ADFOCS 2003

Graph Data Types and Structures

Exercise 1.1:
Copy directory `/LEDA/ADFOCS/exercise-1` to your home directory. It contains a file `acyclic.c` with the source code of the ACYCLIC program. Compile and link it by calling `make acyclic`. This will create the executable `acyclic`. Run this program and learn how to use GraphWin to create and edit a graph and to apply the ACYCLIC algorithm (by pressing the *done* button). Quit the program by clicking the *exit* button in the *File* menu. You should also try to use the generators and layout tool of GraphWin.

Exercise 1.2:
Write new versions of the acyclic program by copying `acyclic.c` to different files and changing the source code. Use `make prog` to compile and link file `prog.c`.

1. Write a variant of the ACYCLIC algorithm that does not modify the graph (and hence does not require to work on a local copy).
   Hint: Simulate the deletion of edges by maintaining the current indegrees of all nodes in a node-array. Then the deletion of an edge (v, w) can be realized by simply reducing the current indegree of w by one.

2. Extend the program in such a way that it shows a cycle if the graph is not acyclic. Think about how you can traverse the remaining part of the graph to find the edges of a circle. Then use `gw.set_color(e,red)` to display all edges of the cycle in red (or use a different color).

3. Use the ACYCLIC algorithm as a basis to write a program that computes a topological ordering of a graph. A topological ordering is a numbering of the nodes such that for every edge (v, w) the number of v is smaller than the number of w. Try to visualize the numbering e.g. by assigning labels to the nodes of the graph `gw.set_label(v,string("..."))`.

4. Write a program that removes edges from a graph to make it acyclic. Try to remove not too many edges.