

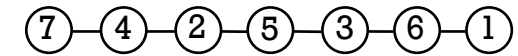


Bomen en Grafen

Les 1

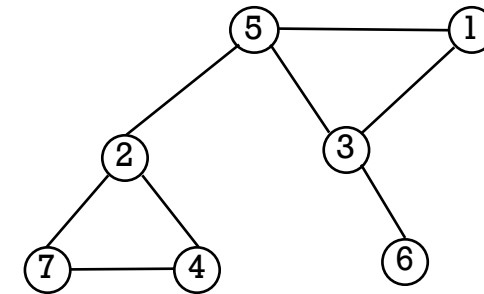
Datastructuren

■ **Lineair:** de elementen vormen een rij.

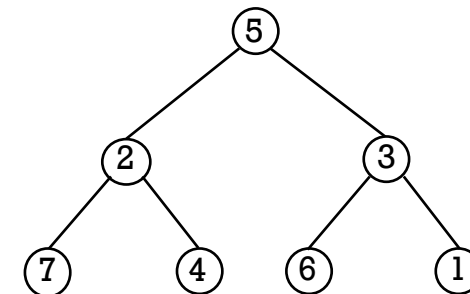


■ **Niet-lineair:** de elementen vormen geen rij.

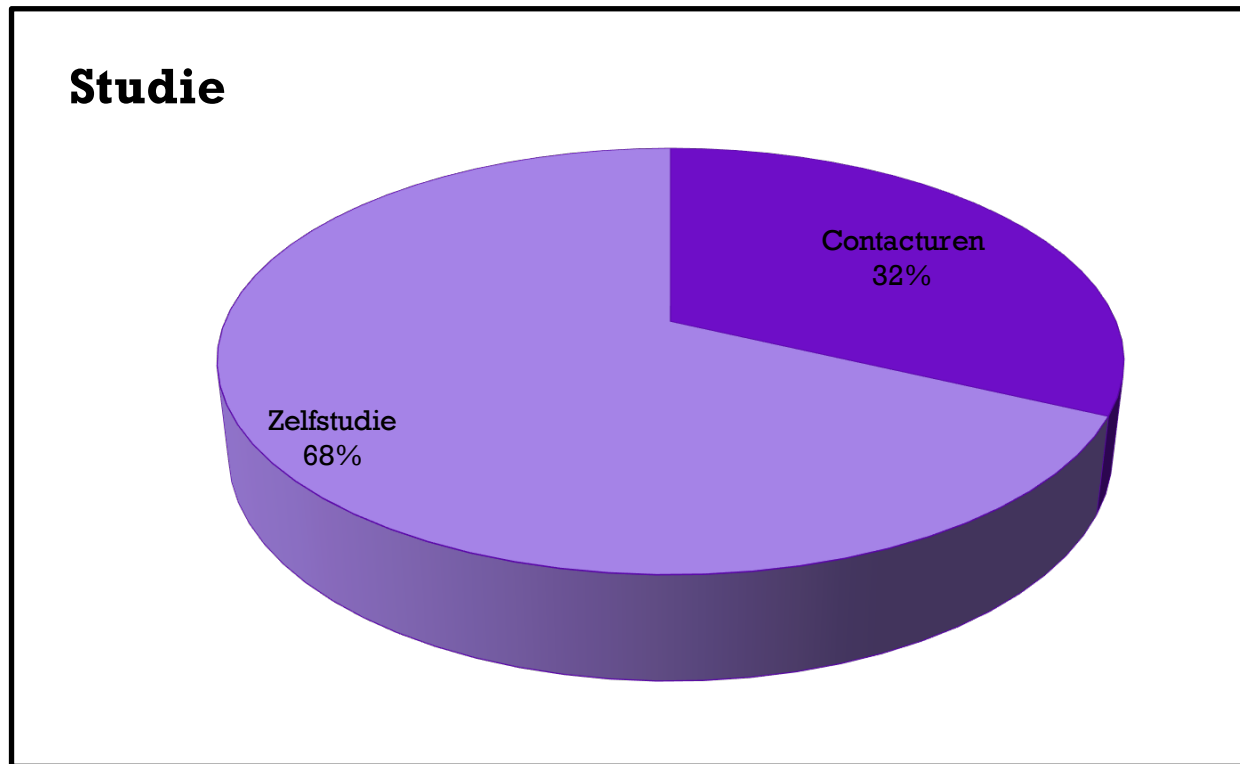
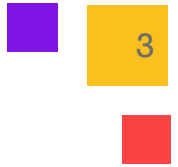
■ **Graaf:** lussen zijn toegelaten.



■ **Boom:** lussen zijn niet toegelaten.



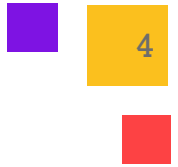
Tijdsbesteding



3 SP = 75 uren tijdsbesteding (*alles inbegrepen*)

- Contacturen: 24 uren (*2u/week ; 12 weken*)
- Zelfstudie: 51 uren

Studiemateriaal



- Geen cursus
- Slides en werkbladen worden via Toledo ter beschikking gesteld.

Evaluatie

- Geen permanente evaluatie of tussentijdse toets/examen.
- Alle punten staan op het examen in juni.
 - Deel 1: schriftelijke vragen in verband met theorie → gesloten boek
 - Deel 2: praktijkexamen op pc waarop student(e) algoritme implementeert op eigen laptop (geen wifi/internetverbinding)

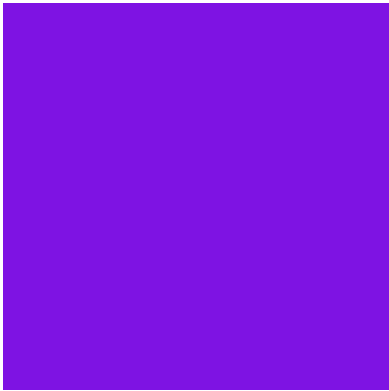
Leertips

Wees actief.

- Kom naar alle lessen.
- Hou de leerstof bij. De lessen bouwen sterk voort op elkaar.
- Werk alle oefeningen van een bepaalde les af tegen de volgende les.
- Zorg dat je alle oefeningen zelf kan oplossen.
- Stel meteen vragen als je iets niet begrijpt of onzeker bent over je oplossing.
- Hou al je opgeloste oefeningen bij zodat je deze opnieuw kan bekijken voor het examen.
- Leer geen code vanbuiten!



Vragen?



Iteratie
vs
recursie

Voorbeeld

8

Print alle gehele getallen vanaf het gegeven getal a tot en met het gegeven getal b.

■ Iteratief

```
public void print(int a, int b){  
    for(int i=a; i<=b; ++i){  
        System.out.print(i + " ");  
    }  
}
```

Correct?

■ Recursief

```
public void print(int a, int b){  
    if(a>=b) return;    if(a>b) return;  
    System.out.print(a + " ");  
    print(a+1,b);  
}
```


Opbouw recursief algoritme



Print alle gehele getallen van a tot en met b

- Basisgeval

Er zijn geen getallen om te printen
($b < a$)

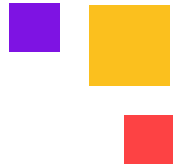
- Algemene regel

die algemeen geval uiteindelijk reduceert tot basisgeval

Print a

Print alle getallen van $a+1$ tot en met b

Opbouw recursief algoritme



Bereken som van eerste n gehele getallen

- Basisgeval

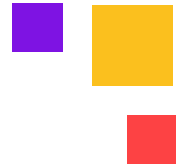
$n=0$: som = 0

- Algemene regel

die algemeen geval uiteindelijk reduceert tot basisgeval

$\text{som} = n + \text{som } n-1$ gehele getallen

Implementatie recursief algoritme



Bereken som van eerste n gehele getallen

```
private static void som (int n) {
```

■ Basisgeval

```
if (basisgeval) { return ...}
```

n=0: som =0

```
if (n==0) { return 0}
```

■ Algemene regel

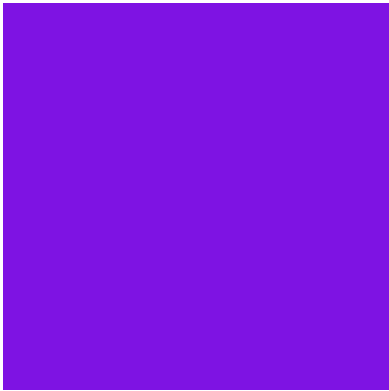
die algemeen geval uiteindelijk reduceert tot basisgeval

```
else { return ...}
```

som = n + som n-1 gehele getallen

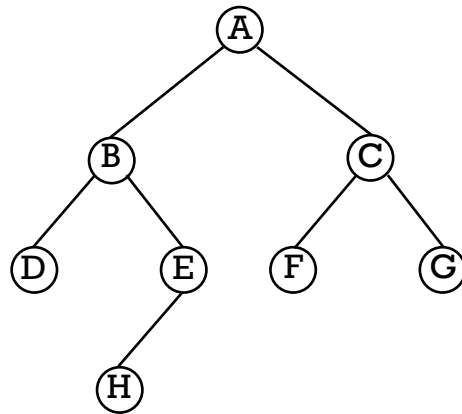
```
else { return n + som(n-1)}
```

```
}
```



Bomen

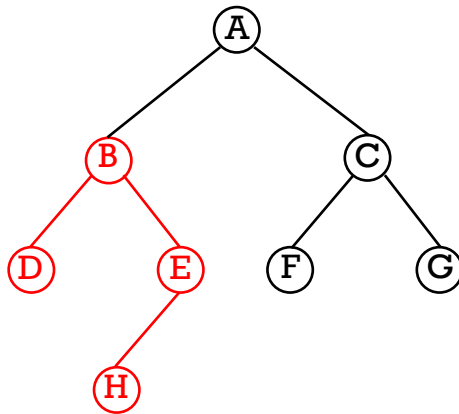
Terminologie



- A noemt men de **wortel (root)** van de boom.
- De knopen B en C zijn de **kinderen** van de wortel, D en E zijn de kinderen van B, ...
- Omgekeerd is knoop C de **ouder (parent)** van F en G en is E de ouder van H.
- Een knoop zonder kinderen wordt een **blad (leaf)** of **externe knoop** genoemd. (D, F, G en H)
- Een knoop die minstens 1 kind heeft, wordt een **interne knoop** genoemd. (A, B, C en E)

Terminologie

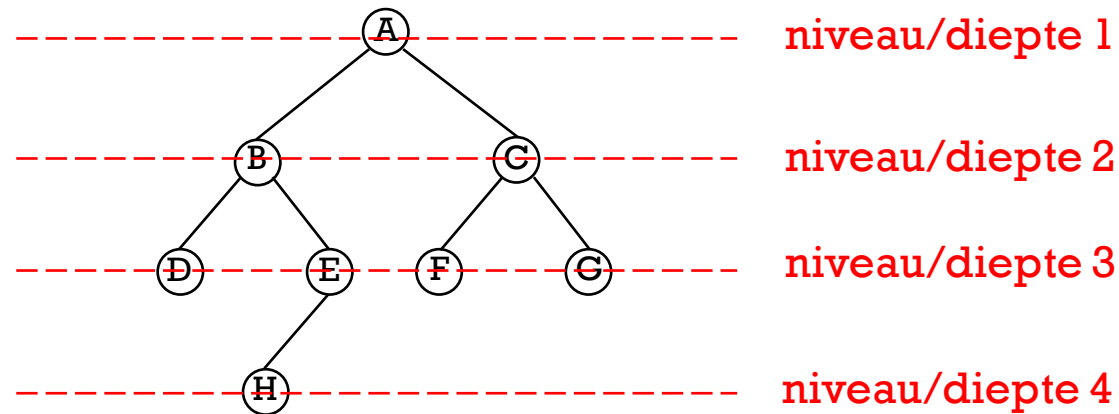
B en al zijn afstammelingen vormen een subboom waarvan B de wortel is.



- A noemt men de **wortel (root)** van de boom.
- De knopen B en C zijn de **kinderen** van de wortel, D en E zijn de kinderen van B, ...
- Omgekeerd is knoop C de **ouder (parent)** van F en G en is E de ouder van H.
- Een knoop zonder kinderen wordt een **blad (leaf)** of **externe knoop** genoemd. (D, F, G en H)
- Een knoop die minstens 1 kind heeft, wordt een **interne knoop** genoemd. (A, B, C en E)
- Elke knoop kan beschouwd worden als de wortel van een **subboom**.

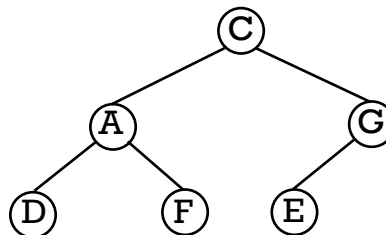
Terminologie

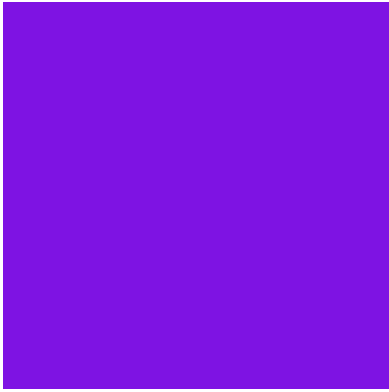
niet-complete
binaire boom



- Een **binaire boom** is een boom waarvan elke knoop ten hoogste twee kinderen heeft.
- Knoop B is het **linkerkind** (**left child**) van A en C is het **rechterkind** (**right child**) van A.
- De **(maximale) diepte** van de boom is het maximaal aantal knopen van een pad van de wortel tot een blad. Bovenstaande boom heeft een (maximale) diepte gelijk aan 4.
- Een binaire boom wordt **compleet** genoemd als al zijn niveaus behalve eventueel de laatste volledig gevuld zijn en alle knopen op het laatste niveau aan de linkerkzijde zijn.

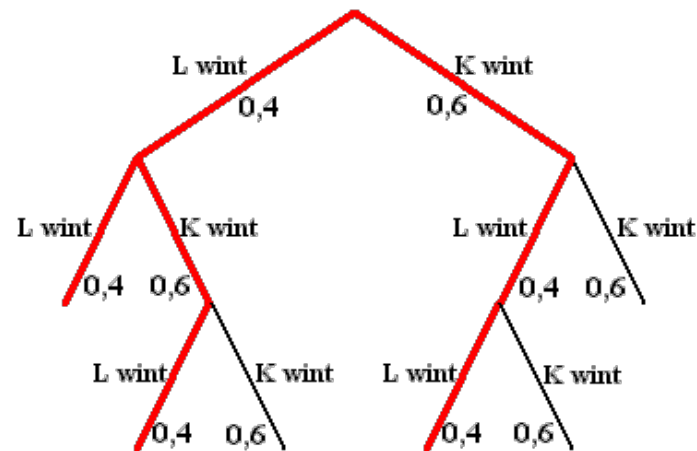
complete
binaire boom
met diepte 3



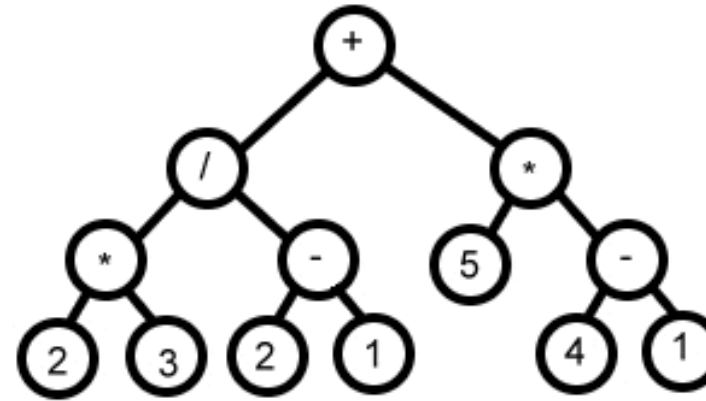


Toepassingen
van bomen

Kansbomen (statistiek)

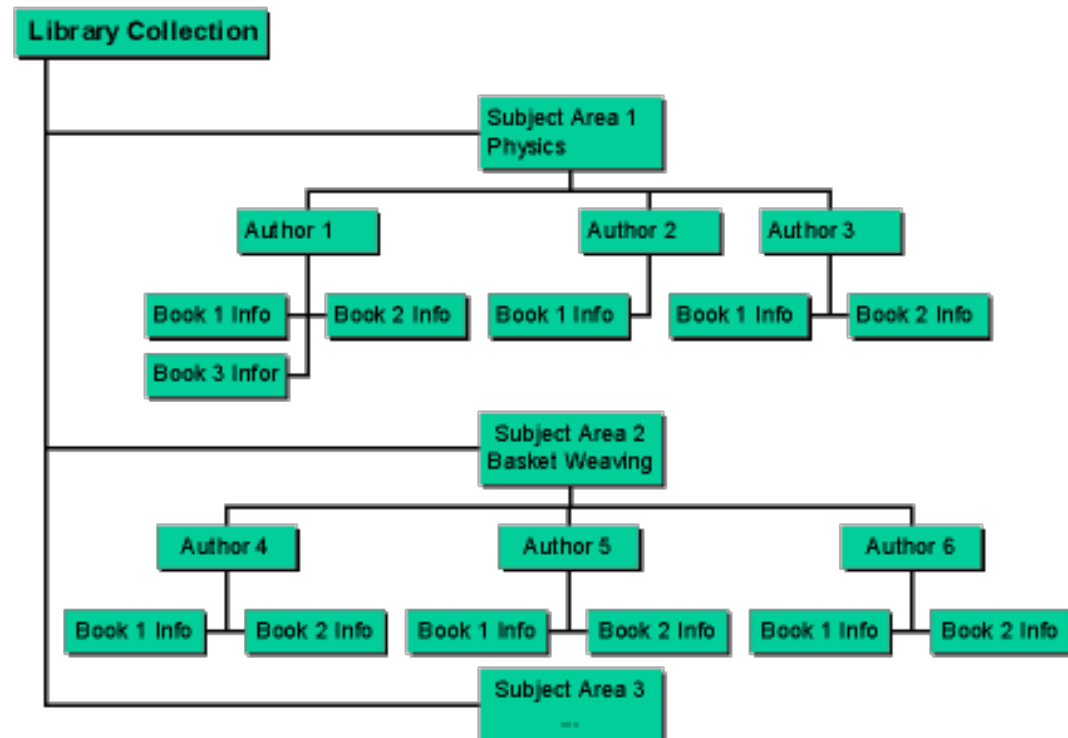


Expressiebomen



Expression tree for $2*3/(2-1)+5*(4-1)$

Hierarchical database model





Vragen?