# 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

## Iteratie vs. recursie

* Iteratief: maakt gebruik van een loop
* Recursief: roept zichzelf terug op in de functie

## Bomen

* Bovenste element van een boom is de wortel(root) van de boom.
* De knopen er onder zijn de kinderen van de wortel.
* Omgekeerd is de knoop boven de kinderen zijn ouder.
* Een knoop zonder kinderen is een blad
* Een knoop met minstens 1 kind is een interne knoop

# Les 3: Binaire boom

* Een binaire boon is een boom waarvan elke knoop ten hoogste twee kinderen heeft.
* Knoop B is het linkerkind van A en C is het rechterkind van A.
* De diepte van de boom is het maximaal aantal knopen van een pad.
* Een binaire boom wordt compleet genoemd als zijn niveaus behalve eventueel de laatste volledig gevuld wijn en alle knopen op het laatste niveau aan de linkerzijde zijn.

Wordt verdeeld in twee delen, linker boom, rechter boom.

## Wandelen door een boom:

### Pre order:

1. Bezoek de knoop
2. Wandel door de linkersubboom
3. Wandelen door de rechtersubboom

### In order:

1. Wandel door de linkersubboom
2. Bezoek de knoop
3. Wandelen door de rechtersubboom

### Post order:

1. Wandelen door de linkersubboom
2. Wandelen door de rechtersubboom
3. Bezoek de knoop

# Les 4: binaire zoekboom

Een binaire zoekboom of gesorteerde binaire boom is een speciaal geval van een binaire boom waarbij de data in de boom voldoen aan de eigenschap:

Voor elke knoop in de boom geldt dat zijn waarde strikt groter is dan alle waarden in zijn linkersubboom en strikt kleiner is dan alle waarden in zijn rechtersubboom

# Les 6: bomen en grafen

## Array implementatie

Je kunt een boom intern als een lineaire datastructuur voor te stellen.

Lees de boom van linksboven naar rechts beneden en schrijf de waarden op in een array.

2i + 1 vind je het linkerkind

2i + 2 vind je het rechterkind

(i-1)/2 naar beneden afgerond vind je de parent

## Heaps

Een heap is een special geval van een boom waarbij de data in de boom voloet aan de heap eigenschap.

### Max heap:

Voor elke knoop in de boom geldt dat zijn waarde groter of gelijk is aan de waarden van zijn kinderen.

### Min heap:

Voor elke knoop in de boom geldt dat zijn waarde klener of gelijk is aan de waarden van zijn kinderen.

## Een binaire heap

Een binaire heap is een complete binaire boom waarbij de data in de boom voldoen aan de heap eigenschap

# Les 8: Grafen

Een graaf is niet-lineair: lussen zijn toegelaten.

Een graaf is een verzameling knooppunten die met mekaar verbonden zijn.

Niet-gerichte graaf: er worden geen richtingen weergegeven.

Gerichte graaf: er wordt wel een richting weergegeven.

Gemengde graaf: combinatie van de twee.

Een verbindingsmatrix is een matrix die de paden weergeeft in tussen twee knooppunten.

## BFS

Gegeven twee knooppunten. Vind het pad met het kleinst aantal tussenliggende knooppunten

Reconstrueer dit pad

Zoek in breedte: voeg knooppunten toe die rechtstreeks verbonden zijn met gegeven knooppunt

# Les 9: Floyd

Zoekt paden met kleinste gewicht.

Geeft lijst met “korstste” paden tussen eender welke twee knooppunten van graaf.

Systematisch een knooppunt meer toelaten als ‘tussenstation’ om afstanden te optimaliseren

* Eerst knooppunt1, dan knooppunt 2, …
* Telkens je een knooppunt als tussenstation toelaat, afstanden optimaliseren.
* Gebruikte tussenstations noteren in pointermatrix
* Einde: pointermatrix met gebruikte tussenstations

1. Knoop 1 als tussenstap toelaten indien afstand korter wordt
2. Niet alleen knoop 1 als tussenstap toelaten, maar ook knoop 2 indien afstand korter wordt
3. Niet alleen knoop 1 en 2 als tussenstap toelaten, maar ook knoop 3 indien afstand korter wordt.
4. Knoop 4 ook toe laten.
5. Knoop 5 ook toelaten

# Les 10: algoritme van Dijkstra

Zoekt pad met kleinste gewicht bij een gewone graaf.

Geeft een lijst van kortste paden die vertrekken uit een gegeven startknoop.

Noteerd alle mogelijke wegen en neemt de kleinste