**Assignment 12**

R-6.1 Draw a simple undirected graph G that has 12 vertices, 18 edges, and 3 connected components. Why would it be impossible to draw G with 3 connected components if G had 66 edges?

Answer:

A picture containing line, diagram

Description automatically generated

In an undirected graph with no self-loops and no parallel edges m ≤ n (n − 1)/2.

m = 66 = n(n -1)/2, this case says it exactly would be a complete graph. So, with 66 edges, we cannot separate it into any more connected components because that makes redundant edges. It is impossible to draw G with 3 connected components if G had 66 edges.

R-6.4 Bob loves foreign languages and wants to plan his course schedule to take the following nine language courses: LA15, LA16, LA22, LA31, LA32, LA126, LA127, LA141, and LA169. The course prerequisites are:

* LA15: (none)
* LA16: LA15
* LA22: (none)
* LA31: LA15
* LA32: LA16, LA31
* LA126: LA22, LA32
* LA127: LA16
* LA141: LA22, LA16
* LA169: LA32

Find a sequence of courses that allows Bob to satisfy all the prerequisites.

Answer: Sequence 🡨 {LA22, LA15, LA16, LA31, LA127, LA141, LA32, LA169, LA126}

A picture containing diagram, line, circle, plot

Description automatically generated

R-6.7 Would you use the adjacency list structure or the adjacency matrix structure in each of the following cases? Justify your choice.

1. The graph has 10,000 vertices and 20,000 edges, and it is important to use as little space as possible.

Answer: The adjacency list structure is preferable. Indeed, the adjacency matrix structure wastes a lot of space. It allocates entries for 100,000,000 edges while the graph has only 20,000 edges.

1. The graph has 10,000 vertices and 20,000,000 edges, and it is important to use as little space as possible.

Answer: the adjacency matrix, since this graph is dense, i.e., its number of edges is of the order of the maximum possible number of edges in that graph (which is about 50,000,000)

1. You need to answer the query areAdjacent as fast as possible, no matter how much space you use.

Answer: The adjacency matrix, since looking up the adjacency is done in constant time.