**Name:** <Haitham Babbili>
Personal number: <19880803-4915>

**Question 2:**

**2.a** :

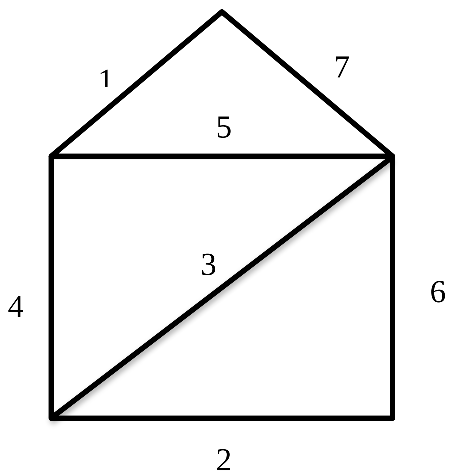
Assumptions

i. There are finite number of P vertices or nodes, and E edges
ii. Every node is activated infinitely often
iii. There is a shared memory or register that keeps in details of all successive iterations or executions
iv. The graph is considered as an undirected graph
V. The weight values of the graphs are from 1 to N, where N is a finite positive integer.
vi. A cycle or loop is not permitted in minimum spanning tree (MST)

For a minimum spanning tree, these are the set of legal executions

1. There is first a sort the graph edges according to their weights. This is done from the lowest weight to the highest
2. Addition of the edges in minimum spanning tree (MST) starts from the edge with the smallest weight up to the edge of the highest weight.
3. Since a loop or cycle is not permitted, only edges that do not form a cycle are considered in the MST calculations. Interest is on edges that connect disjointed nodes or vertices.
4. In a case where all the edge weights of the graph is the same, then there is a minimum spanning tree already

**2.b :**
Spanning-tree algorithm (BFS)
Ideas from Dolev book, on spanning-tree chapter 2.5 and lecture note on Fault-tolerant Algorithms for Computer Networks.




MST = 1+2+3+5 = 11

Also, some built on Kruskal algorithm

```
sort E in ascending order of weights; denote the result by (e1, . . . , e|E|)
T := \phi
for i = 1, . . . , |E| do
if T ∪ {ei} is a forest then
T := T ∪ {ei}
end if
end for
return T
```

**2.c :**
Given G = (P,E) graph

The asymptotic cost of memory is number of edges multiplied by logarithm of the number of vertices.
The shared memory needs to keep all weights and iteration results after minimum and maximum comparison

The largest message size a node can send is the largest weight value of the edges connected to it.
The maximum number of messages until the MST is guaranteed to be constructed is number of vertices.
The largest size of message a node can send is the depends on the result if MST is implemented in reverse. That is if we calculated as Maximum spanning tree, where we start executions from the largest weight.
Maximum number of communication rounds is O(P)
Maximum time, or time complexity is O(E log P)
Space Complexity is O(1)

**2.d** :
Given that there is a finite number of edges and vertices, and all processes are synchronized.
Then, the algorithm is self-stabilizing irrespective of which node the execution starts from because, the first process is sorting. Sorting is arrangement of the weights of all edges in the graph which is now in the shared memory being used. The end result of sorting is arrangement of weighted values from the minimum to the maximum value. The weighted values are always positive.
Then, there is a condition that there should be no loop in iterative edge connections based on their weight.
In a case where all the edge weights of the graph is the same, then there is a minimum spanning tree already