Formeel Denken

Henk Barendregt
Bas Spitters

Herfst 2002

Contents

1. Talen; Opsombare verzamelingen

Het genereren van een verzameling woorden is een belangrijke toepassing van computers. Op deze manier kunnen puzzels en belangrijkere zaken opgelost worden.

1.1. Definitie. (i) Een *alfabet* is een verzameling Σ van symbolen.
(ii) Een *woord* over Σ is een eindig rijtje sybolen uit dit alfabet.
(iii) Σ^* is de verzameling van alle woorden over Σ.
(iv) Een *taal* L over Σ is een deelverzameling van Σ^*.

1.2. Voorbeeld. (i) $\Sigma = \{a, b\}$ is een alfabet.
(ii) *abba* zit in Σ^* (notatie: $abba \in \Sigma^*$).
(iii) $abracadabra \notin \Sigma^*$.
(iv) $abracadabra \in \Sigma_0^*$, met $\Sigma_0 = \{a, b, c, d, r\}$.
(v) λ het *lege woord* zit in Σ^* voor iedere Σ.

1.3. Definitie. (i) $\Sigma_1 = \{\mathsf{M}, \mathsf{I}, \mathsf{U}\}$
(ii) De taal L_1 over Σ_1 is als volgt gedefinieerd ($x, y \in \Sigma_1^*$)

axioma	MI
regels	$x\mathsf{I} \Rightarrow x\mathsf{IU}$ $\mathsf{M}x \Rightarrow \mathsf{M}xx$ $x\mathsf{III}y \Rightarrow x\mathsf{U}y$ $x\mathsf{UU}y \Rightarrow xy$

Dit betekent dat per definitie $\mathsf{MI} \in L_1$;
als $x\mathsf{I} \in L_1$, dan ook $x\mathsf{IU} \in L_1$,
als $\mathsf{M}x \in L_1$, dan ook $\mathsf{M}xx$,
als $x\mathsf{III}y \in L_1$, dan ook $x\mathsf{U}y$ en
als $x\mathsf{UU}y \in L_1$, dan ook xy.

1.4. Voorbeeld. (i) $\mathsf{MI}, \mathsf{MII}, \mathsf{MU}, \mathsf{MUU}, \mathsf{IMMIU}, \ldots \in \Sigma_1^*$
(ii) $\mathsf{MI}, \mathsf{MIU}, \mathsf{MIUIU}, \mathsf{MII}, \mathsf{MIIII}, \mathsf{MUI}, \ldots \in L_1$.

1.5. Probleem (Hofstadter's MU puzzel[1]). $\mathsf{MU} \in L_1$?

1.6. Exercise. Probeer de woorden in L_1 te klassificeren.

1.7. Definitie. (i) $\Sigma_2 = \{\mathsf{p}, \mathsf{q}, -\}$
(ii) De taal L_2 over Σ_2 is als volgt gedefinieerd ($x, y, z \in \Sigma_2^*$).

axioma's	$x\mathsf{pq}x$ als x alleen uit $-$'s bestaat
regel	$x\mathsf{p}y\mathsf{q}z \Rightarrow x\mathsf{p}y{-}\mathsf{q}z{-}$

1.8. Opgave. Welke van de volgende woorden zitten in L_2? Beargumenteer dit bij ieder antwoord.

[1] Zie [Hof79].

1. `--p--p--q----`
2. `--p--q--q----`
3. `--p--q----`
4. `--p--q------`

1.9. Opgave. Laat $\Sigma = \{a, b, c\}$. Definieer L door

axioma	ab
regel	$xyb \;\Rightarrow\; yybx$

Beantwoord de volgende vragen niet alleen met 'ja' of 'nee', maar geef ook een sluitende redenering.

(i) Geldt $ba \in L$?

(ii) Geldt $bb \in L$

Het begrip 'taal' lijkt flauw, maar is het niet.

1.10. Definitie. Laat $\Sigma_3 = \{a\}$.

(i) Definieer L_{31} als volgt

axioma	a
regel	$w \;\Rightarrow\; waa$

(ii) $L_{32} = \{a^p \mid a \text{ is een priemgetal}\}$. Hierbij is $a^0 = \lambda$ en $a^{n+1} = a^n a$. Je kunt ook zeggen $a^n = a \dots a$, met hier n keer a.

(iii) Definieer L_{33} als volgt

axioma	a
regel	$w \;\Rightarrow\; ww$ $wwwa \;\Rightarrow\; w$

Hoe kun je beslissen of een woord over Σ_3 in L_{31} zit? Bepalen of een woord in L_{32} zit is lastiger, maar het lastige zit al in de specificatie van de taal. L_{33} is een gemakkelijk gespecificeerde taal met een moeilijk beslissingsprobleem. Het is al een beetje lastig om te laten zien dat $aaa \in L_{33}$.

Uitdaging. Geldt $L_{33} = \Sigma_3^*$? Wie mij als eerste een bewijs of tegenvoorbeeld stuurt per email naar `<henk@cs.kun.nl>` krijgt 100 Euro. Inzendingen hebben een sluitingstermijn: 31.12.2002 om 24.00 (nieuwjaar 2003).

Grotere(?) uitdaging. De definitie 1.10 (iii) kan iets anders gegeven worden. Dit correspondeert met een bekend open probleem, het Syracuse probleem. Definieer L_{34} als volgt

axioma	a
regel	$w \;\Rightarrow\; ww$ $wwwaa \;\Rightarrow\; wwa$

Bewijs of weerleg $L_{34} = \{a^n \mid n \geq 1\}$. Voor de eerste juiste oplossing per email vóór 1.1.2003 krijgt de zender 150 Euro.

1.11. Opgave. (i) Laat zien dat $\lambda \notin L_{34}$.

(ii) Wat moet je doen om in één klap 250 Euro te verdienen op grond van de twee uitdagingen? Deze opgave is niet zo moeilijk en moeten jullie kunnen oplossen. Voor de eerste juiste oplossing per email vóór 1.1.2003 krijgt de zender 10 Euro.

Er is een andere manier om talen aan te geven.

1.12. Voorbeeld. $\Sigma = \{a, b\}$. De taal $L \subseteq \Sigma^*$ wordt gedefinieerd door producties vanuit S (start).

$$\begin{array}{rcl} S & \to & aSb \\ S & \to & \lambda \end{array}$$

De manier om nu woorden te genereren is als volgt. We beginnen bij S (start). Dit is het enige *hulpsymbool*. We volgen de pijl (twee mogelijkheden). Indien er nog een hulpsymbool staat volgen we nog een pijl. Totdat er geen hulpsymbool meer staat en dan hebben we een woord geproduceerd. Voorbeelden van producties:

$$\begin{array}{l} S \to aSb \to aaSbb \to aabb; \\ S \to aSb \to aaSbb \to aaaSbbb \to aaabbb. \end{array}$$

Deze grammatika wordt ook wel weergegeven als

$$S \to aSb \mid \lambda$$

Dit levert op als taal $L = \{\lambda, ab, aabb, aaabbb, a^4b^4, \ldots, a^nb^n, \ldots\}$. We schrijven ook wel $L = \{a^nb^n \mid n \in \mathbb{N}\}$ met $\mathbb{N} = \{0, 1, 2, 3, \ldots\}$. Dit klopt, want $a^0b^0 = \lambda\lambda = \lambda$.

Talen op deze manier geproduceerd heten *contextvrije talen*. Ze worden systematisch behandeld in het college T1a voor de studierichting Informatica.

1.13. Opgave. Gegeven is de grammatika

$$\begin{array}{rcl} S & \to & aSb \mid A \mid \lambda \\ A & \to & aAbb \mid abb \end{array}$$

De hulpsymbolen zijn dit keer S en A. Probeer eens abb en aab te produceren. Welke woorden zitten in de gegenereerde taal?

1.14. Opgave. Geef de talen uit 1.12 en 1.13 weer met behulp van axioma's en regels zoals in 1.3.

We geven een grammatika voor een klein gedeelte van het Engels.

$S = \langle\text{zin}\rangle \to \langle\text{onderwerpvorm}\rangle\langle\text{werkwoordvorm}\rangle$
$\langle\text{zin}\rangle \to \langle\text{onderwerpvorm}\rangle\langle\text{werkwoordvorm}\rangle\langle\text{lijdendvoorwerpvorm}\rangle$
$\langle\text{onderwerpvorm}\rangle \to \langle\text{eigennaam}\rangle \mid \langle\text{lidwoord}\rangle\langle\text{zelfstandignaamwoord}\rangle$
$\langle\text{eigennaam}\rangle \to \text{John} \mid \text{Jill}$
$\langle\text{zelfstandignaamwoord}\rangle \to \text{bicycle} \mid \text{mango}$
$\langle\text{lidwoord}\rangle \to \text{a} \mid \text{the}$
$\langle\text{werkwoordvorm}\rangle \to \langle\text{werkwoord}\rangle \mid \langle\text{bijwoord}\rangle\langle\text{werkwoord}\rangle$
$\langle\text{werkwoord}\rangle \to \text{eats} \mid \text{rides}$
$\langle\text{bijwoord}\rangle \to \text{slowly} \mid \text{frequently}$
$\langle\text{bijvoegelijknaamwoordlijst}\rangle \to \langle\text{bijvoegelijknaamwoord}\rangle\langle\text{bijvoegelijknaamwoordlijst}\rangle \mid \lambda$
$\langle\text{bijvoegelijknaamwoord}\rangle \to \text{big} \mid \text{juicy} \mid \text{yellow}$
$\langle\text{lijdendvoorwerpvorm}\rangle \to \langle\text{bijvoegelijknaamwoordlijst}\rangle\langle\text{eigennaam}\rangle$
$\langle\text{lijdendvoorwerpvorm}\rangle \to \langle\text{lidwoord}\rangle\langle\text{bijvoegelijknaamwoordlijst}\rangle\langle\text{zelfstandignaamwoord}\rangle$

1.15. Opgave. Laat zien hoe je de volgende zin produceert.

Jill frequently eats a big juicy yellow mango.

Maak zelf andere zinnetjes.

2. Logica 1; Propositielogica

In dit blok behandelen we de propositielogica, ook wel uitspraakrekening genoemd.

Formele taal en natuurlijke taal

Natuurlijke talen (Nederlands, Engels, Duits,...) zijn niet erg precies. Kijk bijvoorbeeld maar naar de volgende voorbeelden:

- Socrates is een mens. Een mens is sterfelijk. Dus Socrates is sterfelijk.
- Ik ben iemand. Iemand schilderde de Mona Lisa. Dus ik ben de schilder van de Mona Lisa.

De eerste redenering klopt, de tweede niet, maar heeft wel een soortgelijke vorm.

Is de zin

$$\boxed{\text{Deze zin is niet waar.}}$$

nu waar of niet?

Om dit soort problemen te voorkomen gebruiken we een formele taal. Dit is een soort laboratorium-model voor de natuurlijke taal. We zullen zien dat we met een heel eenvoudige kunst-taal, toch heel veel uitspraken en redeneringen heel precies kunnen maken. Dit is bijvoorbeeld erg belangrijk voor het specifieren van programma's, over de specificatie mag geen misverstand bestaan.

We zullen nu laten zien hoe we van het Nederlands naar een formele taal gaan. Om te redeneren combineren we vaak een aantal eenvoudige uitspraken, bijvoorbeeld: '*als* het regent *en* ik ben buiten, *dan* wordt ik nat'. Deze uitspraak is opgebouwd uit de eenvoudige uitspraken 'het regent', 'ik ben buiten' en 'ik wordt nat'.

Woordenboek

We kunnen dit ook formeler op schrijven met behulp van een woordenboekje.

R	het regent
Z	de zon schijnt
RB	er is een regenboog
N	ik word nat
D	ik blijf droog
Bui	ik ben buiten
Bin	ik ben binnen

De zin hierboven wordt dan 'als R en Bui, dan N'. Net zo, kunnen we de volgende zinnen maken: 'als RB, dan Z', 'Z en RB', 'als R en Bin, dan D'.

Verbindingswoorden

We kunnen ook de verbindingswoorden formeel opschrijven.

Dat doen we op de volgende manier:

Formele taal	Nederlands
$f \wedge g$	f en g
$f \vee g$	f of g
$f \rightarrow g$	als f, dan g
$f \leftrightarrow g$	f dan en slechts dan als g
$\neg f$	niet f

De zinnetjes hierboven worden dan 'RB→Z', 'Z∧RB' en '(R∧Bin)→D'.

Nederlands	semi-formeel	Formeel
als het regent *en* ik ben buiten, *dan* wordt ik nat	als R en Bui, dan N	$(R \wedge \text{Bui}) \rightarrow N$
als er een regenboog is, *dan* schijnt de zon	als R, dan Z	$R \rightarrow Z$
ik ben binnen *of* buiten	Bin of Bui	$\text{Bin} \vee \text{Bui}$

2.1. Opgave. Vind zinnen uit de formele taal die hetzelfde betekenen als de volgende zinnen.

1. Het regent niet, noch schijnt de zon.
2. De zon schijnt, tenzij het regent
3. Of de zon schijnt, of het regent. (We bedoelen dus niet allebei)
4. Er is alleen een regenboog als de zon schijnt en het regent.
5. Als ik buiten ben, dan wordt ik nat, mits het regent.

2.2. Opgave. Kun je $\leftrightarrow$ ook uitdrukken met behulp van de andere symbolen?

2.3. Opgave. Vertaal de volgende zinnen naar het Nederlands:

1. $R \leftrightarrow Z$
2. $\text{RB} \rightarrow (R \wedge Z)$
3. $\text{Bui} \rightarrow \neg\,\text{Bin}$
4. $\text{Bui} \vee \text{Bin}$

2.4. Definitie. De taal van de uitspraakrekening (of propositielogica) is de volgende formele taal. Het alfabet bestaat uit een oneindige verzameling

$$A := \{a, b, c, d, a_1, a_2, a_3, \ldots\}$$

van Atomen, de verzameling

$$V := \{\wedge, \vee, \rightarrow, \leftrightarrow, \neg\}$$

van Voegtekens en de verzameling

$$H := \{(,)\}$$

van Haakjes.
Woorden uit deze taal worden als volgt opgebouwd:

1. Een atoom is een woord.
2. Als f en g woorden zijn, dan zijn $(f \wedge g), (f \vee g), (f \rightarrow g), (f \leftrightarrow g)$ en $\neg f$ woorden.
3. Alle woorden worden op deze manier gevormd.

De formele definitie is als volgt: $\Sigma = A \cup V \cup H$, ofwel

$$\Sigma = \{a, b, c, \ldots, \wedge, \vee, \rightarrow, \leftrightarrow, \neg, (,), \}.$$

($\cup$ geeft de vereniging van twee verzamelingen aan).

$$S \to a \mid (SvS) \mid \neg S$$

$$v \to \vee \mid \wedge \mid \to \mid \leftrightarrow$$

hier mag a ieder element van A zijn.

Meestal laten we de buitenste haakjes weg, dus we schrijven bijvoorbeeld $a \wedge b$ in plaats van $(a \wedge b)$. De binnenste haakjes mogelijk we natuurlijk niet weg laten: vergelijk $(R \wedge Z) \to RB$ en $R \wedge (Z \to RB)$.

Betekenis en waarheidstabellen

De zin 'als a en b, dan a' is waar, wat je ook voor a en b invult. We zouden nu ook willen zeggen dat de zin '$(a \wedge b) \to a$' waar is, maar we hebben nog niet gedefinieerd wat dat betekend: '$(a \wedge b) \to a$' is zomaar een zin in een taal. We gaan nu definiren wat de betekenis van een zin is, in het bijzonder, wanneer een zin in de taal van de logica *waar* is.

Voor de atomen kunnen we aan eenvoudige uitspraken denken zoals '2=3', 'Jolly Jumper is een paard' of 'het regent'. In de klassieke logica, waar we ons in dit college toe beperken, nemen we aan dat alle atomen of waar zijn of niet waar zijn. We maken ons geen zorgen om het feit dat we soms niet weten of de zin waar is of niet waar is, zoals bij de zin, 'op 1 januari 2050 regent het in Nijmegen'.

De waarheid van de atomen wordt bepaald daar hun interpretatie in een *model*. Bijvoorbeeld '2=3' is niet waar in het model van de natuurlijke getallen. 'Jolly Jumper is een paard' is waar in het model van het stripboek Lucky Luck. De zin 'het regent' was niet waar in Nijmegen op 17 september 2002.

Bekijk nu eens de zin '$a \wedge b$' we willen dat deze zin alleen waar is in een model als zowel a als b waar zijn in dat model. Als we nu een lijstje maken met alle mogelijke waarden voor a en voor b, dan kunnen we ook bepalen wat de mogelijke waarden zijn voor $a \wedge b$.

In de informatica schrijven we vaak 1 voor *waar* en 0 voor *onwaar*. De logische operaties zijn de elementaire operaties op bits.

We krijgen dan de volgende tabel:

x	y	$x \wedge y$
0	0	0
0	1	0
1	0	0
1	1	1

Net zo krijgen we de volgende tabellen:

x	$\neg x$
0	1
1	0

x	y	$x \vee y$
0	0	0
0	1	1
1	0	1
1	1	1

x	y	$x \to y$
0	0	1
0	1	1
1	0	0
1	1	1

x	y	$x \leftrightarrow y$
0	0	1
0	1	0
1	0	0
1	1	1

2.5. Voorbeeld. De waarheidstabel van $(a \vee b) \to a$.

a	b	$a \vee b$	a	$(a \vee b) \to a$
0	0	0	0	1
0	1	1	0	0
1	0	1	1	1
1	1	1	1	1

2.6. Opgave. Schrijf de waarheidstabellen op van: $a \vee \neg a$, $(a \to b) \to a$, $a \to (b \to a)$, $(a \wedge b) \to a$, $a \wedge (b \to a)$, $\neg a \to \neg b$.

2.7. Definitie. Als een formule f altijd waar is, dat wil zeggen dat er in de waarheidstabel van f alleen maar 1-en staan, dan noemen we de formule *logisch waar*. Notatie $\vDash f$. Een logisch ware formule heet ook wel een *tautologie*.

2.8. Opgave. Welke van de volgende formules zijn logisch waar? $a \vee \neg a$, $a \to (a \to a)$, $a \to a$, $(a \to b) \to a$, $a \to (b \to a)$, $(a \wedge b) \to a$, $a \vee (b \to a)$,

2.9. Opgave. Laat f en g formules zijn. Ga na of de volgende uitspraken kloppen:

1. Als $\vDash f$ en $\vDash g$, dan $\vDash f \wedge g$.
2. Als niet $\vDash f$, dan $\vDash \neg f$.
3. Als $\vDash f$ of $\vDash g$, dan $\vDash f \vee g$.
4. Als (als $\vDash f$, dan $\vDash g$), dan $\vDash f \to g$.
5. Als $\vDash \neg f$, dan niet $\vDash f$.
6. Als $\vDash f \vee g$, dan $\vDash f$ of $\vDash g$.
7. Als $\vDash f \to g$, dan (als $\vDash f$, dan $\vDash g$).
8. Als $\vDash f \leftrightarrow g$, dan ($\vDash f$ dan en slechts dan als $\vDash g$).
9. Als ($\vDash f$ dan en slechts dan als $\vDash g$), dan $\vDash f \leftrightarrow g$.

Logisch Equivalent

2.10. Definitie. Twee formules zijn *logisch equivalent* als f waar is in een model, dan en slechts dan als, g waar is in dat model. Dat wil zeggen f en g hebben dezelfde waarheidstabel hebben. Als f en g equivalente formules zijn schrijven we ook wel $f \equiv g$.

Vaak kunnen we een formule vervangen door een eenvoudigere equivalente formule. Bijvoorbeeld $a \wedge a$ is equivalent met a.

2.11. Opgave. Laat zien dat de twee formules $(a \wedge b) \wedge c$ en $a \wedge (b \wedge c)$ logische equivalent zijn.

Laat zien dat de twee formules $(a \vee b) \vee c$ en $a \vee (b \vee c)$ logische equivalent zijn.

In dit geval doen de haakjes er dus eigenlijk niet toe. Soms laten we in zo'n geval de haakjes ook wel weg.

Let op: de formules: $a \wedge b$ en $b \wedge a$ zijn logisch equivalent. In het Nederlands is dit niet altijd eenduidig, met de zin 'ze trouwde en kreeg een kind' bedoelen we waarschijnlijk wat anders dan met de zin 'ze kreeg een kind en trouwde'.

De volgende equivalenties heten de wetten van *DeMorgan*.

1. $\neg(f \wedge g) \equiv \neg f \vee \neg g$.
2. $\neg(f \vee g) \equiv \neg f \wedge \neg g$.
3. $f \wedge (g \vee h) \equiv (f \wedge g) \vee (f \wedge h)$.
4. $f \vee (g \wedge h) \equiv (f \vee g) \wedge (f \vee h)$.

2.12. Opgave. Laat f en g formules zijn. Is de volgende uitspraak waar? $f \equiv g$ dan en slechts dan als $\models f \leftrightarrow g$. waar is.

Logisch gevolg

In het Nederlands volgt uit 'het regent en de zon schijnt' dat 'de zon schijnt'. Net zo willen we dat uit $a \wedge b$ volgt dat a. Dat maken we nu precies.

2.13. Definitie. Een formule g is een *logisch gevolg* van de formule f als g waar is in ieder model waar f waar is. Dat wil zeggen, als er op alle plaatsen waar in de waarheidstabel van f een 1 staat er in de waarheidstabel van g ook een 1 staat. Notatie $f \models g$.

2.14. Opgave. Zijn de volgende uitspraken waar? $a \wedge b \vDash a$, $a \vee b \vDash a$, $a \vDash a \vee b$, $a \wedge \neg a \vDash b$.

1. Stelling. *Laat f en g formules zijn.*

$$\vDash f \to g, \text{ dan en slechts dan als } f \vDash g.$$

Voor meer informatie over logica kun je bijvoorbeeld het boek [vBe] bekijken.

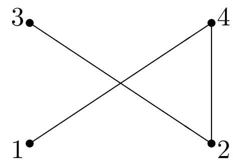
3. Blokjes; Combinatoriek

Graphen

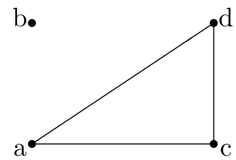
Dit hoofdstuk is een korte inleiding in de graphentheorie. Graphen kom je op allerlei plaatsen tegen bijvoorbeeld bij talen, netwerken, datastructuren, electrische circuits, transport problemen en stroomschema's.

Intuïtief bestaat een graph uit een verzameling P van punten en een verzameling L van lijnen tussen punten. Twee voorbeelden:

G_1



G_2



Natuurlijk zijn er nu allerlei knagende onzekerheden, zoals: 'Wanneer zijn twee graphen gelijk?', 'Moeten die punten in een plat vlak liggen?', 'Mogen de lijnen elkaar snijden?'. Al je twijfels verdwijnen als sneeuw voor de zon, dankzij de volgende formele definitie:

Definitie. Een *graph* is een struktuur $\langle P, L\rangle$ waarbij P een verzameling is, en L een verzameling van uit twee elementen bestaande deelverzamelingen van P. (de elementen van P noemen we 'punten', de elementen van L noemen we 'lijnen'; een lijn noteren we niet als $\{a, b\}$ maar als (a, b). De lijn (a, b) is dus gelijk aan de lijn (b, a).)
De graph G_1 uit ons eerste voorbeeld is dus $\langle P, L\rangle$ met $P = \{1, 2, 3, 4\}$ en $L = \{(1, 4), (2, 3), (2, 4)\}$.
De graph G_2 is de struktuur $\langle P, L\rangle$ met $P = \{a, b, c, d\}$ en $L = \{(a, c), (a, d), (c, d)\}$

Definities. (Hierin is G de graph $\langle P, L\rangle$, en zijn p en q punten)

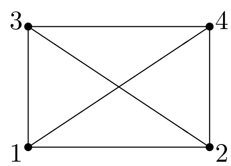
- Een *buur* van p is een punt x met $(p, x) \in L$.
- De *graad* (of: *valentie*) van p is het aantal buren van p.
- Een *pad* van p naar q is een rijtje van verschillende lijnen $(x_0, x_1), (x_1, x_2), \ldots, (x_{n-1}, x_n)$ met $x_0 = p$ en $x_n = q$. Een kortere notatie voor dit pad: $x_0 \to x_1 \to x_2 \to \cdots \to x_n$.
- Met 'G is *samenhangend*' bedoelen we: tussen ieder tweetal punten bestaat een pad.
- Een *component* van G is een zo groot mogelijk samenhangend deel van een graph.
- Een *cykel* is een pad van een punt naar zichzelf.
- Met 'G is *planair*' bedoelen we: je kunt G in het platte vlak zó tekenen dat de (gebogen) lijnen elkaar niet snijden.
- Met 'G is een *boom*' bedoelen we: G is samenhangend en bevat geen cykels.

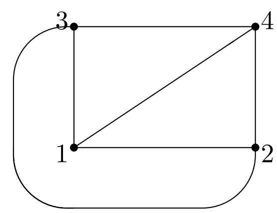
De vraag of een graph planair is is bijvoorbeeld belangrijk als de graph een electrisch circuit voorstelt dat we op een chip willen branden.

Voorbeelden

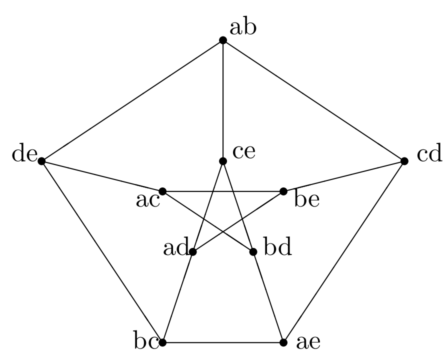
- De 'landgraph' is $\langle P, L\rangle$ waarbij P de verzameling van alle landen van de wereld is, en L de relatie 'grenst aan'. De buren van Nederland zijn dan Duitsland en België. De graad van Nederland is 2. Een pad van Nederland naar Spanje is bijvoorbeeld Nederland $\rightarrow$ Duitsland $\rightarrow$ Frankrijk $\rightarrow$ Spanje. De landgraph is niet samenhangend. $\{\text{Engeland}, \text{Schotland}\}$ is een component. Nederland $\rightarrow$ Duitsland $\rightarrow$ België $\rightarrow$ Nederland is een cykel. De landgraph is planair.

- $K_4 = \langle\{1,2,3,4\}\ ,\ \{(1,2),(1,3),(1,4),(2,3),(2,4),(3,4)\}\rangle$ (de volledige graph met vier punten). Algemeen: K_n is de graph $\langle\{1,\ldots,n\}\,,\,\{(p,q)\,|\,1 \le p < q \le n\}\rangle$. De graph K_4 is planair. Als je dat wilt inzien, moet je niet naar de linker maar naar de rechter tekening van K_4 kijken:





- De Petersen-graph is $\langle P, L\rangle$ met $\begin{cases} P = \{\text{ab},\text{ac},\text{ad},\text{ae},\text{bc},\text{bd},\text{be},\text{cd},\text{ce},\text{de}\} \\ L = \{(p,q)\,|\,p \text{ en } q \text{ hebben geen letter gemeen}\} \end{cases}$

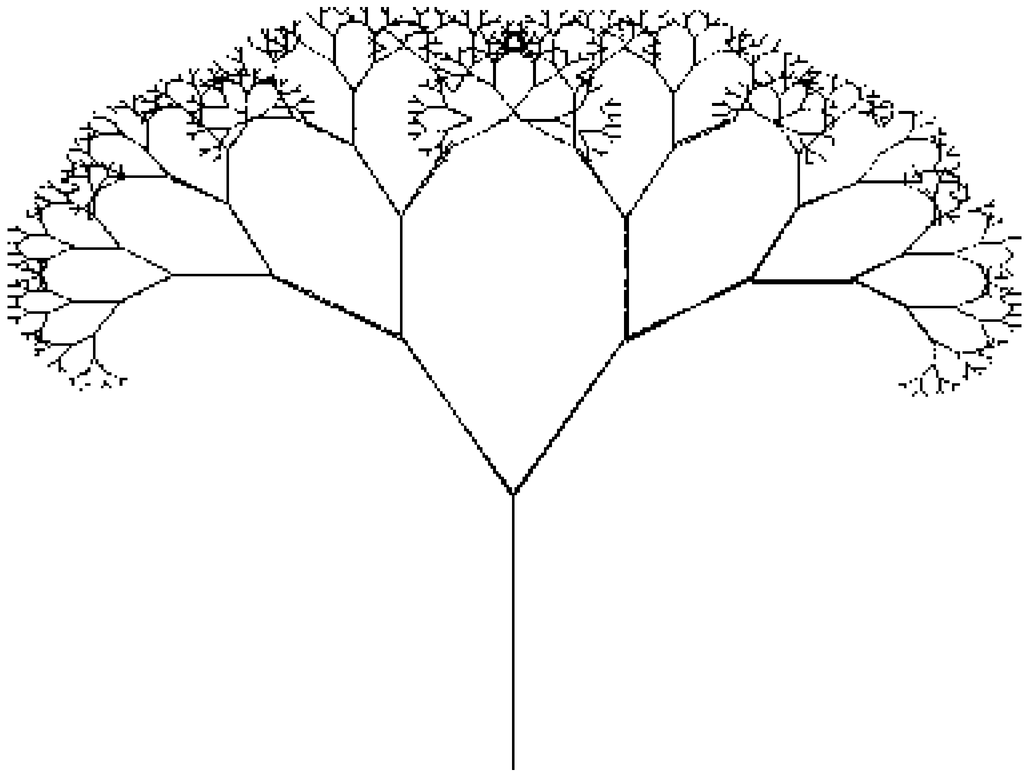


  Je kunt laten zien dat de Petersen graph niet planair is.

- Bij een taal kunnen we op een graph maken. Bekijk voorbeeld eens de taal die als volgt geproduceerd wordt:

axioma	λ		
regel	x	$\Rightarrow$	xa
	y	$\Rightarrow$	yb

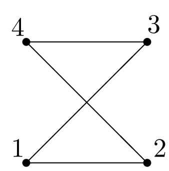
  Bij deze taal kunnen we de volgende graph maken:



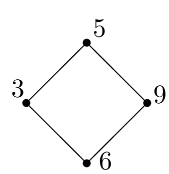
Deze graph is een boom.

Bijectie Een afbeelding f van een verzameling A naar een verzameling B heet een *bijectie* als ieder element in B precies één origineel heeft.

Isomorfie van graphen. We noemen twee graphen $\langle P, L\rangle$ en $\langle P', L'\rangle$ *isomorf* als er een bijectie $\varphi : P \to P'$ bestaat zodat voor alle x, y in P, $(x, y) \in L$ desda $(\varphi(x), \varphi(y)) \in L'$. Anders gezegd: twee graphen zijn isomorf als ze, afgezien van de namen van de punten, hetzelfde zijn. Bijvoorbeeld:



is isomorf met

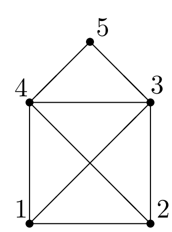


Een isomorfisme φ is:

$$\begin{aligned} 1 &\mapsto 6 \\ 2 &\mapsto 9 \\ 3 &\mapsto 3 \\ 4 &\mapsto 5 \end{aligned}$$

Isomorfe graphen zijn 'hetzelfde' als we alleen geïnteresseerd zijn in graph-theoretische eigenschappen. Bijvoorbeeld: Als G en G' isomorph zijn en G is samenhangend, dan is G' dat ook.

Euler-circuits. Een *Euler-pad* in een graph $\langle P, L\rangle$ is een pad waarin iedere lijn uit L precies één keer voorkomt. Een *Euler-circuit* of *Euler-cykel* is een cykel waarin iedere lijn uit L precies één keer voorkomt. Bijvoorbeeld:



$1 \to 4 \to 5 \to 3 \to 1 \to 2 \to 4 \to 3 \to 2$ is een Euler-pad. Omdat punt 1 graad 3 heeft, is er geen Euler-circuit. Dat kun je inzien door te kijken naar het aantal keren dat je bij een cykel door het punt 1 loopt: is dit aantal 1, dan mis je één van de drie bij 1 samenkomende lijntjes, en is aantal 2, dan doorloop je minstens een van deze lijntjes dubbel. Als je deze redenering iets algemener opschrijft, staat er een bewijs van de volgende eenvoudige stelling:

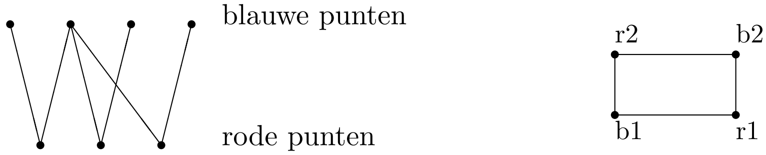
Stelling (Euler). In een samenhangende graph geldt:

1. Er bestaat een Euler-circuit dan en slechts dan als ieder punt een even graad heeft.
2. Er bestaat een Euler-pad dan en slechts dan als er hoogstens twee punten van oneven graad zijn.

Euler-circuits zijn bijvoorbeeld van belang voor de krantenbezorger of de wijkagent, die iedere straat van zijn wijk precies één keer wil doorlopen en bij zijn uitgangspunt wil terugkeren. Een heel ander probleem heeft de 'travelling salesman', die iedere klant (of iedere stad) precies één keer wil bezoeken en daarna weer naar huis wil. Hij is gebaat bij een 'Hamilton-circuit':

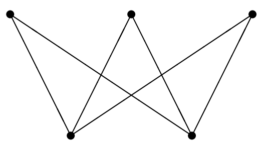
Hamilton-circuits. Een *Hamilton-pad* in een graph $\langle P, L\rangle$ is een pad waarin je ieder punt van P precies één keer ontmoet. Een *Hamilton-circuit* of *Hamilton cykel* is een cykel waarin ieder punt precies één keer optreedt (afgezien van beginpunt = eindpunt natuurlijk)

Bipartite graphen. Een graph $\langle P, L\rangle$ heet *bipartite* als P te schrijven is als $P_1 \cup P_2$ zó dat iedere lijn loopt van een punt uit P_1 naar een punt uit P_2. Anders gezegd: je kunt de punten zó blauw en rood kleuren dat iedere lijn een blauw en een rood uiteinde heeft. Twee voorbeelden:

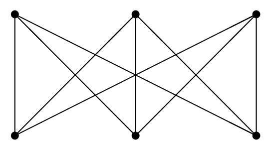


De *volledige bipartite graph* met m rode en n blauwe punten, waarbij ieder rood punt met ieder blauw punt verbonden is, wordt $K_{m,n}$ genoemd

$K_{2,3}$



$K_{3,3}$



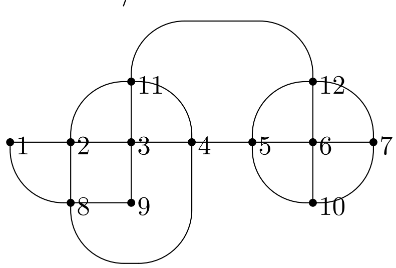
Kleuring van graphen. Een *punt-kleuring* van een graph $\langle P, L\rangle$ is een functie $f : P \to \{1, \ldots, n\}$ zodat als (p, q) een lijn is, dan $f(p) \neq f(q)$ (elk punt krijgt één der n kleuren, buren krijgen niet dezelfde kleur).
Het *kleurgetal* (of: *chromatisch getal*) van de graph is de kleinste n waarvoor dit mogelijk is. Bipartite graphen zijn dus de graphen met kleurgetal 1 of 2. Een interessante stelling van Appel en Haken, waarvan het in 1975 gevonden bewijs het niveau van dit college helaas ontstijgt:

Vierkleurenstelling. Het kleurgetal van een planaire graph is hoogstens 4.

Oefeningen

1. Bewijs dat er in een boom van een punt p naar een ander punt q precies één pad bestaat.

2. Hier is een plattegrond G van een dorpje, waarbij de straten aangegeven worden door lijntjes. Op ieder hoekpunt bevindt zich een kroeg. De kroegen zijn aangegeven door punten, genummerd van 1 t/m 12:

   Formuleer de volgende vragen in termen van Hamilton- of Euler-circuits/paden, en beantwoord ze:

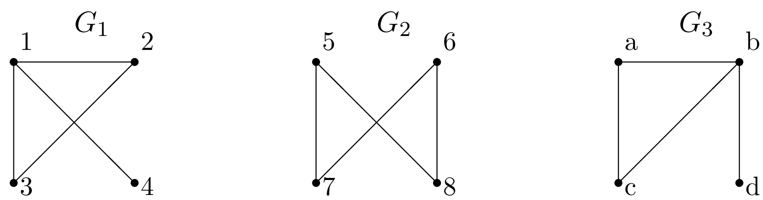
   (a) Is het mogelijk, een wandeling te maken waarbij je iedere straat precies één keer doorloopt?

(b) Is het mogelijk, een wandeling te maken waarbij je iedere straat precies één keer doorloopt en bovendien begint en eindigt bij kroeg 3?

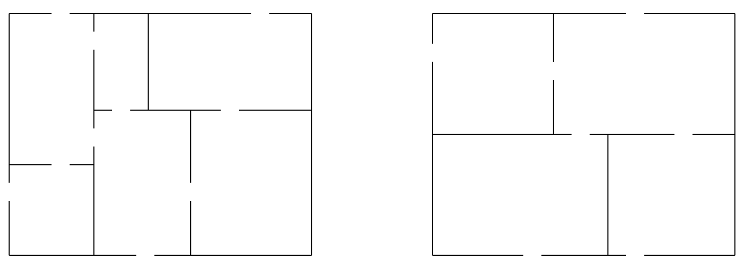
(c) Bestaat er een kroegentocht waarin iedere kroeg precies één keer voorkomt?

3. Een *brug* in een graph G is een lijn l waarvoor geldt: als je l uit G weglaat, wordt het aantal componenten verhoogd.
Bewijs dat in een boom iedere lijn een brug is.

4. Ga na welke van de onderstaande graphen isomorf zijn:



5. Hieronder vind je twee plattegronden van huizen.



(a) Teken bij elke plattegrond de bijbehorende graph, waarbij je de vertrekken (inclusief het buitengebied) door punten voorstelt, en de deuren door lijnen.

(b) Zoek voor elk huis uit of het mogelijk is een rondwandelingetje te maken waarin je iedere deur precies één keer gebruikt en bij je uitgangspunt terugkeert.

(c) Zoek voor elk huis uit of het mogelijk is een rondwandelingetje te maken waarin je ieder vertrek (en de tuin) precies één keer bezoekt en bij je uitgangspunt terugkeert.

6. Welke van de volgende graphen hebben een Hamilton-circuit?

7. Zij G de kubus-graph, met als P de verzameling van de acht hoekpunten, en als L de verzameling van de twaalf ribben

(a) Is G planair?

(b) Heeft G een Hamilton-circuit?

(c) Heeft G een Euler-circuit?

8. Laat zien dat de Peterson graph een Hamilton-pad heeft, maar geen Hamilton-cykel.

9. Laat zien dat als een bipartite graph een Hamilton-pad toelaat het aantal rode en het aantal blauwe punten hoogstens één verschilt.

Veel van de stof uit dit blok is afkomstig uit [Gie]. Een ander goed naslagwerk is [Tru91].

Oplossingen

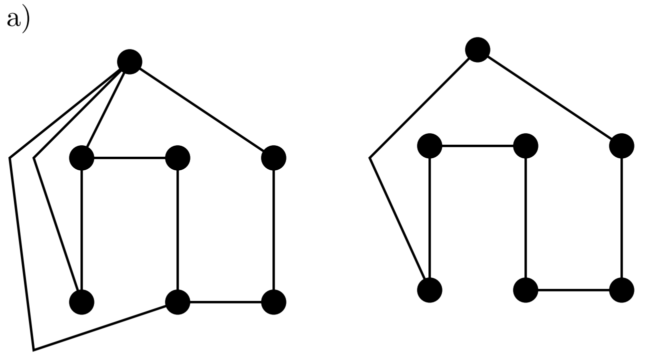
1. Een boom is samenhangend, dus tussen iedere twee punten bestaat een pad. Stel dat er tussen twee punten twee paden bestaan, dan is er een cykel, maar dat is niet mogelijk in een boom.

2. a) Bestaat er een Euler-pad? Ja, tel de graden van de punten maar, alleen 10 en 7 hebben oneven graad, dus volgens de Stelling van Euler bestaat er een Euler-pad.
   b) Bestaat er een Euler-cykel? Nee, weer volgens de Stelling van Euler.
   c) Bestaat er een Hamilton-pad? Ja, bijvoorbeeld, 12,7,10,6,5,4,11,3,9,8,1,2.

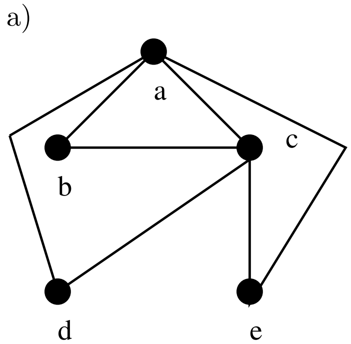
3. Volgens opgave een bestaat er tussen iedere twee punten precies een pad. Als je dus een lijn weg haalt bestaat er tussen die twee punten geen pad meer. Het bijbehorende component valt dus in tweeen.

4. In isomorfe graphen hebben evenveel punten met dezelfde graden. In G_2 hebben alle punten graad 2, in de andere twee graphen niet. Dus G_2 is niet isomorf met de andere twee graphen. G_1 en G_3 zijn isomorf: een isomorfisme is $1 \mapsto b$, $2 \mapsto a$, $3 \mapsto c$, $4 \mapsto d$, want dan kloppen de lijnen ook.

5. a)



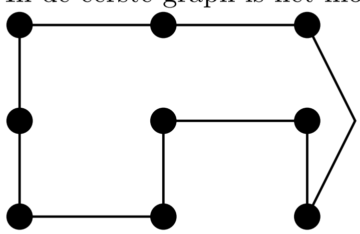
   b) Dit is een Eulercircuit. Met de Stelling van Euler zie je dat je een Eulerpas mogelijk is.
   c) Het tweede plaatje geeft een Hamilton circuit.
   a)



   b) Met de Stelling van Euler zien we dat een Euler-circuit mogelijk is.

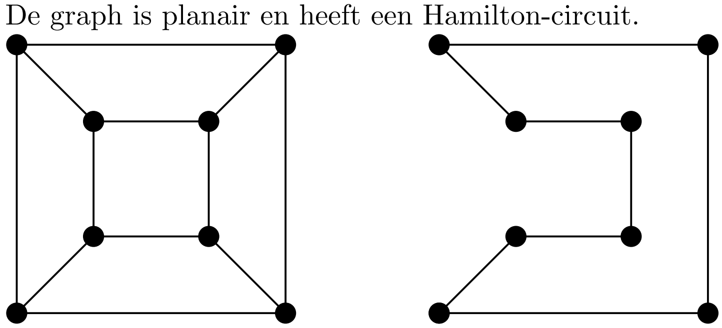
c) Een Hamilton-circuit is niet mogelijk. In een Hamilton circuit moet namelijk ieder lijn dat aan een punt met graad twee ligt voorkomen (we moeten dat punt in en uit). Als we die lijnen tekenen, zien we dat we het circuit niet af kunnen maken.

6. In de eerste graph is het mogelijk:



In de tweede niet, we moeten over iedere lijn dat aan een punt van graad 2 ligt, dus over alle lijnen in het midden. Als we bij een punt links beginnen, gaan we dus eerst naar recht, dan weer naar links, dan weer naar rechts, maar dan kunnen we helaas niet meer terug.

7. De graph is planair en heeft een Hamilton-circuit.



De graph heeft geen Euler-circuit (Stelling van Euler).

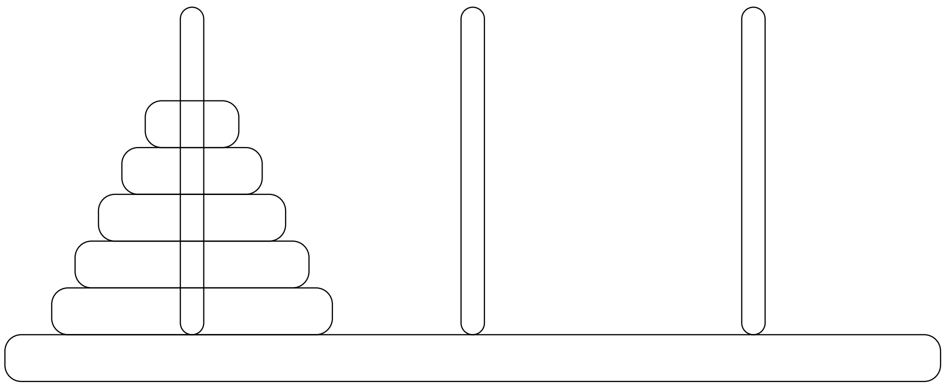
8. Dit bewijs je door netjes gevallen te onderscheiden.

9. Als een graph bipartite is en er een Hamilton-pad is, dan verschilt het aantal rode en het aantal blauwe punten hoogstens een, dit pad loopt gaat steeds van een rood punt naar een blauw punt.

Recursie

Torens van Hanoi

De puzzel de 'torens van Hanoi' bestaat uit 3 pinnen en een aantal schijven van verschillend formaat. De bedoeling is om alle schijven naar één ander pin te verschuiven. Iedere zet mogen we een schijf verzetten, maar er mag nooit een grote schijf op een kleinere staan.



In het plaatje hebben we vijf schijven. Kun je deze puzzel oplossen?

Na wat puzzelen lukt het je misschien wel, maar dan weet je vaak niet meer hoe je het gedaan hebt. Verder wil je natuurlijk ook weten hoe je het met zeven schijven doet.

Het belangrijke idee bij het oplossen deze puzzel is om het probleem te *generaliseren*. Kunnen we het probleem oplossen bij een willekeurig aantal schijven? Daarnaast moeten we opmerken dat het probleem voor vijf schijven lijkt op het probleem van vier schijven, wat weer lijkt op het probleem voor drie schijven, etc.

Als ik het probleem voor vier schijven op kan lossen kan ik het ook voor drie schijven. Want laat de grootste schijf maar even liggen. Verplaats de bovenste vier schijven naar pin 2 (We hebben aangenomen dat we dat kunnen). Pak nu de grootste schijf op en verplaats hem naar pin 3. Verplaats nu de andere schijven van pin 2 naar pin 3.

Nu zeg je misschien wat heb ik hieraan? Het probleem voor vier schijven kan ik ook niet oplossen. Nou ja, dan maken we het nog een beetje makkelijker. Als ik het probleem voor drie schijven kan oplossen, dan kan ik het ook voor vier. En als ik het probleem voor twee schijven kan oplossen, dan kan ik het ook voor drie. Ten slotte als ik het probleem voor een schijf kan oplossen, dan kan ik het ook voor twee. Dus *als* ik het probleem voor een schijf kan oplossen, dan kan ik het ook voor vijf. Maar het probleem voor een schijf is erg eenvoudig! Dus we kunnen het nu ook voor vijf schijven.

Ten slotte kunnen we nog opmerken dat er niet bijzonders is aan vijf. Het probleem kunnen we nu ook oplossen voor tien schijven, of voor ieder ander aantal.

Recursie

We hebben het vorige probleem *recursief* opgelost. Dat wil zeggen, we hebben het probleem zo ingedeeld dat het uit elkaar valt in een aantal deelproblemen die heel veel op elkaar lijken en zodat moeilijkere problemen altijd zijn terug te brengen tot eenvoudigere problemen. Ten slotte moeten we het allereenvoudigste geval natuurlijk wel op kunnen lossen.

Recursie is ook een belangrijke programmeertechniek.

Vermenigvuldigen

Stel je hebt een programmeertaal waarin je wel kunt optellen, maar nog niet kunt vermenigvuldigen. Hoe definieer je de vermenigvuldiging?

Dat kunnen we recursief doen: $n * 1 := n$ en $n * (m + 1) := n * m + m$.

Nu we kunnen vermenigvuldigen kunnen we ook machtsverheffen: $n^1 := n$ en $n^{(m+1)} := n^m * n$.

Hoeveel zetten?

Hoeveel zetten hebben we nu nodig met onze strategie voor de torens van Hanoi? Laat a_n het aantal zetten zijn dat we nodig hebben om de puzzel met n schijven op te lossen. Dus $a_1 = 1$ en $a_2 = 3$. Maar wat is a_5? Het is lastig om dit in een keer in te zien. Maar in ieder geval weten we dat we eerst het probleem met vier schijven moeten oplossen, dan de grote schijf verschuiven en weer het probleem met vier schijven oplossen. Dus $a_5 = a_4 + 1 + a_4 = 2a_4 + 1$. Net zo, $a_4 = 2a_3 + 1$ en $a_3 = 2a_2 + 1$. Maar a_2 kennen we! Dus $a_3 = 7$, $a_4 = 15$ en $a_5 = 31$. Op dezelfde manier kunnen we iedere a_n uitrekenen.

Als we nog eens naar de rij van oplossingen 1,3,7,15,31,... kijken valt je misschien op dat de rij lijkt op de rij 2,4,8,16,32,... De rij van de machten van twee. De eerste rij is steeds eentje minder. Is dit toeval of klopt het altijd? Stel even dat $a_{37} = 2^{37} - 1$, dan is

$$a_{38} = 2a_{37} + 1 = 2 \cdot (2^{37} - 1) + 1 = 2^{38} - 2 + 1 = 2^{38} - 1.$$

Dus als het voor het 37ste element klopt, dan ook voor het 38ste. Nu is 37 natuurlijk helemaal niet bijzonder. Dus op de zelfde manier zien we dat als het voor n klopt dan ook voor $n + 1$. Verder wisten we al dat $a_1 = 2^1 - 1$. Dus het klopt voor 2, dus het klopt voor 3, dus het klopt voor 4, dus ..., dus het klopt voor 37, dus het klopt voor 38, dus ... We zien dus dat voor alle getallen n geldt: $a_n = 2^n - 1$.

Inductie

De bewijsmethode die we net hebben gebruikt heet *inductie*. Inductie lijkt heel veel op de recursieve methode die we gebruiken om functies te definieren.

Inductie kunnen we gebruiken als we voor alle natuurlijke getallen (de getallen 1,2,3,4,5,6,7,...) een bepaalde uitspraak $P(n)$ willen bewijzen. Een bewijs met inductie gaat op de volgende manier:

1. Bewijs eerst $P(1)$.

2. Bewijs dan: Als $P(n)$, dan geldt ook $P(n + 1)$.

Dat is genoeg.

Als we nu willen laten zien dat bijvoorbeeld $P(37)$ geldt, dan weten we dat $P(1)$ waar is, dus (met 2.) ook $P(2)$, dus (weer met 2.) ook $P(3)$, dus ook $P(4)$,..., dus ook $P(36)$, dus ook $P(37)$!

In het vorige voorbeeld was $P(n)$ de uitspraak $a_n = 2^n - 1$.

Inductie kun je zien als het omgekeerde van recursie. Met inductie ligt de nadruk op steeds moeilijkere gevallen, bij recursie is dat omgekeerd.

Optellen

Hoeveel is $1+2+3+\ldots+99$? Dat kun je natuurlijk op papier doen. Je kunt je rekenmachine gebruiken. In beide gevallen ben je lang bezig. Tenzij je het slim doet. Je kunt bijvoorbeeld definiëren $s(n) := 1+2+\cdots+n$. Dan kun je een recursief programmatje schrijven wat $s(99)$ voor je uitrekent, want $s(n+1) = s(n) + n$.

Als je wat waarden uitrekent krijg je het vermoeden dat $s(n) = (n^2 + n)/2$.

Klopt dit nu ook? We gaan het bewijzen met inductie. Neem voor $P(n)$ de uitspraak $s(n) = (n^2 + n)/2$.

1. $P(1) : s(1) = 1 = (1^2 + 1)/2$.

2. Stel dat $P(n)$, d.w.z. $s(n) = (n^2 + n)/2$. Dan

$$s(n+1) = s(n) + n + 1 = \frac{n^2+n}{2} + n + 1 = \frac{n^2+n+2n+2}{2} = \frac{(n+1)^2+(n+1)}{2}.$$

   Dus $P(n+1)$ geldt.

Dus geldt voor alle getallen n, $s(n) = (n^2 + n)/2$.

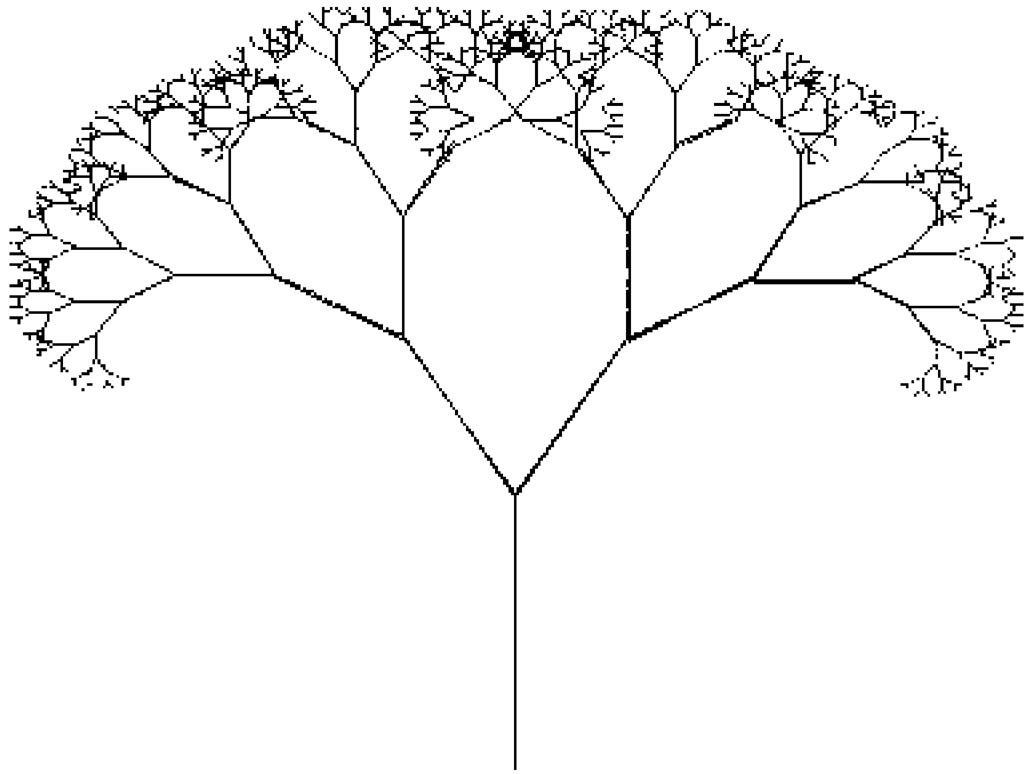
Rangschikkingen

Op hoeveel manieren kunnen we de cijfers 1,2,3,4,5,6,7,8,9 rangschikken? Het is lastig alle mogelijkheden op te schrijven. Maar gelukkig hoeft dat niet. Laat a_n het aantal rangschikkingen zijn van een verzameling van n elementen. We willen dus a_9 weten. Gelukkig weten we dat $a_1 = 1$.Verder is $a_{n+1} = (n+1)a_n$. Want we kiezen eerst een van de $(n+1)$ elementen, dat zetten we vooraan in de rij. Daarna hebben we nog n mogelijkheden.

Hoe rekenen we a_9 nu uit? De definitie die we boven hebben gegeven is precies de definitie van de faculteit-functie, die wordt genoteerd met $n!$. Dus $a_9 = 9!$ en algemeen $a_n = n!$ Deze functie zit standaard op de meeste rekenmachines.

Binaire bomen

Beschouw het volgende plaatje:



Hoewel deze boom er misschien complex uit ziet, is deze getekend met een simpele recursieve procedure. Het basisinzicht hierbij is dat een boom bestaat uit een stam met daarboven twee andere bomen, links en rechts, die net iets kleiner zijn (en iets gedraaid zijn).

Een recursief recept (functie, procedure) is eenvoudig te geven:

1. teken een stam
2. teken de linker (sub)boom
3. teken de rechter (sub)boom

Dit recept kunnen we preciezer maken met de volgende functie/procedure:

$$f(n,x,y,\alpha,l) = \begin{cases} \mathit{doe\,niets} & \mathit{als}\,n=0 \\ 1)\,\mathit{teken\,stam}\,(\mathit{positie}\,x,y;\ \mathit{hoek}\,\alpha;\ \mathit{lengte}\,l) & \\ 2)\,f(n-1,x_1,y_1,\alpha-30,l/2) & \mathit{als}\,n>0 \\ 3)\,f(n-1,x_2,y_2,\alpha+30,l/2) & \end{cases}$$

Hierbij is n de hoogte van de boom; x en y de positie van de boom (startpunt van de stam); α is de hoek van de boom en l is de lengte van de stam.

De hoogte van de boom, n, is belangrijk in deze recursieve procedure; de andere variabelen laten we even buiten beschouwing.

Het tekenen van een boom met hoogte $n$kan dus terug gebracht worden tot een simpeler probleem: het tekenen van twee bomen met hoogte $n-1$. Dit recursieve patroon is duidelijk te herkennen wanneer je de stappen volgt van een computer volgt die de procedure uitvoert:

```
[1] teken boom: f(3)
[1.1] teken stam
[1.2] teken linker boom: f(2)
[1.2.1] teken stam
[1.2.2] teken linker boom: f(1)
[1.2.2.1] teken stam
[1.2.2.2] teken linker boom: f(0)
[1.2.2.2.1] doe niets
[1.2.2.3] teken rechter boom: f(0)
[1.2.2.3.1] doe niets
[1.2.3] teken rechter boom: f(1)
[1.2.3.1] teken stam
[1.2.3.2] teken linker boom: f(0)
[1.2.3.2.1] doe niets
[1.2.3.3] teken rechter boom: f(0)
[1.2.3.3.1] doe niets
[1.3] teken rechter boom: f(2)
[1.3.1] teken stam
[1.3.2] teken linker boom: f(1)
[1.3.2.1] teken stam
[1.3.2.2] teken linker boom: f(0)
[1.3.2.2.1] doe niets
[1.3.2.3] teken rechter boom: f(0)
[1.3.2.3.1] doe niets
[1.3.3] teken rechter boom: f(1)
[1.3.3.1] teken stam
[1.3.3.2] teken linker boom: f(0)
[1.3.3.2.1] doe niets
[1.3.3.3] teken rechter boom: f(0)
[1.3.3.3.1] doe niets
```

Merk op dat een recursieve *definitie* vaak erg eenvoudig is, maar de *uitvoering* van een recursieve functie veel werk met zich mee brengt: er zijn veel stappen, en je moet een goede administratie bijhouden om steeds te weten met welk subprobleem je bezig bent. Typisch werk voor een computer dus.

Driehoek van Pascal

Binomiaalcoëfficiënten. We definiëren, voor natuurlijke getallen n en k met $k \leq n$, $\binom{n}{k}$ als het aantal manieren om k objecten uit een verzameling van n elementen te pakken.

We bekijken nu de roosterpunten (n, k) waarvoor $k \leq n$. Deze verzameling van roosterpunten tekenen we zó, dat het punt $(0, 0)$ de top wordt:

$$
\begin{array}{c}
(0,0) \\
(1,0) \quad (1,1) \\
(2,0) \quad (2,1) \quad (2,2) \\
(3,0) \quad (3,1) \quad (3,2) \quad (3,3) \\
(4,0) \quad (4,1) \quad (4,2) \quad (4,3) \quad (4,4) \\
(5,0) \quad (5,1) \quad (5,2) \quad (5,3) \quad (5,4) \quad (5,5) \\
(6,0) \quad (6,1) \quad (6,2) \quad (6,3) \quad (6,4) \quad (6,5) \quad (6,6) \\
\ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots
\end{array}
$$

We gaan nu ieder van deze roosterpunten voorzien van een getal:

Eerste Driehoek van Pascal. Zet aan de rand eentjes, en vul de rest in door 'schuin optellen'. Anders gezegd: voorzie de roosterpunten $(n, 0)$ en (n, n) van het getal 1, en schrijf bij elk ander roosterpunt (n, k) de som van de getallen, die je bij de roosterpunten $(n-1, k)$ en $(n-1, k-1)$ moet schrijven. De Eerste Driehoek van Pascal (1e $\triangle$vP) is dus het volgende schema van getallen (het gaat naar onder oneindig ver door, ik heb alleen maar een klein beginstuk opgeschreven):

$$
\begin{array}{ccccccccccccccc}
 & & & & & & & 1 & & & & & & & \\
 & & & & & & 1 & & 1 & & & & & & \\
 & & & & & 1 & & 2 & & 1 & & & & & \\
 & & & & 1 & & 3 & & 3 & & 1 & & & & \\
 & & & 1 & & 4 & & 6 & & 4 & & 1 & & & \\
 & & 1 & & 5 & & 10 & & 10 & & 5 & & 1 & & \\
 & 1 & & 6 & & 15 & & 20 & & 15 & & 6 & & 1 & \\
1 & & 7 & & 21 & & 35 & & 35 & & 21 & & 7 & & 1
\end{array}
$$

Tweede Driehoek van Pascal. Zet op plaats (n, k) het getal $\begin{pmatrix} n \\ k \end{pmatrix}$

Opgave: bewijs dat er in 2e $\triangle$vP aan de rand allemaal eentjes staan.

We beweren dat als $k < n$, dan $\binom{n+1}{k+1} = \binom{n}{k+1} + \binom{n}{k}$. Kiezen we namelijk een deelverzameling A van k elementen uit $\{1, 2, 3, 4, \ldots, n+1\}$ dan zijn er twee mogelijkheden $n+1 \in A$ of $n \not\in A$. In het eerste geval is $A - \{n+1\}$ dus een deelverzameling van k elementen uit $\{1, 2, 3, 4, \ldots, n\}$, in het tweede geval is A een deelverzameling van $k+1$ elementen uit $\{1, 2, 3, 4, \ldots, n\}$. Dus ook in de 2e $\triangle$vP het principe van 'schuin optellen' geldt: ieder inwendig getal is de som van de schuin erboven staande getallen.

Derde Driehoek van Pascal. Zet op plaats (n, k) het aantal wegen via roosterpunten van $(0, 0)$ naar (n, k), waarbij we iedere keer een stap naar links-onder of rechts-onder zetten.

Je kunt nu inzien:

$$
\begin{cases}
\text{in de 3e } \triangle \text{ vP staan aan de rand allemaal eentjes} \\
\text{in de 3e } \triangle \text{ vP geldt ook het principe van 'schuin optellen'}
\end{cases}
$$

Gevolg: de 3e $\triangle$vP is precies gelijk aan de 1e $\triangle$vP

Vierde Driehoek van Pascal. Zet op plaats (n, k) de coëfficiënt van x^k in de veelterm $(1+x)^n$.

Ga zelf na, dat ook deze 4e $\triangle$vP weer precies gelijk is aan de 1e $\triangle$vP. De vier behandelde versies van de $\triangle$vP leiden dus allemaal tot hetzelfde schema van getallen

Binomium van Newton.

$$(1+x)^n = \binom{n}{0} + \binom{n}{1} x + \binom{n}{2} x^2 + \cdots + \binom{n}{n} x^n$$

Dit Binomium van Newton is niets anders dan de stelling: 2e $\triangle$vP = 4e $\triangle$vP.

Het meer algemene geval

$$(y+z)^n = \binom{n}{0}y^n + \binom{n}{1}y^{n-1}z + \binom{n}{2}y^{n-2}z^2 + \cdots + \binom{n}{n}z^n$$

volgt hier uit: $(y+z)^n = y^n(1+(\frac{z}{y})^n)$ als $y \neq 0$. Het geval $y = 0$ is eenvoudig.

Som- en productregel.

Somregel Als een eerste taak op m manieren kan worden uitgevoerd, en een tweede taak op n manieren, en deze taken kunnen niet tegelijk worden uitgevoerd, dan zijn er $m + n$ manieren om precies één van deze taken uit te voeren.

Productregel Als een eerste taak op m manieren kan worden uitgevoerd, en een tweede taak op n manieren, dan zijn er $m \cdot n$ manieren om eerst de eerste taak en vervolgens de tweede taak uit te voeren.

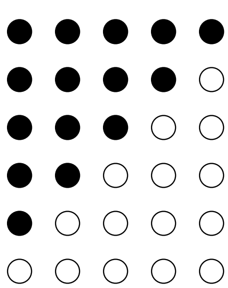
Opgaven

1. De uitvinder van het schaakbord mocht van de koning van Pezië een beloning uitzoeken. Hij mocht vragen wat hij maar wilde. Hij koos voor de volgende beloning. Hij wilde op het eerste vakje van het schaakbord 1 rijstkorrel hebben, op de tweede 2, op de derde 4, etc. Dus op ieder vakje twee keer zoveel rijstkorrels. De koning vond hem erg bescheiden.
   Laten we eens kijken hoeveel hij vroeg: hij wilde dus
   $$1 + 2 + 2^2 + 2^3 + 2^4 + \cdots + 2^{63}$$
   korrels hebben. Kun je een eenvoudige formule vinden voor de uitkomst van deze som? [Hint: schrijf eerst eens wat termen van de rij 1, $1+2$, $1+2+2^2$, ... op. Herken je deze rij? Wat is je vermoeden van de uitkomst? Kun je dit met inductie bewijzen?]

2. [Deze opgave past niet echt in dit hoofdstuk.] Neem aan dat een rijstkorrel breedte en dikte van 1mm heeft en een lengte van 5mm. Spreid alle rijstkorrels van vraag 1 gelijkmatig uit over Nederland (400km×200km). Hoe hoog wordt deze berg?

3. Bewijs met inductie dat voor alle n, $2^n \geq n$.

4. We hebben al een mooie formule voor de som van de rij
   $$1 + 2 + 3 + 4 + 5 + \cdots + n = \frac{(n+1)n}{2}.$$
   Wat is het verband met het volgende plaatje?

5. Ik gooi een rode en een blauwe dobbelsteen. In hoeveel van alle mogelijke uitkomsten gooi ik met de rode een even aantal ogen en met de blauwe een oneven aantal ogen?

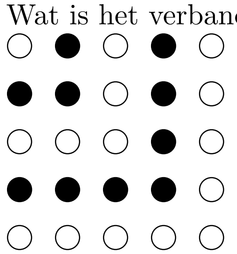
6. Bekijk de volgende taal

axioma	λ
Regels	$x \Rightarrow ax$
	$x \Rightarrow bx$

   Teken de boom die bij deze taal hoort.
   Op hoeveel manieren kan ik een woord met 3 a's en 5 b's maken?

7. **Rijtjes van lengte n.** Hoeveel rijtjes kunnen we maken van lengte n, met daarin alleen de getallen 1 tot en met 5? Dit kunnen we ook weer recursief oplossen. Noem het aantal rijtjes van lengte n a_n. Een rijtje van lengte 1 kunnen we op 5 manieren maken, dus $a_1 = 5$. Een rijtje van lengte $n+1$ kunnen we maken door eerst een rijtje van lengte n te maken en daar een element achter te zetten. Dus $a_{n+1} := 5a_n$. Herinner je nu even de definitie van machtsverheffen. Bewijs met inductie dat voor alle n, $a_n = 5^n$.

8. Bewijs met inductie dat $1 + 3 + 5 + 7 + \cdots + (2n-1) = n^2$.
   Wat is het verband met het volgende plaatje?



Uitwerkingen van deze opgaven

1. Definieer de rij s_n als
$$\begin{cases} 1 & n = 1 \\ s_n + 2^n & n > 1 \end{cases}$$
   nu is s_n dus $1 + 2 + 2^2 + \cdots + 2^{n-1}$. We willen s_{64} weten.
   Na wat proberen vermoed ik dat $s_n = 2^n - 1$. Dat ga ik bewijzen met inductie.
   Bewijs: Neem $P(k)$ als $s_k = 2^k - 1$.
   1) $P(1)$ geldt want $s_1 = 1 = 2 - 1$.
   2) Stel dat $P(k)$ geldt dan wil ik laten zien dat $P(k+1)$ ook klopt:
$$s_{k+1} = s_k + 2^k \overset{P(k)}{=} (2^k - 1) + 2^k = 2 \cdot 2^k - 1 = 2^{k+1} - 1.$$
   Dus als $P(k)$, dan ook $P(k+1)$.
   Met inductie zien we dus dat voor alle n in $\{1, 2, 3, 4, \ldots\}$ $P(n)$, dwz. $s_n = 2^n - 1$.
   Specialiseren: dus $s_{64} = 2^{64} - 1$.

2. Ik kom op 64cm. Ik heb aangenomen dat $2^{10} = 10^3$, een goede gewoonte uit de informatica (1 kilobyte is 2^{10} bytes).

3. We bewijzen dat voor alle $n \geq 1$, $2^n \geq n$.
   Bewijs met inductie: Neem $P(k)$ als $2^k \geq k$.
   1) $P(1)$ geldt want $2^1 \geq 1$.
   2) Stel dat $P(k)$ geldt dan wil ik laten zien dat $P(k+1)$ ook klopt:
   $$2^{k+1} = 2 \cdot 2^k \overset{P(k)}{\geq} 2k = k + k \geq k + 1.$$
   Dus als $P(k)$, dan ook $P(k+1)$.
   Met inductie zien we dus dat voor alle n in $\{1, 2, 3, 4, \ldots\}$ $P(n)$, dwz. $2^n \geq n$.

4. Is in het college behandeld.

5. Pas de product regel toe $3 \cdot 3 = 9$.

6. Deze boom lijkt sprekend op de boom in het hoofdstuk over graphen. De taal is $\{a, b\}^*$. Zo'n woord heeft dus 8 letters. We moeten 3 plaatsen kiezen uit 8 mogelijkheden, dus $\binom{8}{3}$.

7. Net als opgave 1 en 3.

8. Net als opgave 1 en 3.

Kanstheorie

De theorie die we over het tellen hebben opgebouwd is erg handig als we kansen willen berekenen. We maken nu precies wat we met een kans bedoelen.

Het vaasmodel. Laat Ω een verzameling zijn (de *uitkomstruimte*) en X een deelverzameling van Ω, dan is de kans $\mathbb{P}(X)$ op de gebeurtenis X :

$$\boxed{\mathbb{P}(X) := \tfrac{\#X}{\#\Omega}.}$$

Let op: bij dit model nemen we aan dat alle elementen van Ω met gelijke kans voorkomen. Dit is bijvoorbeeld het geval als ik n rode knikkers en k blauwe knikkers in een vaas doe en ze goed meng. Laat K de verzameling van knikkers zijn, R de verzameling van rode knikkers en B de verzameling van blauwe knikkers. De is $\mathbb{P}(R) = \frac{\#R}{\#K} = \frac{n}{n+k}$ en $\mathbb{P}(B) = \frac{k}{n+k}$.

Een ander voorbeeld: de kans dat ik een 5 of een 6 gooi met een (eerlijke) dobbelsteen is:

$$\mathbb{P}(\{5, 6\}) = \frac{\#\{5, 6\}}{\#\{1, 2, 3, 4, 5, 6\}} = \frac{2}{6} = \frac{1}{3}.$$

Maar als we nu kijken naar het aantal keer kop dat we krijgen door twee keer met een munt te gooien dan zijn er drie mogelijke uitkomsten 0, 1 of 2. Toch is de kans op twee keer kop niet 1/3! Het vaas model is hier dus niet van toepassing. Door het een beetje anders te bekijken kunnen we er wel een vaasmodel bij maken. Neem voor de uitkomstruimte $\Omega =$\{kk,km,mk,mm\}. Dan

$$\mathbb{P}(\{\mathrm{kk}\}) = \frac{\#\{\mathrm{kk}\}}{\#\Omega} = 1/4.$$

Oefeningen

3.1. Opgave. Bereken de kans op een 5 en een 6 met twee dobbelstenen.

3.2. Opgave. In een vaas zitten 10 rode en 20 blauwe ballen. Ik doe een willekeurige greep van tien ballen uit deze vaas. Bereken de kans dat deze greep bestaat uit 3 rode en 7 blauwe ballen.

3.3. Opgave. Hoeveel rijtjes van 25 nullen en 6 enen bestaan er?

Soms willen we modeleren dat we al meer weten over de wereld. Dat doen we bijvoorbeeld door naar voorwaardelijke kansen te kijken.

Laat Ω weer de uitkomstenruimte zijn en laat $X, Y \subset \Omega$. Dan is *de kans op X gegeven Y*:

$$\boxed{\mathbb{P}(X|Y) := \tfrac{\mathbb{P}(X\cap Y)}{\mathbb{P}(Y)}.}$$

Bijvoorbeeld de kans dat de som van het aantal ogen op twee dobbelstenen 8 is, gegeven dat allebei een even aantal ogen geven gelijk aan:

$$\mathbb{P}(\text{som is 8}|\text{ allebei even}) = \frac{\mathbb{P}(\text{som is 8 en allebei even})}{\mathbb{P}(\text{allebei even})} = \frac{3/36}{9/36} = 1/3.$$

Soms bestaat er geen verband tussen twee gebeurtenissen X en Y. Dan zijn ze *onafhankelijk*. In dat geval is $\mathbb{P}(X) = \mathbb{P}(X|Y)$. In feite is deze laatste uitspraak de definitie van 'onafhankelijk'.

Bijvoorbeeld de kans dat als ik met twee dobbelstenen gooi de tweede een 6 geeft, gegeven dat de eerste een 6 geeft, is gelijk aan de kans dat de tweede een 6 geeft. Reken maar na!

3.4. Opgave. Bij een quiz heb je gewonnen en nu mag je een van drie deuren kiezen. Helaas zit er maar achter een van de deuren een prijs (de auto!), achter de andere deuren zit niets. Na veel twijfelen kies je een deur, daarna opent de quizmaster een andere deur waarvan hij weet dat er niets achter zit. Hij geeft je de kans om te wisselen van deur. Vergroot je je kans op de auto als je dat doet?

Oplossingen

1. Dit is gelijk aan de kans dat ik eerst een vijf gooi en dan een zes of eerst een zes en dan een vijf. Met de product- en de som regel zien we dat dit gelijk is aan $1/6 \cdot 1/6 + 1/6 \cdot 1/6 = 1/18$.

2. Het aantal grepen van tien ballen is $\binom{30}{10}$. Het aantal grepen met drie rode en zeven blauwe ballen is $\binom{10}{3}\binom{20}{7}$ volgens de productregel. De kans is dus $\frac{\binom{10}{3}\binom{20}{7}}{\binom{30}{10}}$.

3. Ik moet zes plaatsen kiezen uit 31, dus $\binom{31}{6}$.

4. De kans dat ik het gelijk goed had is $1/3$. Dat blijft dus mijn kans als ik niet wissel. De kans dat ik het goed heb nadat ik heb gewisselt is:

$$\begin{aligned}
\mathbb{P}(\text{goed na wissel}) &= \mathbb{P}(\text{1e keer goed}) \cdot \mathbb{P}(\text{goed na wissel}|\text{1e keer goed}) \\
&+ \mathbb{P}(\text{1e keer fout}) \cdot \mathbb{P}(\text{goed na wissel}|\text{1e keer fout}) \\
&= \frac{1}{3} \cdot 0 + \frac{2}{3} \cdot 1 = \frac{2}{3}
\end{aligned}$$

   Wisselen is dus verstandig!

4. Logica 2; Predicatenlogica

Verzamelingen

Voor de volledigheid behandelen we nu verzamelingen. Verzamelingen kom je op veel plekken in de wiskunde en de informatica tegen.

Verzamelingen. Voorbeelden van verzamelingen zijn:

$\{5,6,2\}$ de verzameling, bestaande uit de getallen 5, 6 en 2.

$\{0,1,2,3,\ldots\}$ de verzameling van alle natuurlijke getallen, $\mathbb{N}$

$\emptyset$ de lege verzameling

$\{0,1,2,\ldots,50\}$ de verzameling van alle natuurlijke getallen kleiner dan 51;
deze verzameling wordt ook wel genoteerd als $\{n \in \mathbb{N} | n \leq 50\}$

(lees: de verzameling van al die n uit $\mathbb{N}$, waarvoor $n \leq 50$)

$\boldsymbol{\{\ 0,\ 1,\ 4,\ 9,\ 16,\ \ldots\}}$ de verzameling van alle kwadraten; andere notatie: $\{n^2 \mid n \in \mathbb{N}\}$
(lees: de verzameling van alle n^2 waarbij $n \in \mathbb{N}$)

$\mathbb{Z}$ de verzameling van alle gehele getallen, $\{\ldots,-3,-2,-1,0,1,2,3,4,5,\ldots\}$

$\mathbb{Q}$ de verzameling van alle rationale getallen, $\{\frac{n}{m} \mid n \in \mathbb{Z},\ m \in \mathbb{Z},\ m \neq 0\}$;
voorbeelden: $\frac{7}{3}$, $-\frac{3}{11}$ en 375 zijn elementen van $\mathbb{Q}$.

Nog wat afspraken en notaties:

- $x \in V$ betekent: x is een element van V
  Bijvoorbeeld: $5 \in \{5,6,2\}$ en $5 \in \mathbb{N}$
- $x \not\in V$ betekent: $\neg(x \in V)$
  Bijvoorbeeld: $5 \not\in \emptyset$
- We noemen twee verzamelingen *gelijk* als ze precies dezelfde elementen hebben
  Bijvoorbeeld: $\{5,6,2\} = \{2,5,6,6\}$

  Speciaal geval: twee talen zijn gelijk als ze dezelfde woorden bevatten.
- $\#V$ is een afkorting voor: het aantal elementen van V (je vindt hiervoor ook wel de notatie $|V|$)
  Bijvoorbeeld: $\#\{2,5,6,6\} = 3$
- $V \subseteq W$ betekent: ieder element van V is een element van W (V is een deelverzameling van W)
  Bijvoorbeeld: $\{2,5\} \subseteq \{5,2,6\}$, $\emptyset \subseteq \mathbb{N}$, $\mathbb{Q} \subseteq \mathbb{R}$
  Merk op dat $V = W \quad \leftrightarrow \quad V \subseteq W \wedge W \subseteq V$

- $V \cap W$:= de verzameling van alle x waarvoor $x \in V \wedge x \in W$ (de doorsnijding van V en W)
  Bijvoorbeeld: $\{2,3,4\} \cap \{3,4,5\} = \{3,4\}$
- $V \cup W$:= de verzameling van alle x waarvoor $x \in V \vee x \in W$ (de vereniging van V en W)
  Bijvoorbeeld: $\{2,3,4\} \cup \{3,4,5\} = \{2,3,4,5\}$
- $V \setminus W$:= de verzameling van alle x waarvoor $x \in V \wedge x \notin W$ (V met daaruit weggelaten W)
  Bijvoorbeeld: $\{2,3,4,5\} \setminus \{3,4\} = \{2,5\}$
- $V \times W$:= de verzameling van alle paren (v,w) met $v \in V$ en $w \in W$
  (het carthesisch product van V met W)
  Bijvoorbeeld: $\{2,3\} \times \{7,2\} = \{(2,2),(2,7),(3,2),(3,7)\}$
  Opmerking: we noemen twee paren (a,b) en (c,d) *gelijk* als $(a=c) \wedge (b=d)$;
  dus $(2,3)$ is een ander paar dan $(3,2)$
- V^n := de verzameling van alle rijtjes $(v_1,\ldots,v_n)$ waarbij $v_1 \in V,\ldots,v_n \in V$
  Bijvoorbeeld: $\{1,7\}^3 = \{(1,1,1),(1,1,7),(1,7,1),(1,7,7),(7,1,1),(7,1,7),(7,7,1),(7,7,7)\}$

Probeer zelf de juistheid van de volgende eenvoudige stellingen in te zien. Ze gaan over het aantal elementen van verzamelingen. Gegeven is, dat V en W eindige verzamelingen zijn (zodat $\#V$ en $\#W$ correct gedefinieerde natuurlijke getallen zijn)

Als $V \subseteq W$ dan $(\#V) \leq (\#W)$
$\#(V \times W) \;=\; (\#V) \cdot (\#W)$
$\#(V \cup W) \;=\; (\#V) + (\#W) - (\#(V \cap W))$
$\#(V^5) \;=\; (\#V)^5$
$\#(V^n) \;=\; (\#V)^n$

Oefeningen

4.1. Opgave. Zij $V = \{1,2,3\}$ en $W = \{3,4\}$. Wat is

1. $V \cup W$
2. $V \cap W$
3. $V \setminus W$
4. $V \times W$
5. V^2

4.2. Opgave. Schrijf de volgende verzamelingen in de vorm $\{a_1,\ldots,a_n\}$:

1. $\{n \in \mathbf{N} \mid n^3 \leq 3n^2 + 6\}$
2. $\{n^2 \mid n \in \{1,2,3\}\}$

4.3. Opgave.
Welke van de volgende beweringen zijn waar voor alle deelverzamelingen V en W van $\mathbf{N}$?

a) $\mathbf{N} \setminus (V \cap W) = (\mathbf{N} \setminus V) \cup (\mathbf{N} \setminus W)$

b) $V \subseteq W \quad \to \quad \mathbf{N} \setminus W \subseteq \mathbf{N} \setminus V$

5. Beweging; Continue wiskunde

De exponentiële functie

We gaan verder in op de continue wiskunde die in de vorige les behandeld is. We zullen vooral de e-macht of exponentiële functie bekijken.

Opgave: Een bank beloofd je bij inleg van 1000 euro, ieder jaar 100 euro uit te betalen. Een ander bank bied je 4% rente per jaar. Welke bank kies je?

Een paar eigenschappen van de e-macht:

$$e^{x+y} = e^x e^y.$$

Om dit te begrijpen kun je de volgende interpretatie nemen: e^x is de factor waarmee ons kapitaal is toegenomen nadat er x dagen verstreken zijn. Stel we beginnen met 1 euro op dag 0. dan hebben we naar $x + y$ dagen dus e^{x+y} euro, maar je kunt het ook als volgt zien: na x dagen hebben we e^x euro na nog eens y dagen is dit met een factor e^y vergroot, dus na $x + y$ dagen hebben we $e^x e^y$ euro. We zien dat de formule hierboven klopt.

Opgave: Leg de volgende feiten uit:

- $e^0 = 1$,
- $e^x > e^y$ als $x > y$.
- $e^{-x} = 1/e^x$.
- $e^x > 0$ voor alle reële getallen x.

Logarithme

De bank wil je na een jaar 4% rente geven. De vorige les hebben we geleerd dat als ze na een half jaar 2% rente geven ze duurder uit zijn. Dus hoeveel rente moeten ze je na een half jaar geven, zodat als je continu geld op neemt je 4% rente krijgt?

Om dit probleem op te lossen is de logarithmische functie handig. Dit is de functie ln zodat voor alle getallen x en alle positieve getallen y :

$$e^x = y \Leftrightarrow x = \ln y.$$

Opgave: leidt nu zelf de volgende eigenschappen af:

- $\ln xy = \ln x + \ln y$.
- $\ln x^{-1} = -\ln x$.
- Voor alle natuurlijke getallen n, $\ln x^n = n \ln x$. (Hint: met inductie).
- Waarom is de functie ln niet gedefiniëerd voor negatieve getallen.

Nu lossen we het probleem op: we weten dat $e^r = 1,04$, dus $r = \ln(1,04)$. Na een half jaar krijgen we dus $e^{rt} = e^{\ln(1,04)\cdot(1/2)}$.

Toepassingen

De exponentiële functie groeit erg snel. Als het aantal berekeningsstappen dat in een algorithme gemaakt wordt exponentieel van de invoer afhangt, dan is het meestal geen practisch algrotihme bij een grote invoer. Dit is bijvoorbeeld zo bij de Torens van Hanoi, daar hebben we 2^n stappen nodig om het probleem voor n schijven op te lossen.

Opgave: Schat hoe lang we bezig zijn om het probleem met 100 schijven op te lossen als we een schijf per seconde verplaatsen.

Opgave: Schat eens hoeveel cijfers e^{100} voor de komma heeft. (Hint: gebruik dat $2 < e < 3$).

Opgave: Bacterieën groeien erg snel, we weten dat het exponentieel snel gaat, dus het aantal gram bacterieën $g(t)$ na t seconden is ae^{bt}, waar $a \geq 0$ en $b \geq 0$. Na 1 minuut hebben we 5gram, na twee minuten hebben we $g = 6$gram. Hoeveel gram is er na een uur? Hoeveel is dat ongeveer?

References

[Gie] Wim Gielen. Discrete wiskunde voor informatici. Katholieke Universiteit Nijmegen - Subfaculteit Wiskunde 2002.

[Hof79] Douglas R. Hofstadter. *Gödel, Escher, Bach: an eternal golden braid.* Basic Books Inc. Publishers, New York, 1979.

[Tru91] J.K. Truss. *Discrete Mathematics for computer scientists.* Addison-Wesley, 1991. ISBN 0-201-17564-9.

[vBe] van Benthem *et. al.* *Logica voor informatici.* Addison-Wesley, Nederland. ISBN 90-6789-484-2 (dit boek wordt ook gebruikt in het college Beweren en Bewijzen).