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

by110061212 郭郁翔

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.

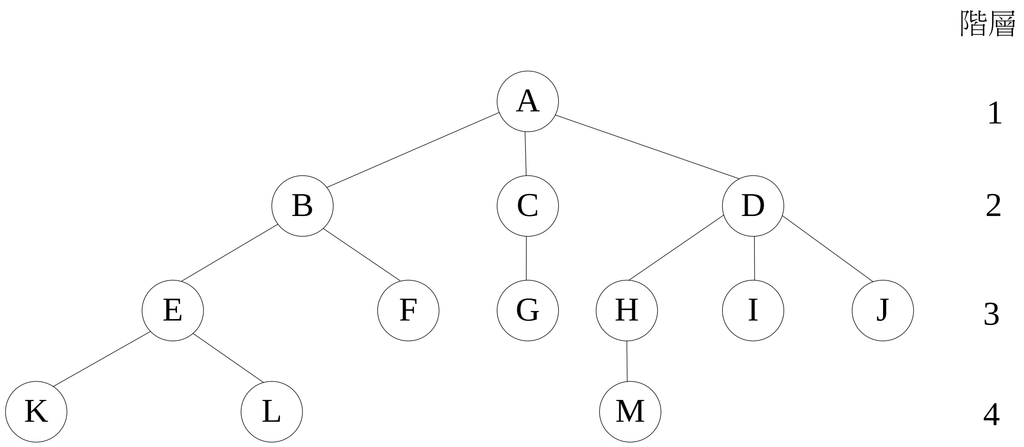
Trees:

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

   A. First level has one node, second level has k nodes, third level has k^2 nodes, so on and so forth.

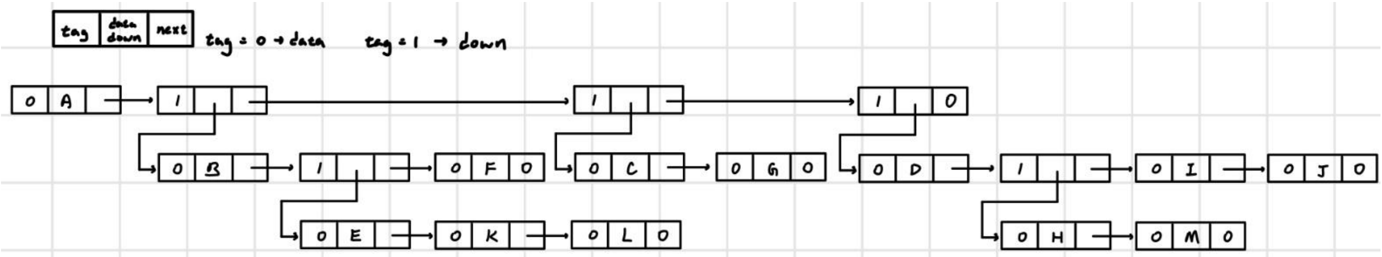
   So, the maximum number of nodes $\sum_{n=0}^{h-1} k^n = \frac{k^h-1}{k-1}$

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.



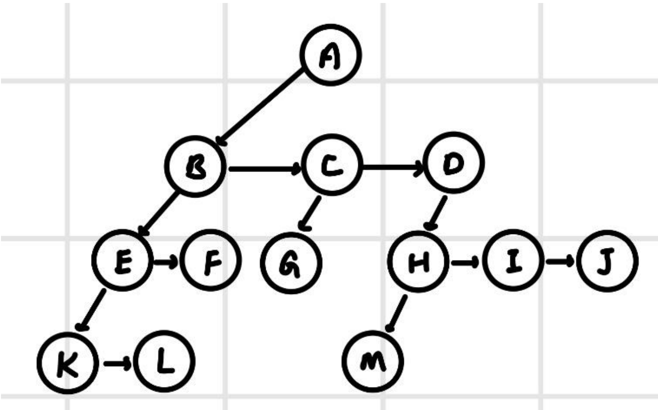
A.

(a)

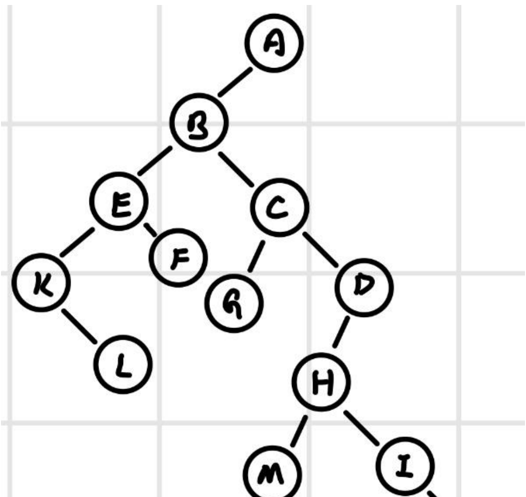


(b) A(B(E(K, L), F), C(G), D(H(M), I, J))

(c)



(d)



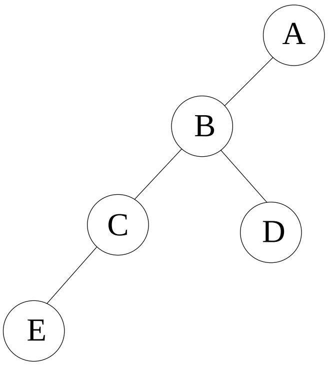
(e) depth of node L : 4 ,   height of node B : 3,   height of the tree : 4

(f) preorder : A B E K L F C G D H M I J

(g) postorder : K L E F B G C M H I J D A

(h) level order : 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.

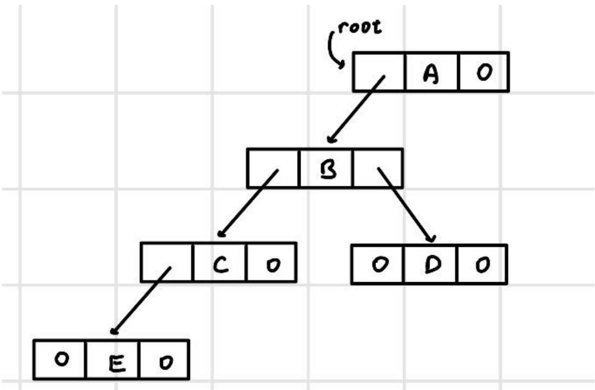


A.

(a)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
—	**A**	**B**		**C**	**D**			**E**							

(b)



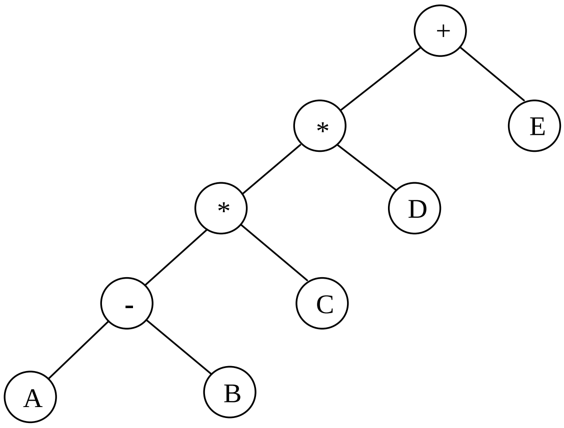
4. (2%) Extend the array representation of a complete binary tree to the case of complete trees whose degree is d, d > 1. Develop formulas for the parent and children of the node stored in position i of the array.

A.

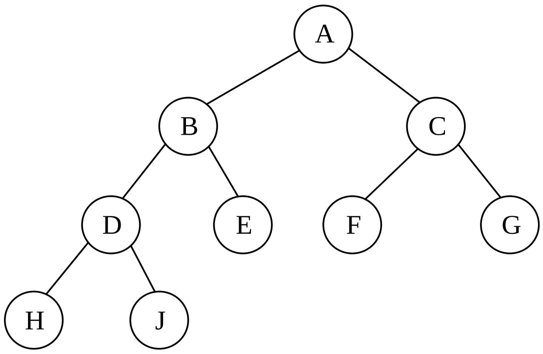
Parent of i : $\left\lfloor \frac{i+d-2}{d} \right\rfloor$

Children of i : $id-(d-2) \sim id+1$

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



(a)



(b)

A.

(a) Inorder : A - B * C * D + E

Perorder : + * * - A B C D E

Postorder : A B - C * D * E +

level order : + * E * D – C A B

(b) Inoder : H D J B E A F C G

Perorder : A B D H J E C F G

Postorder : H J D E B F G C A

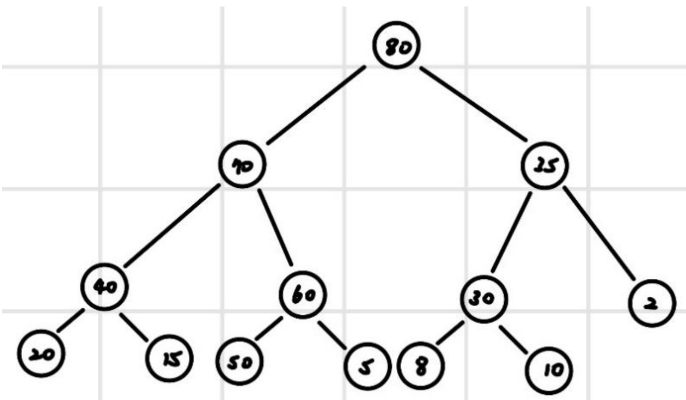
level order : A B C D E F G H J

6. (4%) Given a sequence of 13 integer number: 50, 5, 30, 40, 80, 35, 2, 20, 15, 60, 70, 8, 10.

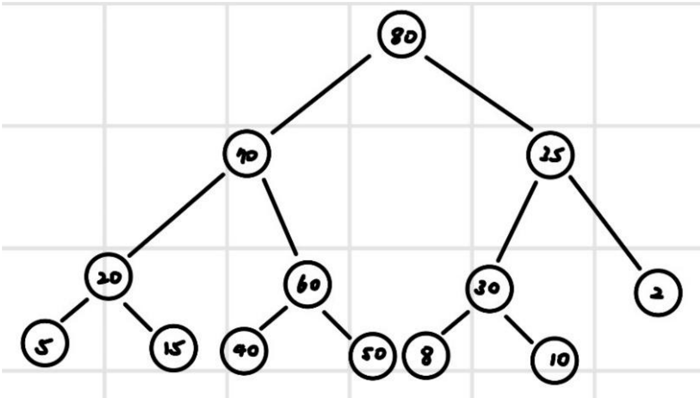
(a) Assume a **max heap** tree is **initialize** with these 13 numbers placed into nodes of the tree according to node numbering of complete binary tree by using the **bottom up heap construction initialization** process. Please draw the final Max heap tree after initialization process.

(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.

A. (a)



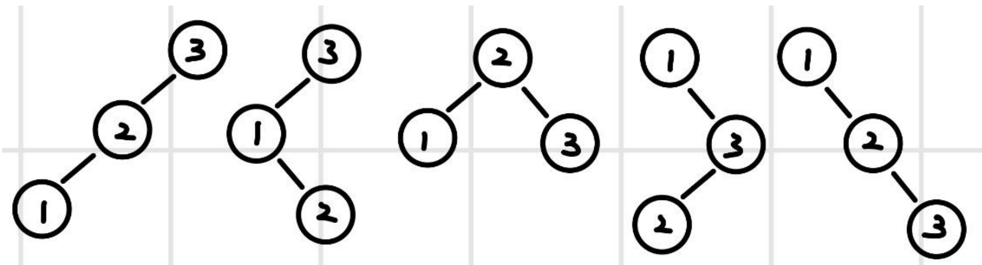
(b)



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.

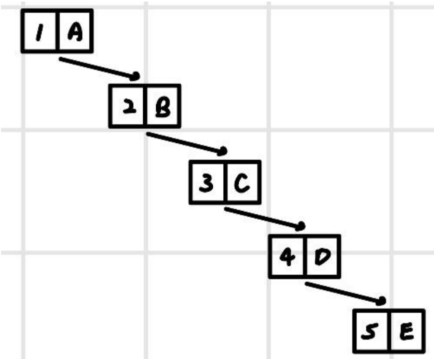
A.

(a)

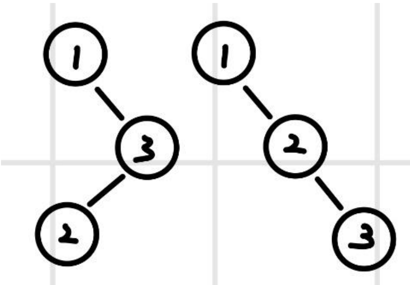


5 kinds

(b)

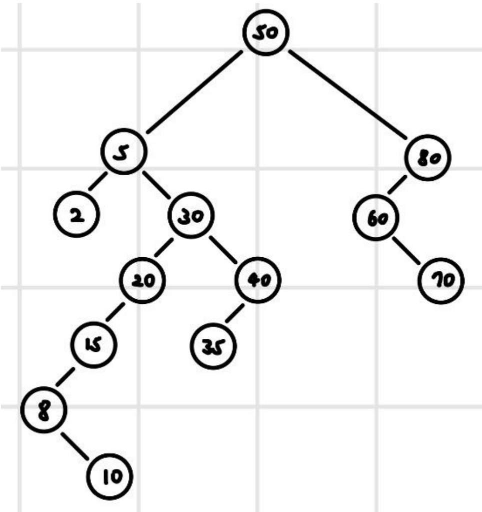


(c)

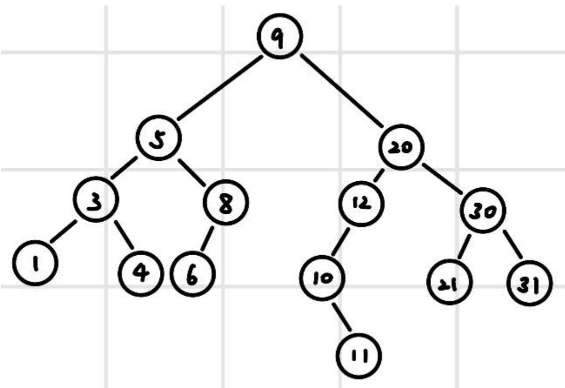


For example we insert 132 and 123 the result will be different

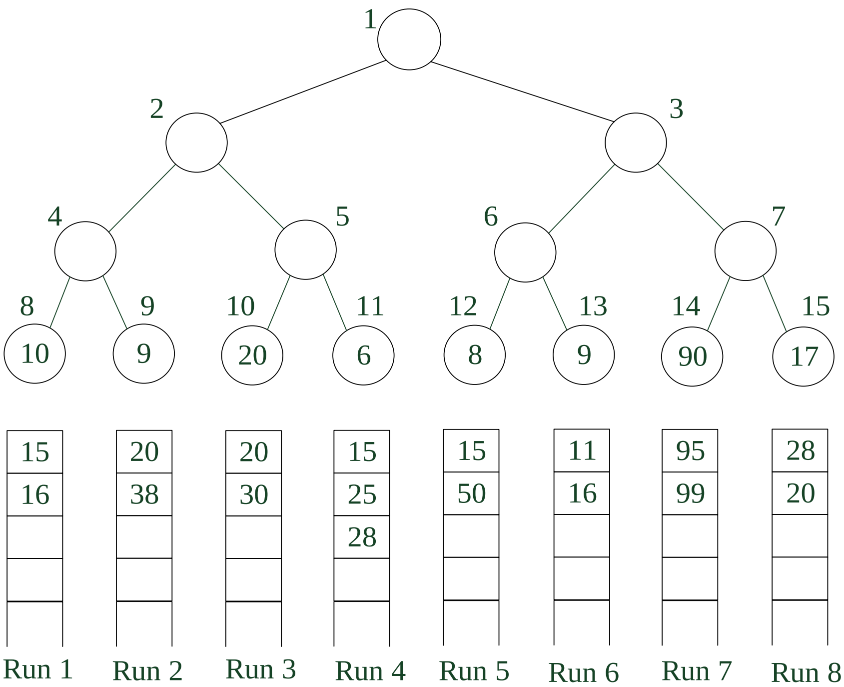
(d)



(e)



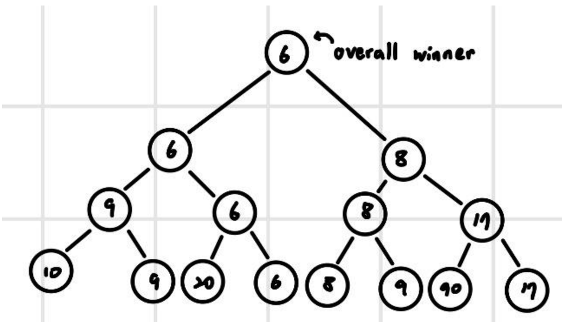
8. (2%) An 8-run with total of 25 numbers are to be merged using Winner tree and Loser tree, respectively. The numbers of the 8 runs are shown below. The first numbers from each of the 8 runs have been placed in the leaf nodes of the tree as shown. Then these eight numbers enter the tournament to get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



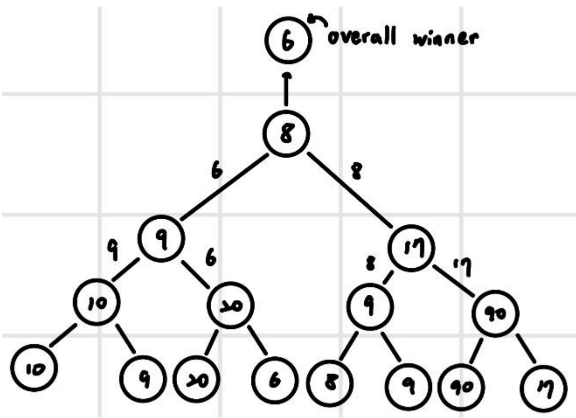
(a) Draw the winner tree and indicate the overall winner of this tournament.

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

A. (a)

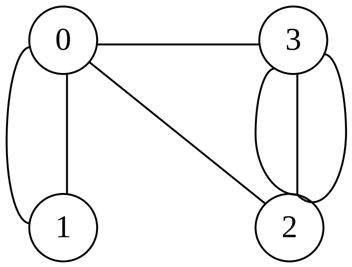


(b)



Graphs:

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



A. Yes, degree of each vertex is even

$0 \rightarrow 1 \rightarrow 0 \rightarrow 2 \rightarrow 3 \rightarrow 2 \rightarrow 3 \rightarrow 0$

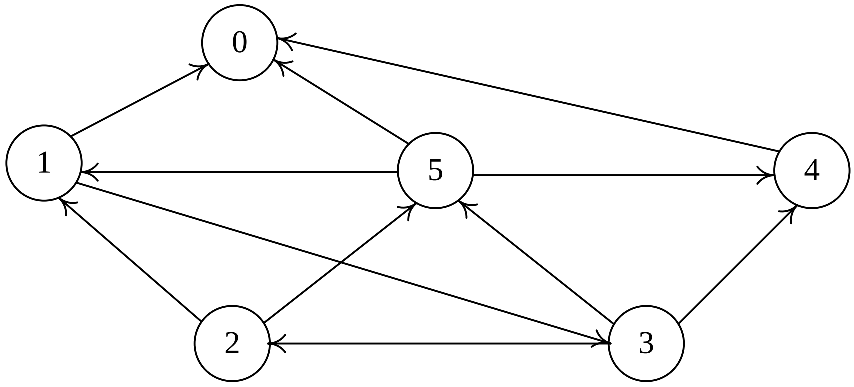
10. (4%) For the digraph below obtain

(a) The in-degree and out-degree of each vertex

(b) Its adjacency-matrix

(c) Its adjacency-list representation

(d) Its strongly connected components



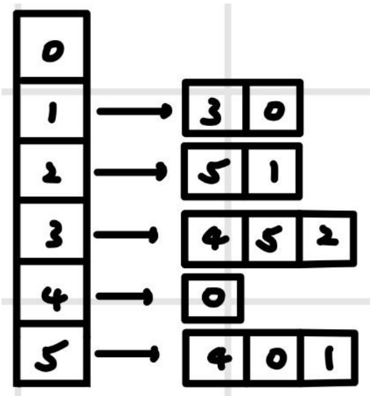
A.

(a)

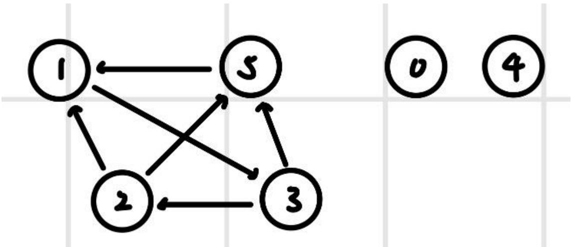
Vertex	0	1	2	3	4	5
In-degree	**3**	**2**	**1**	**1**	**2**	**2**
Out-degree	**0**	**2**	**2**	**3**	**1**	**3**

(b) $$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

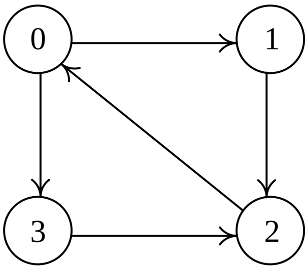
(c)



(d)



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



A.

From 0 :

0 → 1

0 → 1 → 2

0 → 3

From 1 :

1 → 2 → 0

1 → 2

1 → 2 → 0 → 3

From 2 :

2 → 0

2 → 0 → 1

2 → 0 → 3

From 3 :

3 → 2 → 0

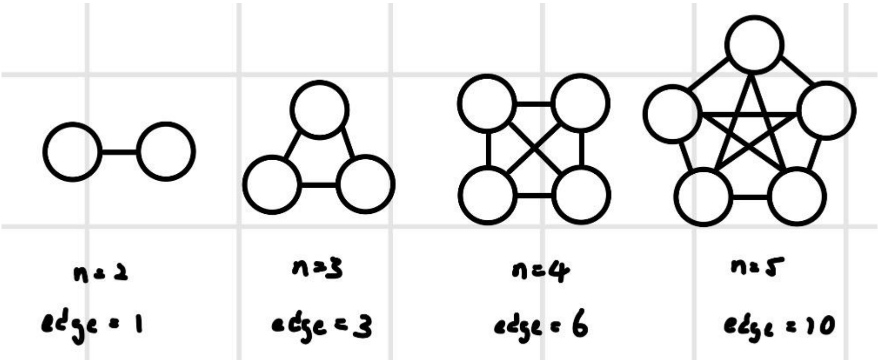
3 → 2 → 0 → 1

3 → 2

Yes, it is strongly connected.

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.

A.



Every two vertex can make an edge = C(n 取 2) = $\frac{n(n-1)}{2}$

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.

A.

Depth-first : 0 → 1 → 2 → 3

Breadth-first : 0 → 1 → 2 → 3

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.

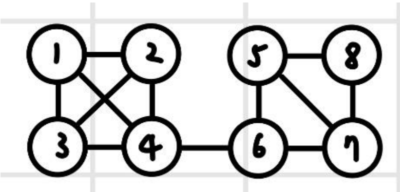
(a) Draw *G*.

(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.

A.

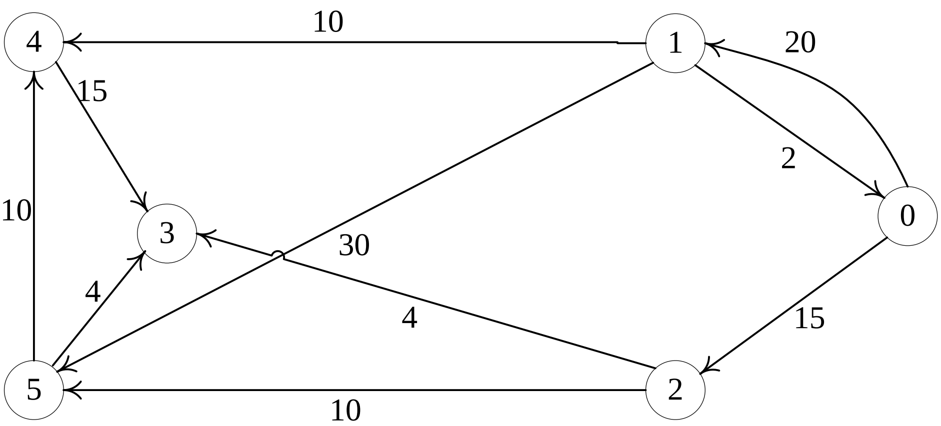
(a)



(b) 1 → 2 → 3 → 4 → 6 → 5 → 7 → 8

(c) 1 → 2 → 3 → 4 → 6 → 5 → 7 → 8

15. (4%) Use ShortestPath (Program 6.8) (Dijkstra's algorithm) 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.



A.

		0	1	2	3	4	5
1	**d**	**0**	**20**	**15**	∞	∞	∞
	p		**0**	**0**			
2		**0**	**20**	**15**	**19**	∞	**25**
			0	**0**	**2**		**2**
3		**0**	**20**	**15**	**19**	∞	**25**
			0	**0**	**2**		**2**

4		**0**	**20**	**15**	**19**	**30**	**25**
			0	**0**	**2**	**1**	**2**
5		**0**	**20**	**15**	**19**	**30**	**25**
			0	**0**	**2**	**1**	**2**

0 → 1 : 0 → 1, length = 20

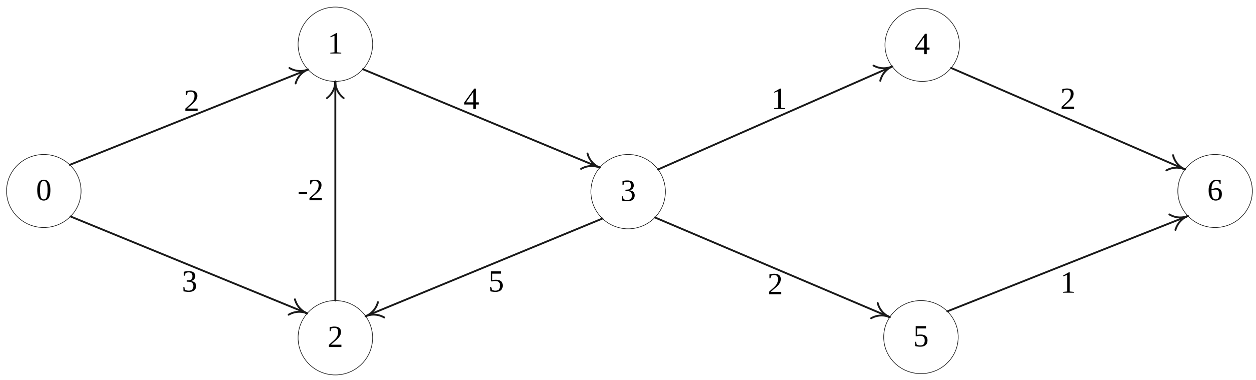
0 → 2 : 0 → 2, length = 15

0 → 3 : 0 → 2 → 3, length = 19

0 → 4 : 0 → 1 → 4, length = 30

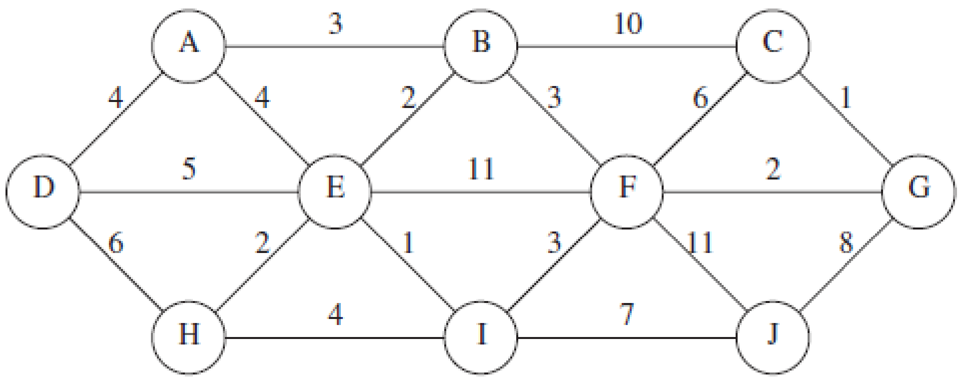
0 → 5 : 0 → 2 → 5, length = 25

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?



A. If we use Dijkstra's algorithm in the first step we will set 2 as the shortest path to 1, but actually 0 → 2 → 1 will be the shortest path length = 1, so we can see when there is negative num we can't use ShortestPath.
The shortest path from 0 to 6 will be 0 → 2 → 3 → 4 → 6, length = 8

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

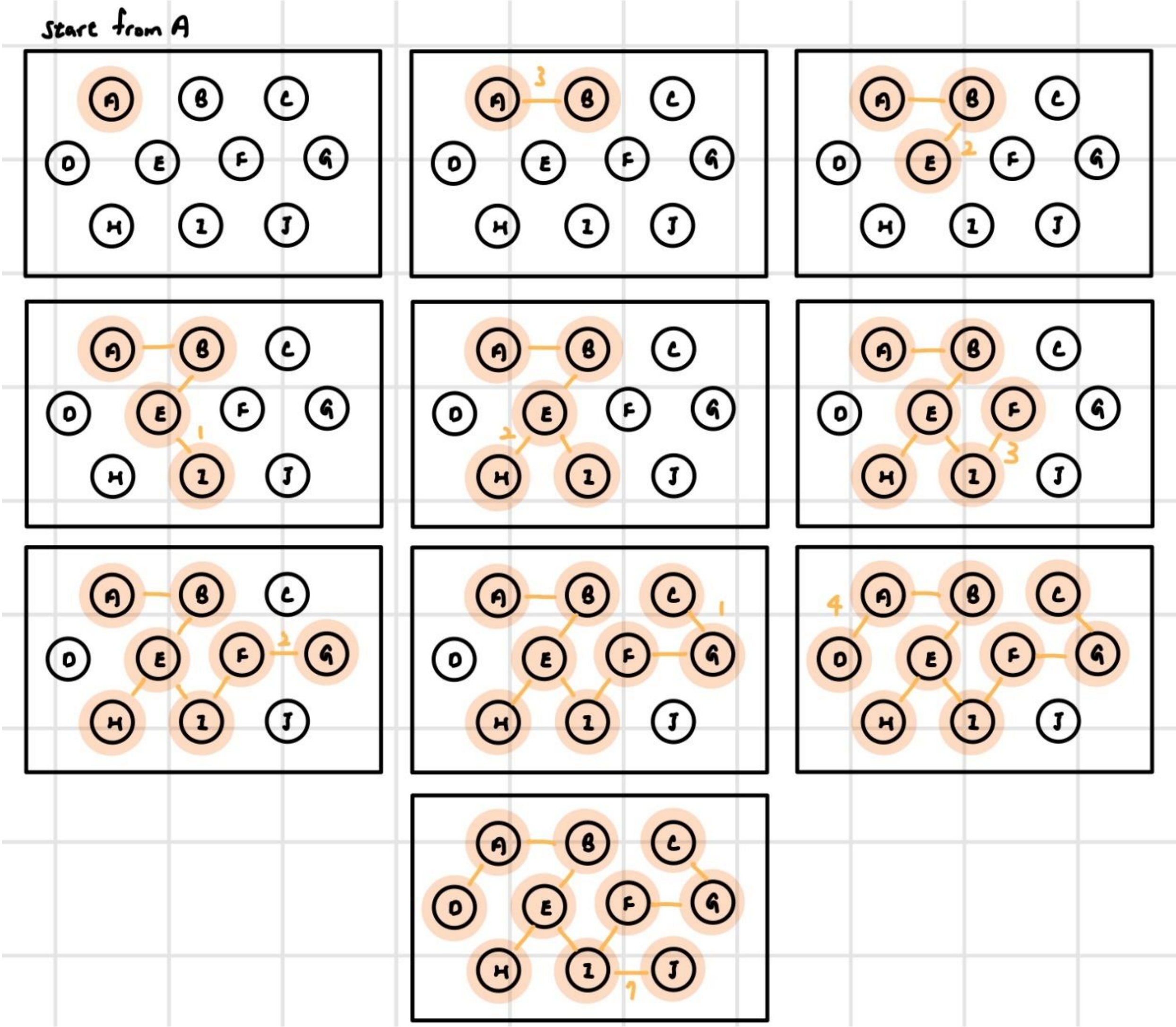


(a) Find a minimum spanning tree for the graph using both Prim's and Kruskal's algorithms.

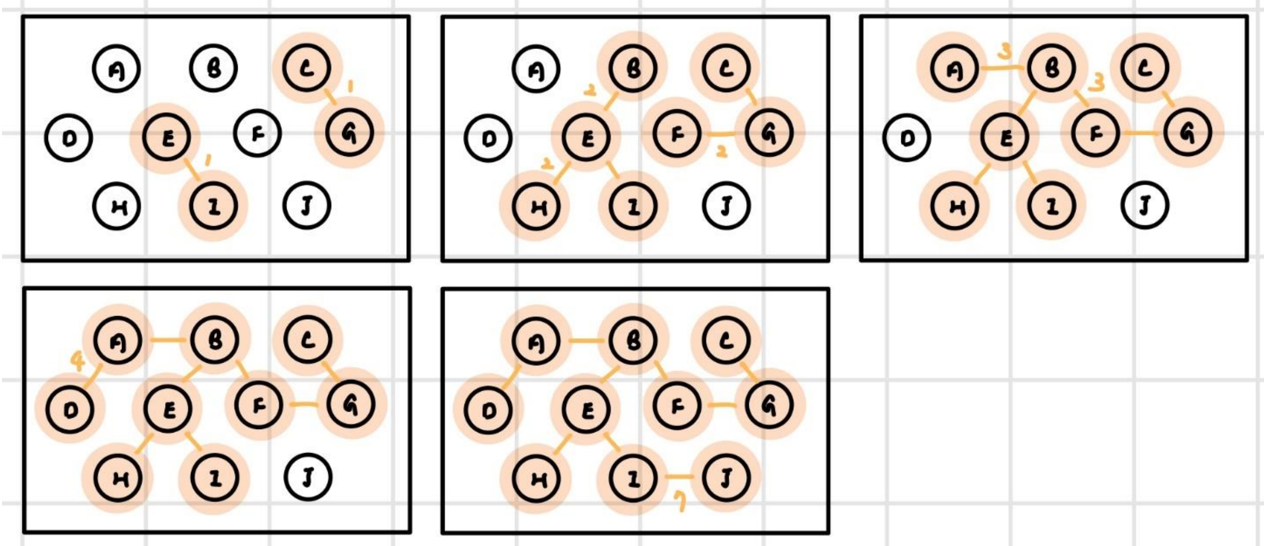
(b) Is this minimum spanning tree unique? Why?

A.

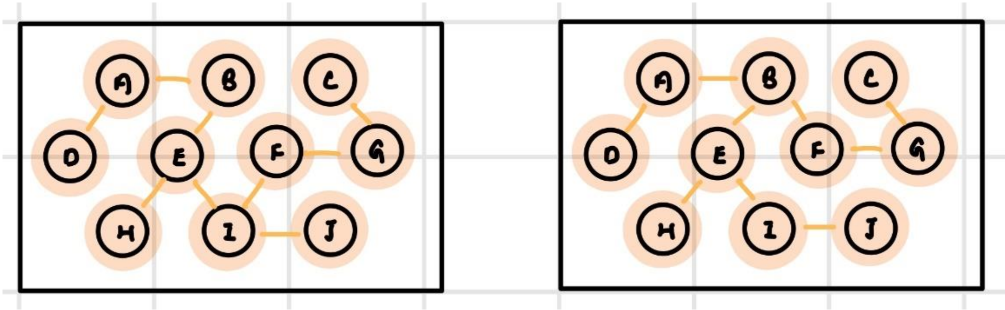
(a) Prim’s



Kruskal’s



(b) both can have these two kinds of minimum spanning tree



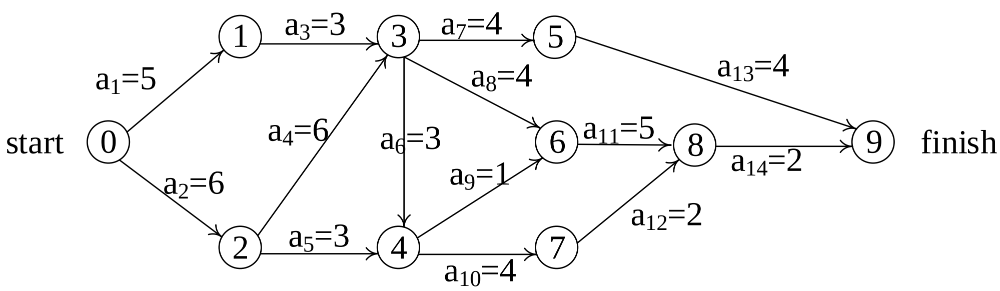
Because the path between (B,F) and (I,F) have the same value and if choose both it would have an loop which is unavailable, it can't be unique.

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

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

A. It has to be transitive and irreflexive, so suppose that it is transitive,
   then (0 · 1), (1 · 2) , (2 · 4) , (4 · 0) = (0 · 0) it is reflexive => it isn't 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?

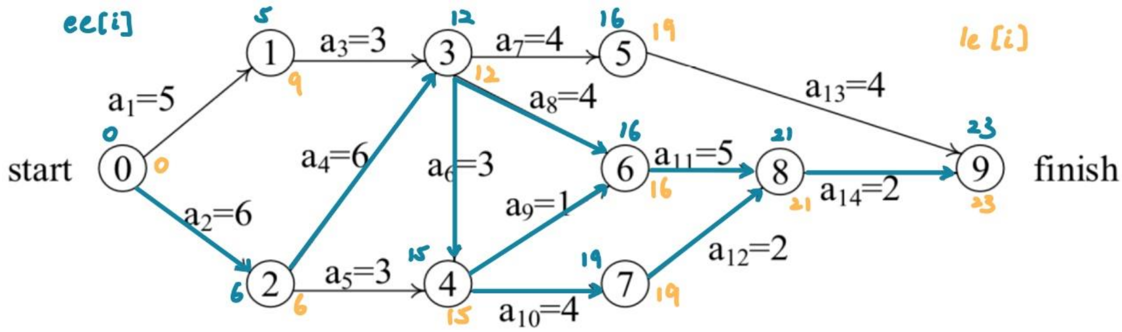


A.

(a) (b) (c)

Topological order : 0 2 1 3 4 7 6 8 5 9

Reverse topological order : 9 5 8 6 7 4 3 1 2 0



activity	Early time	Late time	slack	critical
	e(ai)	l(ai)	l(ai) - e(ai)	
a1	**0**	**9 - 5 = 4**	**4**	**No**
a2	**0**	**6 - 6 = 0**	**0**	**Yes**
a3	**5**	**12 – 3 = 9**	**4**	**No**
a4	**6**	**6**	**0**	**Yes**
a5	**6**	**12**	**6**	**No**
a6	**12**	**12**	**0**	**Yes**
a7	**12**	**15**	**3**	**No**
a8	**12**	**12**	**0**	**Yes**
a9	**15**	**15**	**0**	**Yes**
a10	**15**	**15**	**0**	**Yes**
a11	**16**	**16**	**0**	**Yes**
a12	**19**	**19**	**0**	**Yes**
a13	**16**	**19**	**3**	**No**
a14	**21**	**21**	**0**	**Yes**

The earliest time is 23

(d) we can see a_2, a_4, a_{14} have no other path, so speed-up these activity can reduce the time

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)

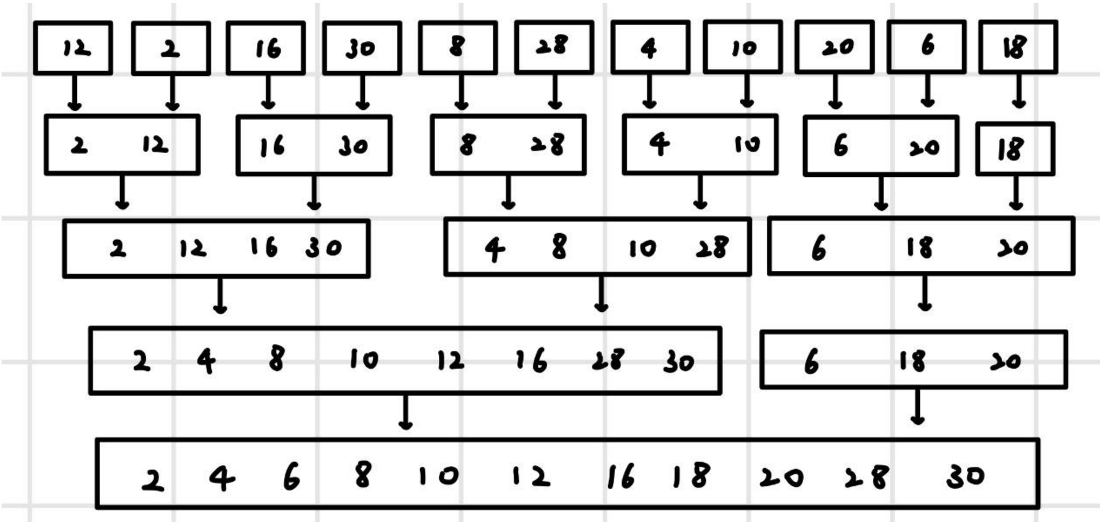
j	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]
1	12	2	16	30	8	28	4	10	20	6	18
2	2	12	16	30	8	28	4	10	20	6	18
3	2	12	16	30	8	28	4	10	20	6	18

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	18	16	20	28	30

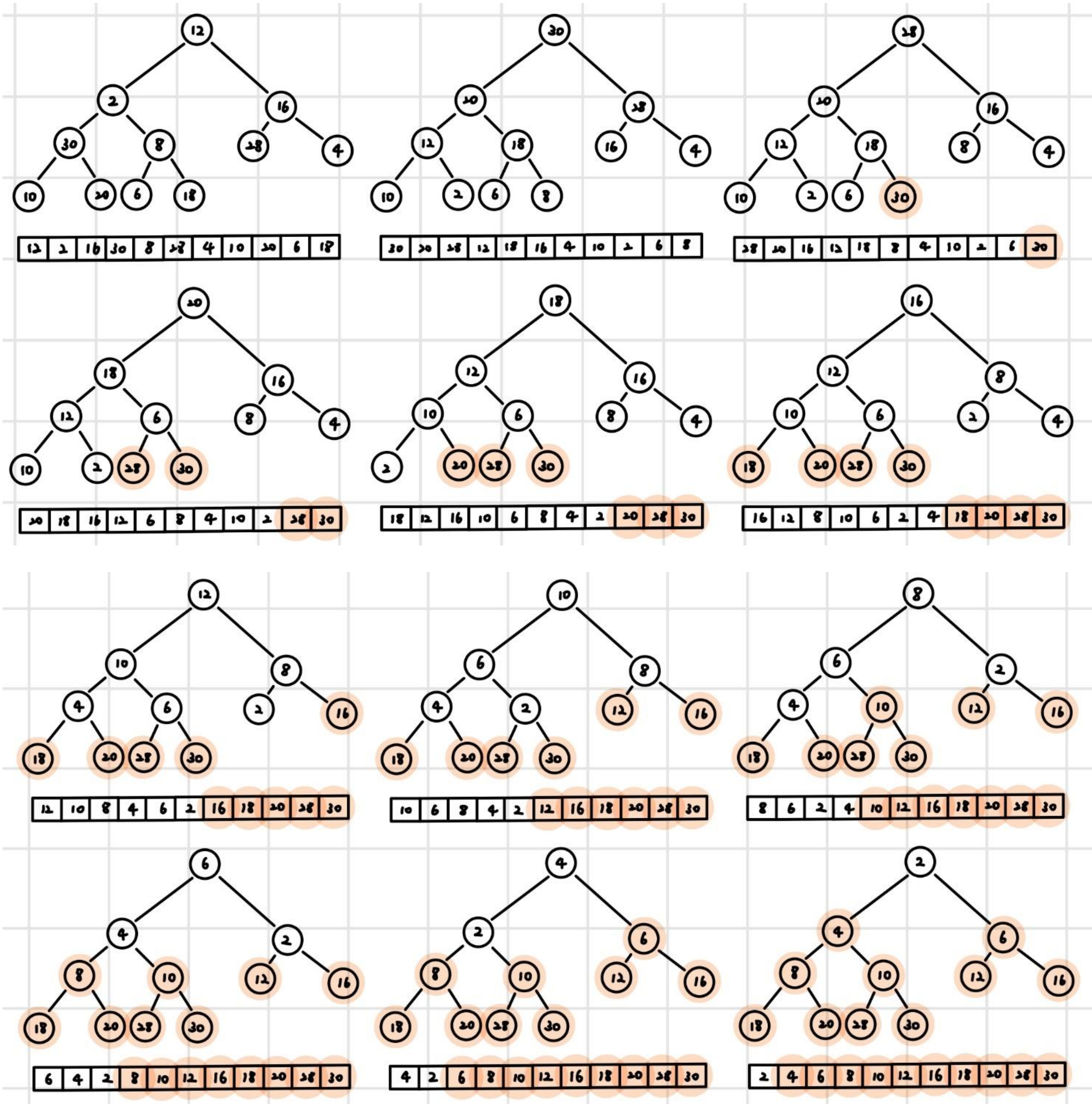
(b) Trace Program 7.6 QuickSort, use it on the list L, and draw a figure similar to Figure 7.1 Quick Sort example starting with the list L. Put your answer in the following table. (add necessary rows for your answer)

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
[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	[8	10]	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	[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		

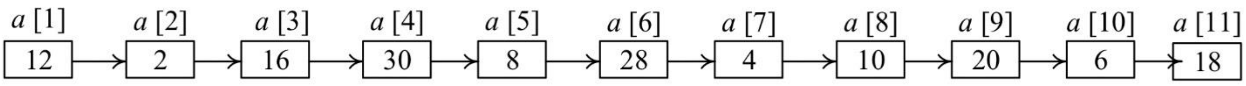
(c) Write the status of the list L at the end of each phase of MergeSort (Program 7.9), i.e., draw the Merge tree (similar to Figure 7.4 in textbook) of this problem.



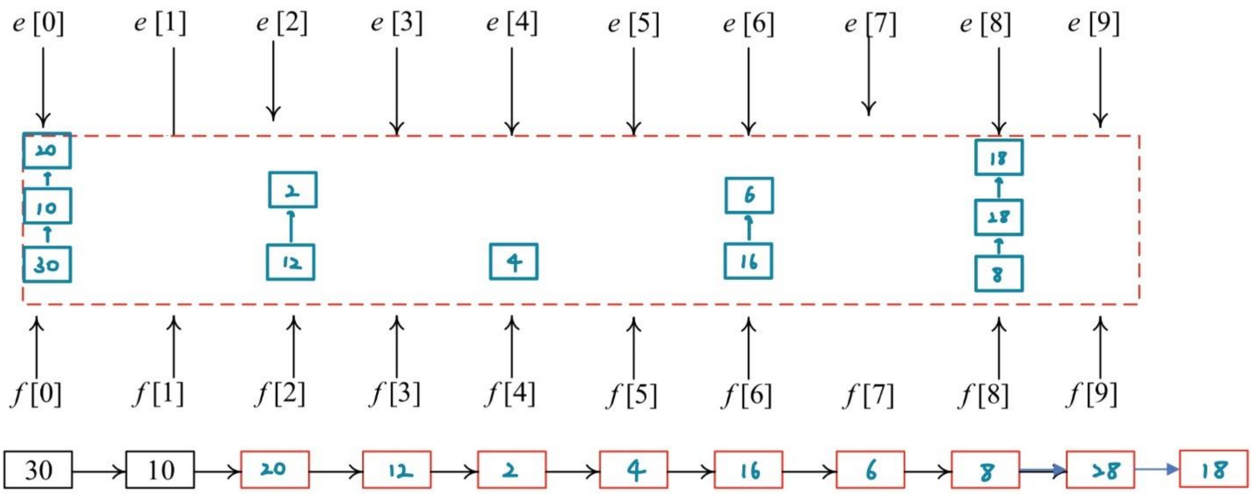
(d) Write the status of the list L at the end of the first **for** loop as well as at the end of the second **for** loop of HeapSort (Program 7.14), i.e., you need to draw the following trees for: 1) input array, 2) initial heap, and 9 more trees with heap size from 10 down to 2 with corresponding sorted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



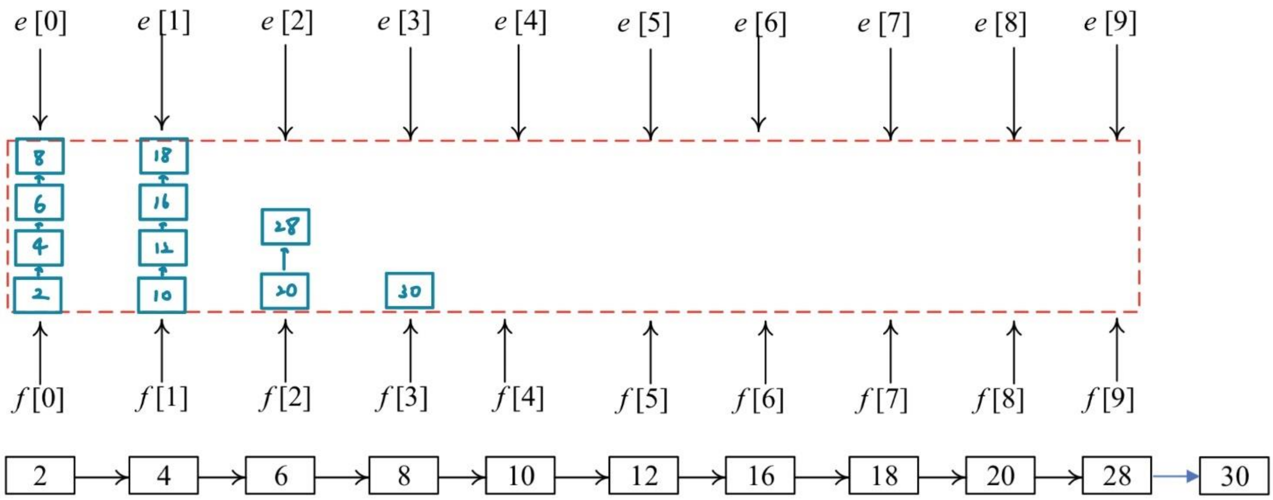
(e) Write the status of the list L at the end of each pass of RadixSort (Program 7.15), using r = 10. That is fill the missing parts (the node boxes with numbers and arrows between e[j] and f[j] enclosed by red dashed rectangle in (ii) and (iii) part of the following figure, and the missing numbers in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



(i) Initial input



(ii) First pass queues & resulting chain



(iii) Second pass queues & resulting chain

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.

A. (a)

one-way property : for a given c it is difficult to find a k such that h(k) = c

weak collision resistance : we have h(x) = c, it is difficult to find a y such that h(y) is also equals to c strong collision resistance : it is difficult to find a pair (x, y) such that h(x) = h(y)

(b)

For h(k) = k % 17, if h(k) = c, we can easily find k = 17n + c, so it doesn't satisfy one-way property

Since we know k = 17n + c, and h(x) = c, we can get y = 17x + c, let h(y) = c, so it doesn't satisfy weak collision resistance

Let h(x) = c = h(y), since we know x = 17n_1 + c, y = 17n_2 + c, just pick a different n we can find a pair(x, y) such that h(x) = h(y), so it doesn't satisfy 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 minimized when $h = \ln 2 * \frac{m}{u}$

A. $P(u) = e^{-u/n}\left(1 - e^{-uh/m}\right)^h = e^{-u/n}e^{\ln(1-e^{-uh/m})^h} = e^{-u/n}e^{h\ln(1-e^{-uh/m})}$

$$\frac{\partial}{\partial h}P = e^{-u/n}\left[\ln(1 - e^{-uh/m}) + h\frac{{}^{u}/_{m}\,e^{-uh/m}}{1 - e^{-uh/m}}\right]e^{h\ln(1-e^{-uh/m})}$$

And we want to find $\frac{\partial}{\partial h}P = 0$, but $e^{-u/n}e^{h\ln(1-e^{-uh/m})}$ can't be 0

So we find $\ln(1 - e^{-uh/m}) + h\frac{{}^{u}/_{m}e^{-uh/m}}{1-e^{-uh/m}} = 0$

⇨ $m\ln(1 - e^{-uh/m})(1 - e^{-\frac{uh}{m}}) + hue^{-uh/m} = 0$

Let $t = (1 - e^{-uh/m})$,

$$t = (1 - e^{-uh/m})$$

⇨ $e^{-uh/m} = 1 - t$

⇨ $-\frac{uh}{m} = \ln(1 - t)$

⇨ $h = -\frac{m}{u}\ln(1 - t)$

plug in $m\ln(1 - e^{-uh/m})(1 - e^{-\frac{uh}{m}}) + hue^{-uh/m} = 0$

⇨ $t\ln t = (1-t)\ln(1-t)$

We can guess t = $\frac{1}{2}$,

$$\Rightarrow \quad h = \frac{m}{u}\ln(2)$$