學號：110060007
姓名：黃俊穎

EE2410 Data Structure Hw #2 (Chapter 5~8)
due date 6/13/2023, 23:59

Format: Use a text editor to type your answers to the homework problem. You need to submit your HW in a PDF file or a DOCX file named as **Hw2-SNo.docx** or **Hw2-SNo.pdf**, where SNo is your student number. Submit the **Hw2-SNo.docx or Hw2-SNo.pdf** file via eLearn. Inside the file, you need to put the **header and your student number, name (e.g., EE2410 Data Structure Hw #2 (Chapter 5~8 of textbook) due date 6/16/2023 by SNo, name)** first, and then the **problem** itself followed by your **answer** to that problem, one by one. The grading will be based on the correctness of your answers to the problems, and the **format**. Fail to comply with the aforementioned format (file name, header, problem, answer, problem, answer,…), will certainly degrade your score. If you have any questions, please feel free to ask me.

Trees:

1. (2%) What is the maximum number of nodes in a k-ary tree of height h? Prove your answer.

   Answer:

   For a k-ary tree of height 1, there's only one node.

   As the height becomes 2, there are one node as the root and k nodes as leaves, so there are k+1 leaves totally.

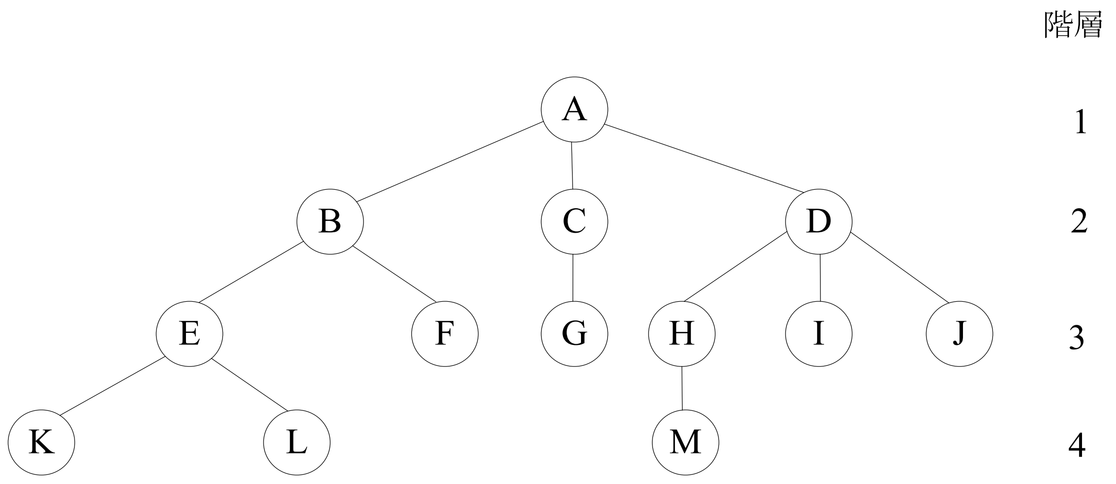
   As the height becomes 3, there are one node as the root, k nodes as middle tree and k^2 leaves totally, so there are k^2+k+1 leaves totally.

   Observe this law, we can derive the equation: For a k-ary tree of height h, the number of nodes are:

$$\sum_{n=0}^{h-1} k^n = 1 + k + k^2 + k^3 + \cdots + k^{h-1} = \frac{1(k^h - 1)}{k-1} = \frac{(k^h - 1)}{k-1}$$

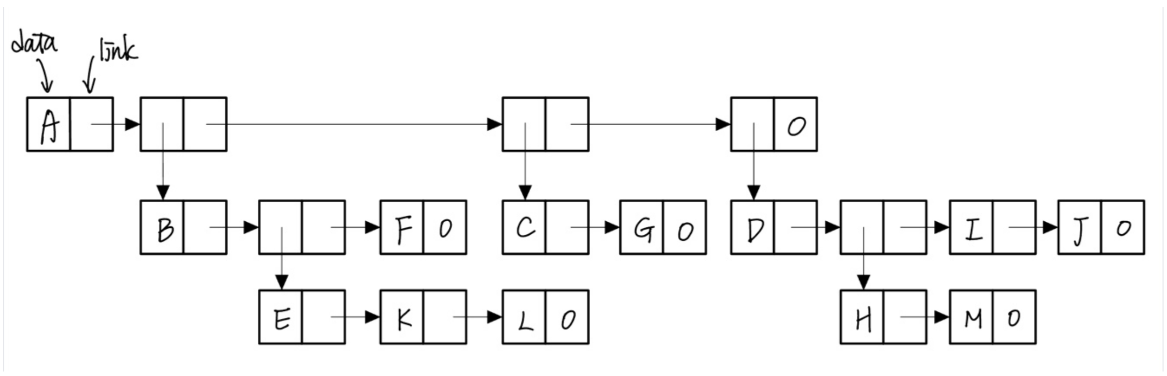
2. (8%) For a simple tree shown below,
   (a) Draw a list representation of this tree using a node structure with three fields: tag, data/down, and next.
   (b) Write down a generalized list expression form for this tree.
   (c) Convert the tree into a left-child and right-sibling tree representation
   (d) Draw a corresponding binary tree for this tree based on (c).
   (e) What is the depth of node L? What is the height of node B? What is the height of the tree?
   (f) Write out the preorder traversal of this tree.
   (g) Write out the postorder traversal of this tree.
   (h) Write out the level order traversal of this tree.

學號：110060007
姓名：黃俊穎



Answer:

(a) List representation of this tree:



(b) Generalized list expression form for this tree:

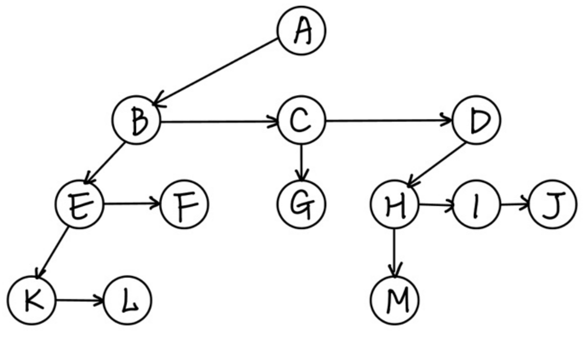
$$\xrightarrow{level\ 1} (A(\ldots))$$

$$\xrightarrow{level\ 2} (A(B(\ldots),C(\ldots),D(\ldots)))$$

$$\xrightarrow{level\ 3} (A(B(E(\ldots),F),C(G),D(H(\ldots),I,J)))$$

$$\xrightarrow{level\ 4} (A(B(E(K,L),F),C(G),D(H(M),I,J)))$$

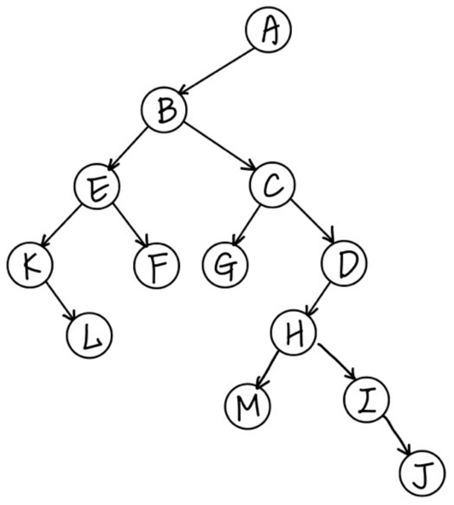
(c) Left-child and right-sibling tree representation:



學號：110060007
姓名：黃俊穎

(d) Binary tree for this tree based on (c).

By doing this, we put children at left trees and put siblings at right trees.



(e) What is the depth of node L? What is the height of node B? What is the height of the tree?

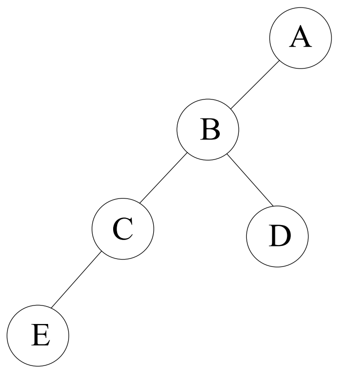
The depth of node L: 4. The height of node B: 3. The height of the tree: 4.

(f) The preorder traversal of this tree: A B E K L F C G D H M I J

(g) The postorder traversal of this tree: K L E F B G C M H I J D A

(h) The level order traversal of this tree: A B C D E F G H I J K L M

3. (4%) Draw the internal memory representation of the binary tree below using (a) sequential and (b) linked representations.

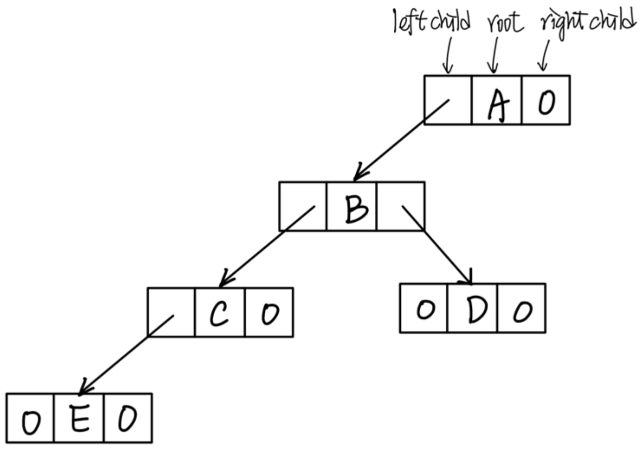


Answer:

(a) Sequential representations:

Array	a[0]	a[1]	a[2]	a[3]	a[4]	a[5]	a[6]	a[7]	a[8]	a[9]
Element	x	A	B	x	C	D	x	x	E	x

(b) Linked representations:



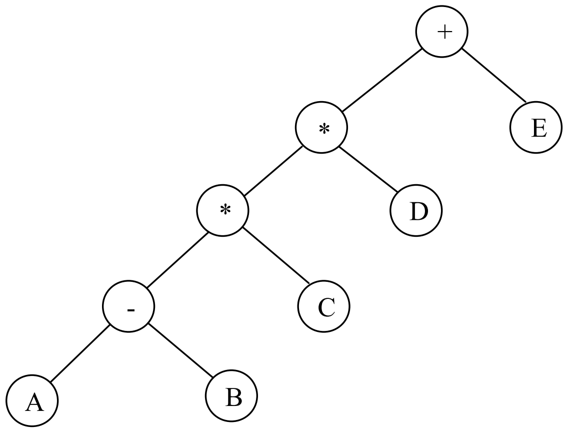
4. (2%) Extend the array representation of a complete binary tree to the case of complete trees whose degree is d, d > 1. Develop formulas for the parent and children of the node stored in position i of the array.

Answer:

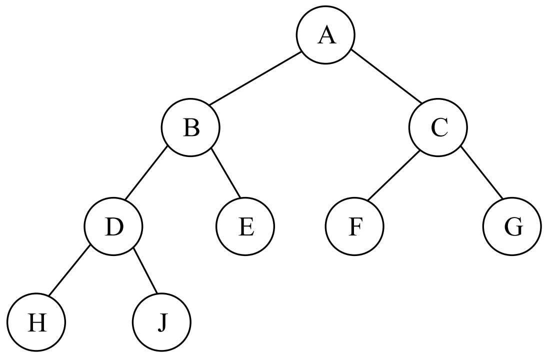
For a node stored in position i, its children's position should be {id-(d-2), id-(d-1), …, id, id+1},

and its parents' position should be $\left\lfloor \frac{i+d-2}{d} \right\rfloor$.

5. (8%) Write out the inorder, preorder, postorder, and levelorder traversals for the following binary trees.



(a)



(b)

Answer:

(a) Inorder: A – B * C * D + E
Preorder: + * * - A B C D E.
Postorder: A B – C * D * E +.
Levelorder: + * E * D – C A B.

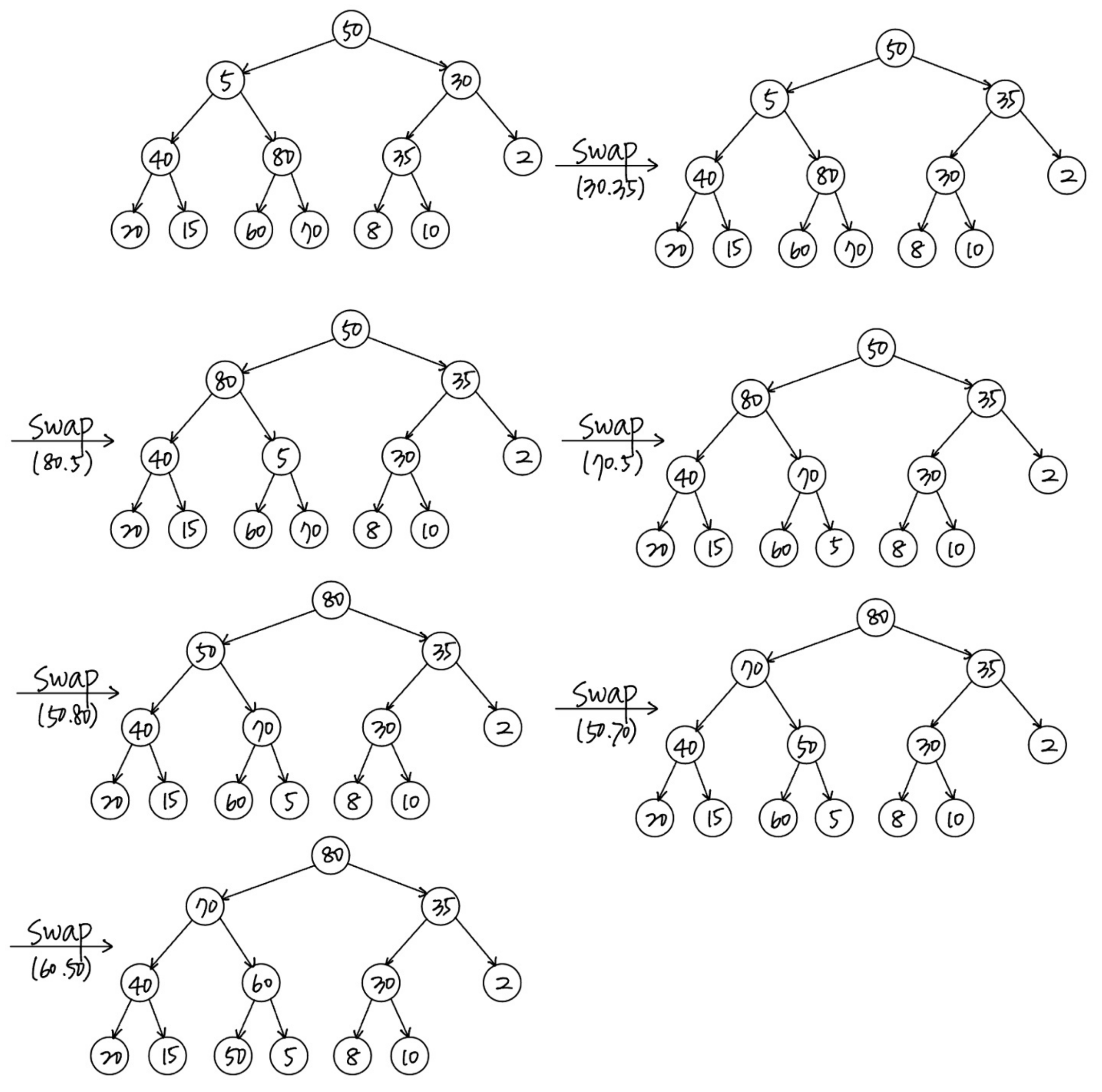
(b) Inorder: H D J B E A F C G
Preorder: A B D H J E C F G
Postorder: H J D E B F G C A
Levelorder: A B C D E F G H I J

學號：110060007
姓名：黃俊穎

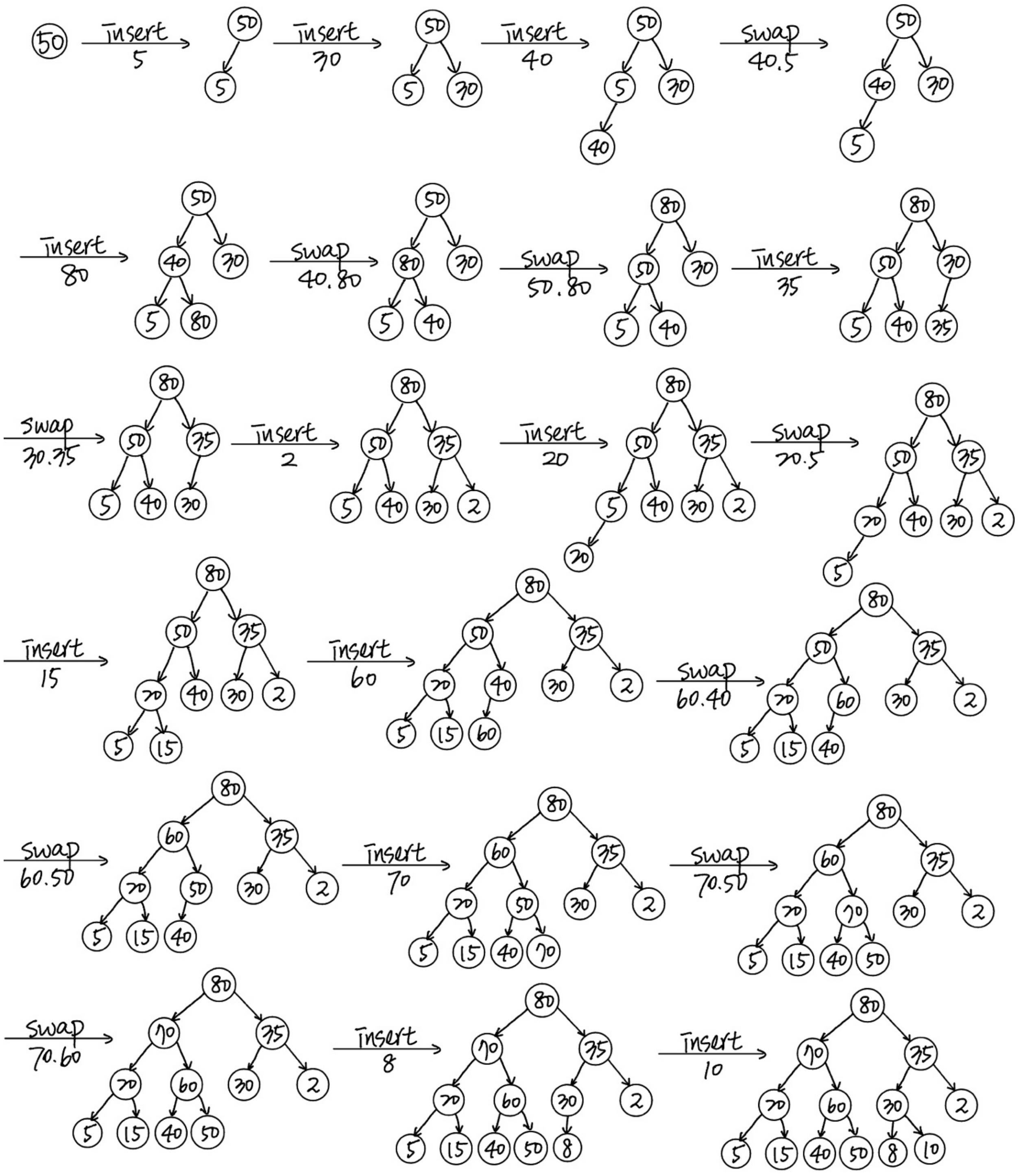
6. (4%) Given a sequence of 13 integer number: 50, 5, 30, 40, 80, 35, 2, 20, 15, 60, 70, 8, 10.
   (a) Assume a **max heap** tree is **initialize** with these 13 numbers placed into nodes of the tree according to node numbering of complete binary tree by using the **bottom up heap construction initialization** process. Please draw the final Max heap tree after initialization process.
   (b) Construct a max heap by **inserting** the given 13 numbers one by one according to the sequence order into an initially empty max heap tree, instead of bottom up heap construction.

Answer:

(a)



(b)

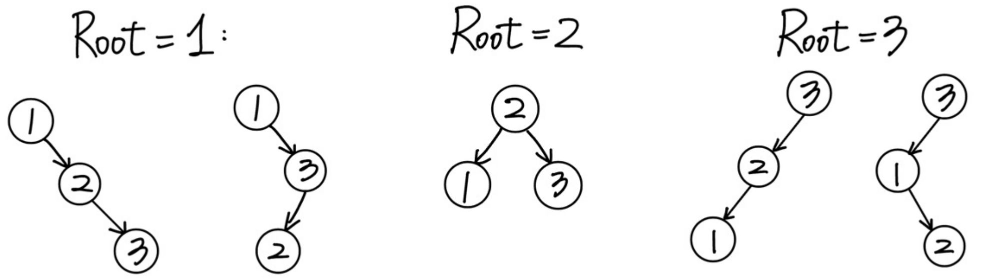

Insert 5
Insert 30
Insert 40
Swap 40.5
Insert 80
Swap 40.80
Swap 50.80
Insert 35
Swap 30.35
Insert 2
Insert 20
Swap 20.5
Insert 15
Insert 60
Swap 60.40
Swap 60.50
Insert 70
Swap 70.50
Swap 70.60
Insert 8
Insert 10

學號：110060007
姓名：黃俊穎

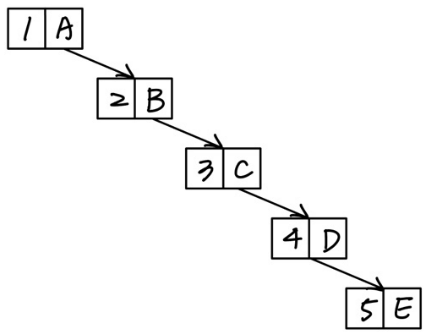
7. (10%) Binary Search Tree
   (a) How many different binary search trees can store the keys {1,2,3}?
   (b) If we insert the entries (1,A), (2,B), (3,C), (4,D), and (5,E), where the number denotes the key value of the node, in this order, into an initially empty binary search tree, what will it look like? Please draw this BST.
   (c) John claims that the order in which a fixed set of entries is inserted into a binary search tree does not matter—the same tree results every time. Give a small example that proves he is wrong.
   (d) Given a sequence of 13 integer number: 50, 5, 30, 40, 80, 35, 2, 20, 15, 60, 70, 8, 10, use the BST Insert function (manually) to insert the 13 number sequentially to construct a binary search tree. Draw the final 13-node BST.
   (e) A binary search tree produces the following preorder traversal, where "null" indicates an empty subtree (i.e. the left/right child is the null pointer).
   9,5,3,1,null,null,4,null,null,8,6,null,null,null,20,12,10,null,11,null,null,null,30,21,null,null,31,null,null
   Draw the tree that produced this preorder traversal.

Answer:

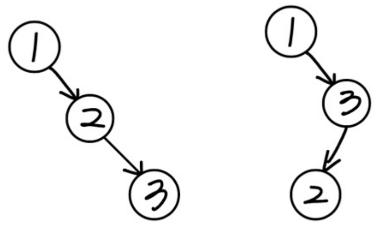
(a) There are 5 different binary search trees can store the keys {1,2,3}.



(b)

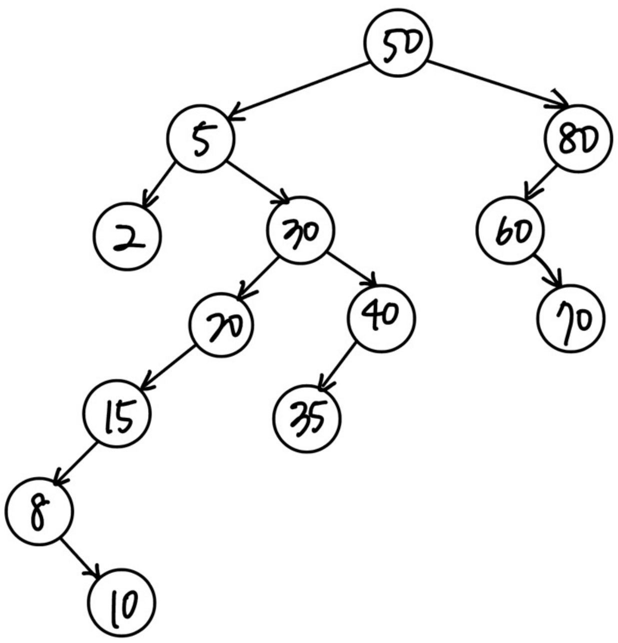


(c) By the plot from (a), we can know that the order in which a fixed set of entries is inserted into a binary search tree will influence the appearance of BST.

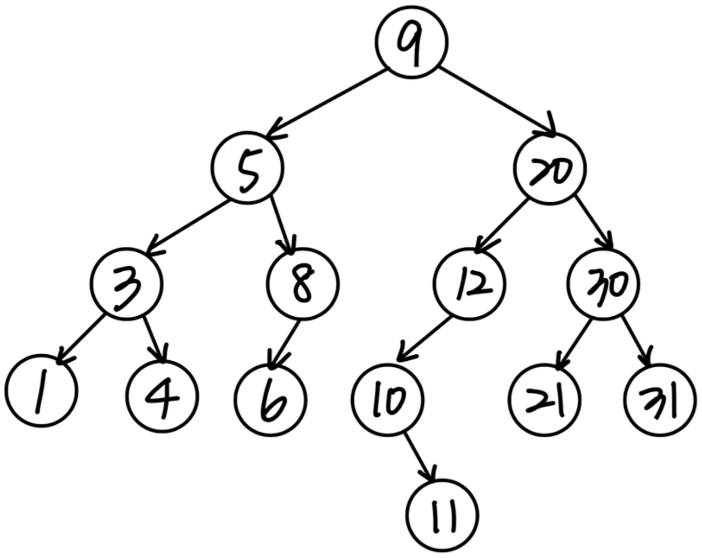


Insert by 1 -> 2 -> 3   Insert by 1 -> 3 -> 2

(d)

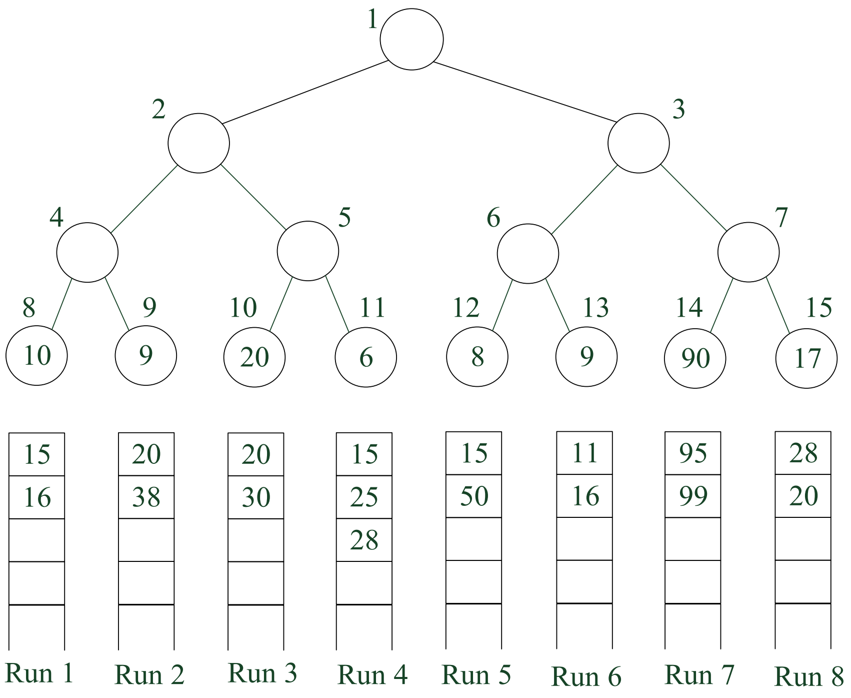


(e) There's a key to find leaves is that if there're 2 NULLs after a node, that node is one of the leaves of BST.



學號：110060007
姓名：黃俊穎

8. (2%) An 8-run with total of 25 numbers are to be merged using Winner tree and Loser tree, respectively. The numbers of the 8 runs are shown below. The first numbers from each of the 8 runs have been placed in the leaf nodes of the tree as shown. Then these eight numbers enter the tournament to get the overall winner.

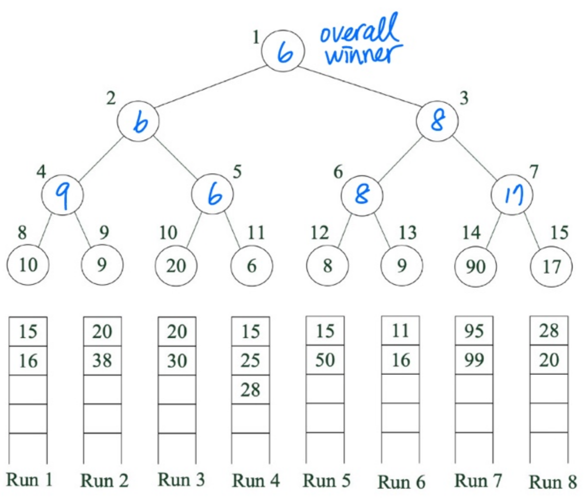


(a) Draw the winner tree and indicate the overall winner of this tournament.

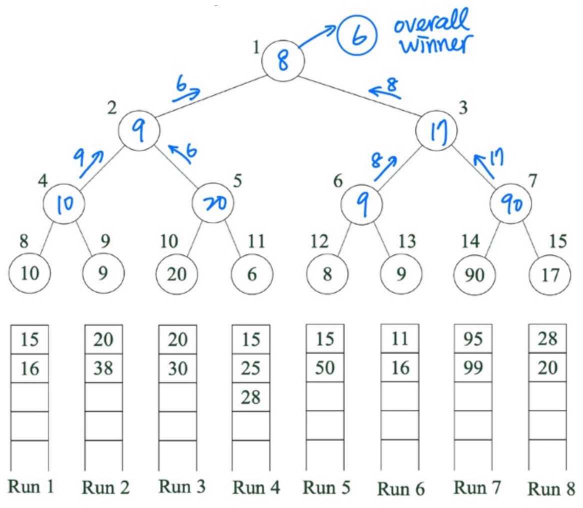
(b) Draw the loser tree and indicate (draw) the overall winner of this tournament.

Answer:

(a) Winner tree



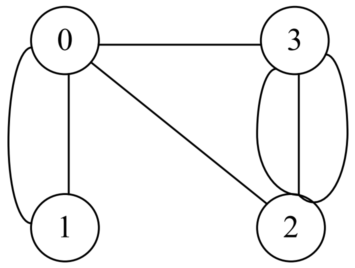
(b) Loser tree



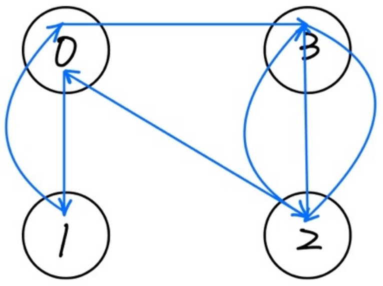
學號：110060007
姓名：黃俊穎

Graphs:

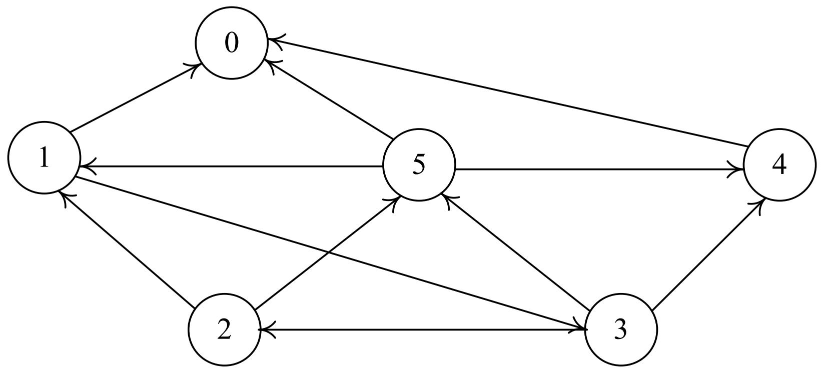
9. (2%) Does the multigraph below have an Eulerian walk? If so, find one.



Answer:



10. (4%) For the digraph below obtain
    (a) The in-degree and out-degree of each vertex
    (b) Its adjacency-matrix
    (c) Its adjacency-list representation
    (d) Its strongly connected components



Answer:

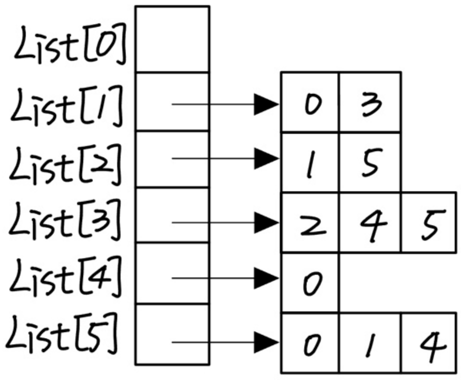
(a) In-degree and out-degree of each vertex:

Node	In-Degree	Out-Degree
Node 0	3	0
Node 1	2	2
Node 2	1	2
Node 3	1	3
Node 4	2	1
Node 5	2	3

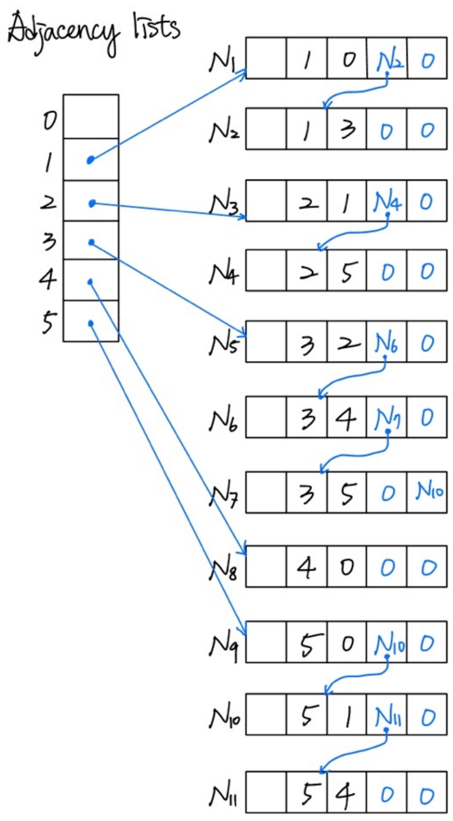
(b) Adjacency-matrix

$$\begin{array}{c|cccccc}
 & 0 & 1 & 2 & 3 & 4 & 5 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 1 & 0 & 0 \\
2 & 0 & 1 & 0 & 0 & 0 & 1 \\
3 & 0 & 0 & 1 & 0 & 1 & 1 \\
4 & 1 & 0 & 0 & 0 & 0 & 0 \\
5 & 1 & 1 & 0 & 0 & 1 & 0
\end{array}$$

(c) Adjacency-list representation

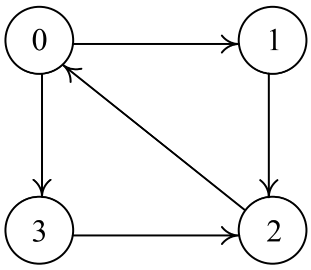


(d) Strongly connected components



學號：110060007
姓名：黃俊穎

11. (2%) Is the digraph below strongly connected? List all the simple paths.



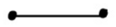
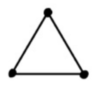
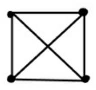
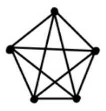
Answer: The digraph above is strongly connected

Starting vertex	Path to travel every vertex
0	$0 \to 1 \to 2 \to 0$ $0 \to 1$ $0 \to 1 \to 2$ $0 \to 3$
1	$1 \to 2 \to 0 \to 1$ $1 \to 2$ $1 \to 2 \to 0 \to 3$ $1 \to 2 \to 0$
2	$2 \to 0 \to 1 \to 2$ $2 \to 0 \to 3$ $2 \to 0$ $2 \to 0 \to 1$
3	$3 \to 2 \to 0 \to 3$ $3 \to 2 \to 0$ $2 \to 0 \to 1$ $3 \to 2$

學號：110060007
姓名：黃俊穎

12. (4%) Draw the complete undirected graphs on two, three, four, and five vertices. Prove that the number of edges in an n-vertex complete graph is n(n-1)/2.

Answer:

1 vertex:		0 edge
2 vertices:		1 edge
3 vertices:		3 edges
4 vertices:		6 edges
5 vertices:		10 edges

Obverse the law, we can conclude that as the graph has n vertices, it has 1+2+…+(n-1) edges.

$$1+2+\cdots+(n-1)=\sum_{k=0}^{n-1}k=\frac{n(n-1)}{2}$$

13. (4%) Apply depth-first and breadth-first searches to the complete graph on four vertices. Assume that vertices are numbered 0 to 3, are stored in increasing order in each list in the adjacency-list representation, and both traversals begin at vertex 0. List the vertices in the order they would be visited.

Answer:

Depth-First Search (DFS): Starting from vertex 0, the order of visiting vertices in a depth-first search is as follows: $0 \to 1 \to 2 \to 3$

Breadth-First Search (BFS): Starting from vertex 0, the order of visiting vertices in a breadth-first search is as follows: $0 \to 1 \to 2 \to 3$

In both DFS and BFS, the order of visiting vertices is the same since it is a complete graph.

學號：110060007
姓名：黃俊穎

14. (6%) Let G be a graph whose vertices are the integers 1 through 8, and let the adjacent vertices of each vertex be given by the table below:

Vertex	***Adjacent Vertices***
1	$(2, 3, 4)$
2	$(1, 3, 4)$
3	$(1, 2, 4)$
4	$(1, 2, 3, 6)$
5	$(6, 7, 8)$
6	$(4, 5, 7)$
7	$(5, 6, 8)$
8	$(5, 7)$

Assume that, in a traversal of G, the adjacent vertices of a given vertex are returned in the same order as they are listed in the table above.

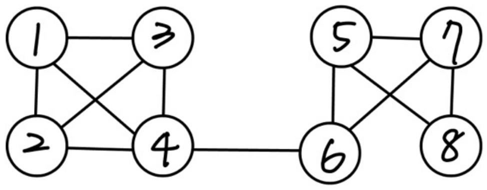
(a) Draw G.

(b) Give the sequence of vertices of G visited using a DFS traversal starting at vertex 1.

(c) Give the sequence of vertices visited using a BFS traversal starting at vertex 1.

Answer:

(a)

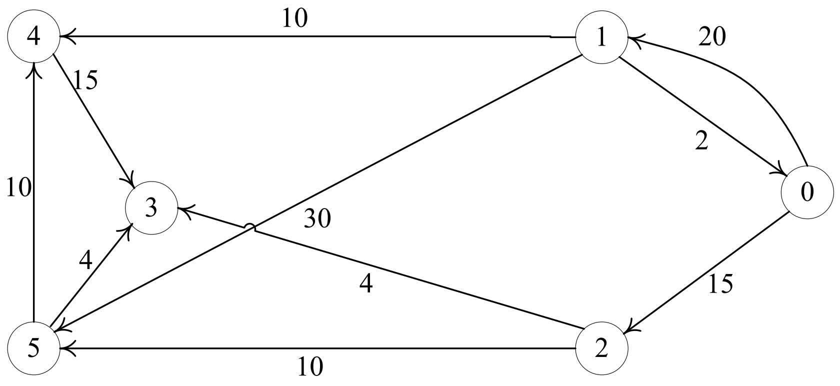


(b) DFS: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 6 \rightarrow 5 \rightarrow 7 \rightarrow 8$

(c) BFS: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 6 \rightarrow 5 \rightarrow 7 \rightarrow 8$

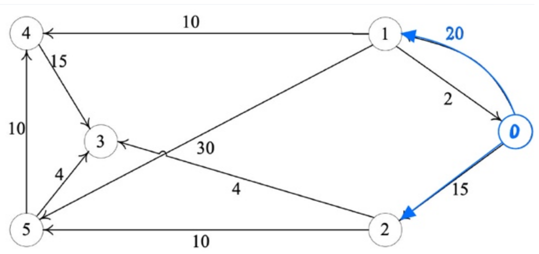
學號：110060007
姓名：黃俊穎

15. (4%) Use ShortestPath (Program 6.8) ( ) to obtain, in nondecreasing order, the lengths and the paths of the shortest paths from vertex 0 to all remaining vertices in the graph below.

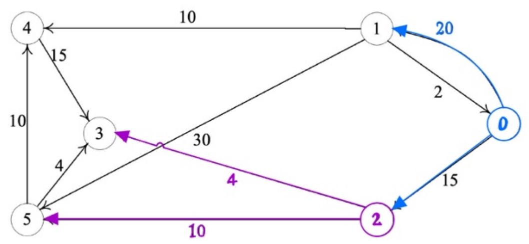


Answer:

Step1: Begin at 0, choose path 0 → 2



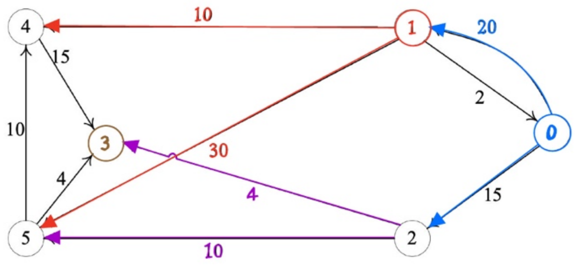
	0	1	2	3	4	5
D[i]	0	20	15	∞	∞	∞
P[i]	0	0	0	-1	-1	-1



Step2: Placed at 2, choose path 2 → 3

	0	1	2	3	4	5
D[i]	0	20	15	19 (15+4)	∞	25 (15+10)
P[i]	0	0	0	2	-1	2

學號：110060007
姓名：黃俊穎



Step3: Choose path 1 → 4

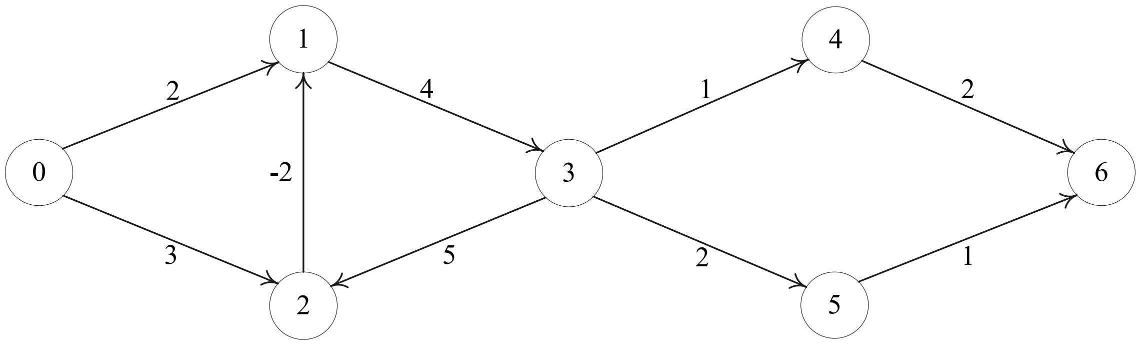
	0	1	2	3	4	5
D[i]	0	20	15	19	30(20 + 10)	25
P[i]	0	0	0	2	1	2

Shortest path of every vertex from vertex 0:

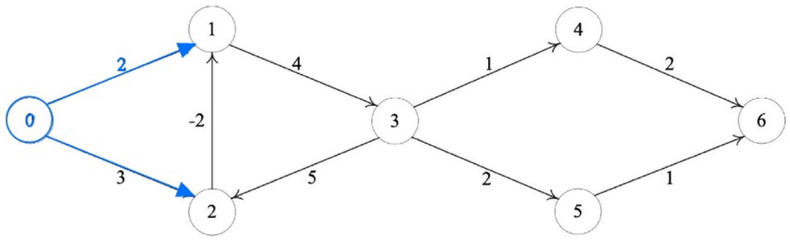
Target	Path	Shortest path length
0 → 1	0 → 1	20
0 → 2	0 → 2	15
0 → 3	0 → 2 → 3	19
0 → 4	0 → 1 → 4	30
0 → 5	0 → 2 → 5	25

學號：110060007
姓名：黃俊穎

16. (4%) Using the directed graph below, explain why ShortestPath (Program 6.8) will not work properly. What is the shortest path between vertices 0 and 6?

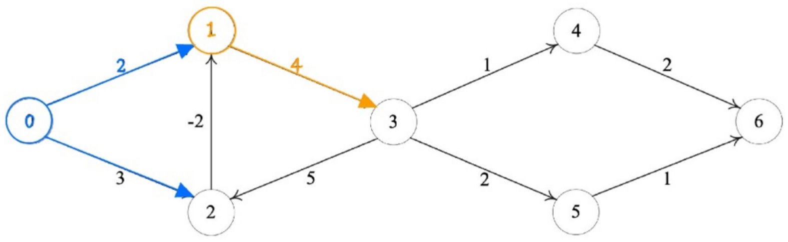


Answer:



Step1: Choose path 0 → 1

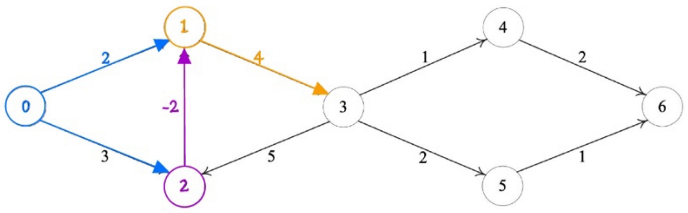
	0	1	2	3	4	5	6
D[i]	0	2	3	∞	∞	∞	∞
P[i]	0	0	0	-1	-1	-1	-1



Step2: Choose path 1 → 3

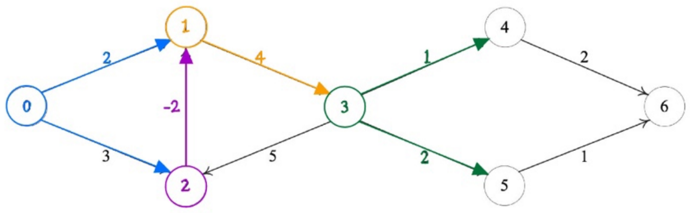
	0	1	2	3	4	5	6
D[i]	0	2	3	6(2 + 4)	∞	∞	∞
P[i]	0	0	0	1	-1	-1	-1

學號：110060007
姓名：黃俊穎



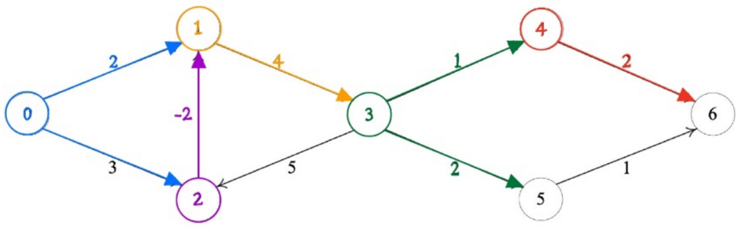
Step3: Choose path 2 → 1

	0	1	2	3	4	5	6
D[i]	0	1	3	6	∞	∞	∞
P[i]	0	2	0	1	-1	-1	-1



Step4: Choose path 3 → 4

	0	1	2	3	4	5	6
D[i]	0	1	3	6	7(2+4+1)	8(2+4+2)	∞
P[i]	0	2	0	1	3	3	-1



Step5: Choose path 4 → 6

	0	1	2	3	4	5	6
D[i]	0	1	3	6	7	8	9(2+4+1+2)
P[i]	0	2	0	1	3	3	4

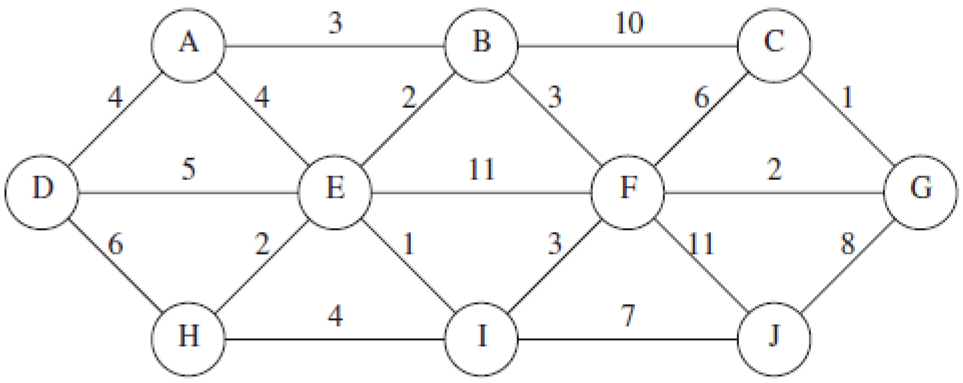
Discussion of ShortestPath from vertex 0 to 6:

Target	Path	Shortest path length
0 → 6	0 → 1 → 3 → 4 → 6	9
0 → 6	0 → 2 → 1 → 3 → 4 → 6	8

The reason why we can’t get the shortest path by Dijkstra’s algorithm is that Dijkstra’s algorithm can’t accept the edge with negative weight.

學號：110060007
姓名：黃俊穎

17. (4%) For the weighted graph G shown below,

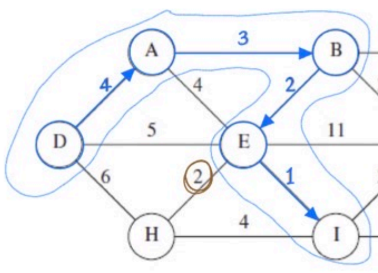


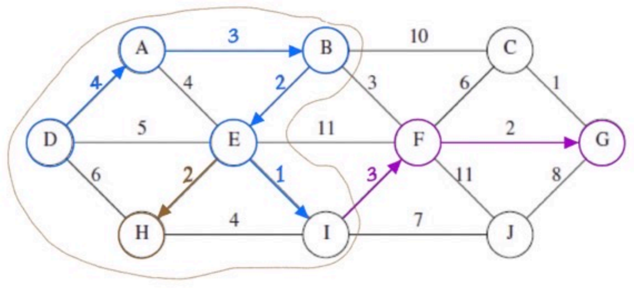
(a) Find a minimum spanning tree for the graph using both Prim’s and Kruskal’s algorithms.

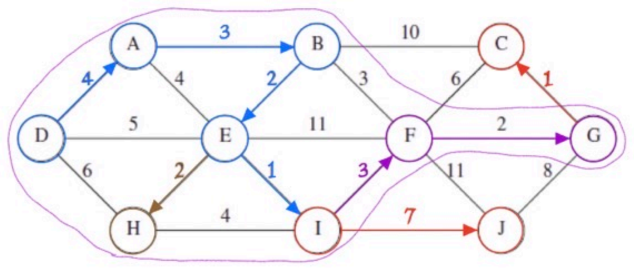
(b) Is this minimum spanning tree unique? Why?

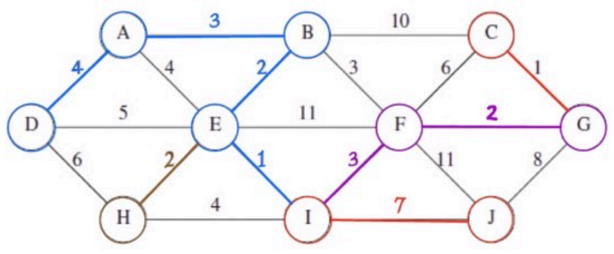
Answer:

(a) By Prim’s algorithms:

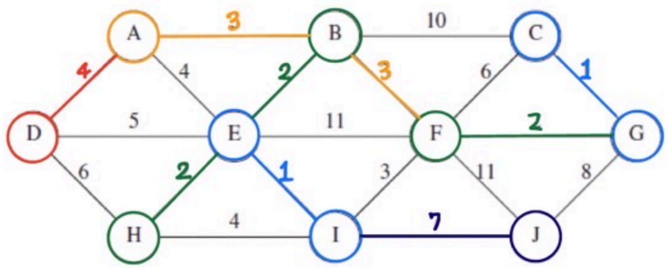








By Kruskal's algorithms:



(b) By (a), we can know that it's NOT unique since edge(I,F) and edge(B,F) have the same weight 3, if we choose them all, it'll form a loop which violates the law of spanning tree.

18. (2%) Does the following set of precedence relations (<) define a partial order on the elements 0 through 4? Why?

$0 < 1; 1 < 3; 1 < 2; 2 < 3; 2 < 4; 4 < 0$

Answer: NO

By 0 < 1and 1 < 2 implies 0 < 2. By 0 < 2 and 2 < 4 implies 0 < 4 instead of 4 < 0, so we can know this set is reflexive rather than partial order.

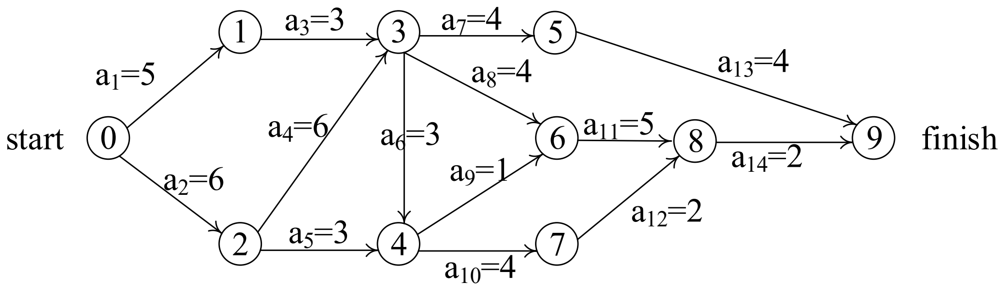
19. (4%) For the AOE network shown below,

(a) Obtain the early, e(a_i), and late, l(a_i), start times for each activity. Use the forward-backward approach.

(b) What is the earliest time the project can finish?

(c) Which activities are critical? Fill the table below for answers to (a), (b), and (c).

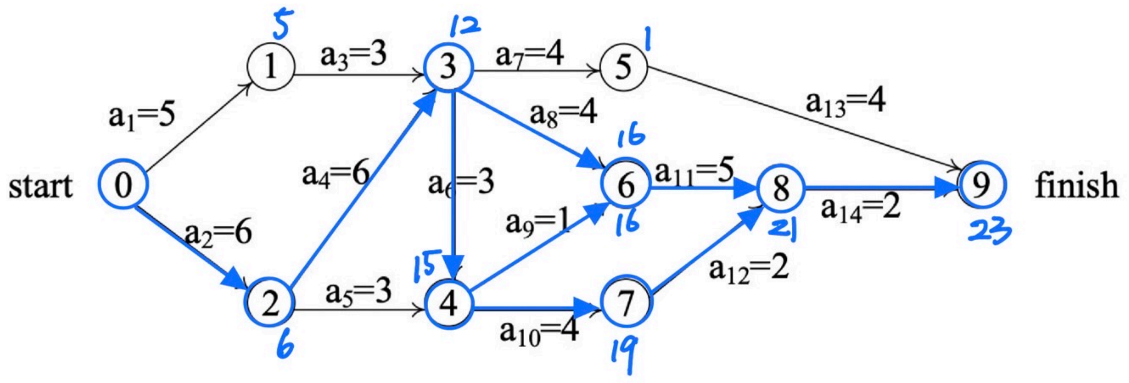
(d) Is there any single activity whose speed-up would result in a reduction of the project finish time?



學號：110060007
姓名：黃俊穎

Answer:

(a) (c) The plot below represents the critical path:



activity	Early time	Late time	slack	critical
	$e(a_i)$	$l(a_i)$	$l(a_i) - e(a_i)$	
a_1	0	4	4	False
a_2	0	0	0	True
a_3	5	9	4	False
a_4	6	6	0	True
a_5	6	12	6	False
a_6	12	12	0	True
a_7	12	15	3	False
a_8	12	12	0	True
a_9	15	15	0	True
a_{10}	15	15	0	True
a_{11}	16	16	0	True
a_{12}	19	19	0	True
a_{13}	16	19	3	False
a_{14}	21	21	0	True

(b) Earliest time: 23

(d) Since a_2, a_4, and a_{14} have no other alternative path, if we could speed up toward these paths, it might result in a reduction of the project finish time.

學號：110060007
姓名：黃俊穎

Sorting:

20. (10%) The list L: (12, 2, 16, 30, 8, 28, 4, 10, 20, 6, 18) is to be sorted by various sorting algorithm.

(a) Write the status of the list at the end of each iteration of the **for** loop of InsertionSort (Program 7.5). Trace the program; understand it. Put your answer in the following table. (add necessary rows for your answer)

(b) Trace Program 7.6 QuickSort, use it on the list L, and draw a figure similar to Figure 7.1 Quick Sort example starting with the list L. Put your answer in the following table. (add necessary rows for your answer)

(c) Write the status of the list L at the end of each phase of MergeSort (Program 7.9), i.e., draw the Merge tree (similar to Figure 7.4 in textbook) of this problem.

(d) Write the status of the list L at the end of the first **for** loop as well as at the end of the second **for** loop of HeapSort (Program 7.14), i.e., you need to draw the following trees for: 1) input array, 2) initial heap, and 9 more trees with heap size from 10 down to 2 with corresponding sorted array. You can refer to similar results shown in Figure 7.8 in textbook.

(e) Write the status of the list L at the end of each pass of RadixSort (Program 7.15), using r = 10. That is fill the missing parts (the node boxes with numbers and arrows between e[j] and f[j] enclosed by red dashed rectangle in (ii) and (iii) part of the following figure, and the missing numbers in the resulting chain (red boxes) in (ii).)

Answer:

(a) InsertionSort

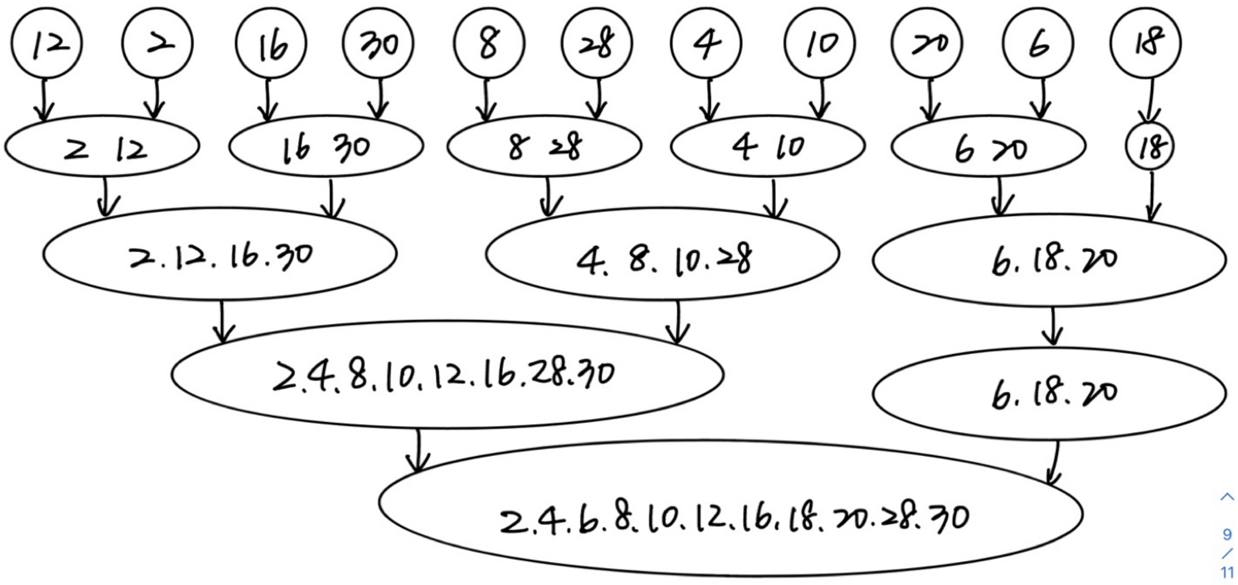
j	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]
1	12	2	16	30	8	28	4	10	20	6	18
2	2	12	16	30	8	28	4	10	20	6	18
3	2	12	16	30	8	28	4	10	20	6	18
4	2	12	16	30	8	28	4	10	20	6	18
5	2	8	12	16	30	28	4	10	20	6	18
6	2	8	12	16	28	30	4	10	20	6	18
7	2	4	8	12	16	28	30	10	20	6	18
8	2	4	8	10	12	16	28	30	20	6	18
9	2	4	8	10	12	16	20	28	30	6	18
10	2	4	6	8	10	12	16	20	28	30	18
11	2	4	6	8	10	12	16	18	20	28	30

學號：110060007
姓名：黃俊穎

(b) QuickSort

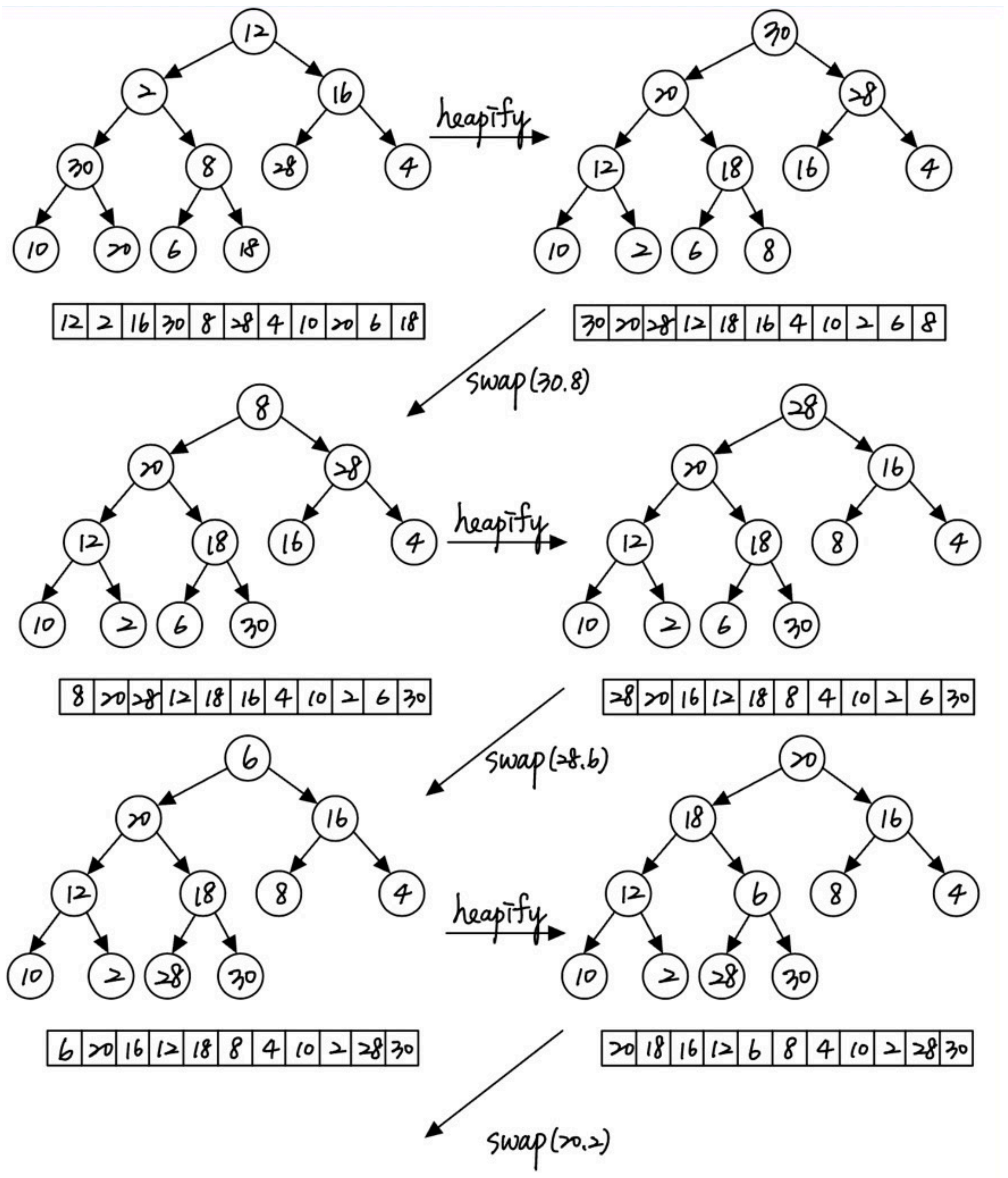
R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	R_9	R_{10}	R_{11}	left	right
[12	2	16	30	8	28	4	10	20	6	18]	1	11
[12	2	**16**	30	8	28	4	10	20	**6**	18]	1	11
[12	2	6	**30**	8	28	4	**10**	20	16	18]	1	11
[12	2	6	10	8	**28**	**4**	30	20	16	18]	1	11
[12	2	6	10	8	4	28	30	20	16	18]	1	11
[4	2	6	10	8]	12	[28	30	20	16	18]	1	5
[2]	4	[6	10	8]	12	[28	30	20	16	18]	1	1
2	4	[6	**10**	8]	12	[28	30	20	16	18]	3	5
2	4	6	[10	8]	12	[28	30	20	16	18]	4	5
2	4	6	8	10	12	[28	**30**	20	16	**18**]	7	11
2	4	6	8	10	12	[28	18	20	**16**	30]	7	11
2	4	6	8	10	12	[16	**18**	20]	28	[30]	7	9
2	4	6	8	10	12	16	[**18**	20]	28	[30]	8	9
2	4	6	8	10	12	16	18	20	28	[30]	11	11
2	4	6	8	10	12	16	18	20	28	30		

(c) MergeSort

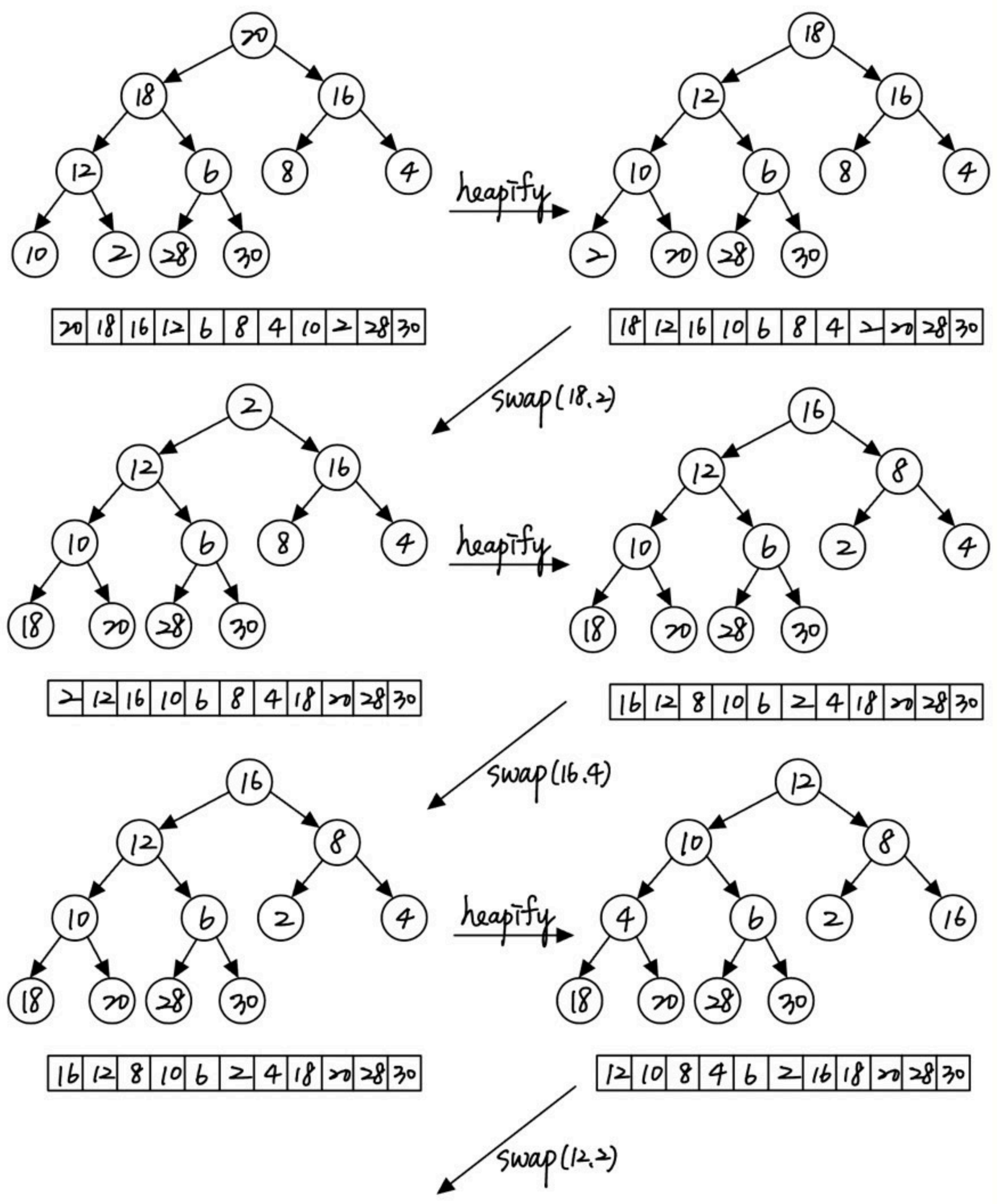


學號：110060007
姓名：黃俊穎

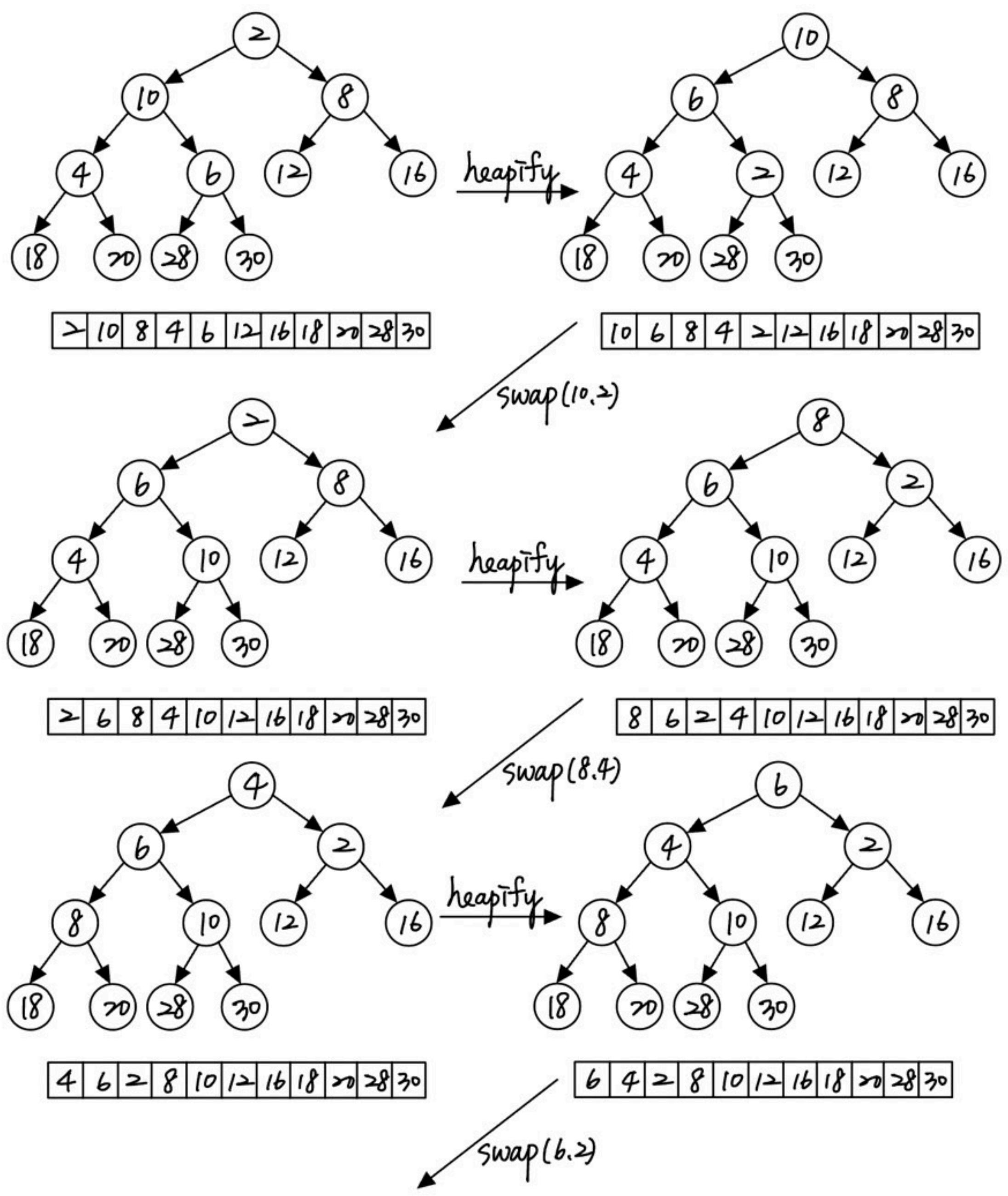
(d) HeapSort

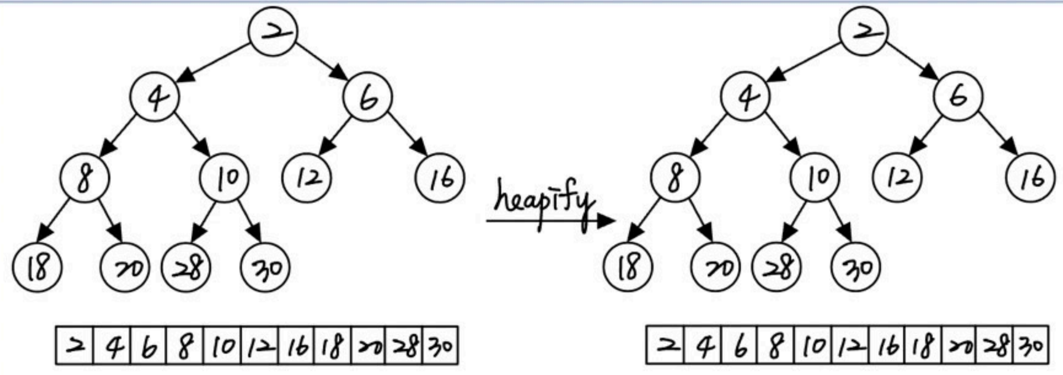


學號：110060007
姓名：黃俊穎

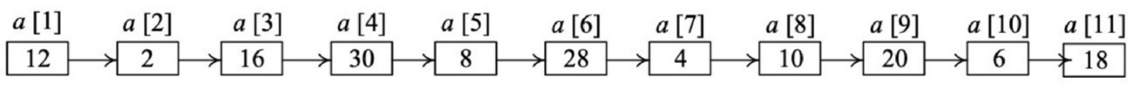


學號：110060007
姓名：黃俊穎

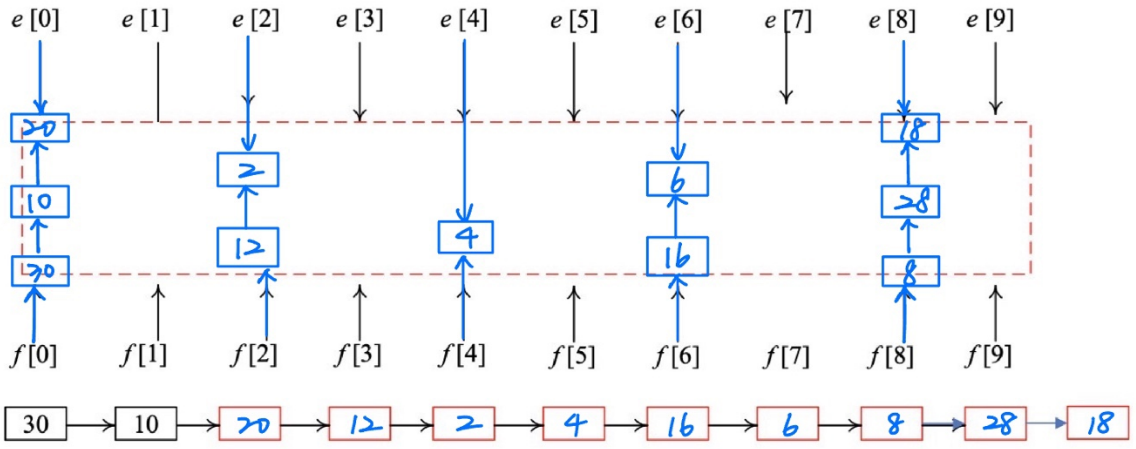




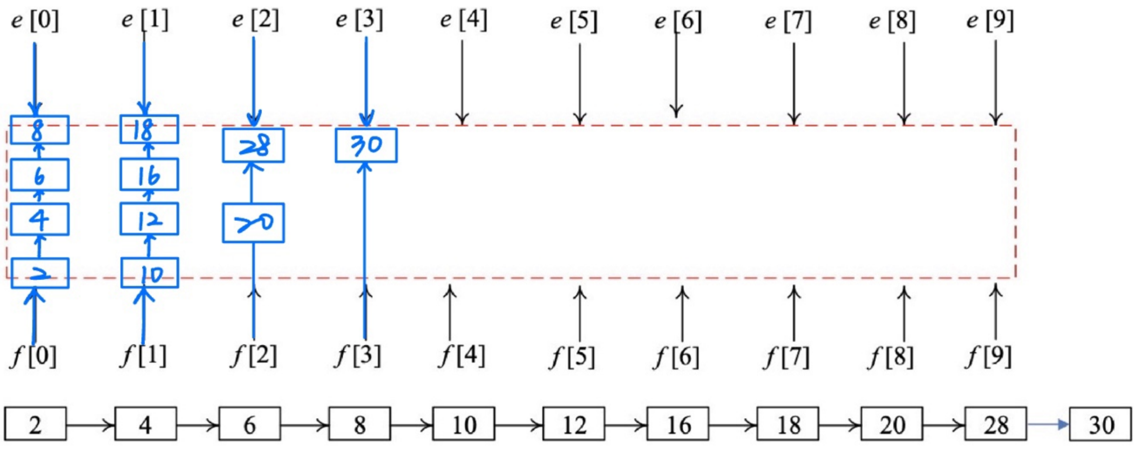
(e) RadixSort:


a [1] a [2] a [3] a [4] a [5] a [6] a [7] a [8] a [9] a [10] a [11]
12 2 16 30 8 28 4 10 20 6 18

(i) Initial input


e [0] e [1] e [2] e [3] e [4] e [5] e [6] e [7] e [8] e [9]
20 10 30
2 12
4
6 16
18 28 8
f [0] f [1] f [2] f [3] f [4] f [5] f [6] f [7] f [8] f [9]
30 10 20 12 2 4 16 6 8 28 18

(ii) First pass queues & resulting chain


e [0] e [1] e [2] e [3] e [4] e [5] e [6] e [7] e [8] e [9]
8 6 4 2
18 16 12 10
28 20
30
f [0] f [1] f [2] f [3] f [4] f [5] f [6] f [7] f [8] f [9]
2 4 6 8 10 12 16 18 20 28 30

(iii) Second pass queues & resulting chain

學號：110060007
姓名：黃俊穎

Hashing:

21. (5%) (a) Briefly explain the one-way property, weak collision resistance, or strong collision resistance regarding hash function. (b) Show that the hash function h(k) = k%17 does not satisfy the one-way property, weak collision resistance, or strong collision resistance.

Answer:

(a) One-way property:

Given a value c and a hash function h(k), it's hard to find a key k s.t. h(k) = c.

Weak collision resistance:

Given a key x, it's hard to find a key x s.t. h(x) = h(y).

Strong collision resistance:

It's hard to find a pair of key(x,y) s.t. h(x) = h(y).

(b) One-way property:

For the hash function h(k) = k%17 = c (c is const.), we can derive as k = 17n + c (n is an integer.) Thus, h(k) = k%17 is not satisfied with one-way property.

Weak collision resistance:

For the hash function h(k) = k%17 = c (c is const.), given a key x we can find the other key y with the relationship: y = 17n + x. Thus, h(k) = k%17 is not satisfied with weak collision resistance.

Strong collision resistance:

Counterexample: (x,y) = (18,35) s.t. h(k) = k%17 is not satisfied with strong collision resistance.

22. (5%) The probability $P(u)$ that an arbitrary query made after u updates results in a filter error is given by $P(u) = e^{-u/n}\left(1 - e^{-uh/m}\right)^h$. By differentiating $P(u)$ with respect to h, show that $P(u)$ is minim

學號：110060007
姓名：黃俊穎

$p(h) = e^{-\frac{u}{n}}(1-e^{-\frac{uh}{m}})^h$

$\ln p(h) = -\frac{u}{n} + h\ln(1-e^{-\frac{uh}{m}})$

$\frac{d\ln p(h)}{dh} = e^{-\frac{u}{n}}(1-e^{-\frac{uh}{m}})^h\left[\ln(1-e^{-\frac{uh}{m}}) - h\frac{e^{-\frac{uh}{m}}\cdot(-\frac{u}{m})}{1-e^{-\frac{uh}{m}}}\right] = 0$

$\Rightarrow \ln(1-e^{-\frac{uh}{m}}) - h\frac{e^{-\frac{uh}{m}}\cdot(-\frac{u}{m})}{1-e^{-\frac{uh}{m}}} = 0$

set $x = 1-e^{-\frac{uh}{m}} \Rightarrow e^{-\frac{uh}{m}} = 1-x \Rightarrow -\frac{uh}{m} = \ln(1-x) \Rightarrow h = -\frac{m}{u}\ln(1-x)$

$\Rightarrow x\ln x = (1-x)\ln(1-x) \Rightarrow x = \frac{1}{2}$ $\therefore 1-e^{-\frac{uh}{m}} = \frac{1}{2} \Rightarrow e^{-\frac{uh}{m}} = \frac{1}{2} \Rightarrow h = \frac{m}{u}\ln 2$

$\therefore$ as $h = \frac{m}{u}\ln 2$, $p(h)$ has the minimum.