



Term Project

CPE 112: Programming with Data Structures

CashierX By ShopWiz

จัดทำโดย

กัญญญาพัชร	ชายทวีป	66070501003
ณัฐวุฒิ	ซิมเจริญ	66070501019
พรประทาน	เกื้อรุ่ง	66070501036
ศรีศุภดิษฐ์	รัตนประเสริฐ	66070501052
อัคนิรุทธิ์	ปานเดช	66070501062
คุณภัทร	อุบลธรา	66070501068
ญาณกร	ต่างประโคน	66070501071

ภาคเรียนที่ 2 ปีการศึกษา พ.ศ. 2566

คณะวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

สารบัญ

	หน้า
สารบัญ	2
บทที่ 1 บทนำ	3
• ที่มาและความสำคัญ	3
• วัตถุประสงค์ของการโครงการ	3
• ขอบเขตการโครงการ	4
• ประโยชน์ที่คาดว่าจะได้รับ	4
• หน้าที่ที่รับผิดชอบ	4
บทที่ 3 วิธีการดำเนินการ	5
• โครงสร้างข้อมูลที่ใช้ประกอบด้วยอะไรบ้างและเหตุผลในการใช้	5
• การทำงานแต่ละฟังก์ชันและการใช้อัลกอริทึมจัดการโครงสร้างข้อมูล	6
• วิธีการใช้งานและการทดสอบการทำงานของโปรแกรม	7
• อุปกรณ์ เครื่องมือ หรือโปรแกรมที่ใช้ในการพัฒนา	7
บทที่ 5 สรุปผลโครงการ อภิปรายผล และข้อเสนอแนะ	8
• อภิปรายผล	9
• ปัญหาและอุปสรรคในการดำเนินงาน	9
• แนวทางการแก้ไข	10

บทที่ 1

บทนำ

ที่มาและความสำคัญ

การพัฒนาโปรแกรมในภาษา C เป็นหนึ่งในวิธีการที่มีประสิทธิภาพสูงในการสร้างซอฟต์แวร์ที่ทำงานได้รวดเร็วและมีการใช้ทรัพยากรระบบอย่างมีประสิทธิภาพ อย่างไรก็ตาม การเขียนโปรแกรมในภาษา C อาจมีความซับซ้อนและต้องการความระมัดระวังในการจัดการหน่วยความจำและการเข้าถึงข้อมูล หากไม่มีการใช้โครงสร้างข้อมูลที่เหมาะสม โปรแกรมอาจมีปัญหาระหว่างการใช้ทรัพยากรที่ไม่เหมาะสมและประสิทธิภาพที่ต่ำ

ดังนั้น การใช้โครงสร้างข้อมูล (Data Structures) ที่มีประสิทธิภาพในการพัฒนาโปรแกรมในภาษา C จึงเป็นสิ่งสำคัญ เพื่อให้โปรแกรมสามารถทำงานได้อย่างมีประสิทธิภาพและจัดการข้อมูลได้อย่างเหมาะสม ตัวอย่างเช่น การใช้ Linked List, AVL Tree, และ Heap เพื่อจัดเก็บและจัดการข้อมูลสินค้าภายในร้านค้า ช่วยให้โปรแกรมสามารถทำงานได้รวดเร็วขึ้นและลดปัญหาการใช้หน่วยความจำที่ไม่จำเป็น

โปรแกรมบริหารจัดการสินค้าภายในร้านค้านี้ได้ถูกพัฒนาโดยใช้โครงสร้างข้อมูลที่หลากหลาย เช่น Linked List ในการจัดการข้อมูลพลวัต (Dynamic Data) และ AVL Tree สำหรับการจัดเก็บข้อมูลที่ต้องการการค้นหาที่รวดเร็วและมีประสิทธิภาพ การเลือกใช้โครงสร้างข้อมูลเหล่านี้ช่วยให้การจัดการสินค้าภายในร้านค้าเป็นไปอย่างราบรื่นและมีประสิทธิภาพสูงสุด

นอกจากนี้ การพัฒนาโปรแกรมในภาษา C ยังมีข้อดีในเรื่องของการควบคุมการทำงานของโปรแกรมอย่างละเอียด ผู้พัฒนาสามารถจัดการหน่วยความจำและทรัพยากรอื่นๆ ของระบบได้อย่างตรงจุด ทำให้โปรแกรมมีความเร็วและประสิทธิภาพที่เหนือกว่าโปรแกรมที่พัฒนาในภาษาระดับสูงอื่นๆ

การเลือกใช้โครงสร้างข้อมูลที่เหมาะสมในภาษา C จึงเป็นการสร้างพื้นฐานที่ดีในการพัฒนาโปรแกรมที่มีประสิทธิภาพและตอบสนองความต้องการของผู้ใช้ได้อย่างมีประสิทธิภาพ ซึ่งเป็นที่มาของการพัฒนาโปรแกรมบริหารจัดการสินค้าภายในร้านค้านี้ โดยมุ่งเน้นไปที่การใช้โครงสร้างข้อมูลเพื่อให้เกิดการทำงานที่มีประสิทธิภาพและสามารถจัดการข้อมูลได้อย่างถูกต้องและรวดเร็ว

วัตถุประสงค์ของการโครงงาน

1. เพื่อเพิ่มประสิทธิภาพในการจัดการสินค้า
2. เพื่อปรับปรุงการติดตามและวิเคราะห์ประวัติการซื้อขาย
3. เพื่อเสริมสร้างการจัดการผู้ใช้ภายในร้านค้า
4. เพื่ออำนวยความสะดวกในการซื้อสินค้าและการคิดเงิน
5. เพื่อส่งเสริมการใช้เทคโนโลยีในการบริหารจัดการธุรกิจ
6. เพื่อนำความรู้ที่ได้จากในห้องเรียนมาประยุกต์ใช้พัฒนากิจกรรมและความสามารถ

7. เพื่อศึกษาความรู้เพิ่มเติมที่นอกเหนือจากชั้นเรียน

ขอบเขตการโครงการงาน

ขอบเขตการศึกษา เรื่องการพัฒนาโปรแกรมด้วยภาษา C ประกอบด้วย ขอบเขตเนื้อหาที่ใช้ และด้านเวลา ดังนี้

1. ขอบเขตเกี่ยวกับเนื้อหาที่ใช้ เนื่องด้วยการศึกษาในครั้งนี้ อยู่ภายใต้หัวข้อการพัฒนาโดยใช้ ภาษา C ทำให้โครงการในครั้งนี้ถูกจัดทำขึ้นด้วย ภาษา C เท่านั้น
2. ขอบเขตด้านเวลา ระยะเวลาในการดำเนินการครั้งนี้มีเวลาจำกัด โดยเริ่มต้นโครงการตั้งแต่วันที่ 19.04.2567 -29.05.2567

ประโยชน์ของโปรแกรมบริหารจัดการสินค้าภายในร้านค้า

การจัดการสินค้าที่มีประสิทธิภาพ : โปรแกรมช่วยให้ผู้ดูแลสามารถเพิ่ม ลบ และแก้ไขข้อมูลสินค้าภายในร้านได้อย่างง่ายดาย และเป็นระบบ ทำให้สามารถตรวจสอบสถานะสินค้าคงคลังได้อย่างรวดเร็วและแม่นยำ

การติดตามประวัติการซื้อขาย : ระบบประวัติการซื้อขายช่วยให้ผู้ดูแลสามารถตรวจสอบและติดตามการซื้อขายของลูกค้าได้อย่างละเอียด ทำให้สามารถวิเคราะห์และปรับปรุงการบริการได้ตามความต้องการของลูกค้า

การจัดการผู้ใช้ : โปรแกรมมีระบบการจัดการผู้ใช้ที่ช่วยให้สามารถเพิ่ม ลบ และแก้ไขข้อมูลผู้ใช้ได้ รวมถึงการตรวจสอบและติดตามกิจกรรมของผู้ใช้แต่ละราย

การปรับปรุงประสิทธิภาพการทำงาน : ด้วยระบบที่ครอบคลุมและเป็นระเบียบ โปรแกรมช่วยลดความซับซ้อนในการจัดการข้อมูลและเพิ่มประสิทธิภาพในการทำงานของผู้ดูแลร้านค้า

หน้าที่รับผิดชอบ

ชื่อ	นามสกุล	เลขประจำตัวนักศึกษา	อีเมล	หน้าที่รับผิดชอบ
กัญญญาพัชร	ชายทวีป	66070501003	Kanyaphat.chai@mail.kmutt.ac.th	Senior Programmer
ณัฐวุฒิ	ซิมเจริญ	66070501019	Nuttawut.simc@kmutt.ac.th	Senior Programmer
พรประทาน	เกื้อรุ่ง	66070501036	Ponprathan.kuea@kmutt.ac.th	Senior Programmer
ศรีศุภดิษฐ์	รัตนประเสริฐ	66070501052	Srisuphadith.ratt@kmutt.ac.th	Senior Programmer
อัคนิรุทธิ์	ปานเดช	66070501062	Arkhanirut.pand@kmutt.ac.th	Project Manager
คุณภัทร	อุบลธรา	66070501068	Khunnapat.aubo@kmutt.ac.th	Adviser PM.
ญาณกร	ต่างประโคน	66070501071	Yanakorn.tang@mail.kmutt.ac.th	Senior Programmer

บทที่ 3

วิธีการดำเนินงาน

การพัฒนาโปรแกรมบริหารจัดการสินค้าภายในร้านค้านี้มีความสามารถหลากหลายที่ช่วยให้ผู้ใช้สามารถจัดการข้อมูลสินค้า ผู้ใช้ และประวัติการซื้อขายได้อย่างมีประสิทธิภาพ ภาพรวมของความสามารถของโปรแกรม

1. การเพิ่ม ลบ และแก้ไขข้อมูลสินค้า
2. การจัดการข้อมูลผู้ใช้
3. การติดตามประวัติการซื้อขายของลูกค้า
4. ระบบตะกร้าสินค้าและการคิดเงิน
5. การแสดงสินค้าทั้งหมด
6. การแสดงสินค้าตามประเภท
7. ระบบการแนะนำสินค้า
8. การแนะนำประเภทสินค้าใกล้เคียง

โปรแกรมนี้ถูกพัฒนาด้วยภาษา C โดยใช้โครงสร้างข้อมูลต่าง ๆ และอัลกอริทึมในการจัดการข้อมูลเพื่อให้การทำงานมีประสิทธิภาพและแม่นยำ

1. โครงสร้างข้อมูล

- 1.1. โครงสร้างข้อมูลสินค้า : ใช้ `'struct'` เพื่อเก็บข้อมูลสินค้า เช่น รหัสสินค้า ชื่อสินค้า จำนวนคงเหลือ และราคาสินค้า เนื่องจาก `'struct'` ช่วยให้การจัดเก็บข้อมูลหลายชนิดที่เกี่ยวข้องกันเป็นเรื่องง่ายและเป็นระบบ
- 1.2. โครงสร้างข้อมูลผู้ใช้ : ใช้ `'struct'` เพื่อเก็บข้อมูลผู้ใช้ เช่น รหัสผู้ใช้ ชื่อผู้ใช้ และสถานะผู้ใช้ โครงสร้างนี้ช่วยให้สามารถจัดการข้อมูลผู้ใช้ได้อย่างเป็นระบบ
- 1.3. โครงสร้างข้อมูลประวัติการซื้อขาย : ใช้ `'struct'` เพื่อเก็บข้อมูลการซื้อขาย เช่น รหัสการขาย รายการสินค้าที่ซื้อ จำนวนสินค้า และราคาทั้งหมด ซึ่งช่วยในการติดตามและวิเคราะห์ประวัติการซื้อขาย
- 1.4. โครงสร้างข้อมูลความสัมพันธ์ประเภทสินค้า: ใช้ `'struct'` เพื่อเก็บข้อมูลความสัมพันธ์ของสินค้าแต่ละประเภท เพื่อให้สามารถขยายขนาดข้อมูลประเภทสินค้าและนำไปใช้ต่อไปในอัลกอริทึมอื่น ๆ ในอนาคต

2. ฟังก์ชันการทำงาน

2.1. การเพิ่มสินค้า

2.1.1. ใช้ฟังก์ชันในการรับข้อมูลสินค้าใหม่จากผู้ใช้แล้วจัดเก็บลงในลิงค์ลิสต์ (Linked list)

2.1.2. อัลกอริทึม : การแทรกข้อมูลในลิงค์ลิสต์ (Linked list)

2.2. การลบสินค้า

2.2.1. ใช้ฟังก์ชันในการค้นหาสินค้าตามรหัสสินค้าและลบในลิงค์ลิสต์ (Linked list)

2.2.2. อัลกอริทึม : การค้นหาเชิงเส้น (linear search) และการลบในลิงค์ลิสต์ (Linked list)

2.3. การแก้ไขข้อมูลสินค้า

2.3.1. ใช้ฟังก์ชันในการค้นหาสินค้าแล้วแก้ไขข้อมูลตามที่ผู้ใช้ต้องการ

2.3.2. อัลกอริทึม : การค้นหาเชิงเส้น (linear search) และการปรับปรุงข้อมูลในลิงค์ลิสต์ (Linked list)

2.4. การแสดงสินค้าทั้งหมด

2.4.1. การใช้ฟังก์ชันในการแสดงสินค้าในแต่ละโหนดจนกระทั่งครบทุกโหนดของลิงค์ลิสต์ (Linked list)

2.4.2. อัลกอริทึม : การค้นหาเชิงเส้น (linear search) และการแสดงข้อมูล

2.5. การแสดงสินค้าแต่ละประเภท

2.5.1.

การใช้ฟังก์ชันในการแสดงสินค้าในโหนดที่เป็นประเภทที่เลือกจนกระทั่งครบทุกโหนด

2.5.2. อัลกอริทึม : การค้นหาเชิงเส้น (linear search) และการปรับปรุงข้อมูลในลิงค์ลิสต์ (Linked list)

2.6. การแนะนำสินค้า

2.6.1. ใช้การคำนวณคะแนนสินค้าในการแนะนำสินค้า

2.6.2. อัลกอริทึม: การแสดงข้อมูลโดยเรียงตามค่าที่ใช้ในการแนะนำ (suggestion key) ที่มาจากการคำนวณภายในโปรแกรมผ่านค่าที่เก็บข้อมูลจากการตอบสนองกับผู้ใช้ (user) ซึ่งเป็นแบบพลวัต (dynamic)

2.7. การแนะนำประเภทที่ใกล้เคียง

2.7.1. ใช้ค่าคงที่ (static) โดยให้ประเภทที่มีความใกล้เคียงกันมาก มีค่าในการแนะนำที่มาก

2.7.2. อัลกอริทึม: การกลับค่าจากค่าแนะนำมากเป็นเส้นทางที่สั้น และค่าที่แนะนำน้อยเป็นเส้นทางที่ยาว และการใช้ขั้นตอนวิธีของฟลอยด์-วอร์เชล (Floyd-Warshall algorithm) ในการหาเส้นทางที่สั้นที่สุด กล่าวคือการหาประเภทที่มีความสัมพันธ์กันมากที่สุดในการแนะนำ

2.8. การจัดการข้อมูลผู้ใช้

2.8.1. ใช้ฟังก์ชันในการเพิ่ม ลบ และแก้ไขข้อมูลผู้ใช้

2.8.2. อัลกอริทึม: คล้ายกับการจัดการข้อมูลสินค้าโดยใช้ลิงค์ลิสต์ (Linked list)

2.9. การติดตามประวัติการซื้อขาย

2.9.1. ใช้ฟังก์ชันในการบันทึกข้อมูลการซื้อขายทุกครั้งที่มีการทำธุรกรรม

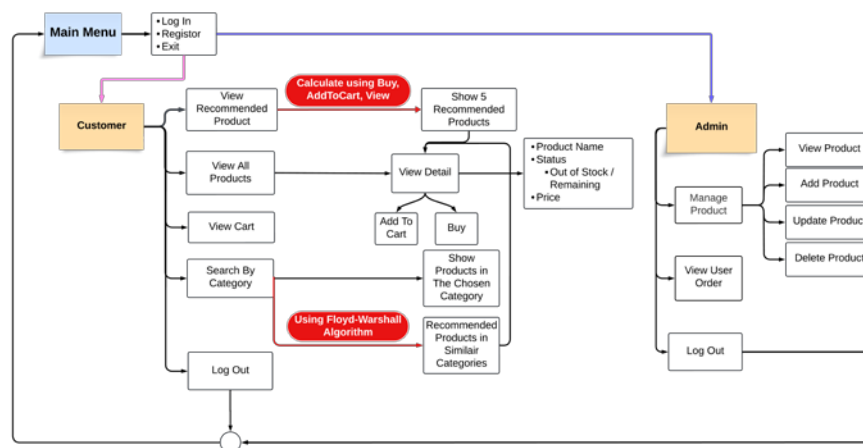
2.9.2. อัลกอริทึม : การแทรกข้อมูลในลิงค์ลิสต์ (Linked list) และเขียนประวัติลงบนไฟล์ (File)

2.10. ระบบตะกร้าสินค้าและการคิดเงิน

2.10.1. ใช้ฟังก์ชันในการเพิ่มสินค้าลงตะกร้า คำนวณยอดรวม และแสดงรายการสินค้าที่จะซื้อ

2.10.2. อัลกอริทึม : การแทรกและค้นหาข้อมูลในลิงค์ลิสต์ (Linked list) และการคำนวณผลรวม

3. วิธีการใช้งานและการทดสอบการทำงานของโปรแกรม



3.1. ผู้ใช้เริ่มต้นโดยการเข้าสู่ระบบด้วยข้อมูลผู้ใช้

3.2. เมนูหลักจะแสดงตัวเลือกต่างๆ เช่น การจัดการสินค้า การจัดการผู้ใช้ และการดูประวัติการซื้อ

3.3. ผู้ใช้สามารถเลือกฟังก์ชันที่ต้องการดำเนินการ เช่น การเพิ่มสินค้าใหม่ การลบสินค้า หรือการดูประวัติการซื้อ

3.4. เมื่อเลือกฟังก์ชัน ระบบจะแสดงขั้นตอนการดำเนินการและรับข้อมูลจากผู้ใช้

4. อุปกรณ์ เครื่องมือ หรือโปรแกรมที่ใช้ในการพัฒนา

4.1. Visual Studio Code

4.2. GitHub

4.3. Microsoft Word

4.4. Discord

5. โค้ดโปรแกรม

ตรวจสอบที่ <https://github.com/Srisuphadith/CashierXByShopWiz>

บทที่ 5

สรุปผลโครงการ อภิปรายผล และข้อเสนอแนะ

ผลการดำเนินงานของโครงการพัฒนาโปรแกรมบริหารจัดการสินค้าภายในร้านค้าสามารถสรุปได้ดังนี้

1. ความสำเร็จในการพัฒนาโปรแกรม
 - 1.1. โปรแกรมบริหารจัดการสินค้าภายในร้านค้าสามารถพัฒนาได้สำเร็จตามแผนที่กำหนด โดยมีฟังก์ชันหลักต่างๆ ครบถ้วนตามวัตถุประสงค์ที่วางไว้
 - 1.2. โปรแกรมมีการจัดการข้อมูลสินค้าที่มีประสิทธิภาพ สามารถเพิ่ม ลบ และแก้ไขข้อมูลสินค้าได้อย่างง่ายดายและรวดเร็ว
2. การจัดการข้อมูลผู้ใช้
 - 2.1. ระบบการจัดการข้อมูลผู้ใช้งานได้ดี ผู้ดูแลสามารถเพิ่ม ลบ และแก้ไขข้อมูลผู้ใช้ได้อย่างมีประสิทธิภาพ
 - 2.2. ระบบสามารถตรวจสอบและติดตามกิจกรรมของผู้ใช้ได้ ซึ่งช่วยให้การจัดการผู้ใช้งานมีความเป็นระบบและปลอดภัยมากยิ่งขึ้น
3. การติดตามประวัติการซื้อขาย
 - 3.1. ระบบประวัติการซื้อขายสามารถบันทึกและติดตามการซื้อขายของลูกค้าได้อย่างละเอียดและแม่นยำ
 - 3.2. ข้อมูลการซื้อขายสามารถนำมาใช้วิเคราะห์เพื่อปรับปรุงกลยุทธ์การขายและการบริการลูกค้าได้อย่างมีประสิทธิภาพ
4. การใช้งานและการตอบรับจากผู้ใช้
 - 4.1. โปรแกรมมีอินเทอร์เฟซที่ใช้งานง่ายและเป็นมิตรกับผู้ใช้ ทำให้ผู้ดูแลร้านค้าสามารถเรียนรู้และใช้งานได้อย่างรวดเร็ว
 - 4.2. ผู้ใช้ทดลองใช้งานแสดงความพึงพอใจในประสิทธิภาพและความสะดวกในการใช้งานของโปรแกรม
5. การทดสอบและปรับปรุงโปรแกรม
 - 5.1. การทดสอบหน่วย (Unit Testing) และการทดสอบระบบ (System Testing) แสดงผลลัพธ์ที่ดี ไม่มีข้อผิดพลาดหรือบั๊กที่สำคัญ
 - 5.2. การทดสอบผู้ใช้ (User Acceptance Testing) ได้รับผลตอบแทนเชิงบวก โดยมีการปรับปรุงตามข้อเสนอแนะของผู้ใช้เพื่อให้โปรแกรมใช้งานได้ง่ายและมีประสิทธิภาพยิ่งขึ้น
6. ประโยชน์ที่ได้รับ
 - 6.1. โปรแกรมช่วยลดเวลาและความยุ่งยากในการจัดการสินค้าและผู้ใช้ ทำให้การดำเนินงานภายในร้านค้ามีความเป็นระเบียบและมีประสิทธิภาพมากขึ้น

6.2. การติดตามและวิเคราะห์ประวัติการซื้อขายช่วยให้ร้านค้าสามารถปรับปรุงการบริการและการบริหารจัดการให้ตอบสนองความต้องการของลูกค้าได้ดียิ่งขึ้น

อภิปรายผล

จากผลการดำเนินงานทั้งหมด โปรแกรมบริหารจัดการสินค้าภายในร้านค้านี้ได้แสดงให้เห็นถึงความสำเร็จในการพัฒนาตามวัตถุประสงค์ที่วางไว้ และสามารถนำมาใช้ประโยชน์ในการบริหารจัดการร้านค้าได้อย่างแท้จริง

ปัญหาและอุปสรรคในการดำเนินงาน

ในการพัฒนาโปรแกรมบริหารจัดการสินค้าภายในร้านค้านี้ ได้พบกับปัญหาและอุปสรรคหลายประการที่ส่งผลกระทบต่อการทำงานทั้งในส่วนของการพัฒนาโปรแกรมและการแบ่งงานภายในทีม ซึ่งสามารถสรุปได้ดังนี้

1. ปัญหาด้านโปรแกรม

1.1. ความแตกต่างของ Compiler

1.1.1. โปรแกรมสามารถทำงานได้เพียงแค่เครื่องเดียว เนื่องจากความแตกต่างของ Compiler ที่ใช้ในการพัฒนา เมื่อทำการย้ายโปรแกรมไปยังเครื่องอื่น กลับพบว่าปัญหาในการคอมไพล์และรันโปรแกรม ทำให้ต้องเสียเวลาในการปรับแก้ไขโค้ดเพื่อให้สามารถทำงานได้ในทุกเครื่อง ซึ่งเป็นปัญหาที่สร้างความยุ่งยากและทำให้กระบวนการทดสอบโปรแกรมใช้เวลานานกว่าที่คาดคิด

1.2. การเรียงลำดับใน Heap

1.2.1. ในตอนแรก โปรแกรมใช้โครงสร้างข้อมูลแบบ Heap ในการจัดการและเรียงลำดับค่าต่างๆ เช่น การเรียงลำดับสินค้าตามค่า suggestion จากมากไปน้อย แต่พบว่า Heap ไม่สามารถเรียงลำดับได้อย่างถูกต้องหรืออาจต้องใช้เวลามากกว่าการ traversal ปกติ ส่งผลให้การทำงานล่าช้าและมีความซับซ้อนเพิ่มขึ้น ในที่สุดจึงต้องเปลี่ยนไปใช้ AVL Tree เพื่อแก้ไขปัญหานี้

1.3. การใช้งาน AVL Tree

1.3.1. เมื่อใช้ AVL Tree กับข้อมูลแบบพลวัต (Dynamic) พบว่ามีความยุ่งยากในการเพิ่มและลบสินค้าต่างๆ เนื่องจากการปรับสมดุลของต้นไม้ (tree balancing) ที่ต้องทำทุกครั้งเมื่อมีการเพิ่มหรือลบโนด ทำให้กระบวนการเพิ่มและลบข้อมูลมีความซับซ้อนและทำงานช้าลง สุดท้ายจึงตัดสินใจเปลี่ยนไปใช้โครงสร้างข้อมูลแบบ Linked List ซึ่งสามารถจัดการกับข้อมูลพลวัตได้ง่ายกว่าและมีความยืดหยุ่นมากขึ้นในการจัดการข้อมูลสินค้า

2. ปัญหาด้านการแบ่งงาน

2.1. การสื่อสารภายในทีม

2.1.1. การสื่อสารภายในทีมบางครั้งเกิดความเข้าใจที่คลาดเคลื่อน เนื่องจากการตีความที่ไม่ตรงกันหรือการสื่อสารที่ไม่ชัดเจน ส่งผลให้การทำงานเกิดความล่าช้าและ

คลาดเคลื่อนจากจุดประสงค์ที่ตั้งไว้ การแบ่งงานในบางครั้งไม่ได้เป็นไปตามแผนที่วางไว้ ทำให้ต้องมีการปรับเปลี่ยนและแก้ไขอยู่บ่อยครั้ง ซึ่งเป็นอุปสรรคในการดำเนินงาน และทำให้การพัฒนาโปรแกรมช้ากว่าที่คาดการณ์ไว้

แนวทางการแก้ไข

1. แก้ไขความแตกต่างของ Compiler

1.1. ปรับแก้โค้ดให้มีความยืดหยุ่นและสามารถคอมไพล์ได้ในทุกเครื่องที่ใช้ Compiler ที่ต่างกัน พร้อมทั้งจัดทำเอกสารแนะนำการตั้งค่า Compiler ที่เหมาะสมเพื่อให้ทุกคนในทีมสามารถตั้งค่าการพัฒนาในเครื่องของตนเองได้อย่างถูกต้อง

2. ปรับโครงสร้างข้อมูล

2.1. จากการทดลองใช้ Heap และ AVL Tree แล้วพบปัญหา ข้าพเจ้าได้ตัดสินใจเปลี่ยนไปใช้ Linked List ซึ่งช่วยให้การจัดการข้อมูลสินค้ามีความง่ายขึ้นและลดความซับซ้อนในการพัฒนา

3. ปรับปรุงการสื่อสารในทีม

3.1. จัดประชุมทีมอย่างสม่ำเสมอเพื่อสรุปความคืบหน้าและปัญหาที่พบในการพัฒนา พร้อมทั้งใช้เครื่องมือสื่อสารออนไลน์ที่ช่วยในการแบ่งปันข้อมูลและการติดต่อสื่อสารภายในทีมได้อย่างรวดเร็วและมีประสิทธิภาพยิ่งขึ้น