

...hier wordt gedacht zoals een computer...

taalvaardig en lichaamsbewust programmeren zonder computer

Leen Brouns
opleiding Industrieel Ingenieur UGent
leen.brouns@ugent.be



*...hier wordt **gedacht***

*zoals een **computer***
taalvaardiger...
lichaamsbewust
programmeren
zonder computer

Leen Brouns
opleiding Industrieel Ingenieur UGent
leen.brouns@ugent.be

programmeren zonder computer

PROGRAMMEREN

dagprogramma

- opsomming van wat er achtereenvolgens moet gebeuren
(om tevreden te gaan slapen)

PROGRAMMEREN

vs.

COMPUTE(r)

een lijst taken vastleggen
om iets te bereiken

berekenen

- ongeacht de nieuwe
gegeven situatie

= het uitvoeren van de
vastgelegde taken,
rekening houdend met
de huidige situatie

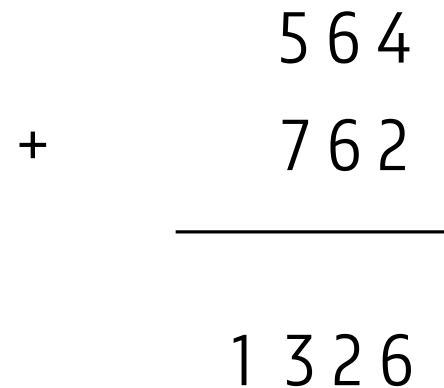
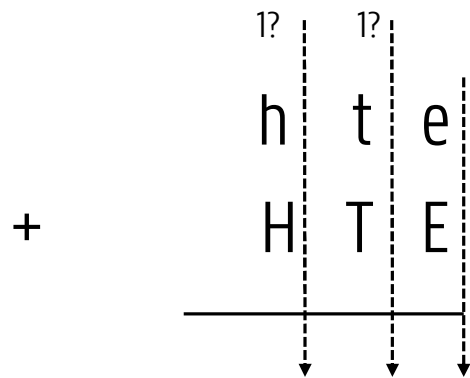
PROGRAMMEREN

vs.

COMPUTE(r)

wij werden zelf ook
geprogrammeerd...

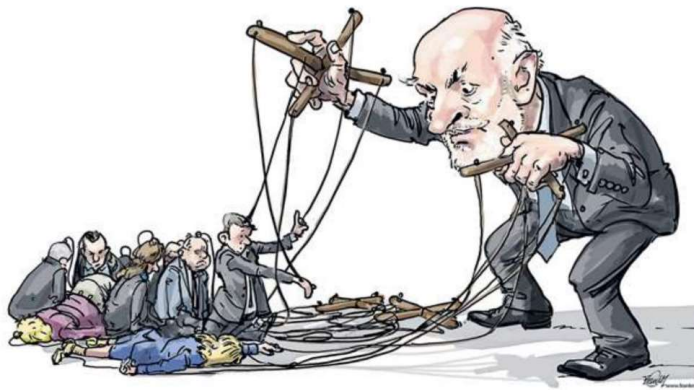
... om taken uit te voeren of te
berekenen



PROGRAMMEUR

vs.

COMPUTER



bepaalt wat er gebeurt



doet het domme rekenwerk
voert de taken uit

computationeel denken

losse set probleemoplossende vaardigheden

≠ hoe de computer denkt

= hoe mensen moeten denken om de computer
ongelooflijke resultaten te laten bereiken



logisch denken

juiste voorstelling
van data

*creatief
zijn*



aandacht voor
details

generalisatie

oog voor

patroonherkenning

efficiëntie & evaluatie

computationeel denken

o.a. algoritmisch denken

zie de oplossing op problemen als
'algoritmes':

een set regels die je dient te volgen om een
probleem op te lossen

dankbare onderwerpen om dit
denken in te oefenen

alles wat je dagelijks doet

– zonder bij na te denken –

en waar wat 'structuur / herhaling' in zit

selection sort - selecteer de grootste / kleinste



bubble sort - wissel om met je buur (en blijf dat doen..)



insertion sort - inschuifsorteren



GEEF EEN ALFABETISCHE LIJST



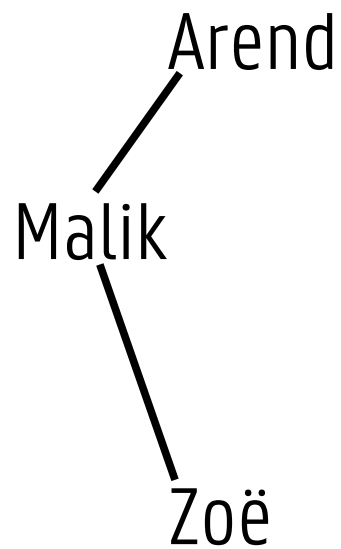
GEEF EEN ALFABETISCHE LIJST



GEEF EEN ALFABETISCHE LIJST

Malik
Zoë
Arend
Jana
Seppe
Thijs
Anne

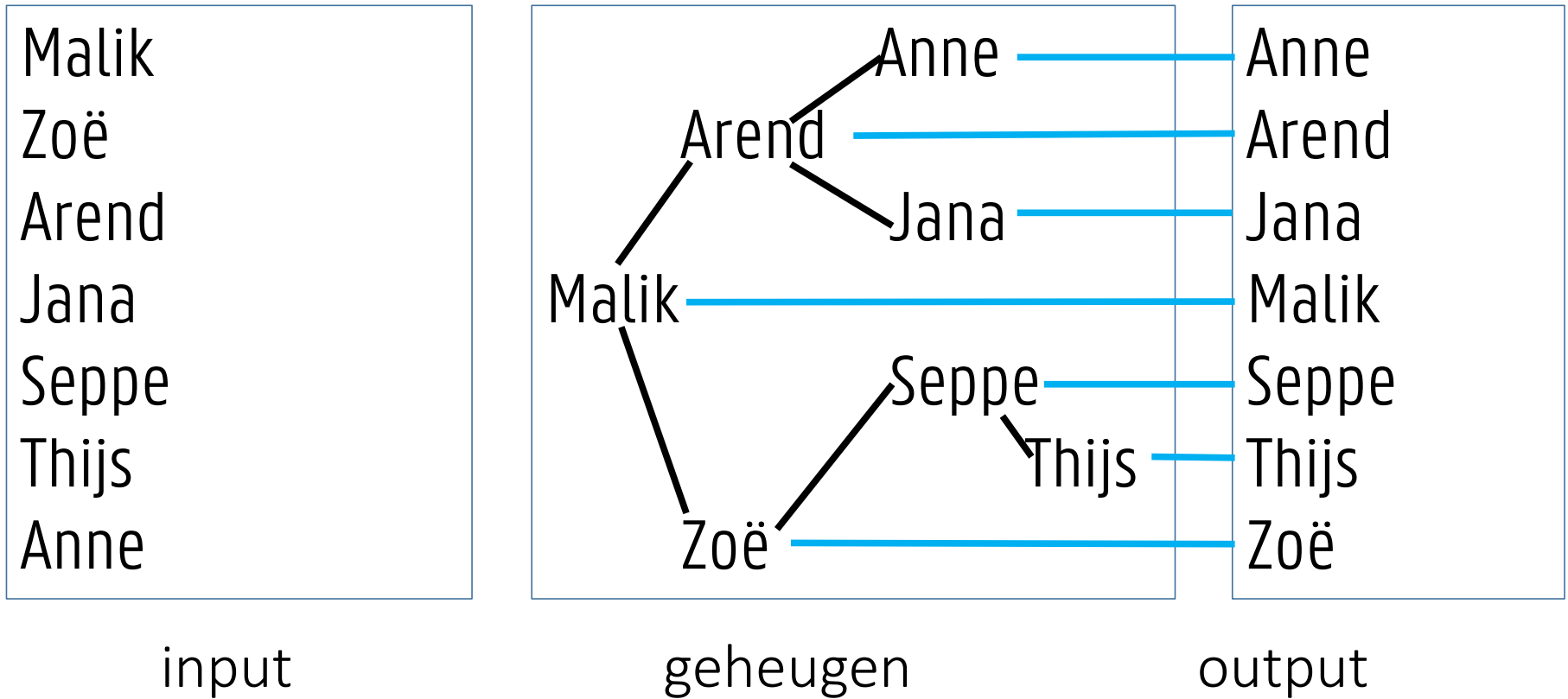
input



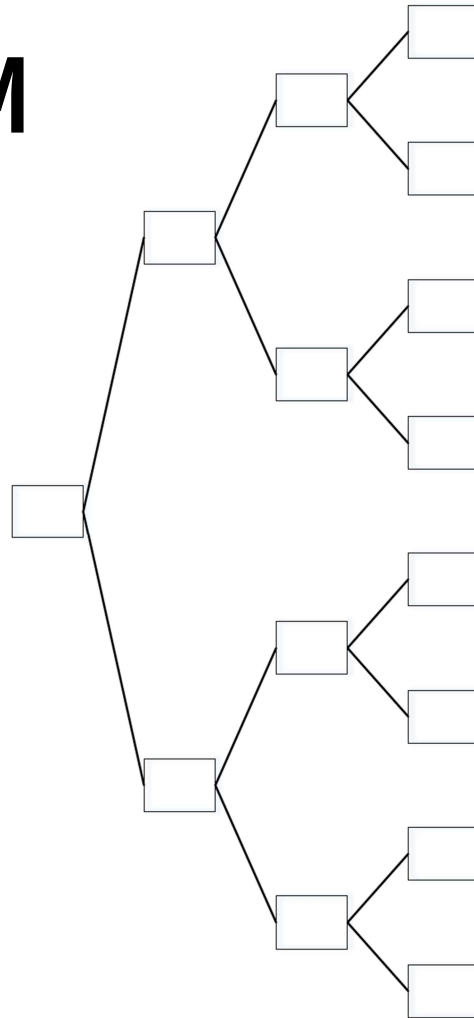
geheugen

output

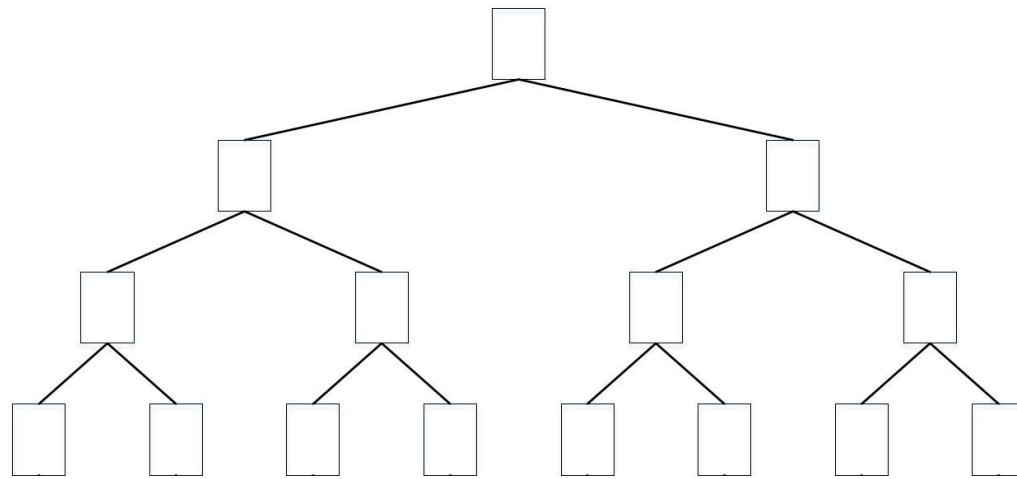
BOOMSTRUCTUUR



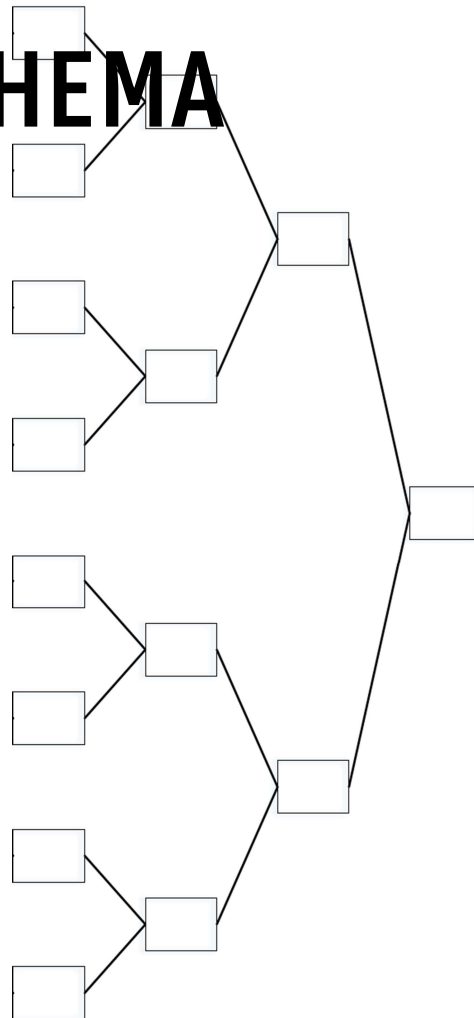
SORTEERBOOM



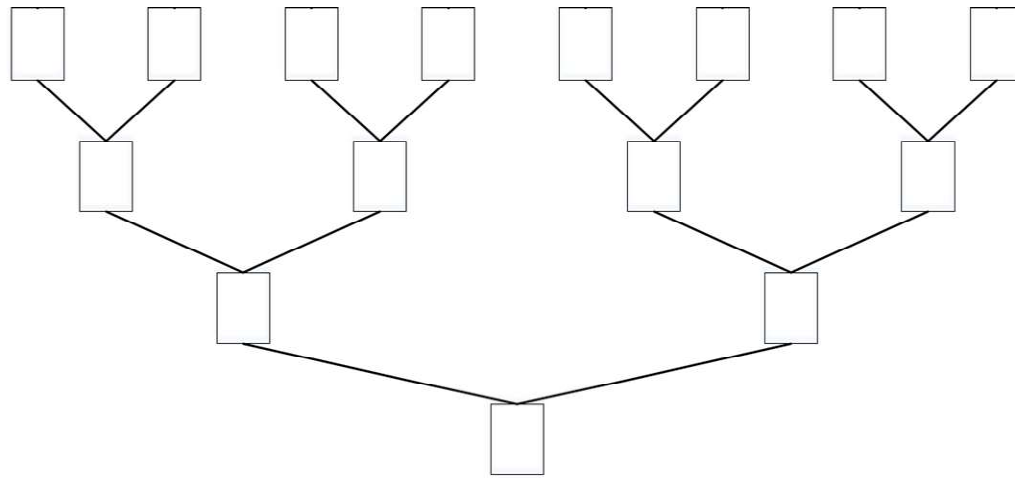
DETERMINATIETABEL



WEDSTRIJDSHEMA



STAMBOOM



BOOMSTRUCTUUR

hoe beschrijf ik
een
boomstructuur

in een
tekst-
bestand?

input

waar zijn
boomstructuren nog
nuttig

in het geheugen van
een computer?

geheugen

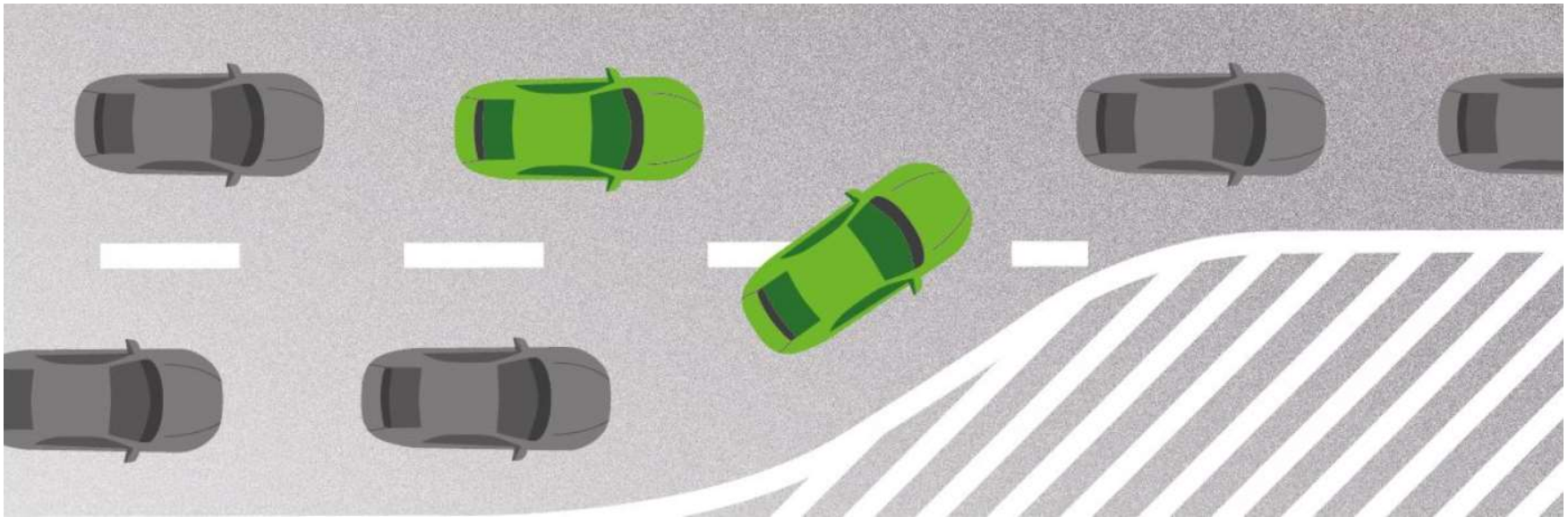
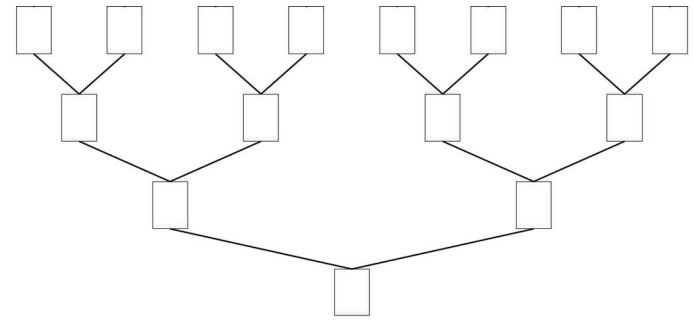
hoe schrijf
ik een
boom uit

in een
tekst-
bestand?

output

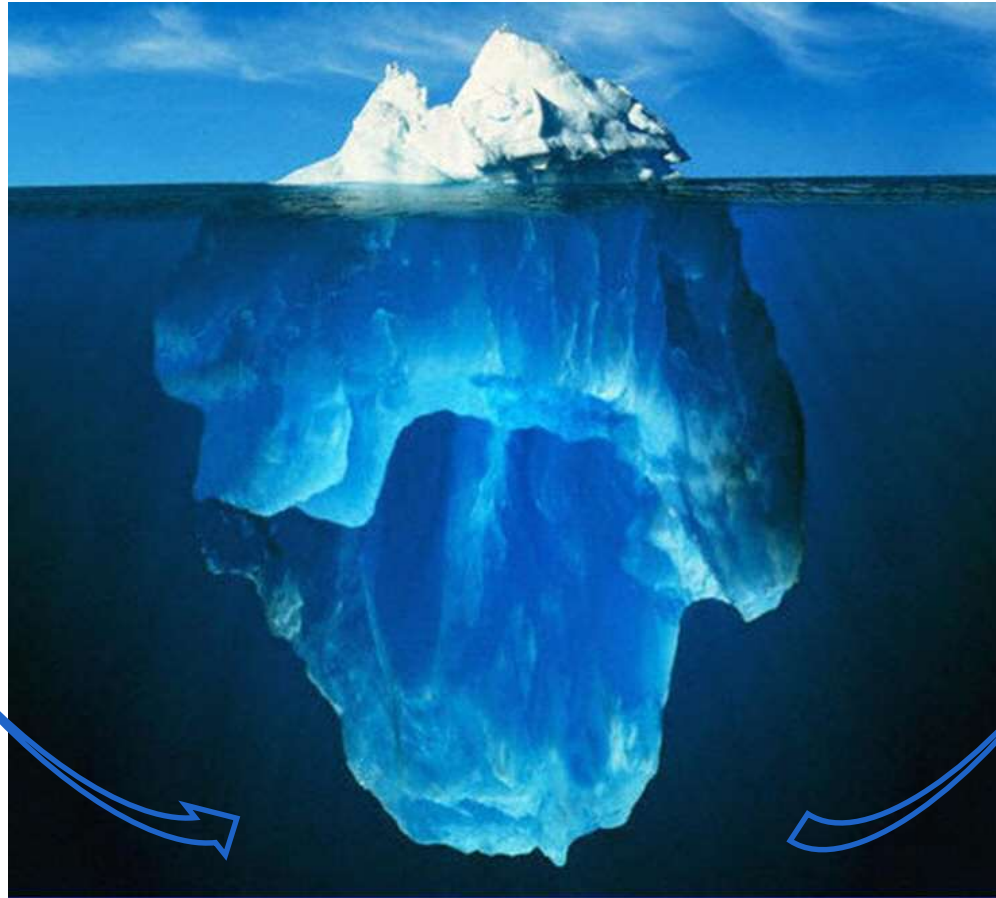
taalvaardig en lichaamsbewust

RITSEN



WAAR WILLEN WE NAARTOE?

tekst
opgave



tekst
code



STAPPENPLAN

- Start met een klein voorbeeld
- Formuleer je eigen oplossingsstrategie
- Speel advocaat van de duivel (of laat die spelen door...)
 - Pas je oplossingsstrategie aan
- Blaas het probleem op tot belachelijk grote proporties
- Nog altijd efficiënt? Bravo!

... ER IS MEER

- Zoeken, sorteren, snelste traject van station A naar station B,...
tabellen, bomen, grafen,....
- Beveiligen van informatie
from Caesar Cipher (geheimschrift) to Public Key Encryption (CSUnplugged Kid Krypto)
- Foutjes in informatie oplossen
een kras op m'n CD... welke CD-speler kan het toch lezen?
- Doorsturen van informatie
netwerkprotocols (CSUnplugged Tablets of stone),
coderen en decoderen (Dwengo CSUnplugged Fax)

ENKELE BELANGRIJKE CONCLUSIES

- Met CS UNPLUGGED-activiteiten kan je **computationeel denken** aanbrengen in de klas zonder computer
- Verklap nooit de oplossingsmethode maar laat de jongeren **zelf ontdekken**
- Hou na de activiteit steeds een discussie waarin je terugkoppelt met de **informaticawetenschappen**

BRONNEN – EN MEER ACTIVITEITEN

- www.csunplugged.nl informatica zonder draadjes

- www.dwengo.org/teach



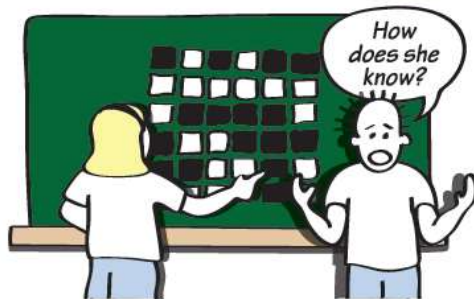
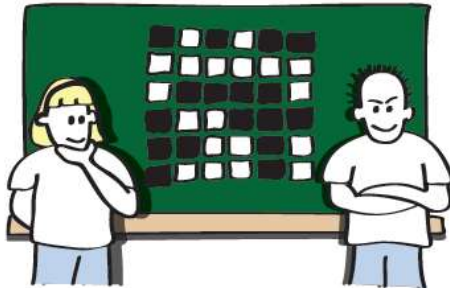
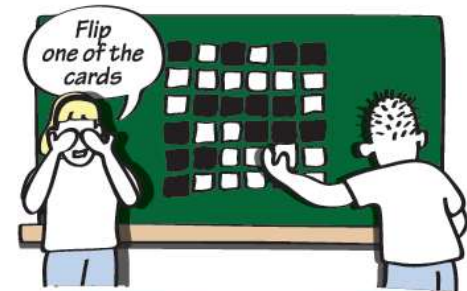
a movement, a vision and

a set of values about science education  - Francis Wyffels

@fwyffels



- robotcompetitie georganiseerd door  
- workshops en lezingen van UGent voor leerlingen en scholieren
www.ugent.be/educatief-aanbod
- www.program-uurtje.org



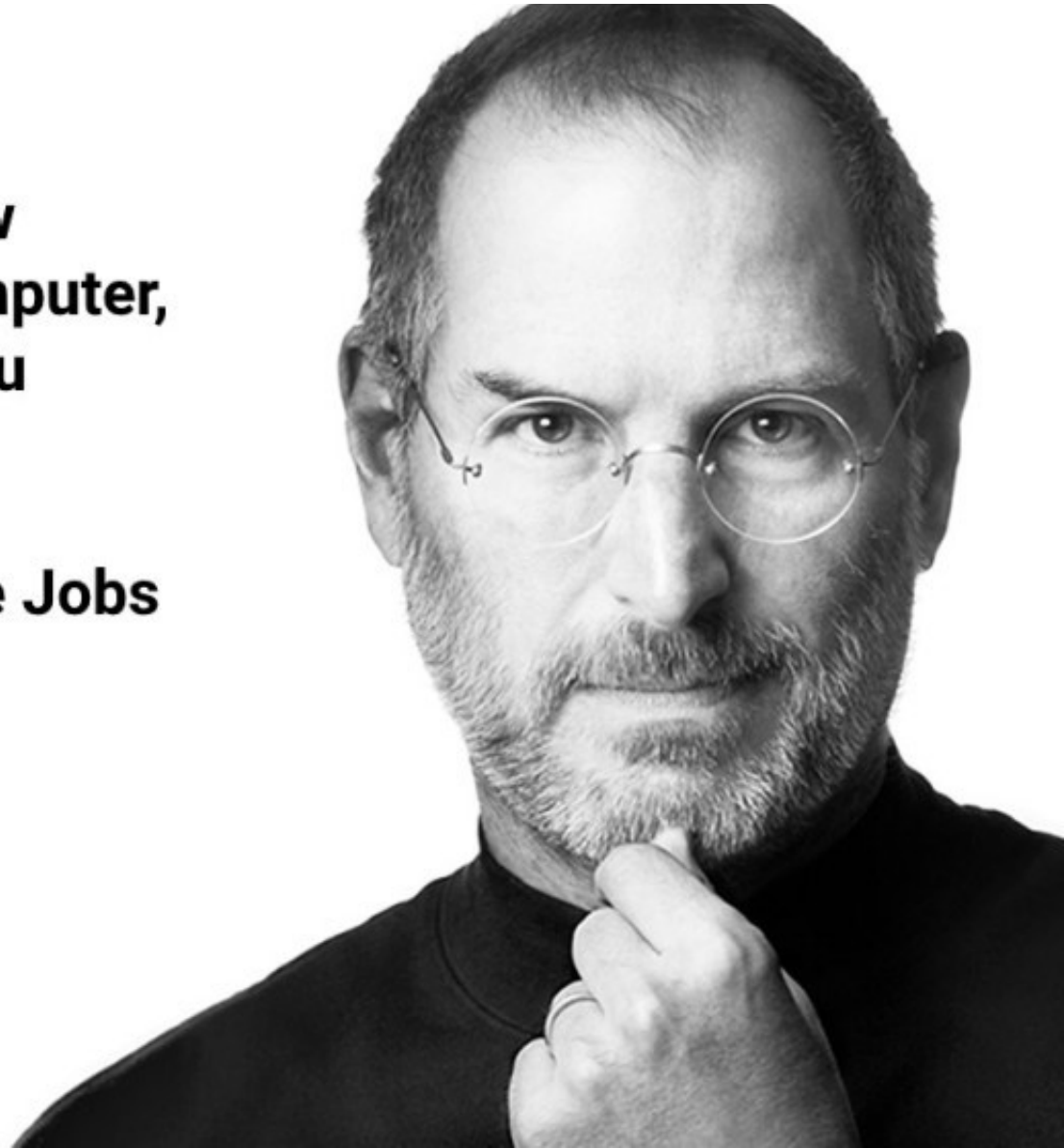
COMPUTER SCIENCE
Unplugged

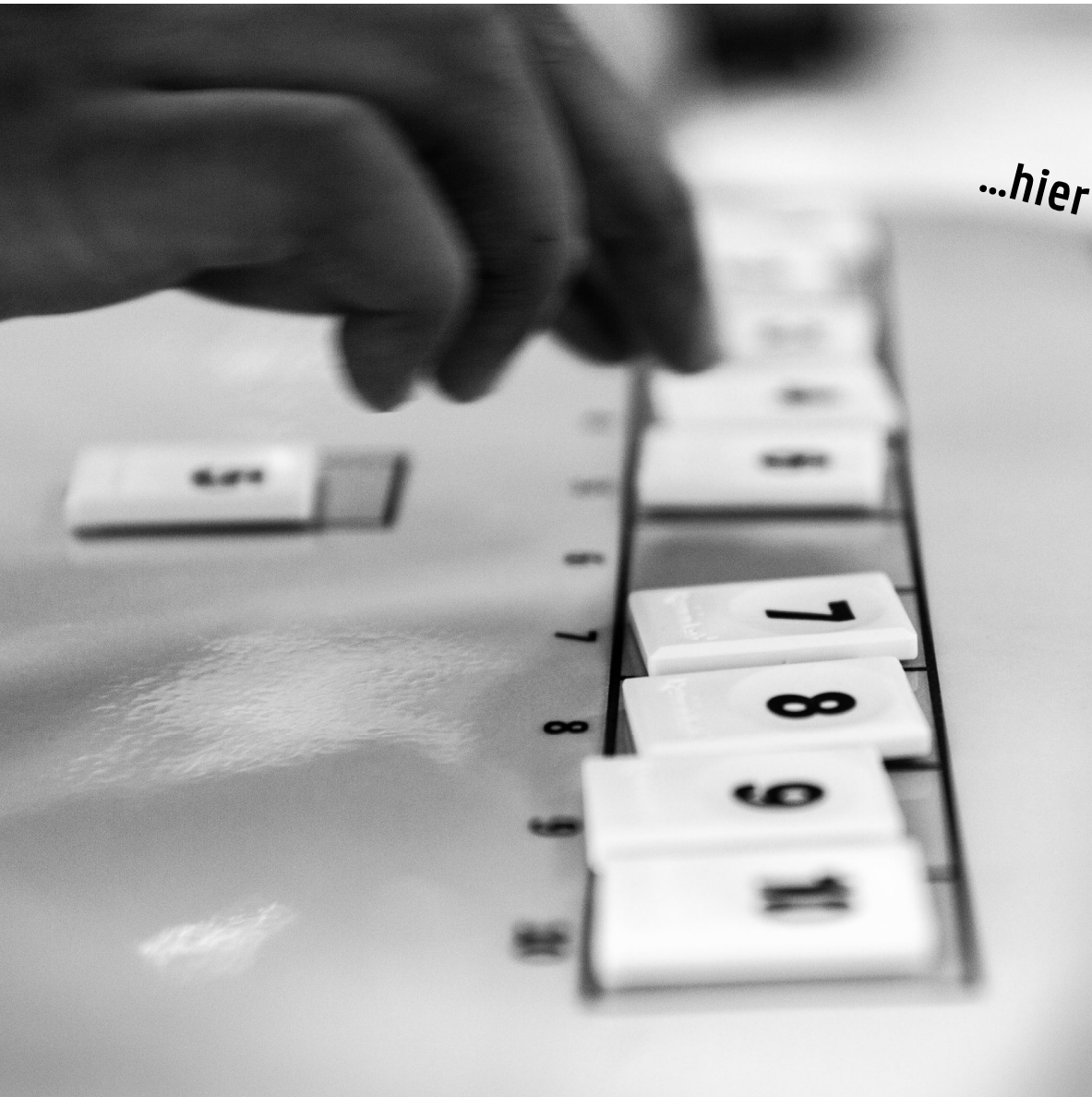
CS UNPLUGGED

- Opgestart door Tim Bell
- Open source
- Leren over informaticawetenschappen en computationeel denken zonder computer
- Instructiefilmpjes op het internet beschikbaar
- Vertaald in tal van talen waaronder het Nederlands: <http://www.csunplugged.nl>
- Eens proberen?

**"Everyone should know
how to program a computer,
because it teaches you
how to think."**

Steve Jobs





*...hier wordt gedacht zoals een computer,
zoals jullie net deden...*

**TAALVAARDIG EN LICHAAMSBEWUST
PROGRAMMEREN ZONDER
COMPUTER**

VRAGEN?

Leen Brouns
opleiding Industrieel Ingenieur UGent
leen.brouns@ugent.be

TOT HIER DE SLIDE-SHOW ZOALS AANGEBODEN OP DE ICT-PRAKTIJKDAG VAN 3 FEBR 2020

Hierna volgt nog een aantal ideeën, oefeningen, tips, herbruikbaar materiaal,...

HANDLEIDING

Beste deelnemer van de workshop,

Hier vindt u een collectie van slides, ideeën, voorbeelden,...

Knip-en-plak zoals het uitkomt voor uw les / workshop / lezing / ...

Ik heb tijdens de workshop zelf ook geknipt-en-geplakt, dat ziet u aan de dubbeltjes die af en toe voorkomen 😊

De 'oefeningen' die we gedaan hebben (of niet gedaan hebben), beschrijf ik telkens op een aparte slide in (dit) klein lettertype.

Zo heeft u een geheugensteuntje – en een paar nieuwe ideeën.

Allicht vindt u er ook slides waar u niets bij kan verzinnen – dan vond deze oefening geen plaats in de workshop die u volgde; goed voor een volgende keer!

Veel plezier met het uitproberen,

Leen

OEFENING: SORTEREN (individueel en per twee)

DEEL 1

Geef elke deelnemer een zak Rummikub-blokjes en een papier met 10 + 1 vakjes.

Laat de 10 vakjes opvullen met blokjes in willekeurig volgorde.

Opdracht: gebruik slechts 1 vinger, verschuif de blokjes, laat ze enkel los // een vakje, zorg ervoor dat de 10 blokjes na afloop van klein naar groot gesorteerd liggen.



DEEL 2

Laat de deelnemers per 2 zitten. Een van hen is de programmeur, de andere speelt computer. De computer start opnieuw met 10 blokjes in willekeurige volgorde. De programmeur moet niet per se blokjes voor zich hebben, maar het mag.

Opdracht: de programmeur geeft mondelinge opdrachten aan de computer; als de computer deze allemaal correct uitvoert, moeten zijn blokjes op het einde van klein naar groot gesorteerd liggen.

Randvoorwaarde:

(1) De computer mag niets zeggen – enkel crashen als een onmogelijke opdracht gegeven wordt. Zo kan de opdracht “neem blokje met getal 1” niet uitgevoerd worden, als de computer geen blokje 1 heeft.

(2) De programmeur mag niet gaan spieken bij de computer: diens blokjes zijn onzichtbaar!

Als leerkracht / begeleider: ga goed luistervinken welke soort sorteermethodes aan bod gekomen zijn. Verderop in de slides staan er drie eenvoudige sorteermethodes, waarvan de eerste twee zeer frequent toegepast worden in deze oefening.

5	7	1	10	3	8	4	6	12	9
0	1	2	3	4	5	6	7	8	9



x



OEFENING: SORTEERMETHODE (in groep)

Benodigdheden voor een klas: 4 x 5 bekers of potjes van dezelfde kleur en een grote open ruimte. Zet de bekers in de ruimte: volg hierbij de cirkels in het schema. (Zet de bekers op elkaar bij meer kleuren.) Laat alle deelnemers plaatsnemen achter één van de eerste acht bekers. Je krijgt dus acht rijtjes op de plaats van de gekleurde strepen. Laat iedereen in alfabetische (voornaam-)volgorde gaan staan – belangrijk! Dan start het sorteren.



Elke deelnemer die vooraan in het rijtje staat, aan de beker, converseert met de deelnemer aan de beker met dezelfde kleur. De eigenaar van de voornaam die eerst komt in het alfabet, stapt tot aan de volgende beker in dezelfde kleur (stippellijn). De deelnemer die achter deze eigenaar stond, staat nu als eerste aan de beker. Hetzelfde herhaalt zich, tot iedereen op de stippellijn van de juiste kleur staat. Dan herhaalt het spel zich: de rode en oranje stippellijn 'ritsen', de groene en blauwe ook. Tot de hele klas gerangschikt staat achter de stapel van vier bekers.

Ideaal als kennismaking in september...

VARIANTEN OP VORIGE OEFENING

Vorige methode vraagt veel uitleg. Daarom een paar voorbereidende (en extra) oefeningen, die het systeem van het ritsen eenvoudiger aanbrengeen. Naar eigen inspiratie, en aangepast aan de doelgroep, in te zetten.

- (0) Openritsen: dit is makkelijker uit te leggen en uit te voeren. Laat de kinderen vanuit 1 lange rij opsplitsen (afwisselden links/rechts) in 2 rijen.
- (1) Toeritsen: laat vanuit vorige oefening de leerlingen zich een halve slag omdraaien. Laat de 2 rijen nu weer toeritsen. Een stukje verkeersopvoeding...
- (2) Ritsen op klasnummer: zet de klasgroep in twee rijen parallel aan elkaar, laat hen binnen elke rij gaan staan op oplopend klasnummer, en rits dan rekening houdend met het klasnummer. Idealiter staan de oneven nummers NIET allemaal samen in een rijtje, maar zitten er soms clusters (7-8-9-10-11) in dezelfde starrij.
- (3) Ritsen vanaf kleinere groepjes: zelfde oefening als daarnet (op klasnummer ritsen), maar starten van vier rijen die twee aan twee ritsen naar twee rijen, en tenslotte naar één rij.
- (4) Ritsen vanaf groepjes van één: geef elke deelnemer een set Rummikubblokjes en laat ze op een (vertikaal) rijtje leggen. Vraag nu hoe deze blokjes, met de 'rits'methode allemaal in een (horizontaal) oplopend rijtje kunnen terechtkomen. De oefening wordt dus nagebouwd met blokjes in plaats van deelnemers. Laat deelnemers hier eerst zelf over denken, laat ze dan bij elkaar te rade gaan. Tenslotte vraag je één of twee deelnemers of ze de klas willen begeleiden bij de oefening 'zet de hele klas op alfabetische (voornaam)volgorde'. TIP: in deze oefening zijn de starrijtjes heel erg kort: ze hebben lengte 1. Nu zullen er som rijen zijn die geen 'maatje' hebben om mee te ritsen. Dat is niet erg, die rij mag dan meteen doorstappen naar het volgende 'stadium'.

Voor wie graag meer informatie opzoekt over deze sorteermethode: deze draagt de Engelse naam 'merge sort'. De zoekterm 'ritssorteren' zal niets opleveren 😊

OEFENING: TER PLAATSE SORTEREN (individueel)

Stel dat de gegevens die gesorteerd moeten worden, gekend zijn. Zelfs zo gekend, dat je al weet waar hun juiste plaats is in het gesorteerde eindresultaat. Dan kunnen we een speciale sorteermethode toepassen, die zo min mogelijk verplaatsingen doet.

Benodigd materiaal: een lange strook papier met vakjes genummerd van 1 tot 10, en een pyramide van 10 genummerde stapelblokken. Zet de blokken in de genummerde vakjes, maar niet op volgorde. Een goede start is bvb

10	4	9	2	7	1	8	5	6	3
----	---	---	---	---	---	---	---	---	---



Belangrijk om te vermelden: de stapelblokken zijn zo zwaar (of we doen toch alsof...) dat we ze écht geen twee keer willen verzetten als ze ook in één keer op hun juiste plaats kunnen gezet worden.

Laat de deelnemers er zelf achter komen dat er wel één extra stockeerplaats nodig is om een blok eventjes uit de rij te halen, maar van zodra er één gaatje vrij is, kan het sorteren beginnen. Een tweede stockeerplaats is niet nodig.

Extra oefening: laat elke deelnemer diezelfde sorteeroefening doen op een andere startsituatie. Gebruik Rummikubblokjes of speelkaarten. Hoeveel keer moet er een blok verzet worden? Wat is het minimum aantal 'bewegingen' om een rij die niet op volgorde stond, te sorteren? (Kan het dat je twee bewegingen nodig hebt?) Kan je een formule opstellen om te bepalen hoeveel bewegingen er nodig zijn voor een gegeven startsituatie? De term 'cykel' kan handig zijn: in het gegeven voorbeeld liggen de blokken 5, 7 en 8 in een cykel want 7 ligt op de plaats van 5, 5 ligt op de plaats van 8, 8 ligt op de plaats van 7. Dan is het cirkeltje of de cykel rond.

... EN MEER

Zo zijn er nog veel meer sorteermethodes die als basis van een doe-oefening kunnen dienen.

Kijk op <https://www.youtube.com/watch?v=kPRAOW1kECg> voor een visualisatie van sorteeralgoritmes, of zoek varianten hiervan op het net.

waarom zouden we wakker liggen van sorteren?



ENKELE RAADSELTJES OM HET NUT VAN SORTEREN TE ILLUSTREREN

Laat de deelnemers / leerlingen een getal tussen 1 en 20 raden. Als Murphy in de klas zit, hebben ze 20 gokken nodig. En als ze alle getallen door elkaar roepen, hebben ze ook een goed geheugen nodig. De computer is slimmer en luier (luiheid = goede eigenschap van een programmeur, mits op het gepaste moment ingezet). Hij zal van 1 tot 20 op volgorde raden, zo moet hij altijd maar 1 getal onthouden. Dit zoeken noemen we **lineair** (in een lijntje).

Laat de deelnemers / leerlingen een getal tussen 1 en 100 raden. Nu zeg je na het eerste getal wél “hoger” of “lager”. Dit komt neer op **binair** zoeken: we kunnen bij elke verkeerde gok de verzameling van mogelijkheden halveren. Als Murphy in de klas zit, moeten we toch slechts 7 gokken doen.

Een laatste zoekopdracht: teken een tabel van 10 vakjes op bord, genummerd vanaf 0. Leer de deelnemers aan wat de ‘**recursieve cijfersom**’ betekent: hiervoor blijf je de som van de cijfers van een getal berekenen, tot je een cijfer (dus een getal < 10) uitkomt. Een voorbeeld: 753 levert als eerste cijfersom $7+5+3 = 15$ op, wat op zijn beurt $1+5 = 6$ oplevert als cijfersom.

Hang nu post-its in de tabel, waarop (op de achterkant) getallen staan. De plaats waar je een getal hangt, bereken je aan de hand van de recursieve cijfersom. Zo zal 0 in het eerste vakje hangen, de getallen 1, 10, 100,... in het tweede vakje etcetera. Laat een paar vakjes in de tabel leeg, en hang niet meer dan vier post-its in één vakje. Laat de leerlingen nu zoeken naar een paar getallen. (Eerst eentje dat er wel in zit, dan eentje dat er niet in zit,...) De snelheid waarmee ze het juiste getal vinden, hangt af van hoeveel post-its er in één vakje hangen. Met andere woorden: als je een ‘formule’ vindt die alle getallen zo gelijkmatig verdeelt, EN je maakt de tabel veel langer (minstens twee keer zo lang als het aantal getallen dat je weg wil steken), dan vind je een getal quasi onmiddellijk.

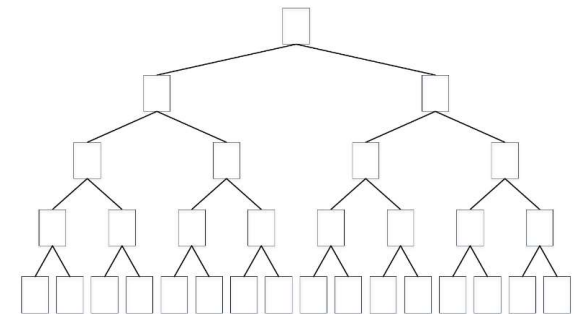
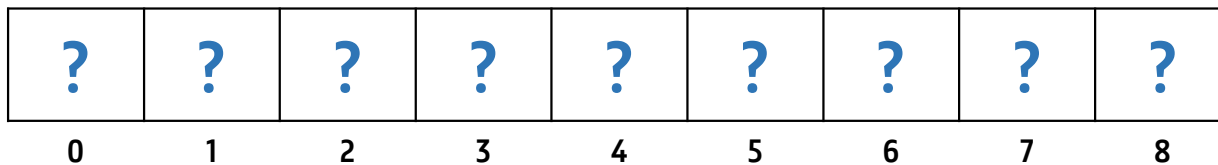
Dus: de voorgestelde formule (‘bereken de recursieve cijfersom’) is GEEN goede formule, en de tabel is ook veel te kort. Maar het principe, **hashing** genoemd, is wel heel sterk. In volgende slides wordt de gelijkenis doorgetrokken naar het vinden van het juiste bestek voor het ontbijt. Ook met de ogen nog half dicht vind je de lepels, omdat je weet dat ze énkél in dat ene bakje kunnen liggen. Is dat bakje leeg, dan moet je in de vaatwasser of tussen de vaat op het aanrecht gaan zoeken.

waarom zouden we wakker liggen van sorteren?

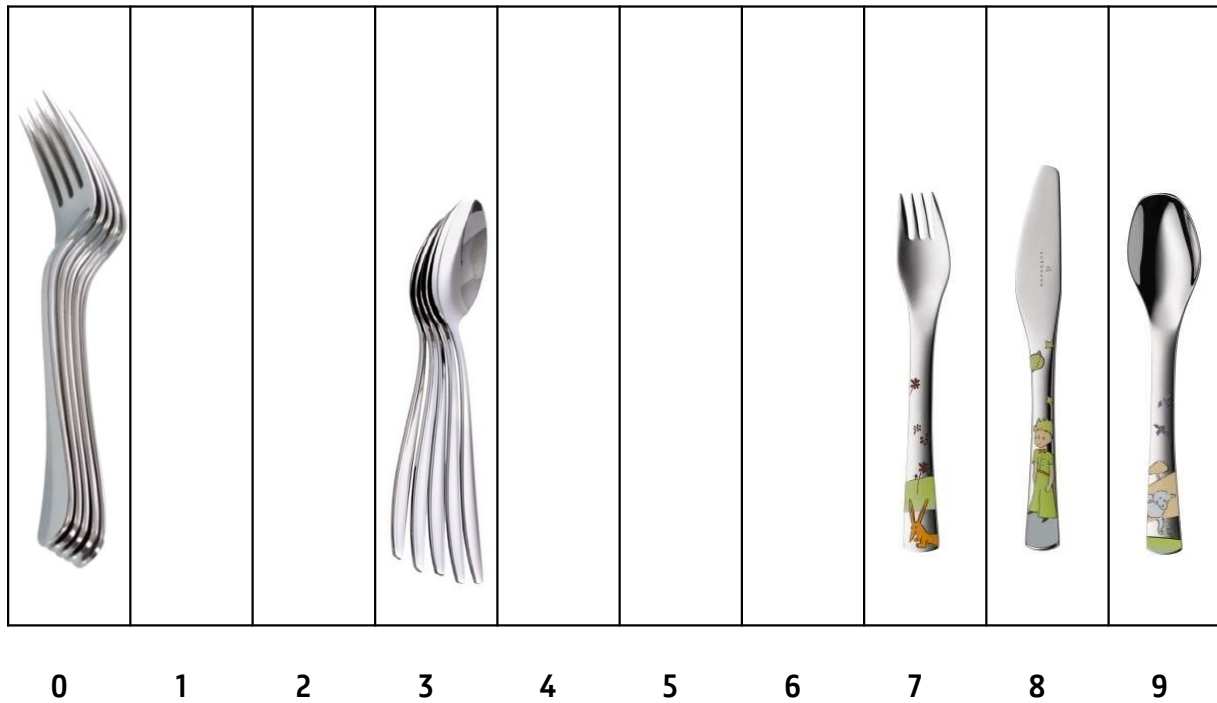
- áls alles gesorteerd is,
hoe vind je dan snelst wat je zoekt?



- áls alles gesorteerd is,
maar je kent de inhoud niet...
hoe vind je dan snelst wat je zoekt?

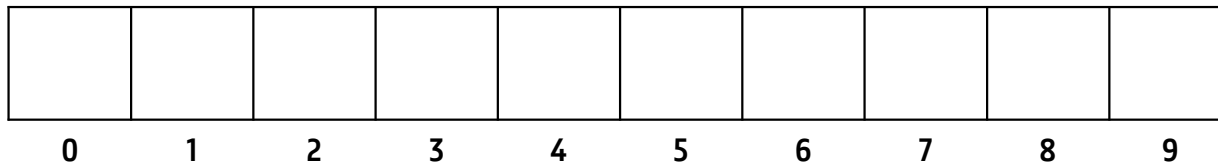


waarom zouden we wakker liggen van sorteren?



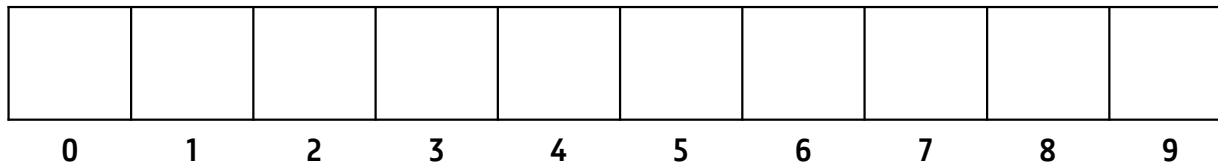
- waar leg jij je messen, vorken en lepels?

waarom zouden we wakker liggen van sorteren?



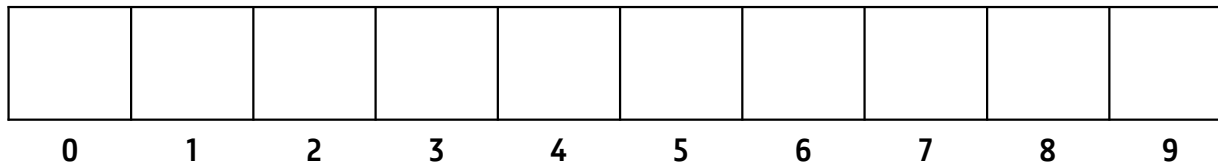
- steek hier de getallen **2 3 4 5 8 9** in weg
waar zal je ze het snelst vinden?
- steek hier de getallen **20 30 40 50 80 90** in weg
waar zal je ze het snelst vinden?
- steek hier de getallen **1002 1003 1004 1005 1008 1009** in weg
waar zal je ze het snelst vinden?

waarom zouden we wakker liggen van sorteren?



- steek hier de getallen **1002 1003 104 105 1008 1009** in weg
waar zal je ze het snelst vinden?
- steek hier de getallen **1002 102 1003 103 105 1005 1009** in weg
waar zal je ze het snelst vinden?
- steek hier de getallen **152 59 896 123 5333 122** in weg
waar zal je ze het snelst vinden?

**deze techniek noemen we hashing (mengelmoes,
gehakt, medley...)**



- verzin vooral een juiste 'toverspreuk'
of **formule**
- **zoekmachines** gebruiken dit constant!

BINAIR ZOEKEN VRAAGT GESORTEERDE GEGEVENS

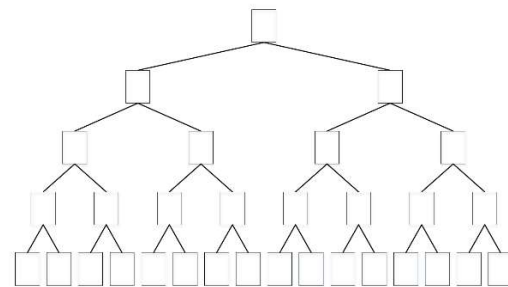
Omdat binair zoeken gesorteerde gegevens vraagt, is sorteren dus een belangrijk topic. We gaven al veel voorbeelden van sorteeralgoritmes en mogelijke oefeningen daarop. Maar tot nu toe werd alles weggestoken in een tabel, een lineaire structuur: alles staat op een rijtje. Indien we gegevens wegsteken in een binaire structuur, eentje die beter het binair zoeken weerspiegelt, dan komen we uit bij de zoekbomen.

Een boom is een datastructuur bestaande uit verschillende knopen (waar informatie in opgeslagen kan worden) en verbindingen tussen deze knopen. Elke boom bevat een 'bovenste knooppunt', de wortel genoemd, van waaruit één of meerdere verbindingen naar onderliggende knopen vertrekken. Elke knoop op zijn beurt kan ook weer verbindingen naar onderliggende knopen bevatten, maar er zijn nooit verbindingen die terugkeren of naar knopen op hetzelfde niveau verwijzen. (Dan zou je immers een rondje kunnen lopen in de structuur, en dan noemen we de structuur geen boom meer maar een graaf.)

Een binaire boom is een boom waarbij elke knoop maximaal twee verbindingen naar kinderen heeft.

Een zoekboom is een binaire boom waarbij elke knoop aan volgende voorwaarde voldoet: zijn linkerkind (indien bestaand) bevat informatie die kleiner is dan de eigen informatie, en het rechterkind (indien bestaand) bevat informatie die groter is dan de eigen informatie.

Hiernaast zie je een speciale binaire boom met 5 niveaus: alle niveaus zijn in deze boom volledig gevuld. Dat hoeft niet per se zo te zijn, er mogen knopen ontbreken.



OEFENING: ZOEKBOOM OPVULLEN (individueel)

Materiaal: per deelnemer een set Rummikubblokjes (nrs 1 tot 13) en een A3-papier met de boomstructuur die hieronder staat. Elk vakje heeft ongeveer de grootte van een Rummikubblokje.

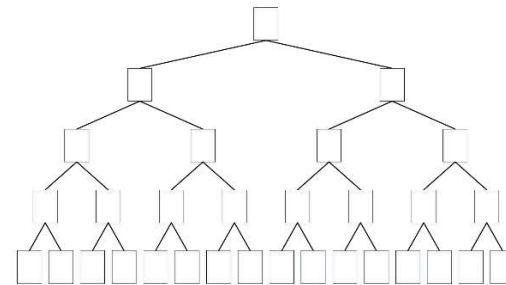
Opdracht: Laat de deelnemers willekeurige Rummikubblokjes uit het zakje trekken, en in die volgorde toevoegen aan de boom. Opgelet: de ingang van de boom is de wortel, het vakje dat bovenaan staat. Enkel langs die weg mag een blokje toegevoegd worden. Is dat vakje al bezet, dan wordt er verder gezocht in de juiste richting (links als het te plaatsen blokje kleiner is dan het blokje dat er al ligt, rechts in het andere geval) tot er een open vakje gevonden wordt. Het is goed mogelijk dat het blad te klein is, en dat de blokjes 'uit de boom vallen'.

Laat de deelnemer nadien zoeken naar het getal 8.5. (Dat zit er niet in, dat weten we.) Laat hem/haar aanduiden op welke plaats in de boom dat het 100% zeker is dat dit getal niet in de boom zit. Iedereen zou zijn vinger op een LEEG vakje moeten hebben!! (Niet op het vakje met blokje 8 of blokje 9, dan weet je nog niet zeker dat 8.5 er niet in zit!)

Laat de deelnemers bij elkaar gaan kijken: welke vorm kan de boom allemaal aannemen?

Een onevenwichtige boom met een paar lange takken en veel lege plaatsen is niet optimaal: het zoeken naar een bepaald getal kan dan toch nog lang duren, in extreme gevallen zelfs zolang als in een tabel.

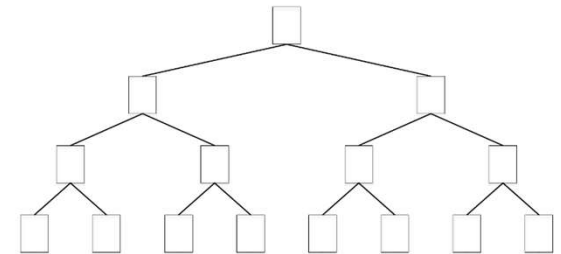
Als de boom zo evenwichtig mogelijk gevuld is, dus de eerste niveaus dicht bij de wortel zijn volledig opgevuld, dan zal de gemiddelde zoektijd naar een getal kort zijn.



OEFENING: BINAIRE BOOM OVERLOPEN (individ)

Materiaal: per deelnemer een half A4-blad met een binaire boom met 4 volledige niveaus en lege knopen.

Vraagstelling: Een binaire boom met 4 volledige niveaus heeft 15 knopen. Als we in die boom de getallen 1 tot 15 willen bewaren zoals het in een zoekboom betaamt (linkerkind kleiner dan ouder, rechterkind groter dan ouder), welk getal steken we dan best in de wortel? En welk getal komt dan links resp. rechts onder de wortel? Laat de boom volledig opvullen, zodat hij voldoet aan de zoekboom-voorwaarde. Dit is een voorbeeld van een zeer evenwichtige boom.



Volgende opdracht: overloop nu alle knopen in de boom, zonder er eentje over te slaan of dubbel te benoemen, en lees het getal luidop voor. Welke 'leesrichting' werd aangehouden? Niveau per niveau, van links naar rechts? Dit noemen we **levelorder**.

Laat daarna alle getallen van 1 tot en met 15 aflopen. Welke structuur kunnen de deelnemers ontdekken bij het overlopen van de knopen in die volgorde? (Tip voor de lesgever: er wordt in steeds groeiende 'bergjes' gelezen.)

Geef daarna de niet-volledige boom hiernaast, en laat de leerlingen die boom luidop overlopen. Hoor je een duidelijke zin? (Let op: de momenten zijn kostbaar, de verzameling niet!) Dan werd de boom in **inorder** doorlopen:

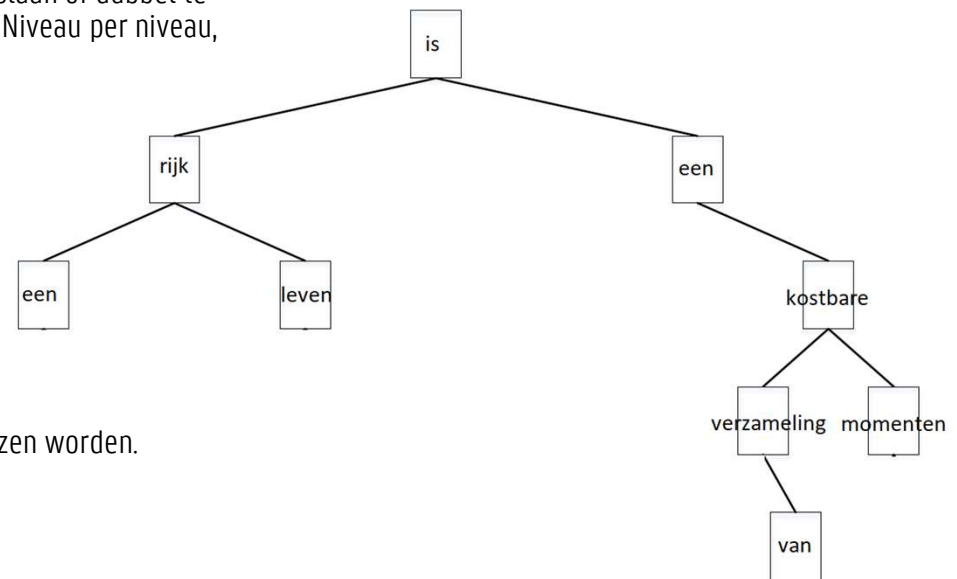
voor elke knoop die je wil voorlezen, lees je EERST

zijn linkerkind, dan de knoop zelf, dan zijn rechterkind. Dat is ook de volgorde

die gebruikt wordt als in een zoekboom de getallen in oplopende volgorde afgelezen worden.

Voor **postorder** lees je eerst linkerkind, dan rechterkind, dan de knoop zelf.

Voor **preorder** lees je eerst de knoop zelf, dan linkerkind, dan rechterkind.



OEFENING: BINAIRE BOOM DOORGEVEN

(individueel en per twee)

Materiaal: per deelnemer een blad met een binaire boom waar al letters in staan; op de volgende slides vind je voorbeelden (bewaars de afbeeldingen en gebruik ze gerust als lesmateriaal).

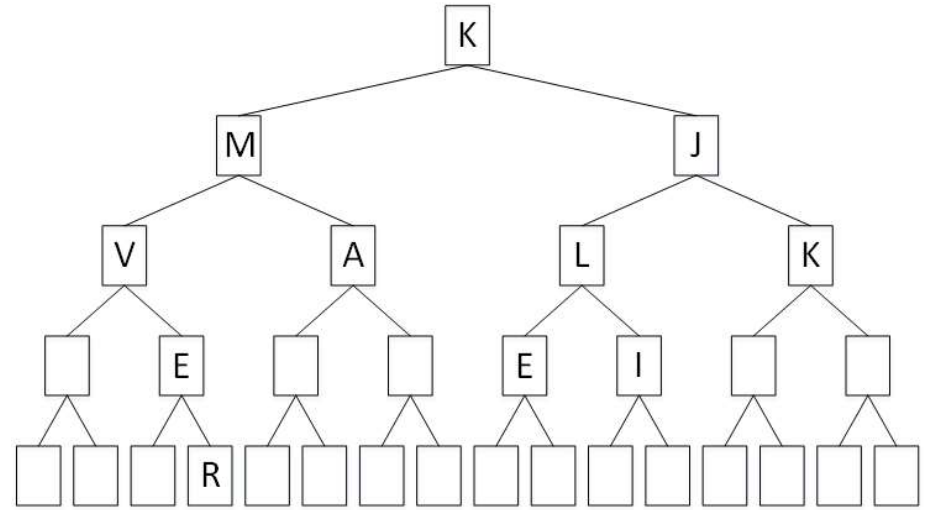
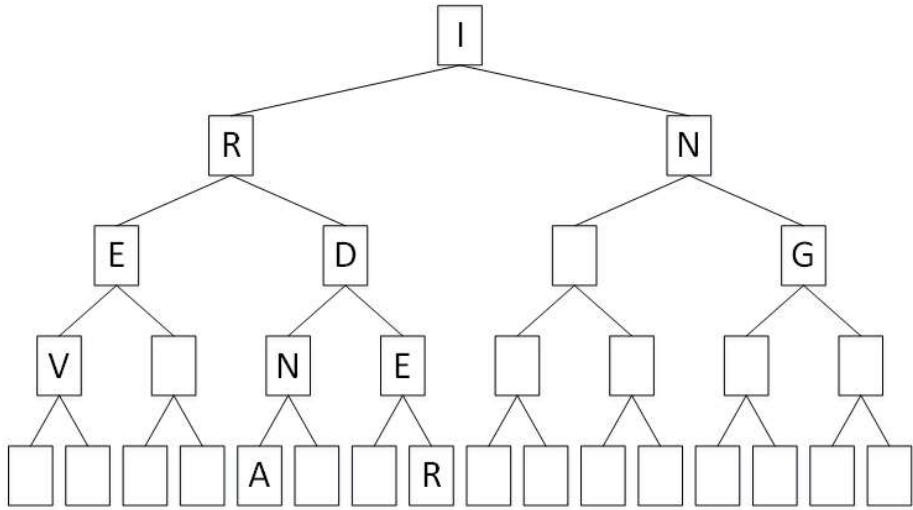
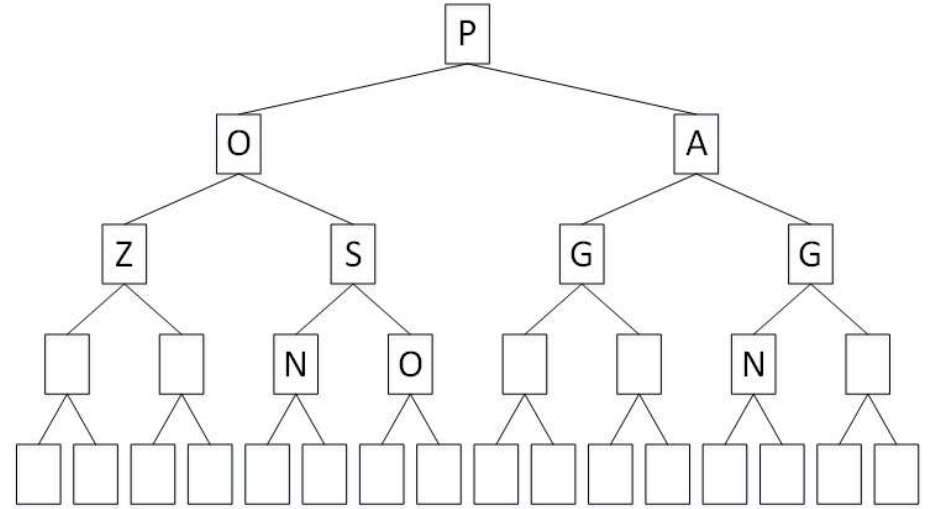
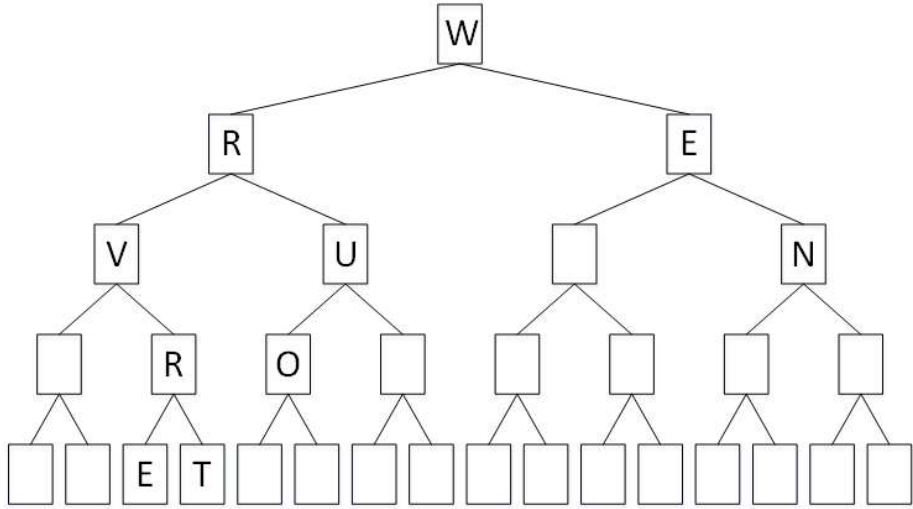
Duiding vooraf: als een computerprogramma veel informatie moet verwerken, dan wordt die informatie best op een gestructureerde, uniforme manier aangeboden. Zo zullen de gegevens van personen in bijvoorbeeld een tekstbestand aangeboden worden, waarbij op elke regel alle informatie van één persoon gegeven wordt, telkens met dezelfde onderdelen.

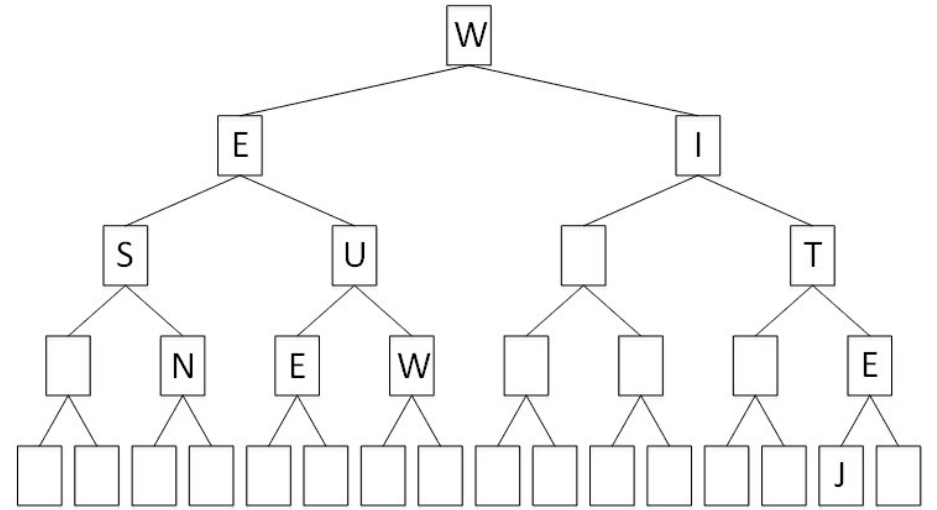
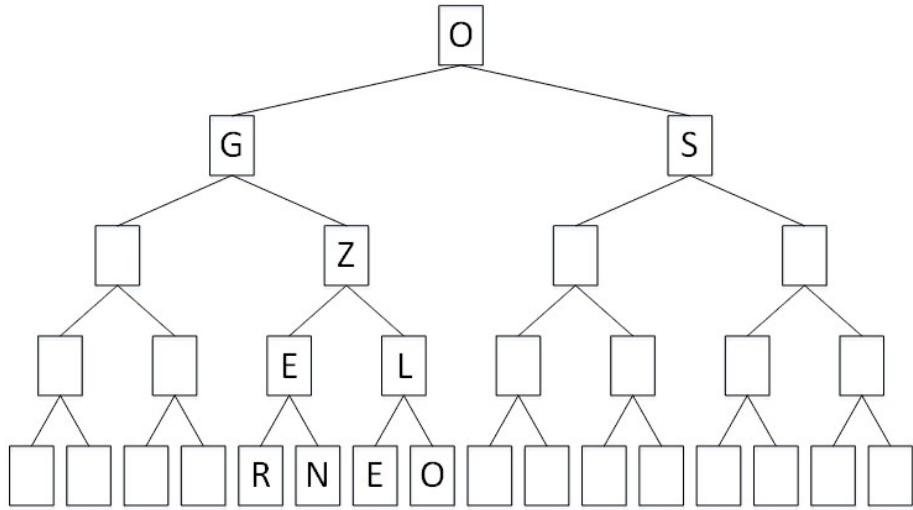
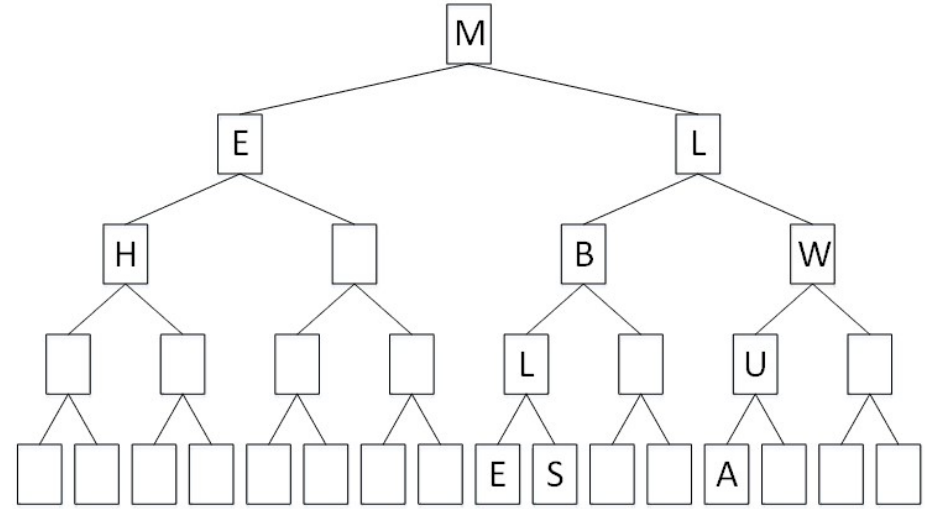
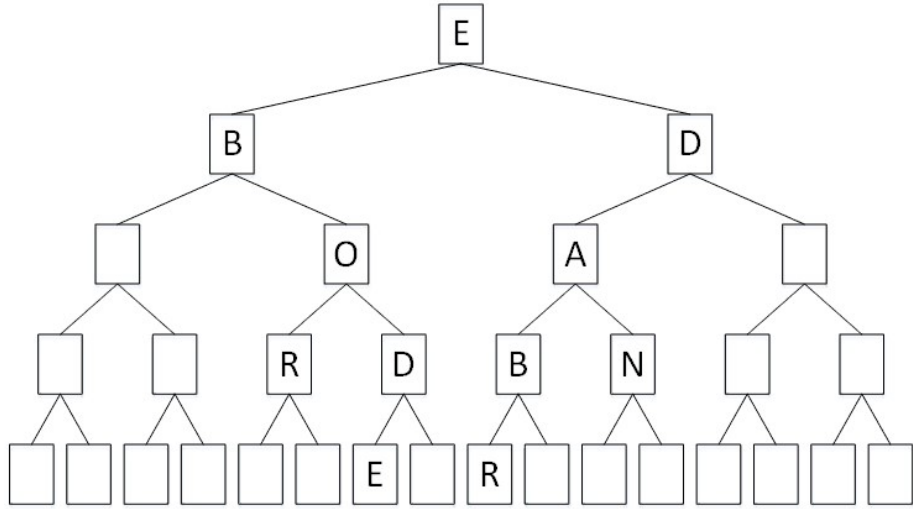
```
Jan;Janssen;Azalealei;72A;2000;Antwerpen  
Pieter-Jan;Peeters;Voskenslaan;122;9000;Gent  
Mehdi;Günsür;Begoniastraat;5;9000;Gent
```

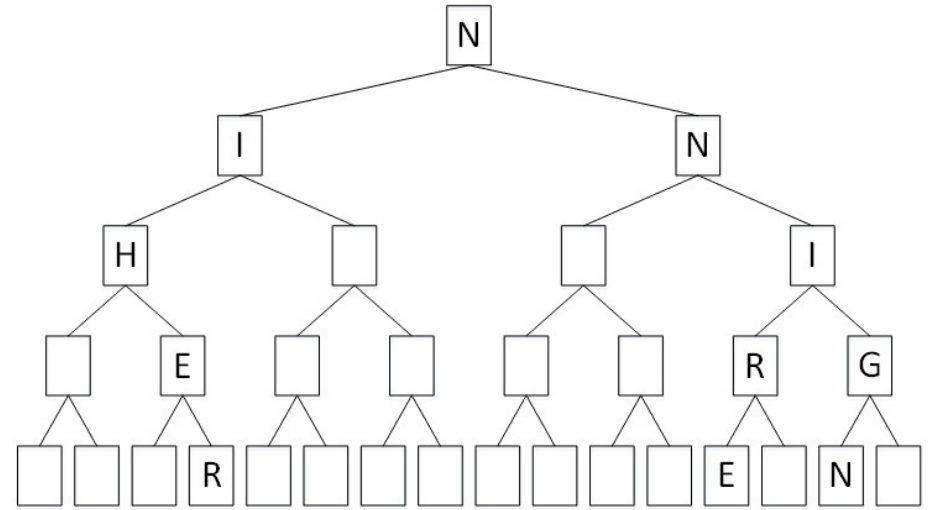
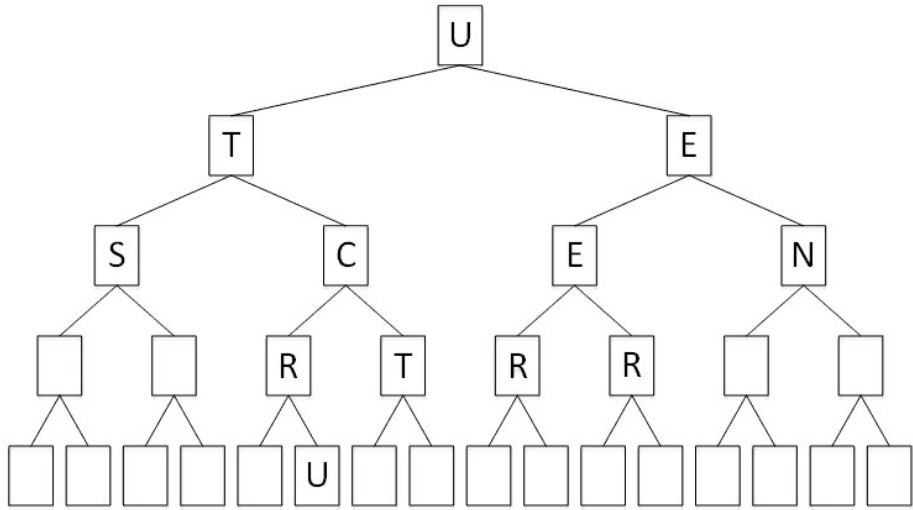
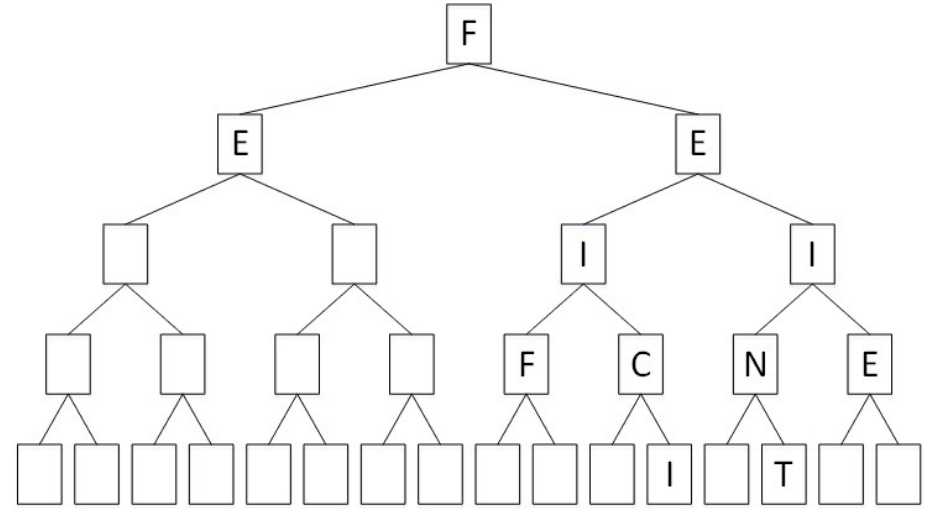
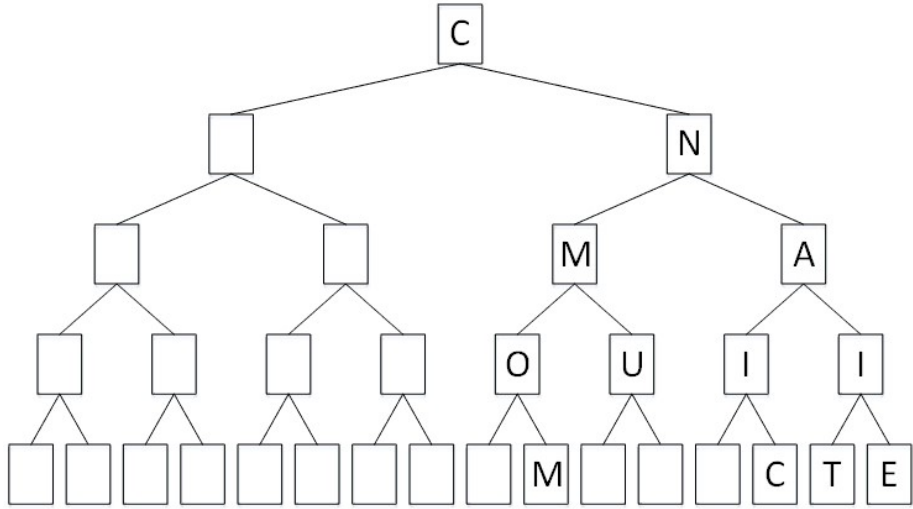
Iets dergelijks zoeken we nu. Hoe kan de structuur van de boom die voor je ligt in een tekstbestand gegoten worden zo dat een andere deelnemer jouw boom exact kan nabouwen aan de hand van dat tekstbestand? Voor elke knoop mag je één regel schrijven, waarin getallen en/of letters voorkomen. Je mag ook, op de achterkant van het tekstbestandje, een handleiding neerpennen. De handleiding voor het tekstbestand met personen hierboven zou zijn: op elke regel vind je deze informatie (in die volgorde): voornaam;familienaam;straat;huisnummer;postcode;stad of gemeente.

De oplossing van dit raadsel werd gegeven in de workshop.

Laat daarna de leerlingen hun boom ook nog eens in in-order lezen: dan zouden ze een woord moeten ontcijferen.







OEFENING: STAMBOOM OPSTELLEN

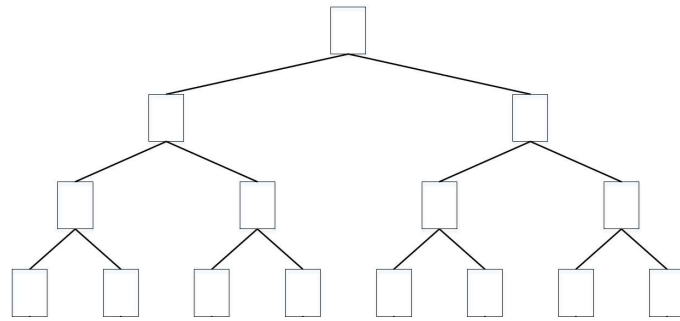
Hier volgen een aantal slides die kunnen gebruikt worden bij de oefening 'stamboom opstellen'.

Het idee: je geeft de relaties tussen ouders en kinderen op een papiertje. De leerlingen stellen de stamboom samen. Dat zal eerst wat werk in het klad vragen, omdat het niet onmiddellijk duidelijk is wie helemaal bovenaan de stamboom moet komen. Tenzij je eerst een 'truukje' vindt om toch na te gaan wie bovenaan de stamboom staat!

Deze oefening kan individueel opgelost worden (elk kind krijgt een lijst 'relaties' en een lege boom) of in groep (elk kind krijgt één relatie, en moet zijn plaats in de boom zoeken). Dat laatste kan werken, maar is niet evident! De individuele oefening is rustiger en overzichtelijker als de klas te groot is.

STAMBOOM OPSTELLEN

An, ouder 1 van Els
Els,ouder 1 van Jen



input

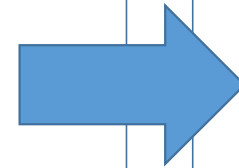
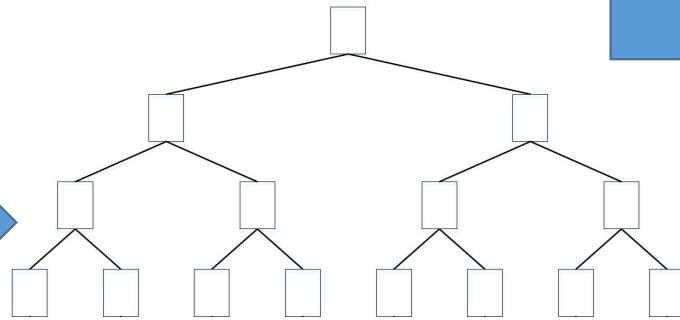
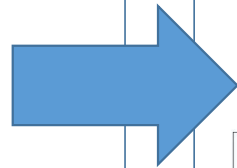
geheugen

output

STAMBOOM OPSTELLEN

RECURSIE

An, ouder 1 v nr 2
Els,ouder 1 v nr 3
Jen, ouder 1 v nr -1



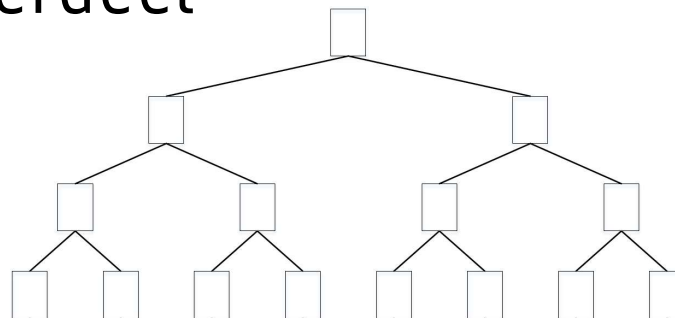
input

geheugen

output

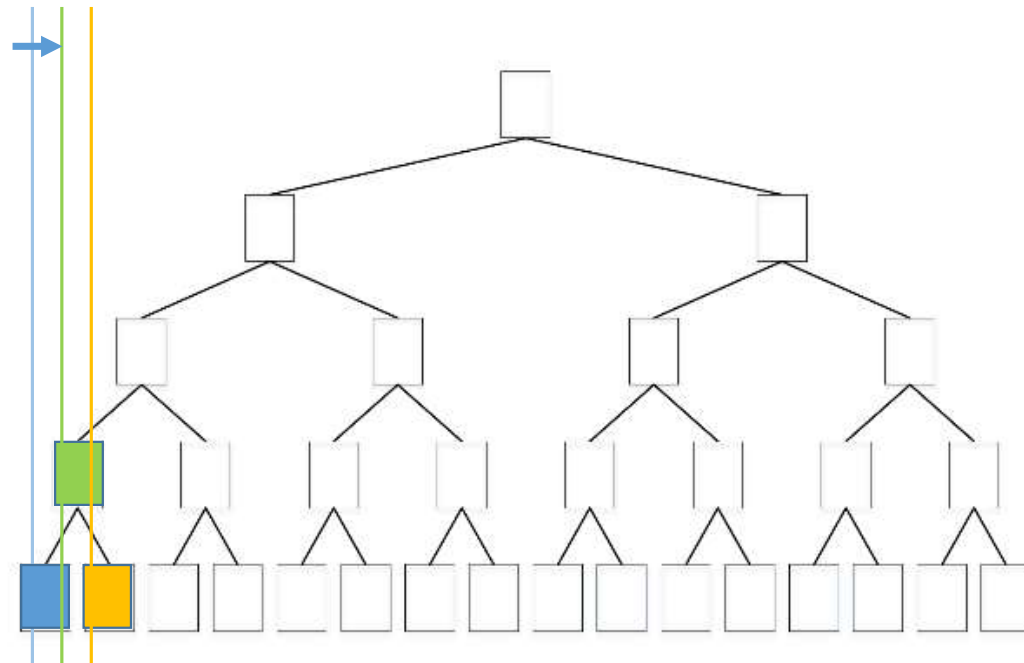
RECURSIE

Het optreden van een
constructie als onderdeel
van zichzelf

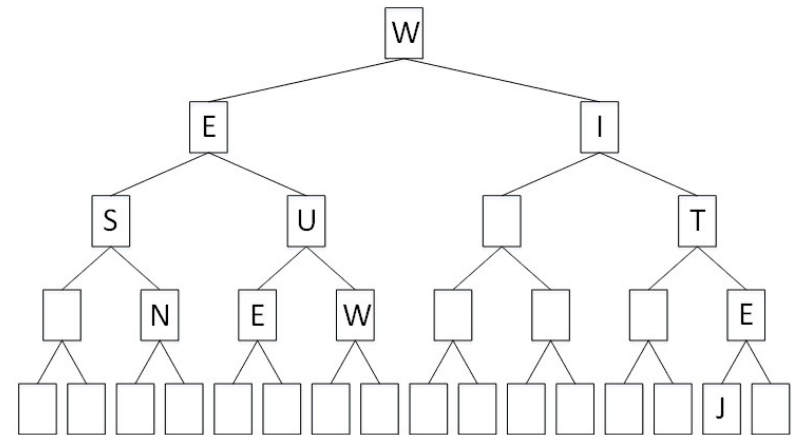


STAMBOOM OPSTELLEN: voorbeeld van individuele opdracht

Op de volgende slide een mogelijke 'stamboom'. Ter controle: als de stamboom wordt afgelopen in 'inorder' (of nog: je start met een verticale lat aan de knoop linksonder en verschuift de lat stilletjeaan naar de knoop rechtsonder; elke keer als je een knoop tegenkomt schrijf je zijn inhoud neer) dan vormen de eerste letters van de naam een sprookjesfiguur.



Nele is ouder 2 van Sam
 Will is ouder 2 van Ulrike
 Ulrike is ouder 2 van Erik
 Erik is ouder 1 van Wout
 Sam is ouder 1 van Erik
 Eduard is ouder 1 van Ulrike
 Tess is ouder 2 van Imra
 Eva is ouder 2 van Tess
 Jamal is ouder 1 van Eva
 Imra is ouder 2 van Wout



OEFENING: RECURSIE (in groep)

Vorbereiding: neem een lange strook (ruitjes)papier, zet er om de 4 cm willekeurige getallen op, plooi de strook toe, en zet op de buitenkant markeringen die de vakjes scheiden van elkaar. Voor een groep met x deelnemers, heb je $x+1$ getallen nodig.

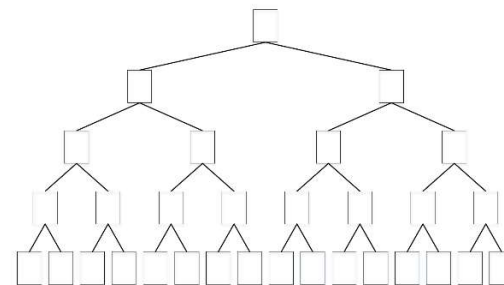
Vraag de jarige (van de dag, week of maand) naar voor. Zijn/haar opdracht: het grootste getal zoeken in het rijtje. Maar omdat de jarige het feestvarken is, doet die dat niet zelf: veel te veel werk. Hij/zij mag de strook in twee delen scheuren (langs een markering), en uitdelen aan twee helpers die voor de jarige komen staan. Maar omdat ook zij binnenkort (binnen de 365 dagen) jarig zijn, spelen zij ook het werk door aan twee helpers. Tot de strook bijvoorbeeld nog maar drie getallen telt: dan geef je er twee door aan een helper, en houd je het derde. Een helper die twee getallen vast heeft, is de pineut. Die moet het werk opknappen: strookje openen, en het grootste getal teruggeven aan afzender. Vanaf dan doet iedereen die twee getallen in de hand houdt of krijgt, hetzelfde: het grootste doorsturen richting oorspronkelijke jarige.

Dit illustreert het principe van recursie. Voor het schrijven van een recursieve functie moet je met twee mogelijkheden rekening houden: ofwel is het probleem simpel en dan los je het zelf op, ofwel is het probleem te groot en dan geef je het (in een licht vereenvoudigde versie weliswaar) door aan een of meerdere helpers. De oplossing die je van hen krijgt, moet je dan alleen nog verwerken tot het gewenste resultaat.

Merk in de tekening hieronder de verschillende recursieniveaus. De 'recursieboom' zal ook niet per se eindigen op een volledig opgevuld niveau: het laatste niveau vertoont allicht gaten. Zitten er hogerop in de boom ook gaten, dan werd het probleem allicht onevenwichtig verdeeld.

Tip: vooral fijn in een theaterzaaltje, met trappen.

Duiding: deze boomstructuur zal je niet expliciet in het geheugen van de computer terugvinden, maar wordt impliciet wel gevolgd bij het afhandelen van recursieve functie-oproepen.



WAAROM?

Waarom al deze oefeningen?

Ze dienen al excuus en body om de vier goede vaardigheden vermeld in een vorige slide in te oefenen:

- nadenken over het eigen nadenken
- structuur zoeken en vinden
- uitleggen en checken of de boodschap goed aankwam
- werken – maar dan efficiënt

Daarnaast is het ook belangrijk dat leerlingen / studenten inzien dat een opdracht gegeven in de les ('de les' = in eender welk vak) dikwijls een puzzel is die opgelost moet worden, en dat je puzzels met handen en voeten mag oplossen – tot je echt weet waar de addertjes onder het gras zitten.

Volgende slide illustreert dat: opgave en oplossing bestaan dikwijls uit tekst, maar als de lerende niet via de puzzel passeert die diep daaronder verscholen zit, zal het oplossen van de opdracht niet het gewenste leereffect hebben.

In het vak Algoritmen of bij het Computacionele denken komen er dikwijls oefeningen voor die geen pasklaar 'oplossingsmodel' hebben. Vergelijk het met de analyse van een reële functie: daarvoor is een heel specifiek stappenplan, een oplossingsmodel dat alleen kan toegepast worden op de analyse van functies. We geven toch een stappenplan voor het vak 'Computacioneel denken', al is dit veel vager. Daardoor is het wel breder inzetbaar, voor het ruime scala aan opdrachten dat binnen dit vakgebied te vinden is.

VOORBEELD

- Gegeven: een 'lottotrommel' die telkens een getal uit $\{1,2,3,\dots,999,1000\}$ genereert – dus mogelijk twee (of zelfs drie) keer hetzelfde getal!
- Gevraagd: een methode om 30 *verschillende* getallen uit $\{1,2,3,\dots,999,1000\}$ te genereren.
- Tip: gebruik pen en papier
want de computer doet het niet.
- Gebruik het stappenplan!

OPLOSSING

12	4	60	3	9	78				
-----------	----------	-----------	----------	----------	-----------	--	--	--	--

1	2	3	4	5	6	7	8	9	10	11	12	13
		V	V					V			V	

MEDIAAN ZOEKEN GEBASEERD OP QUICKSORT

Deze oefening komt in sommige workshops aan bod. Ze wordt hier (nog) niet uitgewerkt.

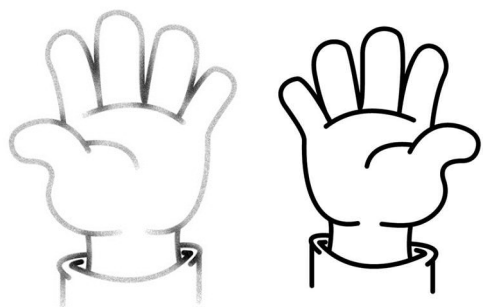
Voor wie zelf nieuwsgierig is: zoektermen 'quicksort' en 'quickselect' lichten een tipje van de sluier.

vind de mediaan - sorteer niet meer dan nodig

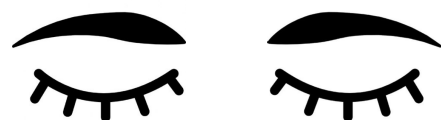


Wie is het slimst:

de computer
of de mens?



Niet zo handig



Een beetje blind



Tot slot:

wie graag dit bestand in powerpoint-formaat heeft,
kan dit vinden op <https://users.ugent.be/~lbrouns/>,
de 'verzamelplaats' voor workshops die ik geef.

Informatie over het aanbod is te vinden via
<https://www.ugent.be/educatief-aanbod/nl>

of met een mailtje naar leen.brouns@ugent.be