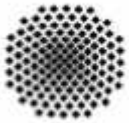
Universität Stuttgart
Fakultät 5, Institut FMI
EfidI 1 Zusatzkurs



WS 2005/2006
Wolfgang Schmid
20. Dezember 2005

Aufgabenblatt 8

Besprechung am Dienstag, den 10. Januar 2006, 8:00 Uhr V38.01

Aufgabe 1: (Bäume)

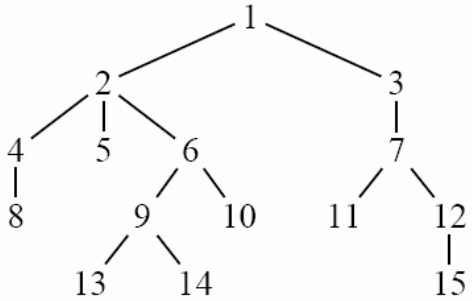
Definition Baum

Ein (gerichteter) Graph G = (*V, E*) heißt **Baum**, wenn

1. *G* keine geschlossenen Wege besitzt (azyklischer Graph).
2. Es einen ausgezeichneten Knoten *w* gibt, der nicht Endknoten einer Kante ist. *w* heißt **Wurzel**.
3. Jeder Knoten ungleich *w* Endknoten von genau einer Kante ist.
4. Zu jedem Knoten *v* ein Weg von *w* nach *v* existiert.

Sei eine Kante (*u,v*) in einem Baum gegeben, dann heißt *u* **Vater** von *v* und *v* **Sohn** von *u*. Ein Knoten ohne Söhne heißt **Blatt**. Ein Baum, bei dem jeder Knoten maximal zwei Söhne besitzt, heißt **Binärbaum.** Die Söhne von *v* heißen *v*.left und *v*.right.

Gegeben sei die grafische Darstellung eines Baums (mit Wurzel 1):



Geben Sie den Baum als Graph (*V,E*) an. Geben Sie die Tiefe des Baums an. Geben Sie die Blätter, und die inneren Knoten des Baums an. Geben Sie für jedes Blatt einen Pfad an, der von der Wurzel zu führt. Zeichnen Sie alle Teilbäume des Baums, die den Knoten 7 als Wurzel haben.

Aufgabe 2 (Binärbäume)

Unter der Höhe $h(b)$ eines (binären) Baumes verstehen wir folgendes:

> ein leerer Baum besitze die Höhe -1
> ein Baum b mit Wurzel w, linkem Unterbaum l und rechtem Unterbaum r besitzt die Höhe $h(b)=1+\max(h(l),h(r))$.

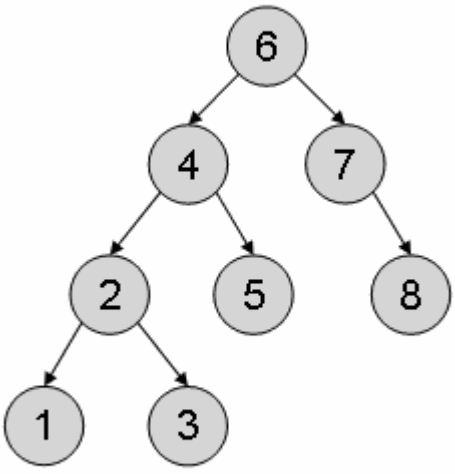
Schreiben Sie eine **function hoehe(*t* : tree) return integer**, die die Höhe des Baumes *t* nach obiger Definition berechnet.

Unter der Tiefe eines Knotens *k* im (binären) Baum *b* verstehen wir die Anzahl der Kanten, die man minimal durchlaufen muss, um von der Wurzel zum Knoten zu gelangen. Die Wurzel eines Baumes hat danach die Tiefe 0. Unter einer Schicht eines Baumes verstehen wir die Menge der Knoten gleicher Tiefe. Unter der Dicke eines Baumes verstehen wir das Maximum der Kardinalitäten aller seiner Schichten.

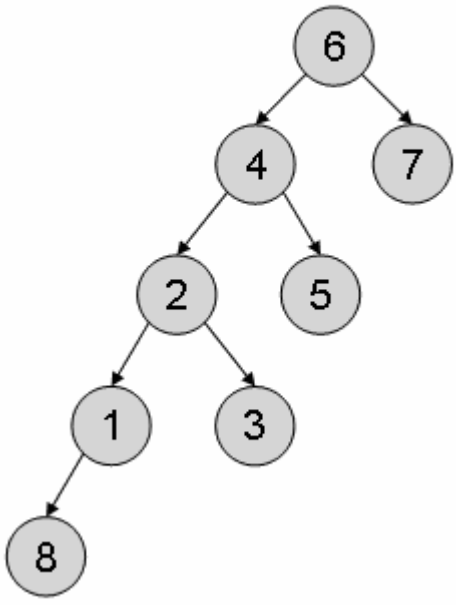
Schreiben Sie eine **function dicke(t: tree): integer**, die die Dicke des Baumes t nach obiger Definition ermittelt.

Testen Sie Ihr Programm an den folgenden Bäumen.

Baum 1

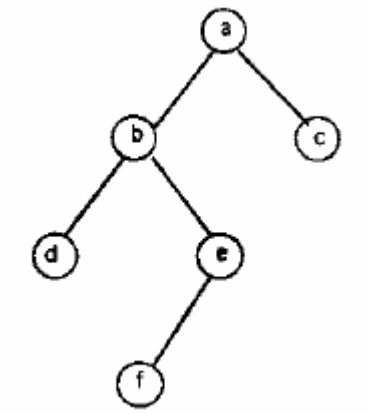


Baum 2



Aufgabe 3 (Traversierung von binären Bäumen)

Ein binärer Baum kann auf verschiedene Arten durchlaufen werden, so dass jeder Knoten genau einmal besucht wird. Wir wollen die vier verschiedenen Arten an folgendem Beispiel verdeutlichen.



a) Führen Sie eine Tiefensuche vom Wurzelknoten aus. Die Reihenfolge der besuchten Knoten ist:

a , b , d , e , f , c

Dieser Durchlauf durch den Baum heißt **Preorder** - Traversierung.

```
procedure TS(v:Knoten) is begin
  Put(v.value);
  if v.left  /= NULL then TS(v.left);
  if v.right /= NULL then TS(v.right);
end TS;
```

b) Betrachten Sie den folgenden Algorithmus mit Start IO(*w*):

```
procedure IO(v:Knoten) is begin
  if v.left  /= NULL then IO(v.left);
  Put(v.value);
  if v.right /= NULL then IO(v.right);
end IO;
```

Diese Durchlaufart ist die sogenannte **Inorder**-Traversierung. Die Reihenfolge der besuchten Knoten ist:

d , b , f , e , a , e

c) Betrachten Sie den folgenden Algorithmus mit Start PO(*w*):

```
procedure PO(v:Knoten) is begin
  if v.left  /= NULL then PO(v.left);
  if v.right /= NULL then PO(v.right);
  Put(v.value);
end PO;
```

Diese Durchlaufart ist die sogenannte **Post**-Traversierung. Die Reihenfolge der besuchten Knoten ist:

d , e , f , b , e , a

d) Führen Sie eine Breitensuche vom Wurzelknoten aus. Die Reihenfolge der besuchten Knoten ist:

a , b , c , d , e , f

Diese Durchlaufart ist die sogenannte **Level-Order** -Traversierung.

Geben Sie die Besuchsreihenfolge aller Traversierungen von den Bäumen aus Aufgabe 3 an.

Aufgabe 4: (Umwandlung einer Tiefensuche in eine Breitensuche)

Gegeben sei der folgende Algorithmus.

Richte leeren Kanten-Stack *S* ein
Markiere den Startknoten *u*
Füge alle Kanten in *S* ein, die aus *u* hinauszeigen
while *S* not empty loop
  Entferne die oberste Kante *e* = (*v;w*) von *S*
  if *w* ist noch nicht markiert then
    markiere *w*
    füge alle Kanten in *S* ein, die aus *w* hinauszeigen
  endif
endloop

Wie müssen Sie den obigen Algorithmus Tiefensuche verändern, um eine Breitensuche zu erhalten?

Allgemeine Hinweise:

- Bei weiteren Fragen, wenden Sie sich bitte an W. Schmid (sltsoftware@yahoo.de).
- Weitere Hinweise finden Sie auf unserer Veranstaltungswebseite unter:
  http://www.info2.de.vu
  http://www.zusatzkurs.de.vu