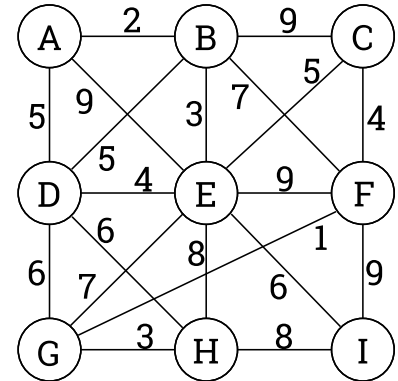


## Exercise 4

No obligatory X-exercise this week, but next week there will be an X-exercise on undirected graphs.

19. Simulate by drawing manually (a) Prim algorithm, (b) Kruskal algorithm for the attached tree to find minimum spanning tree. Use *A* as the starting vertex in Prim algorithm. It is easiest to draw by marking the vertices and edges of the spanning tree directly to the picture.



20. Draw following undirected graph so that bipartite property is apparent:  $G = \{ V = \{ a, b, c, d, e, f, g, h, i, j, k, l, m, n \}, E = \{ (a, d), (a, b), (b, c), (b, i), (c, d), (c, l), (e, f), (e, g), (e, j), (f, i), (f, n), (g, m), (h, l), (h, k), (j, m), (k, n) \} \}$ . Find the maximal matching using pen and paper. For matching the last vertices, using augmenting path technique helps a lot.
21. Mafiosi live (and die) by the following Code of Ethics: "Any enemy of my enemy is my friend." Don Corleone wants to invite all his friends to his daughter's wedding. He has drawn a graph of all his business acquaintances, where an edge means "these two persons are enemies of each other".
- How can he calculate who his friends are using this graph?
  - A person is a possible traitor, if he is both a friend and an enemy of Don Corleone. How can they be discovered using this graph (and handed to Luca Brasi for reprimands)?

In the following "write an algorithm" tasks, you should make a working Java method that gets as parameter input and possibly returns a new collection in accordance with the assignment, but does nothing else. Thus, for example, does not alter the input data (unless requested to do so) or print anything (at least in the final version). Please take the input generating main program from course web page. At exercise classes, we'll show your answers using projector, thus bring it with you by saving it to the cs.uef.fi server, somewhere else in the network, or a memory stick.

As the algorithms use trees and/or graphs, we need to use our data structure library which you can find from course www-page. There are instructions how to use it from command line, Eclipse, IntelliJ IDEA, and Netbeans. At cs.uef.fi you can use `trajc` and `traj`.

22. Write an algorithm that augments an unconnected undirected graph to a connected one using as few added edges as possible. Take skeleton from course www-page. You can use examples and previous exercises as bases. What is the time complexity of your algorithm?
- 23-24. Write an algorithm that augments an undirected graph to 2-connected by adding edges. You can assume graph to be 1-connected. Use the articulation point algorithm from course www-page as a basis and add edges between (tree) neighbours of cut vertices. The algorithm marks tree edges with black colour. At easiest, modify `cutVertices` method. Note, that you can repeat search of cut vertices to check that the adds you made were sufficient. No need to get optimal (minimum) amount of added edges, but try to keep amount of added edges reasonable.