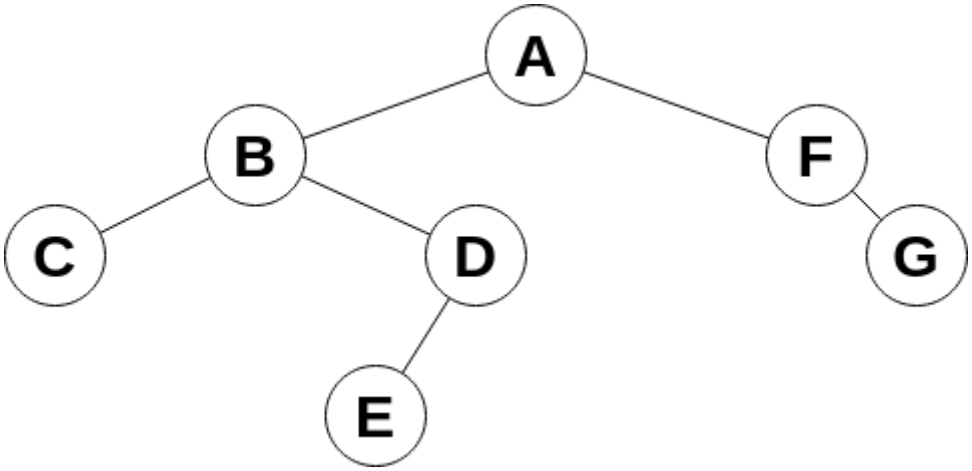
Levelorder Traversierung, Iterative Tiefensuche

Bei den drei rekursiv implementierbaren Traversierungen wird der Baum zuerst in die Tiefe durchwandert bis hin zu seinen Blättern ("Tiefensuche") - hier noch einmal am Beispiel bei der Preorder-Traversierung:



Bei der Levelorder Traversierung werden auf jedem Niveau des Baums erst alle Knoten besucht, bevor auf das nächste Niveau gewechselt wird, in unserem Beispielbaum ergibt sich damit die Traversierungsreihenfolge: A→B→F→C→D→G→E. Der Algorithmus zur Levelorder Traversierung ist nicht rekursiv.

Iterative Traversierung

Die rekursiven Implemetationen der TRraversierungen versagen ihren Dienst, wenn die Bäume zu tief werden, da der Call-Stack für die Rekursion nicht beliebig wachsen kann. Bei Java ist seine Größe auf ca. 256kB beschränkt, wenn diese Größe überschritten wird, erhältst du einen *Stack Overflow Error*:

```
java.lang.StackOverflowError
        at Baumalgorithmen.anzahl(Baumalgorithmen.java:73)
        at Baumalgorithmen.anzahl(Baumalgorithmen.java:73)
        at Baumalgorithmen.anzahl(Baumalgorithmen.java:73)
        at Baumalgorithmen.anzahl(Baumalgorithmen.java:73)
```

baeume_traversierung_iterativ.odp	106.8 KiB	14.02.2022	14:21
baeume_traversierung_iterativ.pdf	170.4 KiB	14.02.2022	14:21
levelorder.gif	44.1 KiB	14.02.2022	19:37
stackoverflow.png	13.9 KiB	14.02.2022	14:24

From:
https://www.info-bw.de/ -

Permanent link:
https://www.info-bw.de/faecher:informatik:oberstufe:adt:baeume:breitensuche:start?rev=1644848705

Last update: **14.02.2022 14:25**