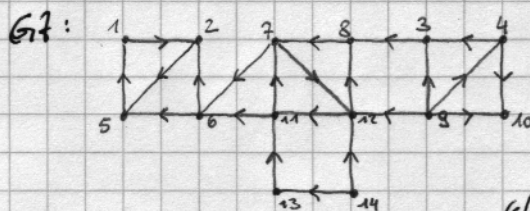


20.05.2007

176 (ohne Reduktion), 182 (ohne  $A^2$ ), 210

TEST!

176 Man bestimme die starken Zusammenhangskomponenten des Graphen  $G_7$ !



(starke) Zusammenhangskomponente:

starke Zusammenhangskomponenten:

maximaler stark zusammenhängender Teilgraph

• 1-2-5

stark zusammenhängend:

• 7-12-11

jeder Knoten ist von jedem anderen Knoten erreichbar

• 7-12-8

$v, w \in V$ :  $w$  von  $v$  erreichbar  $\Rightarrow$  es existiert ein

Pfad: 6, 3, 4, 9, 10, 13, 14

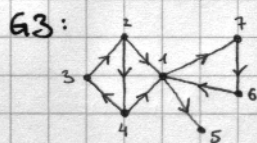
Weg/Bahn von  $w$  nach  $v$

Bahn: Knotenfolge, wo kein Knoten und keine Kante mehrfach durchlaufen wird.

Kantenfolge:  $e_1, e_2, \dots, e_n$

Aufgangsknoten von  $e_{j+1}$  = Endknoten von  $e_j$  ( $1 \leq j \leq n$ )

182 Man bestimme die Adjazenzmatrix  $A_{G_3}$ , sowie  $A_{G_3}^2$



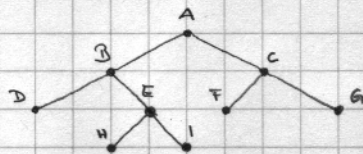
$$A_{G_3} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

2.10 Zum „Abarbeiten“ der Knoten eines Binärbaumes verwendet man gerne rekursive Algorithmen, die in wohldefinierter Reihenfolge die folgenden Schritte ausführen:

- (1) Bearbeite den aktuellen Knoten
- (2) Gehe zur Wurzel des linken Nachfolgerbaums des aktuellen Knotens
- (3) Gehe zur Wurzel des rechten Nachfolgerbaums des aktuellen Knotens

Am Beginn steht man bei der Wurzel des Gesamtbaums. Führt man die genannten Schritte (1) bis (3) rekursiv in der angegebenen Reihenfolge aus, so spricht man von „Preordertraversierung“. Beim unterstehen den Baum werden die Knoten also in folgender Reihenfolge bearbeitet:

A, B, D, E, H, I, C, F, G. Wie ändert sich diese Reihenfolge, wenn man im Algorithmus jeweils die Abfolge (2)(1)(3) nimmt („Inordertraversierung“), wie wenn man die Abfolge (2)(3)(1) wählt („Postordertraversierung“)?



Inordertravers.: D, B, H, E, I, A, F, C, G

Postordertravers.: D, H, I, E, B, A, F, G, C