บทที่ 8

การค้นหา (searching)

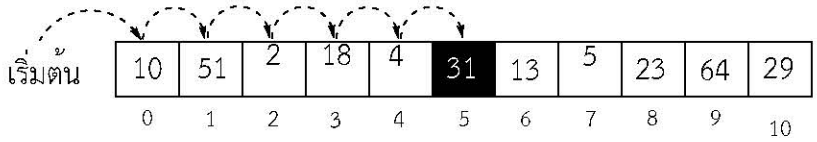
การค้นหาเป็นหัวข้อที่สำคัญสำหรับนักศึกษาในสาขาวิทยาการคอมพิวเตอร์ ที่เกี่ยวข้องกับอัลกอริทึมพื้นฐานที่ใช้ในการค้นหาข้อมูลในโครงสร้างข้อมูลต่าง ๆ ที่สำคัญคือการเลือกใช้อัลกอริทึมค้นหาขึ้นอยู่กับโครงสร้างข้อมูลที่ใช้งาน เมื่อมีการใช้งานลิสต์ จะพบกับเทคนิคการค้นหาสองวิธีหลัก คือ การค้นหาแบบเชิงเส้นที่สามารถใช้กับข้อมูลที่เรียงและไม่เรียง แม้อาจมีความซับซ้อนของเวลาสูง และการค้นหาแบบไบนารีที่เป็นวิธีเฉพาะที่ใช้กับข้อมูลที่เรียงแล้ว การค้นหาแบบเชิงเส้นเน้นการตรวจสอบทุกข้อมูลในลิสต์อย่างตามลำดับเพื่อค้นหาค่าที่ระบุ ในทางตรงกันข้าม การค้นหาแบบไบนารีจะใช้ลิสต์ที่เรียงแล้ว และลดขนาดของลิสต์ย่อยที่กำลังพิจารณาซ้ำ ๆ จนกว่าจะค้นพบค่าที่ต้องการหรือลิสต์ย่อยจะว่างเปล่า โดยการเปรียบเทียบค่าที่ต้องการกับค่ากลาง เพื่อกำหนดว่าค่าเป้าหมายมากกว่า น้อยกว่า หรือเท่ากับค่ากลาง นี้ช่วยให้สามารถลบครึ่งขององค์ประกอบที่เหลือในทุก ๆ ขั้นตอน ขั้นตอนการลดขนาดของลิสต์ย่อยที่ซ้ำนี้ทำให้การค้นหาแบบไบนารีมีเวลาเร็นทาง O(log n) ที่มีประสิทธิภาพสำหรับการค้นหาข้อมูลในลิสต์ที่เรียง

การค้นหาแบบเชิงเส้น

การค้นหาเชิงเส้น (linear search) หรือ การค้นหาแบบลำดับ (sequential search) เป็นวิธีที่ง่ายที่สุดในการค้นหาภายในลิสต์ การค้นหาเป้าหมายด้วยวิธีนี้จะค้นหาตามลำดับในลิสต์สามารถนำไปใช้กับลิสต์เรียง (sorted list) หรือไม่เรียง (unsorted list) ก็ได้ หากพบตำแหน่งเป้าหมายที่ต้องการภายในลิสต์ แสดงว่ากระบวนการค้นหาสำเร็จ (successful search) ถ้าไม่พบตำแหน่งเป้าหมายที่ต้องการภายในลิสต์ แสดงว่ากระบวนการค้นหาล้มเหลว (unsuccessful search) เช่น ต้องการค้นหาค่า 31 ในลิสต์ที่ไม่ได้เรียง การค้นหาเริ่มตั้งแต่ข้อมูลแรกภายในลิสต์ และตรวจสอบข้อมูลภายในลิสต์ว่าตรงกับเป้าหมายที่ค้นหาหรือไม่ ถ้าไม่ตรงจะทำการค้นหาทีละข้อมูลจนพบในตำแหน่งที่หก ดังภาพประกอบที่ 8.1 ถ้าต้องการค้นหาค่า 8 การค้นหาเริ่มต้นที่ข้อมูลแรกเช่นเดิม ซึ่งข้อมูลทั้งหมดในลิสต์จะถูกค้นหาและไม่สามารถระบุได้ว่าค่าไม่อยู่ในลำดับจนกระทั่งลิสต์ทั้งหมดถูกค้นหา ดังภาพประกอบที่ 8.2

Sample

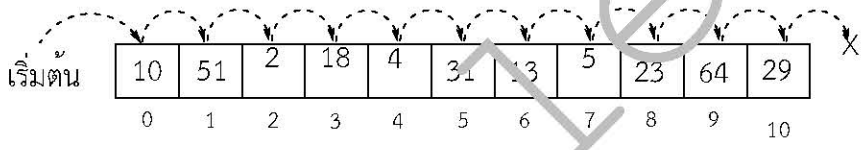
การค้นหา 31



ภาพประกอบที่ 8.1 การค้นหาเชิงเส้นในลิสต์ไม่เรียงลำดับแบบพบเป้าหมาย

ที่มา : ดัดแปลงจาก Necaise (2011, p 126)

การค้นหา 8

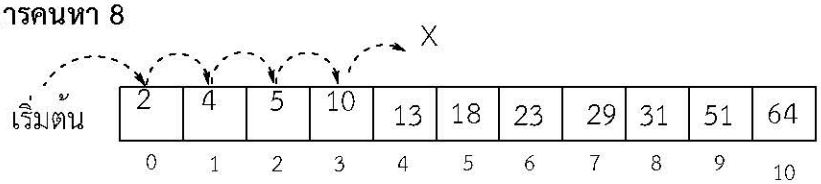


ภาพประกอบที่ 8.2 การค้นหาเชิงเส้นในลิสต์ไม่เรียงลำดับแบบไม่พบเป้าหมาย

ที่มา : ดัดแปลงจาก Necaise (2011, p 126)

การค้นหาเชิงเส้นในลิสต์เรียงมีหลักการเช่นเดียวกับลิสต์ไม่เรียง โดยจะยุติการค้นหาได้ เมื่อค่าไม่ได้อยู่ในลำดับที่จัดเรียง เช่น ต้องการค้นหา 8 ในลิสต์เรียงลำดับตัวเลขจากน้อยไปมาก ดังภาพประกอบที่ 8.3 เมื่อค้นหาข้อมูลในลำดับที่สี่ซึ่งมีค่า 10 สรุปได้ว่าค้นหาไม่พบและยุติการค้นหาได้ เพราะลำดับที่ 4 มีค่ามากกว่า 8 แสดงว่าลำดับถัดไปจะมีค่ามากขึ้นตามลำดับ

การค้นหา 8



ภาพประกอบที่ 8.3 การค้นหาเชิงเส้นในลิสต์ที่เรียงลำดับ

ที่มา : ดัดแปลงจาก Necaise (2011, p 127)

sample

ตัวอย่างที่ 8.1 อัลกอลิทึมสำหรับการค้นหาแบบเชิงเส้น

กำหนดให้ A เป็นลิสต์ มี n element ดังนี้ A[1], A[2], A[3],... A[n] ทำการค้นหาเป้าหมายในลิสต์ A ถ้าค้นหาข้อมูลในลิสต์ loc มีค่าเท่ากับตำแหน่งหรือดัชนี (index) ของข้อมูลที่พบ ถ้าค้นหาไม่พบข้อมูล loc มีค่าเท่ากับ -1 (Das, 2006, p. 207)

1. Input an array A of n elements and "data" to be searched and initialise loc = – 1.
2. Initialise i = 0; and repeat through step 3 if (i < n) by incrementing i by one .
3. If (data = A[i])
   (a) loc = i
   (b) GOTO step 4
4. If (loc > 0)
   (a) Display "data is found and searching is successful"
5. Else
   (a) Display "data is not found and searching is unsuccessful"
6. Exit

การวิเคราะห์ของการค้นหาแบบเรียงลำดับสามารถสรุปได้ดังนี้ :

จำนวนการเปรียบเทียบที่ต้องทำในกระบวนการค้นหาที่พบค่าเป้าหมายขึ้นอยู่กับตำแหน่งที่ค่านั้นถูกเก็บอยู่ หากค่าที่ต้องการค้นหาอยู่ที่ตำแหน่งที่ 1 คือสมาชิก m จะต้องใช้การเปรียบเทียบจำนวน 1 ครั้ง เพื่อค้นหาสมาชิก m (สมลักษณ์ จิตถาวร, 2550, หน้า 364 Mohanty และ Tripathy, 2021, p. 350 & Lambert, 2014, p. 61)

กรณีที่แย่ที่สุด: ถ้าค่าที่ต้องการค้นหาอยู่ที่ตำแหน่งสุดท้ายของลิสต์ จำนวนการเปรียบเทียบในการค้นหา T(n) = O(n) ซึ่งหมายถึงกรณีที่แย่ที่สุดมีเวลาเชิงเส้น

กรณีที่ดีที่สุด: เมื่อค่าที่ต้องการค้นหาอยู่ที่ตำแหน่งแรกในลิสต์ จำนวนการเปรียบเทียบในการค้นหา T(n) = O(1) ซึ่งหมายถึงกรณีที่ดีที่สุดมีเวลาคงที่

กรณีเฉลี่ย: ความซับซ้อนในกรณีเฉลี่ย คือ (n+n-1+n-2+...+1) / n หรือ (n+1)/2 (จำนวนการเปรียบเทียบในกรณีที่ดีที่สุด + จำนวนการเปรียบเทียบในกรณีที่แย่ที่สุด) / 2 คือ (n+1)/2 ผลลัพธ์นี้จะได้จำนวนวนการเปรียบเทียบในกรณีเฉลี่ยเท่ากับ T(n) = O(n) ซึ่งหมายถึงในกรณีเฉลี่ยต้องใช้เวลาเชิงเส้นสำหรับการค้นหาสำเร็จ

Sample