

# Prim's Algorithm

Prim's Algorithm ถูกพัฒนาโดยนักคณิตศาสตร์ชื่อ Vojtech Jarnik ในปี 1930 ต่อมาถูกพัฒนาต่อโดยนักคอมพิวเตอร์ ชื่อ Robert C. Prim ในปี 1957 และ Edsger Dijkstra ในปี 1959 ดังนั้น อัลกอริทึมนี้ บางทีจึงมักเรียกว่า DJP Algorithm , Jarnik Algorithm หรือ Prim-Jarnik Algorithm

Prim Algorithm เป็นอัลกอริทึมที่ใช้ในการหาขนาด หรือน้ำหนักของต้นไม้ทอดข้ามที่น้อยที่สุด โดยมีวิธีการดังนี้

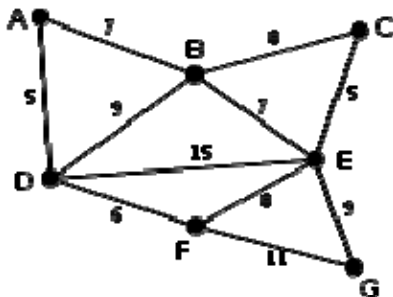
1. นำกราฟที่ถูกเชื่อมกันด้วย เส้น E เส้น และจุดยอด V จุด
2. โดย ให้  $V_{new} = \{a\}$  โดยที่ a เป็นจุดเริ่มต้น และ  $E_{new} = \{\}$
3. เลือกคู่ของจุดยอด (x,y) โดยที่ x เป็นสมาชิกของ  $V_{new}$  แต่ y ไม่เป็น และมีน้ำหนักของ Edge น้อยที่สุด (หากมีมากกว่า 1 อัน ให้เลือกมาหนึ่งอัน)
4. เก็บค่า y ใส่  $V_{new}$  และ (x,y) ใส่  $E_{new}$
5. ทำซ้ำขั้นตอนที่ 3-4 ไปเรื่อยๆจนกระทั่ง สมาชิกใน  $V_{new}$  มี V ตัว
6. จะได้  $V_{new}$  และ  $E_{new}$  ซึ่งเก็บค่าของต้นไม้ทอดข้ามน้อยที่สุด

## วิธีการที่ใช้

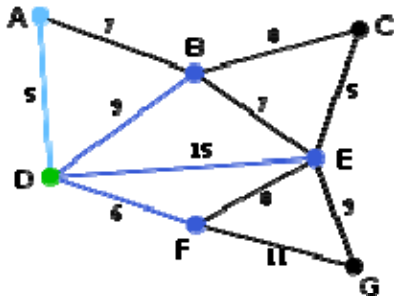
1. เมตริกซ์ประชิด + Searching  $O(V^2)$
2. Binary Heap + Adjacency list  $O((V + E) \log(V)) = O(E \log(V))$
3. Fibonacci Heap + Adjacency list  $O(E + V \log(V))$

## ตัวอย่าง

1. กราฟเริ่มต้นที่จะนำมาหาต้นไม้ทอดข้ามน้อยที่สุด



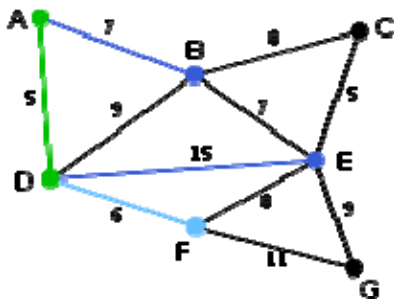
2. เลือก Vertex D เป็นจุดเริ่มต้น จะได้ว่า A , B , E , F เป็นจุดที่เชื่อมกับ D และ A มีระยะห่างน้อยที่สุด ดังนั้นจึงเลือกขอบ AD



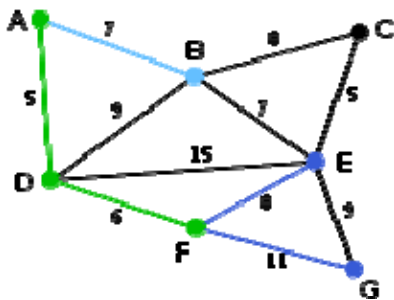
3. ต่อไปให้เลือกจุดยอดที่เชื่อมกับ A หรือ D ในที่นี้จะได้ว่า

- B ห่าง A 7
- B ห่าง D 9
- E ห่าง D 15
- F ห่าง D 6

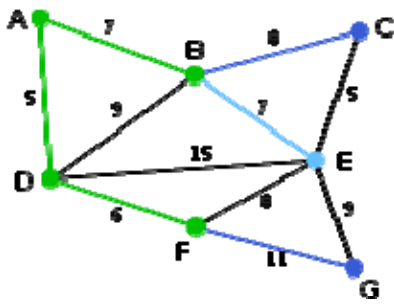
ดังนั้นเราจึงการเลือก DF



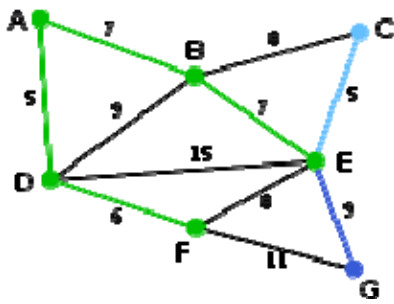
4. ในทำนองเดียวกันจะพบว่าต่อไป จะพบว่าจะมีจุดยอดที่เพิ่มขึ้น จากนั้นเราจะทำการเลือกจุดที่เชื่อมกันจุดยอดที่เลือกไปแล้ว และมีระยะห่างจากกันน้อยที่สุด จะได้ดังนี้



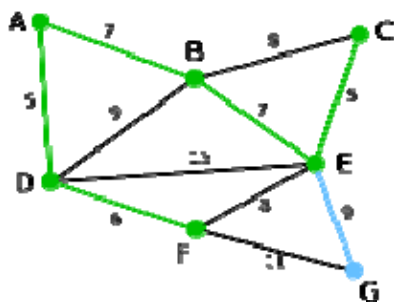
5.



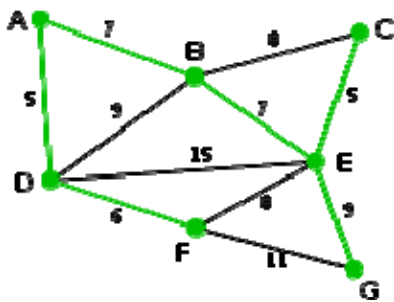
6.



7.



8. จะได้ต้นไม้ทอดข้ามที่น้อยที่สุด ซึ่งในที่นี้ใช้น้ำหนัก 39



อ้างอิง

[http://en.wikipedia.org/wiki/Prim\\_algorithm](http://en.wikipedia.org/wiki/Prim_algorithm)