

3. (3 คะแนน) ด้วยความรู้จากการบรรยายในคาบสุดท้ายโดยอาจารย์พีรพล จงแปลงข้อมูล training data ในตารางด้านซ้ายให้เป็น binary training data sets เมื่อกำหนดให้มีคลาสทั้งหมด 3 คลาส (เติมคำตอบลงในตารางด้านขวา)

ข้อมูล	คลาส
X1	C1
X2	C1, C2
X3	C3
X4	C1,C3

ข้อมูล	C1	C2	C3
X1			
X2			
X3			
X4			

4. (5 คะแนน) จงวาด state space tree สำหรับค้นคำตอบของปัญหา 3CNF-SAT กำหนดให้มีตัวแปร 4 ตัว เขียนกำกับด้วยการแตกกิ่งในแต่ละระดับมีความหมายอะไร และแสดงให้ดูว่าการใช้อัลกอริทึม depth-first search เพื่อค้นคำตอบกรณี 3CNF คือ $(x_1 + \bar{x}_2 + x_3)(x_2 + x_3 + \bar{x}_4)(\bar{x}_1 + \bar{x}_3 + \bar{x}_4)$ นั้นเป็นอย่างไร

5. (10 คะแนน) พิจารณาอัลกอริทึมของ Floyd-Warshall ข้างล่างนี้ (ให้สังเกตตัวแปร K ที่เพิ่มขึ้นมา) แล้วตอบคำถาม

```

FloydWarshall(w[1..n][1..n]) {
    K[1..n][1..n] = new integer array of size n x n, and initialized to all 0's
    D[1..n][1..n] = w[1..n][1..n] // copy array w to a new array D
    for (k0 = 1 to n)
        for (i = 1 to n)
            for (j = 1 to n)
                if (D[i][j] > D[i][k0] + D[k0][j]) {
                    D[i][j] = D[i][k0] + D[k0][j]
                    K[i][j] = k0
                }
    return K
}

```

- ก) หลังจากใช้อัลกอริทึมของ Floyd-Warshall เพื่อหา all-pair shortest paths ของกราฟ G เราจะรู้ได้อย่างไรว่า G มี negative-weight cycle หรือไม่ โดยพิจารณาจากเมทริกซ์ D ที่เก็บระยะทางของวิถีสั้นที่สุดต่าง ๆ

K	1	2	3	4	5	6
1	0	0	4	0	2	0
2	4	0	4	0	0	0
3	0	1	0	1	0	0
4	0	1	0	0	2	1
5	3	0	0	3	0	0
6	4	0	4	2	0	0

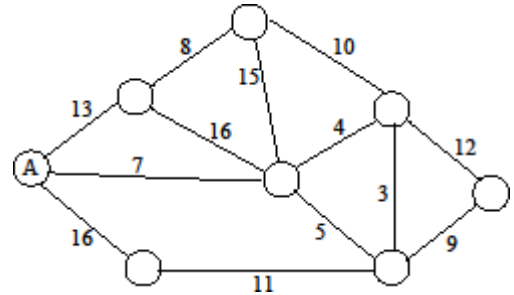
- ข) หลังจากใช้อัลกอริทึมของ Floyd-Warshall เพื่อหา all-pair shortest paths ของกราฟ G ได้อาเรย์สองมิติ K ดังรูปทางขวานี้

อยากทราบว่า วิถีสั้นสุดจากปม 5 ไป 4 และจากปม 6 ไป 1 ผ่านปมอะไรบ้าง

วิถีสั้นสุดจากปม 5 ไป 4 : _____

วิถีสั้นสุดจากปม 6 ไป 1 : _____

6. (5 คะแนน) พิจารณาว่าข้อความในแต่ละข้อข้างล่างนี้ จริงหรือเท็จ เขียนคำตอบไว้หน้าข้อ
- backtracking เป็นกลไกที่ใช้เลือก state ที่เหมาะสมในการเข้าสู่คำตอบได้รวดเร็วขึ้น
 - ปัจจุบันเป็นที่รู้กันแน่แล้วว่า ปัญหาที่อยู่ในกลุ่ม P ย่อมเป็นปัญหา NP อย่างแน่นอน
 - ปัจจุบันเป็นที่รู้กันแน่แล้วว่า มีปัญหาที่อยู่ใน NP บางปัญหาที่ไม่ได้อยู่ใน P
 - หากปัญหา A ลดรูปไปเป็นปัญหา B ได้ ย่อมสรุปได้ว่า ปัญหา A ไม่ยากไปกว่าปัญหา B
 - ปัญหา sort, fractional knapsack, minimum spanning tree, 1SAT ล้วนเป็นปัญหาที่หาคำตอบได้ในเวลาแบบ polynomial ทั้งสิ้น
7. (10 คะแนน) จากกราฟข้างบน จงระบุลำดับของเส้นที่อัลกอริทึมดังต่อไปนี้เลือกให้เข้ามา อยู่ใน Minimum Spanning Tree (ให้สังเกตว่าเส้นแต่ละเส้นนั้นมีน้ำหนักต่างกัน การระบุลำดับของเส้นนั้นให้ใช้ตัวเลขที่บอกน้ำหนัก)



ก) (5 คะแนน) Prim Algorithm โดยเริ่มพิจารณา จาก Node A

ข) (5 คะแนน) Kruskal Algorithm

***** ข้อ 8 – 11 ให้เลือกทำเพียง 3 ข้อจาก 4 ข้อ (ส่วนข้อ 12 ต้องทำ) โดยให้ระบุข้อที่ต้องการทำลงในช่องว่างด้านล่างนี้ ถ้าไม่ได้ระบุ หรือระบุมาเกิน 3 ข้อจะตรวจเฉพาะข้อ 8,9,10 เท่านั้น *****

ต้องการทำข้อ: _____.

8. (15 คะแนน) ในการต้อนวัวข้ามแม่น้ำสายหนึ่งนั้น เราจะต้องใช้แพข้ามฟาก โดยเราจะให้วัวไม่เกิน k ตัวขึ้นแพได้ในคราวเดียวกัน (แพสามารถรับน้ำหนักของวัวได้ไม่เกิน k ตัว) โดยแพจะใช้เวลาข้ามแม่น้ำต่อเที่ยว x นาที (หากข้ามไปกลับ จะใช้เวลา $2x$ นาที) โดยเรามีวัวทั้งหมด N ตัว และวัวตัวที่ i มาถึงท่านี้ เมื่อเวลา a_i (เป็นนาทีนับจากเวลาเริ่ม) โดยลำดับของวัวจะเรียงตามเวลาที่มาถึง ($a_1 < a_2 < a_3 < \dots < a_N$) เรามีแพเพียงแพเดียวเท่านั้น เป้าหมายของเรานั้นคือการหาว่าเวลาที่น้อยที่สุดในการพาวัวทั้งหมดข้ามแม่น้ำคือเท่าใด โดยให้ระบุเป็นเวลา (นาทีจากเวลาเริ่ม) ที่วัวที่มาถึงช้าที่สุดมาถึงฝั่งตรงข้าม คุณสามารถสมมุติได้ว่าวัวเดินขึ้นและลงจากแพเร็วมากๆ จึงไม่ต้องนำเวลาในส่วนนี้มาคิด

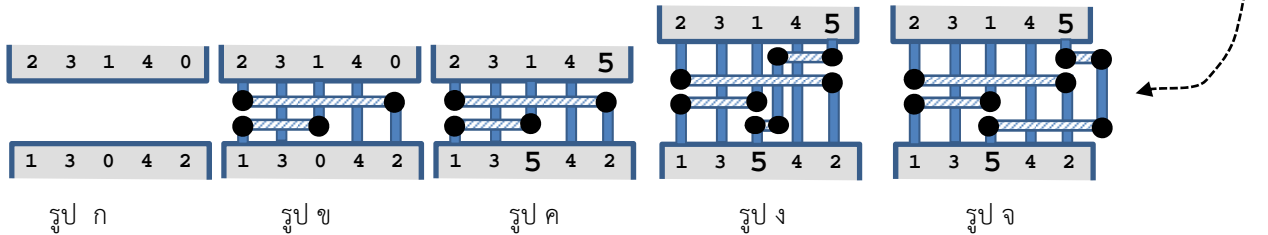
ตัวอย่าง

หากแพจุวัวได้ 2 ตัว และใช้เวลาข้ามแม่น้ำต่อเที่ยว 3 นาที และ วัวมาถึงนาทีที่ 4 5 14 15 วัวตัวสุดท้ายจะถึงอีกฝั่งเมื่อเวลา 18 นาที (พาวัวสองตัวแรกข้ามฟาก แพจะออกได้ที่เวลา 5 และจะกลับมาถึงเวลา $5+3+3 = 11$ จากนั้นรอจนวัวอีกสองตัวมาถึง และ ออกแพเมื่อเวลา 15 นาทีจะไปถึงอีกฝั่งเวลา $15+3 = 18$ นาที)

- (7 คะแนน) จงอธิบาย Greedy Algorithm เพื่อใช้แก้ปัญหานี้ได้คำตอบที่ดีที่สุด
- (5 คะแนน) จงเขียน Pseudo Code เพื่อแก้ปัญหา
- (3 คะแนน) จงบอกคำตอบเมื่อเราเจอข้อมูลดังต่อไปนี้
 - $k = 3, x = 2, N = 5, a = (1, 3, 5, 8, 10)$
 - $k = 4, x = 1, N = 6, a = (1, 4, 6, 7, 8, 9)$
 - $k = 2, x = 3, N = 7, a = (1, 2, 5, 8, 10, 11, 13)$

9. (15 คะแนน) จงออกแบบอัลกอริทึมแบบ Branch & Bound เพื่อแก้ปัญหา longest simple path ซึ่งเป็นดังนี้ ให้กราฟ $G = (V,E)$ เป็นกราฟแบบ undirected weighted graph โดยที่ $w(a,b)$ เป็นน้ำหนักของเส้นเชื่อมที่เชื่อมระหว่างปม a และปม b ($w(a,b)$ มีค่าเป็นบวกเสมอ) เราต้องการจะทราบความยาวของวิถีอย่างง่าย (simple path ซึ่งคือวิถีที่ไม่ใช่ปมซ้ำเลย) ที่ยาวที่สุดที่เริ่มต้นที่ปม S
- ก) (3 คะแนน) จงอธิบาย state space tree โดยสังเขปเพื่ออธิบายว่า state เป็นอย่างไร และมีการแตกกิ่งอย่างไร
- ข) (5 คะแนน) จงเขียน Pseudo Code เพื่อแก้ปัญหา
- ค) (7 คะแนน) จงอธิบาย heuristic function (bounding function) ที่เลือกใช้

10. (15 คะแนน) ในการออกแบบวงจรรวม (IC) มีปัญหาที่ย่อยปัญหาหนึ่งชื่อว่า channel routing ปัญหาที่มี input คือ ลำดับของหมายเลขสัญญาณที่ขอบล่างของกล่องวงจรด้านบน และลำดับของหมายเลขสัญญาณที่ขอบบนของกล่องวงจรด้านล่าง ดังตัวอย่างในรูป ก จากนั้นก็หาวิธีต่อสัญญาณที่มีหมายเลขเหมือนกันเข้าด้วยกัน (หมายเลข 0 ในรูปแทนการไม่มีสัญญาณ) ดังแสดงในรูป ข



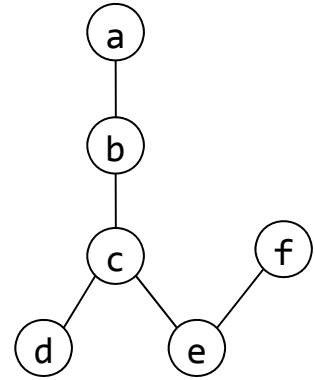
การต่อสัญญาณทำได้โดยใช้สายสัญญาณ โดยจะต่อได้ในแนวนอนและแนวตั้งเท่านั้น สายสัญญาณในแนวนอนและแนวตั้งจะสร้างด้วยวัสดุคนละสาร แนวนอนเป็นสาร A แนวตั้งเป็นสาร B สองสารนี้อยู่กันคนละชั้นของแผงวงจร สัญญาณที่เห็นทับกันในรูป ข จึงไม่ได้ต่อกัน ถ้าจะให้ต่อกันต้องมีหมุดเชื่อม (ในรูปแสดงด้วยวงกลมเข้ม) สิ่งที่ต้องระวังคือ มันมีลำดับของหมายเลขสัญญาณที่อาจสร้างปัญหา ไม่สามารถต่อสัญญาณได้ เช่น รูป ค คือรูป ข ที่เพิ่มสัญญาณหมายเลข 5 แต่เราต่อสัญญาณ 5 แบบที่แสดงในรูป ง ไม่ได้ เพราะเราให้วางสายแนวตั้งได้เฉพาะในแนวของเลขสัญญาณเท่านั้น หากทำแบบรูป ง สายมันก็ลัดกันเกินไป อาจลัดวงจร (ไม่อนุญาตให้เลื่อนตำแหน่งของเลขสัญญาณในแนวนอน) และก็ไม่อยากต่อสัญญาณ 5 แบบรูป จ เพราะกินเนื้อที่นอกขอบ

สิ่งที่ต้องการให้ทำคือ จงออกแบบอัลกอริทึมที่รับลำดับของหมายเลขสัญญาณทั้งด้านบนและด้านล่าง กำหนดให้มีจำนวนหมายเลขสัญญาณอยู่ N ตำแหน่ง และให้ $T[i]$ และ $B[i]$ ระบุถึงลำดับของหมายเลขสัญญาณ ณ ตำแหน่ง i ขอบล่างของกล่องวงจรด้านบนและที่ขอบบนของกล่องวงจรด้านล่างตามลำดับ แล้วตรวจสอบว่า เป็นลำดับสัญญาณที่อาจต่อสายสัญญาณไม่ได้ตามปัญหาที่น่าเสนอข้างต้นหรือไม่ เขียนอธิบายแนวคิดแนวทางการตรวจสอบ โดยอาศัยอัลกอริทึมที่ได้เรียนกันมาในวิชานี้ อธิบายเข้าใจในสิ่งที่เรียนมาแล้ว และก็ไม่จำเป็นต้องเขียน pseudo code ด้วย (ข้อแนะนำ : เราสามารถลดรูป (reduce) ปัญหาไปเป็นปัญหาหนึ่งทางกราฟที่ได้นำเสนอในชั้นเรียน)

11. (15 คะแนน) กำหนดให้มีกราฟ $G = (V,E)$ เป็นกราฟแบบ undirected weighted graph โดยที่ $w(a,b)$ เป็นน้ำหนักของเส้นเชื่อมที่เชื่อมระหว่างปม a และปม b ($w(a,b)$ มีค่าเป็นบวกเสมอ) และเราอยากทราบ Shortest Path จากปม S ไปยังปมอื่น ๆ ในกราฟนี้ โดยที่เราได้กระทำอัลกอริทึมของ Dijkstra บนกราฟนี้เรียบร้อยแล้ว และได้ผลลัพธ์มาเก็บไว้ในตัวแปร $D[*]$ และ $P[*]$ โดยที่ $D[v]$ นั้นเก็บระยะทางสั้นสุดจากปม S ไปยังปม v และ $P[v]$ นั้นเก็บปมที่บอกว่าวิถีสั้นสุดจากปม S มายังปม v นั้นมีเส้นเชื่อมสุดท้ายเป็น $(P[v],v)$ สมมติหลังจากนั้น น้ำหนักของเส้นเชื่อม 1 เส้นเชื่อมได้ “ลดลง” (กำหนดให้เส้นเชื่อม (q,r) มีน้ำหนักลดลง x) ซึ่งอาจจะทำให้วิถีสั้นสุดมีการเปลี่ยนแปลง จงออกแบบขั้นตอนวิธีสำหรับการคำนวณหาวิถีสั้นสุดจากปม S ไปยังปมอื่น ๆ ทั้งหมดใหม่ ควรใช้ข้อมูลใน D และ P ที่ทำได้ก่อนน้ำหนักของเส้น (q,r) จะลด ให้เป็นประโยชน์ ในการหา D และ P ใหม่หลังน้ำหนักของเส้น (q,r) ลดแล้ว พร้อมทั้งวิเคราะห์ประสิทธิภาพในการทำงาน (หมายเหตุ: ปัญหานี้แก้ได้ง่ายมาก เพียงแค่ทำอัลกอริทึมหาวิถีสั้นสุดใหม่ทั้งหมด ซึ่งจะเสียเวลามากเกินไป นิสิตจะต้องออกแบบขั้นตอนวิธีที่ทำงานได้ดีกว่าการหาวิถีสั้นสุดใหม่ทั้งหมด)

12. (7 คะแนน) กำหนดให้ปัญหา Set Cover เป็นดังต่อไปนี้ มีเซต $A = \{1,2,3,\dots,n\}$ และมีเซต s_1, s_2, \dots, s_m โดยที่ union ของ s_1, s_2, \dots, s_m นั้นเท่ากับ A เราอยากทราบว่าเป็นไปได้หรือไม่ที่จะเลือกเซตเป็นจำนวนไม่มากกว่า k เซตจาก s_1, s_2, \dots, s_m ที่ทำให้ union ของเซตที่เลือกนั้นเท่ากับ A ตัวอย่างเช่น ให้ $A = \{1,2,3,4,5\}$, $s_1 = \{1,2,3\}$, $s_2 = \{2,4\}$, $s_3 = \{3,4\}$, $s_4 = \{4,5\}$ และให้ $k = 2$ คำตอบคือเป็นไปได้ โดยเลือก s_1 และ s_4 เพราะ $s_1 \cup s_4 = A$

กำหนดให้ปัญหา “เติมน้ำหนัก” เป็นดังต่อไปนี้ กำหนดให้มีต้นไม้ $T = (V,E)$ ซึ่งเป็นต้นไม้แบบ undirected non-negative integer weighted tree (เส้นเชื่อมมีน้ำหนักเป็นจำนวนเต็มที่ไม่ติดลบ) และมีปมเป็นจำนวน $|V|$ ปม และกำหนดให้มีคู่ปม จำนวน p คู่ได้แก่ $(u_1, v_1), \dots, (u_p, v_p)$ และกำหนดให้ระยะทางของวิธีสั้นสุดจาก u_i ไปยัง v_i นั้นมีค่าเป็น $d(i)$ เราอยากทราบว่าเป็นไปได้หรือไม่ที่จะกำหนดน้ำหนักให้แก่แต่ละเส้นเชื่อมใน E โดยที่ผลรวมของน้ำหนักของทุกเส้นเชื่อมนั้นไม่มากกว่า L และทำให้ $d(i)$ นั้นมีค่าไม่น้อยกว่า J สำหรับทุก ๆ i ตัวอย่างเช่น กำหนดให้ T เป็นดังรูปด้านขวา โดยมี 6 ปม และให้มีคู่ปม $p = 3$ คู่คือ (a,d) , (b,e) และ (e,f) และให้ L มีค่าเป็น 4 และ J มีค่าเป็น 2 คำตอบคือเป็นไปได้ โดยกำหนดให้น้ำหนักของเส้นเชื่อม $b-c$ เป็น 2 และให้ เส้นเชื่อม $e-f$ เป็น 2 ส่วนเส้นเชื่อมอื่น ๆ มีน้ำหนักเป็น 0



จงแสดงให้เห็นว่าเราสามารถลดรูปปัญหา “เติมน้ำหนัก” ให้เป็นปัญหา Set Cover ได้

(หมายเหตุ: ข้อนี้เป็นข้อโบนัส สำหรับแต่ละคะแนนที่ได้จากข้อนี้ จะถูกนำไปเพิ่มให้กับคะแนนเต็มอีกครั้งหนึ่ง ตัวอย่างเช่น ถ้าข้อนี้ได้ 7 คะแนนเต็ม จะได้โบนัสอีก 7 คะแนนนำไปรวมกับคะแนนรวมทั้งหมดของข้อสอบชุดนี้)

ก) (3 คะแนน) จงอธิบายวิธีการแปลง instance ของปัญหา Set Cover เป็นปัญหา เติมน้ำหนัก

ข) (4 คะแนน) จงแสดงให้เห็นว่าวิธีการแปลงในข้อ ก) นั้นเป็นการลดรูปปัญหาที่ถูกต้อง

**** ในวิชานี้ อัตราส่วนของคะแนนสอบกลางภาค อยู่ที่ 40% และข้อสอบปลายภาคอยู่ที่ 40% จงปรับอัตราส่วนของคะแนนสอบกลางภาคตามที่นิสิตต้องการ โดยนิสิตสามารถเลือกปรับน้ำหนักของคะแนน สอบกลางภาค ได้ตั้งแต่ 30% ถึง 40% ค่าน้ำหนักในส่วนที่ขาดหายไปนั้นจะถูกนำไปคิดเป็นน้ำหนักของข้อสอบปลายภาคแทน จงระบุอัตราส่วนของคะแนนสอบกลางภาค ที่ต้องการ โดยใส่เป็นตัวเลขนับจำนวนเต็มตั้งแต่ 30 ถึง 40 ลงในช่องด้านขวานี้ (ถ้าไม่ระบุหรือระบุค่าที่ไม่ถูกต้องจะถือว่าระบุเลข 40 ไว้) (ตัวอย่างการคิดคะแนน ถ้ากรอกเลข 34 หมายความว่านิสิตต้องการให้สอบกลางภาคมีอัตราส่วนที่ 34% และสอบปลายภาคอยู่ที่ 46%) ****