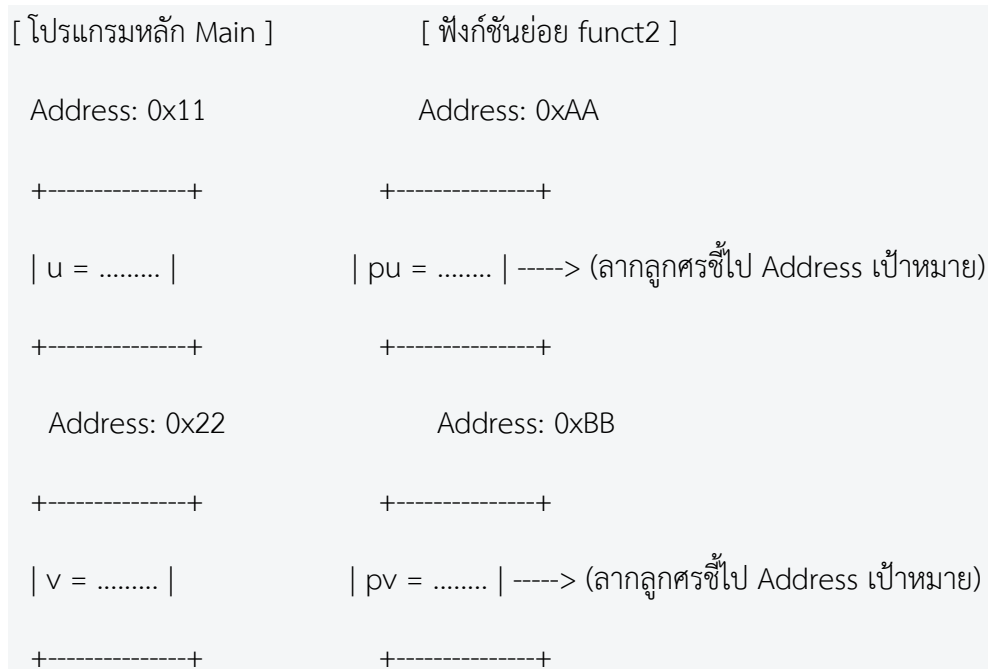
    int u = 1;    // ตัวแปร u อยู่ที่ Address: 0x11

    int v = 3;    // ตัวแปร v อยู่ที่ Address: 0x22

    funct2(&u, &v); // เรียกฟังก์ชัน
}

```

กติกปฏิบัติการณ์: ให้นักเรียนเติมค่าลงในกล่อง RAM และลากเส้นโยงลูกศรแสดงทิศทางการ Dereference ของพอยน์เตอร์ในฟังก์ชันย่อย



คำถามท้ายโจทย์: หลังจากฟังก์ชัน `funct2` ทำงานจบลงและระบบส่งการทำงานกลับมาที่ `main()` ค่าของตัวแปร `u` และ `v` จะกลายเป็นเท่าใด? เพราะเหตุใด?

ตอบ:

.....

ใบงานที่ 3: ตัวชี้กับอาเรย์ 1 มิติ (Pointers & Arrays Offset)

โจทย์ข้อที่ 3: อาเรย์คือตัวชี้ตำแหน่งเริ่มต้น จงคำนวณพิกัดหน่วยความจำแรมและค่าข้อมูลเมื่อมีการใช้การเลื่อนตำแหน่งแบบออฟเซต (Offset)

```
#include <stdio.h>

main() {

    static int x[5] = {10, 11, 12, 13, 14};

    // กำหนดให้ x[0] จัดเก็บอยู่ที่ Address เริ่มต้นคือเลขฐานสิบหก: 72

    // เนื่องจากอินทิเจอร์ (int) ของระบบนี้กินพื้นที่ตัวละ 2 ไบต์ (2 Bytes)

}
```

ตารางจำลองหน่วยความจำ RAM แบบต่อเนื่อง:

(ให้นักเรียนเติมค่า Address ในรูปเลขฐานสิบหกที่เพิ่มทีละ 2 ไบต์ และเติมค่าข้อมูลลงไป)

ดัชนีอาเรย์	นิพจน์แบบ Pointer	พิกัดที่อยู่ (Address)	ค่าข้อมูล (Value)
x[0]	*(x + 0)	72	10
x	*(x + 1)		
x[2]	*(x + 2)		
x[3]	*(x + 3)		
x[4]	*(x + 4)		

คำถามท้ายโจทย์: นิพจน์ $*(x + 3)$ จะได้ผลลัพธ์ข้อมูลตรงกับตัวแปรอาร์ดชันที่เท่าใด และมีค่าข้อมูลคืออะไร?

ตอบ:

แนวทางการนำไปใช้งานในชั้นเรียน (PBL Step)

1. **ขั้นเสนอปัญหา (10 นาที):** คุณครูเปิดโค้ดภาษาซีบนจอ แล้วรันโปรแกรมโชว์บั๊ก Segmentation Fault หรือแสดงผลลัพธ์ที่นักเรียนคาดไม่ถึง
2. **ขั้นคิดวิเคราะห์และใช้สื่อ (40 นาที):** แจกใบงานชุดนี้ให้นักเรียน ปวช.2 ทั้ง 19 คนนั่งทำ โดยให้จับคู่กัน คิดและเขียนกระบวนการแชร์หน่วยความจำลงในกระดาษ
3. **ขั้นปฏิบัติการจริง (40 นาที):** ให้นักเรียนพิมพ์โค้ดเหล่านี้ลงในคอมพิวเตอร์ สั้รันโปรแกรมเพื่อดูค่าที่คอมพิวเตอร์พิมพ์ออกมาเปรียบเทียบกับสิ่งที่ตนเองกรอกในตารางใบงาน

แบบทดสอบวัดผลสัมฤทธิ์ทางการเรียน เรื่อง ตัวชี้ (Pointers) ภาษาซี

รายวิชา: พื้นฐานการเขียนโปรแกรมคอมพิวเตอร์ (21909-1005)

คำชี้แจง: แบบทดสอบแบ่งออกเป็น 2 ตอน รวมคะแนนเต็ม 20 คะแนน ให้อ่านเวลาทำ 45 นาที

ตอนที่ 1: ข้อสอบปรนัยเลือกตอบ 4 ตัวเลือก (15 ข้อ - 15 คะแนน)

1. ตัวแปรประเภท "ตัวชี้" (Pointer Variable) ในภาษาซี ทำหน้าที่จัดเก็บสิ่งใดในหน่วยความจำ?

- ก. ค่าของตัวเลขทศนิยมความละเอียดสูง (Double value)
- ข. พิกัดตำแหน่งที่อยู่ของหน่วยความจำ (Memory Address)
- ค. ตัวอักษรหรือข้อความ (String/Character)
- ง. ค่าความจริงทางตรรกศาสตร์ (Boolean)

2. ตัวดำเนินการใดต่อไปนี้เรียกว่า "Address Operator" (ตัวดำเนินการตำแหน่งที่อยู่) ในภาษาซี?

- ก. *
- ข. ->
- ค. &
- ง. %

3. หากกำหนดให้ `int v = 3;` นิพจน์ใดหมายถึงการดึง "หมายเลขพิกัดหน่วยความจำ" ของตัวแปร v?

- ก. *v
- ข. &v
- ค. v*
- ง. v&

4. ตัวดำเนินการทางอ้อม (Indirection Operator หรือ Dereference Operator) คือเครื่องหมายใด?

- ก. แอมเพอร์แซนด์ (&)
- ข. ดอกจัน (*)
- ค. อัศเจรีย์ (!)
- ง. เครื่องหมายคำถาม (?)

5. รูปแบบการประกาศตัวแปรตัวชี้เพื่อชี้ไปยังข้อมูลประเภทเลขจำนวนเต็ม (Integer) ข้อใดถูกต้อง?

- ก. `int pv*;`
- ข. `*int pv;`
- ค. `int *pv;`
- ง. `pointer int pv;`

6. พิจารณาโค้ดต่อไปนี้ ผลลัพธ์สุดท้ายค่าของตัวแปร v จะเป็นเท่าใด?

```
int v = 3;  
int *pv = &v;  
*pv = 0;
```

- ก. 3
- ข. 0
- ค. หมายเลขพิกัดหน่วยความจำ
- ง. เกิดข้อผิดพลาดโปรแกรมรันไม่ได้

7. ข้อใดคือความหมายของคำว่า "NULL Pointer" ในการเขียนโปรแกรมภาษาซี?

- ก. พอยน์เตอร์ที่ชี้ไปยังหน่วยความจำที่เสียหาย
- ข. พอยน์เตอร์ที่มีค่าตำแหน่งที่อยู่เริ่มต้นเป็น 0 (ไม่ชี้ไปที่ใดเลย)
- ค. พอยน์เตอร์ที่เก็บค่าว่างเปล่าที่เป็นช่องว่าง (Space)
- ง. พอยน์เตอร์ที่ถูกทำลายทิ้งไปแล้ว

8. เมื่อส่งผ่านค่าตัวแปรไปยังฟังก์ชันย่อยแบบ "Pass by Reference" สิ่งใดจะถูกส่งไปยังฟังก์ชัน?

- ก. ค่าสำเนาข้อมูลของตัวแปร
- ข. ชื่อของตัวแปร
- ค. ตำแหน่งที่อยู่จริง (Address) ในหน่วยความจำ
- ง. ตัวแปรทุกตัวที่อยู่ในโปรแกรมหลัก

9. หากต้องการส่งพิกัดของตัวแปร `int u;` เข้าไปยังฟังก์ชัน `funct2(int *pu, int *pv)` ต้องเขียนโค้ดเรียกใช้งานอย่างไร?

- ก. `funct2(u);`
- ข. `funct2(*u);`

ค. `funct2(&u);`

ง. `funct2(int u);`

10. เหตุใดการสลับค่าตัวแปรในฟังก์ชันย่อยทั่วไป (Pass by Value) จึงไม่มีผลเปลี่ยนค่าในฟังก์ชันหลัก (Main)?

ก. ฟังก์ชันหลักลอกค่าเอาไว้มันให้แก้ไข

ข. ฟังก์ชันย่อยแก้ไขข้อมูลบน "ค่าสำเนา" ไม่ใช่พิกัดจริงของตัวแปรต้นทาง

ค. แรมของคอมพิวเตอร์ประมวลผลไม่ทัน

ง. ไวยากรณ์ภาษาซีไม่อนุญาตให้แก้ไขข้อมูลใด ๆ ทั้งสิ้น

11. สมมติให้ `static int x[5] = {10, 11, 12, 13, 14};` นิพจน์พอยน์เตอร์ในข้อใดมีค่าเท่ากับค่าข้อมูล `x[0]` ?

ก. `&x`

ข. `*x`

ค. `x + 0`

ง. `*(x + 1)`

12. จากข้อ 11 นิพจน์ใดมีค่าเทียบเท่ากับคำว่า `x[3]` ?

ก. `*(x + 3)`

ข. `*x + 3`

ค. `&(x + 3)`

ง. `x[3]*`

13. หากกำหนดให้ `int *px;` จัดเก็บอยู่ที่พิกัดหน่วยความจำ `117E` และข้อมูลระบบนี้มีขนาดเลขจำนวนเต็มเท่ากับ 2 ไบต์ นิพจน์ `px + 1` จะเลื่อนพิกัดไปยังตำแหน่งใด?

ก. `117F`

ข. `1180`

ค. `117F` และ `1180`

ง. `1181`

14. ความล้มเหลวของการเข้าถึงหน่วยความจำที่เรียกว่า "Segmentation Fault" มักเกิดจากสาเหตุใดในเรื่องตัวชี้?

- ก. ตัวแปรเก็บข้อมูลมากเกินไปจนล้น (Overflow)
- ข. พอยน์เตอร์ชี้ไปทำงานยังพิกัดตำแหน่งหน่วยความจำที่ไม่มีอยู่จริงหรือไม่ได้รับอนุญาต
- ค. คอมพิวเตอร์ไม่ได้ติดตั้งการดหน่วยความจำแรม
- ง. เขียนชื่อตัวแปรยาวเกินกว่าหลักเกณฑ์

15. การคำนวณทางคณิตศาสตร์ข้อใดต่อไปนี้ที่ "ไม่ได้รับอนุญาต" ให้กระทำกับตัวแปรพอยน์เตอร์ในภาษาซี?

- ก. การบวกค่าคงที่เข้าไปในพอยน์เตอร์ (`px + 3`)
- ข. การเพิ่มค่าขึ้นทีละหนึ่งพิกัด (`++px`)
- ค. การนำพอยน์เตอร์สองตัวมาคูณหรือบวกกันเอง (`px * py` หรือ `px + py`)
- ง. การนำพอยน์เตอร์สองตัวในอาเรย์เดียวกันมาลบกันเพื่อหาความห่าง (`py - px`)

ตอนที่ 2: ข้อสอบอัตนัยวิเคราะห์และเขียนโค้ด (5 ข้อ - 5 คะแนน)

คำชี้แจง: จงเติมคำสั่งภาษาซีลงในช่องว่าง หรือแก้ไขโค้ดให้ทำงานได้สมบูรณ์ถูกต้อง

16. จงเติมโค้ดในช่องว่างเพื่อทำการกำหนดให้ตัวแปรพอยน์เตอร์ `pv` ชี้ไปยังตำแหน่งของตัวแปร `v` ให้ถูกต้อง

```
int v = 5;
int *pv;
pv = _____;
```

17. จงเติมเครื่องหมายในช่องว่างเพื่อให้โปรแกรมทำการแก้ไขค่าข้อมูลในหน่วยความจำของพอยน์เตอร์ `pu` ให้กลายเป็น 10 แบบทางอ้อม

```
int u = 2;
int *pu = &u;
_____pu = 10; // คำ u ใน main จะกลายเป็น 10 ทันที
```

18. จากโครงสร้างฟังก์ชัน `swap` ด้านล่าง จงเติมส่วนหัวของฟังก์ชัน (Function Header) ให้รองรับการรับค่าแบบพอยน์เตอร์อ้างอิงตำแหน่ง

```
void swap( _____ )
{
```

```
int temp = *px;

*px = *py;

*py = temp;

}
```

19. จากโค้ดอาเรย์ จงเปลี่ยนรูปแบบคำสั่ง `printf` บรรทัดล่างสุดให้อยู่ในรูปของนิพจน์แบบพอยน์เตอร์ (Pointer Notation)

```
int a[5] = {10, 20, 30, 40, 50};

// โดนบังคับให้เปลี่ยนจาก: printf("%d", a[2]);

printf("%d", _____);
```

20. โดนบักบ่อย! จงเขียนคำสั่งเปิดใช้งานฟังก์ชันจองหน่วยความจำแบบไดนามิก (Dynamic Memory Allocation) ของตัวแปรตัวชี้ `x` เพื่อจองพื้นที่เลขจำนวนเต็มจำนวน 10 ตัว

```
#include <stdlib.h>

int *x;

x = (int*) _____;
```

เฉลยแบบทดสอบ (Answer Key)

ตอนที่ 1 (ปรนัย)

1. ข (ตัวชี้ใช้สำหรับเก็บตำแหน่งพิกัด Address เท่านั้น)
2. ค (เครื่องหมาย `&` ใช้ดึงค่า Address ออกมา)
3. ข (`&v` เป็นการดึง Address ของ `v`)
4. ข (เครื่องหมาย `*` คือตัวดำเนินการเข้าถึงค่าทางอ้อม)
5. ค (ไวยากรณ์คือ `data-type *name;`)
6. ข (คำสั่ง `*pv = 0` ส่งผลให้เปลี่ยนค่าของ `v` ตั้งเดิมทางอ้อมเป็น 0)
7. ข (NULL ในพอยน์เตอร์มีค่าแทนเลข 0 เพื่อบ่งบอกว่ายังไม่ได้ตั้งชี้ไปที่พิกัดใด)
8. ค (Pass by Reference คือการส่งตำแหน่งที่อยู่จริงไปให้ฟังก์ชันทำงาน)

9. ค (ต้องเติมเครื่องหมาย & นำหน้าเพื่อตั้งค่า Address ส่งเข้าไป)
10. ข (Pass by Value มีคุณลักษณะเป็นการก๊อปปี้ข้อมูลไปเปิดตัวแปรใหม่เฉพาะกิจในฟังก์ชัน)
11. ข (ชื่ออาร์เรย์ใด ๆ มีพฤติกรรมเหมือนพอยน์เตอร์ชี้ไปที่ดัชนีตัวแรก *x จึงเท่ากับ x[0])
12. ก ($*(x + 3)$ คือการคำนวณแบบออฟเซตเพื่อขยับพิกัดไป 3 ล็อกแล้วเปิดดึงค่าข้างในออก)
13. ข (พอยน์เตอร์คำนวณตามขนาดชนิดข้อมูล 117E ตัวถัดไปบวก 2 ไบต์ในฐานะสิบหกจะเท่ากับ 1180)
14. ข (เกิดจากการพยายามเข้าถึงพิกัดแรมที่เลื่อน ไม่ได้รับสิทธิ์ในการจองพื้นที่)
15. ค (ในทางสถาปัตยกรรมคอมพิวเตอร์ การนำพิกัด RAM สองฝั่งมาคูณหรือบวกกันไม่มีความหมายเชิงโลจิก)

ตอนที่ 2 (อัตนัย)

1. &v (ส่งพิกัดที่อยู่ของ v ไปเก็บในตัวแปรตัวชี้)
2. * (ใช้เครื่องหมายดัดแปลงทางอ้อมหน้าชื่อพอยน์เตอร์)
3. int *px, int *py (การตั้งประเภทอาร์กิวเมนต์แบบพอยน์เตอร์เพื่อรับค่า Pass by Reference)
4. *(a + 2) (มีค่าเทียบเท่าพิกัดอาร์เรย์ดัชนีตัวที่ 2)
5. malloc(10 * sizeof(int)) (ฟังก์ชันจัดสรรพื้นที่หน่วยความจำในระบบแบบพลวัต)

1. ตารางลงคะแนนและสถิติสำเร็จรูปสำหรับบรรจุใน บทที่ 4

ตารางที่ 4.1 คะแนนระหว่างเรียนจากการทำชุดแบบฝึกทักษะ (E_1) คะแนนทดสอบก่อนเรียน และหลังเรียน (E_2) ของนักเรียนรายบุคคล

นักเรียนคนที่	คะแนนแบบฝึกทักษะรวม (เต็ม 30 คะแนน)	คะแนนก่อนเรียน (เต็ม 20 คะแนน)	คะแนนหลังเรียน (เต็ม 20 คะแนน)	คะแนนความก้าวหน้า (Post - Pre)
1	26	7	17	10
2	24	6	16	10
3	25	8	17	9
4	28	9	19	10
5	23	5	16	11
6	25	7	16	9
7	24	8	15	7
8	26	9	17	8
9	22	6	15	9
10	27	10	18	8
11	25	7	16	9
12	24	8	17	9
13	26	7	16	9
14	27	9	18	9
15	23	6	15	9
16	25	8	16	8
17	26	7	17	10
18	24	6	16	10
19	24	8	14	6
รวม (Sum)	474	141	311	170
ค่าเฉลี่ย ($\bar{x}$)	24.95	7.42	16.37	8.95
ร้อยละ (%)	83.16% (E_1)	37.11%	81.84% (E_2)	44.74%

ตารางที่ 4.2 ผลการเปรียบเทียบผลสัมฤทธิ์ทางการเรียน เรื่อง Pointers ของนักเรียน ปวช.2 (ทค.2/1) (N = 19)

การทดสอบ	คะแนนเต็ม	ค่าเฉลี่ย ($\bar{X}$)	ส่วนเบี่ยงเบนมาตรฐาน (S.D.)	ค่า t-test	ระดับนัยสำคัญ
ก่อนเรียน	20	7.42	1.17	25.14*	.000
หลังเรียน	20	16.37	1.16		

*มีนัยสำคัญทางสถิติที่ระดับ .05 (จากตารางสถิติกำหนดค่าวิกฤต $t = 1.734$ ที่ $(df = 18)$)

2. แบบสอบถามความพึงพอใจ 5 ระดับ (Likert Scale) เพื่อใช้เก็บข้อมูลจริง

คำชี้แจง: โปรดทำเครื่องหมาย ✓ ลงในช่องที่ตรงกับระดับความพึงพอใจของนักเรียนมากที่สุด

(5 = มากที่สุด, 4 = มาก, 3 = ปานกลาง, 2 = น้อย, 1 = น้อยที่สุด)

รายการประเมินความพึงพอใจต่อชุดแบบฝึกทักษะ เรื่อง Pointers	5	4	3	2	1
ด้านเนื้อหาและการจัดกิจกรรมเรียนรู้ (PBL)					
1. โจทย์และสถานการณ์ปัญหามีความน่าสนใจและท้าทายกระบวนการคิด					
2. การเรียนรู้แบบแก้ไขปัญหาลงมือช่วยให้เข้าใจโลกภาษาซีได้ดีขึ้น					
3. เนื้อหาในแบบฝึกทักษะมีการเรียงลำดับจากง่ายไปยากอย่างเป็นระบบ					
ด้านนวัตกรรมและการจำลองหน่วยความจำ (Memory Layout)					
4. ตารางและแผนภาพจำลองหน่วยความจำ (RAM) ช่วยเปลี่ยนนามธรรมให้เห็นภาพชัดเจน					
5. เครื่องหมาย & (Address) และ * (Dereference) เข้าใจง่ายขึ้นเมื่อใช้ภาพจำลอง					
6. รูปเล่มและรูปแบบของแบบฝึกทักษะมีความชัดเจน เข้าใจง่าย ไม่ซับซ้อน					
ด้านประโยชน์และความคงทนในการเรียนรู้					
7. นักเรียนสามารถวิเคราะห์และตรวจหาข้อผิดพลาด (Debug) บั๊กในโปรแกรมได้แม่นยำขึ้น					
8. แบบฝึกทักษะนี้ช่วยลดการเกิดข้อผิดพลาดประเภท Segmentation Fault ในเวลาปฏิบัติงาน					
9. นักเรียนมีความมั่นใจในการเขียนโปรแกรมภาษาคอมพิวเตอร์ในหัวข้อขั้นสูงต่อไป					
10. บรรยากาศในการเรียนรู้กระตุ้นให้เกิดการแลกเปลี่ยนความคิดเห็นกับเพื่อนร่วมชั้น					

3. Interactive Quiz เรื่อง Pointers ภาษาซี (สำหรับให้นักเรียนลองฝึกฝน)

[โจทย์ข้อที่ 1] พื้นฐานตัวชี้อ้างอิง

```
#include <stdio.h>
```

```
int main() {
```

```
    int score = 85;
```

```
    int *p_score = &score;
```

```
    *p_score = 99;
```

```
    printf("%d", score);
```

```
    return 0;
```

```
}
```

คำถาม: โปรแกรมนี้จะแสดงผลลัพธ์ (Output) ตัวเลขใดออกมาบนหน้าจอ?

- ก. 85
- ข. 99
- ค. หมายเลข Address (เลขฐานสิบหก)
- ง. เกิดข้อผิดพลาดโปรแกรมคอมไพล์ไม่ผ่าน
- เฉลย: ข. 99 (เพราะระบุตัวดำเนินการทางอ้อมเปลี่ยนค่าต้นทาง)

[โจทย์ข้อที่ 2] พอยน์เตอร์และฟังก์ชัน (Pass by Reference)

```
#include <stdio.h>
```

```
void updateValue(int *p) {
```

```
    *p = *p + 10;
```

```
}
```

```
int main() {
```

```
    int num = 15;
```

```
    updateValue(&num);
```

```
    printf("%d", num);
```

```
    return 0;
```

```
}
```

คำถาม: ผลลัพธ์สุดท้ายที่ฟังก์ชันหลักพิมพ์ออกมาคืออะไร?

- ก. 15
- ข. 10
- ค. 25
- ง. 0
- เฉลย: ค. 25 (พิกัดหน่วยความจำโดนบวกเพิ่มเข้าไป 10)

[โจทย์ข้อที่ 3] พอยน์เตอร์คำนวณตำแหน่งกับอาเรย์

```
#include <stdio.h>

int main() {
    int arr[4] = {10, 20, 30, 40};
    int *ptr = arr; // ptr ชี้ที่ตำแหน่งเริ่มต้นของอาเรย์
    printf("%d", *(ptr + 2));
    return 0;
}
```

คำถาม: ตัวเลขใดจะปรากฏบนคอนโซลหลังจากสักรันโปรแกรมนี้?

- ก. 10
- ข. 20
- ค. 30
- ง. 40
- เฉลย: ค. 30 (Offset + 2 หมายถึงขยับไปดัชนีตำแหน่งที่ 2 ซึ่งก็คือเลข 30)

[โจทย์ข้อที่ 4] บั๊กตัวร้ายประจำวิชาเขียนโปรแกรม

```
#include <stdio.h>

int main() {
    int *ptr = NULL;
    *ptr = 100; // บั๊กที่อันตราย
    printf("%d", *ptr);
    return 0;
}
```

คำถาม: เมื่อนักเรียนสั่งรันโค้ดภาษาซีข้างต้นนี้ในห้องปฏิบัติการ จะส่งผลลัพธ์อย่างไรกับตัวระบบ?

- ก. พิมพ์เลข 100 ออกมาปกติ
- ข. พิมพ์เลข 0 ออกมา
- ค. เกิดข้อผิดพลาดของระบบปฏิบัติการคือ "Segmentation Fault" (โปรแกรมหยุดทำงานกะทันหัน)
- ง. โปรแกรมจะแจ้งเตือนว่าหน่วยความจำแรมเต็ม
- **เฉลย: ค. เกิดข้อผิดพลาดเนื่องจากพยายามเขียนค่าข้อมูลลงพอยน์เตอร์ที่เป็นสายว่าง (NULL)**

ตารางเกณฑ์ Rubric การให้คะแนนแบบฝึกทักษะ (ภาคเน้นปฏิบัติงาน)

เกณฑ์การประเมินทักษะการปฏิบัติงานเขียนและวิเคราะห์พอยน์เตอร์ (Scoring Rubrics)

ประเด็นการประเมิน	ระดับคุณภาพ 3 (ดี)	ระดับคุณภาพ 2 (พอใช้)	ระดับคุณภาพ 1 (ปรับปรุง)
1. ทักษะการระบุพิกัดที่อยู่ (Address Allocation) <i>(ค่าน้ำหนักคะแนน: 30%)</i>	ระบุความแตกต่างระหว่าง Value และ Memory Address ได้อย่างสมบูรณ์แบบวาดพิกัดเลขฐานสิบหกตรงตามสถาปัตยกรรมของตัวแปรในโจทย์ไม่มีผิดพลาด	ระบุความแตกต่างได้ แต่กรอกรหัสเลขพิกัดสลับช่องตัวแปรเล็กน้อย หรือเขียนรูปแบบตัวเลขฐานสิบหกตกหล่นบางส่วน	ไม่สามารถแยกแยะความแตกต่างระหว่าง Value และ Address ได้ กรอกรหัสผิดพลาดตำแหน่งทั้งหมด
2. ทักษะการชี้ทิศทางและการดึงค่า (Dereferencing & Operators) <i>(ค่าน้ำหนักคะแนน: 40%)</i>	วาดเส้นลูกศรเชื่อมโยงจากพอยน์เตอร์ตัวแปรไปยังที่อยู่เป้าหมายได้ถูกต้อง แปลงกลไกเครื่องหมาย & และ * เพื่อเปลี่ยนค่าในเซลล์ RAM แบบทางอ้อมได้ถูกต้องแม่นยำ	วาดเส้นลูกศรชี้พิกัดได้ถูกต้อง แต่ทำตามคำสั่งการแปลงค่าทางอ้อม (Dereference) ผิดพลาดส่งผลให้ค่าสุดท้ายในเซลล์ RAM คลาดเคลื่อน 1-2 จุด	ไม่สามารถลากเส้นโยงความสัมพันธ์ของพอยน์เตอร์ได้ และใช้ตัวดำเนินการ & หรือ * ผิดสลับกันส่งผลให้ระบบพัง
3. ผลลัพธ์โค้ดและการแก้ไข (Code Output & Debugging) <i>(ค่าน้ำหนักคะแนน: 30%)</i>	คาดการณ์ผลลัพธ์การรันโปรแกรมภาษาซี (Output) ได้ถูกต้องสมบูรณ์ และเขียนโค้ดแก้ปัญหา Segmentation Fault ให้ระบบคอมไพล์ผ่านได้ด้วยตนเอง	คาดการณ์ผลลัพธ์ของโปรแกรมได้ถูกต้อง แต่โค้ดที่เขียนแก้ไขบั๊กยังมีข้อผิดพลาดด้านไวยากรณ์เล็กน้อย (Syntax Error) ต้องได้รับคำแนะนำจากครู	ไม่สามารถบอกผลลัพธ์การทำงานของโค้ดได้ และไม่สามารถเขียนโค้ดแก้ไขข้อผิดพลาดใด ๆ ได้เลย

เกณฑ์การสรุปคะแนนแต่ละใบงาน (คิดเป็นร้อยละ):

- คะแนนรวมได้ร้อยละ 80 - 100 = ระดับ ดีมาก (ผ่านเกณฑ์ประสิทธิภาพขั้นสูง)
- คะแนนรวมได้ร้อยละ 70 - 79 = ระดับ ดี (ผ่านเกณฑ์)
- คะแนนรวมต่ำกว่าร้อยละ 50 = ระดับ ปรับปรุง (ไม่ผ่านเกณฑ์ ต้องเรียนซ่อมเสริม)