Term Project : Insertion Sort

เสนอ

อาจารย์ ธิติพร ประมวล

จัดทำโดย

1.	นาย กฤษฎา	ประเสริฐศิลป์	รหัสนิสิต	64109010178
2.	นาย รัตนพงษ์	สว่างอารมณ์	รหัสนิสิต	64109010179
3.	นาย กันตพัฒน์	ตั้งธนพาณิชย์	รหัสนิสิต	64109010397
4.	นาย ภูเบศ	โควาภิรักษ์	รหัสนิสิต	64109010402

รายงานนี้เป็นส่วนหนึ่งของวิชา CPE200 Discrete Mathematics

ภาคเรียนที่ 1 ปีการศึกษา 2565

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ

ชื่อวิชา รายวิชา CPE200 Discrete Mathematics

ชื่อโครงงาน Cash Book

หัวข้อหลักที่ได้ Insertion Sort

ผู้ดำเนินงาน

1.	นาย	กฤษฎา	ประเสริฐศิลป์	รหัสนิสิต	64109010178
2.	นาย	รัตนพงษ์	สว่างอารมณ์	รหัสนิสิต	64109010179
3.	นาย	กันตพัฒน์	ตั้งธนพาณิชย์	รหัสนิสิต	64109010397
4.	นาย	ภูเบศ	โควาภิรักษ์	รหัสนิสิต	64109010402

อาจารย์ที่ปรึกษา อาจารย์ธิติพร ประมวล

ปีการศึกษา ภาคเรียนที่ 1 ปีการศึกษา 2565

คำนำ

รายงานฉบับนี้เป็นส่วนหนึ่งของวิชา CPE200 Discrete Mathematics จัดทำขึ้นเพื่อแสดงถึงแผนงานต่าง ๆ ของโปรเจค โดยจะนำอัลกอริทึมแบบ Insertion Sort มาประยุกต์ใช้เพื่อสร้างเว็บไซต์สำหรับบันทึกรายรับ-รายจ่าย โดยรายละเอียดในรายงานฉบับนี้มีจะมี แผนงานโปรเจคของสมาชิกแต่ละคน ระยะเวลา ในแต่ละขั้นตอน ทฤษฎีที่เกี่ยวข้อง รายละเอียดของเครื่องมือที่ใช้ลักษณะและส่วนประกอบของโปรเจค รวมไปถึงวิธีการใช้งานโปรเจคนี้อีกด้วย

คณะผู้จัดทำ

สารบัญ

Project Planing

วัน/เวลา	แผนการดำเนินงาน
10-16 ส.ค.	ศึกษาข้อมูลของ Insertion Sort ที่เลือก และเลือกหัวข้อที่จะทำ
17-23 ส.ค.	ออกแบบโครงสร้างส่วน ประกอบของหน้าเว็บ ศึกษา Code HTML ทำสไลด์และเตรียมนำเสนอ
24-30 ส.ค.	นำเสนอครั้งที่1ศึกษา Code css
31 ส.ค. - 6 ก.ย.	ศึกษาอัลกอริทึมการทำงานของ Insertion Sort เพิ่มเติม
7-13 ก.ย.	ศึกษาและทดลองเขียน Code JavaScript
14-20 ก.ย.	นำ Code มารวมกันเพื่อสร้างเว็บ ออกแบบกราฟฟิก หา Error เพื่อนำมาปรับปรุง ทำสไลด์และเตรียมนำเสนอ
21-27 ก.ย.	นำเสนอครั้งที่2 ปรับปรุงตามคำแนะนำของอาจารย์
28ก.ย.-4 ต.ค.	ปรับปรุงหน้าเว็บไซต์ หา Error เพื่อนำมาปรับปรุง เตรียมตัวสอบ Midterm
5-11 ต.ค.	สอบ Midterm
12-18 ต.ค.	พัฒนาหน้าเว็บ ทดสอบการใช้ หาข้อผิดพลาด
19-25 ต.ค.	หาฟังก์ชันมาพัฒนาหน้าเว็บเพิ่มเติม ปรับปรุงแก้ไขสิ่งที่ผิดพลาดต่างๆ

26ต.ค.-1พ.ย.	ทดสอบครั้งสุดท้ายก่อนนำใส่รายงาน
2-8 พ.ย.	ส่งรายงาน ปรับปรุงและพัฒนา
9-15 พ.ย.	ปรับปรุงงาน ทดสอบหน้าเว็บ
16-22 พ.ย.	ปรับปรุงและพัฒนาหน้าเว็บ ทดสอบหน้าเว็บ เตรียมนำเสนอ ซ้อมนำเสนอ
23 พ.ย.	นำเสนองาน final

ทฤษฎีที่เกี่ยวข้อง

การเรียงลำดับข้อมูล

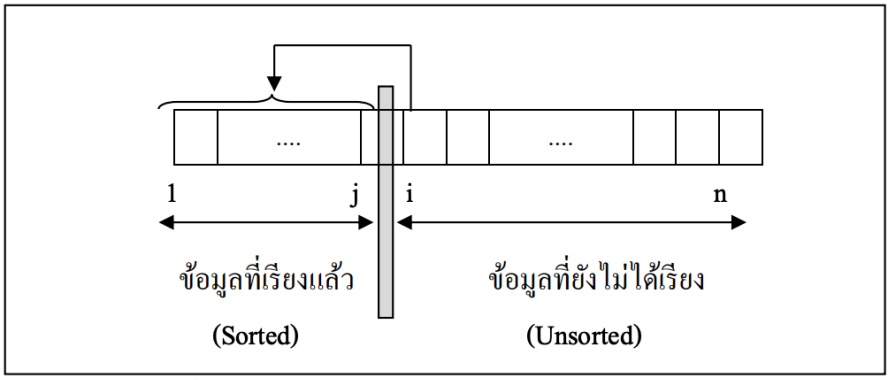
การเรียงลำดับข้อมูลหรือการจัดเรียงข้อมูล(DataSorting) เป็นการนำค่าของข้อมูลทุกค่ามาจัดลำดับต่อเนื่องกันไปจากน้อยไปมากหรือจากมากไปน้อยก็ได้ซึ่งนับว่าเป็นสิ่งสำคัญมากอีกอย่างหนึ่งในการดำเนินการกับข้อมูล เพราะนอกจากจะเป็นการจัดระเบียบข้อมูลแล้ว ยังช่วยทำให้สามารถค้นหาข้อมูลหรือดำเนินการอย่างอื่นเป็นไปอย่างสะดวกและรวดเร็วมากกว่าข้อมูลที่ไม่ได้ทำการเรียงลำดับ สำหรับการเรียงลำดับข้อมูลในคอมพิวเตอร์นั้นมีหลากหลายรูปแบบ แต่ละรูปแบบมีทั้งข้อดีและข้อเสียที่แตกต่างกันไป ดังนั้นผู้ใช้จะต้องเลือกใช้ให้เหมาะสมกับข้อมูลแต่ละชนิด

ประเภทของการเรียงลำดับข้อมูล ประเภทของการเรียงลำดับข้อมูลในคอมพิวเตอร์ แบ่งออกเป็น 2 ประเภท คือ

1. การเรียงลำดับข้อมูลภายใน (Internal Sorting) เป็นวิธีการจัดเรียงข้อมูลภายใน หน่วยความจำหลัก (Main Memory) ของเครื่องคอมพิวเตอร์ นั่นคือ
ข้อมูลทั้งหมดจะอยู่ในหน่วยความจำหลักในระหว่างการประมวลผลเพื่อการจัดเรียงข้อมูล
ซึ่งวิธีนี้เหมาะกับข้อมูลที่มี ปริมาณไม่มาก หรือไม่เกินขนาดของหน่วยความจำหลัก
ตัวอย่างการเรียงลำดับข้อมูลภายใน เช่น การเรียงลำดับข้อมูลแบบ Insertion Sort, Shell Sort, Selection Sort, Heap Sort, Bubble Sort และ Quick Sort เป็นต้น

2. การเรียงลำดับข้อมูลภายนอก (External Sorting) เป็นวิธีการจัดเรียงข้อมูลที่
เหมาะกับข้อมูลที่มีปริมาณมาก ไม่สามารถนำไปเก็บไว้ในหน่วยความจำหลักเพื่อประมวลผลได้หมดในคราวเดียว จึงจำเป็นต้องแบ่งข้อมูลออกเป็นส่วนๆ พอที่จะจัดเก็บในหน่วยความจำหลักได้ และมีข้อมูลบางส่วนถูกจัดเก็บไว้ในหน่วยความจำสำรอง เช่น ฮาร์ดดิสก์ หรือดิสก์เก็ต เมื่อข้อมูลที่อยู่ในหน่วยความจำหลักจัดเรียงเรียบร้อยแล้วข้อมูลจะถูกนำไปบันทึกไว้ที่หน่วยความจำสำรอง จากนั้นจะทำการอ่านข้อมูลส่วนอื่นที่ยังไม่มีการเรียงลำดับเข้าไปในหน่วยความจำหลัก ทำการจัดเรียงข้อมูลด้วยขั้นตอนต่างๆ จนเสร็จ แล้วจึงนำข้อมูลที่เรียงลำดับแล้วแต่ละส่วนมารวมเข้าด้วยกันอีกครั้งหนึ่ง
ตัวอย่างการเรียงลำดับข้อมูลภายนอก เช่น การเรียงลำดับข้อมูลแบบ Merge Sort เป็นต้น

การเรียงลำดับข้อมูลแบบ Insertion Sort การเรียงลำดับข้อมูลแบบ Insertion Sort หรือการเรียงลำดับแบบแทรก มีลักษณะคล้าย การเรียงไพ่ด้วยมือ โดยจะทำการจัดเรียงไพ่ด้วยการดึงไพ่ที่ต้องการออกมาแทรกใน ตำแหน่งที่ เหมาะสมในช่วงลำดับของการเรียงในขณะนั้น



รูปที่ 1 หลักการเรียงลำดับข้อมูลแบบ Insertion Sort

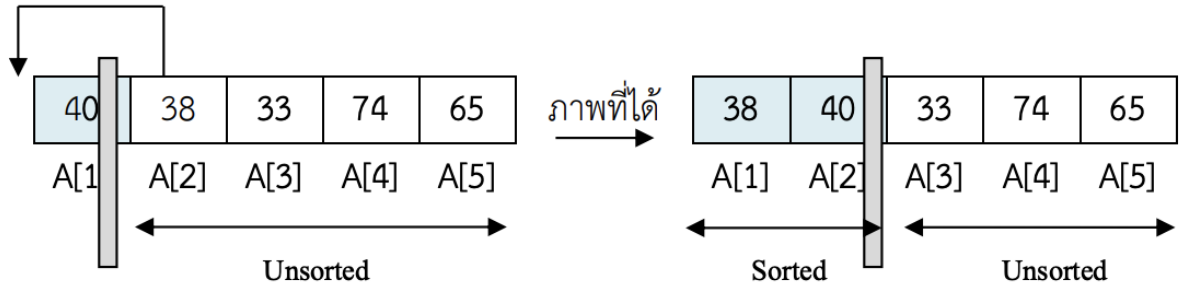
จากรูปที่ 1 การเรียงลำดับข้อมูลแบบ **Insertion Sort** จะใช้หลักการกำหนดพื้นที่เป็น 2 ส่วน โดยส่วนแรกใช้ในการจัดเก็บข้อมูลที่เรียงลำดับแล้ว **(Sorted)** และส่วนที่สองใช้เก็บข้อมูลที่ยังไม่ได้เรียงลำดับ **(Unsorted)** ซึ่งมีวิธีการจัดเรียงดังนี้คือ นำข้อมูลในส่วนที่ยังไม่ได้จัดเรียง **A[i]** ทีละตัวมาเปรียบเทียบกับข้อมูลก่อนหน้าที่เรียงลำดับแล้วทั้งหมด ถ้ามีค่าน้อยกว่าจะนำไปแทรกในตำแหน่งที่ถูกต้อง เพื่อความเข้าใจให้พิจารณาตัวอย่างที่ 8.2 แสดงภาพขั้นตอนการเรียงลำดับข้อมูลแบบ **Insertion Sort** โดยจัดเรียงข้อมูลจากน้อยไปมาก

ตัวอย่างที่ 1 แสดงภาพขั้นตอนการเรียงลำดับข้อมูลแบบ **Insertion Sort** โดยจัดเรียงข้อมูลจากน้อยไปมาก กำหนดให้ข้อมูลถูกจัดเก็บในอาร์เรย์ A ดังนี้

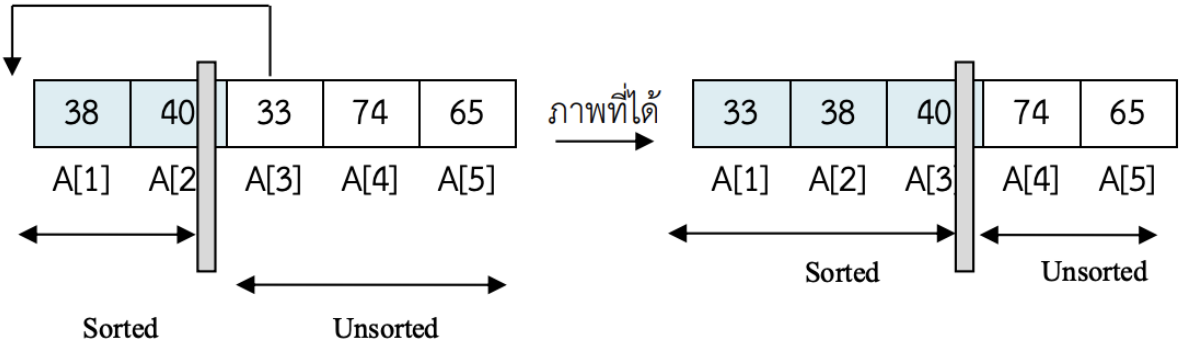
40	38	33	74	65
A[1]	A[2]	A[3]	A[4]	A[5]

หลักการทำงาน การจัดเรียงข้อมูลจะเริ่มจากซ้ายไปขวา โดยกำหนดให้ข้อมูลในช่องแรกเป็นข้อมูลที่เรียงลำดับแล้ว ส่วนตั้งแต่ช่องที่ 2 เป็นต้นไปเป็นข้อมูลส่วนที่ยังไม่เรียงลำดับ ขั้นตอนการจัดเรียงข้อมูลเป็นดังนี้

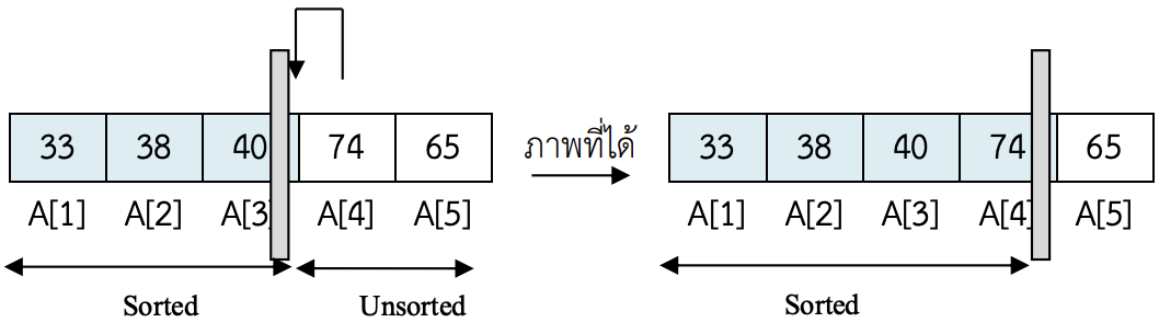
รอบที่ 1 จากข้อมูลต้นฉบับ ซึ่งยังไม่ผ่านการจัดเรียง พิจารณาเมื่อ i = 2; A[i] มีค่าเป็น 38 ทำการเปรียบเทียบค่าตัวเลขช่องที่ 1 และช่องที่ 2 ผลปรากฏว่าค่าตัวเลข 38 มีค่าน้อยกว่า จึง นำไปแทรกในตำแหน่ง A[1]



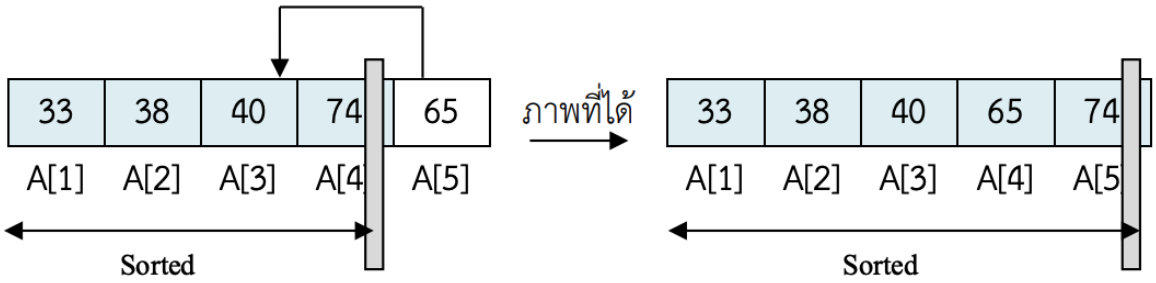
รอบที่ 2 พิจารณาเมื่อ i = 3; A[i] มีค่าเป็น 33 เมื่อนำไปเปรียบเทียบกับชุดตัวเลข ที่ เรียงลำดับแล้วทั้งหมด ซึ่งอยู่ส่วนหน้า ผลปรากฏว่าค่าตัวเลข 33 จะถูกนำไปแทรกที่ตำแหน่ง A[1]



รอบที่ 3 พิจารณาเมื่อ i = 4 ; A[i] มีค่าเป็น 74 เมื่อนำไปเปรียบเทียบกับชุดตัวเลขที่ เรียงลำดับแล้วทั้งหมด ซึ่งอยู่ส่วนหน้า ผลปรากฏว่าค่าตัวเลข 74 อยู่ในลำดับถูกต้องแล้วจึงไม่มีการ แทรก



รอบที่ 4 พิจารณาเมื่อ i = 5 ; A[i] มีค่าเป็น 65 เมื่อนำไปเปรียบเทียบกับชุดตัวเลขที่ เรียงลำดับแล้วทั้งหมด ซึ่งอยู่ส่วนหน้า ผลปรากฏว่าค่าตัวเลข 65 จะถูกนำไปแทรกที่ตำแหน่ง A[4] เนื่องจากรอบนี้เป็นรอบสุดท้าย เมื่อทำการแทรกเรียบร้อยแล้วก็จะได้ชุดตัวเลขที่ผ่านการจัดเรียงแล้ว อย่างสมบูรณ์



การเรียงลำดับข้อมูลแบบ Insertion Sort โดยทั่วไปจะมีการจัดเรียง n-1 รอบ เมื่อ n คือ จำนวนข้อมูลทั้งหมด และในแต่ละรอบจำนวนข้อมูลการเปรียบเทียบคีย์ไม่แน่นอน ขึ้นอยู่กับ สภาพแวดล้อม วิธีนี้เหมาะกับชุดข้อมูลขนาดเล็กหรือชุดข้อมูลที่มีการเรียงลำดับบ้างแล้ว

การเขียนโค้ดการเรียงลำดับข้อมูลแบบ Insertion Sort

```
1   const insertion_Sort = (arr) => {
2       for (let i = 1 ; i < arr.length; i++) {
3           let temp = arr[i]
4           let j = i - 1
5
6         while (j >= 0 && arr[j] < temp) {
7           arr [j + 1] = arr [j]
8           j--
9         } //End While
10        arr[j+1] = temp
11      } //End for
12      return arr
13    }
14    console.log(insertion_Sort([50, 130,22, 55, 400, 30]));
15    //End
```

บรรทัดที่ 1 :ประกาศตัวแปรที่ชื่อว่า insertion_Sort ให้เก็บค่า Parameter ที่ชื่อว่า arr

บรรทัดที่ 2 :ใช้คำสั่ง For ให้วนลูป ตามค่าที่ i กำหนดไว้นั้นก็คือให้ i = 1 ตราบใดที่ i < arr.length เมื่อเป็นจริงให้ทำในวงเล็ปและเมื่อเสร็จคำสั่งในลูป ให้เพิ่มค่า i 1 ค่า จนกว่าเงื่อนไขจะเป็นเท็จ

(ฟังก์ชัน length ⟶ คือคำสั่งที่นับจำนวน Array และจะคืนค่ากลับมาเป็นตัวเลขที่นับได้)

บรรทัดที่ 3:กำหนดตัวแปร temp เท่ากับ arr[i] หมายถึงใน Parameten arr ให้เก็บค่าที่ตำแหน่ง i

บรรทัดที่ 4: กำหนดตัวแปรให้ j เท่ากับ i-1 หมายถึง j เท่ากับ ตำแหน่ง i ลดไป 1 ตำแน่ง ให้เก็บค่าที่ตำแหน่ง j

บรรทัดที่ 6: j >= 0 && arr[j] < temp เมื่อเป็นจริงให้ทำงานในปีกกา
(Loop while โปรแกรมจะทำงานในลูปในขณะที่เงื่อนไขเป็นจริง
ดังนั้นต้องทำอะไรบ้างอย่างเพื่อทำให้เงื่อนไขเป็นเท็จ เพื่อจบการทำงานของลูป while)

บรรทัดที่ 7: arr [j + 1] = arr [j] arr [j + 1] หมายถึง เลื่อนตำแหน่งไปหนึ่งตำแหน่งโดยให้ arr [j] = ตำแหน่ง 0 เหมือนเดิม
บรรทัดที่ 8: j-- หมายถึง ลดค่า 1 ค่า ซึ่ง j = ตำแหน่งที่ 0 ลดไป 1 ตำแหน่ง = ตำแหน่งที่ -1พอวนลูปใหม่ j = ตำแหน่ง -1 ซึ่งไม่ตรงกับเงื่อนไขที่ให้ไว้ว่า J >= 0 ดังนั้นสมการนี้จึงเป็นเท็จทำให้จบ Loop while

บรรทัดที่ 9: จบการทำงานใน Loop While

บรรทัดที่ 10 : arr[j+1] = temp หลังจากจบลูป while ให้นำค่า temp ไปไว้ในตำแหน่ง j ที่เลื่อนไปอีก 1 ตำแหน่ง

บรรทัดที่ 11: จบการทำงานใน Loop For

บรรทัดที่ 12: เป็นการสั่งให้ฟังก์ชันจบการทำงานและส่งค่ากลับไปยังจุดเรียกฟังก์ชัน arr

บรรทัดที่ 14:เป็นการแสดงค่าตัวแปร insertion_Sort ที่เรากำหนดตัว Array ไว้ใน Square brackets

รายละเอียดเครื่องมือที่ใช้ในการพัฒนาโปรเจค

LAPTOP (เครื่องมือที่ใช้เขียนและพัฒนาเว็บไซต์)

NGROK (ฐานข้อมูลในการอัพโหลดขึ้นserverชั่วคราว)

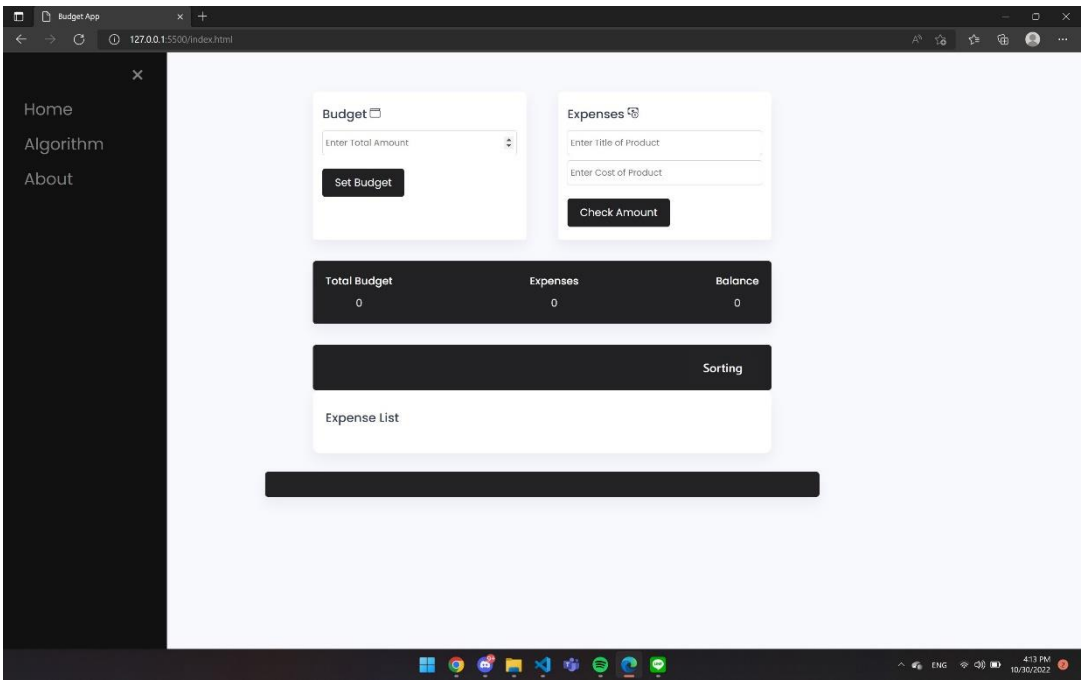
HTML (เขียนหน้าเว็บให้แสดงผลผ่านบราวเซอร์)

CSS (ช่วยจัดรูปแบบแสดงผลให้ภาษาhtmlลดลงและจัดการเรื่องการตกแต่งเว็บไซต์)

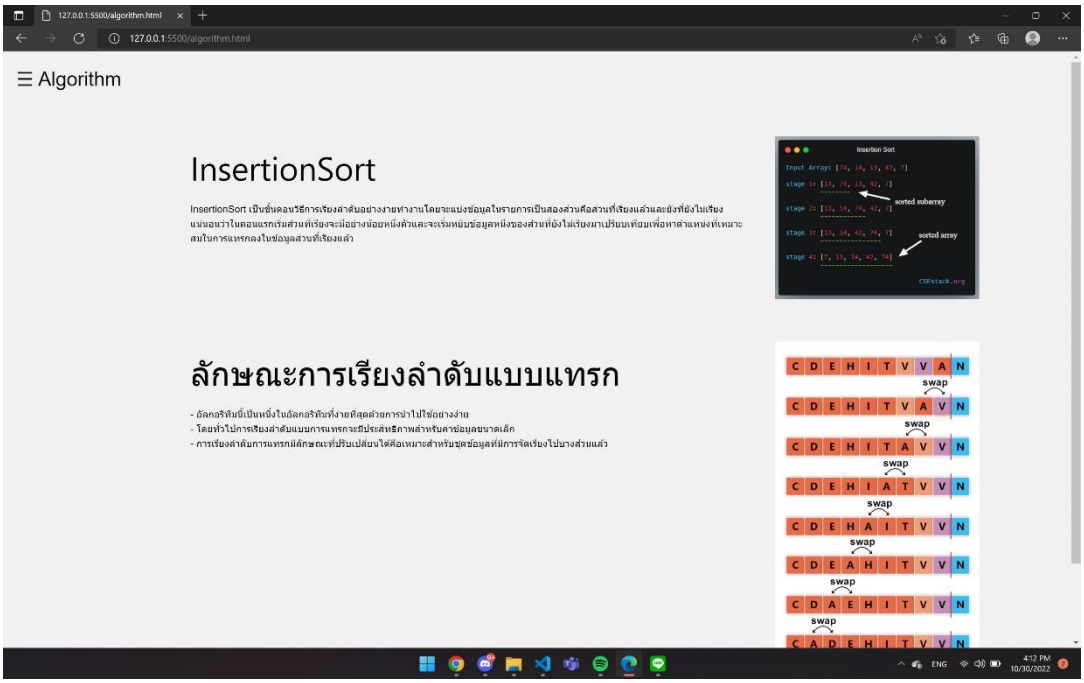
JAVASCRIPT (ใช้เขียนฟังก์ชันในการพัฒนาหน้าเว็บไซต์)

Visual Studio Code (ใช้เป็นcompiler)

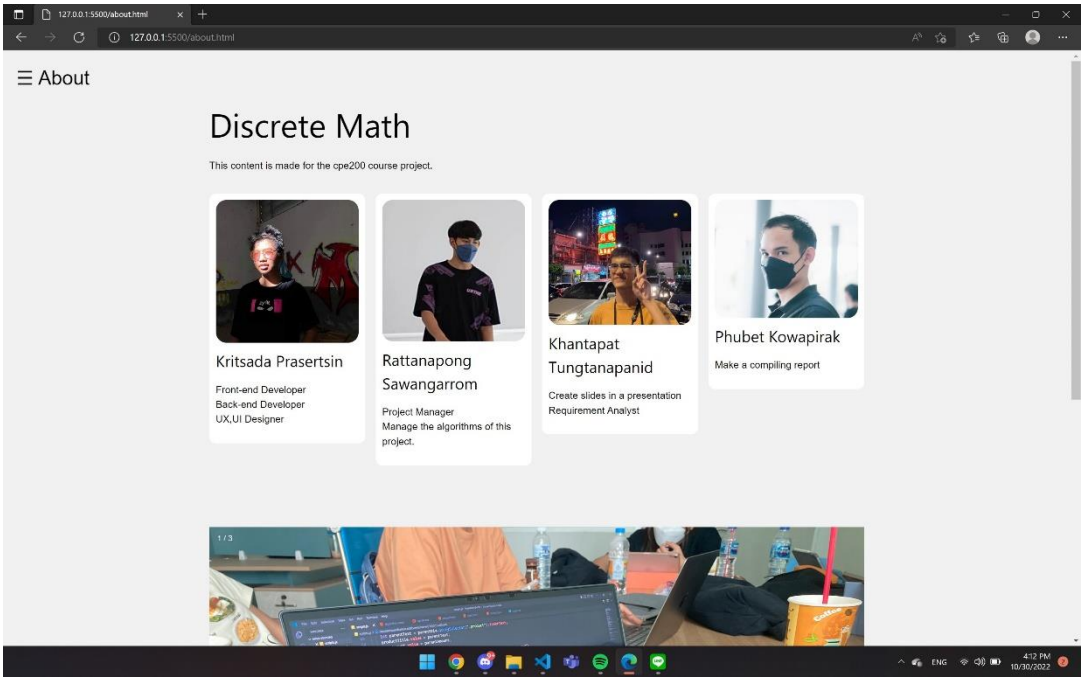
ลักษณะและส่วนประกอบของโปรเจค



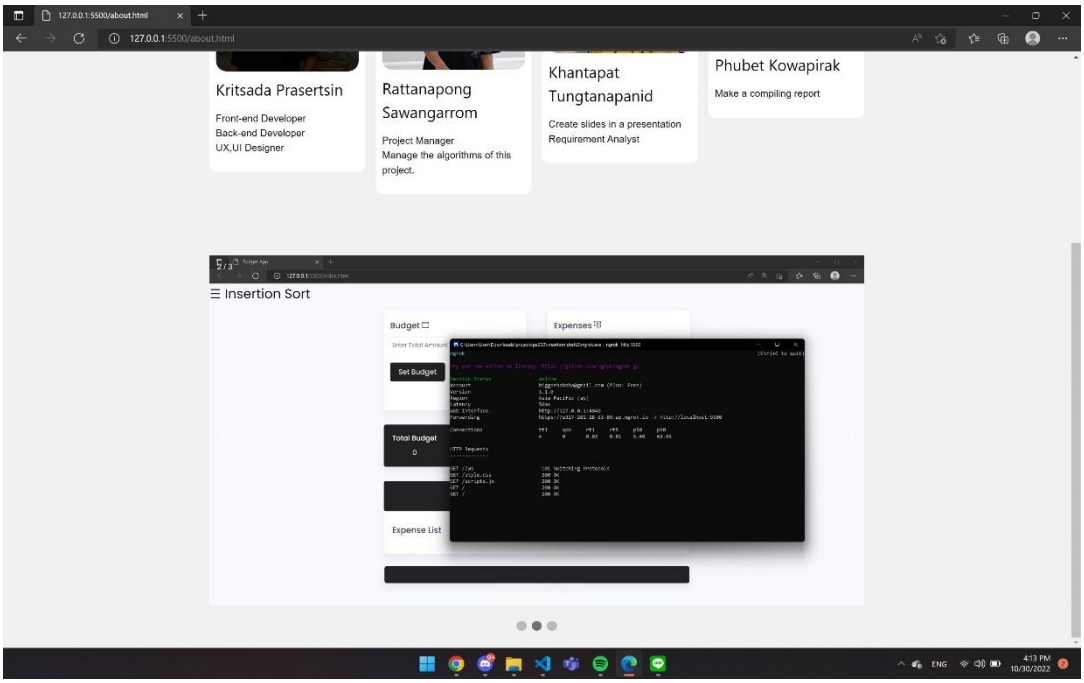
หน้าหลัก



หน้าอธิบาย **Algorithm**



เกี่ยวกับสมาชิก



Server ชั่วคราว

วิธีใช้งานโปรเจค