108-1Data Structure Final Exam

1. We use the adjacency matrix to represent the undirected graph. The zeros imply the corresponding edges do not exist, and the non-zero values represent the weight of the edges.

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

(1) Please draw this undirected graph. (2%)

(2) Which vertex has the largest edge degree? (2%)

(3) What is the cost of the minimum cost spanning tree in this graph? (3%)

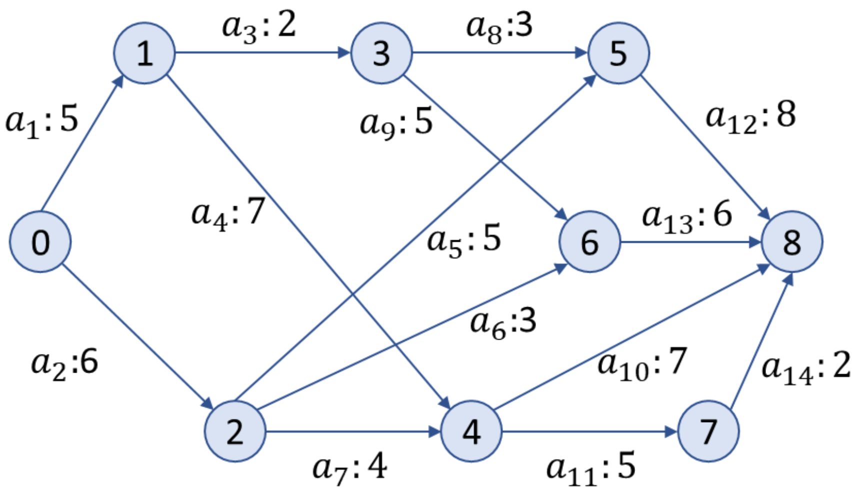
---

2. The AOE network as shown below, please answer the following questions.

(1) Please describe the AOE Networks? (2%)

(2) Which activities are critical? (2%)

(3) There is a single activity to speed up, and it would result in a reduction of the project length or not? If the answer is "yes", please list the activities. (3%)



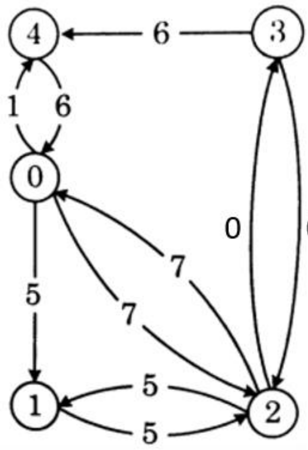
3. The height of this binary search tree is α and the number of nodes is β.
   (1) Space complexity of this binary search trees? (2%)
   (2) Time complexity of this binary search trees? (2%)

---

4. Let x be a node in a binary search tree. If y is a node in the left subtree of x, then key[y] ≤ key[x]. If y is a node in the right subtree of x, then key[x] ≤ key[y]. Suppose that we have an integer number between 1 and 10000 in a binary search tree and want to search for the number 2000. The following sequence is the sequence of nodes examined. Please give all the feasible ranges of the variable k in the sequence.
   1000, 5566, 2303, k, 1314, 1510, 2381, 2006. (4%)

---

5. Let $A^k[i][j]$ be the length of the shortest path from i to j going through no intermediate vertices of index greater than k. Please show the largest value of the non-infinity entries in matrix A^2 in the following graph. (4%)



---

6. Given a binary search tree, tree traversals have been defined: preorder, inorder, and postorder. It returns the relative position of a node in the corresponding traversal. Given the following preorder traversal of a binary search tree 8, 2, 1, 4, 6, 5, 16, 32, 24, 27. Please list the results of the other two traversals and draw the corresponding BST tree as well. (6%)

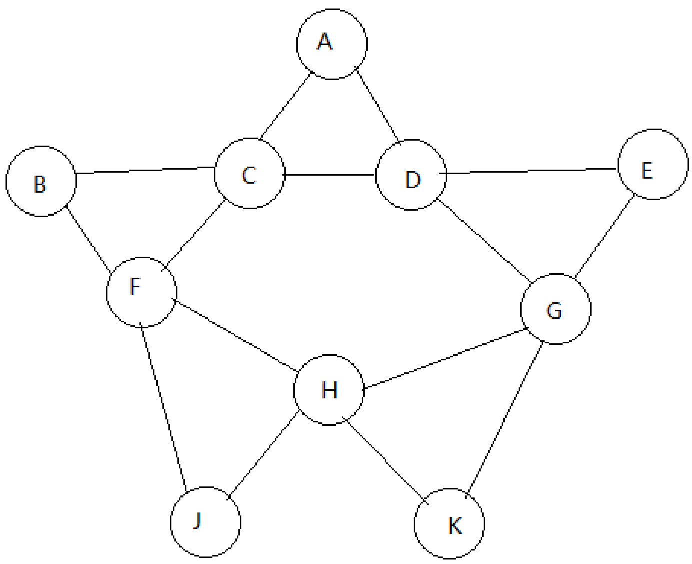
---

7. Suppose the worst case of a sort algorithm is that it needs the maximum number of data exchange. Give the number from 1 to 10, what conditions are the worst case when we use quick sort to arrange the numbers from small to large. Please prove your answers. (3%)

8. Suppose that we have the following key value: 7, 9, 16, 30, 49, 82, 5, 33, 31, 6, 2, 1. (2%)
   (1) Please draw the min heap tree after each value is inserted into the heap.
   (2) Please draw the min heap tree after deleting 9 and 49 from the min heap obtained in (1) above.

---

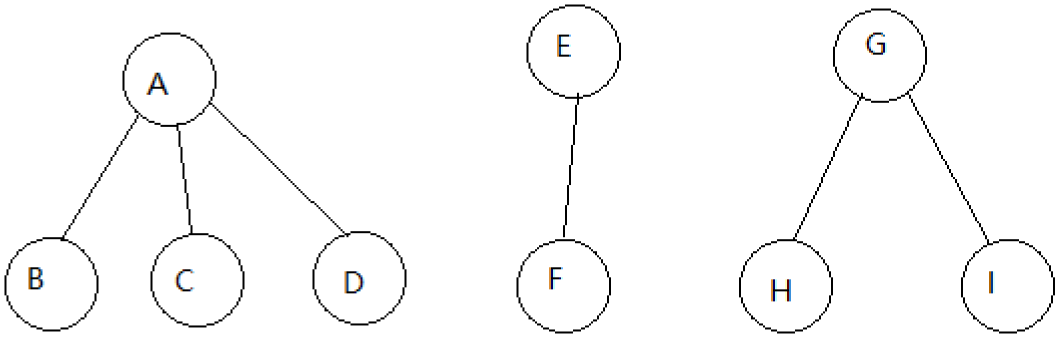
9. Given a connected undirected graph $G = (V, E)$ and $|V| > 1$. Let Path (i, j) denote the simple path between node i and node j. The length of Path (i, j) is denoted by L (i, j) which is defined as the number of edges in Path (i, j). Let BFS(i) and DFS(i) denote the outcome of visiting all nodes in a graph G starting from node i by breadth-first search and depth-first search respectively. Please answer the result based on the further given conditions:
   (1) If G is acyclic and V = {A, B, C, D, E, F} and BFS(A) = DFS(A), please give a possible example of G. (4%)
   (2) The graph as shown below, if $M = \{v \mid v \in V$ and BFS(v) = DFS(v)$\}$, then M =? (4%)
   (3) The graph as shown below, please give one example of the possible minimum cost spanning tree. (4%)



---

10. We use three kinds of algorithms (Kruskal, Prim, Sollin) to find out Minimum Cost Spanning Tree. When we find out Minimum Cost Spanning Tree, which algorithm(s) can form the forest in the process? (4%)

11. If $T_1, \ldots, T_n$ is a forest of trees, then the binary tree corresponds to this forest, denoted by B ($T_1, \ldots, T_n$). Please draw binary tree representation of the following forest. (4%)



---

12. In graph theory, the Seven Bridges of Königsberg is a historically notable problem in mathematics. An Eulerian path is a trail in a finite graph that visits every edge exactly once. An Eulerian circuit is an Eulerian trail that starts and ends on the same vertex. Please answer the following questions.

(1) Please describe the properties of the Eulerian Path. We assume that the graph is directed. (3%)

(2) Does Figure 1 have an Eulerian Path? If so, find one. If not, at least, how many edges do I need to add to form the Eulerian Path? (4%)

(3) Does Figure 2 have an Eulerian Circuit? If so, find one. If not, at least, how many edges do I need to add to form the Eulerian Circuit? (4%)

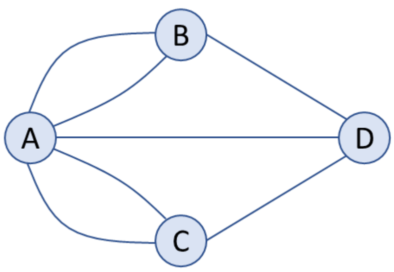


Figure 1

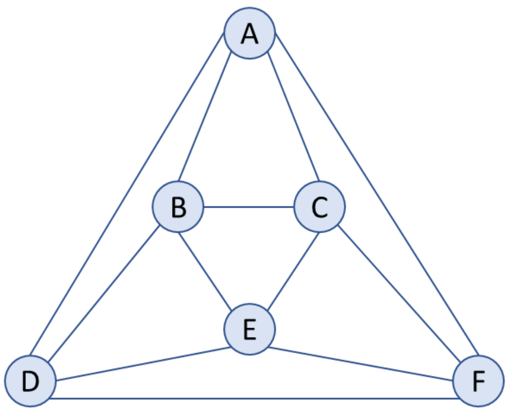


Figure 2

---

13. Please show the number of different binary trees, as there are ten nodes. (5%)

14. Node set of undirected graph G is G(V)= {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}, edge set is G(E)= {(0, 1), (1, 2), (1, 3), (2, 4), (3, 4), (3, 5), (5, 6), (5, 7), (6, 7), (7, 8), (7, 9)}.

(1) Please explain the definition of Biconnected Graph. (3%)

(2) Please list all articulation points of graph G. (4%)

(3) Please draw all biconnected component of graph G. Biconnected component must be represented as a subgraph of nodes and edges. (4%)

---

15. The following depictions are about the threaded binary tree. According to the following information to answer questions.

- The binary tree is represented by array:

position	1	2	3	4	5	6	7	8,9	10	11	12,13	14
node/data	A	B	C	D	E	--	F	--	G	H	--	I

- Data structures for threaded binary tree:

```
typedef struct threadedTree *threadedPointer;
typedef struct threadedTree {
    short int leftThread;
    threadedPointer leftChild;
    char data;
    threadedPointer rightChild;
    short int rightThread;
}
```

(1) Please explain the advantages and disadvantages of using a threaded binary tree. (3%)

(2) Which nodes have “dangling” in threaded binary trees? (4%)

(3) Please use the data structure to draw the threaded binary tree by memory representation. (4%)

(4) I want to insert *r* as the right child of *s* in a threaded binary tree. Please finish the following code using C. (5%)

```
threadedPointer insucc(threadedPointer tree){
    threadedPointer temp;
    temp = tree → rightChild;
    if(!tree → rightThread)
```

```
        while(!temp → leftThread)
            temp = temp → leftChild;
    return temp;
}
```

```
void insertRight (threadedPointer s, threadedPointer r)
{
    threadedPointer temp;
    r → rightChild = parent → rightChild;
    /*
    * finish the code
    */
    if (!r → rightThread)
    {
        temp= insucc(r);
        temp → leftChild = r;
    }
}
```