

Lab 09: Greedy Algorithms

6 พฤศจิกายน 2568

Huffman Coding

Huffman Coding เป็นเทคนิคการเข้ารหัสข้อมูลที่ใช้หลักการของการจัดเรียงข้อมูลแบบต้นไม้ (Tree) เพื่อสร้างรหัสที่มีความยาวแตกต่างกันสำหรับแต่ละสัญลักษณ์ โดยสัญลักษณ์ที่มีความถี่สูงจะได้มีรหัสที่สั้นกว่า ในขณะที่สัญลักษณ์ที่มีความถี่ต่ำจะได้รับรหัสที่ยาวกว่า เทคนิคนี้มักใช้ในการบีบอัดข้อมูล เช่น การบีบอัดไฟล์ภาพหรือเสียง ซึ่งเป็นแนวคิดแบบ Greedy Algorithm โดยรวมโหนดที่มีค่าน้ำหนักน้อยที่สุดสองโหนดเข้าด้วยกันซ้ำ ๆ จนเหลือต้นไม้เพียงต้นเดียว

ขั้นตอนหลักในการทำงานของ Huffman Coding มีดังนี้

1. สร้างตารางความถี่ (Frequency Table) สำหรับสัญลักษณ์ที่ต้องการเข้ารหัส
2. สร้างต้นไม้ Huffman โดยการรวมสัญลักษณ์ที่มีความถี่ต่ำสุดเข้าด้วยกันจนกว่าจะครบ
3. สร้างรหัสสำหรับแต่ละสัญลักษณ์โดยการทำ traversal ซึ่งกำหนดให้ทางซ้ายเป็น 0 และขวาเป็น 1

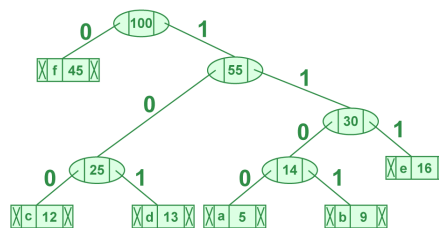


Figure 1: ตัวอย่างต้นไม้ Huffman และรหัสที่ได้

งานของนักศึกษา

เขียนโปรแกรมเพื่อสร้าง Huffman Tree จะต้องมียอดประกอบดังนี้

- Huffman Tree: โครงสร้างข้อมูลสำหรับเก็บ Huffman Tree
- buildTree(): ฟังก์ชันสำหรับสร้าง Huffman Tree
- encode(): ฟังก์ชันสำหรับเข้ารหัสข้อมูลโดยใช้ Huffman Tree ที่สร้างขึ้น
- decode(): ฟังก์ชันสำหรับถอดรหัสข้อมูลที่เข้ารหัสแล้วกลับเป็นข้อมูลต้นฉบับ

คำแนะนำ: ในการสร้าง Huffman Tree ควรใช้ Priority Queue เพื่อช่วยในการเลือกโหนดที่มีความถี่ต่ำสุดได้อย่างมีประสิทธิภาพ และหลังจากที่ tree ถูกสร้างขึ้นแล้วรหัสของแต่ละตัวอักษร สามารถหาได้จากการทำ preorder traversal ของ tree

Egyptian Fraction

ในยุคอียิปต์โบราณ เมื่อมนุษย์ยังไม่มีระบบตัวเลขทศนิยมเหมือนในปัจจุบัน นักคณิตศาสตร์ชาวอียิปต์ได้พัฒนาวิธีการเขียน “เศษส่วน” ที่แสดงถึงส่วนของสิ่งของหรือวัตถุ โดยใช้รูปแบบที่เข้าใจง่ายและเป็นระบบ นั่นคือ การเขียนเศษส่วนทุกตัวให้อยู่ในรูปของผลรวมของเศษส่วนที่มีตัวเศษเท่ากับ 1 รูปแบบนี้ถูกเรียกว่า “Egyptian Fraction” หรือ “เศษส่วนแบบอียิปต์” ซึ่งต่อมาถูกค้นพบหลักฐานในเอกสารทางคณิตศาสตร์เก่าแก่ที่เรียกว่า Rhind Mathematical Papyrus เอกสารนี้มีอายุมากกว่า 3,000 ปี และถูกมองว่าเป็นหนึ่งในรากฐานทางคณิตศาสตร์ที่เก่าแก่ที่สุดของโลก

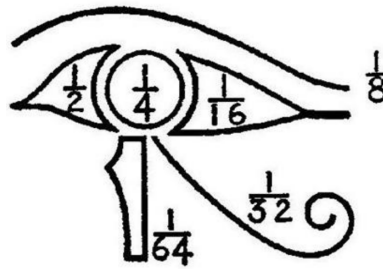


Figure 2: ตัวอย่าง Egyptian Fraction

หัวใจของแนวคิด Egyptian Fraction คือความเชื่อว่าทุกเศษส่วนสามารถเขียนได้เป็นผลรวมของเศษส่วนย่อย ๆ ที่มีตัวเศษเป็น 1 ทั้งหมด เช่น

$$\frac{2}{3} = \frac{1}{2} + \frac{1}{6}$$

$$\frac{3}{16} = \frac{1}{8} + \frac{1}{16}$$

โดยที่ Egyptian Fraction จะมีคุณสมบัติสำคัญดังนี้

- ทุกพจน์ของ Egyptian Fraction ต้องมี ตัวเศษ = 1
- ตัวส่วนทุกตัวต้องเป็น จำนวนเต็มบวกไม่ซ้ำกัน
- การเขียนรูปแบบนี้มักจะ ไม่ซ้ำกัน (แต่บางเศษส่วนอาจมีมากกว่าหนึ่งวิธีได้)

งานของนักศึกษา

การพัฒนาโปรแกรมที่สามารถแปลงเศษส่วนใด ๆ ให้อยู่ในรูปแบบ Egyptian Fraction ได้โดยหลักการของ Greedy Algorithm โดยมีองค์ประกอบดังนี้

- numerator: ตัวเศษของเศษส่วน
- denominator: ตัวส่วนของเศษส่วน
- toEgyptian(): ฟังก์ชันสำหรับแปลงเศษส่วนให้อยู่ในรูปแบบ Egyptian Fraction