Rainer Feldmann

Paderborn, Apr, 13th 1:00 pm
Return until Apr, 20th 1:00 pm
to office F2.401

Home work for

Fundamental Algorithms

SS 2007

Sheet 1

Exercise 1: Prove or disprove:
If there is a path from a node u to a node v in a directed graph G such that after a depth first search on G $d[u] < d[v]$, then v is a descendant of u in the DFS forest produced.

Exercise 2: A vertex cover of an undirected graph $G = (V, E)$ is a subset $C \subset V$ such that for each $\{u, v\} \in E$ we have $u \in C$ or $v \in C$.

a) Develop an efficient algorithm that, for a given undirected connected graph G, outputs a vertex cover N for G.

b) Let G be any undirected connected graph, let N be the vertex cover computed by your algorithm from a). Let C be any vertex cover of G.
Either show that $|N| \leq 2 \cdot |C|$ or give a counterexample.

Exercise 3: (Programming task: to be returned by email until April 27th)
Write a program that

a) reads a directed graph form an input file, with the input specification as given below,

b) represents the graph internally by adjacency lists,

c) checks using DFS whether the graph is cycle free,

d) checks using DFS whether the graph is strongly connected,

d) outputs the result according to the output specification given below.

To complete this task you must send by email to obelix@upb.de

1) The source code of your program

2) The command to compile your program (this must be possible on computer mistel.cs.upb.de (Linux system))

3) The command to start your program. The input file name must be given as argument

Input specification:

<comment line>*
n = <int>
m = <int>
<int> : <list of int>

where <comment line> is a line starting with #, and <list of int> is a list of nonnegative integers separated by a blank (the adjacency lists). An example is given below:

```
# This is a graph with 3 nodes, 5 edges
n = 3
m = 5
0 : 1 2
1 : 2
2 : 1 0
```

Output specification:

Output should consist of exactly two lines of the form

cycle-free $< \{0, 1\} >$
s-connected $< \{0, 1\} >$

where 0 means that the graph is not cycle-free (or strongly connected) and 1 means that it is. An example is given below:

```
cycle-free 0
s-connected 1
```