Answers to exercises of Chapter 2 (Graph Theory)

1. Different biological networks are modeled by different graphs. Which types of graphs are typically used to model the following networks: gene regulation networks, protein interaction networks, metabolic networks?

Gene regulation networks are typically represented by directed graphs. In this modeling genes are represented by vertices and transcriptional regulatory interactions between genes are modeled as directed edges. Protein interaction networks are commonly represented by undirected graphs. And metabolic networks can be shown as hyper-graphs because usually several substances react with each other to build other substances. As many graph algorithms cannot be directly applied to hyper-graphs these networks are also modeled by bipartite graphs (a structure generally used to represent hyper-graphs), see Section 2.3.2.

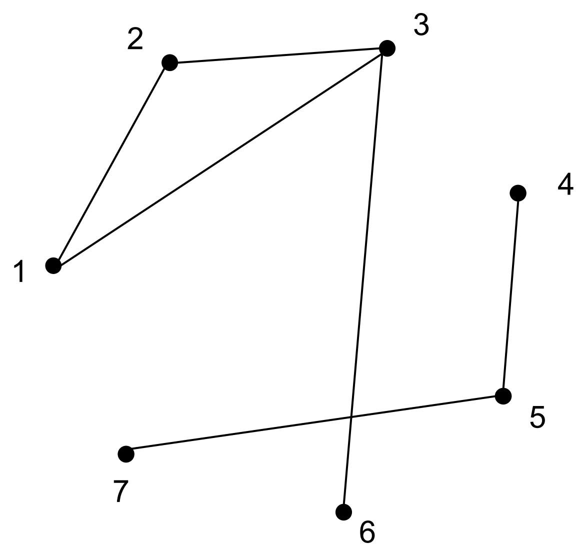
2. Consider the undirected graph $G = (V,E)$ shown in Figure 2.2 (right). For each vertex $v \in V$ do the following:

• Compute the degree of v.
• List all neighbors of v.
• Find paths to all other vertices which are in the same connected component as the vertex v.

• Degrees:
v_1: 2, v_2: 2, v_3: 3, v_4: 1, v_5: 2, v_6: 1, v_7: 1
• Neighbors:
v_1: v_2, v_3
v_2: v_1, v_3
v_3: v_1, v_2, v_6
v_4: v_5
v_5: v_4, v_7
v_6: v_3
v_7: v_5
• Paths to all other vertices which are in the same connected component as the vertex v.
For example, for vertex v_1:
path to v_2: $(v_1, e_{1,2}, v_2)$
path to v_3: $(v_1, e_{1,3}, v_3)$
path to v_6: $(v_1, e_{1,2}, v_2, e_{2,3}, v_3, e_{3,6}, v_6)$

3. For the graph shown in Figure 2.2 (right) find a different graphical representation of this graph and show how the graph is represented using an adjacency matrix and an adjacency list representation.

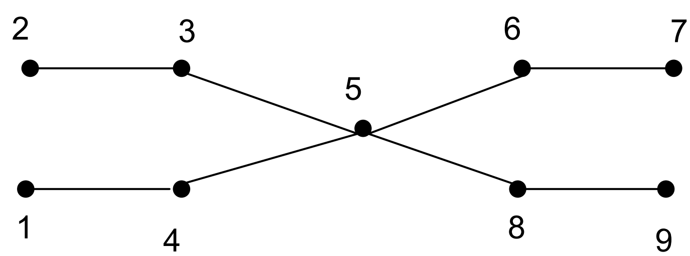
The graph can be drawn in many different ways, a different graphical representation is (all vertices are placed on a circle):



Adjacency matrix: see Figure 2.13
Adjacency list: see Figure 2.14

4. Take a graph with 9 vertices, 4 of them of degree two and 4 of degree one. Is this graph connected?

Yes, an example is

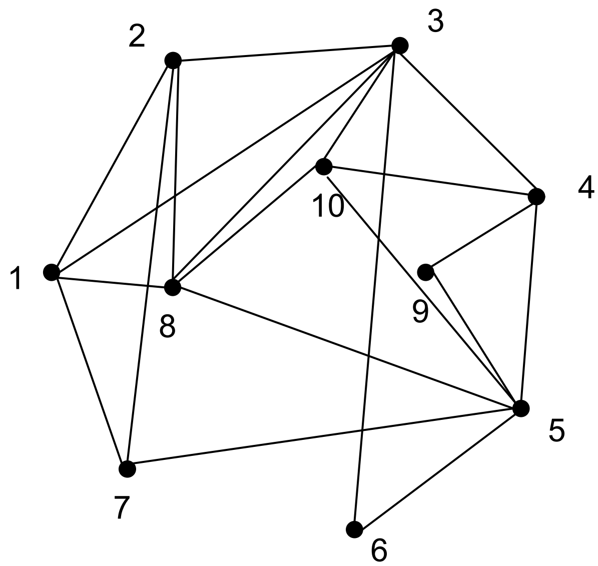


Note: If 5 vertices would be required to have degree two and 4 vertices to have degree one the answer would be no. In this case all vertices have a degree of 2 or smaller. Therefore one vertex can be connected to at most two other vertices. The 4 vertices with degree one can only be the start or end of paths, but cannot be connected to other vertices of a path. As four of such vertices exist we have two unconnected paths, therefore the graph is not connected.

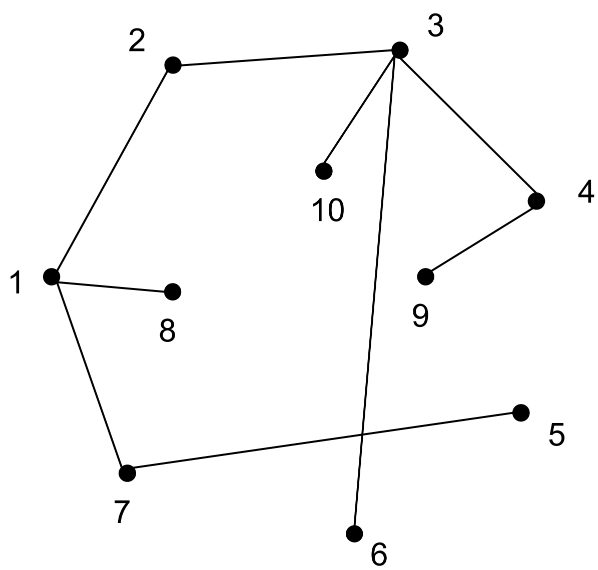
5. An undirected loop-free graph with n vertices has at most $n*(n-1)/2$ edges. Is this statement correct? Can you prove it?

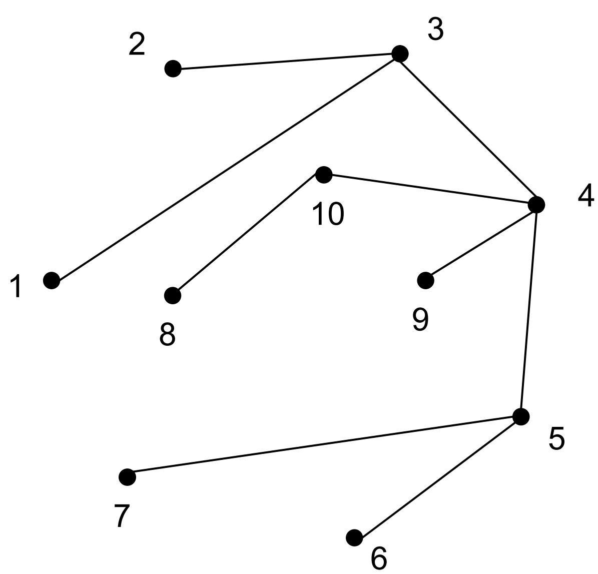
Yes, it is correct. Each of the n vertices can be at most connected to n-1 different vertices (loop-free), and each unordered pair of vertices can only be connected by one edge (therefore ½).

6. Draw an undirected, connected graph *G* with 10 vertices and 20 edges. Construct two different spanning trees of *G*.
A connected graph:

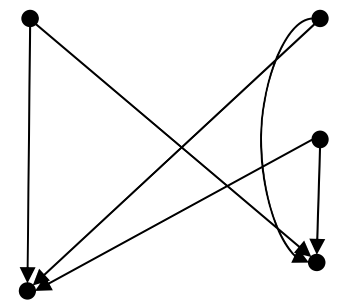


and two different spanning trees:





7. Metabolite networks can be constructed from metabolic networks modeled as bipartite graphs by removing all vertices representing reactions and connecting substrates (vertices with an outgoing edge to the reaction) with products (vertices with an incoming edge from the reaction) directly. Construct a metabolite network from the metabolic network given in Figure 2.11 (right).



8. Apply the algorithms DFS and BFS to traverse the graph in Figure 2.2 (right). Start with vertex 1, then apply the algorithms again starting with vertex 5.

Note: there are several different solutions (depending of the order neighbors are considered) and in this example the differences between DFS and BFS are only small.
DFS (start with vertex v_1): v_1, v_3, v_2, v_6, v_7, v_5, v_4
BFS (start with vertex v_1): v_1, v_3, v_2, v_6, v_7, v_5, v_4
DFS (start with vertex v_5): v_5, v_4, v_7, v_2, v_3, v_6, v_1
DFS (start with vertex v_5): v_5, v_4, v_7, v_2, v_3, v_1, v_6

9. An Eulerian path is a path $(v_0, e_1, v_1, e_2, v_2, \ldots, v_{k-1}, e_k, v_k)$ in an undirected graph which contains each edge of the graph exactly once. Write an algorithm to check whether an undirected graph $G = (V,E)$ has an Eulerian path.

Note that a connected undirected graph G contains an Eulerian path if every vertex of G has an even degree or if every vertex of G has an even degree except for 2 vertices and for these 2 vertices with odd degree, one must be the start vertex and the other the end vertex of the path.
Eulerian path algorithm (graph G)
Start with an empty stack S and an empty path/circuit p (Eulerian path).

```
if (G is unconnected)
   return no
else {
   if (all vertices have even degree) or (there are exactly 2 vertices with an odd degree)
      return yes
   else
      return no
}
```