Faculty of Computers & AI,
Benha University

Student Name: ………………………………………………
Seat Number: ………………………………………………

Academic Year: 2019 / 2020

☐ First Semester

Program Name: 3rd Year Students (Security)

Course Name: Algorithms
Exam Date: …05/01/ 2020

Question No	Marks	Examiner
Q1		
Q2		
Q3		
Q4		
Q5		
Q6		
Q7		
Q8		
Total For written exam		
Class Work		
TOTAL MARKS		

Total Marks

Total Marks (in Letters)	……………………………………………………………………………………..		
Examination Committee	Examiner No. 1	Examiner No. 2	Examiner No. 3

Benha University
1st Term 2019/2020 -- Final
Class: 3rd Year Students (Security)
Subject: Algorithms
Course Code: FCS313

Faculty of Computers & AI
Date: 5/1/2020
Time: 3 hours
Total Marks: 50 Marks
Examiner(s): Dr. Karam Gouda

Answer the following questions:

Question No. 1 [8 Marks]

a- Define what does an "Algorithm" mean, its correctness ?

b- Why do we study the course "Design and Analysis of Computer Algorithms?

c- How many iterations of the following nested loops:

```
i=1;
loop (i <= N)
        j  = 1;
        loop (j <=  N)
                j = j * 2;
        i = i + 1;
```

d- How many iterations of the following nested loops:

```
i=1;
loop (i <= N)
        j  = 1;
        loop (j <=  i)
                j = j + 1;
        i = i + 1;
```

a- Algorithm is a sequence of unambiguous instructions for solving a problem, i.e., for obtaining a required output for any legitimate input in a finite amount of time. An algorithm is correct if it works correctly for all legitimate inputs (all instances). An input to an algorithm specifies an instance of the problem the algorithm solves.

b- Efficient algorithms lead to efficient programs. Efficient programs sell better. Efficient programs make better use of hardware. Programmers who write efficient programs are preferred.

c- #Iterations = #outer loop iterations x #inner loop iterations
The number of iterations in the inner loop is log 2 10 . In the above program code, the inner loop is controlled by an outer loop. The above formula must be multiplied by the number of times the outer loop executes, which is 10. this gives us 10 x $\log_2$ 10. In general, T(N) = N x $\log_2$ N

d- on. The number of iterations in the body of the inner loop is 1+2+3+4+……+8+9+10 = 55.
If we compute the average of this loop, it is 5.5 (55/10), which is the same as the number of iterations (10) plus 1 divided by 2. this can be written as (N+1)/2 Multiply the inner loop by the number of times the outer loop is executed gives the following formula: T(N) =N . (N+1)/2

Question No. 2 [9 Marks]

- State whether the following are **true** or **false**.

1- The same algorithm can be represented in only one way.
2- There may exist several algorithms for solving the same problem.
3- Algorithms for the same problem can be based on very different ideas and can solve the problem with dramatically different speeds.
4- Divide & Conquer design technique breaks the problem into overlapping sub-problems.
5- Greedy design technique repeatedly do what is the best now.
6- Dynamic Programming design technique breaks the problem into distinct sub-problems.
7- Time complexity computes how much memory is taken by the algorithm to run.
8- CPU speed can affect the running time of an algorithm.
9- In sorting algorithms, the number of items to be sorted detemines the input size of the algorithm.

1- F	2- T	3- T	4- F	5- T
6- F	7- F	8- T	9- T	

Question No. 3 [3 Marks]

Define the Greatest Common Divisor (GCD) of two nonnegative integers m and n. Write an algorithm in the pseudo-code format to calculate GCD(m, n) of m and n, using the following property GCD(m, n) = GCD(n, m mod n) ?

Gcd(m, n) is the largest integer that divides m , n with a remainder of zero where m and n are nonnegative integers.

Step 1: If n = 0, return m and stop; otherwise go to Step 2.
Step 2: Divide m by n and assign the value of the remainder to r.
Step 3: Assign the value of n to m and the value of r to n. Go to Step 1.

```
ALGORITHM Euclid(m, n)
//Computes gcd(m, n) by Euclid s algorithm
//Input: Two nonnegative, not both zero integers m and n
//Output: Greatest common divisor of m and n

While n ≠ 0 do
        r ← m mod n;
        m← n;
        n ← r;
return m;
```

Question No. 4 [5 Marks]

- Define the big oh (o) notation and directly apply this definition to prove that $f(n) = 3n + 7 = O(n)$.
- Prove that any binary tree of height h has at most 2_h leaves.

$T(n) = O(g(n))$: if $\exists c, n_0 : T\ n \leq c \cdot g(n),\ n \geq n_0$

$O(g(n))$: reads "order $g(n)$" or "Big-Oh $g(n)$"

Suppose that c = 4 then

$3n+7 \leq 4n$

$7 \leq n$

which is true for all when $n_0=7$

. . . . .

The depth s of a binary tree with n leaves is $s \geq \log_2 n$ (Equiv: $n \leq 2^h$, where n is the # of leaves and h is the hight)

•Initial case : For n =2, $s = \log_2 n = 1$

•Induction hypothesis : Trees of height s=k have at most $n \leq 2^k$ leaves so $s \geq \log_2 n$

•Induction step :

–Trees of height s = k +1 consists of one or two sub trees of size k . Denote the number of leaf nodes as n_1 and n_2

– According to induction hypothesis, $n_1 \leq 2_k$ and $n_2 \leq 2_k$.

– Number of leafs $n = n_1 + n_2 \leq 2^k + 2^k = 2^{k+1}$.

Hence, the hypothesis holds for k + 1

Question No. 5 [5 Marks]

Define: Complete and full binary trees, min-heap structure. Simulate the HEAPSORT algorithm on the data array A(1 : 8) = {20; 1; 9; 10; 5; 25; 13; 13}. Note: Sort in decreasing order.

Question No. 6 [9 Marks]

Write the pseudocode of Merge Sort and prove that its worst-case time is O(n lg n) by using Recursion tree method. Then simulate the Merge Sort algorithm on the data array A(1 : 10) = {2; 10; 1; 20; 15; 30; 1; 19; 25; 45}. (Sort in decreasing order).

```
MergeSort(A[i, j])
if i<j then
        k := (i+j)/2;                    O(1)
        MergeSort (A[i, k]);             T(N/2)
        MergeSort(A[k+1, j]);            T(N/2)
        Merge(A[i, j], k);               c·N
```

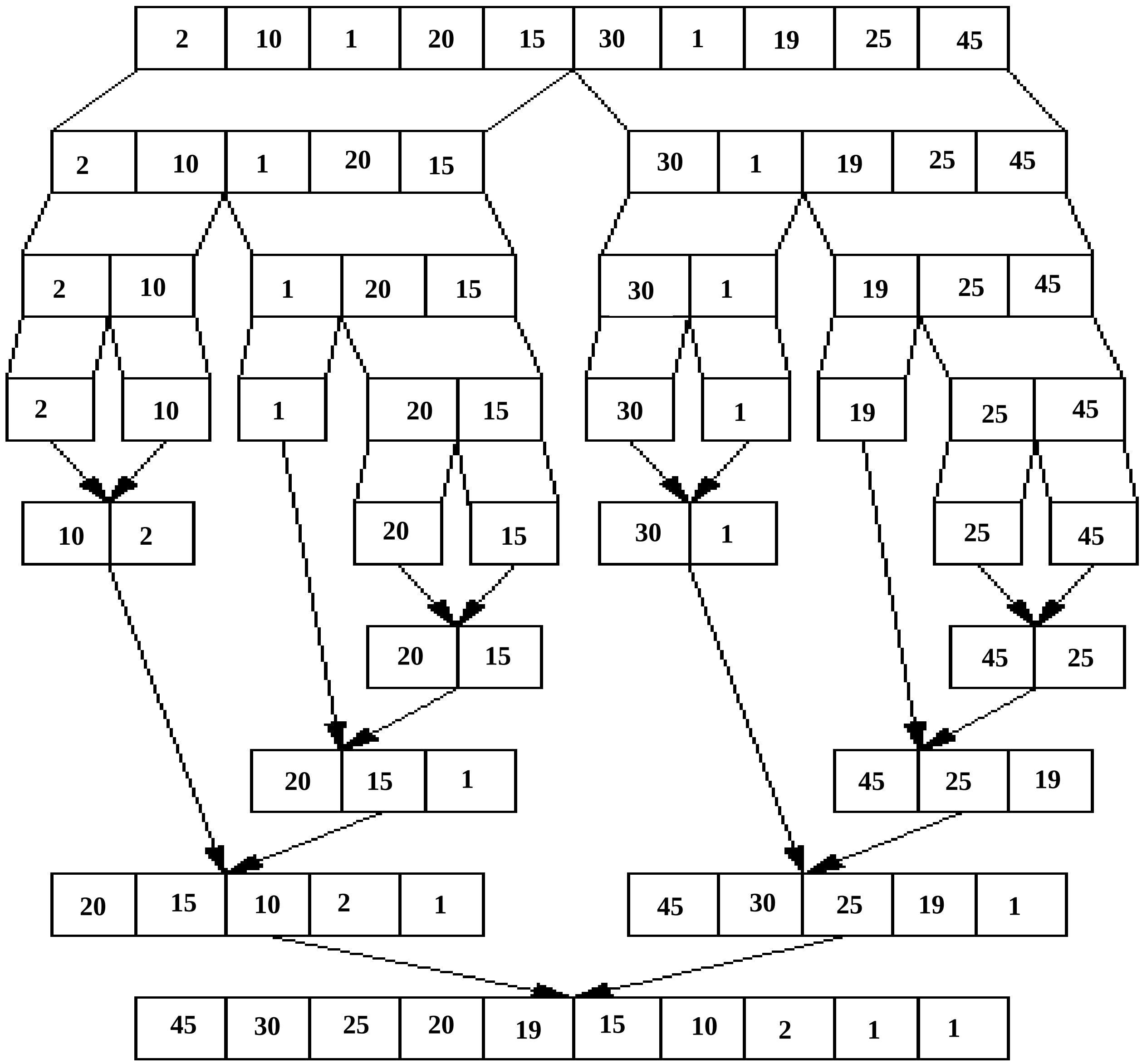
```
Merge(A[i, j], k
{
   l←i; m←k+1; t←i;
   while (l≤ k) or (m j)
        if l>k then                  B[t] t]←A[ m←m+1;
        else if m>j then             B[t] A[l]; l ←l+
        else if A[l]<A[m] then       B[t] t]←A[m]; m←m+
        else                         B[t] t]←A[l]; l←l+1;
        t←t+1
   for t← i to j do A[t] ←B[t]
}
```

T_{merge} (n) = O(7n+3) = O(n)
T(n) = 2T(n/2)+1+T_{merge} = 2T(n/2)+7n+4

Since we have b=c $\rightarrow$ O(n log n)

$$
\begin{aligned}
T(n) &= 2T(n/2)+7n+4 \\
&= 2(2T(n/2^2)+7n/2+4)+7n+4 \\
&= 2\cdot 2T(n/2^2)+7n+7+4+4\cdot 2 \\
&= 2^2(2T(n/2^3)+7(n/2^2)+4)+7n+7n+4+4\cdot 2 \\
&= 2^3T(n/2^3)+7n+7n+7n+4+4\cdot 2+4\cdot 2^2 \\
&= 2^3T(n/2^3)+3\cdot 7n+4\cdot(1+2+2^3) \\
&= 2^{\log_2 n}+\log_2 n\cdot 7n+4\sum_{i=0}^{\log n}2^i \\
&= 7n\cdot\log n+9n-4 \\
&\Rightarrow O(n\log n)
\end{aligned}
$$

. . . .


2 10 1 20 15 30 1 19 25 45
2 10 1 20 15
30 1 19 25 45
2 10
1 20 15
30 1
19 25 45
2
10
1
20 15
30
1
19
25 45
10 2
20
15
30 1
25
45
20 15
45 25
20 15 1
45 25 19
20 15 10 2 1
45 30 25 19 1
45 30 25 20 19 15 10 2 1 1

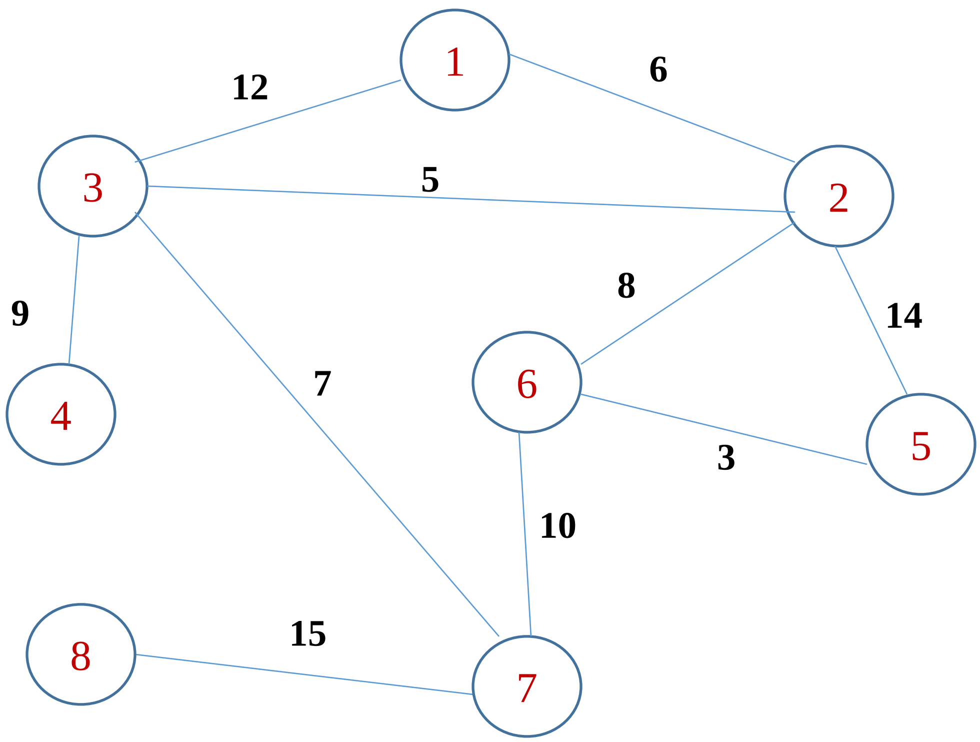
Question No. 7 [11 Marks]

1. Consider the following graph, given by its adjacency list, and followed by edge weights.

aList[1] = [2,3], weights: 6, 12
aList[2] = [1,3,5,6], weights: 6, 5, 14, 8
aList[3] = [1,2,4,7], weights: 12, 5, 9, 7
aList[4] = [3], weights: 9
aList[5] = [2,6], weights: 14, 3
aList[6] = [2,5,7], weights: 8, 3, 10
aList[7] = [3,6,8], weights: 7, 10, 15
aList[8] = [7], weights: 15

(a) Draw the equivalent graph. [2 Marks]
(b) Write its equivalent adjacency matrix. [1 Mark]
(c) What is the shortest path between node 3 and node 5. [1 Mark]
(d) What is the degree of node 1 and node 3. [1 Mark]
(e) List two spanning trees. [2 Marks]
(f) Is the graph connected? [1 Mark]
(g) Simulate the Kruskal algorithm on that graph to find a minimum spanning tree. [3 Marks]

a)



b)

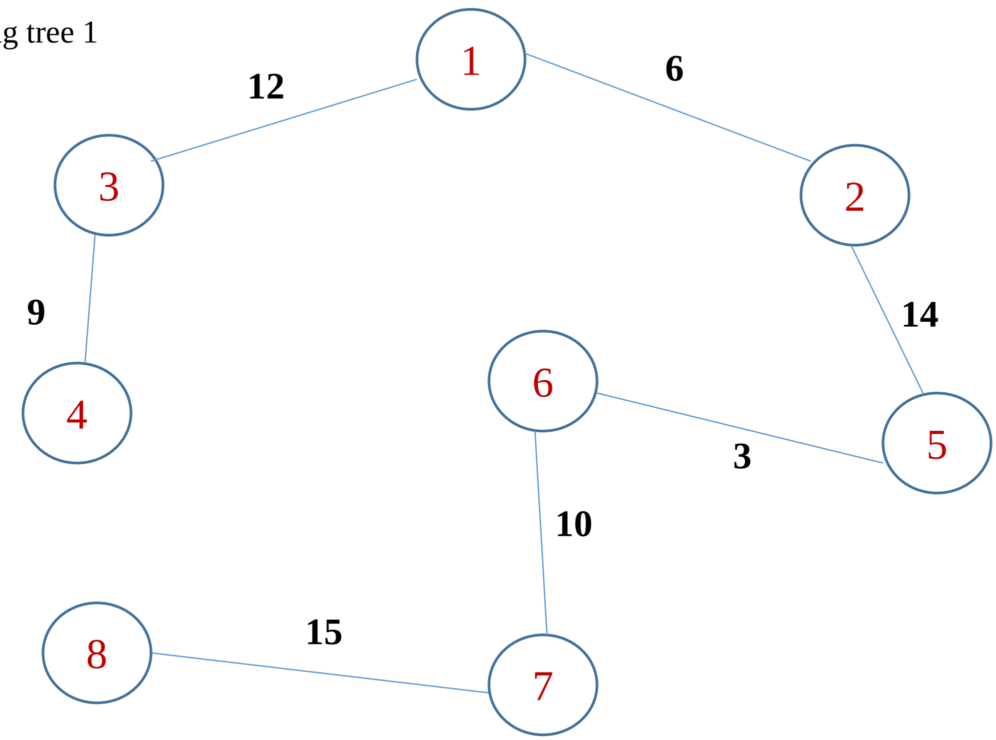
	1	2	3	4	5	6	7	8
1	∞	6	12	∞	∞	∞	∞	∞
2	6	∞	5	∞	14	8	∞	∞
3	12	5	∞	9	∞	∞	7	∞
4	∞	∞	9	∞	∞	∞	∞	∞
5	∞	14	∞	∞	∞	3	∞	∞
6	∞	8	∞	∞	3	∞	10	∞
7	∞	∞	7	∞	∞	10	∞	15
8	∞	∞	∞	∞	∞	∞	15	∞

c) 3 , 2 , 6 , 5
5 + 8 + 3= 16

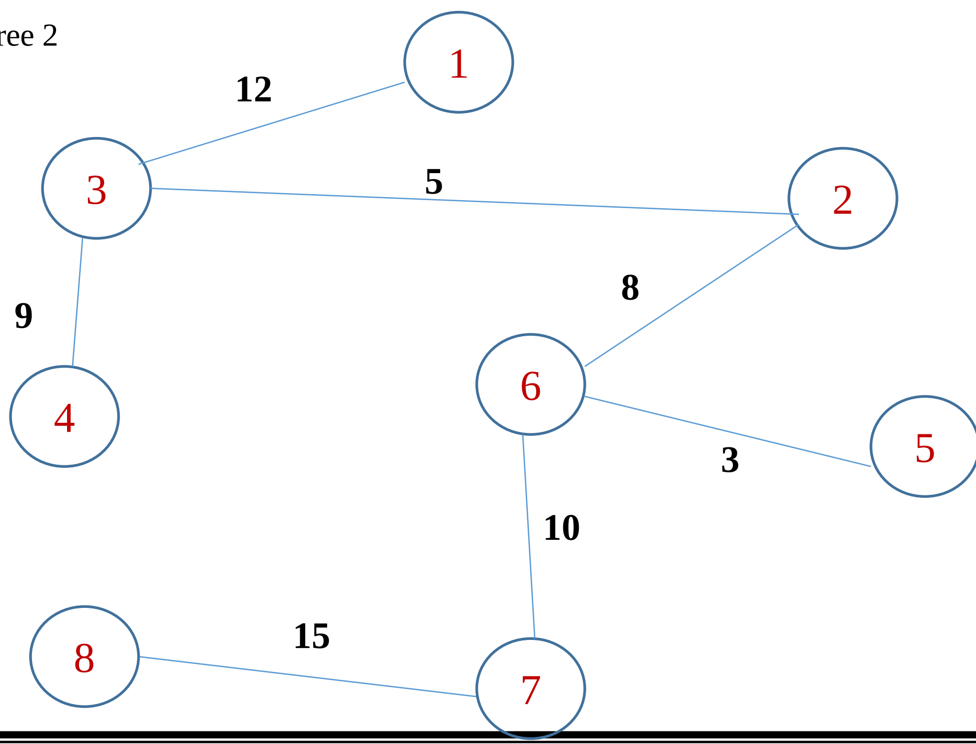
d) node 1: 2
node 3: 4

f) yes all nodes are connected with at least one node.

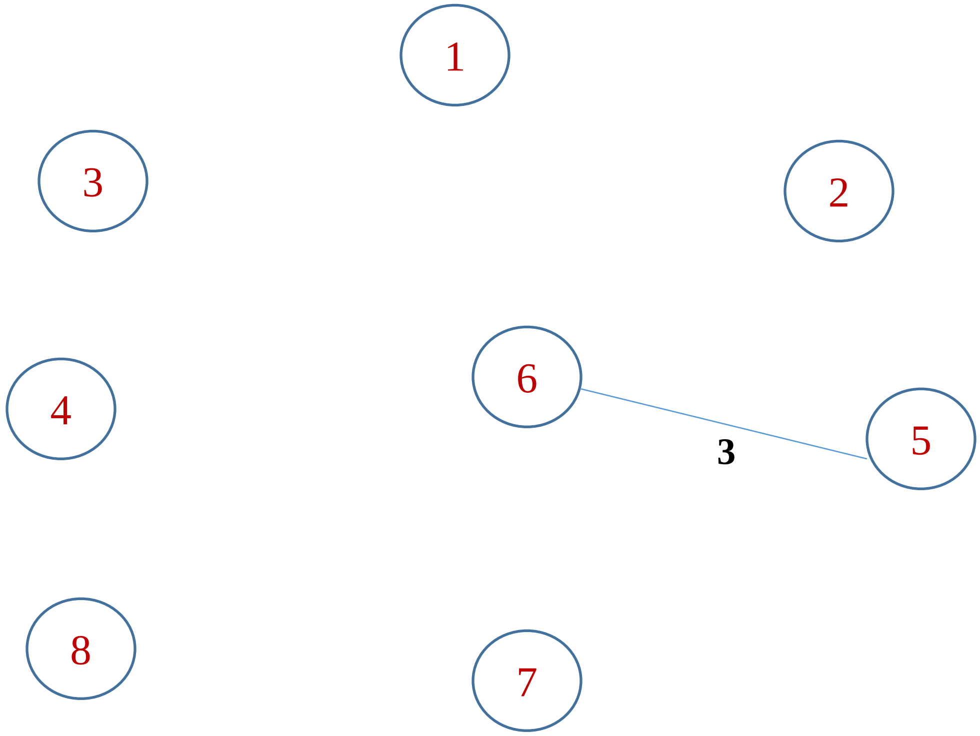
e) spanning tree 1

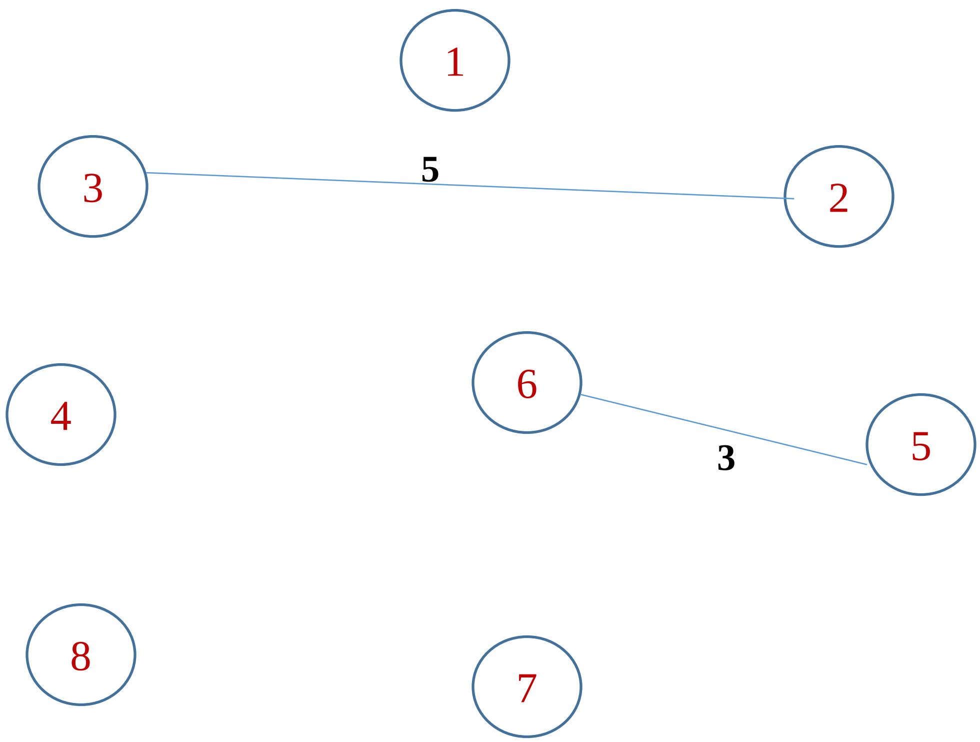


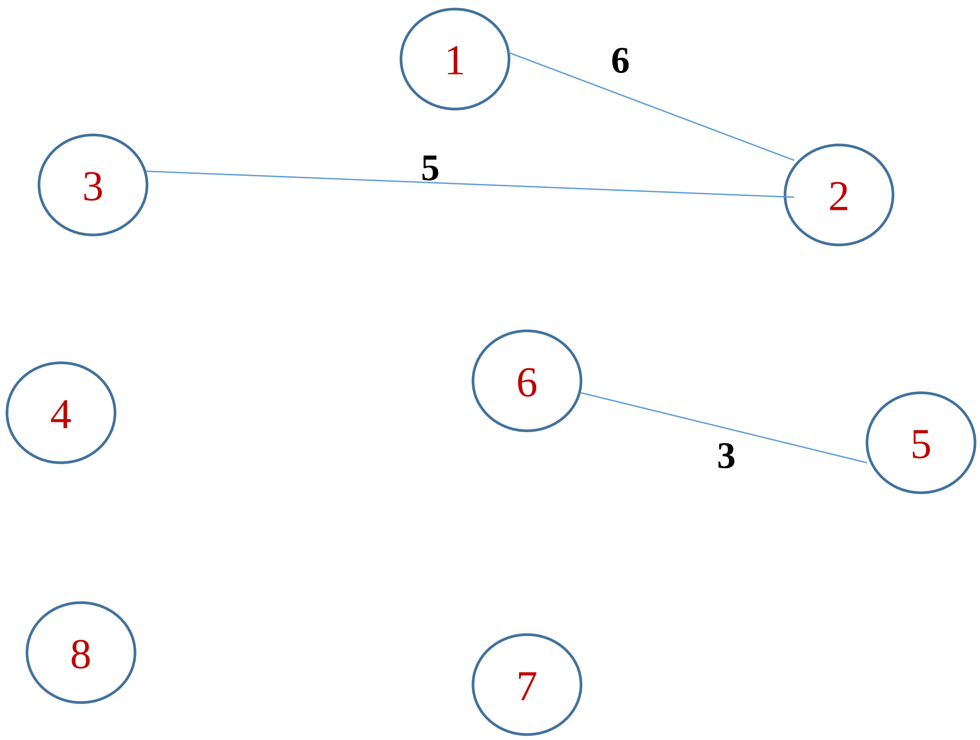
Spanning tree 2

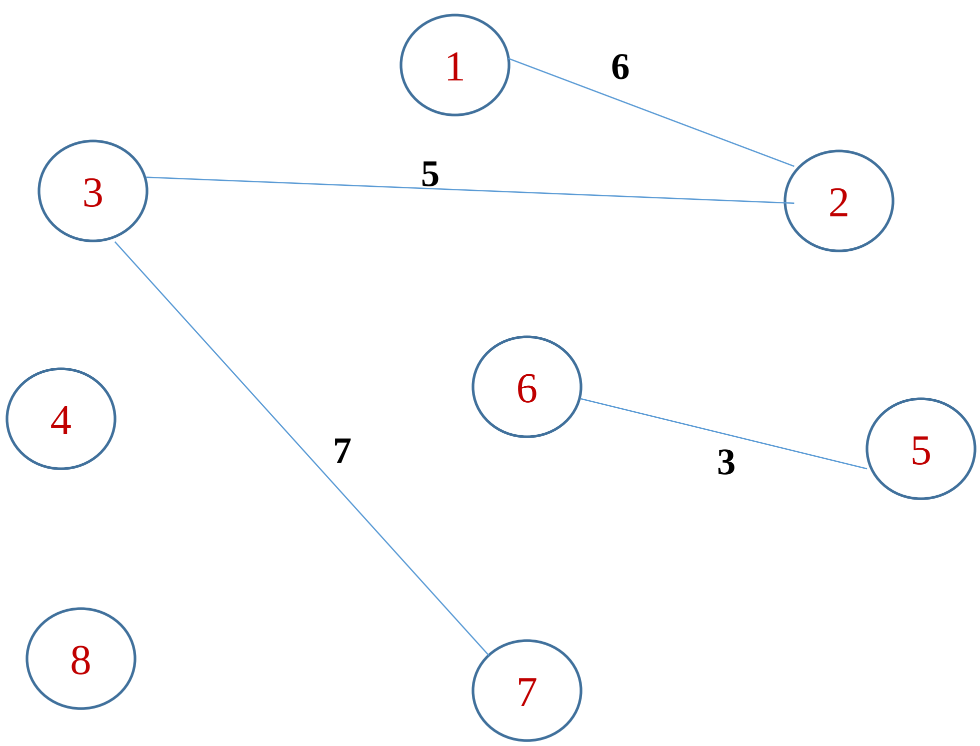


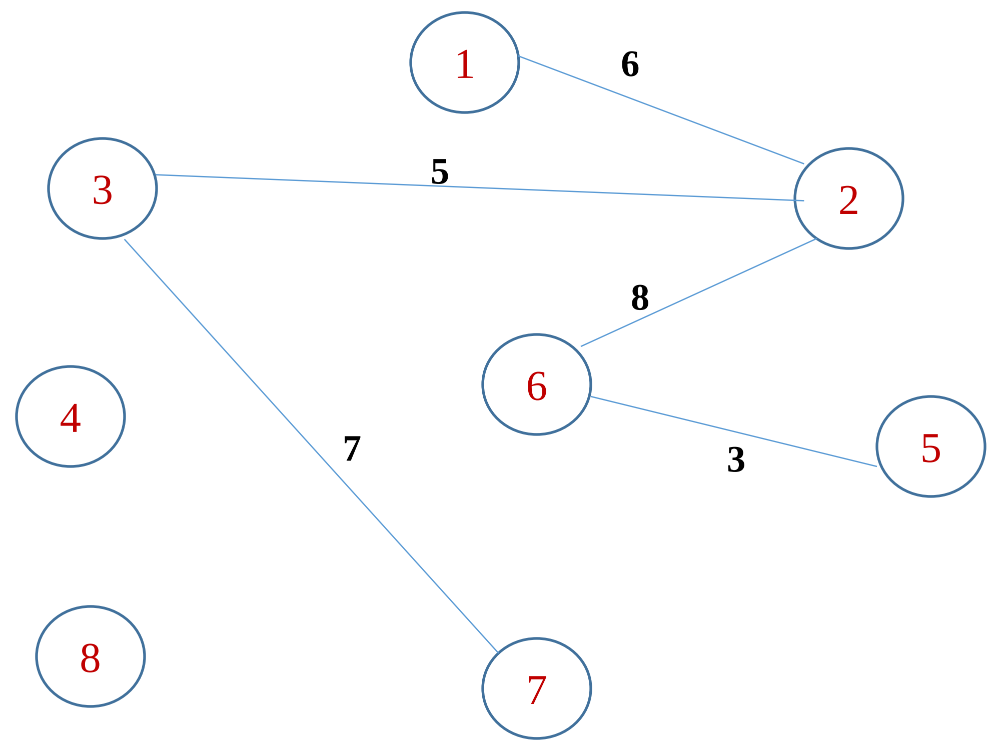
g)

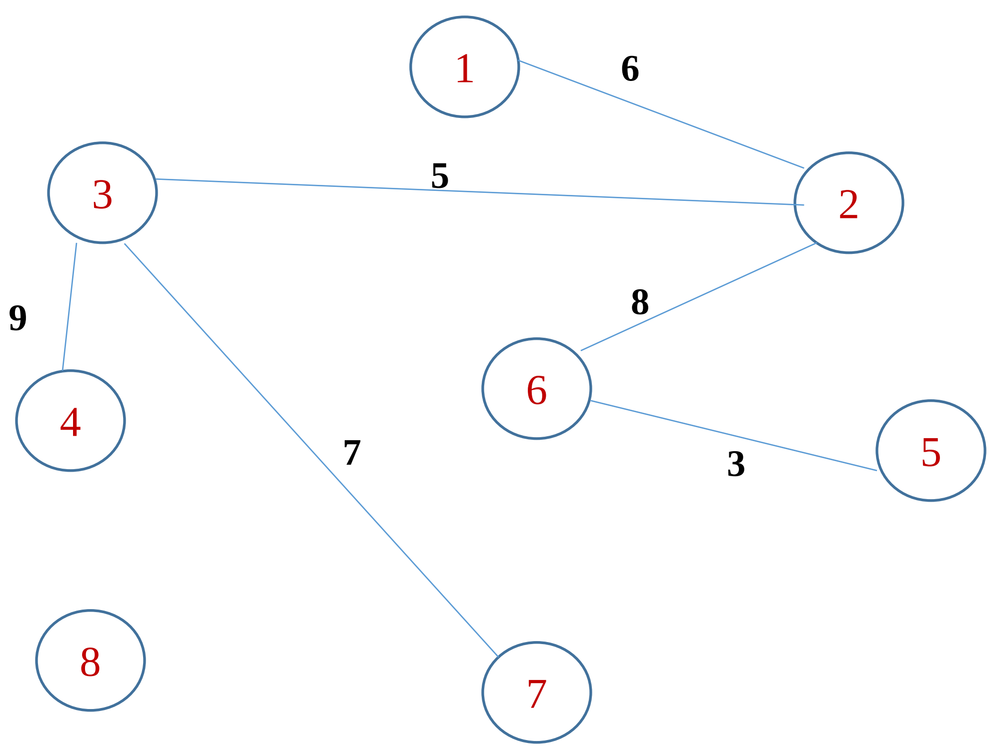


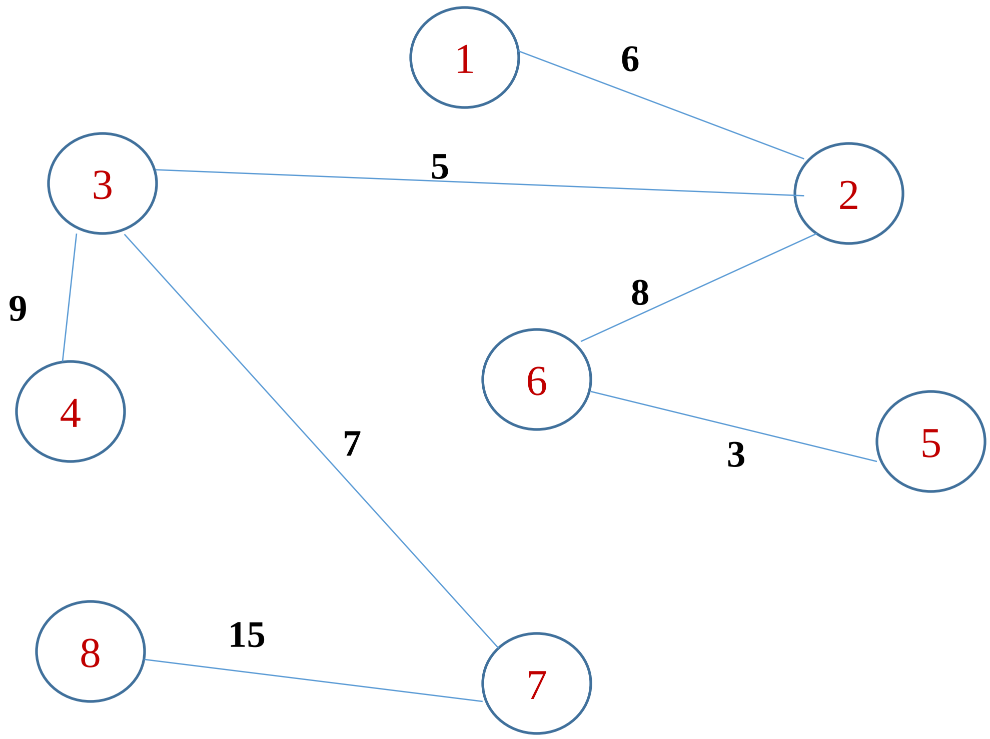



1
6
3
5
2
4
6
5
3
8
7


1
6
3
5
2
4
6
7
5
3
8
7


1
6
3
5
2
8
6
4
7
5
3
8
7


1
6
3
5
2
9
8
6
4
7
5
3
8
7



Total weight is 3+5+6+7+8+9+15= 53

GOOD LUCK,
Dr. Karam Gouda