แนวข้อสอบ วิชา Data Structure And Algorithm Analysis

1. Hashing

1.1 Hashing กำหนดให้ TableSize = 17 Rehash ที่ 85% ให้ทำการ insert 302,234,183,200, 230,331,187 แก้ปัญหาด้วย Double Hashing

ถ้าเต็ม 85% ขึ้นไป

$\frac{85 \times 17}{100}$ ← Table size

$\frac{x \times 100}{17} = 85$

$\frac{x}{17} \times 100 = 85$ $\frac{85 \times 17}{100}$

1.2 จากข้อ 1.1 หากใส่ครบแล้วจะต้องใส่อีกกี่ตัวจึงทำ Rehashing

8 ตัว

230	0
	1
	2
	3
200	4
	5
	6
	7
183	8
234	9
	10
	11
	12
302	13
	14
331	15
187	16

h_0 302 = 302 mod 17
= 13

h_0 234 = 234 mod 17
= 13 ชนครั้งที่ 1

F_1 = 1 × (13 − 234 mod 13)
= 13 − 0
= 13

h_1 234 = 26 % 17
= 17 + 9

h_0 230 = 230 mod 17
= 9 ชน 1

F_1 = 1 × (13 − 230 mod 13)
= 1 (13 − 9)
= 4

h_1 230 = (4 + 9) mod 17
= 13 ชน 2

F_2 = 2 × (13 − 230 mod 13)
= 2 × (13 − 9)
= 2 × (4)
= 8

h_2 230 = (8 + 9) mod 17
= (17)
= 0

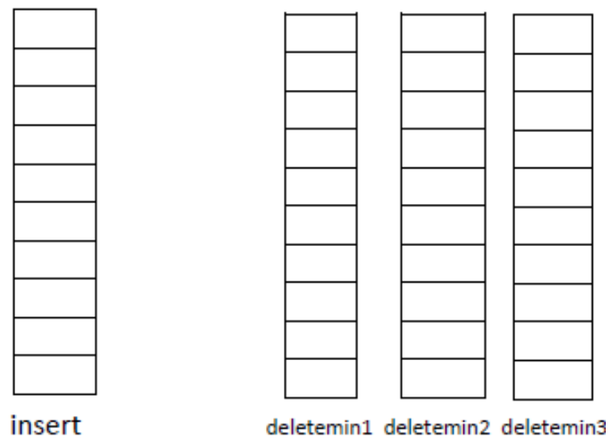
h_0 187 = 187 mod 17
= 0 ชน 1

F_1 = 1 × (13 − 187 mod 13)
= 1 × (8)
= 8

h_1 187 = (0 + 8) mod 17
= 8 ชน 2

F_2 = 2 × (13 − 187 mod 13)
= 2 × 8
= 16

h_2 187 = (0 + 16) mod 17
= 16



2. Binary Heaps มท 8

2.1 ให้นักศึกษาใส่ 64,19,20,29,41,25,30,80,52,90 ลงใน Binary Heap และผลลัพธ์สุดท้ายหลัง Insert แล้วเสร็จ แล้วเขียนผลลัพธ์

2.2 ให้ทำ DeleteMin 3 ครั้ง แล้วเขียนผลลัพธ์

3. Insertion Sort มท 9

3.1 กำหนดข้อมูล Input ทา Insertion Sort 10 ตัวดังนี้ 22,38,14,53,1,60,9,44,18,31 เขียนผลการทางานพร้อม Position Move

3.2 เมื่อ after p = 9 ให้อธิบายการทางาน ว่าแต่ละรอบทำอะไรบ้าง ค่า j มีค่าเท่าไรเมื่อจบโปรแกรม

4. มีตัวเลข 15 ตัว คือ 189,157,123,109,93,81,42,76,5,64,21,33,12,145,201 ให้ทำการ quick sort หาว่าเมื่อ pivot = 42 แล้ว array จะมีค่าอะไรบ้าง เมื่อทำการ quick sort เสร็จสิ้นแล้วทั้งหมด หาว่า pivot ทุกตัวมีค่าอะไรบ้าง

h_0 183 = 183 mod 17
= 13 ชนครั้งที่ 1

F_1 = 1 × (13 − 183 mod 13)
= 12

h_1 183 = (13 + 12) mod 17
= 8

h_0 200 = 200 mod 17
= 13 ชน 1

F_1 200 = 1 × (13 − 200 mod 13)
= 8

h_1 200 = (13 + 8) mod 17
= 4

h_0 331 = 331 mod 17
= 8 ชน 1

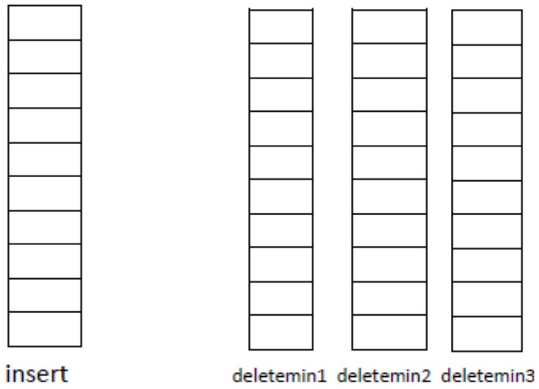
F_1 331 = 1 × (13 − 331 mod 13)
= 1 × (7)
= 7

h_1 331 = (8 + 7) mod 17
= 15

2. Binary Heaps มาก

2.1 ให้นักศึกษาใส่ 64,19,20,29,41,25,30,80,52,90 ลงใน Binary Heap และผลลัพธ์สุดท้ายหลัง Insert แล้วเสร็จ แล้วเขียนผลลัพธ์

2.2 ให้ทำ DeleteMin 3 ครั้ง แล้วเขียนผลลัพธ์



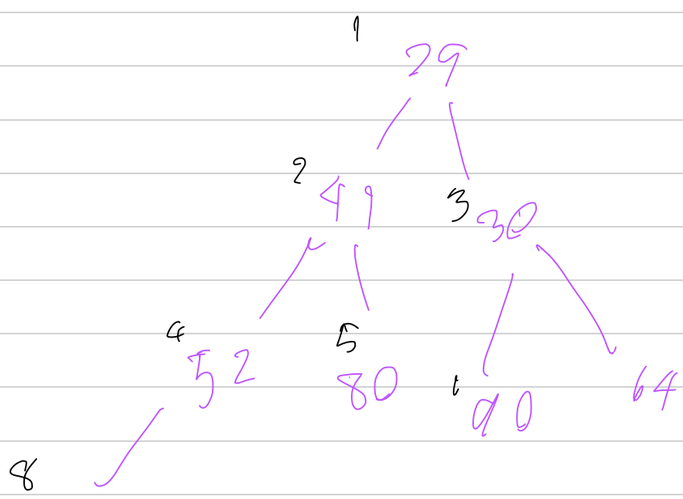
ผลลัพธ์

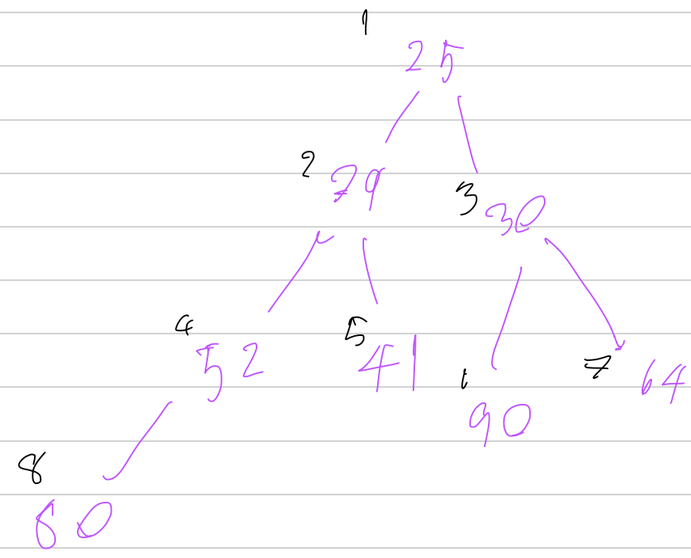
0	
1	19
2	29
3	20
4	52
5	41
6	25
7	30
8	80
9	64
10	90

insert

deletemin1	deletemin2	deletemin3
20	25	29
29	29	41
25	30	30
52	52	52
41	41	80
90	90	90
30	29	64
80	80	
64		

64, 19, 20, 29, 41, 25, 30, 80, 52, 90





3. Insertion Sort มทา

3.1 กำหนดข้อมูล Input ทา Insertion Sort 10 ตัวดังนี้ 22,38,14,53,1,60,9,44,18,31 เขียนผลการทางานพร้อม Position Move

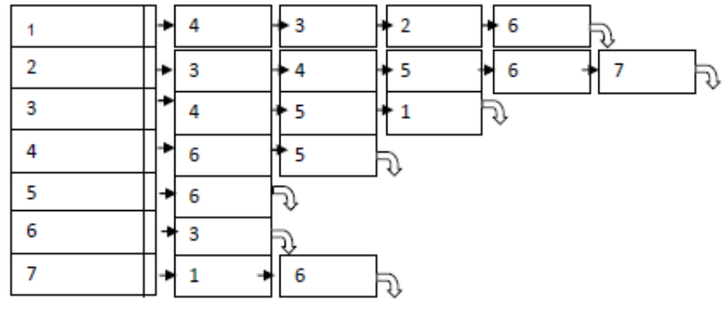
3.2 เมื่อ after p = 9 ให้อธิบายการทางาน ว่าแต่ละรอบทำอะไรบ้าง ค่า j มีค่าเท่าไรเมื่อจบโปรแกรม

```
for( int p = 1; p < a.size( ); p++ )
{
    Comparable tmp = a[ p ];

    int j;
    for( j = p; j > 0 && tmp < a[ j - 1 ]; j-- )
        a[ j ] = a[ j - 1 ];
    a[ j ] = tmp;
}
```

4. มีตัวเลข 15 ตัว คือ 189,157,123,109,93,81,42,76,5,64,21,33,12,145,201 ให้ทำการ quick sort หาว่าเมื่อ pivot = 42 แล้ว array จะมีค่าอะไรบ้าง เมื่อทำการ quick sort เสร็จสิ้นแล้วทั้งหมด หาว่า pivot ทุกตัวมีค่าอะไรบ้าง

มท 11

5. ให้ Vertex 4 เป็นจุดเริ่มต้น ทำ Unweight shortest-path แล้วหาค่า front,back เมื่อจบการทำงาน



คิวที่ให้สำหรับการใส่ลำดับ

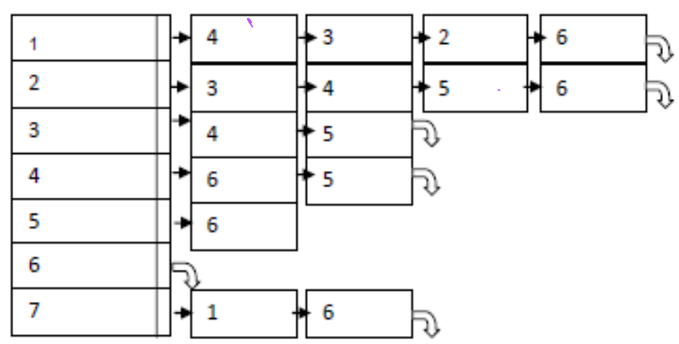
0	1	2	3	4	5	6

```
/*Pseudocode for unweighted shortest-path
  algorithm USE this Code*/
 void Graph::unweighted( Vertex s)
{
   Queue q (NUM_VERTICES);
   Vertex v, w;
   q.enqueue( s );
    s.dist = 0;

   while ( !q.isEmpty()เท็จ )
   {
       v = q.dequeue ( );
       v.known = true;

       for each w adjacent to v
              if (w.dist == INFINITY)
              {
                     w.dist = v.dist + 1;
                     w.path = v;
                     q.enqueue (w);
              }
   }
}
```

7

มท.10

6. ทำ topological sort หาค่า front,back เมื่อจบการทำงาน และเขียนลำดับของการ sort ลงใน queue ที่ให้มา



	1	2	3	4	5	6	7
v_1	1	0	0	0	0	0	0
v_2	1	1	0	0	0	0	0
v_3	2	2	1	0	0	0	0
v_4	3	3	2	1	0	0	0
v_5	3	3	3	2	1	0	0
v_6	5	4	3	2	2	1	0
v_7	0	0	0	0	0	0	0
	v_7	v_1	v_2	v_3	v_4	v_5	v_6
	v_7	v_1	v_2	v_3	v_4	v_5	v_6

คิวที่ให้สำหรับการใส่ลำดับ

7	1	2	3	4	5	6
0	1	2	3	4	5	6

ลำดับตามบัญชีรายชื่อ .....

คณะเทคโนโลยีและการจัดการอุตสาหกรรม

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

การสอบปลายภาคเรียนที่ 1 **ปีการศึกษา 2564**

รหัสและชื่อวิชา 060243106 Data Structure and Algorithm **ตอนที่ 1 , 2**

060223119 Data Structure and Algorithm **ตอนที่ 1**

สอบวันที่ 8 พฤศจิกายน พ.ศ. 2564 **เวลา 13.00-16.00 น.**

ผู้ออกข้อสอบ อ.ดร.ประดิษฐ์ พิทักษ์เสถียรกุล

รหัสนักศึกษา _____ ชื่อ-นามสกุล_____ ตอนที่

ทุจริตในการสอบ เป็นความผิดวินัยขั้นร้ายแรง
ให้ปรับตกวิชานั้น ไม่พิจารณาผลวิชาอื่น และ พักการศึกษาภาคถัดไป

คำสั่ง :

1.ข้อสอบมีทั้งหมด 7 ข้อ 5 หน้า ให้ทำทุกข้อ คะแนน 80 คะแนน เขียนคำตอบด้วยลายมือลงบนกระดาษ และ ถ่ายรูปส่งเป็นไฟล์ pdf เท่านั้น ให้เขียนเลขข้อที่ทำอย่างชัดเจนบนกระดาษ ไม่จำเป็นต้องลอกโจทย์ ให้เขียนเลขข้ออย่างชัดเจน

2.**อนุญาตให้เปิดตำราได้ทุกชนิด แต่อ้างอิงโปรแกรมตามที่สอน เป็นเกณฑ์ในการวัดผล**

3.เขียนชื่อ-นามสกุล รหัสประจำตัว ลงบนกระดาษที่ส่ง **ทุกหน้า**
ไม่ตรวจกระดาษที่ไม่มีชื่อ-นามสกุล

1	
2	
3	
4	
5	
6	
7	
รวม	

ข้อสอบฉบับนี้ผ่านการประเมิน
จากกรรมการภาควิชา <u>เทคโนโลยีสารสนเทศ</u>
ลงชื่อ.....
หัวหน้าภาควิชา
...../...../.....

ชื่อ-นามสกุล ..... รหัสประจำตัว .....

1. ให้นักศึกษาเขียนตารางอะเรย์ ตามโครงสร้าง Hash Table (ใน 1 ช่อง มี element กับ info)
กำหนด **TableSize ให้เป็น 19** คิดเปอร์เซ็นต์ของการทำ Rehashing เมื่อ Table ใกล้จะเต็ม ที่ **80%**

1.1. ให้นักศึกษาทำการ insert ข้อมูลจำนวน 8 ตัวต่อไปนี้ เข้าไปใน Hash Table ดังกล่าว ชนิด Open Addressing โดย**ใช้วิธีการแก้การชนกันแบบ Double Hashing กำหนด f(i) = i * (R – (x mod R))**

26 , 45 , 70 , 85 , 102 , 123 , 143 , 155 [รวม insert ทุกตัวเสร็จสิ้น 8 คะแนน]

โดยที่

1) ไม่จำเป็นต้องไล่โปรแกรมเสมอไป นักศึกษาสามารถคำนวณตามสูตรก็ได้

2) การชนกันและแก้การชนกันในแต่ละครั้ง นับเป็น 1 ไม่ได้นับเป็น 2 ได้อธิบายไว้ในห้องเรียนแล้ว

1.2. ข้อมูลตัวใด ขณะใส่เข้าไปใน Hash Table แล้ว มีการชนกันและแก้การชนกัน มากที่สุด (มีมากกว่า 1 ตัวก็ได้) และ ขณะใส่ข้อมูลตัวนั้น เกิดการชนกันกี่ครั้ง ที่ตำแหน่งใด

[3 คะแนน 1 + 1 + 1]

[ข้อมูลตัวใดมีการชนกันมากที่สุด ? + เกิดการชนกันกี่ครั้ง? + ที่ตำแหน่งใด?]

1.3. หลังจากใส่ข้อมูลลงใน Hash Table ดังกล่าว**ครบทุกตัว** ถามว่า มีการชนกันและแก้การชนกัน**รวมทั้งสิ้น**กี่ครั้ง (การชนกันและแก้การชนกัน นับเป็น 1 ไม่ได้นับเป็น 2 ได้อธิบายไว้ในห้องเรียนแล้ว) [3 คะแนน]

1.4. หลังจากใส่ข้อมูลลงใน Hash Table ดังกล่าว**ครบทุกตัว** ถามว่า จะต้องใส่ข้อมูลเพิ่มอีกกี่ตัว จึงจะเกิดการ Rehash [4 คะแนน]

2. ให้นักศึกษาเขียนตารางอะเรย์ ตามโครงสร้าง Binary Heap เป็นแนวตั้ง 4 ตาราง เพื่อรองรับข้อมูลตาม 2.1

2.1. จงใส่ข้อมูลตัวเลขต่อไปนี้ 64, 14, 24, 1, 20, 2, 5, 10, 6, 8, 15, 20 เข้าไปในตารางอะเรย์ ดังกล่าว โดยให้เขียนเฉพาะผลลัพธ์สุดท้ายที่ Insert ข้อมูลเสร็จแล้วเท่านั้น ในตารางแรก [1 คะแนน]

2.2. ให้ทำการ **deleteMin 1 ครั้ง** แล้วเขียนผลลัพธ์ลงในตารางอะเรย์ ที่ 2 [2 คะแนน]

2.3. หลังจากนั้นให้ทำการ insert ข้อมูลตัวเลข 3 เข้าไปใน เขียนผลในตารางอะเรย์ ที่ 3 [2 คะแนน]

2.4. หลังจากนั้น ให้ทำการ **deleteMin อีกครั้ง** แล้วเขียนผลลัพธ์ลงในตารางอะเรย์ ที่ 4 [3 คะแนน]

เขียนหัวตารางด้วย ว่า 2.1 2.2 deleteMin 2.3 Insert 2.4 **deleteMin**

การดำเนินการนั้น ขอให้ใช้โปรแกรมตามที่เรียน เพราะหากใส่ตามความรู้สึกแล้ว อาจจะได้ผลลัพธ์ผิดพลาดได้ การตรวจยึดผลลัพธ์จากการทำงานของโปรแกรมเป็นหลัก

1.1. ให้นักศึกษาทำการ insert ข้อมูลจำนวน 8 ตัวต่อไปนี้ เข้าไปใน Hash Table ดังกล่าว ชนิด Open Addressing โดยใช้วิธีการแก้การชนกันแบบ **Double Hashing** กำหนด **f(i) = i * (R – (x mod R))**

26 , 45 , 70 , 85 , 102 , 123 , 143 , 155 [รวม insert ทุกตัวเสร็จสิ้น 8 คะแนน]

โดยที่

$h_0\ 26 = 26 \bmod 19$
$= 7$

$h_0\ 45 = 45 \bmod 19$
$= 7$ ชน1

$F_1 = 1 \times (17 - 45 \bmod 17)$
$= 1 \times (17 - 11)$
$= 1 \times 6$

$h_1\ 45 = (6 + 7)$
$= 13$

$h_0\ 85 = 85 \bmod 19$
$= 9$ ชน1

$F_1 = 1 \times (17 - 85 \bmod 17)$
$= 17 - 0$
$= 17$

$h_1\ 85 = (9 + 17) \bmod 19$
$= 7$ ชน2

$F_2 = 2 \times (17 - 85 \bmod 17)$
$= 34$

$h_2\ 85 = (9 + 34) \bmod 19$
$= 5$ (43)

$h_0\ 70 = 70 \bmod 19$
$= 13$ ชน1

$F_1 = 1 \times (17 - 70 \bmod 17)$
$= 1 \times (17 - 2)$
$= 15$

$h_1\ 70 = (13 + 15) \bmod 19$
$= 9$

$h_0\ 102 = 102 \bmod 19$
$= 7$ ชน1

$F_1 = 1 \times (17 - 102 \bmod 17)$
$= 1 \times (17 - 0)$
$= 17$

$h_1\ 102 = (17 + 7) \bmod 19$
$= 5$ ชน2

$F_2 = 2 \times (17 - 102 \bmod 17)$
$= 2 \times 17$
$= 34$ (38)

$h_2\ 102 = (34 + 7) \bmod 19$
$= 3$

0	
1	
2	
3	102
4	
5	85
6	
7	26
8	
9	70
10	44
11	
12	
13	45
14	
15	
16	123
17	
18	155

$h_0\,123 = 123 \bmod 19$

$= 9$ 8 6 1

$F_1 = 1 \times (17 - 123 \bmod 17)$

$= 13$

$h_1\,123 = (9 + 13) \bmod 19$

$= 3$ 8 6 2

$F_2 = 2 \times (17 - 123 \bmod 17)$

$= 26$

$h_2\,123 = (h + 26) \bmod 19$

$=$

$h_0\,143 = 143 \bmod 19$

$= 19$

$h_0\,155 = 155 \bmod 19$

$= 3$ 8 6 1

$F_1 = 1 \times (17 - 155 \bmod 17)$

$= 1 \times (17 - 2)$

$= 15$

$h_1\,155 = (19 + 3) \bmod 19$

$= 16$

$\underline{80 \times 19}$ ← Table size

100

↑

80 %

mod 15

3. ให้นักศึกษาเขียนตารางอะเรย์ ตามโครงสร้าง vector สำหรับการทำ insertionSort 2 ตาราง สำหรับข้อ 3.1 และ 3.2 ขนาดของตาราง ขอให้ดูข้อมูลที่เป็น input ในแต่ละข้อเป็นหลัก

3.1. ให้กำหนดข้อมูลเป็น input เริ่มต้นสำหรับการทำ insertionSort จำนวน **12** ตัว โดยเป็นข้อมูลแบบ Worst Case โดยเขียนตัวเลขข้อมูล input เริ่มต้นลงในตารางอะเรย์แรก ที่เตรียมไว้ เมื่อทำ insertionSort เสร็จสิ้นแล้ว ถามว่า เกิด position move รวมทั้งสิ้น กี่ครั้ง [4 คะแนน]

3.2. กำหนดข้อมูลเป็น input เริ่มต้นสำหรับการทำ insertionSort จำนวน **12** ตัว ดังนี้
61 , 52 , 12 , 24 , 8 , 35 , 32 , 5 , 71 ,68 ,18 , 2 ใส่ในตารางอะเรย์ ตารางที่ 2 เป็น ข้อมูลตั้งต้น หลังจากนั้น ให้เขียนผลของการทำงานในรอบที่ p = 3 และ p = 6 พร้อมนับ Position Move ของทั้ง 2 รอบนี้ ใส่ในตาราง ขอให้เขียนตัวเลขตรงกันในแนวคอลัมน์ [6 คะแนน]

4. กำหนดข้อมูลเป็น input ให้กับฟังก์ชัน quicksort จำนวน 17 ตัว ดังนี้ **เขียนตัวเลขผิด ถือว่าได้ 0 คะแนน**

81, 198, 157, 145 , 123, 109, 93, 5, 76, 64, 42, 33, 21, 12, 201, 57, 89

สมมติข้อมูลอยู่ในตัวแปร Object a ตามโครงสร้างคลาส vector **กำหนดตัวเลข Cutoff เท่ากับ** 3 เมื่อทำงานฟังก์ชัน quicksort **โดยทำการเรียงข้อมูลจากน้อยไปมาก**

4.1. จงเขียนข้อมูลในตัวแปร Object a ทั้ง 17 ตัว **หลังจากใช้ pivot ตัวที่ 3 ทำงานเสร็จแล้ว**

ตอบ **pivot ตัวที่ 3** คือ ..... [7 คะแนน 5 + 2]

[0] เขียนข้อมูลในตัวแปร Object a ให้ครบทั้ง 17 ช่อง ถ้ามีการทำงาน insertionsort ต้องทำ insertionsort ให้เสร็จก่อนเขียน

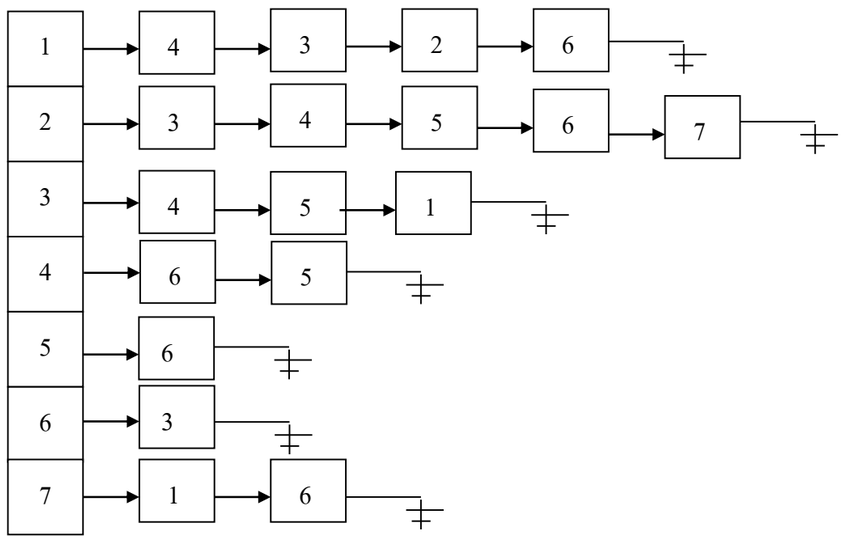
4.2. จากข้อ 4.1. ข้างต้น ขณะที่ pivot **ตัวที่ 1** ทำงานอยู่ ถามว่า**ค่าในอะเรย์**ตำแหน่งที่ i ชี้อยู่ หลังจากทำขั้นตอน partition เสร็จแล้ว ก่อนจะไปทำบรรทัดที่ /*11*/ และ /*12*/ ต่อ **มีค่าเท่าไร** [3 คะแนน]

ตอบ **ค่าในอะเรย์**ตำแหน่งที่ i ชี้อยู่ หลังจากทำขั้นตอน partition เสร็จแล้วมีค่า .....

4.3. เมื่อโปรแกรมทำงานฟังก์ชัน quicksort ในการเรียงข้อมูลทั้ง 17 ตัวเสร็จสิ้น ถามว่า มีการทำงานฟังก์ชัน insertionsort รวมทั้งสิ้น กี่ครั้ง [4 คะแนน]

ตอบ มีการทำงานฟังก์ชัน insertionsort รวมทั้งสิ้น คือ .....

5. ให้นักศึกษาเขียนตารางอะเรย์ สำหรับประมวลผล Shortest Path โดย 1 ช่องของอะเรย์ ประกอบด้วยฟิลด์ Known, Dist (DV) และ Path (Pv) และ ตารางอะเรย์ตามโครงสร้าง Class Queue ตามจำนวน vertex กำหนดข้อมูลของกราฟชนิดหนึ่ง เป็น Adjacency List มี 7 Vertex ดังภาพ



5.1. จงหา Shortest Path แบบ **Unweighted** โดยมี **Vertex 1 เป็นจุดเริ่มต้น ต้องการให้นักศึกษาเขียน** ตารางอะเรย์ผลลัพธ์สุดท้าย **ที่แสดง Known, Dv และ Pv** ทำทุก vertex โดย Known เป็น True หรือ 1 ทั้งหมดแล้ว [5 คะแนน]

5.2. จงเขียนภาพของ Queue เพื่อ**แสดงข้อมูลใน Queue** หลังจากทำงานข้อ 5.1. เสร็จ โดยให้เขียนตำแหน่งของ front และ back ด้วย ว่าชี้อยู่ที่ใด [2 คะแนน]

front อยู่ที่ตำแหน่ง .....

back อยู่ที่ตำแหน่ง .....

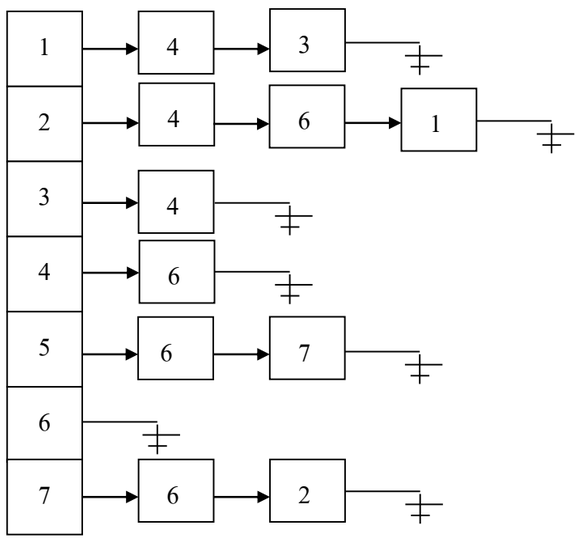
การนำข้อมูลเข้าออกจาก Queue ยึดตามฟังก์ชัน enqueue และ dequeue ในหนังสือที่ใช้เรียน บทที่ 3 เป็นหลัก

	Known	QV	Path
A	T	3	D
B	T	0	D
C	T	1	B
D	T	2	C
E	T	1	B
F	T	2	E
G	T	1	B

6	A	
5	F	
4	D	*
3	G	
2	E	*
1	C	*
0	B	*

ชื่อ-นามสกุล ..... รหัสประจำตัว .....

6. ให้นักศึกษาเขียนตารางอะเรย์ ชื่อ Indegree Array สำหรับประมวลผล Topological Sort กำหนดข้อมูลของกราฟ DAG อันหนึ่ง เป็น Adjacency List มี 7 Vertex ดังภาพ



6.1. จงแสดงวิธีหา Topological Sort **ต้องการให้นักศึกษาเขียนผลลัพธ์สุดท้ายของตาราง Indegree Array**

[5 คะแนน]

6.2. กำหนด Queue เพื่อใช้ประกอบ Algorithm ในการหา Topological Sort ขนาดของ Queue เป็น 7 ช่อง จงเขียนภาพของ Queue เพื่อ**แสดงข้อมูลใน Queue** หลังจากทำงานข้อ 6.1. เสร็จ โดยให้เขียนตำแหน่งของ front และ back ด้วย ว่าชี้อยู่ที่ใด [2 คะแนน]

front อยู่ที่ตำแหน่ง .....

back อยู่ที่ตำแหน่ง .....

การนำข้อมูลเข้าออกจาก Queue ยึดตามฟังก์ชัน enqueue และ dequeue ในหนังสือที่ใช้เรียน บทที่ 3 เป็นหลัก

6.3. เมื่อทำการหา Topological Sort เสร็จแล้ว จงเขียน Topological Ordering ที่เป็นคำตอบของกราฟนี้

ตอบ

..... [3 คะแนน]

ชื่อ-นามสกุล ..... รหัสประจำตัว .....

7. จงตอบคำถามต่อไปนี้

7.1. คุณสมบัติของ hash function ตามทฤษฎีในทางอุดมคติ มี 3 ข้อ คืออะไร

ตอบ [3 คะแนน]
1……..
2……..
3……..
ข้อใดเป็นไปไม่ได้ ในโลกแห่งความเป็นจริง .....
ด้วยเหตุผล 2 ประการ คือ 1.....
2.....

7.2. จงเขียนประโยคคำสั่ง (Statement) เพื่อใช้แทนฟังก์ชัน HashEntry ใน member function insert ของ Class HashTable โดยใน member function insert มีการเรียกใช้งานฟังก์ชัน HashEntry ดังนี้

```
array[ currentPos ] = HashEntry( x, ACTIVE );
```

ตอบ [2 คะแนน]
……..
……..

7.3. ในการทำ Quick Sort ขั้นตอนการทำ Partitioning แต่ละรอบนั้น พบว่า มีข้อมูลบางตัวที่ไม่ได้นำไปทำ Partition ด้วย ถามว่า มีข้อมูลตัวใดใน array ที่ไม่ได้ถูกทำ Partition เพราะเหตุผลใด

ตอบ [2 คะแนน]
……..
……..

7.4. Complete Binary Tree ที่มีความสูงเท่ากับ 16 จะมีจำนวน node อย่างน้อยที่สุด และ อย่างมากที่สุด จำนวนเท่าไร ตอบเป็นตัวเลขจำนวนเต็มของจำนวน node เท่านั้น **ไม่ติดอยู่ในรูปของเลขยกกำลัง**

(คำตอบที่ติดในรูปของเลขยกกำลัง จะได้ 0 คะแนน)

ตอบ…… [3 คะแนน]

7.5. Complete Graph ที่มีจำนวน vertex เท่ากับ 16 จะมีจำนวน edge ทั้งหมดจำนวนเท่าไร ตอบเป็นตัวเลขจำนวนเต็มของจำนวน edge เท่านั้น **ไม่ติดอยู่ในรูปของสมการ**

(คำตอบที่ติดในรูปของสมการ จะได้ 0 คะแนน)

ตอบ…… [3 คะแนน]

ขอให้นักศึกษาโชคดีในการสอบ

7. ให้นักศึกษาตอบคำถามต่อไปนี้

7.1 ให้บอกคุณสมบัติ 3 ประการของการ Hashing

7.2 จากข้อ 7.1 ข้อใดเป็นไปไม่ได้ในทางปฏิบัติ เนื่องจาก เหตุใด 2 ประการ

7.3 Median-of-three ทำให้ Quick Sort เร็วขึ้น 5% เพราะเหตุใด

7.4 Binary tree สูง 10 ชั้นจะมีโหนดอย่างน้อยและอย่างมากเท่าใด

****