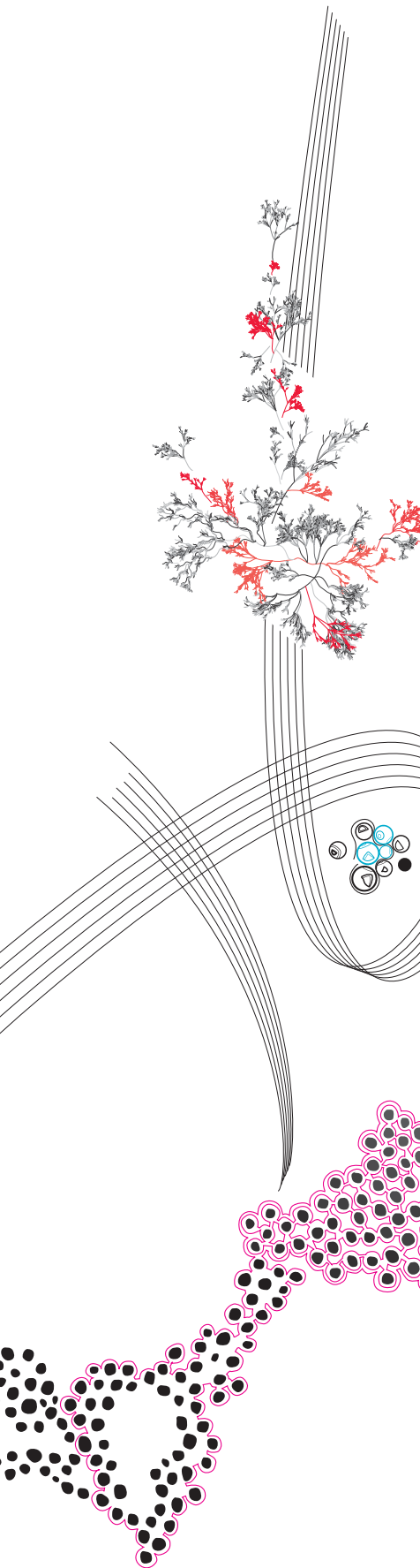




MSc Educatie in de
Bètawetenschappen

Onderzoek van Onderwijs

Van automaat tot artefact

**Over de contextuele invulling van grammatica's en
eindige automaten in het informaticaonderwijs**

Wouter van den Brink
s1803905

Begeleiders:

dr.ir. M. Timmer

dr.ir. T.J.M. Coenen

24 juni 2025

Onderzoek van onderwijs (10 EC)
voor het schoolvak informatica

Samenvatting

Met de invoer van het nieuwe examenprogramma informatica zijn de onderwerpen eindige automaten en grammatica's toegevoegd aan het programma. De auteurs van het examenprogramma hadden hierbij de intentie om deze onderwerpen instrumentaal in te zetten, en niet om er diepe theorie mee te bedrijven. Echter passen zowel de handreiking van de SLO als de gangbare lesmethoden een theoretische insteek toe.

In dit onderzoek wilden we het onderwijs over eindige automaten en grammatica's verbeteren. We hebben gepoogd de volgende onderzoeksvragen te beantwoorden:

1. Welke contextuele invulling geven docenten in het voortgezet onderwijs aan de informaticalesen over eindige automaten en grammatica's?
2. Welke aanbevelingen kunnen worden gedaan aan docenten en onderwijsondersteunende partijen om het voortgezet informatica-onderwijs over eindige automaten en grammatica's te verbeteren?

Voor de beantwoording van de vragen hebben we gebruikgemaakt van literatuuronderzoek, vragenlijsten en interviews. We construeerden een gesloten vragenlijst en namen die af onder zo veel mogelijk informaticadocenten. Een deel van de respondenten stelde zich beschikbaar voor een verdiepend interview. Met twee respondenten hebben we zo'n interview afgenomen.

Op basis van de resultaten kunnen we ten eerste indicatief concluderen dat de meeste docenten de onderwerpen te theoretisch aanvielen, hetgeen niet in lijn is met de gedachte achter deze onderwerpen in het examenprogramma, zoals opgesteld door de commissie. De theoretische insteek is waarschijnlijk het gevolg van de interpretatie van de eindtermen door de SLO en de educatieve uitgeverijen.

We brengen drie concrete adviezen uit: we adviseren (1) docenten om elkaar op te zoeken om ervaringen, ideeën en lesmateriaal uit te wisselen, (2) educatieve uitgeverijen om hun materiaal over deze onderwerpen praktischer te maken, en (3) het Ministerie van OCW om te overwegen om deze onderwerpen onder te brengen in een vwo-keuzethema en te schrappen uit het havo-examenprogramma.

Door het lage aantal respondenten is ons onderzoek indicatief van aard. Een vervolgonderzoek met een kortere vragenlijst levert weliswaar minder inzicht op, maar kan wel een hogere representativiteit hebben. Verder zien we heil in een vervolgonderzoek waarin men ontwerponderzoek doet naar geschikt lesmateriaal voor eindige automaten en grammatica's. Een geschikt ontwerpdoel voor zulk onderzoek is een betere aansluiting op de gedachtegang achter deze onderwerpen: als praktisch instrument voor de constructie van digitale artefacten.

Inhoudsopgave

1	Inleiding	5
1.1	Aanleiding en achtergrond	5
1.2	Onderzoeksopzet en hoofdvraag	6
1.3	Belang van het onderzoek	7
2	Theoretisch kader	9
2.1	Wat zijn eindige automaten en grammatica's?	9
2.2	Invulling van informaticalessen	23
3	Methode	27
3.1	Procedure	27
3.2	Respondenten	30
3.3	Instrumenten	31
3.4	Analyse	34
4	Resultaten	37
4.1	Open vragenlijst	37
4.2	Gesloten vragenlijst	39
4.3	Interviews met docenten uit het voortgezet onderwijs	41
5	Conclusies en discussie	45
5.1	Invulling van eindige automaten en grammatica's	45
5.2	Adviezen	46
5.3	Beperkingen en vervolgonderzoek	47
	Bibliografie	49
A	Open vragenlijst	51
B	Gesloten vragenlijst	61
C	Interviewvragen	83
D	Interviewtranscripties	85

1 Inleiding

Anekdote

Tijdens een SLO-evaluatiesessie over het nieuwe examenprogramma voor informatica (7 november 2024) riep een docent: „dat ellendige BNF moet eruit; niemand zit daarop te wachten!” De zaal knikte instemmend.

1.1 Aanleiding en achtergrond

De docent in bovenstaande anekdote spreekt een duidelijke wens uit: BNF moet geschrapt worden uit het examenprogramma informatica. BNF is een soort programmeertaal¹, waarmee grammatica's kunnen worden beschreven. Grammatica's zijn inderdaad een verplicht onderdeel van het examenprogramma informatica („Examenprogramma informatica havo/vwo 2019”, 2019). Dit onderwerp is in 2016 geïntroduceerd (Barendsen & Tolboom, 2016) en maakt sinds 2019 onderdeel uit van het examenprogramma.

Echter wordt er in dat examenprogramma geen woord gerept over BNF. Wel vinden we dit woord terug in de *handreiking* voor het examenprogramma: „Er zijn verschillende manieren om de productieregels van een grammatica weer te geven. Een veelgebruikt formalisme is de zogenaamde BN-vorm (...)” (SLO, 2020). Waar komt deze verwarring vandaan?

In tegenstelling tot het programma dat tot 2019 werd gehanteerd, sluiten de eindtermen van het nieuwe examenprogramma vrijwel naadloos aan bij de inhoud van informatica-vervolgopleidingen. Dat betekent dat niet alleen de praktische informaticazaken aan bod komen, zoals programmeren, maar dat er ook ruimte is gemaakt voor de theoretische grondslagen van de informatica, zoals algoritmen en datastructuren. De denkwijze die komt kijken bij deze grondslagen kan de kwaliteit van het praktische informaticawerk aanzienlijk bevorderen (Dongol e.a., 2024).

De informaticawereld ontwikkelt zich in hoog tempo. Dat vraagt om een duurzaam examenprogramma. Bij de vernieuwing van het programma is daarom gekozen voor een uitgebreid *conceptueel* programma. Dat betekent dat het programma bij abstracte concepten blijft, zoals ‘algoritmen’ (subdomein B1), ‘security’ (subdomein E2) of ‘privacy’ (subdomein F3). Deze concepten zijn redelijk waardevast. Het is aan docenten om actuele en aantrekkelijke *contexten* aan deze concepten te koppelen. Die contexten blijven zich in hoog tempo ontwikkelen. Door deze concept-contextbenadering kan het informatica-onderwijs met de tijd meegaan, terwijl het examenprogramma duurzaam wordt opgesteld: het hoeft niet met iedere ontwikkeling te worden bijgewerkt (Barendsen & Tolboom, 2016).

Om docenten te assisteren bij de ontwikkeling van hun curriculum, en het vinden van geschikte contexten, publiceerde de SLO een handreiking (SLO, 2020). In dit hulpmiddel legt de SLO uit wat de

¹Hier leggen we meer over uit in paragraaf 2.1.2.

verschillende concepten in het programma allemaal inhouden, en welke huidige contexten er zoal aan deze onderwerpen kunnen worden gekoppeld. Deze handreiking is niet leidend: docenten hoeven zich niet te houden aan de gegeven voorschriften. Helaas is het facultatieve karakter van de handreiking nergens expliciet opgenomen. We vermoeden dat docenten hierdoor in de war raken, en aannemen dat ze hun PTA en curriculum moeten aansluiten op de onderwerpen in de handreiking.

De in de handreiking gegeven uitleg en voorbeelden over eindige automaten en grammatica's zijn erg droog en theoretisch van aard. Ook (en misschien zelfs daardoor) is het beschikbare lesmateriaal over deze onderwerpen theoretisch van aard. Er wordt in de handreiking en de lesmethoden gesproken over toestandsdiagrammen, overgangen, BNF en productieregels. Het was echter nooit de bedoeling van de commissie – die het examenprogramma samenstelde – om deze onderwerpen zo droog en theoretisch aan te pakken: „het is dus de bedoeling om deze twee modellen instrumenteel te gebruiken en niet om er allerlei diepe theorie mee te bedrijven” (Barendsen & Tolboom, 2016).

Het lijkt er dus op dat de handreiking en de lesmethoden hebben gezorgd voor negatieve connotaties over deze onderwerpen in het examenprogramma's Docenten, waaronder de docent in de anekdote, maken tussen de onderwerpen en hun theoretische uitwerkingen een haast onlosmakelijke koppeling, die ertoe leidt dat ze de onderwerpen willen schrappen. Wij zijn van mening dat eindige automaten en grammatica's wel degelijk een positie kunnen krijgen in een volwaardig examenprogramma informatica, maar dat er daarvoor wel beter onderzoek moet worden gedaan naar wat die positie precies is, en hoe we het onderwijs over deze onderwerpen beter kunnen vormgeven. Met dit onderzoek hopen we daar een bijdrage in te leveren.

Waarom specifiek deze onderwerpen?

Met het nieuwe examenprogramma zijn allerlei nieuwe onderwerpen geïntroduceerd, waaronder eindige automaten en grammatica's. Er is, zowel online als in onderzoek — bijvoorbeeld Grgurina e.a. (2023) — genoeg kritiek te vinden over deze onderwerpen, en over de afwezigheid van andere onderwerpen. Waarom hebben we dan in dit onderzoek specifiek voor deze onderwerpen gekozen?

Ten eerste vinden we het eenvoudigweg hele boeiende onderwerpen. Ook hebben we de persoonlijke ervaring dat deze onderwerpen snel als eerste worden genoemd wanneer het examenprogramma met een kritische blik wordt besproken. Bovendien voelt een significant aantal docenten zich minder competent in het onderwijzen van deze onderwerpen (Grgurina e.a., 2023).

1.2 Onderzoeksopzet en hoofdvraag

Informeel uitgedrukt is het hoofddoel van dit onderzoek de verbetering van het onderwijs over eindige automaten en grammatica's. We hebben door middel van een vragenlijst en semi-gestructureerde interviews op mesoniveau in kaart gebracht welke contextuele keuzes er door docenten worden gemaakt voor de subdomeinen eindige automaten en grammatica's. We analyseerden niet alleen hoe deze onderwerpen vorm krijgen in het curriculum, maar mede met behulp van diepgaande interviews bekeken we ook hoe die vormgeving tot stand is gekomen en waar docenten mogelijkheden en uitdagingen zien voor verbetering van hun lessen over deze onderwerpen.

De onderzoeksvragen luiden als volgt:

Onderzoeksvragen

OV1. Welke contextuele invulling geven docenten in het voortgezet onderwijs aan de informati-

calessen over eindige automaten en grammatica's?

- (a) Wat is de inhoud, omvang, toetsing, plaats in het curriculum en inbedding van lessen over eindige automaten en grammatica's in het voortgezet onderwijs?
- (b) Waarom hebben docenten voor deze invulling gekozen?
- (c) Welke wensen hebben docenten voor de toekomst van hun lessen over eindige automaten en grammatica's?

OV2. Welke aanbevelingen kunnen worden gedaan aan docenten en onderwijsondersteunende partijen om het voortgezet informatica-onderwijs over eindige automaten en grammatica's te verbeteren?

De leerdoelen voor ieder subdomein zullen door de vrijheid in de inrichting van het vak verschillend zijn van school tot school. Dat maakt het ingewikkeld om een lijn te trekken waarmee de leerresultaten per subdomein voor iedere school met elkaar te vergelijken zijn. Daarom zijn we in het onderzoek niet ingegaan op de behaalde leerresultaten.

We benadrukken dat deze beperkingen van het domein van het onderzoek beslist niet aangeven dat de uitgezonderde onderwerpen het onderzoeken niet waard zijn. De lezer wordt aangemoedigd prangende vragen als „welke werkvormen zijn waardevol voor onderwijs over eindige automaten en grammatica's” en „wat is de leeropbrengst van lessen over subdomein B3 en B4” te beantwoorden in toekomstig onderzoek.

1.3 Belang van het onderzoek

Dit onderzoek laat zien hoe docenten en onderwijsondersteunende partijen samen kunnen werken aan beter en aantrekkelijker informatica-onderwijs voor eindige automaten en grammatica's.

We geven inzicht in:

- De huidige informaticalessen over eindige automaten en grammatica's;
- Het 'waarom' achter deze invulling, en de wensen van docenten voor verbetering ervan.

Bovendien brengen we adviezen en opbrengsten voor:

- Docenten, die hun lessen kunnen vernieuwen en verbeteren aan de hand van onze inzichten en conclusies, en inspiratie kunnen halen uit de input van hun (landelijke) collega's;
- Educatieve uitgeverijen, die hun lesmateriaal op gelijke wijze kunnen verbeteren;
- Het ministerie van OCW, dat onze input kan gebruiken om het examenprogramma te verbeteren;
- Leerlingen, die aantrekkelijker onderwijs kunnen krijgen over deze onderwerpen, die beter aansluiten bij hun leefwereld en misschien zelfs hun vervolgopleiding.

2 Theoretisch kader

2.1 Wat zijn eindige automaten en grammatica's?

Eindige automaten en grammatica's zijn theoretische concepten die in de informatica worden gebruikt om een bepaald gedrag te modelleren. Dat gedrag kan van alles zijn: het gedrag van een robot, de communicatie tussen twee computers of (in het bijzonder bij grammatica's) de correcte schrijfwijze van een bepaalde taal. In deze paragraaf geven we een definitie voor zowel een eindige automaat als een grammatica. Ook lichten we toe hoe deze concepten worden gebruikt in het vervolgonderwijs.

Vervolgens zetten we een didactische bril op. We bekijken de positie van deze concepten in internationale informatica-examenprogramma's, en proberen de herkomst in het Nederlandse examenprogramma informatica te duiden. We bekijken ook of deze onderwerpen terugkomen in het vervolgonderwijs. Ten slotte duiken we weer de literatuur in: welke didactische kennis is er al opgedaan over deze onderwerpen? Welke uitdagingen komen er kijken bij het gebruik van eindige automaten en grammatica's in het voortgezet onderwijs?

2.1.1 Wat is een eindige automaat?

Een eindige automaat is een schematische weergave van een bepaald systeem. Dat systeem verkeert altijd in één van een eindig aantal toestanden. Elke toestand heeft een bepaald label. De automaat kan naar een andere toestand gaan. Zo'n overgang naar een andere toestand is altijd vergezeld van een bepaald symbool. Ten slotte zijn er twee bijzondere toestanden: er is één begintoestand, waar het systeem in verkeert 'aan het begin', en er kunnen meerdere eindtoestanden zijn.

Er zijn meerdere soorten eindige automaten. De eenvoudigste soort is de *deterministische* eindige automaat. De formele definitie hiervan is gegeven in definitie 1. Het determinisme slaat hierbij op de transities: voor iedere toestand en voor ieder symbool is bekend wat de volgende toestand zal zijn.

Definitie 1 (Deterministische eindige automaat)

Een deterministische eindige automaat is een 5-tupel $M = (Q, \Sigma, \delta, q_0, F)$, met:

- Q een eindige set van toestanden;
- $q_0 \in Q$ de unieke begintoestand;
- Σ het alfabet — een set symbolen;
- δ een totale functie $\delta : Q \times \Sigma \rightarrow Q$ de overgangsfunctie;
- $F \subseteq Q$ de set van eindtoestanden.

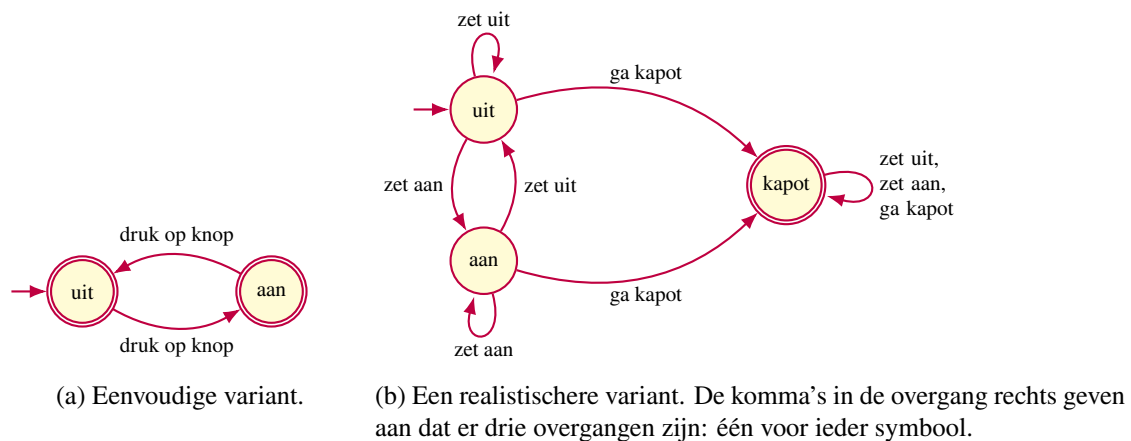
(Hopcroft e.a., 2007, p. 45)

Schematisch worden toestanden doorgaans getekend als cirkels of ovalen. Het label van de toestand wordt in de cirkel of ovaal genoteerd. Een toestand met dubbele rand is een eindtoestand. Toestands-overgangen worden aangegeven met een pijl. Naast de pijl wordt het symbool van die overgang genoteerd. De begintoestand wordt gemarkeerd door een pijl te tekenen die de begintoestand in gaat, maar ‘nergens’ vandaan komt. Deze pijl heeft nooit een symbool.

Wat het precies betekent wanneer een systeem zich in een bepaalde toestand bevindt, en of die toestand een eindtoestand is, is niet gedefinieerd. Die betekenis wordt gegeven aan de hand van de toepassing van de eindige automaat. Veel introducerende teksten over eindige automaten gebruiken een verkeerslicht als voorbeeld. De toestanden zijn dan ‘rood’, ‘oranje’ en ‘groen’.

Voor een eindige automaat die het gedrag van een computerprogramma modelleert, betekent de eindtoestand bijvoorbeeld waarschijnlijk ‘het programma is afgesloten’. Bij talen betekent een eindtoestand juist ‘deze zin is correct’. Toestanden in een automaat voor het gedrag voor een robot zijn vaak direct te zien als de ‘modus’ waarin de robot werkt, terwijl ze bij een taal-automaat een veel abstractere betekenis hebben.

In fig. 2.1a is een eenvoudige deterministische eindige automaat weergegeven, die het gedrag van een lichtknop modelleert. In deze automaat is een toestand een fysieke toestand: de knop staat uit of aan. De eindige automaat ernaast, in fig. 2.1b, is realistischer: de lichtknop kan kapotgaan, en dat is dan ook de eindtoestand.



Figuur 2.1: Twee deterministische eindige automaten voor een lichtknop.

Vrij naar Hopcroft e.a., 2007, fig. 1.1, p. 3.

Niet-deterministische eindige automaat

Bij veel toepassingen van eindige automaten zijn we niet geïnteresseerd in *alle* overgangen. Een deterministische eindige automaat van een verkeerslicht zou bijvoorbeeld ook overgangen moeten hebben van rood naar oranje, en oranje naar groen. Maar in Nederland zijn zulke overgangen niet mogelijk. Er zijn dan twee mogelijke oplossingen. We kunnen een extra toestand maken, met het label ‘fout’, en laten iedere ‘onmogelijke’ overgang hiernaartoe gaan. Ten slotte voegen we overgangen toe voor ieder symbool die terugkeren naar deze toestand, zodat de automaat als het ware ‘vastloopt’.

We kunnen er ook voor kiezen deze overgangen niet te tekenen. In dat geval is er sprake van een *niet-deterministische* eindige automaat (definitie 2). Het is dan niet langer bekend voor iedere toestand en voor ieder symbool wat de volgende toestand zal zijn. Bovendien is het bij niet-deterministische eindige automaten mogelijk om meerdere overgangen met hetzelfde symbool toe te kennen vanaf dezelfde toestand. Dat betekent dat het mogelijk wordt dat er meerdere volgende toestanden zijn voor

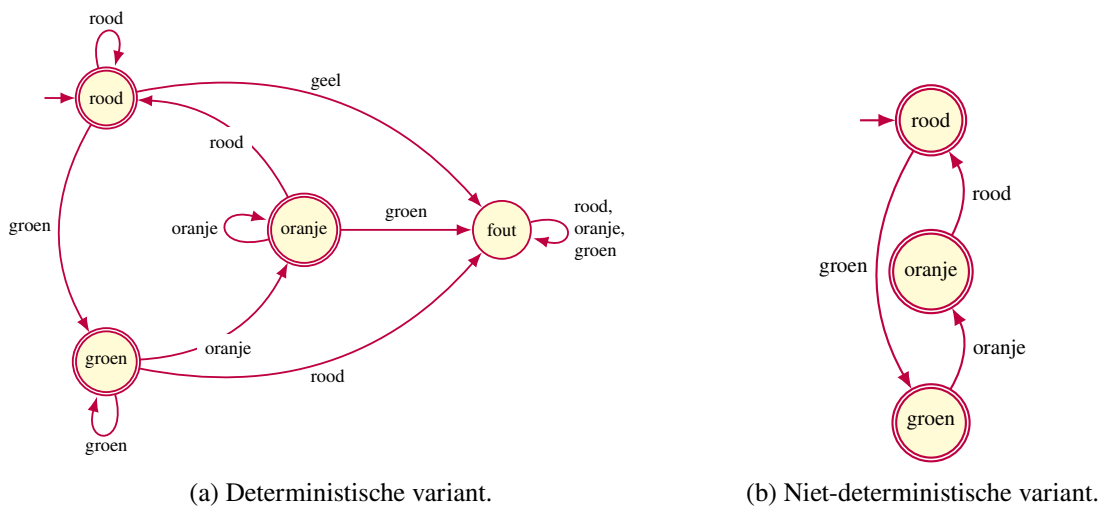
een bepaalde combinatie van toestand en symbool. De eindige automaat moet dan als het ware ‘raden’ welke overgang er genomen moet worden.

Definitie 2 (Niet-deterministische eindige automaat)

Een niet-deterministische eindige automaat is een 5-tupel $M = (Q, \Sigma, \delta, q_0, F)$, met:

- Q een eindige set van toestanden;
- $q_0 \in Q$ de unieke begintoestand;
- Σ het alfabet — een set symbolen;
- δ een functie $\delta : Q \times \Sigma \rightarrow \mathcal{P}(Q)$ de overgangsfunctie;
- $F \subseteq Q$ de set van eindtoestanden.

(Hopcroft e.a., 2007, p. 57)



Figuur 2.2: Twee eindige automaten voor een verkeerslicht.

Figuren 2.2a en 2.2b laten het verschil tussen een equivalent deterministische en niet-deterministische eindige automaat voor een Nederlands verkeerslicht zien. De deterministische variant heeft meer toestanden en overgangen en is daarom moeilijker om te interpreteren. De niet-deterministische variant is gemakkelijker om te interpreteren, maar lastiger om te analyseren. Voor analyses worden zulke automaten dan ook veelal eerst omgezet in hun deterministische variant.

Eindige automaten en taalstudie

Uit definitie 1 volgt dat de labels van de overgangen van een eindige automaat samen een *alfabet* vormen. Dit is een set van *symbolen*. In het geval van de lichtknop zijn de symbolen ‘zet uit’, ‘zet aan’ en ‘ga kapot’. Het alfabet is dus $\Sigma = \{\text{zet uit, zet aan, ga kapot}\}$.

Met het alfabet kunnen *woorden* worden gevormd. Dat zijn combinaties van één of meerdere symbolen uit het alfabet. ‘Zet uit, zet aan, ga kapot’ is bijvoorbeeld een woord, bestaande uit drie symbolen. Een *taal* is ten slotte een set, bestaande uit een aantal woorden. Deze termen zijn formeel gedefinieerd in definitie 3.

Definitie 3 (Taal)

Gegeven een alfabet Σ :

- Σ^* is de set van alle woorden $a_1 a_2 \dots a_n$, zodanig dat $a_i \in \Sigma$ voor elke $1 \leq i \leq n$. $n = 0$ is mogelijk.
- Een woord met geen symbolen is λ .
- Een taal $\mathcal{L}$ is een set van woorden: $\mathcal{L} \subseteq \Sigma^*$.

(Hopcroft e.a., 2007, p. 28–31)

Bij eindige automaten (en grammatica's) proberen we een taal eenvoudig weer te geven. Het geeft geen zin om alle mogelijke Nederlandse zinnen op te schrijven; daarvan zijn er oneindig veel. Daarom schrijven we woordenboeken en stellen we grammaticale regels op. Zo ook heeft het geen zin om iedere mogelijke levensloop van een lichtknop op te schrijven; in plaats daarvan stellen we een eindige automaat op.

Met een eindige automaat kunnen woorden worden gevormd. Door verschillende overgangen te doorlopen en telkens te noteren welk symbool bij iedere overgang hoort, wordt een woord gevormd. We zeggen dat de automaat de symbolen in het woord *produceert*.

Door de eenvoudige lichtknop aan en daarna uit te zetten, ontstaat bijvoorbeeld het woord 'zet aan, zet uit'. Het woord *kan* af zijn bij een eindtoestand. Dit betekent ook dat een woord nog niet af is, wanneer de automaat zich niet in een eindtoestand bevindt. Het woord 'zet aan', 'zet uit' is bijvoorbeeld *niet* een geldig woord voor de realistische lichtknop in fig. 2.1b, omdat 'uit' geen eindtoestand is.

Andersom kan ook geverifieerd worden dat een bepaald woord geproduceerd kon worden door de eindige automaat, door één voor één de symbolen in het woord te volgen in de automaat. Als dit pad eindigt in een eindtoestand, dan kan het woord geproduceerd worden door de automaat. We zeggen dan dat het woord is *verwerkt* door de automaat.

Formeel maken we gebruik van de uitgebreide overgangsfunctie (definitie 4) voor het bepalen van de toestand waarin de automaat zich bevindt na het verwerken van alle symbolen in een woord.

Definitie 4 (Uitgebreide overgangsfunctie)

Gegeven een overgangsfunctie δ , beschrijft de uitgebreide overgangsfunctie $\hat{\delta}$ wat er gebeurt wanneer we beginnen in een bepaalde staat en een woord van symbolen volgen.

De functie accepteert een toestand q en een woord w en produceert een toestand p — de toestand die de eindige automaat bereikt na het beginnen in toestand q en het verwerken van alle symbolen in w .

$\hat{\delta}$ wordt opgebouwd met behulp van inductie op de lengte van het woord:

- Basis: $\hat{\delta}(q, \epsilon) = q$. Als we in toestand q zijn, en er zijn geen symbolen om te verwerken, dan blijven we in toestand q .
- Inductie: Laat w de vorm $x\alpha$ hebben, met x een woord en α een symbool. Met andere woorden: α is het laatste symbool in w , en x is alles in w behalve het laatste symbool. Dan geldt:

$$\hat{\delta}(q, w) = \delta(\hat{\delta}(q, x), \alpha)$$

(Hopcroft e.a., 2007, p. 49)

We kunnen de uitgebreide overgangsfunctie vervolgens gebruiken om een formele definitie te geven voor de taal die wordt geaccepteerd door een eindige automaat:

Definitie 5 (Taal van een deterministische eindige automaat)

Gegeven een deterministische eindige automaat $M = (Q, \Sigma, \delta, q_0, F)$, en uitgebreide overgangsfunctie $\hat{\delta}$:

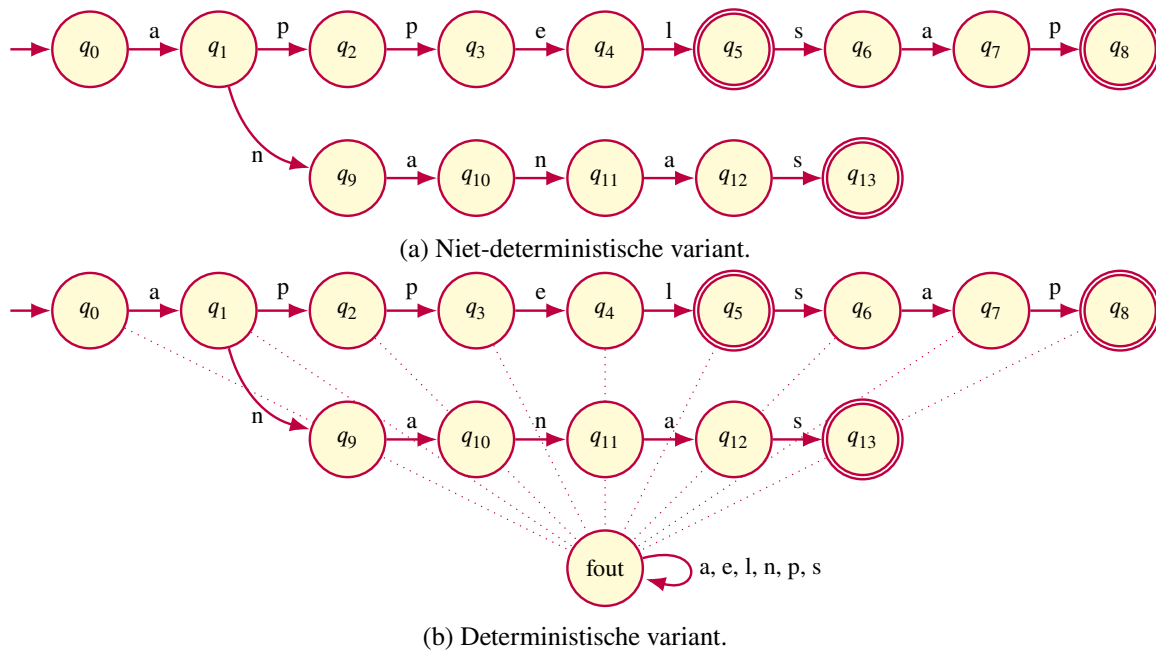
$$\mathcal{L}(M) = \{w \mid \hat{\delta}(q_0, w) \in F\}$$

In woorden: de taal van een deterministische eindige automaat bestaat uit ieder woord waarvoor de uitgebreide overgangsfunctie, beginnend bij de begintoestand, resulteert in een eindtoestand.

(Hopcroft e.a., 2007, p. 52)

Voor de meeste talen geldt dat de set geldige woorden veel kleiner is dan de set *mogelijke* woorden. Een deterministische automaat voor zulke talen zou erg groot worden: voor elke ‘spelfout’ moet een overgang worden getekend. Niet-deterministische eindige automaten bieden hierin een uitkomst. Is er een symbool in een woord waarvoor geen overgang is gedefinieerd, dan wordt het woord klaarblijkelijk niet door deze automaat geaccepteerd.

De niet-deterministische eindige automaat in fig. 2.3a beschrijft bijvoorbeeld de taal bestaande uit de woorden ‘appel’, ‘ananas’ en ‘appelsap’. Het alfabet van deze eindige automaat is $\Sigma = \{a, e, l, n, p, s\}$. De taal is $\mathcal{L} = \{\text{appel}, \text{ananas}, \text{appelsap}\}$.



Figuur 2.3: Eindige automaten voor de woorden ‘appel’, ‘ananas’ en ‘appelsap’.

Dezelfde automaat als deterministische automaat is deels weergegeven in fig. 2.3b. Deze variant heeft een extra toestand ‘fout’, met vanaf iedere toestand voor bijna iedere letter een overgang naar deze extra toestand. Op de stippelijntjes horen dus eigenlijk meerdere overgangen te staan, maar omwille van de leesbaarheid zijn die niet getekend.

Definitie 5 volstaat niet om te bewijzen dat $\mathcal{L} = \{\text{appel}, \text{ananas}, \text{appelsap}\}$ inderdaad de taal van deze automaat is. Het resultaat van $\delta(q, a)$ voor een symbool a is bij non-deterministische automaten een set, die leeg kan zijn. Daarom kan $\hat{\delta}$ niet worden opgebouwd zoals in definitie 4.

Voor het vaststellen van de taal van een niet-deterministische eindige automaat maken we gebruik van configuraties (definitie 6).

Definitie 6 (Configuratie van een niet-deterministische eindige automaat)

Een configuratie van een niet-deterministische eindige automaat $M = (Q, \Sigma, \delta, q_0, F)$ is een paar $[q, w] \in Q \times \Sigma^*$, met:

- q de huidige toestand van de automaat;
- w het woord dat nog door de automaat moet worden gelezen.

De configuratie $[q_1, \text{nanas}]$ geeft bijvoorbeeld aan dat de automaat zich in toestand q_1 bevindt, en het woord *nanas* nog moet lezen. Naar alle waarschijnlijkheid is de automaat begonnen met het verwerken/produceren van het woord ‘ananas’.

We definiëren nu, met behulp van definitie 6, de eenstapsrelatie:

Definitie 7 (Eenstapsrelatie)

Gegeven een niet-deterministische eindige automaat $M = (Q, \Sigma, \delta, q_0, F)$, is de eenstapsrelatie gedefinieerd als $[q, aw] \vdash [q', w]$, met $q' \in \delta(q, a)$.

$\vdash^*$ betekent het meermaals (of niet) toepassen van $\vdash$.

De $\vdash^*$ -relatie is een soort *orakel-variant* van de uitgebreide transitiefunctie. Als $[q_0, w] \vdash^* [q_1, \lambda]$ geldt, dan is er klaarblijkelijk *een* pad in de automaat te bewandelen vanaf q_0 met de symbolen in het woord w , dat eindigt in q_1 . De relatie ‘weet’ als het ware welke toepassingen van $\vdash$, welke overgangen, hiertoe leiden. In de praktijk is dit niet altijd in één oogopslag te zien. De automaat moet daarvoor eerst vaak worden omgezet in een deterministische automaat.

Ten slotte kunnen we een definitie maken voor de taal van een niet-deterministische eindige automaat:

Definitie 8 (Taal van een niet-deterministische eindige automaat)

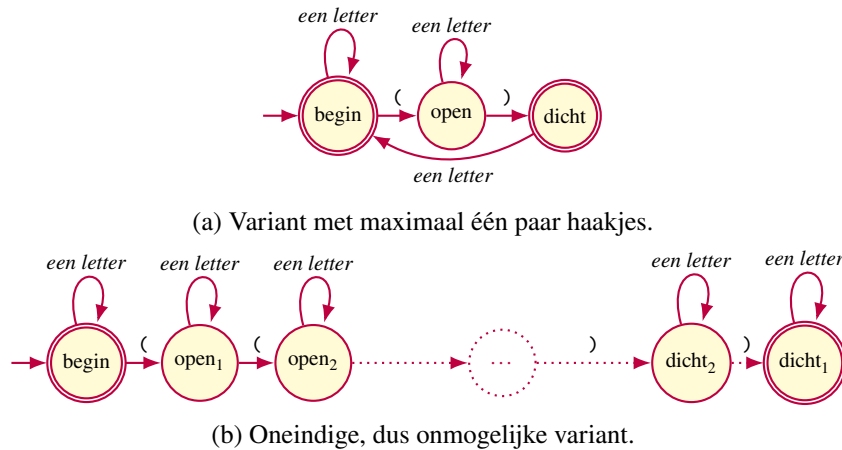
Gegeven een niet-deterministische eindige automaat $M = (Q, \Sigma, \delta, q_0, F)$:

M accepteert het woord w als $[q_0, w] \vdash^* [q, \lambda]$, met $q \in F$. In symbolen:

$$\mathcal{L}(M) = \{w \in \Sigma^* \mid \exists q \in F : [q_0, w] \vdash^* [q, \lambda]\}$$

Sommige talen kunnen niet worden beschreven met eindige automaten, omdat die talen te complexe regels hebben. Dat komt omdat eindige automaten geen ‘geheugen’ hebben. Het is altijd alleen bekend in welke toestand de eindige automaat zich *op een bepaald moment* bevindt, maar nooit in welke toestanden de automaat zich hiervoor bevond¹.

¹Natuurlijk zijn de toestanden apart te noteren in een lijst. Deze lijst maakt geen deel uit van de eindige automaat, en daarom zeggen we dat de eindige automaat geen geheugen heeft.



Figuur 2.4: Twee deterministische eindige automaten voor zinnen met haakjes.

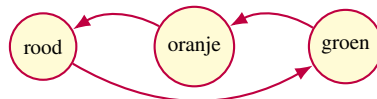
Deze tekortkoming maakt het bijvoorbeeld onmogelijk om een eindige automaat te maken die correct bijhoudt of haakjes worden afgesloten in een zin. Ieder openend haakje ‘(’ dient *ergens in de zin* te worden afgesloten met een sluitend haakje ‘)’’. Er moet dus worden bijgehouden hoeveel haakjes er ‘open zijn’. In een eindige automaat moet er dan voor ieder mogelijk open haakje een (aantal) toestand(en) zijn. Een zin kan een oneindig aantal haakjes hebben, en dat zou een oneindig aantal toestanden vereisen. Dat is volgens definitie 1 niet mogelijk. Daarom is het onmogelijk om dit soort talen te omschrijven met een eindige automaat. Dit probleem is zichtbaar gemaakt in fig. 2.4. Soms bieden grammatica’s een uitkomst, die verderop in dit hoofdstuk worden beschreven.

Eindige automaten in de lesmethoden

Tot nu toe hebben we een zeer theoretische beschrijving gegeven van eindige automaten. De grote informatica-lesmethoden — Fundament, Informatica-Actief en VO-Content/Enigma — behandelen dit onderwerp behoorlijk theoretisch, maar met minder diepgang. De meeste formele definities worden bijvoorbeeld niet gegeven. Ook is er geen uitleg over taal, woorden en symbolen.

Verder maken de methoden alle drie gebruik van niet-deterministische eindige automaten in het lesmateriaal. Deze automaten geven meer vrijheid in de constructie. Bovendien worden symbolen bij overgangen vaak weggelaten. Een automaat bevat dan alleen toestanden en pijlen. In een toestandsdiagram van bijvoorbeeld een stoplicht kan dat didactisch wenselijk zijn: de tekst ‘verander van kleur’ bij iedere overgang kan worden gezien als ruis. Ook worden de labels van toestanden met regelmaat weggelaten, en is er niet altijd een duidelijke begintoestand.

In fig. 2.5 is weergegeven hoe zo’n eindige automaat met minimale ruis eruit kan zien. Deze automaat voldoet niet meer aan de formele definitie van een eindige automaat (definities 1 en 2), maar volstaat nog steeds om het gedrag van een verkeerslicht schematisch weer te geven.



Figuur 2.5: Niet-deterministische eindige automaat voor een Nederlands verkeerslicht. Er zijn geen begin- en eindtoestanden en de overgangen hebben geen labels.

Turingmachine

In de lesmethoden Fundament en Informatica-Actief worden Turingmachines behandeld als verdiepend onderdeel van het lesmateriaal over eindige automaten en grammatica's. Een Turingmachine is een combinatie van een deterministische eindige automaat en een geheugenband.

De geheugenband is een oneindige band, waarop de automaat symbolen kan lezen en schrijven. Het alfabet van de geheugenband is gelijk aan het alfabet van de eindige automaat, plus een speciaal blanco-symbool B , om aan te geven dat de geheugenband op een bepaalde positie leeg is.

Een overgang in de eindige automaat van een Turingmachine bestaat niet langer uit een symbool, maar uit twee symbolen en een bewegingsrichting — L of R voor links en rechts.

Een overgang van q_0 naar q_1 kan bijvoorbeeld genoteerd zijn als $q_0 \xrightarrow{a/b R} q_1$. Met deze overgang worden de volgende stappen beschreven:

1. De eindige automaat bevindt zich in toestand q_0 ;
2. Symbool a wordt van de band gelezen;
3. Symbool b wordt op de band geplaatst;
4. De machine verplaatst één positie naar rechts;
5. De eindige automaat bevindt zich in toestand q_1 .

Uit deze beschrijving volgt dat de machine zich dus niet alleen in een bepaalde toestand bevindt, maar ook in een bepaalde positie op de geheugenband. Ook is de invoer inherent onderdeel geworden van de machine: men kan de Turingmachine van invoer voorzien door symbolen op de geheugenband te plaatsen.

De overgang is in zekere zin voorwaardelijk geworden: een overgang van q_0 naar q_1 is alleen mogelijk als er zich een a op de geheugenband bevindt, op de huidige positie van de machine.

Turingmachines zijn bedacht door de wiskundige Alan M. Turing, als instrument om te beredeneren wat de limieten van computers zijn. Elke berekening die met een computer te maken is, is ook met een Turingmachine te maken, en andersom. Elke berekening die *niet* met een Turingmachine te maken is, kan dus ook niet door een computer worden uitgevoerd. Turingmachines vormen hiermee een belangrijk instrument voor informatici om een beeld te kunnen vormen over de mogelijkheden en beperkingen van computers.

Een formele definitie van de Turingmachine is gegeven in definitie 9.

Definitie 9 (Turingmachine)

Een Turingmachine is een 6-tupel $T = (Q, \Sigma, \Gamma, \delta, q_0, F)$, met:

- Q een set toestanden;
- Σ het invoeralfabet – een set symbolen;
- $\Gamma = \Sigma \cup \{B\}$ het geheugenbandalfabet, met speciaal blanco-symbool B ;
- δ een functie $\delta: Q \times \Gamma \rightarrow \Gamma \times \{L, R\} \times Q$ – de overgangsfunctie;
- $q_0 \in Q$ de unieke begintoestand;
- $F \subseteq Q$ de set van eindtoestanden.

(Hopcroft e.a., 2007, p. 327)

2.1.2 Wat is een grammatica?

Een grammatica is, net als een eindige automaat, een manier om een bepaalde taal te omschrijven. Grammatica's maken het mogelijk om complexere talen te beschrijven, met strengere regels. Ze beschikken in zekere zin over een soort geheugen.

Grammatica's bestaan uit verschillende productieregels. Die beschrijven hoe bepaalde variabelen – niet te verwarren met variabelen in een computerprogramma – moeten worden opgebouwd. Een variabele kan bestaan uit een combinatie van symbolen en andere variabelen. Bovendien introduceren we bij grammatica's het lege symbool ϵ , om de constructie van lege variabelen mogelijk te maken. De formele definitie van een grammatica is gegeven in definitie 10.

Definitie 10 (Grammatica)

Een grammatica is een 4-tupel $G = (V, \Sigma, P, S)$, met:

- V een set van variabelen;
- Σ het alfabet — een set symbolen;
- $P \subseteq V \times (V \cup \Sigma)^*$ een set productieregels;
- $S \in V$ het startsymbool.

(Hopcroft e.a., 2007, p. 173)

$S \rightarrow \text{een letter } S$

$S \rightarrow (S) S$

$S \rightarrow \epsilon$

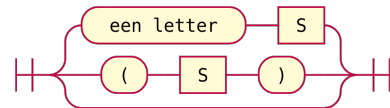
(a) Eenvoudige variant.

$S \rightarrow \text{een letter } S$

| $(S) S$

| ϵ

(b) Met compacte notatie.



(c) Als railroad-diagram.

Figuur 2.6: Grammatica voor zinnen met haakjes.

De grammatica in fig. 2.6a is een grammatica voor zinnen met haakjes – een taal die niet te beschrijven is met eindige automaten. Deze grammatica is beschreven in de Backus-Naurvorm (BNF). In deze vorm bestaat een grammatica uit een aantal regels. Iedere regel is te lezen als "de variabele ... bestaat uit ...". De eerste regel zegt bijvoorbeeld "de variabele S bestaat uit een letter, gevolgd door een S ". In deze grammatica bestaat de variabele S dus ófwel uit een letter, gevolgd door een S , ófwel uit een S tussen haakjes met daar nog een S achter, ófwel uit het lege symbool. Figuur 2.6b laat een compactere notatie van dezelfde grammatica zien, waar verticale strepen alternatieven voor dezelfde variabele weergeven.

Zowel fig. 2.6a als fig. 2.6b maken gebruik van de Backus-Naurvorm (BNF) om de grammatica te beschrijven. In fig. 2.6c is dezelfde grammatica voor zinnen met haakjes afgebeeld als railroad-diagram. In zo'n diagram is ieder woord van de taal te construeren door een 'treinreis' te maken van het beginpunt (de linkerkant) naar het eindpunt (de rechterkant). Onderweg komt de 'trein' langs verschillende 'stations' — de gele vlakken. Een rond vlak stelt een symbool voor; een rechthoekig vlak is een variabele. Door de symbolen, gegenereerd door ieder station, te noteren, ontstaat een woord.

Een alternatief voor BNF en railroad-diagrammen is *extended BNF* (EBNF). Dit is een uitbreiding van BNF, die de volgende extra mogelijkheden biedt:

- Met vierkante haakjes kan men aangeven dat iets 0 of 1 keer moet voorkomen. Onderstaande

regel zegt bijvoorbeeld dat een S bestaat uit een S tussen haakjes, *eventueel* gevolgd door nog een S :

$$S \rightarrow (S) [S]$$

- Met accolades kan men aangeven dat iets 0 of meer keer moet voorkomen. Onderstaande regel zegt bijvoorbeeld dat een S bestaat uit 0 of meer letters, gevolgd door nog een S :

$$S \rightarrow \{ \text{een letter} \} S$$

De '1 of meer keer'-operator bestaat niet. Door het te herhalen symbool vóór de accolades te herhalen, kan men aangeven dat iets 1 of meer keer moet voorkomen. Onderstaande regel zegt bijvoorbeeld dat L bestaat uit 1 of meer keer 'ha'.

$$L \rightarrow \text{ha} \{ \text{ha} \}$$

De taal van L is dus $\mathcal{L}_L = \{\text{ha}, \text{haha}, \text{hahaha}, \dots\}$.

Zowel BNF-grammatica's als railroad-diagrammen zijn modellen voor hetzelfde concept: een taal. Hieruit volgt dat iedere BNF-grammatica is om te zetten in een railroad-diagram, en vice versa. Railroad-diagrammen zijn doorgaans eenvoudiger voor mensen om te interpreteren, en worden derhalve vaak ingezet in bijvoorbeeld documentatie over programmeertalen².

Woorden genereren met een grammatica is niet veel complexer dan met een eindige automaat. Door telkens een productieregel voor een symbool uit te werken, ontstaat een woord. Productieregels met ϵ zorgen ervoor dat dit proces niet voor altijd door blijft gaan. Het woord 'a(b)' ontstaat bijvoorbeeld als volgt:

1. S
2. $a S$ (regel 1)
3. $a (S) S$ (regel 2)
4. $a (S) \epsilon$ (regel 3)
5. $a (b S) \epsilon$ (regel 1)
6. $a (b \epsilon) \epsilon$ (regel 3)

De inverse van deze operatie — het vaststellen of een woord wordt geaccepteerd door een grammatica — is minder eenvoudig. Het is van tevoren niet bekend welke productieregels er zijn gebruikt. Daarom moeten alle alternatieven worden overwogen. Er bestaan verschillende algoritmen om dit proces efficiënter te maken.

Grammatica's vinden in de informatica hun toepassing vooral in de programmeertalen. Programmeertalen worden gedefinieerd met een grammatica. Het is bovendien relatief eenvoudig om een *parser* — het computerprogramma dat andere computerprogramma's leest en interpreteert — te schrijven voor een bepaalde grammatica. Auteurs van programmeertalen gebruiken deze eigenschap om hun programmeertaal te ontwikkelen: ze ontwerpen een programmeertaal in de vorm van een grammatica, en gebruiken bestaande algoritmen om er (eventueel geautomatiseerd) een parser voor te maken. Dat kan bijvoorbeeld met het gratis programma ANTLR³.

²Zie bijvoorbeeld de diagrammen van SQLite: sqlite.org/syntaxdiagrams.html

³www.antlr.org

2.1.3 Eindige automaten en grammatica's in andere curricula

Eindige automaten en grammatica's zijn niet alleen in het Nederlandse examenprogramma opgenomen. Ook in andere landen leren middelbare scholieren over deze onderwerpen. Zo ontdekten Barendsen en Steenvoorden dat automaten onder andere in het Franse curriculum terugkomen, maar niet in de CAS-richtlijnen, noch in de CSTA-standaarden (Barendsen & Steenvoorden, 2016). De CAS-richtlijnen worden gepubliceerd door de organisatie *Computing at School* en zijn in 2013 in gebruik genomen in Engeland, voor het vak *Computing*. De CSTA-standaarden worden door de Amerikaanse organisatie *Computing Science Teachers Association* gepubliceerd en worden door verschillende Amerikaanse docenten gebruikt.

De onderwerpen komen ook terug bij informatica-vervolgopleidingen. Studenten maken (opnieuw) kennis met de theoretische grondslagen en leren de formele eigenschappen kennen en gebruiken. Vervolgens kunnen de kennis en vaardigheden worden toegepast in andere vakken en projecten, zoals het construeren van een nieuwe programmeertaal.

Dat een onderwerp in het vervolgonderwijs voorkomt, is geen argument om het niet (meer) te behandelen op het voortgezet onderwijs. Zo laat onderzoek zien dat het uitspreiden van leeractiviteiten over één onderwerp over langere tijd leidt tot beter leerrendement (o.a. Cepeda e.a. (2006)). Bovendien kan het leren van de kennis en vaardigheden die horen bij de informatica-grondslagen, waar eindige automaten en grammatica's deel van uitmaken, leiden tot betere resultaten in andere onderwerpen van het vak, zoals programmeren (Dongol e.a., 2024).

De *association for computing machinery* (ACM) publiceert met regelmaat richtlijnen voor de inrichting van een informaticacurriculum voor bacheloropleidingen („undergraduate degree programs”). In deze richtlijnen worden eindige automaten en grammatica's genoemd als onderwerpen waar iedere afgestudeerde informaticastudent mee bekend moet zijn (Kumar e.a., 2023).

2.1.4 Vervolgonderwijs

We analyseerden de studieprogramma's van informatica-vervolgopleidingen op hogescholen en universiteiten. In deze analyse bekeken we studieprogramma's en vakbeschrijvingen om te bepalen of eindige automaten en grammatica's worden behandeld in deze opleidingen.

Eerst hebben we met behulp van de website studiekeuze123.nl gezocht naar alle voltijd informatica-vervolgopleidingen die door informaticaleerlingen na de middelbare school te volgen zijn. We maakten hiervoor gebruik van de volgende filters:

- Zoekterm: 'informatica'
- Soort studie: hbo bachelor OF wo bachelor
- Studierichting: exact en informatica
- Opleidingsvorm: voltijd

Deze zoekopdracht resulteerde in 148 opleidingen. Deze resultaten zijn handmatig gescreend om dubbele opleidingen op verschillende locaties en niet-relevante opleidingen uit te sluiten. Dit bracht de uiteindelijke dataset naar 21 opleidingen. Dit selectieproces is grafisch weergegeven in fig. 2.7.

De betrouwbaarheid is gewaarborgd door alle opleidingen systematisch te analyseren aan de hand van dezelfde zoekstrategie en criteria.

Van iedere geselecteerde opleiding is via de officiële website van het opleidingsinstituut het studieprogramma geraadpleegd. Daarbij is gezocht naar vakken met expliciete aandacht voor eindige automaten en grammatica's. Aan de hand van de beschrijvingen van deze vakken zijn de volgende gegevens over de opleiding genoteerd:

- Zijn er vakken over eindige automaten en grammatica's?
- Zijn de vakken verplicht in het studieprogramma?
- Worden de vakken schriftelijk en/of praktisch getoetst?

Alle geanalyseerde opleidingen en de resultaten van de analyses zijn opgenomen in tabel 2.1. Meest opvallend in deze tabel is dat geen van de geanalyseerde hbo-opleidingen een vak aanbiedt waarin eindige automaten of grammatica's worden behandeld.

Geen enkele opleiding in de dataset biedt losse vakken aan voor eindige automaten en grammatica's. Opleidingen die wel deze onderwerpen behandelen, doen dat in één breder vak. Dit vak betreft altijd een combinatievak waar andere theoretische onderwerpen worden behandeld die verwant zijn aan eindige automaten en grammatica's, zoals Turingmachines.

De tabel toont per opleiding of zo'n vak aanwezig is en of het vak verplicht is. Ten slotte noteerden we van iedere opleiding, indien van toepassing, in de laatste kolom hoe het vak summatief wordt getoetst. Deze kolom bevat de codes 'S' (schriftelijk) en 'S+P' (schriftelijk en praktisch). Geen van de geanalyseerde opleidingen toetst het vak uitsluitend praktisch.

2.1.5 Positie en herkomst in het Nederlandse onderwijs

In opdracht van het ministerie van Onderwijs, Cultuur en Wetenschap is onderzoek gedaan naar de uitvoeringspraktijk van het Nederlandse keuzevak informatica (Tolboom e.a., 2014). De uitkomsten van dit onderzoek hebben het ministerie doen besluiten het vak te vernieuwen in de bovenbouw havo en vwo. Er is een vernieuwingscommissie informatica havo/vwo opgesteld, die in 2016 hun adviezen rapporteerde (Barendsen & Tolboom, 2016). Op basis van hun advies is in 2019 het vernieuwde examenprogramma in gebruik genomen („Examenprogramma informatica havo/vwo 2019”, 2019).

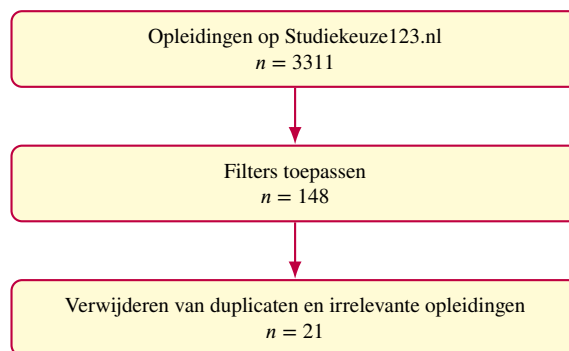
Het nieuwe examenprogramma bestaat uit een vast kernprogramma (domein A–F) en een aantal keuzethema's (domein G–R). Alle havo- en vwo-leerlingen moeten, ongeacht hun profiel, alle eindtermen in het kernprogramma beheersen. Havo-leerlingen moeten twee keuzethema's behandelen: één domein uit G–N en één domein uit O–R. Vwo-leerlingen moeten in totaal vier keuzethema's behandelen, waarvan minimaal één uit G–N en minimaal één uit O–R. Eindige automaten en grammatica's maken onderdeel uit van het kernprogramma.

Barendsen en Tolboom (2016) geeft tal van achtergrondinformatie over de totstandkoming en inhoud van het examenprogramma. Misschien wel de belangrijkste overweging is geweest om het programma te laten spreken over concepten, en niet over contexten. De commissie schrijft hierover dat de „conceptuele inhoud van het vak [...] meer waardevast [is] dan de in razend tempo veranderende contexten”.

Tolboom e.a. stellen in hun rapport dat het vernieuwde examenprogramma „meer diepgang” en „een vast kerndeel, basisonderwerpen die voor iedereen gelden” moet kunnen faciliteren. De commissie heeft vervolgens op basis van onder andere „nationale en internationale rapporten en referentiecurricula” een „reeks van concepten en leerdoelen” opgesteld (Barendsen & Tolboom, 2016). Hiermee kregen eindige automaten en grammatica's hun intrede in het Nederlandse informatica-onderwijs; deze onderwerpen kwamen niet voor in het oude examenprogramma („Examenprogramma informatica havo/vwo 2007”, 2007).

In het domein *Grondslagen* worden inderdaad slechts vier hoofdconcepten in het vakgebied omschreven. We hebben ons in dit onderzoek gericht op de laatste twee subdomeinen van dit domein: eindige automaten (B3) en grammatica's (B4). De eindtermen voor deze subdomeinen luiden als volgt:

- B3. De kandidaat kan eindige automaten gebruiken voor de karakterisering van bepaalde algoritmen.



Figuur 2.7: Selectie van opleidingen

Type	Instituut	Opleiding	Vak	Verplicht	Toetsvorm
hbo	Avans Hogeschool	Informatica	✗	–	–
hbo	Avans Hogeschool	Technische Informatica	✗	–	–
hbo	Fontys Hogeschool	Informatica	✗	–	–
hbo	Hogeschool Inholland	Informatica	✗	–	–
hbo	Hogeschool Inholland	Technische Informatica	✗	–	–
hbo	Hogeschool Leiden	Informatica	✗	–	–
hbo	Hogeschool Rotterdam	Informatica	✗	–	–
hbo	Hogeschool Rotterdam	Technische Informatica	✗	–	–
hbo	NHL Stenden	Informatica	✗	–	–
hbo	NHL Stenden	Technische Informatica	✗	–	–
hbo	Saxion	Technische Informatica	✗	–	–
wo	Maastricht University	Computer Science	✓	✗	S
wo	Radboud Universiteit	Computing Science	✓	✓	S
wo	Rijksuniversiteit Groningen	Informatica	✓	✓	S
wo	Technische Universiteit Delft	Technische Informatica	✓	✓	S
wo	Technische Universiteit Eindhoven	Technische Informatica	✓	✓	S
wo	Universiteit Leiden	Informatica	✓	✓	S
wo	Universiteit Twente	Technical Computer Science	✓	✓	S
wo	Universiteit Utrecht	Informatica	✓	✗	S+P
wo	Universiteit van Amsterdam	Informatica	✓	✗	S+P
wo	Vrije Universiteit Amsterdam	Informatica	✓	✓	S

Tabel 2.1: Resultaten van de systematische analyse van informatica-vervolgopleidingen

B4. De kandidaat kan grammatica's hanteren als hulpmiddel bij de beschrijving van talen.

De commissie geeft in hun verslag een toelichting over het domein *grondslagen*, waar deze onderwerpen deel van uitmaken. Bovendien lichten ze toe waarom ervoor is gekozen om eindige automaten en grammatica's op te nemen in het examenprogramma (Barendsen & Tolboom, 2016, p. 15):

In dit domein worden digitale artefacten op een abstracte manier bekeken, dat wil zeggen onafhankelijk van een concrete implementatie. Een groot voordeel van werken met algoritmen, bijvoorbeeld, is dat we kunnen redeneren over de efficiëntie van een oplossing nog vóór er een programma in een concrete programmeertaal is geschreven. Ook het redeneren over een bestaand programma wordt gemakkelijker als we abstraheren van de implementatiedetails en kijken naar het achterliggende algoritme.

[...]

De achterliggende gedachten voor het opnemen van automaten en grammatica's zijn als volgt. Met automaten (toestandsovergangssystemen) zijn artefacten vaak elegant en simpel te beschrijven of ontwerpen zonder direct in programmacode te vervallen. Verder geven grammatica's structuur (en daarmee eenvoud) in beschrijvingen van (programmeer)talen. Het is dus de bedoeling om deze twee modellen instrumenteel te gebruiken en niet om er allerlei diepe theorie mee te bedrijven. Eventueel kan een geïnteresseerde leerling wat dieper gaan via een keuzethema.

Voor informaticadocenten betekent dit dat zij deze onderwerpen niet zo theoretisch hoeven te behandelen als bijvoorbeeld in dit verslag is gedaan voor definities 1 en 10, maar dat de concepten praktisch en instrumenteel moeten worden ingezet om *artefacten* – „programma's, computersystemen, interfaces, enzovoort” (Barendsen & Tolboom, 2016) – te construeren en erover te redeneren.

Deze benadering is in het Nederlandse onderwijs niet nieuw. In een NLT-module over robotica uit 2009 gebruiken leerlingen in het elfde hoofdstuk eindige automaten om het gedrag van een robot te programmeren (van Lith e.a., 2009, p. 117–120).

De nuance uit het adviesrapport komt in de handreiking (SLO, 2020) helaas niet meer voor. Er wordt in dit document op summiere wijze toegelicht wat verschillende vaktermen als eindige automaat en productieregel betekenen. De uitleg over wat wel en niet de bedoeling is om te behandelen ontbreekt.

2.1.6 Hindernissen en mogelijkheden

Het is moeilijk om eindige automaten en grammatica's aan te leren in het onderwijs. Leerlingen zijn minder gemotiveerd voor en geïnteresseerd in deze onderwerpen. Dat komt niet alleen door de (waargenomen) complexiteit, maar ook omdat de onderwerpen als erg wiskundig worden gezien, en niet per se als onderdeel van informatica (Chesnevar e.a., 2004).

Chesnevar e.a. geven een aantal strategieën die het leren moeten bevorderen:

- Geef de onderwerpen historische context. De Turingmachine, is bijvoorbeeld door Alan Turing gebruikt als onderdeel van zijn werk in de Tweede Wereldoorlog. De film *The Imitation Game* (2014) vertelt over dit verhaal en kan worden gebruikt in de lessen over automaten.
- Theoretische computersimulators kunnen worden gebruikt in actieve werkvormen om leerlingen actief te laten leren over eindige automaten en grammatica's. Zulke simulators slaan een brug tussen theorie en praktijk, en geven verschillende perspectieven op de onderwerpen.
- De onderwerpen zijn sterk gerelateerd aan kennis over programmeertalen. Maak deze verbanden zichtbaar en gebruik ze als context in het behandelen van de onderwerpen.

- Laat leerlingen zelf onderzoek doen naar eindige automaten, grammatica's en de theoretische limieten van deze instrumenten. Ga met elkaar in gesprek over de bevindingen. Zo nemen de leerlingen zelf een positie aan ten opzichte van de onderwerpen.

De Souza e.a. (2015) gaan zelfs een stap verder: zij stellen voor om studenten zelf een theoretische computersimulator te construeren voor het onderwerp waar ze interesse in hebben, om zo een breed en diepgaand begrip over het onderwerp te krijgen.

Hazzan e.a. (2014) gaan niet specifiek in op de didactiek van eindige automaten, maar geven ze wel een overzicht van verschillende soorten vragen die docenten kunnen stellen om het begrip van eindige automaten te bevorderen.

Bezáková en Winczer (2011) probeerden theoretische informatica-onderwerpen, waaronder eindige automaten, aan docenten te onderwijzen. Ze gebruikten een theoretische aanvliegroute, waarbij participanten zelf automaten gingen construeren voor bijvoorbeeld een koffieautomaat. Ze concludeerden dat de beste manier om deze onderwerpen te doceren ergens tussen de extreem theoretische en formele kant enerzijds, en de extreem praktische kant anderzijds moet liggen. In het verlengde hiervan zouden docenten deze technieken ook kunnen inzetten om de concepten aan hun leerlingen uit te leggen. Ze kunnen hierbij differentiatie toepassen om te kiezen voor de theoretische of praktische aanpak.

We kunnen voorzichtig concluderen dat een puur theoretische aanpak voor het behandelen van eindige automaten en grammatica's weinig leerrendement zal hebben. Het wordt effectiever geacht om de praktijk op te zoeken, met actuele voorbeelden en sociale/historische contexten die passen bij de leefwereld van de leerling. Simulators kunnen bovendien helpen om de systemen en de werking ervan inzichtelijk te maken.

2.2 Invulling van informaticalessen

Bij de invulling van lessen draait het in eerste instantie om de inrichting van het programma van toetsing en afsluiting (PTA). Het PTA vormt de formele weerslag van de inrichting van het schoolexamen. De vaststelling en invulling van het PTA zijn wettelijk bepaald (Wet Voortgezet Onderwijs, 2020). De VO-raad heeft deze wet op een voor docenten en schoolleiders toegankelijke wijze samengevat:

Het bevoegd gezag stelt jaarlijks voor 1 oktober een programma van toetsing en afsluiting (PTA) vast voor het desbetreffende schooljaar. Het PTA vermeldt in ieder geval:

- a. welke examenstof van het examenprogramma in het schoolexamen wordt getoetst;
- b. welke door het bevoegd gezag vast te stellen examenstof in het schoolexamen wordt getoetst;
- c. de inhoud van de toetsen die onderdeel uitmaken van het schoolexamen;
- d. de wijze waarop en tijdvakken waarbinnen de toetsen en herkansingen van het schoolexamen plaatsvinden;
- e. de regels voor de wijze waarop het cijfer voor het schoolexamen voor een kandidaat tot stand komt.

(VO-raad, 2025)

Het PTA is dus in feite een lijst van toetsen waarmee het examenprogramma wordt afgesloten. Het PTA geeft slechts een afspiegeling van het schoolexamen, en niet van het gehele lesprogramma. Het lesprogramma wordt ingericht op basis van het PTA. Het beschrijft welke hoofdstukken op welk moment worden behandeld, wanneer en waarover toetsen worden afgenomen, hoe die eruit zien, en wat

er van docent(en) en leerlingen wordt verwacht.

Een toets staat nooit op zichzelf. Met behulp van een toets kan een docent vaststellen of leerdoelen zijn behaald. De leerdoelen zijn een uiting van de conceptuele eindtermen in het examenprogramma, in de contexten die de docent heeft gekozen. Ten slotte zet de docent leeractiviteiten in om ervoor te zorgen dat de leerling zijn/haar doelen zal behalen.

Een eindterm in het examenprogramma is bijvoorbeeld „de kandidaat kan een informatiebehoefte vertalen in een zoekopdracht op een verzameling gestructureerde data” („Examenprogramma informatica havo/vwo 2019”, 2019, Subdomein C5: Gestructureerde data). De docent kan er vervolgens voor kiezen om gebruik te maken van een MySQL-database als verzameling gestructureerde data. Daarna stelt hij hiervoor een leerdoel op, zoals „de leerling kan MySQL-query’s opstellen waarin meerdere tabellen worden gecombineerd met INNER JOIN”. Door middel van verschillende activerende werkvormen zorgt de docent er in de lessen voor dat de leerling zelf zulke query’s kan opstellen. Afsluitend wordt deze vaardigheid getoetst op een schriftelijke toets.

De chronologische aard van het PTA, maar in het bijzonder het lesprogramma, maakt bovendien dat ze iets vertellen over afhankelijkheden: sommige onderwerpen worden bewust ná andere onderwerpen behandeld, omdat de in het ene onderwerp vergaarde kennis en kunde nodig zijn om het andere onderwerp succesvol te kunnen behandelen.

Ten slotte vertelt de gelijktijdigheid van onderwerpen in het lesprogramma iets over *inbedding*: het behandelen van een bepaald onderwerp als onderdeel van een groter hoofdonderwerp. Denk hierbij bijvoorbeeld aan het inbedden van eindige automaten bij programmeren.

Voor dit onderzoek hebben we de invulling van informaticalessen over een bepaald onderwerp gedefinieerd als een combinatie van inhoud, omvang, toetsing, plaats in het curriculum en inbedding. In de volgende paragrafen lichten we verder toe wat we met elk van deze aspecten bedoelen, en welke rollen ze kunnen spelen in de inrichting van het onderwijs. We benadrukken dat dit *onze invulling* is, die we instrumentaal hebben gebruikt om de vragenlijsten en interviews georganiseerd te houden, en om gerichtere vragen te kunnen stellen.

2.2.1 Inhoud

Met de inhoud van de lessen bedoelen we de onderwerp(en) die aan bod komen, en de context waarin deze onderwerp(en) worden behandeld.

Het examenprogramma informatica beschrijft alleen concepten. Deze zijn bijzonder abstract van aard en moeten door de docenten gespecificeerd worden aan de hand van actuele contexten. Bij lessen over programmeren (domein D) moet de docent bijvoorbeeld kiezen welke programmeertaal er wordt behandeld, en welke onderdelen en functionaliteiten van deze taal worden geleerd.

Het gaat bij inhoud ook over de manier waarop een onderwerp wordt behandeld: welke leeractiviteiten vinden er op macroniveau plaats? Het gaat in dit onderzoek dan *niet* om specifieke werkvormen per les; dat zou te granulair en complex worden. We richten ons wel bijvoorbeeld op hulpmiddelen die worden gebruikt — tekenen de leerlingen een eindige automaat met pen en papier, of gebruiken ze er een programma voor?

Ten slotte gaat het bij inhoud om de toepassingen van de gekozen onderwerpen. Bij programmeren kan een docent er bijvoorbeeld voor kiezen dat leerlingen een game moeten programmeren. Dit gedeelte van de inhoud is nauw verwant aan de leerdoelen (die we onderbrengen onder het aspect toetsing). Dat is logisch: in de meeste onderwijsmodellen zijn leeractiviteiten en toetsing onlosmakelijk aan elkaar verbonden. Het onderscheid ligt voor ons in de specificiteit: een inhoudelijk onderwerp geeft aan wat leerlingen in het kader van een bepaald subdomein doen, en een leerdoel specificeert wat ze daarbij precies zullen leren.

2.2.2 Omvang

In essentie is de omvang van een lessenserie uit te drukken in het aantal studielasturen (SLU) dat gepland is voor de lessenserie. Een studielastuur is doorgaans één klokuur (zestig minuten) dat door de leerling — thuis of op school — actief wordt gespendeerd aan het onderwerp. Dit uur kan worden ingevuld door allerlei (leer)activiteiten: luisteren naar uitleg, werken aan een praktische opdracht en studeren voor of zelfs het maken van een toets.

In totaal is er voor informatica 320 uur (havo) en 440 uur (vwo) aan studielast ‘beschikbaar’. In tabel 2.2 is het aantal SLU dat de gangbare informaticamethoden zelf adviseren aan de onderwerpen in domein B uiteengezet.

Methode	B1 Algoritmen	B2 Datastructuren	B3 Automaten	B4 Grammatica's	Totaal
VO-Content/Enigma	<i>Onbekend</i>				
Fundament Informatica (minimale behandeling)	6	10	4–8	4–5	24–29
Fundament Informatica (advies methode)	12	10	8	8	38
Informatica-Actief (advies methode)	–	–	–	–	13–18

Tabel 2.2: Geadviseerde studielasturen voor grondslagen, volgens de gangbare informaticamethoden. Een bereik aan getallen geeft aan dat er differentiatie in het aantal SLU is voor havo- en vwo-leerlingen. We hebben VO-Content gevraagd naar de geadviseerde SLU's voor hun module over grondslagen, maar hebben nooit een inhoudelijke reactie ontvangen.

Er kunnen twee redenen zijn voor docenten om niet tevreden te zijn met het aantal SLU dat ze aan een onderwerp besteden. Ten eerste kan er wrijving ontstaan tussen de tijd die nodig is om de leerdoelen te behalen, en de tijd die daadwerkelijk beschikbaar is voor leeractiviteiten om die leerdoelen te behalen. Verder kunnen de eigen interesse van de docent en zijn/haar leerlingen een (grote) rol spelen. De docent kan bijvoorbeeld eigenlijk meer tijd willen besteden aan een onderwerp dat hem of haar interesseert, of juist liever minder tijd willen besteden aan een onderwerp dat hij of zij niet belangrijk vindt.

Ten slotte rekenen we onder omvang ook het type activiteit dat leerlingen verrichten. We richten ons dan op het al dan niet in groepsverband werken: wordt er individueel werk verricht, wordt er in tweetallen gewerkt, of zelfs in grotere groepen? We rekenen deze eigenschap onder omvang, omdat de verdeling individueel werk – groepswerk bepalend kan zijn voor het aantal uren dat wordt besteed aan een onderwerp.

2.2.3 Toetsing

Leerdoelen moeten altijd in verband gebracht worden met leeractiviteiten en toetsvormen. Een leerdoel opstellen is niet voldoende; het leerdoel moet worden bereikt met leeractiviteiten, en dat moet weer worden gecontroleerd met een toets. In dit onderdeel van de invulling van een les bekijken we wat er wordt getoetst, en hoe dat precies gaat.

Wat er wordt getoetst wordt uitgedrukt als een set leerdoelen. Dat zijn observeerbare vaardigheden die de leerling verkrijgt na het succesvol doorlopen van de les(senserie): iets kunnen uitleggen, iets kunnen toepassen, iets kunnen uitvoeren, et cetera. Voor dit onderzoek bekijken we niet alleen welke leerdoelen een leerling volgens een docent dient te beheersen, maar ook *waarom* de docent voor deze leerdoelen heeft gekozen. Docenten hebben namelijk behoorlijk veel vrijheid in het opstellen van

leerdoelen: alle leerlingen moeten dezelfde eindtermen beheersen, maar die zijn uit te drukken in uiteenlopende leerdoelen.

Het toetsen van de leerdoelen kan formatief en summatief plaatsvinden. Bij formatieve toetsing krijgen zowel de docent als de leerling tijdens het leren zicht op de behaalde en nog te behalen leerdoelen. Het eigenaarschap van het leerproces wordt hiermee bevorderd: leerlingen zijn zich bewuster van hun vorderingen en weten waar ze nog aan moeten werken. Bij summatieve toetsing controleert de docent achteraf of de leerdoelen behaald zijn, bijvoorbeeld met een schoolexamen.

In dit onderzoek richten we ons op de inrichting van de summatieve toetsen. Bij alle vakken, maar in het bijzonder bij informatica, hebben docenten de vrijheid om leerdoelen theoretisch, maar ook praktisch te toetsen. Leerlingen moeten dan bijvoorbeeld iets ontwikkelen: een computerprogramma of een werkstuk. Het is zelfs mogelijk om de leerdoelen te toetsen met een presentatie of mondelinge overhoring.

2.2.4 Plaats in het curriculum

Hoewel het examenprogramma informatica zeer uiteenlopende onderwerpen bevat, is er, afhankelijk van de gekozen contexten, veel sprake van afhankelijkheden tussen de onderwerpen. Het ene onderwerp moet dan behandeld zijn alvorens men kan beginnen aan het andere onderwerp, omdat de bij het ene onderwerp vergaarde kennis en kunde wordt gebruikt bij het leren van het andere onderwerp. Docenten kunnen er dan bewust voor kiezen om een bepaald onderwerp eerder te behandelen dan een ander onderwerp.

2.2.5 Inbedding

Bij onderlinge afhankelijkheden tussen onderwerpen kan de docent er ook voor kiezen om de onderwerpen gelijktijdig te behandelen. Leerlingen maken dan bijvoorbeeld een website in de (programmeer)talen HTML, CSS en PHP, en koppelen die website aan een MySQL-database. Ze leren gelijktijdig programmeren (eindterm D1) en omgaan met een database (eindterm C5).

3 Methode

In dit hoofdstuk leggen we uit met welke methoden we hebben geprobeerd de onderzoeksvragen te beantwoorden. Eerst gaan we in paragraaf 3.1 dieper in op de procedure die is toegepast om het onderzoek in grote lijnen vorm te geven. In dit onderzoek zijn vragenlijsten en interviews afgenomen. De respondenten van deze bevragingen worden beschreven in paragraaf 3.2. In paragraaf 3.3 lichten we toe hoe de vragenlijsten en interviews precies zijn opgezet, en hoe we de validiteit en betrouwbaarheid van dit onderzoek hebben gegarandeerd. Ten slotte leggen we in paragraaf 3.4 uit hoe we de resultaten hebben geanalyseerd.

3.1 Procedure

De opzet van dit onderzoek is schematisch weergegeven in fig. 3.1. In de rest van deze paragraaf lichten we de onderzoeksopzet en de ethische aspecten verder toe.

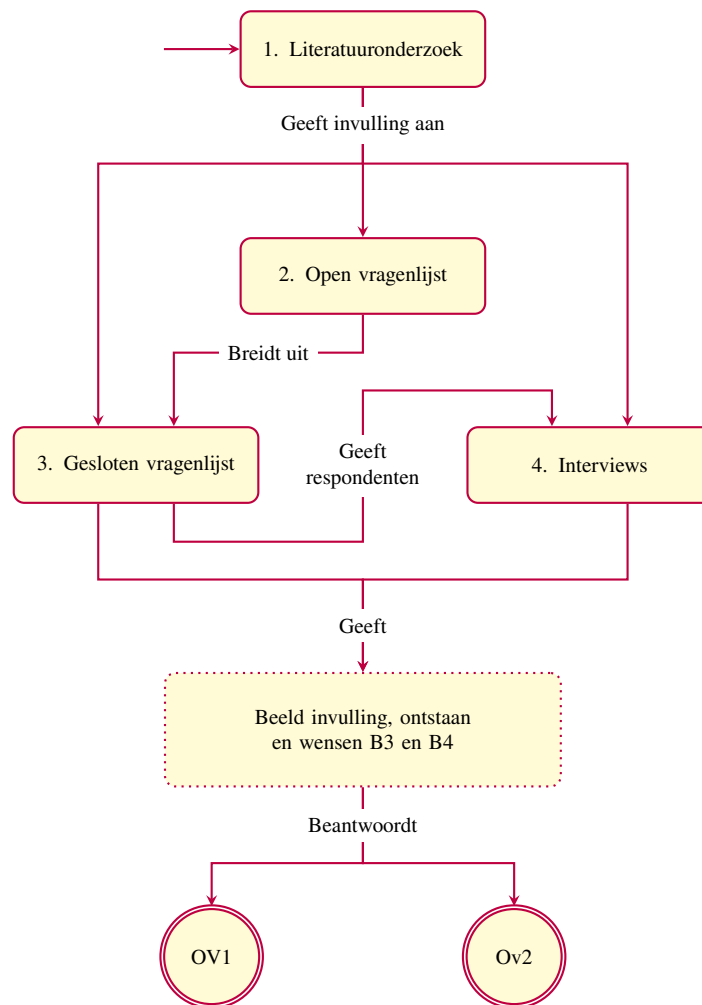
3.1.1 Onderzoeksopzet

Voor het beantwoorden van de eerste onderzoeksvraag moesten we een beeld krijgen van hoe informaticadocenten invulling geven aan de onderwerpen eindige automaten en grammatica's in hun curriculum. Als eerste stap in dit onderzoek hebben we een open digitale vragenlijst afgenomen onder zeven informaticadocenten uit het netwerk van de onderzoeker. De vragen in deze vragenlijst zijn gebaseerd op het literatuuronderzoek in hoofdstuk 2, maar ook op eigen intuïtie.

Zes docenten hebben aan deze vragenlijst deelgenomen. De respondenten gaven aan hoe hun curriculum is ingericht wat betreft subdomein B3 en B4, waarom het op die manier is ingericht, en wat ze verder nodig denken te hebben om hun lessen over deze onderwerpen verder te kunnen verbeteren. Deze eerste stap was nadrukkelijk vooronderzoek, en de gegevens die eruit voortvloeien zijn niet gebruikt om de onderzoeks(deel)vragen te beantwoorden.

Op basis van de gegeven antwoorden, literatuuronderzoek en eigen ervaring en inschatting, construeerden we vervolgens een gesloten digitale vragenlijst, die in feite hetzelfde bevroeg. Deze vragenlijst hebben we afgenomen onder zo veel mogelijk informaticadocenten. Het gesloten karakter moest het voor participanten eenvoudiger maken om deel te nemen aan het onderzoek. Ook zorgde het voor eenvoudigere analyse achteraf.

Respondenten konden op vrijwillige basis hun contactgegevens achterlaten. Zo konden ze zich beschikbaar stellen voor een semi-gestructureerd interview. Niet iedere respondent is vervolgens uitgenodigd voor een interview. We verwachtten meerdere antwoorden meermaals terug te zien in de gegeven antwoorden. Het zou weinig meerwaarde hebben geboden om meerdere docenten te vragen uitleg te geven over dezelfde lesinhoud. De bedoeling was dan ook om enkele respondenten te selecteren op basis van antwoorden die buiten de norm vielen. Het doel van de interviews was om de



Figuur 3.1: De opzet van dit onderzoek schematisch weergegeven.

gegeven lesinhouden beter te kunnen beschrijven en afbakenen, en om achter het ‘waarom’ van de lesinhouden te kunnen komen.

Na het afnemen van de vragenlijst namen we de antwoorden met contactgegevens door en hebben we drie respondenten uitgekozen, die we uitnodigden voor een verdiepend interview naar aanleiding van hun gegeven antwoorden. Alle contactgegevens werden, na het afnemen van de interviews, onmiddellijk vernietigd.

Ten slotte hebben we de resultaten uit de gesloten vragenlijst en de interviews verwerkt, zodat ze gebruikt konden worden om de onderzoeksvragen te beantwoorden.

3.1.2 Ethische overwegingen

Deelname aan alle onderzoeken was geheel vrijwillig. Antwoorden op de vragenlijst en de interviews zijn geanonimiseerd. De koppeling tussen contactgegevens en antwoorden in de vragenlijst is vernietigd na het afnemen van de interviews. Het is dus niet meer mogelijk om individuele antwoorden of uitingen te koppelen aan een persoon.

Informaticadocenten kunnen profiteren van de uitkomsten van het onderzoek. Idealiter ontdekten we nieuwe manieren om subdomein B3 en B4 te onderwijzen. Docenten kunnen die nieuwe manieren toepassen in hun eigen lessen.

Voor de vragenlijsten en interviews hebben we gebruik gemaakt van formulieren voor *informed consent*. Hierop legden we uit wat het onderzoek inhoudt, wie de onderzoekers zijn en welke rechten de respondenten hebben. Wanneer we oproepen plaatsten voor deelname aan de grote, gesloten vragenlijst, gaven we deze informatie ook.

Het onderzoek is beoordeeld en goedgekeurd door de ethische commissie van de faculteit BMS (domain Humanities & Social Sciences) onder aanvraagnummer 250110.

3.1.3 Chronologische leeswijzer

In dit verslag zijn de onderzoeksactiviteiten niet in chronologische volgorde beschreven. We adviseren de lezer die op chronologische volgorde wil lezen hoe de vragenlijsten en interviews zijn geconstrueerd en afgenomen om de volgende volgorde aan te houden:

1. Paragraaf 3.3.1: Constructie van de open vragenlijst;
2. Paragraaf 3.4.1: Geplande analyse van de open vragenlijst;
3. Paragraaf 3.3.2: Constructie van de gesloten vragenlijst voorafgaand aan de afname van de open vragenlijst;
4. Paragraaf 4.1: Bijwerken van de gesloten vragenlijst aan de hand van de resultaten van de open vragenlijst;
5. Paragraaf 3.4.2: Geplande analyse van de gesloten vragenlijst;
6. Paragraaf 4.2: Resultaten van de gesloten vragenlijst;
7. Paragraaf 3.3.3: Constructie van de interviews;
8. Paragraaf 3.4.3: Geplande analyse van de interviews;
9. Paragraaf 4.3: Resultaten van de interviews;
10. Paragraaf 5.1: Conclusies op basis van de gesloten vragenlijst en de interviews.

3.2 Respondenten

We wilden zo veel mogelijk informaticadocenten betrekken bij dit onderzoek. We definieerden de informaticadocent hier als iemand die op het moment van deelname informatica gaf op een of meerdere middelbare scholen in Nederland. Of hij of zij daartoe bevoegd was, speelde voor ons geen rol. We hebben deze keuze gemaakt naar aanleiding van de huidige arbeidsmarkt — een relatief groot deel van de posities voor informaticadocenten wordt in 2023 vervuld door anders- of onbevoegde docenten (den Uijl e.a., 2023). Om een correct beeld te schetsen van de invulling van het vak, moeten ook die docenten kunnen meedoen aan het onderzoek.

Hoeveel informaticadocenten zijn er eigenlijk?

De Dienst Uitvoering Onderwijs houdt sinds 2015 het aantal informaticadocenten in Nederland bij. Dat aantal schommelt sindsdien rond de 750 tot 800. In 2023 waren er 767 informaticadocenten werkzaam (Dienst Uitvoering Onderwijs, 2025).

3.2.1 Verkennende open vragenlijst

Voor de open vragenlijst hebben we ons berust op het netwerk van de onderzoeker. In totaal zijn zeven informaticadocenten uitgenodigd om deel te nemen aan het onderzoek. De vragenlijst is zes keer ingevuld. Het was in theorie mogelijk de vragenlijst meerdere keren in te vullen, maar de gegeven antwoorden en de schrijfstijl ervan liggen dusdanig ver uit elkaar, dat we niet verwachten dat hier sprake van is geweest. Overigens was dit ook niet belangrijk geweest voor dit deel van het onderzoek. Doel van de open vragenlijst was de constructie van een brede en inclusieve gesloten vragenlijst, en niet om een representatief beeld te schetsen van de curricula van informaticadocenten.

De vragenlijst was behoorlijk lang: 10 tot maximaal 48 vragen, afhankelijk van of de respondent aan beide subdomeinen invulling geeft in zijn/haar curriculum. We zijn erg blij met het aantal respondenten dat wilde meewerken aan dit deel van het onderzoek.

3.2.2 Gesloten vragenlijst

Om de betrouwbaarheid van de gegevens te bevorderen, hebben we voor de gesloten vragenlijst zo veel mogelijk docenten, met zo veel mogelijk uiteenlopende achtergronden, geprobeerd te bereiken. Bovendien hebben we de vragen meermaals getest: eerst als open vraag in de open vragenlijst, maar ook door de gesloten vragenlijst gespreid af te nemen, zodat fouten en onduidelijkheden snel konden worden opgelost. Verder hebben we boven elke sectie van de vragenlijst alle begrippen toegelicht en gedefinieerd, om te voorkomen dat respondenten de begrippen anders zouden interpreteren. Ten slotte hebben we zo veel mogelijk vragen verplicht gesteld in de online omgeving.

De gesloten vragenlijst is op 4 april 2025 opengezet voor respondenten. Vervolgens hebben we gespreid uitnodigingen verstuurd en laten versturen:

- Op vrijdag 4 april stuurden we een uitnodiging naar dezelfde zeven docenten in het netwerk van de onderzoeker die waren uitgenodigd om de open vragenlijst in te vullen.
- Op diezelfde dag namen we contact op met het bestuur van i&i. Die hebben op 15 april een nieuwsbrief gestuurd, met daarin een uitnodiging om de vragenlijst in te vullen (fig. 3.3).
- Ook plaatsten we een oproep om deel te nemen aan het onderzoek op het forum van i&i.
- Op dinsdag 8 april publiceerden we een bericht¹ op de blog van Fundament Informatica waarin

¹<https://www.3iblog.nl/Wil-je-mij-helpen-met-mijn-onderzoek-347>

lezers werden opgeroepen te helpen met het onderzoek.

- Op woensdag 9 april vroegen we Frans Peeters van informaticavo.nl om over de vragenlijst te schrijven op zijn website. De volgende dag werd een bericht² op de website gepubliceerd (fig. 3.4). Met Frans is afgesproken om het bericht ook op te nemen in de nieuwsbrief van de website, maar die nieuwsbrief is helaas niet meer tijdens de openstelling van de vragenlijst verschenen.
- Op woensdag 16 april vroegen we het bestuur van Informatica-Actief om het onderzoek onder haar gebruikers te verspreiden. Op woensdag 14 mei is dat gebeurd.
- Op woensdag 16 april verspreidden we ook de oproep om deel te nemen aan het onderzoek in een WhatsApp-groep met informaticadocenten in opleiding van de Universiteit Twente.
- Op woensdag 21 mei hebben we het onderzoek kort gepresenteerd bij de docentennetwerkbijeenkomst in Nijmegen. Afsluitend werden de bezoekers gevraagd de enquête in te vullen.

De impact van deze uitnodigingen op het aantal respondenten is grafisch weergegeven in fig. 3.2.

Op maandag 26 mei is de vragenlijst dichtgezet. Nieuwe respondenten konden vanaf dat moment geen bijdrage meer leveren aan het onderzoek. In totaal hebben 19 respondenten de gesloten vragenlijst ingevuld. Dat is significant minder dan de ruwe inschatting van in totaal ongeveer 800 potentiële respondenten. We vermoeden dat de lengte van de vragenlijst hier een rol in heeft gespeeld. Helaas hebben we geen inzicht in het aantal respondenten dat is begonnen met invullen, maar halverwege is opgehouden. Daarmee blijft dit vermoeden een vermoeden.

Het lage aantal respondenten heeft de representativiteit van het onderzoek aangetast. Wel hebben de respondenten uiteenlopende achtergronden: ze maken gebruik van verschillende (combinaties van) lesmethoden en geven aan uiteenlopende combinaties van leerjaren les. We vermoeden (maar kunnen niet bevestigen) dat dit mede te danken is aan de grote hoeveelheid kanalen die is gebruikt om docenten uit te nodigen deel te nemen aan het onderzoek.

3.2.3 Interviews

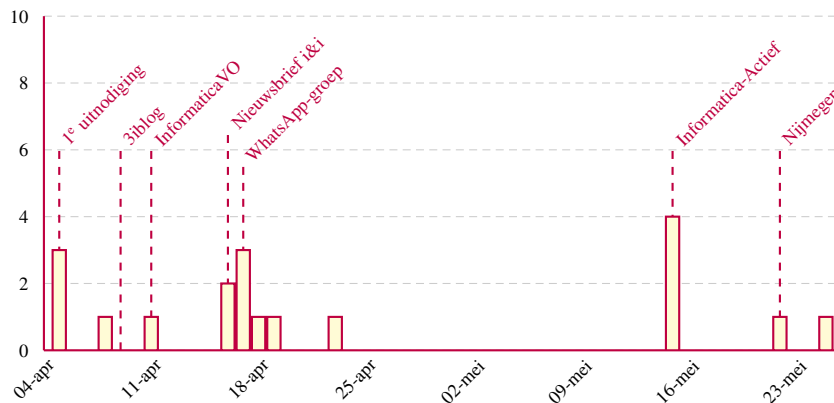
Van de 19 respondenten vulden 7 hun contactgegevens in. Van deze respondenten hebben we systematisch gezocht naar respondenten die we wilden uitnodigen voor een interview. Eerst markeerden we gegeven antwoorden die buiten de norm vielen, zoals een bovengemiddeld hoog of laag aantal SLU's voor een vak, of een toetsvorm die door de overige respondenten niet wordt ingezet. Ook gingen we op zoek naar antwoorden die opvielen doordat ze in een 'Anders'-vak werden ingevuld. Deze selectieprocedure heeft geleid tot het uitnodigen van drie respondenten voor een verdiepend interview. Twee hiervan zijn ingegaan op de uitnodiging. Van de derde hebben we helaas nooit een reactie ontvangen.

3.3 Instrumenten

Beantwoording van de eerste drie deelvragen geschiedde in de eerste plaats door de afname van een landelijke gesloten vragenlijst onder informaticadocenten. Deze vragenlijst diende om een globale indruk te krijgen van de curricula informatica in het Nederlands voortgezet onderwijs.

Deze vragenlijst bestond uit voornamelijk gesloten vragen, die direct slaan op de eerste onderzoeksvraag. Van groot belang was de representativiteit van het onderzoek — welke docenten nemen deel? We verwachtten een sterke correlatie te zien tussen de lesmethode (het 'boek') die een docent gebruikt en de contexten die hij of zij gebruikt in het curriculum. De boeken schrijven immers zelf hun eigen

²<https://www.informaticavo.nl/nieuws/7122/vragenlijst-voor-afstudeerder-eerstegraads-docent-informatica>



Figuur 3.2: Aantal respondenten per dag. Een verticale stippellijn is een moment waarop we uitnodigingen verstuurd.

Onderzoek automaten en grammatica's

In het kader van de opleiding tot eerstegraads docent informatica doet Wouter van den Brink onderzoek naar de manier waarop de subdomeinen automaten en grammatica's binnen het huidige examenprogramma worden ingevuld.

Het onderzoek richt zich op de volgende vragen:

Hoe krijgt dit subdomein momenteel vorm in de lespraktijk?

Welke overwegingen spelen een rol bij de gekozen invulling?

Wat is er nodig om het onderwijs over dit onderwerp verder te versterken?

Informatica-docenten die willen bijdragen aan dit onderzoek of hun ervaringen willen delen, worden van harte uitgenodigd om te reageren.

Vul de vragenlijst in

Figuur 3.3: Uitnodiging om de gesloten vragenlijst in te vullen in de nieuwsbrief van i&i.

Vragenlijst voor afstudeerder eerstegraads docent informatica

10 april 2025

Voor zijn masteropleiding tot eerstegraads docent informatica doet Wouter van den Brink onderzoek naar de invulling van subdomein B3 en B4 in het informatica-onderwijs. Hiervoor zou hij graag meer te weten willen komen over de manier waarop je in je curriculum invulling geeft aan deze subdomeinen. Ook wil hij graag weten hoe deze invulling tot stand is gekomen, en wat je nog mist om je lessen over deze onderwerpen nog beter te kunnen vormgeven.

Ga naar zijn [website](#) voor meer informatie en voor de vragenlijst.

Figuur 3.4: Uitnodiging om de gesloten vragenlijst in te vullen op informaticavo.nl.

contextuele keuzes voor, en niet iedere docent zal de tijd of motivatie hebben om eigen contextuele keuzes te maken voor ieder concept. Daarom diende de enquête ook een profielschets van de docent te maken: welke methode(n) gebruikt hij/zij om eindige automaten en grammatica's te onderwijzen?

In paragraaf 2.2 hebben we een uitgebreide definitie opgesteld voor het onderwerp van dit onderzoek. Toch verwachtten we op basis van alleen ons literaire vooronderzoek en eigen voorkennis geen dek-kende gesloten vragenlijst te kunnen construeren waarin alle lesstijlen van alle informaticadocenten worden vertegenwoordigd. Daarom hebben we eerst een verkennende open vragenlijst geconstrueerd, die in de volgende paragraaf wordt beschreven. Deze vragenlijst bevatte dezelfde vragen die we wilden stellen in de gesloten vragenlijst, maar respondenten konden de antwoorden vrij opgeven. De gegeven antwoorden zijn vervolgens gebruikt om de vragen en mogelijke antwoorden in de gesloten vragenlijst verder te verfijnen, hetgeen beschreven wordt in paragraaf 3.4.1.

3.3.1 Verkennende open vragenlijst

Voorafgaand aan de constructie van de gesloten vragenlijst zijn zeven informaticadocenten in het netwerk van de onderzoeker uitgenodigd om deel te nemen aan een open vragenlijst, waarin zij zijn gevraagd uit te leggen hoe hun lessen over eindige automaten en grammatica's zijn opgezet, waarom dat op die manier is gebeurd, en welke wensen zij hebben voor de toekomst.

Voor de constructie van deze vragenlijst hebben we gebruikgemaakt van het stappenplan uit van der Donk en van Lanen, 2024, § 6.4.3 en van Lazar e.a., 2017. De uiteindelijke vragenlijst is opgenomen in appendix A.

Hoewel deze vragen antwoord geven op de eerste drie deelvragen, is de groep respondenten niet representatief en hebben we de antwoorden niet meegenomen in de conclusies van dit onderzoek. We hebben de antwoorden gebruikt voor de constructie van een gesloten vragenlijst, die verder wordt toegelicht in de volgende paragraaf.

3.3.2 Gesloten vragenlijst

De gesloten vragenlijst is gebaseerd op de open vragenlijst en werd aangepast naar aanleiding van de resultaten uit de open vragenlijst (zie ook paragraaf 3.4.1). In beide lijsten werden per onderwerp en per aspect (inhoud, omvang, toetsing, plaats in het curriculum en inbedding) vragen gesteld over de invulling in de informaticalessen van de respondent. De uiteindelijke vragenlijst is opgenomen in appendix B.

Voor de waarborging van de validiteit hebben we tijdens de constructie telkens twee kanten op gewerkt: van onderzoeksvraag naar instrument, maar ook van gewenste resultaten naar instrument. We vroegen ons, op basis van de onderzoeksvragen, actief af waar we conclusies over wilden kunnen trekken, en welke gegevens we daarvoor nodig hadden. Op basis daarvan konden we beredeneren welke vragen we moesten stellen, en welke vragen eigenlijk niet zouden bijdragen aan het beantwoorden van de onderzoeksvragen. Ten slotte hebben we de vragenlijsten laten beoordelen door de onderzoeksbegeleiders.

3.3.3 Interviews met vo-docenten

Voor de constructie van de interviewvragen hebben we gebruikgemaakt van het stappenplan uit van der Donk en van Lanen, 2024, § 6.4.2. De uiteindelijke interviewvragen zijn opgenomen in appendix C.

Bij het verwerken van de antwoorden op de open vragenlijst en het samenstellen van de gesloten vragenlijst merkten we dat een vragenlijst, en in het bijzonder een gesloten vragenlijst, zich niet goed leent voor het vaststellen van het 'waarom' achter de contextuele keuzes. Daarom hebben we gepoogd dit onderdeel van de eerste onderzoeksvraag voornamelijk te beantwoorden met de interviews.

Het interview bestond uit drie grote onderdelen. We begonnen met een introductie, waarin we vroegen of de docent de onderwerpen behandelde in zijn/haar lessen. Indien dat niet het geval was, konden we vragen waarom dat zo was, en wat er moest veranderen alvorens het onderwerp door de docent zou worden behandeld.

Daarna gingen we per onderwerp de diepte in: wat doen de leerlingen van de respondent precies voor deze onderwerpen, en waarom zijn die keuzes gemaakt? En welke veranderingen zou de respondent graag willen doorvoeren om de lessen verder te verbeteren?

De interviews vonden online plaats. Voor de betrouwbaarheid was het beter om de interviews in een gecontroleerde ruimte af te nemen, maar de afstand tussen de onderzoeker en de respondenten maakte dat geen praktische oplossing.

Zoals ook bij de vragenlijsten, hebben we bij de constructie van de interviewvragen twee kanten op gewerkt: van onderzoeksvraag naar instrument, maar ook van conclusies — waar willen we uitspraken over doen? — naar instrument. Verder hebben we de vragen laten beoordelen door de onderzoeksbegeleiders, alvorens we aan de interviews begonnen.

3.4 Analyse

3.4.1 Van open naar gesloten vragenlijst

Het doel van de open vragenlijst was het verfijnen van de gesloten vragenlijst. We wilden voorkomen dat een respondent van de gesloten vragenlijst zijn/haar verhaal niet goed kwijt kon door een gebrek aan keuzemogelijkheden in de antwoorden. In de eerste plaats moesten we dus kijken naar gesloten vragen in de open vragenlijst, waar respondenten gebruik moesten maken van het ‘Anders’-veld. Duidelijk ontbrekende antwoorden die waarschijnlijk vaker ingevuld zouden worden door andere docenten, werden toegevoegd als mogelijke antwoorden aan de gesloten vragenlijst.

Andere ‘anders’-antwoorden ontstaan wanneer de respondent de vraag niet interpreteerde zoals wij hem hadden bedoeld. Zulke antwoorden werden niet opgenomen in de gesloten vragenlijst, maar konden wel aanleiding geven om de vraagstelling te verbeteren.

Verder keken we op een hoger niveau naar de verworven resultaten. Gaven de antwoorden op de open vragen wel antwoord op de onderzoeksvragen? Zo niet, dan moest er iets verkeerd zijn met de vraag: de vraag werd niet duidelijk genoeg gesteld, of had eigenlijk (te) weinig te maken met de onderzoeksvragen. De vervolgactie was dan om de vraag anders te stellen, of te laten vallen.

Een laatste mogelijkheid was dat de antwoorden blootlegden dat de vraag niet op gesloten wijze kon worden gesteld. Dan werd de vraag verplaatst naar de interviews.

Ten slotte verwachtten we dat de gegeven antwoorden andere, nieuwe inzichten konden geven, die niet van tevoren konden worden voorspeld in deze beschrijving van de analyse vooraf. Deze inzichten komen tot uiting in paragraaf 4.1.

3.4.2 Gesloten vragenlijst

Voor alle gesloten vragen hebben we een staafdiagram geproduceerd van de gegeven antwoorden. Een aantal vragen leverden kwantificeerbare antwoorden op. Voor deze vragen hebben we de descriptieve statistieken berekend en in tabellen met behulp van boxplots weergegeven.

Voor twee combinaties van vragen hebben we bovendien de verbanden tussen de gegeven antwoorden weergegeven, te weten:

- Het aantal SLU dat besteed wordt aan een bepaald onderwerp, en de (on)tevredenheid over dit aantal SLU wat betreft de omvang van de leerdoelen en het enthousiasme/de passie van de respondent;
- De subdomeinen die bewust vóór, tegelijk met en bewust ná een bepaald onderwerp worden behandeld.

Ten slotte hebben we, ter voorbereiding op de concluderende fase van het onderzoek, een tekstuele samenvatting van de resultaten opgesteld.

3.4.3 Interviews met docenten

Van ieder interview is een transcriptie gemaakt op basis van een audio-opname. De opnames zijn, ter bescherming van de privacy van de respondenten, na transcriptie vernietigd.

Vervolgens is ieder interview systematisch geanalyseerd aan de hand van de eerste drie deelvragen. Per deelvraag is er voor beide onderwerpen beschreven hoe de docent in het interview antwoord gaf. Deze gestructureerde samenvattingen vormen, samen met de resultaten van de gesloten vragenlijst, de basis voor de beantwoording van de eerste drie deelvragen van dit onderzoek.

De volledige transcripties zijn opgenomen in appendix D. Hierbij zijn onderdelen van de transcriptie die mogelijk traceerbaar waren naar respondenten, of de school waar zij werkzaam waren, weggelaten.

4 Resultaten

4.1 Open vragenlijst

In deze paragraaf lichten we toe hoe de antwoorden op de open vragenlijst zijn verwerkt om de gesloten vragenlijst te construeren.

4.1.1 Informatie over de respondenten

Alle respondenten hebben aangegeven les te geven in alle leerjaren. Ook gebruikt iedere respondent Fundament Informatica. Dat is wellicht te wijten aan de aard van de respondenten — allemaal docenten in het netwerk van de onderzoeker, werkzaam bij de uitgeverij achter Fundament — maar niet geheel, omdat we ook medestudenten en vakdidactici hebben gevraagd om deel te nemen. Het benadrukt wel dat deze gegevens niet representatief zijn voor het beantwoorden van de onderzoeksvragen.

Alle respondenten gebruiken een combinatie van lesmethoden voor hun informaticalessen. Deze combinaties zijn erg complex en we kunnen ze niet reduceren naar een gesloten set mogelijkheden. Om toch een goed beeld te schetsen van het verband tussen de lesmethode en de toegepaste contexten, hebben we ervoor gekozen om in de gesloten vragenlijst per onderwerp (eindige automaten en grammatica's) te vragen welke lesmethode(n) de respondent gebruikt voor dat onderwerp. Om een goed beeld te kunnen vormen van de respondenten, hebben we ervoor gekozen om de algemene vraag over gebruikte lesmethoden te behouden.

4.1.2 Invulling per onderwerp

Helaas gaf de helft van de respondenten aan het onderwerp grammatica's in het geheel niet te behandelen. Dat heeft ervoor gezorgd dat we de gesloten vragen moeilijk konden uitbreiden.

Inhoud

Voor de onderwerpen (contexten) die docenten gebruiken voor eindige automaten en grammatica's zijn zeer uiteenlopende antwoorden gegeven. Bovendien leek een aantal respondenten de vraag niet goed te begrijpen. Voor de constructie van de gesloten vragenlijst maken we hier een gesloten vraag van, met een 'Anders'-veld. De gesuggereerde mogelijke antwoorden baseerden we op het materiaal over de onderwerpen zoals die door de gangbare lesmethoden worden gegeven. Zo konden we duidelijker maken wat de bedoeling van deze vraag was, en hoopten we de meeste respondenten te kunnen bedienen met de gesuggereerde antwoorden.

De gereedschappen die docenten en hun leerlingen gebruiken om eindige automaten en grammatica's te noteren zijn grofweg onder te verdelen in fysieke gereedschappen (pen/potlood en papier) en een aantal bekende softwarepakketten. Deze vraag hebben we geheel gesloten gehouden.

Het ‘waarom’ achter een onderwerp is moeilijk te duiden. We hebben ervoor gekozen dit soort vragen te laten vallen, en in plaats daarvan terug te laten komen in de interviews.

Omvang

Alle docenten gaven een zo goed als numeriek antwoord op de vragen over SLU. In de gesloten vragenlijst is dit dan ook een vraag geworden waar alleen een getal kan worden ingevuld. De verklaringen voor het aantal SLU lieten eerder een (on)tevredenheid over het aantal SLU zien, en deze informatie is eigenlijk waardevoller voor het onderzoek. In de gesloten vragenlijst werd dit een meerkeuzevraag: wat vind je van deze hoeveelheid SLU? Zijn het er eigenlijk te weinig, genoeg, of te veel? Bovendien splitsten we deze vraag uit in twee aspecten: tevredenheid ten opzichte van het aantal leerdoelen dat moet worden behaald, en tevredenheid ten opzichte van de eigen interesse in het onderwerp.

Leerlingen van de respondenten verrichten voornamelijk (combinaties van) individueel werk en/of groepswork in de lessen over eindige automaten en grammatica's. Eén respondent gaf aan de werkvorm ‘specialisten’ in te zetten, maar uiteindelijk komt deze werkvorm ook neer op groepswork. In de gesloten vragenlijst is hier een multiple-select-vraag van gemaakt, waarin respondenten konden aangeven of hun leerlingen individueel en/of groepswork verrichten in de lessen over subdomein B3 en B4.

Toetsing

Respondenten gaven zeer uiteenlopende leerdoelen voor zowel eindige automaten als grammatica's. We merkten dat het open karakter van de vragenlijst zorgde voor een hoog niveau van specificiteit in de leerdoelen. We hebben de gegeven leerdoelen gecombineerd met die van de gangbare lesmethoden. Deze leerdoelen hebben we vervolgens minder specifiek gemaakt. Dat leverde een breed scala aan mogelijke leerdoelen op. De vraag is niet voorzien van een ‘Anders’-veld, maar van een laatste keuzemogelijkheid: „een ander leerdoel”. We hebben hiervoor gekozen, omdat een respondent met andere leerdoelen waarschijnlijk interessant is om uit te nodigen voor een verdiepend interview.

Plaats in het curriculum

De ‘waarom’-vragen zijn komen te vervallen en worden beantwoord in de interviews. In de gesloten vragenlijst vroegen we niet meer welke onderwerpen er bewust voor/na eindige automaten en grammatica's worden behandeld, maar welke subdomeinen, zodat de vragen gesloten bleven.

Inbedding

Een aantal respondenten gaf aan gebruik te maken van inbedding voor eindige automaten en grammatica's. Ook voor dit deel van de vragenlijst hebben we de vraag veranderd in een multiple-select-vraag, met alle subdomeinen in het examenprogramma als mogelijke antwoorden.

4.1.3 Wensen voor de toekomst

Voor beide onderwerpen gaven docenten verschillende concrete verbetermogelijkheden aan. Deze verbetermogelijkheden hebben we gereduceerd tot een aantal concrete categorieën, plus een extra optie: het schrappen van het onderwerp uit het examenprogramma.

De vraag over de inzet van onderwijsondersteunende partijen is veranderd in een meerkeuzevraag, waarin we vroegen welke partij(en) volgens de respondent een bijdrage zouden kunnen leveren aan de verbetering van de lessen.

4.2 Gesloten vragenlijst

De antwoorden op de gesloten vragenlijst zijn opgenomen in appendix B. In deze bijlage is per vraag, en in enkele gevallen voor een combinatie van vragen, met grafieken en/of tabellen aangegeven welke antwoorden erop zijn gegeven. In deze paragraaf geven we een tekstuele samenvatting van de resultaten.

4.2.1 Algemeen

In totaal hebben 19 respondenten de vragenlijst ingevuld. Alle respondenten gaven aan eindige automaten te behandelen in hun informaticacurriculum, maar slechts 13 docenten behandelen ook grammatica's.

10 van de 19 respondenten gaven aan (onder andere) gebruik te maken van eigen lesmateriaal, ook voor de onderwerpen van dit onderzoek. 14 van de 19 respondenten maken gebruik van een combinatie van eigen lesmateriaal en materiaal van een lesmethode van een educatieve uitgeverij.

Eén respondent gaf aan gebruik te maken van een experimentele methode genaamd 'De Taalkist'. De exacte uitgavedatum van dit boek is onbekend, maar aan de hand van een NIOC-bijdrage (Lepeltak, 1990) en een artikel in *Onze Taal* (Groen & Viskil, 1994) kunnen we aannemen dat het ergens tussen 1990 en 1994 is gepubliceerd.

4.2.2 Eindige automaten

De respondenten maken vooral gebruik van eigen lesmateriaal voor hun lessen over eindige automaten. Leerlingen gebruiken eindige automaten vervolgens voor uiteenlopende activiteiten: (toekomstige) systemen beschrijven, automaten analyseren en algoritmes beschrijven. Eén respondent gaf bovendien aan eindige automaten te gebruiken om de noodzaak van grammatica's voor het beschrijven van natuurlijke talen te demonstreren.

Voor het noteren van eindige automaten gebruiken de respondenten voornamelijk klassieke instrumenten: pen en papier (13 respondenten), programma's om flowcharts mee te maken (10) en reguliere kantoorsoftware (3). Een enkeling maakt gebruik van specifieke programmeertools, zoals JFLAP, Graphviz, TikZ of Python met Matplotlib (3). Ten slotte geven 4 respondenten aan andere digitale tools te gebruiken.

Onder de respondenten worden er gemiddeld 6 studielasturen (SLU) aan dit onderwerp besteed, met uitschieters richting de 24 – hoewel het overgrote deel van de respondenten onder de 9 SLU blijft. De meeste respondenten (17 van de 19) geven aan voldoende tijd te hebben om de leerdoelen te behandelen. Kijkend naar de eigen interesse in het onderwerp, geven 14 van de 19 respondenten aan genoeg tijd te besteden aan eindige automaten. Twee respondenten zouden vanuit deze overweging zelfs meer tijd willen besteden. Drie respondenten geven daarentegen in dit aspect aan dat de 1 SLU die ze besteden aan eindige automaten te veel is.

Leerlingen van 15 van de respondenten zijn vooral bezig met individueel werk wanneer ze bezig zijn met eindige automaten. Groepswerk in tweetallen komt minder vaak voor (10 respondenten), en er zijn slechts twee respondenten waar de leerlingen in grotere groepen aan eindige automaten werken.

Vrijwel alle (16 van de 19) respondenten willen dat hun leerlingen in ieder geval een nieuwe eindige automaat kunnen construeren. Van vergelijkbare populariteit zijn leerdoelen die ingaan op de theoretische kant: bepalen of een automaat input accepteert (13), bepalen welke toestand de automaat zich in zal bevinden (11) en de formele eigenschappen kennen en beschrijven (10). De kennis toepassen in

bijvoorbeeld een reguliere expressie of programmeercode is minder populair; slechts 6 respectievelijk 5 respondenten stellen deze leerdoelen op voor hun leerlingen. Evenmin komt optimaliseren en/of vereenvoudigen van een bestaande automaat veel voor in de lessen: slechts twee docenten noemen dit als leerdoel.

12 respondenten geven aan deze leerdoelen bij uitstek belangrijk te vinden, omdat zij deze kennis en kunde vinden horen bij het vak informatica. Slechts enkele respondenten (3) noemen de noodzaak tot toepassing van deze leerdoelen in andere vakken, andere informaticalessen of vervolgoopleidingen als reden om de leerdoelen op te stellen.

Alle respondenten toetsen de leerdoelen summatief. Zes respondenten combineren een praktische en een theoretische toets; de overige kiezen voor slechts één van beide toetsvormen: 9 respondenten passen alleen een schriftelijke toets toe, en 4 respondenten toetsen het onderwerp alleen op praktische wijze.

Eindige automaten worden door de meeste respondenten in de eerdere leerjaren gegeven. Meestal worden de lessen bewust tegelijk met of ná de lessen over andere onderwerpen in het kernprogramma behandeld.

De lessen zijn voor een aantal (9) respondenten een bewuste voorbereiding op subdomeinen uit zowel het kernprogramma — in het bijzonder subdomein B4, over grammatica's — als de keuzethema's.

4.2.3 Grammatica's

Bestaande lesmethoden worden voor dit onderwerp vaker dan eigen lesmateriaal ingezet door de respondenten. De meeste leerlingen zullen gebruikmaken van grammatica's om bestaande talen te analyseren en vergelijken, voornamelijk aan de hand van de BNF-notatie. Twee respondenten geven aan dat hun leerlingen een nieuwe programmeertaal ontwerpen. Zes andere respondenten geven aan een andere notatie te gebruiken: railroad- of syntaxdiagrammen (vier respondenten) en/of iets anders (twee respondenten).

Ook voor het noteren van grammatica's gebruiken de respondenten voornamelijk klassieke instrumenten: pen en papier (7 respondenten), programma's om flowcharts mee te maken (5) en reguliere kantoorsoftware (3). Maar ook andere digitale tools (5) en een enkele keer zelfs een fysieke tool worden door de respondenten ingezet om grammatica's te tekenen.

De respondenten besteden gemiddeld 3 SLU aan grammatica's, maar enkele respondenten besteden meer tijd – tot wel 15 SLU. Kijkend naar de omvang van de leerdoelen vinden de meeste respondenten dat ze genoeg (9) of zelfs te weinig (4) tijd hebben. Kijkend naar de eigen interesse in het onderwerp, vinden 3 respondenten dat ze met 1 SLU te veel tijd besteden aan grammatica's.

Voor grammatica's vindt er overwegend (12 van de 13 respondenten) individueel werk plaats. Slechts drie respondenten passen groepswerk in tweetallen toe, en slechts twee verhogen de groepsgrootte naar meer dan twee leden.

De leerdoelen richten zich vooral op de theoretische kant: grammatica's ontwikkelen (6 respondenten), de formele eigenschappen kennen en beschrijven (4), en het analyseren van input en talen die door de grammatica's worden geaccepteerd (10). 8 respondenten geven aan dat leerlingen zich bewust moeten zijn van de relatie tussen eindige automaten en grammatica's. Twee respondenten laten hun leerlingen een eigen parser ontwikkelen.

8 respondenten geven aan dat deze leerdoelen horen bij informatica en dat ze daarom moeten worden onderwezen. Vijf respondenten aangeven deze leerdoelen eigenlijk niet belangrijk te vinden.

11 van de 13 respondenten maken alleen gebruik van een theoretische toets om het onderwerp summatief af te sluiten. Slechts twee respondenten combineren deze toetsvorm met een praktische toets.

Geen van de respondenten past alleen een praktische toets toe.

Grammatica's worden voornamelijk behandeld in de eerdere leerjaren, maar wel bewust tegelijk met of ná de lessen over andere onderwerpen in het kernprogramma. Vervolgens worden vooral lessen over programmeren (subdomein D1 en D2) en enkele keuzethema's bewust ná de lessen over grammatica's georganiseerd.

4.2.4 Wensen voor de toekomst

Respondenten geven voor beide onderwerpen aan dat de tools in de breedste zin van het woord verbeterd moeten worden om de lessen erover te verbeteren: beter lesmateriaal en werkvormen, maar ook meer geschikte onderwerpen en betere tools om met de automaten en grammatica's te werken. Dit laatste geldt in het bijzonder voor eindige automaten: precies de helft van de respondenten geeft betere tools aan als verbeterpunt.

Twee respondenten zouden eindige automaten willen schrappen uit het examenprogramma. Zes respondenten willen grammatica's zien verdwijnen uit het programma. Ten slotte zijn per onderwerp vijf respondenten van mening dat beter/meer bij- en nascholing zou kunnen bijdragen aan betere lessen voor eindige automaten en grammatica's.

De meeste respondenten denken dat zij zelf (11), hun collega-docenten (10) en educatieve uitgeverijen (11) een bijdrage kunnen leveren aan de voorgestelde verbeteringen. Ook de SLO (8) en de vakvereniging i&i (7) worden veel gezien als partijen die hieraan zou kunnen bijdragen.

4.2.5 Overige opmerkingen

Twee respondenten geven aan grammatica's niet puur of apart te behandelen, maar vooral tussendoor, zoals tijdens het programmeren. Een andere respondent geeft aan dat de onderwerpen op dit moment te abstract en academisch zijn uitgewerkt in het bestaande lesmateriaal. Het materiaal sluit daarmee niet aan op de belevingswereld van de leerling. Ten slotte oppert een andere respondent om op zoek te gaan naar oplossingen om dit probleem te verhelpen, maar ook om de weerstand bij docenten tegen B3 te verminderen. Hij of zij geeft een aantal concrete voorstellen voor onderwerpen die dit doel moeten bereiken.

4.3 Interviews met docenten uit het voortgezet onderwijs

Na het afnemen van de gesloten vragenlijst hebben we drie respondenten uitgenodigd voor een interview. Met twee van deze uitgenodigde respondenten hebben we een interview afgenomen. De volledige transcripties van deze interviews zijn opgenomen in appendix D.

De respondenten zijn beschreven in de mannelijke vorm, bijvoorbeeld 'hij vindt' en 'zijn lessen'. Dit is gedaan om de leesbaarheid te bevorderen. De genderidentiteit van de respondenten wordt omwille van de anonimiteit niet vermeld; eigenlijk moet overal 'hij of zij' en 'zijn of haar' worden gelezen.

4.3.1 Interview 1

Eindige automaten

De respondent behandelt eindige automaten als onderdeel van een bredere module. Voor deze module gelden geen bijzondere afhankelijkheden.

In de module maken leerlingen eerst de schateiland-opdracht van *CS Unplugged* (Bell e.a., 2014). Vervolgens kiezen leerlingen een bestaand apparaat uit, zoals een wasmachine of een magnetron. Ze bestuderen het apparaat vanuit verschillende aspecten van de informatica. Zo produceren ze onder andere een eindige automaat die het gedrag van het apparaat beschrijft. Ze bekijken het apparaat ook met andere ‘informatica-brillen’, zoals usability en decompositie.

De automaten tekenen ze in Visio. Voorheen raadde de respondent draw.io aan, maar Visio heeft tegenwoordig de voorkeur. Het programma is geïntegreerd in de Teams-omgeving van de school en leerlingen kunnen er gemakkelijk in samenwerken.

De respondent vindt eindige automaten een nuttig hulpmiddel om het gedrag van systemen te modelleren — de praktische insteek. De theoretische insteek is leuk, maar minder nuttig. Die zou wellicht kunnen worden opgenomen in een hypothetisch keuzethema over computer science.

Grammatica's

De respondent behandelt het onderwerp grammatica's niet. Er kon geen logische plek voor dit onderwerp worden gevonden in het curriculum. Het onderwerp kan volgens hem ook geen zelfstandige module vormen. De respondent vermoedt dat zijn collega, om geen argwaan te wekken bij de onderwijsinspectie, in het PTA heeft aangegeven het onderwerp wél te behandelen in de module waarin eindige automaten worden behandeld.

Het onderwerp is volgens hem te theoretisch en betreft „harde computer science” die niet geschikt is als onderwerp voor het kernprogramma. Een havo-leerling met een CM-profiel is volgens de docent bijvoorbeeld niet geïnteresseerd in grammatica's. Hij ziet wel heil in het onderbrengen van dit onderwerp in een nieuw keuzethema, waarin deze ‘harde computer science’ wordt behandeld.

Wensen voor de toekomst

Dit schooljaar is de respondent begonnen met meer tijd uittrekken voor het oefenen met eindige automaten vóór de eindopdracht. Dat gaat aan de hand van het maken van eindige automaten voor kleinere, minder complexe, door hem ontwikkelde online apparaten.

De respondent zou graag grammatica's behandelen in de les, maar heeft hier expliciet externe hulp bij nodig. Hij staat open voor ondersteuning door uitgeverijen en collega-docenten. Hij zou graag beter geschikte onderwerpen en werkvormen vinden. Maar eigenlijk vindt hij het schrappen van het onderwerp realistischer dan een succesvolle uitvoering van het onderwerp in zijn curriculum.

4.3.2 Interview 2

Eindige automaten

Eerst hanteerde de respondent bij het behandelen van eindige automaten een theoretisch perspectief. Leerlingen haakten toen snel af. Daarom heeft hij de lesinhoud vereenvoudigd en praktischer gemaakt. Hij legt de focus op globaal begrip van automaten, en behandelt geen formele details. Hij merkt ook verschillen op tussen havo en vwo: vwo-leerlingen vinden het makkelijker dan havo-leerlingen. En bij havo-leerlingen kun je in zijn ervaring discussies over details beter uit de weg gaan.

Hij legt de link met andere vakken en met het dagelijks leven: welke processen en situaties kunnen leerlingen zelf als een eindige automaat beschouwen? Hij maakt zo bijvoorbeeld de link naar de onderzoek- en ontwerpcycli, die in zekere zin als eindige automaat zijn te beschouwen. Leerlingen van de respondent gaan inzien dat toestandsdiagrammen er zijn om overzicht te krijgen.

Voor het tekenen van eindige automaten gebruikte de respondent vroeger JFLAP. Door technische problemen is hij overgestapt naar een online tool (madebyevan.com/fsm).

De kennis over eindige automaten wordt summatief getoetst met een korte theoretische toets, om te verifiëren of de leerlingen de basis hebben begrepen. De respondent besteedt meer tijd en aandacht aan de behandeling en toetsing van grammatica's.

Grammatica's

De respondent kiest bij grammatica's bewust voor EBNF in plaats van BNF. De lessen over grammatica's geven begrip over hoe programmeertalen werken en waarom ze zo streng zijn. Hij legt ook verbanden tussen grammatica's en ChatGPT. Bovendien zorgen de lessen ervoor dat leerlingen begrijpen hoe parsers werken. Leerlingen onderschatten het vak in zijn ervaring vaak, maar zien uiteindelijk wel het nut en de relevantie van de onderwerpen in.

Tijdens de lessen over grammatica's ontwikkelen de leerlingen van de respondent een eigen grammatica die dingen produceert die nabij de belevingswereld van de leerling liggen. Te denken valt aan specificaties van een denkbeeldige computer, uitvoeringen van sportauto's en recepten voor hamburgers. De logica van hun grammatica moeten ze uitbeelden in een toestandsdiagram of flowchart. Vervolgens worden leerlingen gestimuleerd om de uitvoer van hun grammatica als input voor generatieve AI te maken. De AI maakt op basis van die uitvoer een afbeelding. Leerlingen ervaren dit als motiverend en uitdagend. Het roept de vraag op: hoe kunnen ze hun grammatica verbeteren om realistischere afbeeldingen te genereren?

Leerlingen met veel interesse in de grondslagen, gaan aan de slag met Turingmachines. Dit onderwerp is volgens de respondent wel hoog gegrepen. De meeste leerlingen komen er niet aan toe: in de laatste vijf jaar hebben slechts vier tot vijf leerlingen Turingmachines ontwikkeld.

De lessen over eindige automaten en grammatica's krijgen een begin bij de lessen over programmeren. De onderwerpen worden dan nog niet expliciet benoemd. Doel is om later, wanneer de leerlingen aan de slag gaan met deze onderwerpen, een 'oh ja'-moment te stimuleren. Leerlingen zien dan in dat de onderwerpen met elkaar te maken hebben: er ontstaat begrip van de onderlinge samenhang.

Wensen voor de toekomst

De respondent hekelt de theoretische insteek van de educatieve uitgeverijen, maar legt de schuld hiervan bij het examenprogramma. Dat is erg theoretisch opgezet, en daardoor zijn de auteurs in zijn optiek dicht bij de theorie en het examenprogramma gebleven in de ontwikkeling van hun lesmateriaal. Hij vindt dat niet nuttig voor havo-lesmateriaal.

De respondent ziet meerwaarde in meer gesprekken met andere collega-docenten om ervaringen te delen en de lessen te verbeteren. Ook zou hij graag een toelichting op het examenprogramma zien, waarin wordt uitgelegd hoe er gedifferentieerd moet worden voor havo- en vwo-leerlingen. Bovendien hoopt hij dat educatieve uitgeverijen zich minder richten op de theoretische kant, en werken aan lesmateriaal dat dichterbij de belevingswereld van de leerling staat.

5 Conclusies en discussie

In dit hoofdstuk geven we antwoord op de onderzoeksvragen. We leggen ook uit wat de beperkingen van het onderzoek zijn, en geven suggesties voor vervolgonderzoek.

5.1 Invulling van eindige automaten en grammatica's

Onderzoeksvraag 1

Welke contextuele invulling geven docenten in het voortgezet onderwijs aan de informaticalessen over eindige automaten en grammatica's?

- (a) Wat is de inhoud, omvang, toetsing, plaats in het curriculum en inbedding van lessen over eindige automaten en grammatica's in het voortgezet onderwijs?
- (b) Waarom hebben docenten voor deze invulling gekozen?
- (c) Welke wensen hebben docenten voor de toekomst van hun lessen over eindige automaten en grammatica's?

5.1.1 De invulling

Alle respondenten behandelen eindige automaten in zowel havo als vwo, maar slechts twee derde heeft aangegeven ook grammatica's te behandelen in de lessen. Dat betekent dus dat het PTA van een significant deel van de respondenten in feite niet voldoet aan de eisen uit het examenprogramma. Een van de geïnterviewde docenten vermoedt dat dit gebrek wordt opgelost door het onderwerp wel op te nemen in het PTA, maar niet in de daadwerkelijke lessen.

De meeste respondenten van ons onderzoek maken gebruik van een combinatie van eigen materiaal en lesmateriaal van een educatieve uitgeverij. Echter, voor grammatica's geldt dat ze meer berusten op het materiaal van de lesmethoden dan op het eigen materiaal. Een geïnterviewde docent gaf aan dat dit in de praktijk kan betekenen dat de lessen zeer theoretisch van aard zijn. Ons eigen onderzoek ondersteunt dit vermoeden; in paragraaf 2.1.1 constateerden we dat alle gangbare lesmethoden het lesmateriaal behoorlijk theoretisch inrichten. Leerlingen houden zich dan bezig met de formele eigenschappen van eindige automaten en grammatica's en blijven in de theorie. Een transfer naar de toepassing als instrument voor het ontwikkelen van artefacten, zoals gewenst door de auteurs van het examenprogramma, blijft grotendeels uit.

Een aantal respondenten geeft voor beide onderwerpen aan weinig eigen interesse te hebben. Een deel zou de onderwerpen zelfs willen schrappen uit het examenprogramma. Dat geldt in het bijzonder voor grammatica's. Het onderwerp zou volgens de geïnterviewde docenten te hoog gegrepen zijn voor havo-leerlingen. Een respondent vindt grammatica's bovendien moeilijk om te koppelen aan andere

onderwerpen in het examenprogramma.

Respondenten geven aan verschillende tools te gebruiken voor het tekenen/beschrijven van grammatica's. Pen en papier blijven erg populair, maar er zijn ook veel mogelijkheden op het gebied van digitale tools. Zo raadt één docent aan om Visio te gebruiken, door de integratie in de Teams-omgeving van de school. Ook nemen we een voorkeur voor online tools waar, die niet geïnstalleerd hoeven te worden. Dat levert minder problemen op wanneer leerlingen thuis met de tools aan de slag moeten.

Waar eindige automaten nog wisselend theoretisch, praktisch of gecombineerd theoretisch en praktisch worden getoetst, past bijna iedere respondent alleen een theoretische toets toe voor grammatica's. Dit is een bijzonder verschijnsel: docenten hekelen het theoretische karakter van de onderwerpen en het lesmateriaal, maar zetten nauwelijks praktische toetsvormen in.

Uit de vragenlijst en in het bijzonder uit de interviews blijkt dat docenten vooral naar de invulling van de onderwerpen door de educatieve uitgeverijen hebben gekeken. Die is behoorlijk theoretisch van aard. We vermoeden dat dat ertoe heeft geleid dat een deel ervoor koos het onderwerp grammatica's helemaal niet te behandelen. Het kan ook verklaren waarom docenten dit onderwerp, ondanks hun kritiek, theoretisch toetsen: ze weten niet hoe het anders zou kunnen. De geïnterviewde respondenten versterken dit vermoeden, en we gaan hier verder op in in de volgende paragraaf.

5.1.2 De wensen

Voor beide onderwerpen nemen we in het algemeen waar dat de respondenten behoefte hebben aan meer ondersteuning van onderwijsondersteunende partijen om hun lessen over eindige automaten en grammatica's te verbeteren. Ze doen ten eerste een beroep op hun collega's: door vaker samen te zitten en lesmateriaal en -ideeën uit te wisselen, kan men van elkaar leren. Ons onderzoek bevestigt dit: de geïnterviewde docenten lijken over ideeën en lesmateriaal te beschikken waar enkele respondenten van de vragenlijst naar vroegen.

Verder dienen educatieve uitgeverijen kritisch te kijken naar hun lesmateriaal over deze onderwerpen. De respondenten geven aan te verlangen dat de methoden zich minder bezighouden met de theoretische kanten van eindige automaten en grammatica's, en meer met de praktische inzetbaarheid, ook bij andere vakken en andere onderwerpen. Het huidige materiaal blijft te veel in het academische hangen. In feite wordt hier dezelfde hulpvraag als in de vorige alinea gesteld, maar aan een andere partij in het onderwijs.

Ten slotte horen we een kritische noot over de positie van deze onderwerpen in het examenprogramma. Een deel van de respondenten geeft aan dat eindige automaten en grammatica's niet thuishoren in het examenprogramma. Deze kritiek horen we het meest over grammatica's. In de interviews wordt de kritiek enigszins genuanceerd: het zou bijvoorbeeld opgenomen kunnen worden in een keuzethema, of alleen verplicht worden voor vwo-leerlingen. Het schrappen van deze onderwerpen voor havo-leerlingen zou ervoor zorgen dat het examenprogramma beter aansluit op de vervolgopleidingen: in paragraaf 2.1.4 lieten we zien dat deze onderwerpen helemaal niet terugkomen in hbo-opleidingen informatica.

5.2 Adviezen

Onderzoeksvraag 2

Welke aanbevelingen kunnen worden gedaan aan docenten en onderwijsondersteunende partijen om het voortgezet informatica-onderwijs over eindige automaten en grammatica's te verbeteren?

We hebben indicatief aangetoond dat de huidige invulling van de informaticalessen over eindige automaten en grammatica's niet helemaal aansluiten op het examenprogramma. Sommige docenten behandelen het tweede onderwerp helemaal niet, maar dat is eigenlijk niet toegestaan. Veel respondenten duiken de diepte in en gaan theoretisch met de onderwerpen aan de slag. Dat is weliswaar toegestaan volgens de letter van het examenprogramma, maar niet volgens de geest — de lessen moeten praktischer worden, waar de concepten als instrument worden ingezet voor de constructie van digitale artefacten. Docenten hebben behoefte aan hulp van elkaar en van de educatieve uitgeverijen om dit doel te bereiken.

Bovendien hebben we laten zien dat de onderwerpen totaal niet terugkomen in hbo-opleidingen. Dat is frappant, want ze zijn wel verplicht onderdeel van het examenprogramma voor havo-leerlingen. Er wordt hier een belangrijk verschil blootgelegd tussen het examenprogramma op de middelbare school enerzijds, en het mogelijke studieprogramma van een havo-informaticaleerling die een informatica-vervolgopleiding gaat volgen anderzijds. Ook hebben we indicatief aangetoond dat docenten de hieruit volgende implicatie ondersteunen: eindige automaten en grammatica's zouden op basis van hun ervaringen eigenlijk niet thuishoren in het havo-kernprogramma.

Op basis van de in dit onderzoek gepresenteerde resultaten en conclusies brengen we de volgende concrete adviezen uit voor docenten en onderwijsondersteunende partijen.

1. We adviseren docenten om elkaar op te zoeken en in gesprek te gaan over de invulling van lessen over deze onderwerpen. Het doel van deze gesprekken is uitwisseling van lesmateriaal en -ideeën, om de lessen over eindige automaten en grammatica's minder theoretisch te maken.
2. We adviseren educatieve uitgeverijen om hun lesmateriaal over eindige automaten en grammatica's te vernieuwen. Hierbij moet de aandacht gericht zijn op de praktische inzetbaarheid van deze concepten als instrumenten voor de constructie van digitale artefacten, in lijn met de oorspronkelijke intenties van de commissie die het examenprogramma heeft opgesteld.
3. Ten slotte adviseren we het Ministerie van Onderwijs, Cultuur en Wetenschap om bij een volgende actualisering van het examenprogramma te kijken naar de positie van deze onderwerpen in het kernprogramma. We hebben indicatief aangetoond dat de onderwerpen wellicht beter thuishoren in een keuzethema, en dat ze voor havo zelfs helemaal kunnen worden geschrapt.

5.3 Beperkingen en vervolgonderzoek

Op basis van de resultaten van ons onderzoek kunnen we indicatieve conclusies trekken over de invulling van eindige automaten en grammatica's in het voortgezet informatica-onderwijs. Helaas zorgt het kleine aantal respondenten (19) ten opzichte van de geschatte hoeveelheid informaticadocenten (ongeveer 800) ervoor dat deze conclusies niet representatief zullen zijn. Er is meer vervolgonderzoek nodig naar de invulling, mogelijkheden en toekomst van eindige automaten en grammatica's in het informatica-onderwijs.

We vermoeden dat de vragenlijst uiteindelijk een lage respons heeft gekregen door de lengte ervan. De lage respons van de vragenlijst heeft ook gezorgd voor minder potentiële respondenten voor de interviews. Een vervolgonderzoek met een kortere vragenlijst levert weliswaar minder inzicht op, maar kan wel een hogere representativiteit hebben. Bovendien zouden meer respondenten voor een kortere vragenlijst kunnen zorgen voor meer potentiële respondenten voor nieuwe interviews.

Verder zien we heil in een vervolgonderzoek waarin men ontwerponderzoek doet naar geschikt lesmateriaal voor eindige automaten en grammatica's. Een geschikt ontwerpdoel voor zulk onderzoek is een betere aansluiting met de gedachtegang achter deze onderwerpen: als praktisch instrument voor de constructie van digitale artefacten.

Ten slotte merken we op dat het zinvol kan zijn om verdiepend onderzoek te doen naar de positie en invulling van andere onderwerpen in het examenprogramma, zoals in dit onderzoek is gedaan voor eindige automaten en grammatica's. Zo'n onderzoek kan ook specifiekere vragen proberen te beantwoorden, zoals welke werkvormen geschikt zijn voor een bepaald onderwerp, en wat de leeropbrengst is van lessen zoals die nu worden gegeven.

Bibliografie

- Barendsen, E., & Steenvoorden, T. (2016). Analyzing Conceptual Content of International Informatics Curricula for Secondary Education. *Informatics in Schools: Improvement of Informatics Knowledge and Perception*, 14–27. https://doi.org/10.1007/978-3-319-46747-4_2
- Barendsen, E., & Tolboom, J. (2016). *Advies Examenprogramma Informatica vwo-havo: inhoud en invoering* (tech. rap.). SLO. Enschede.
- Bell, T., Witten, I. H., Fellows, M., Adams, R., McKenzie, J., van Kessel, S., van der Meer, M., van Montfort, J., & van de Ven, C. (2014). Computer Science Unplugged: Een verdiepings- en uitbreidingsprogramma voor het basisonderwijs. (g.u.)
- Bezáková, D., & Winczer, M. (2011). Teaching theoretical informatics to secondary school informatics teachers. *Informatics in Schools. Contributing to 21st Century Education*, 117–128. https://doi.org/10.1007/978-3-642-24722-4_11
- Cepeda, N. J., Pashler, H., Vul, E., Wixted, J. T., & Rohrer, D. (2006). Distributed practice in verbal recall tasks: A review and quantitative synthesis. *Psychological bulletin*, 132(3), 354.
- Chesnevar, C. I., González, M. P., & Maguitman, A. G. (2004). Didactic strategies for promoting significant learning in formal languages and automata theory. *ACM SIGCSE Bulletin*, 36(3), 7–11.
- De Souza, G. S., Olivete, C., Correia, R. C. M., & Garcia, R. E. (2015). Teaching-learning methodology for formal languages and automata theory. *2015 IEEE Frontiers in Education Conference (FIE)*, 1–7. <https://doi.org/10.1109/FIE.2015.7344185>
- den Uijl, M., Adriaens, H., & Elshout, M. (2023). *Personeelstekorten voortgezet onderwijs* (onderzoeksrap.). Centerdata.
- Dienst Uitvoering Onderwijs. (2025). *Lesgevend personeel in personen* (statistisch rap.). Dienst Uitvoering Onderwijs.
- Dongol, B., Dubois, C., Hallerstede, S., Hehner, E., Morgan, C., Müller, P., Ribeiro, L., Silva, A., Smith, G., & de Vink, E. (2024). On Formal Methods Thinking in Computer Science Education. *Form. Asp. Comput.*, 37(1). <https://doi.org/10.1145/3670419>
- Examenprogramma informatica havo/vwo 2007. (2007). https://www.examenblad.nl/system/files/2007/exprog/havovwo/inf_havovwo.pdf
- Examenprogramma informatica havo/vwo 2019. (2019). https://www.examenblad.nl/system/files/2018/examenprogramma_Informatica_havo-vwo.pdf
- Grgurina, N., Tolboom, J., & de Vries, B. P. (2023). Evaluating the New Secondary Informatics Curriculum in The Netherlands: The Teachers' Perspective. *Informatics in Schools. Beyond Bits and Bytes: Nurturing Informatics Intelligence in Education*, 155–166. https://doi.org/10.1007/978-3-319-46747-4_2
- Groen, N., & Viskil, E. (1994). InZicht [Geraadpleegd via DBNL (KB, nationale bibliotheek)]. *Onze Taal*, 63(1), 26. Verkregen mei 26, 2025, van https://www.dbnl.org/tekst/_taa014199401_01/_taa014199401_01_0022.php
- Hazzan, O., Lapidot, T., & Ragonis, N. (2014). Guide to Teaching Computer Science. Springer.

- Hopcroft, J. E., Motwani, R., & Ullman, J. D. (2007). *Automata Theory, Languages, and Computation*. Pearson.
- Kumar, A. N., Raj, R. K., Aly, S. G., Anderson, M. D., Becker, B. A., Blumenthal, R. L., Eaton, E., Epstein, S. L., Goldweber, M., Jalote, P., Lea, D., Oudshoorn, M., Pias, M., Reiser, S., Servin, C., Simha, R., Winters, T., & Xiang, Q. (2023). Computer Science Curricula 2023. <https://doi.org/10.1145/3664191>
- Lazar, J., Feng, J. H., & Hochheiser, H. (2017). *Research Methods in Human-Computer Interaction*. Morgan Kaufmann Publishers.
- Lepeltak, J. (1990). De taalkist: een zinsgenerator voor het onderwijs in informatica en Nederlands in de tweede fase van het voortgezet onderwijs. *NIOC 1990 Maastricht*.
- SLO. (2020). Handreiking SE Informatica.
- Tolboom, J., Krüger, J., & Grgurina, N. (2014). *Informatica in de bovenbouw havo/vwo: naar aantrekkelijk en actueel onderwijs in informatica* (tech. rap.). SLO. Enschede.
- van Lith, P., Schalkx, F., van Inge, A., van Dijk, H., van Veen, N., & de Beurs, C. (2009). *Leerling-handleiding NLT-module Robotica*.
- van der Donk, C., & van Lanen, B. (2024). *Praktijkonderzoek in de school*. Uitgeverij Coutinho.
- VO-raad. (2025). Checklist 'Programma van Toetsing en Afsluiting' (versie 2025–2026).
- Wet Voortgezet Onderwijs. (2020). Verkregen februari 4, 2025, van https://wetten.overheid.nl/BWBR0044212/2025-01-01#Hoofdstuk2_Paragraaf5_Artikel2.60a

A | Open vragenlijst

Introductie

Bedankt dat je deze open vragenlijst wilt invullen! Jouw antwoorden helpen mij enorm veel in het doen van mijn onderzoek. In dit gedeelte van het onderzoek wil ik eerst kennismaken: aan wie geef je les, en met welke methode(n) doe je dat?

1. Aan welke klas(sen) geef je informaticalessen?

☐ havo 4

☐ vwo 4

☐ vwo 6

☐ havo 5

☐ vwo 5

2. Welke lesmethode(n) gebruik je voor je lessen over informatica?

☐ Fundament Informatica

☐ Informatica-Actief

☐ VO-Content/Enigma

☐ Eigen lesmateriaal

☐ Lesmateriaal van collega's op eigen school

☐ Lesmateriaal van collega's op andere scholen

3. Indien je meerdere lesmethoden gebruikt, hoe verdeel je dat dan?

Voorbeeld 1: Je gebruikt een bestaande lesmethode in de voorexamenklassen, en eigen materiaal in de examenklassen.

Voorbeeld 2: Je gebruikt een bestaande lesmethode, en vult die aan met eigen lesmateriaal.

.....
.....
.....

Eindige automaten — behandel je dit onderwerp?

4. Behandel je het subdomein 'eindige automaten' (B3) in welke vorm dan ook in de informaticalessen?

Het gaat hier om elke vorm van behandeling. Dus ook als je het onderwerp technisch gezien wel behandelt, maar bijvoorbeeld nooit bij naam noemt. Juist in zulke invullingen zijn we enorm geïnteresseerd!

Vul dus alleen ‘nee’ in als je echt helemaal niets met dit onderwerp doet in jouw informaticalessen.

- Ja (Ga door naar **vraag 5**)
- Nee (Ga door naar **vraag 24**)

Eindige automaten

Je geeft les over eindige automaten. In mijn onderzoek wil ik erachter komen op welke manieren dat allemaal kan. Ik kijk daarbij naar de aspecten inhoud, omvang, toetsing, plaats in het curriculum en inbedding.

Inhoud

Met de **inhoud** bedoel ik welke concrete onderwerpen je behandelt. De formele term hiervoor is context. Je zou bijvoorbeeld domein D (programmeren) kunnen onderwijzen door leerlingen een game te laten programmeren met Pygame. Dat is dan de inhoud.

5. Welke onderwerpen gebruik je om het subdomein eindige automaten te behandelen?

.....

.....

.....

6. Welke tools gebruik je/gebruiken je leerlingen om eindige automaten te tekenen / construeren?

.....

.....

.....

7. Waarom heb je voor deze onderwerpen gekozen voor de lessen over eindige automaten?

.....

.....

.....

8. Welke mogelijke onderwerpen heb je nog meer gevonden/overwogen?

.....

.....

.....

Omvang

De **omvang** is de tijdsduur van het onderwerp.

9. Hoeveel SLU besteed je aan eindige automaten?

Een SLU is één studielastuur. Het gaat hier om alle studielasturen die leerlingen besteden aan eindige automaten. Weet je het niet zeker, geef dan een inschatting.

.....

10. Waarom heb je voor deze hoeveelheid gekozen?

.....

.....

.....

11. Wat voor werk verrichten leerlingen voor het onderwerp eindige automaten?

Te denken valt aan:

- individueel werk
- groepswerk (geef dan ook de groepsgrootte aan)
- iets anders
- een combinatie hiervan

.....

.....

.....

Toetsing

Toetsing houdt niet alleen in hoe je de eindterm toetst, maar ook wat je toetst. Het gaat hier dus om toetsvormen en leerdoelen.

12. Wat moeten leerlingen kunnen aan het einde van de lessen over eindige automaten?

Het gaat hier in feite om de leerdoelen. Je mag dit zo uitgebreid omschrijven als je zelf wilt. Je mag er ook voor kiezen om de leerdoelen enigszins samen te vatten.

.....

.....

.....

13. Hoe ben je gekomen tot de huidige leerdoelen voor dit onderwerp?

.....

.....

.....

14. Waarom vind je deze leerdoelen belangrijk?

.....

.....

.....

15. Hoe wordt de kennis/kunde over eindige automaten getoetst?

- ☐ Met een theoretische toets, die leerlingen op papier maken.
- ☐ Met een theoretische toets, die leerlingen met de computer maken.
- ☐ Met een praktische toets, met als eindproduct (onder andere) een fysiek product.
- ☐ Met een praktische toets, met als eindproduct (onder andere) een digitaal product.
- ☐ Met een presentatie door leerling(en).
- ☐ Anders:

16. Waarom heb je voor deze vorm van toetsing gekozen?

.....

.....

.....

Plaats in het curriculum

De **plaats in het curriculum** wordt vaak bewust gekozen. In de eerste plaats natuurlijk omdat je nu eenmaal gebonden bent aan een jaarrooster. Maar vaak kies je er ook bewust voor om eerst het ene, en dan het andere onderwerp te behandelen.

17. In welk(e) leerjaar/leerjaren behandel je eindige automaten?

- | | | |
|---------------------------------|--------------------------------|--------------------------------|
| ☐ havo 4 | ☐ vwo 4 | ☐ vwo 6 |
| ☐ havo 5 | ☐ vwo 5 | |

18. Waarom behandel je eindige automaten in dit leerjaar/deze leerjaren?

.....

.....

.....

19. Welke onderwerpen behandel je bewust vóór de lessen over eindige automaten?

Leg ook uit waarom.

.....

.....

.....

20. Welke onderwerpen behandel je bewust ná de lessen over eindige automaten?

Leg ook uit waarom.

.....

.....

.....

Inbedding

De term **inbedding** houdt in dat een onderwerp onderdeel is van een lessenserie over één of meerdere grotere onderwerpen. Je kunt bijvoorbeeld een project hebben waar leerlingen programmeren in HTML, PHP en MySQL om een website te maken. Je behandelt dan de eindtermen D (programmeren) en C5 (databases).

21. Maken de lessen over eindige automaten uit van een grotere lessenserie?

- o Nee
- o Ja

22. Indien ja, waar gaat de lessenserie over?

.....
.....
.....

23. En indien ja, waarom heb je hiervoor gekozen?

.....
.....
.....

Grammatica's — behandel je dit onderwerp?

24. Behandel je het subdomein 'eindige automaten' (B3) in welke vorm dan ook in de informaticalessen?

Het gaat hier om elke vorm van behandeling. Dus ook als je het onderwerp technisch gezien wel behandelt, maar bijvoorbeeld nooit bij naam noemt. Juist in zulke invullingen zijn we enorm geïnteresseerd!

Vul dus alleen 'nee' in als je echt helemaal niets met dit onderwerp doet in jouw informaticalessen.

- o Ja (Ga door naar **vraag 25**)
- o Nee (Ga door naar **vraag 44**)

Grammatica's

Je geeft les over grammatica's. In mijn onderzoek wil ik erachter komen op welke manieren dat allemaal kan. Ik kijk daarbij naar de aspecten inhoud, omvang, toetsing, plaats in het curriculum en inbedding.

Inhoud

Met de **inhoud** bedoel ik welke concrete onderwerpen je behandelt. De formele term hiervoor is context. Je zou bijvoorbeeld domein D (programmeren) kunnen onderwijzen door leerlingen een game te laten programmeren met Pygame. Dat is dan de inhoud.

25. Welke onderwerpen gebruik je om het subdomein grammatica's te behandelen?

.....

.....

.....

26. Welke tools gebruik je/gebruiken je leerlingen om grammatica's te noteren / construeren?

.....

.....

.....

27. Waarom heb je voor deze onderwerpen gekozen voor de lessen over grammatica's?

.....

.....

.....

28. Welke mogelijke onderwerpen heb je nog meer gevonden/overwogen?

.....

.....

.....

Omvang

De **omvang** is de tijdsduur van het onderwerp.

29. Hoeveel SLU besteed je aan grammatica's?

Een SLU is één studielastuur. Het gaat hier om alle studielasturen die leerlingen besteden aan grammatica's. Weet je het niet zeker, geef dan een inschatting.

.....

30. Waarom heb je voor deze hoeveelheid gekozen?

.....

.....

.....

31. Wat voor werk verrichten leerlingen voor het onderwerp grammatica's?

Te denken valt aan:

- individueel werk
- groepswork (geef dan ook de groepsgrootte aan)
- iets anders
- een combinatie hiervan

.....
.....
.....

Toetsing

Toetsing houdt niet alleen in hoe je de eindterm toetst, maar ook wat je toetst. Het gaat hier dus om toetsvormen en leerdoelen.

32. Wat moeten leerlingen kunnen aan het einde van de lessen over grammatica's?

Het gaat hier in feite om de leerdoelen. Je mag dit zo uitgebreid omschrijven als je zelf wilt. Je mag er ook voor kiezen om de leerdoelen enigszins samen te vatten.

.....
.....
.....

33. Hoe ben je gekomen tot de huidige leerdoelen voor dit onderwerp?

.....
.....
.....

34. Waarom vind je deze leerdoelen belangrijk?

.....
.....
.....

35. Hoe wordt de kennis/kunde over grammatica's getoetst?

- ☐ Met een theoretische toets, die leerlingen op papier maken.
- ☐ Met een theoretische toets, die leerlingen met de computer maken.
- ☐ Met een praktische toets, met als eindproduct (onder andere) een fysiek product.
- ☐ Met een praktische toets, met als eindproduct (onder andere) een digitaal product.
- ☐ Met een presentatie door leerling(en).
- ☐ Anders:

36. Waarom heb je voor deze vorm van toetsing gekozen?

.....
.....
.....

Plaats in het curriculum

De **plaats in het curriculum** wordt vaak bewust gekozen. In de eerste plaats natuurlijk omdat je nu eenmaal gebonden bent aan een jaarrooster. Maar vaak kies je er ook bewust voor om eerst het ene, en dan het andere onderwerp te behandelen.

37. In welk(e) leerjaar/leerjaren behandel je grammatica's?

☐ havo 4

☐ vwo 4

☐ vwo 6

☐ havo 5

☐ vwo 5

38. Waarom behandel je grammatica's in dit leerjaar/deze leerjaren?

.....
.....
.....

39. Welke onderwerpen behandel je bewust vóór de lessen over grammatica's?

Leg ook uit waarom.

.....
.....
.....

40. Welke onderwerpen behandel je bewust ná de lessen over grammatica's?

Leg ook uit waarom.

.....
.....
.....

Inbedding

De term **inbedding** houdt in dat een onderwerp onderdeel is van een lessenserie over één of meerdere grotere onderwerpen. Je kunt bijvoorbeeld een project hebben waar leerlingen programmeren in HTML, PHP en MySQL om een website te maken. Je behandelt dan de eindtermen D (programmeren) en C5 (databases).

41. Maken de lessen over grammatica's uit van een grotere lessenserie?

o Nee

o Ja

42. Indien ja, waar gaat de lessenserie over?

.....
.....
.....

43. En indien ja, waarom heb je hiervoor gekozen?

.....

.....

.....

Wensen voor de toekomst

De onderwerpen ‘eindige automaten’ en ‘grammatica’s’ zijn bron van hevige discussie. Eén ding is duidelijk: er is ruimte voor verbetering. Graag vragen we je hoe jij hierin staat!

44. Hoe denk je dat lessen over eindige automaten nog beter zouden kunnen worden?

.....

.....

.....

45. Hoe denk je dat lessen over grammatica’s nog beter zouden kunnen worden?

.....

.....

.....

46. Wat denk je hier **zelf** aan te kunnen bijdragen?

.....

.....

.....

47. Wat denk je dat **onderwijsondersteunende partijen** hieraan zouden kunnen bijdragen?

Onderwijsondersteunende partijen zijn alle partijen die een bijdrage leveren aan het onderwijs in de breedste zin van het woord:

- Het ministerie van OCW;
- De onderwijsraad;
- Gemeentes;
- Schoolbesturen;
- De SLO;
- Het CvTE;
- Cito;
- De onderwijsinspectie;
- Schoolleiders, teamleiders, teams, leraren, coördinatoren, intern begeleiders, leerlingen – en alle andere partijen op school;
- Lerarenopleidingen;
- Educatieve uitgeverijen en ontwikkelaars;

- Universiteiten en onderzoeksinstellingen;
- Schoolbegeleidingsdiensten, adviesbureaus en steunpunten;
- Sectorraden, vakbonden, koepelorganisaties en vakverenigingen;
- Ouders, lobbygroepen, opiniemakers;
- (Algemene) onderwijsvakbladen, kranten, tv, radio, websites en sociale media.

.....

.....

.....

48. Wil je nog iets anders kwijt, wat ik nog niet heb gevraagd, maar waarvan je wel denkt dat het relevant is voor dit onderzoek?

.....

.....

.....

Bedankt!

Je bent aan het eind van deze vragenlijst gekomen. Allereerst wil ik je ontzettend bedanken voor het invullen van de vragenlijst. Je input is enorm waardevol en betekent veel voor dit onderzoek!

Let op: je antwoorden zijn nog niet opgeslagen. Klik onderaan op ‘Verzenden’ om je antwoorden op te slaan.

Verzenden

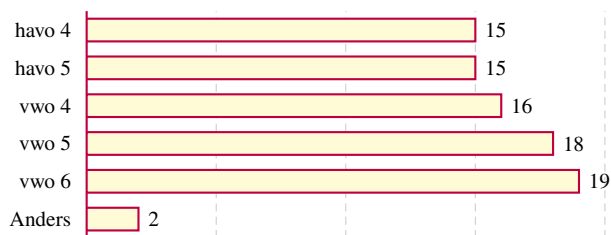
B | Gesloten vragenlijst

Introductie

Bedankt dat je deze open vragenlijst wilt invullen! Jouw antwoorden helpen mij enorm veel in het doen van mijn onderzoek. In dit gedeelte van het onderzoek wil ik eerst kennismaken: aan wie geef je les, en met welke methode(n) doe je dat?

1. Aan welke klas(sen) geef je informatica's?

- | | | |
|---------------------------------|--------------------------------|--|
| ☐ havo 4 | ☐ vwo 4 | ☐ vwo 6 |
| ☐ havo 5 | ☐ vwo 5 | ☐ Anders: |

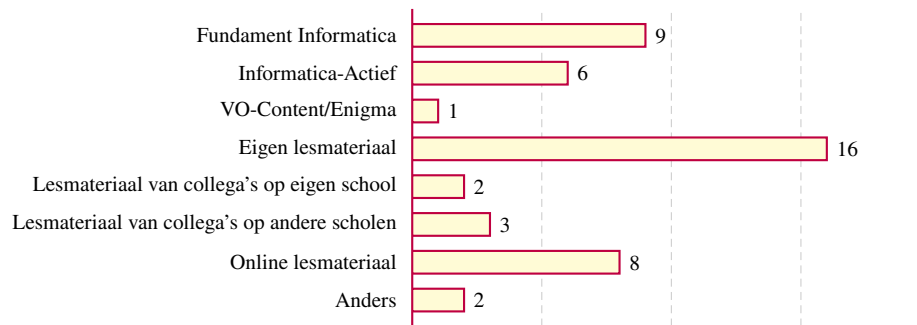


Anders:

- „Agora en hoogbegaafden”
- „heb les gegeven en bovenbouw en was als taalkundige in jaren '90 betrokken bij curriculum ontwikkeling informatica bij de SLO”

2. Welke lesmethode(n) gebruik je voor je lessen over informatica?

- | |
|---|
| ☐ Fundament Informatica |
| ☐ Informatica-Actief |
| ☐ VO-Content/Enigma |
| ☐ Eigen lesmateriaal |
| ☐ Lesmateriaal van collega's op eigen school |
| ☐ Lesmateriaal van collega's op andere scholen |
| ☐ Online lesmateriaal |
| ☐ Anders: |



Anders:

- „Informatica voor het VO”
- „experimentele methode ontwikkeld met software getiteld de Taalkist”

3. Mogen we je benaderen voor een interview?

Als je dat wilt, kun je je contactgegevens achterlaten. We kunnen je dan uitnodigen voor een interview naar aanleiding van de antwoorden die je hebt gegeven.

Het achterlaten van je contactgegevens is geheel vrijwillig. Ook als je dat liever niet doet, maar wel de vragenlijst invult, help je ons enorm met dit onderzoek!

Indien je uitgenodigd wil worden voor een interview, laat dan hieronder je e-mailadres achter. Indien je dit niet wil, laat dit veld dan leeg.

.....

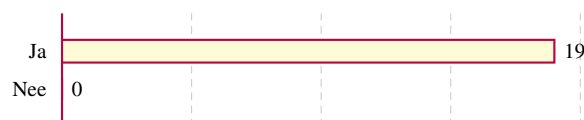
Eindige automaten — behandel je dit onderwerp?

4. Behandel je het subdomein ‘eindige automaten’ (B3) in welke vorm dan ook in de informatica-lessen?

Het gaat hier om elke vorm van behandeling. Dus ook als je het onderwerp technisch gezien wel behandelt, maar bijvoorbeeld nooit bij naam noemt. Juist in zulke invullingen zijn we enorm geïnteresseerd!

Vul dus alleen ‘nee’ in als je echt helemaal niets met dit onderwerp doet in jouw informatica-lessen.

- Ja (Ga door naar **vraag 5**)
- Nee (Ga door naar **vraag 19**)



Eindige automaten

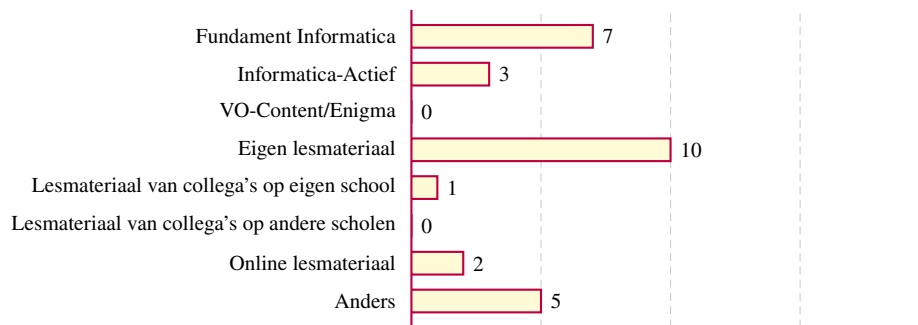
Je geeft les over eindige automaten. In mijn onderzoek wil ik erachter komen op welke manieren dat allemaal kan. Ik kijk daarbij naar de aspecten inhoud, omvang, toetsing, plaats in het curriculum en inbedding.

Inhoud

Met de **inhoud** bedoel ik welke concrete onderwerpen je behandelt. De formele term hiervoor is context. Je zou bijvoorbeeld domein D (programmeren) kunnen onderwijzen door leerlingen een game te laten programmeren met Pygame. Dat is dan de inhoud.

5. Welke lesmethode(n) gebruik je voor je lessen over eindige automaten?

- ☐ Fundament Informatica
- ☐ Informatica-Actief
- ☐ VO-Content/Enigma
- ☐ Eigen lesmateriaal
- ☐ Lesmateriaal van collega's op eigen school
- ☐ Lesmateriaal van collega's op andere scholen
- ☐ Online lesmateriaal
- ☐ Anders:



Anders:

- „De Taalkist. over Taal en computers”
- „Informatica voor het VO”
- „De module Physical computing met de micro:bit op wikiwijs voor SLO”
- „csunplugged”
- „piratenbaai en jflap zie inf4all”

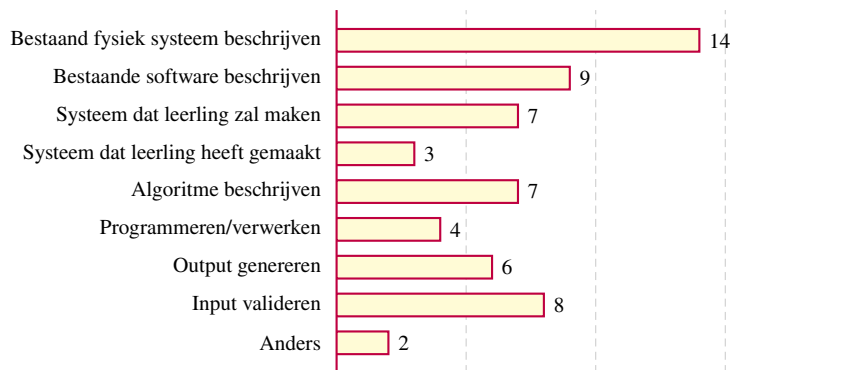
6. Welke onderwerpen gebruik je om het subdomein eindige automaten te behandelen?

In de antwoorden is ‘eindige automaat’ afgekort tot EA.

Het verschil in de antwoorden ‘[...] dat de leerling zelf zal maken’ en ‘[...] dat de leerling zelf heeft gemaakt’ is het moment waarop de eindige automaat wordt geconstrueerd. In de eerste variant maakt de leerling eerst een eindige automaat, en moet het systeem daarop worden gebaseerd. In de tweede variant maakt de leerling achteraf een eindige automaat, die het gedrag van het door hem/haar ontwikkelde systeem beschrijft.

- ☐ Een EA maken, om het gedrag van een bestaand, fysiek systeem te beschrijven
- ☐ Een EA maken, om het gedrag van een bestaand softwaresysteem te beschrijven

- ☐ Een EA maken, om het gedrag van een systeem te beschrijven dat de leerling zelf zal maken
- ☐ Een EA maken, om het gedrag van een systeem te beschrijven dat de leerling zelf heeft gemaakt
- ☐ Een EA maken, om een algoritme te beschrijven
- ☐ Een EA programmeren/verwerken in een computerprogramma
- ☐ Een EA gebruiken om output (bv. tekst) te genereren
- ☐ Een EA gebruiken om input (bv. tekst) te valideren
- ☐ Anders:



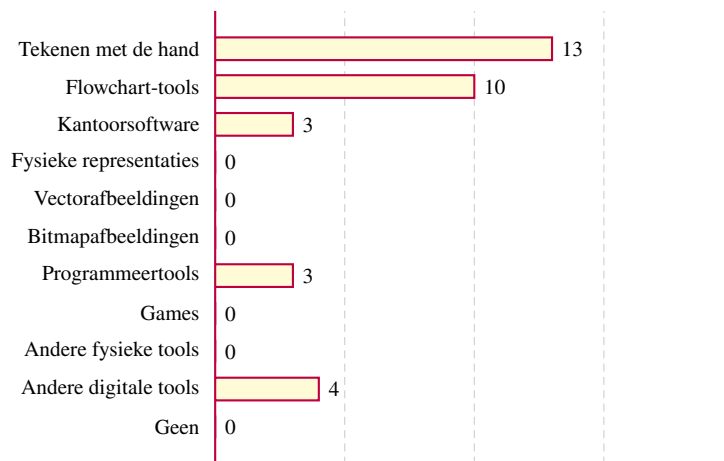
Anders:

- „waarom een EA tekort schiet als model voor een grammatica can een natuurlijke taal”
- „Ik gebruik alleen de opdrachten uit de methode”

7. Welke tools gebruik je/gebruiken je leerlingen om eindige automaten te tekenen / construeren?

Probeer zo specifiek mogelijk te zijn. Kies alleen de laatste twee antwoorden als jouw tools echt niet onder de overige categorieën vallen.

- ☐ Tekenen met de hand, zoals met pen/potlood/stift en papier/whiteboard/krijtbord
- ☐ Digitale diagram-, flowchart- en mindmap-tools, zoals Draw.io, Lucidchart of Visio
- ☐ Kantoorsoftware, zoals PowerPoint, LibreOffice Present of Google Slides
- ☐ Fysieke representaties, zoals bouwstenen en knutselmaterialen
- ☐ Grafische ontwerpsoftware voor vectorafbeeldingen, zoals InkScape of Adobe Illustrator
- ☐ Grafische ontwerpsoftware voor bitmapafbeeldingen, zoals GIMP, Paint.net of Photoshop
- ☐ Programmeertools voor eindige automaten, zoals JFLAP, Graphviz, TikZ of Python met Matplotlib
- ☐ Games, zoals Minecraft of Roblox
- ☐ Andere fysieke tools
- ☐ Andere digitale tools
- ☐ Mijn leerlingen tekenen/construeren zelf geen eindige automaten



Omvang

De **omvang** is de tijdsduur van het onderwerp.

8. Hoeveel SLU besteed je aan eindige automaten?

Een SLU is één studielastuur van 60 minuten. Het gaat hier om alle studielasturen die leerlingen besteden aan eindige automaten. Weet je het niet zeker, geef dan een inschatting.

.....

9. Wat vind je van deze hoeveelheid SLU ten opzichte van de leerdoelen?

Het gaat hier om de relatie tussen de leerdoelen en de tijd die je besteed aan het behalen van de leerdoelen.

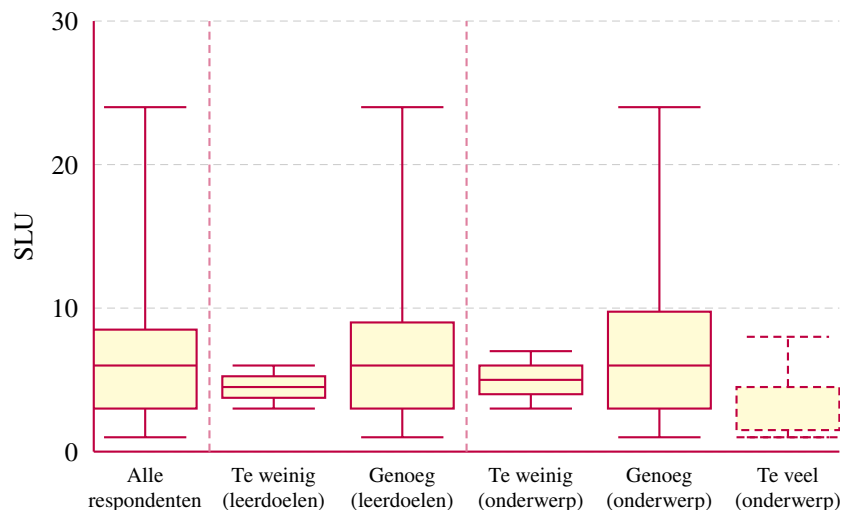
- ☐ Te weinig
 ☐ Genoeg
 ☐ Te veel

10. Wat vind je van deze hoeveelheid SLU ten opzichte van het onderwerp?

Het gaat hier om de relatie tussen jouw eigen enthousiasme/passie over het onderwerp en de tijd die je besteed aan het onderwerp. Vind je het onderwerp totaal niet interessant, dan is je antwoord waarschijnlijk 'te veel'. Vind je het onderwerp enorm interessant, en zou je er meer tijd aan willen besteden, dan is je antwoord waarschijnlijk 'te weinig'.

- ☐ Te weinig
 ☐ Genoeg
 ☐ Te veel

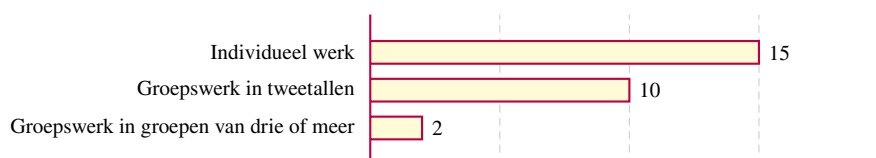
De antwoorden op bovenstaande vragen zijn hieronder in boxplots en in een tabel weergegeven.



Groep	n	Min	Q1	Mediaan	Q3	Max	SD
Alle respondenten	19	1.0	3.00	6.0	8.50	24.0	5.65
Te weinig (leerdoelen)	2	3.0	3.75	4.5	5.25	6.0	2.12
Genoeg (leerdoelen)	17	1.0	3.00	6.0	9.00	24.0	5.90
Te veel (leerdoelen)	0	–	–	–	–	–	–
Te weinig (onderwerp)	2	3.0	4.00	5.0	6.00	7.0	2.83
Genoeg (onderwerp)	14	1.0	3.00	6.0	9.75	24.0	6.20
Te veel (onderwerp)	3	1.0	1.50	1.0	4.50	8.0	3.78

11. Wat voor werk verrichten leerlingen voor het onderwerp eindige automaten?

- ☐ Individueel werk
- ☐ Groepswork in tweetallen
- ☐ Groepswork in groepen van drie of meer



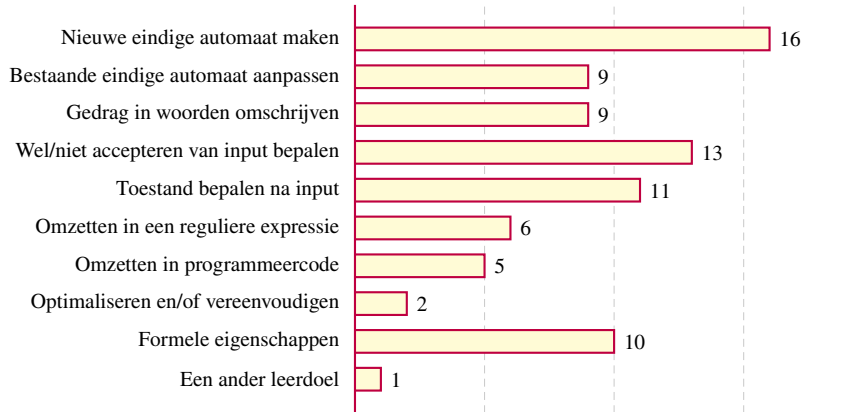
Toetsing

Toetsing houdt niet alleen in hoe je de eindterm toetst, maar ook wat je toetst. Het gaat hier dus om toetsvormen en leerdoelen.

12. Wat moeten leerlingen kunnen aan het einde van de lessen over eindige automaten?

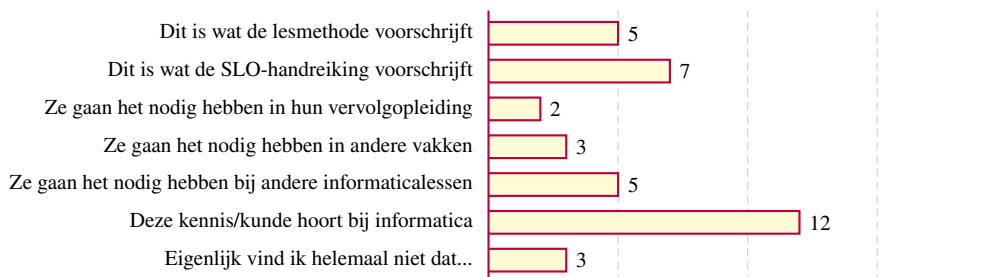
- ☐ Op basis van een beschrijving/opdracht een nieuwe eindige automaat maken
- ☐ Op basis van een beschrijving/opdracht een bestaande eindige automaat aanpassen
- ☐ Het gedrag van een eindige automaat in woorden omschrijven
- ☐ Bepalen of een eindige automaat een bepaalde input wel/niet accepteert
- ☐ Bepalen in welke toestand een eindige automaat zich bevindt na een bepaalde input
- ☐ Een eindige automaat omzetten in een reguliere expressie

- ☐ Een eindige automaat omzetten in programmeercode
- ☐ Een eindige automaat optimaliseren en/of vereenvoudigen
- ☐ De formele eigenschappen (toestand, transitie, etc) kennen en benoemen
- ☐ Een ander leerdoel dat hier nog niet is genoemd



13. Waarom vind je dat je leerlingen deze leerdoelen moeten beheersen?

- ☐ Dit is wat de lesmethode voorschrijft
- ☐ Dit is wat de SLO-handreiking voorschrijft
- ☐ Ze gaan het nodig hebben in hun vervolgopleiding
- ☐ Ze gaan het nodig hebben in andere vakken
- ☐ Ze gaan het nodig hebben bij andere informaticalessen
- ☐ Deze kennis/kunde hoort bij informatica
- ☐ Eigenlijk vind ik helemaal niet dat mijn leerlingen deze leerdoelen moeten beheersen.

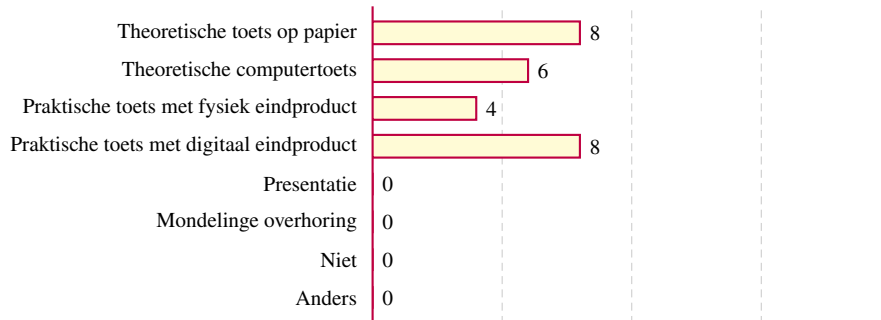


14. Hoe wordt de kennis/kunde over eindige automaten summatief getoetst?

Heeft de kennis/kunde van jouw leerlingen invloed op hun cijfers voor informatica? Dan ben je die kennis/kunde summatief aan het toetsen. Dat gebeurt ook als het een klein onderdeel vormt van een grotere theoretische toets of praktische opdracht. Kies dus alleen de laatste optie als de kennis/kunde over eindige automaten echt nooit voorbij komt in werk dat voor een cijfer wordt gemaakt.

- ☐ Met een theoretische toets, die leerlingen op papier maken.
- ☐ Met een theoretische toets, die leerlingen met de computer maken.
- ☐ Met een praktische toets, met als eindproduct (onder andere) een fysiek product.

- ☐ Met een praktische toets, met als eindproduct (onder andere) een digitaal product.
- ☐ Met een presentatie door leerling(en).
- ☐ Met een mondelinge overhoring.
- ☐ Dit onderwerp wordt op geen enkele manier getoetst.
- ☐ Anders:

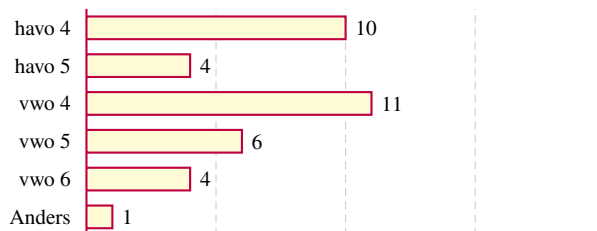


Plaats in het curriculum

De **plaats in het curriculum** wordt vaak bewust gekozen. In de eerste plaats natuurlijk omdat je nu eenmaal gebonden bent aan een jaarrooster. Maar vaak kies je er ook bewust voor om eerst het ene, en dan het andere onderwerp te behandelen.

15. In welk(e) leerjaar/leerjaren behandel je eindige automaten?

- ☐ havo 4 ☐ vwo 4 ☐ vwo 6
- ☐ havo 5 ☐ vwo 5 ☐ Anders:



Anders:

- „als onderdeel van een module, op een moment dat leerlingen daarvoor kiezen. Vaak eind klas 3 of begin klas 4, maar soms later.”

16. Welke subdomeinen behandel je bewust vóór de lessen over eindige automaten?

Bekijk eventueel **het examenprogramma informatica** voor de beschrijvingen bij ieder subdomein.

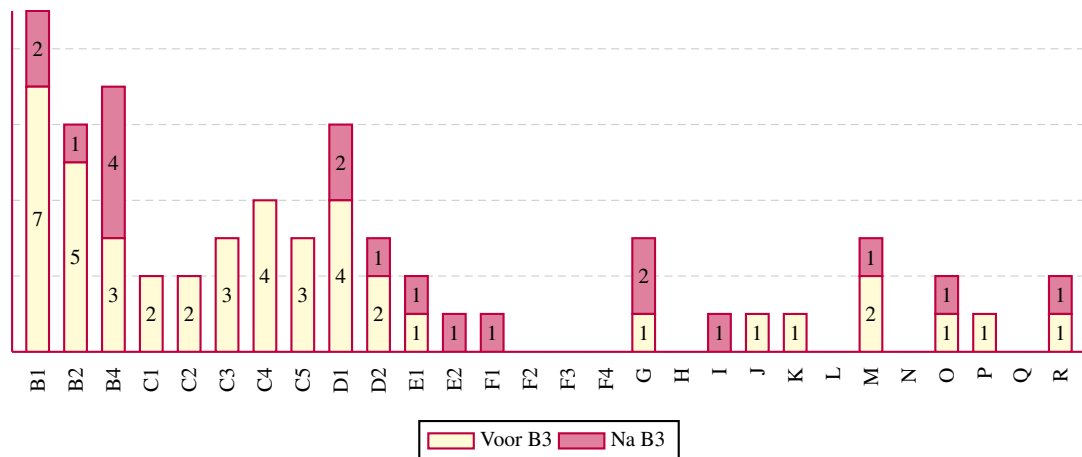
- | | |
|--|---|
| ☐ B1 Algoritmen | ☐ C2 Identificeren |
| ☐ B2 Datastructuren | ☐ C3 Representeren |
| ☐ B4 Grammatica's | ☐ C4 Standaardrepresentaties |
| ☐ C1 Doelstellingen | ☐ C5 Gestructureerde data |

- | | |
|--|--|
| ☐ D1 Ontwikkelen | ☐ I Cognitive computing |
| ☐ D2 Inspecteren en aanpassen | ☐ J Programmeerparadigma's |
| ☐ E1 Decompositie | ☐ K Computerarchitectuur |
| ☐ E2 Security | ☐ L Netwerken |
| ☐ F1 Usability | ☐ M Physical computing |
| ☐ F2 Maatschappelijke aspecten | ☐ N Security |
| ☐ F3 Privacy | ☐ O Usability |
| ☐ F4 Security | ☐ P User experience |
| ☐ G Algoritmië, berekenbaarheid en logica | ☐ Q Maatschappelijke en individuele invloed van informatica |
| ☐ H Databases | ☐ R Computational science |

17. Welke subdomeinen behandel je bewust ná de lessen over eindige automaten?

Bekijk eventueel **het examenprogramma informatica** voor de beschrijvingen bij ieder subdomein.

- | | |
|---|--|
| ☐ B1 Algoritmen | ☐ F4 Security |
| ☐ B2 Datastructuren | ☐ G Algoritmië, berekenbaarheid en logica |
| ☐ B4 Grammatica's | ☐ H Databases |
| ☐ C1 Doelstellingen | ☐ I Cognitive computing |
| ☐ C2 Identificeren | ☐ J Programmeerparadigma's |
| ☐ C3 Representeren | ☐ K Computerarchitectuur |
| ☐ C4 Standaardrepresentaties | ☐ L Netwerken |
| ☐ C5 Gestructureerde data | ☐ M Physical computing |
| ☐ D1 Ontwikkelen | ☐ N Security |
| ☐ D2 Inspecteren en aanpassen | ☐ O Usability |
| ☐ E1 Decompositie | ☐ P User experience |
| ☐ E2 Security | ☐ Q Maatschappelijke en individuele invloed van informatica |
| ☐ F1 Usability | ☐ R Computational science |
| ☐ F2 Maatschappelijke aspecten | |
| ☐ F3 Privacy | |



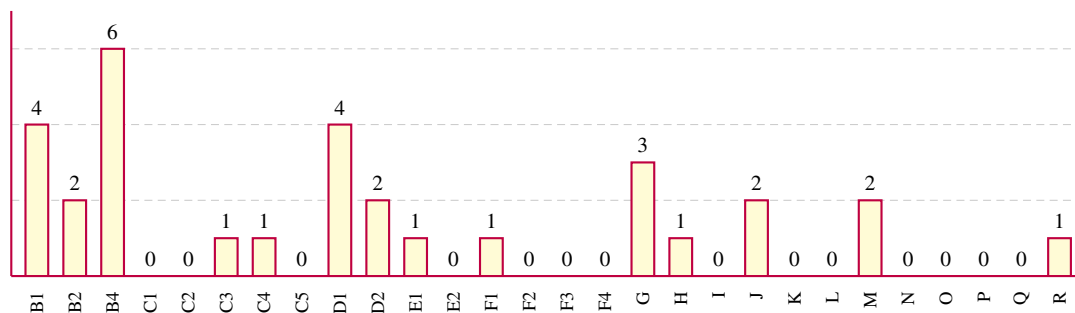
Inbedding

De term **inbedding** houdt in dat een onderwerp onderdeel is van een lessenserie over één of meerdere grotere onderwerpen. Je kunt bijvoorbeeld een project hebben waar leerlingen programmeren in HTML, PHP en MySQL om een website te maken. Je behandelt dan de eindtermen D (programmeren) en C5 (databases).

18. Welke subdomeinen behandel je bewust **tegelijk met** de lessen over eindige automaten?

Bekijk eventueel **het examenprogramma informatica** voor de beschrijvingen bij ieder subdomein.

- | | |
|---|--|
| ☐ B1 Algoritmen | ☐ F4 Security |
| ☐ B2 Datastructuren | ☐ G Algoritmie, berekenbaarheid en logica |
| ☐ B4 Grammatica's | ☐ H Databases |
| ☐ C1 Doelstellingen | ☐ I Cognitive computing |
| ☐ C2 Identificeren | ☐ J Programmeerparadigma's |
| ☐ C3 Representeren | ☐ K Computerarchitectuur |
| ☐ C4 Standaardrepresentaties | ☐ L Netwerken |
| ☐ C5 Gestructureerde data | ☐ M Physical computing |
| ☐ D1 Ontwikkelen | ☐ N Security |
| ☐ D2 Inspecteren en aanpassen | ☐ O Usability |
| ☐ E1 Decompositie | ☐ P User experience |
| ☐ E2 Security | ☐ Q Maatschappelijke en individuele invloed van informatica |
| ☐ F1 Usability | ☐ R Computational science |
| ☐ F2 Maatschappelijke aspecten | |
| ☐ F3 Privacy | |



Grammatica's — behandel je dit onderwerp?

19. Behandel je het subdomein 'eindige automaten' (B3) in welke vorm dan ook in de informatica-lessen?

Het gaat hier om elke vorm van behandeling. Dus ook als je het onderwerp technisch gezien wel behandelt, maar bijvoorbeeld nooit bij naam noemt. Juist in zulke invullingen zijn we enorm geïnteresseerd!

Vul dus alleen 'nee' in als je echt helemaal niets met dit onderwerp doet in jouw informatica-lessen.

- o Ja (Ga door naar **vraag 20**)
- o Nee (Ga door naar **vraag 34**)



Grammatica's

Je geeft les over grammatica's. In mijn onderzoek wil ik erachter komen op welke manieren dat allemaal kan. Ik kijk daarbij naar de aspecten inhoud, omvang, toetsing, plaats in het curriculum en inbedding.

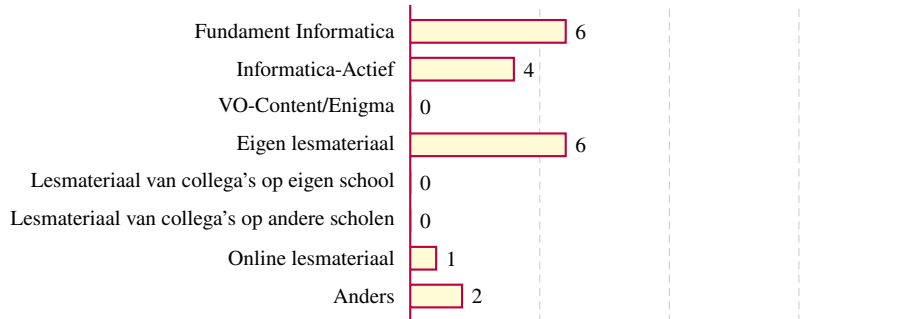
Inhoud

Met de **inhoud** bedoel ik welke concrete onderwerpen je behandelt. De formele term hiervoor is context. Je zou bijvoorbeeld domein D (programmeren) kunnen onderwijzen door leerlingen een game te laten programmeren met Pygame. Dat is dan de inhoud.

20. Welke lesmethode(n) gebruik je voor je lessen over eindige automaten?

- ☐ Fundament Informatica
- ☐ Informatica-Actief
- ☐ VO-Content/Enigma
- ☐ Eigen lesmateriaal
- ☐ Lesmateriaal van collega's op eigen school

- ☐ Lesmateriaal van collega's op andere scholen
- ☐ Online lesmateriaal
- ☐ Anders:

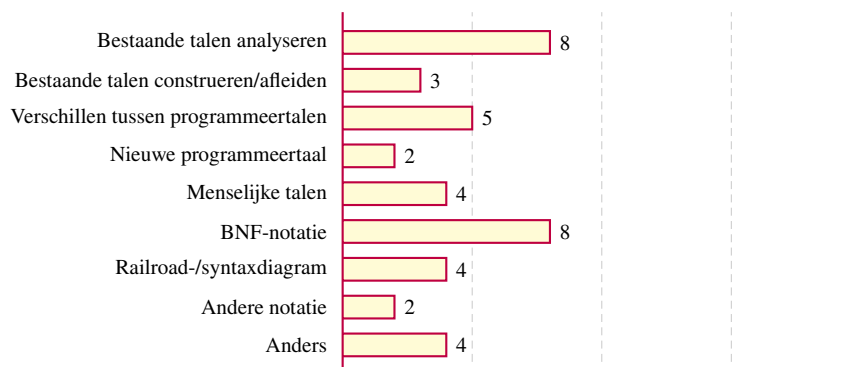


Anders:

- „De Taalkist. over Taal en computers”
- „Informatica voor het VO”

21. Welke onderwerpen gebruik je om het subdomein grammatica's te behandelen?

- ☐ Grammatica's van bestaande programmeertalen analyseren
- ☐ Grammatica's van bestaande programmeertalen construeren/afleiden
- ☐ Verschillen tussen programmeertalen bekijken
- ☐ Grammatica van een nieuwe programmeertaal verzinnen
- ☐ Grammatica's van 'menselijke' talen, zoals Nederlands of Esperanto
- ☐ Grammatica's bekijken/construeren in BNF-notatie
- ☐ Grammatica's bekijken/construeren als railroad-/syntaxdiagram
- ☐ Grammatica's in een andere notatie bekijken/construeren
- ☐ Anders:



Anders:

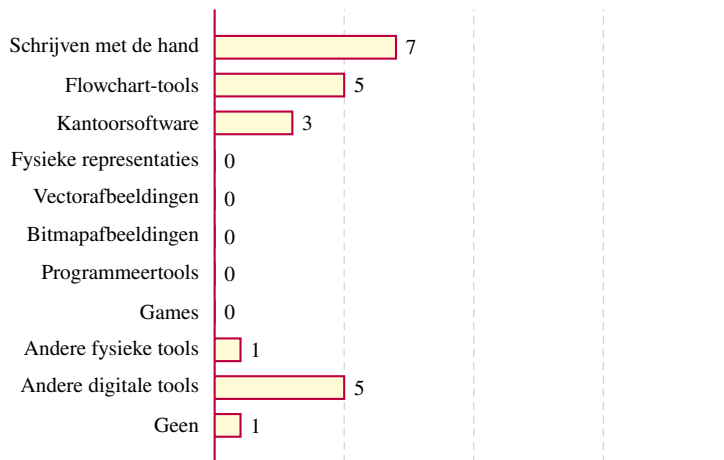
- „hierarchie van grammatica's”
- „theorie soorten programmeertalen, syntax, soorten fouten (combi met programmeertalen die we gebruiken)”

- „morsecode, terug naar schakelingen, python kernwoorden toelichten”
- „Ik volg de methode” (Fundament Informatica)

22. Welke tools gebruik je/gebruiken je leerlingen om grammatica's te noteren/construeren?

Probeer zo specifiek mogelijk te zijn. Kies alleen de laatste twee antwoorden als jouw tools echt niet onder de overige categorieën vallen.

- ☐ Schrijven/tekenen met de hand, zoals met pen/potlood/stift en papier/whiteboard/krijtbord
- ☐ Digitale diagram-, flowchart- en mindmap-tools, zoals Draw.io, Lucidchart of Visio
- ☐ Kantoorsoftware, zoals PowerPoint, LibreOffice Present of Google Slides
- ☐ Fysieke representaties, zoals bouwstenen en knutselmateriaal
- ☐ Grafische ontwerpsoftware voor vectorafbeeldingen, zoals Inkscape of Adobe Illustrator
- ☐ Grafische ontwerpsoftware voor bitmapafbeeldingen, zoals GIMP, Paint.net of Photoshop
- ☐ Programmeertools voor grammatica's, zoals ANTLR of Bison
- ☐ Games, zoals Minecraft of Roblox
- ☐ Andere fysieke tools
- ☐ Andere digitale tools
- ☐ Mijn leerlingen noteren/construeren zelf geen grammatica's



Omvang

De **omvang** is de tijdsduur van het onderwerp.

23. Hoeveel SLU besteed je aan grammatica's?

Een SLU is één studielastuur van 60 minuten. Het gaat hier om alle studielasturen die leerlingen besteden aan eindige automaten. Weet je het niet zeker, geef dan een inschatting.

.....

24. Wat vind je van deze hoeveelheid SLU ten opzichte van de leerdoelen?

Het gaat hier om de relatie tussen de leerdoelen en de tijd die je besteed aan het behalen van de leerdoelen.

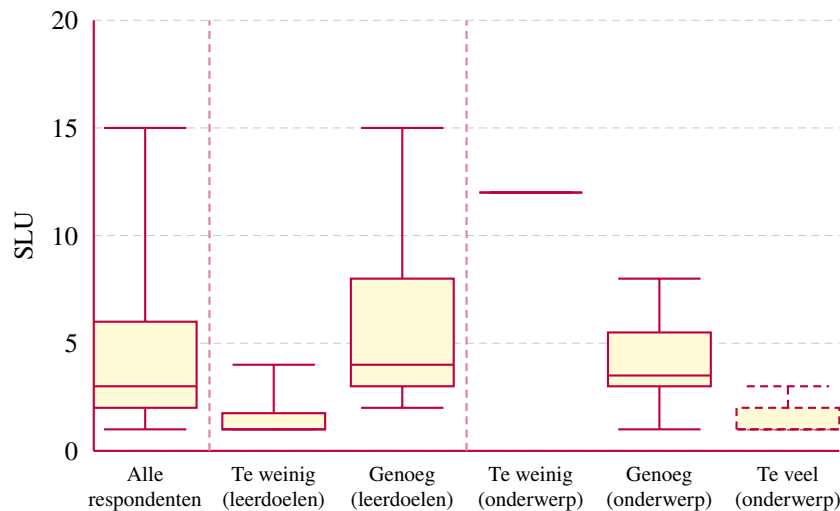
- Te weinig
- Genoeg
- Te veel

25. Wat vind je van deze hoeveelheid SLU ten opzichte van het onderwerp?

Het gaat hier om de relatie tussen jouw eigen enthousiasme/passie over het onderwerp en de tijd die je besteed aan het onderwerp. Vind je het onderwerp totaal niet interessant, dan is je antwoord waarschijnlijk 'te veel'. Vind je het onderwerp enorm interessant, en zou je er meer tijd aan willen besteden, dan is je antwoord waarschijnlijk 'te weinig'.

- Te weinig
- Genoeg
- Te veel

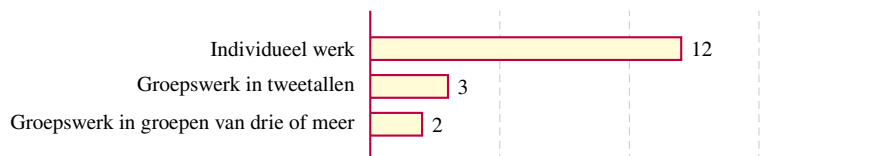
De antwoorden op bovenstaande vragen zijn hieronder in boxplots en in een tabel weergegeven.



Groep	n	Min	Q1	Mediaan	Q3	Max	SD
Alle respondenten	13	1.0	2.0	3.0	6.0	15.0	4.38
Te weinig (leerdoelen)	4	1.0	1.0	1.0	1.75	4.0	1.50
Genoeg (leerdoelen)	9	2.0	3.0	4.0	8.0	15.0	4.58
Te veel (leerdoelen)	0	–	–	–	–	–	–
Te weinig (onderwerp)	1	12.0	12.0	12.0	12.0	12.0	–
Genoeg (onderwerp)	6	1.0	3.0	3.5	5.5	8.0	2.74
Te veel (onderwerp)	3	1.0	1.0	1.0	2.0	3.0	1.15

26. Wat voor werk verrichten leerlingen voor het onderwerp grammatica's?

- ☐ Individueel werk
- ☐ Groepswerk in tweetallen
- ☐ Groepswerk in groepen van drie of meer

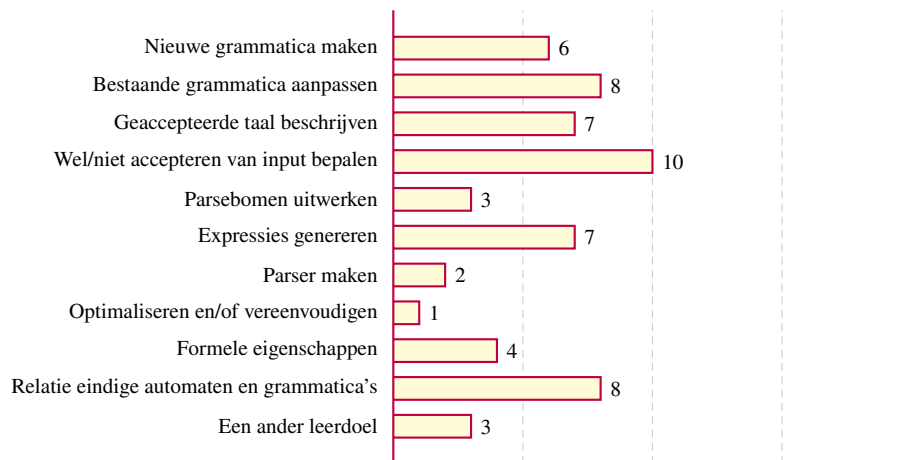


Toetsing

Toetsing houdt niet alleen in hoe je de eindterm toetst, maar ook wat je toetst. Het gaat hier dus om toetsvormen en leerdoelen.

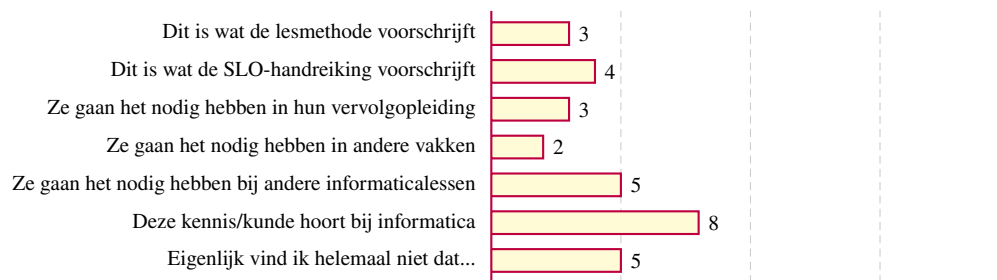
27. Wat moeten leerlingen kunnen aan het einde van de lessen over grammatica's?

- ☐ Op basis van een beschrijving/opdracht een nieuwe grammatica maken
- ☐ Op basis van een beschrijving/opdracht een bestaande grammatica aanpassen
- ☐ De geaccepteerde taal van een grammatica in woorden omschrijven
- ☐ Bepalen of een grammatica een bepaalde input wel/niet accepteert
- ☐ Derivatiestappen/parsebomen van een grammatica uitwerken op basis van een bepaalde input
- ☐ Zinnen/expressies genereren met een grammatica
- ☐ Een parser maken voor een grammatica
- ☐ Een grammatica optimaliseren en/of vereenvoudigen
- ☐ De formele eigenschappen (productieregel, alfabet, etc) kennen en benoemen
- ☐ De relatie tussen eindige automaten en grammatica's uitleggen
- ☐ Een ander leerdoel dat hier niet is genoemd



28. Waarom vind je dat je leerlingen deze leerdoelen moeten beheersen?

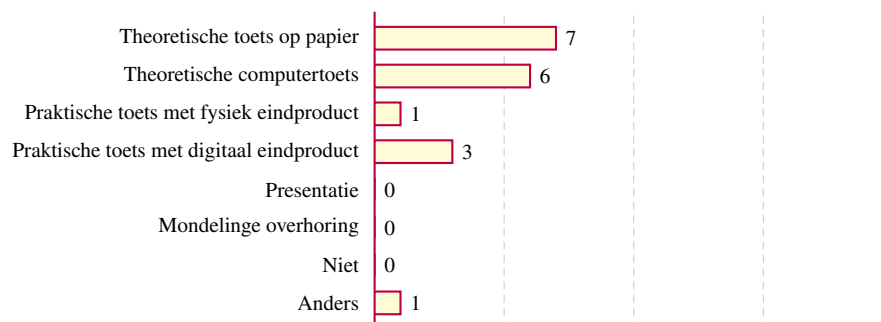
- ☐ Dit is wat de lesmethode voorschrijft
- ☐ Dit is wat de SLO-handreiking voorschrijft
- ☐ Ze gaan het nodig hebben in hun vervolgopleiding
- ☐ Ze gaan het nodig hebben in andere vakken
- ☐ Ze gaan het nodig hebben bij andere informaticalessen
- ☐ Deze kennis/kunde hoort bij informatica
- ☐ Eigenlijk vind ik helemaal niet dat mijn leerlingen deze leerdoelen moeten beheersen.



29. Hoe wordt de kennis/kunde over grammatica's summatief getoetst?

Heeft de kennis/kunde van jouw leerlingen invloed op hun cijfers voor informatica? Dan ben je die kennis/kunde summatief aan het toetsen. Dat gebeurt ook als het een klein onderdeel vormt van een grotere theoretische toets of praktische opdracht. Kies dus alleen de laatste optie als de kennis/kunde over eindige automaten echt nooit voorbij komt in werk dat voor een cijfer wordt gemaakt.

- ☐ Met een theoretische toets, die leerlingen op papier maken.
- ☐ Met een theoretische toets, die leerlingen met de computer maken.
- ☐ Met een praktische toets, met als eindproduct (onder andere) een fysiek product.
- ☐ Met een praktische toets, met als eindproduct (onder andere) een digitaal product.
- ☐ Met een presentatie door leerling(en).
- ☐ Met een mondelinge overhoring.
- ☐ Dit onderwerp wordt op geen enkele manier getoetst.
- ☐ Anders:



Anders:

- „gebruik maken van de Nearley Parser Playground”¹

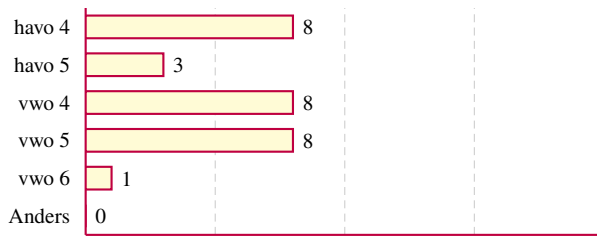
Plaats in het curriculum

De **plaats in het curriculum** wordt vaak bewust gekozen. In de eerste plaats natuurlijk omdat je nu eenmaal gebonden bent aan een jaarrooster. Maar vaak kies je er ook bewust voor om eerst het ene, en dan het andere onderwerp te behandelen.

30. In welk(e) leerjaar/leerjaren behandel je grammatica's?

¹<https://omrelli.ug/nearley-playground/>

- ☐ havo 4 ☐ vwo 4 ☐ vwo 6
☐ havo 5 ☐ vwo 5 ☐ Anders:



31. Welke subdomeinen behandel je bewust vóór de lessen over grammatica's?

Bekijk eventueel **het examenprogramma informatica** voor de beschrijvingen bij ieder subdomein.

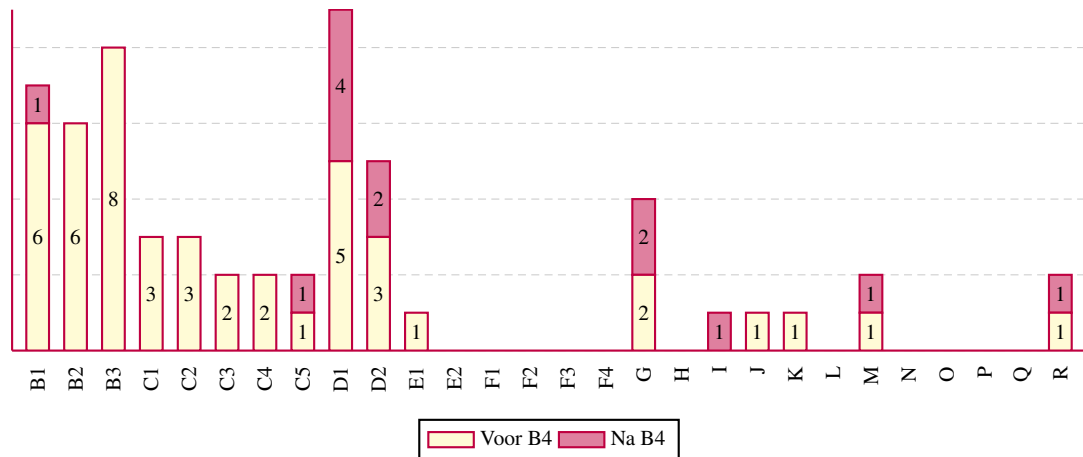
- | | |
|---|--|
| ☐ B1 Algoritmen | ☐ F4 Security |
| ☐ B2 Datastructuren | ☐ G Algoritmiek, berekenbaarheid en logica |
| ☐ B3 Eindige automaten | ☐ H Databases |
| ☐ C1 Doelstellingen | ☐ I Cognitive computing |
| ☐ C2 Identificeren | ☐ J Programmeerparadigma's |
| ☐ C3 Representeren | ☐ K Computerarchitectuur |
| ☐ C4 Standaardrepresentaties | ☐ L Netwerken |
| ☐ C5 Gestructureerde data | ☐ M Physical computing |
| ☐ D1 Ontwikkelen | ☐ N Security |
| ☐ D2 Inspecteren en aanpassen | ☐ O Usability |
| ☐ E1 Decompositie | ☐ P User experience |
| ☐ E2 Security | ☐ Q Maatschappelijke en individuele invloed van informatica |
| ☐ F1 Usability | ☐ R Computational science |
| ☐ F2 Maatschappelijke aspecten | |
| ☐ F3 Privacy | |

32. Welke subdomeinen behandel je bewust ná de lessen over grammatica's?

Bekijk eventueel **het examenprogramma informatica** voor de beschrijvingen bij ieder subdomein.

- | | |
|---|--|
| ☐ B1 Algoritmen | ☐ C3 Representeren |
| ☐ B2 Datastructuren | ☐ C4 Standaardrepresentaties |
| ☐ B3 Eindige automaten | ☐ C5 Gestructureerde data |
| ☐ C1 Doelstellingen | ☐ D1 Ontwikkelen |
| ☐ C2 Identificeren | ☐ D2 Inspecteren en aanpassen |

- | | |
|---|--|
| ☐ E1 Decompositie | ☐ J Programmeerparadigma's |
| ☐ E2 Security | ☐ K Computerarchitectuur |
| ☐ F1 Usability | ☐ L Netwerken |
| ☐ F2 Maatschappelijke aspecten | ☐ M Physical computing |
| ☐ F3 Privacy | ☐ N Security |
| ☐ F4 Security | ☐ O Usability |
| ☐ G Algoritmiëk, berekenbaarheid en logica | ☐ P User experience |
| ☐ H Databases | ☐ Q Maatschappelijke en individuele invloed van informatica |
| ☐ I Cognitive computing | ☐ R Computational science |



Inbedding

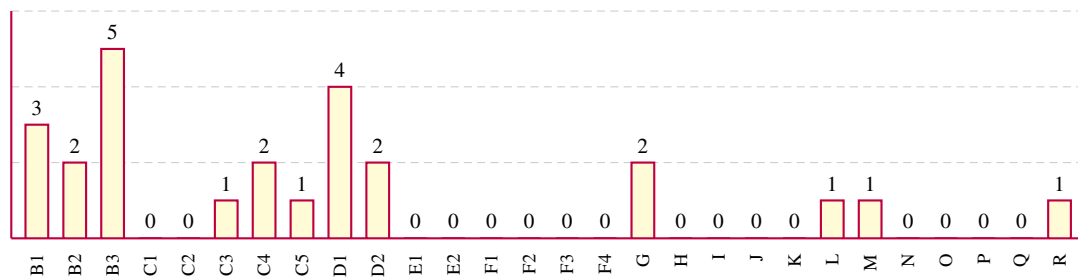
De term **inbedding** houdt in dat een onderwerp onderdeel is van een lessenserie over één of meerdere grotere onderwerpen. Je kunt bijvoorbeeld een project hebben waar leerlingen programmeren in HTML, PHP en MySQL om een website te maken. Je behandelt dan de eindtermen D (programmeren) en C5 (databases).

33. Welke subdomeinen behandel je bewust **tegelijk met** de lessen over grammatica's?

Bekijk eventueel **het examenprogramma informatica** voor de beschrijvingen bij ieder subdomein.

- | | |
|---|---|
| ☐ B1 Algoritmen | ☐ C5 Gestructureerde data |
| ☐ B2 Datastructuren | ☐ D1 Ontwikkelen |
| ☐ B3 Eindige automaten | ☐ D2 Inspecteren en aanpassen |
| ☐ C1 Doelstellingen | ☐ E1 Decompositie |
| ☐ C2 Identificeren | ☐ E2 Security |
| ☐ C3 Representeren | ☐ F1 Usability |
| ☐ C4 Standaardrepresentaties | ☐ F2 Maatschappelijke aspecten |

- ☐ F3 Privacy
- ☐ F4 Security
- ☐ G Algoritmiëk, berekenbaarheid en logica
- ☐ H Databases
- ☐ I Cognitive computing
- ☐ J Programmeerparadigma's
- ☐ K Computerarchitectuur
- ☐ L Netwerken
- ☐ M Physical computing
- ☐ N Security
- ☐ O Usability
- ☐ P User experience
- ☐ Q Maatschappelijke en individuele invloed van informatica
- ☐ R Computational science

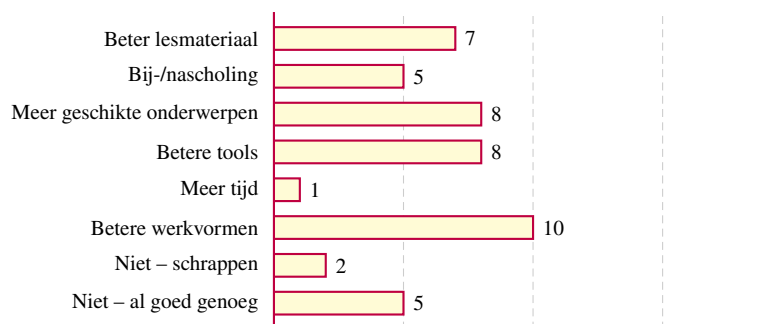


Wensen voor de toekomst

De onderwerpen 'eindige automaten' en 'grammatica's' zijn bron van hevige discussie. Eén ding is duidelijk: er is ruimte voor verbetering. Graag vragen we je hoe jij hierin staat!

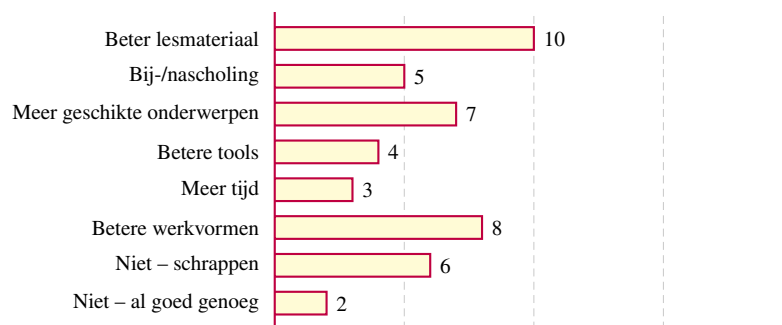
34. Hoe denk je dat lessen over eindige automaten nog beter zouden kunnen worden?

- ☐ Er moet beter lesmateriaal voor worden ontwikkeld
- ☐ Er moet bij-/nascholing komen voor docenten
- ☐ Er moeten onderwerpen komen die beter geschikt zijn voor leerlingen
- ☐ Er moeten betere tools komen om met eindige automaten te werken
- ☐ Er moet meer tijd voor het onderwerp beschikbaar komen
- ☐ Er moeten betere werkvormen worden ontwikkeld
- ☐ Niet; het onderwerp moet worden geschrapt uit het examenprogramma
- ☐ Niet; het is al goed genoeg.



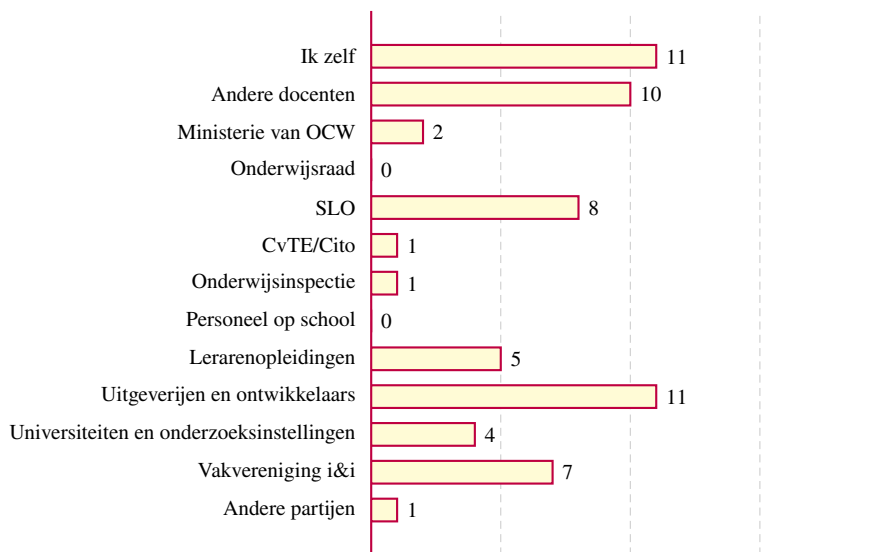
35. Hoe denk je dat lessen over grammatica's nog beter zouden kunnen worden?

- ☐ Er moet beter lesmateriaal voor worden ontwikkeld
- ☐ Er moet bij-/nascholing komen voor docenten
- ☐ Er moeten onderwerpen komen die beter geschikt zijn voor leerlingen
- ☐ Er moeten betere tools komen om met grammatica's te werken
- ☐ Er moet meer tijd voor het onderwerp beschikbaar komen
- ☐ Er moeten betere werkvormen worden ontwikkeld
- ☐ Niet; het onderwerp moet worden geschrapt uit het examenprogramma
- ☐ Niet; het is al goed genoeg.



36. Welke partijen zouden hier een bijdrage aan kunnen leveren?

- ☐ Ik zelf (dus degene die deze lijst invult)
- ☐ Andere informaticadocenten
- ☐ Het ministerie van OCW
- ☐ De onderwijsraad
- ☐ De SLO
- ☐ Het CvTE/Cito
- ☐ De onderwijsinspectie
- ☐ Personeel op school anders dan de docenten, zoals teamleiders
- ☐ Lerarenopleidingen
- ☐ Educatieve uitgeverijen en ontwikkelaars van lesmateriaal
- ☐ Universiteiten en onderzoeksinstituten
- ☐ Vakvereniging i&i
- ☐ Andere onderwijsondersteunende partijen die niet hierboven zijn genoemd



37. Wil je nog iets anders kwijt, wat ik nog niet heb gevraagd, maar waarvan je wel denkt dat het relevant is voor dit onderzoek?

.....

.....

.....

- Ik heb wel lesmateriaal voor B4, maar heb die nog niet kunnen gebruiken omdat ik mijn jaarplanning aan het aanpassen was. Ik heb het materiaal van informatica actief.
- Delen van met name grammatica's, komen meer tussendoor aan bod, bv. ergens tijdens het programmeren. Het aantal uren dat ik puur gericht aan grammatica's besteed en niet zijdelings naast het behandelen van programmeertalen, is beperkt. Het deel dat tussendoor komt, durf ik niet zo in te schatten. Ik mis de optie 'mondeling bespreken' (klassikaal en één op één of een paar, al dan niet met whiteboard erbij) als oefen-/verwerkingsvorm.
- crash course computer science. of een leuke oefening die goed is uitgewerkt
- 1) Wat is de invloed van de aangeboden context op of docenten deze onderwerpen wel of niet interessant vinden om te behandelen?
2) Kan het helpen om B4 te verweven in lesmateriaal over D om B4 zo heel praktisch toe te passen met het doel dat leerlingen de syntax van de aangeboden programmeertaal goed aanleren?
3) Ik begrijp de weerstand tegen B3 van docenten niet: in de context van apparaten (e.g., CV-ketel) is het heel relevant voor keuzethema M (en wordt ook bij natuurkunde ook op een andere manier behandeld bij het onderwerp "technische automatisering"), en je kunt het ook toepassen bij game design (e.g., gedrag van enemies, wanneer welke animatie moet afspelen, etc.) wat absoluut tot de verbeelding van leerlingen spreekt.
- Ik gebruik Fundament als basis. Dan weet ik zeker dat ik aan de norm heb voldaan. Maar grammatica's behandel ik niet separaat. Natuurlijk hebben we het uitgebreid over de terminologie voordat we gaan programmeren, maar niet via B4. En elke keer als de lln met een nieuwe taal in aanraking komt staan we even stil bij vreschillen en overeenkomsten.
- Het probleem met beide onderwerpen is dat de uitwerking tot nu toe te abstract en te academisch uitgewerkt is. Het sluit niet of nauwelijks aan op de belevingswereld van de

leerlingen omdat het door academici is uitgewerkt en vormgegeven. Leerlingen zien in grammatica niet eens dat hier de basis van een AI ligt. Ook jammer dat de verbinding met taalkunde en de Generatieve Grammatica van Noam Chomsky niet gelegd wordt als deels een parallelle ontwikkeling naast BNF.

Bedankt!

Je bent aan het eind van deze vragenlijst gekomen. Allereerst wil ik je ontzettend bedanken voor het invullen van de vragenlijst. Je input is enorm waardevol en betekent veel voor dit onderzoek!

Let op: je antwoorden zijn nog niet opgeslagen. Klik onderaan op 'Verzenden' om je antwoorden op te slaan.

Verzenden

C Interviewvragen

Introductie

1. Behandel je het subdomein ‘eindige automaten’ in de informaticalessen?

Indien nee:

- (a) Waarom niet?
 - (b) Wat zou er moeten veranderen, zodat je het onderwerp wél gaat behandelen?
 - (c) Welke partijen spelen hierin volgens jou een belangrijke rol?
2. Behandel je het subdomein ‘grammatica’s’ in de informaticalessen?

Indien nee:

- (a) Waarom niet?
- (b) Wat zou er moeten veranderen, zodat je het onderwerp wél gaat behandelen?
- (c) Welke partijen spelen hierin volgens jou een belangrijke rol?

Eindige automaten

Alleen behandelen als de geïnterviewde het onderwerp behandelt in de lessen. Doel van het interview is om antwoord te krijgen op alle onderstaande (deel)vragen, maar niet om de vragen op deze volgorde te stellen. Het natuurlijke verloop van het gesprek is belangrijker. De vragen hoeven niet chronologisch of letterlijk te worden behandeld.

3. Wat doen jouw leerlingen met eindige automaten?
4. Welke tool(s) gebruik je om eindige automaten te tekenen? Gebruiken je leerlingen andere tools? Waarom?
5. Wat vind je belangrijker: de theoretische kennis over eindige automaten, of de praktische inzetbaarheid als hulpmiddel bij bijvoorbeeld programmeren? Waarom?
6. Hoe wordt de kennis en kunde over eindige automaten getoetst? Waarom?
7. Welke leerdoelen toets je daarbij? Waarom?
8. Welke onderwerpen behandel je bewust voor, na, of tegelijk met eindige automaten? Waarom?
9. Wat zou je in de toekomst willen veranderen aan jouw lessen over eindige automaten? Waarom?

10. Welke partijen spelen hierin volgens jou een belangrijke rol? Wat kunnen ze hierin bijdragen?

Grammatica's

Alleen behandelen als de geïnterviewde het onderwerp behandelt in de lessen. Doel van het interview is om antwoord te krijgen op alle onderstaande (deel)vragen, maar niet om de vragen op deze volgorde te stellen. Het natuurlijke verloop van het gesprek is belangrijker. De vragen hoeven niet chronologisch of letterlijk te worden behandeld.

11. Wat doen jouw leerlingen met grammatica's?
12. Ontwikkelen de leerlingen eigen grammatica's, of maken ze alleen gebruik van bestaande grammatica's? Waarom?
13. Wat voor grammatica's behandel je eigenlijk? Gebruik je bijvoorbeeld BNF, of werken leerlingen met Antlr? Waarom?
14. Hoe wordt de kennis en kunde over grammatica's getoetst? Waarom?
15. Welke leerdoelen toets je daarbij? Waarom?
16. Welke onderwerpen behandel je bewust voor, na, of tegelijk met grammatica's? Waarom?
17. Wat zou je in de toekomst willen veranderen aan jouw lessen over grammatica's? Waarom?
18. Welke partijen spelen hierin volgens jou een belangrijke rol? Wat kunnen ze hierin bijdragen?

Overig en afsluiting

19. Wil je nog iets anders kwijt wat ik nog niet heb gevraagd, maar waarvan je wel denkt dat het relevant is voor dit onderzoek?

D Interviewtranscripties

Veranderingen t.b.v. anonimiteit van de respondenten

Delen van de interviews die de identiteit van de respondent zouden kunnen blootstellen zijn verwijderd. Deze verwijderingen zijn gemarkeerd met (...). Op sommige plekken hebben de bewoording van de respondent minimaal aangepast om de anonimiteit te waarborgen. De respondent zei dan bijvoorbeeld „Voorbeeldschool Talentklas”, maar zo’n term zou herleidbaar zijn naar die school, en daarmee de docent. Zulke veranderingen zijn gemarkeerd met blokhaken: [aparte klas voor leerlingen met extra talent].

D.1 Interview 1

Interviewer: Ik heb je uitgekozen om twee redenen: (1) je geeft aan niets te doen met grammatica’s in je curriculum en (2) je behandelt eindige automaten in een losse module. Voordat we daarin duiken, wil ik naar het onderdeel grammatica’s. Waarom geef je dat niet?

Respondent: We konden geen goede plek vinden om het logisch in te bouwen. Het onderwerp kan niet op zichzelf een module vormen. We gaan niet een hele module grammatica’s doen. Daar krijgen we niemand voor mee. Waar ga je het dan bij in zetten? We hebben er geen geschikte plek voor kunnen vinden, waarvan we ook vinden: we kunnen onze leerlingen hier wat mee bijbrengen. Als we het ergens bij inschuiven, blijft het niet hangen, dat heeft geen zin.

Interviewer: Hoe los je dat op? Je hebt dan geen sluitend PTA.

Respondent: Dat klopt.

Interviewer: Stop je het dan bij een andere module, en hoop je dat de inspectie niet opvalt?

Respondent: Ik ben niet degene die over die tabellen gaat. Volgens mij gaat het inderdaad op die manier. Het zou me niet verbazen als het subdomein is genoteerd als onderdeel van deze module.

Interviewer: Als ik het goed begrijp: het is moeilijk om het onderwerp op zichzelf te geven, want het is niet boeiend genoeg, en je kunt het niet goed koppelen aan iets anders.

Respondent: Dat hebben wij nog niet gevonden. Misschien ga jij iets in je onderzoek vinden, dat zou heel leuk zijn!

Interviewer: Voor de hand liggend is het natuurlijk om het te koppelen aan eindige automaten in de bestaande module. Dat is wel vrij harde computer science.

Respondent: Dat is hele harde computer science.

Interviewer: Iemand in havo 4 met cultuur en maatschappij heeft daar geen zin in.

Respondent: Dat klopt. Onze [module] wordt veel gekozen door leerlingen die net beginnen. Dat doen ze vaak ook als eerste module. Dat zijn inderdaad havo 4 en vwo 4-leerlingen. Vaak ook derdeklassers die denken: dat is wat voor mij, en dan beginnen ze er halverwege of aan het eind van de derde klas al mee. Om ze dan met grammatica's te laten werken gaat ver.

Interviewer: Je gaf het zelf ook aan in de enquête: er moeten betere onderwerpen en werkvormen komen, en beter lesmateriaal. Ik heb in het onderzoek gezien dat de auteurs van het examenprogramma ook zeker geen harde theorie wilden zien bij eindige automaten en grammatica's, maar dat je ze praktisch inzet om digitale artefacten te maken. Jij gaat bijvoorbeeld een programma schrijven, en van tevoren gebruik je een eindige automaat of een grammatica om na te denken over de structurering. Met een robot gebruik je dan een toestandsdiagram om het gedrag van de robot te modelleren.

Respondent: De grammatica is dan een beschrijving van valide stappen die hij kan nemen?

Interviewer: Ja, en volgens mij ook om te kijken hoe een programmeertaal werkt. Ik vind de beschrijving vaag.

Respondent: Precies. En als je er dan probeert een les mee te maken, dan moet je er wel even over nadenken. Een van de opdrachten die we wel doen voor eindige automaten is de schateilandopdracht van CS Unplugged. Je kunt dan wel de link maken naar grammatica, maar dan zit je het er wel in te buigen. Dat heeft dan weinig toegevoegde waarde.

Interviewer: Denk je dat het überhaupt toegevoegde waarde heeft om grammatica's te behandelen op de middelbare school?

Respondent: Nee, ik denk het niet. Ik denk niet dat er een manier is. Zeker niet in een kerndomein. Misschien is er ruimte voor een keuzethema 'harde computer science'. Dat hebben we met algoritmiek al een klein beetje. Daar past het beter bij dan als basisdomein.

Interviewer: Differentiatie is ook lastig met grammatica's. Het is moeilijk om te vereenvoudigen.

Respondent: Ja.

Interviewer: Laten we de eindige automaten bekijken. Ik ben benieuwd hoe [de module] eruitziet.

(...)

Interviewer: Wat gaan ze precies doen met eindige automaten tijdens de module?

Respondent: De module steekt in elkaar. We proberen de basisprincipes in vogelvlucht langs te gaan. Het gaat over decompositie. Eerst de fysieke hardware. Daarna: wat kunnen we met die hardware doen? Dat is waar de automaten voorbij komen. Vervolgens kijken we naar hoe de hardware wordt gebruikt voor een stukje usability.

Interviewer: Je gaf ook aan dat je eindige automaten gebruikt om het gedrag van bestaande systemen te beschrijven.

Respondent: Voor de eindopdracht kiezen ze zelf een systeem, zoals een magnetron of een wasmachine. Ze gaan uitzoeken welke onderdelen er in het systeem zitten. Voor de automaten kijken ze dan: wat zijn de toestanden van dit apparaat? Hoe ga je van de ene naar de ander? En daarna maken ze de eindige automaat. Het ingewikkeldste is dat een apparaat vaak erg ingewikkeld is, dus ze moeten de toestanden wat samenvatten. We hebben één les van 90 minuten die we besteden aan 'dit is hoe dit werkt'. We doen dan de schateiland-opdracht. En online heb ik wat nepapparaten geknutseld, een eenvoudig en een ingewikkelder, om te oefenen hoe je eindige automaten tekent. Die bespreken we tijdens de les. Op het eind behandel ik kort de Turingmachine om te laten zien waar de computer science voor wordt gebruikt. Dat is ter vermaak.

Interviewer: Welke tools gebruik je?

Respondent: Tot nu toe raadde ik draw.io aan. Tegenwoordig gebruiken we Visio. Dat zit direct in Teams ingebouwd. Dat werkt best goed, ook de samenwerking tussen leerlingen. Er is minder setup. Meer dan cirkels en pijltjes heb je niet nodig, dus je bent gauw klaar.

Interviewer: Je toetst het onderwerp door ze bij een nieuw apparaat een eindige automaat te laten maken. Waarom?

Respondent: Omdat het een geheel vormt met de andere onderdelen. Je kunt van het apparaat onderdelen bekijken, de hardware uitzoeken, en de usability bekijken.

Interviewer: Dus ze bekijken van één apparaat verschillende aspecten?

Respondent: Ja. Een mogelijk nadeel is wel dat het meteen vrij analytisch is. We schieten vrij hoog in de Bloom-taxonomie voor een beginmodule.

Interviewer: Als je het in leerdoelen omschrijft...

Respondent: Aan het eind van deze module kun je automaten gebruiken om de logische werking van een systeem weer te geven.

Interviewer: En is daar nog een bijzondere reden voor?

Respondent: Ik denk dat dit verenigbaar was met wat er in het examenprogramma staat. Het is praktisch, iets dat ze kunnen doen, en dat haalbaar is voor de meesten. En het past in het geheel.

Interviewer: Je kunt eindige automaten heel theoretisch aanvliegen. Je kunt ze ook praktischer aanvliegen: je gaat praten over toestanden en pijltjes en je hoeft de automaten niet eens bij naam te noemen. Je presenteert het dan als instrument. In de context van voortgezet onderwijs, welke aanvliegroute zou je belangrijker vinden?

Respondent: De theoretische kennis is leuk. Voor een keuzethema is dat top. Zeker als leerlingen er zelf voor kiezen. Het is dan niet per se te moeilijk voor middelbare scholieren, zeker voor de laatste leerjaren. Mits ze het interessant vinden. Het interessant maken voor degenen die nog geen interesse hebben is erg moeilijk. Dan heeft de praktische aanvliegroute zin. Dat is een gereedschap dat ze daadwerkelijk zou kunnen helpen. Zeker als ze ingewikkeldere dingen programmeren kan het nuttig zijn om er op die manier over te denken.

Interviewer: Samengevat: de theoretische kennis is leuk, maar de praktische kant is waardevol en nuttig, omdat je het kunt toepassen.

Respondent: Ja. En ik denk dat het een gereedschap kan zijn om te snappen hoe een apparaat werkt.

Interviewer: Je gaf aan dat je geen subdomein bewust voor of na [deze module] geeft.

Respondent: Inderdaad (...).

Interviewer: Je stelt dus ook geen voorwaarden?

Respondent: Nee.

Interviewer: Waarom?

Respondent: De standaard is bij ons zo open mogelijk. (...) Alleen als dat niet haalbaar is, stellen we voorwaarden. (...)

Interviewer: Wat zou je in de toekomst nog voor veranderingen willen zien in je lessen over eindige automaten?

Respondent: De praktische opdracht tijdens de les al oefenen, daar ben ik dit jaar voor het eerst mee begonnen. Ze zijn nu begonnen aan de eindopdracht en gaan het binnenkort inleveren; ik hoop dat het

verbetering oplevert. Voor de volledigheid zou het leuk zijn als we grammatica's ergens kwijt kunnen om het netjes te doen, maar ik weet niet hoe.

Interviewer: Dus je zegt expliciet: ik heb hulp nodig van andere partijen hiervoor. Van uitgeverijen en andere docenten. Of door het onderwerp te schrappen in de komende vernieuwing.

Respondent: Die laatste is de meest haalbare. Uitgeverijen zouden ons kunnen helpen, maar we gaan niet voor één subdomein een methode afnemen. We gebruiken alleen ons eigen lesmateriaal.

Interviewer: Wil je nog iets anders kwijt?

Respondent: Nee, we hebben alles gehad.

D.2 Interview 2

Interviewer: Ik weet dat je zowel eindige automaten als grammatica's in de lessen behandelt. Per onderwerp wil ik de diepte in gaan, zodat we niet alleen weten wat je doet, maar ook waarom je dat doet. Die 'waarom' vind ik erg interessant. Laten we beginnen met eindige automaten. De eerste, brede vraag: wat doe jij precies met eindige automaten in de lessen? Waar zijn je leerlingen mee bezig?

Respondent: Ik denk dat we meer met de theorie bezig zijn geweest van eindige automaten: een idee krijgen hoe een apparaat werkt. We zijn steeds beter geworden in het ontdekken van wat leerlingen wel en niet oppakken. Er is ook een verschil tussen havo en vwo. Een moeilijke opdracht is bijvoorbeeld een hotelkluis als eindige automaat bekijken. Een deel van de leerlingen pakt het op, een ander deel niet. We zijn naar steeds eenvoudigere opdrachten en uitleg gegaan. Voor havo-leerlingen is het idioot moeilijk, voor vwo-leerlingen gaat het gemakkelijker. Grammatica's vinden leerlingen uiteindelijk wel leuk, maar de weg ernaartoe is lang. Je moet eindige automaten meenemen, want het een heeft met het ander te maken.

Het viel ook op dat sommige leerlingen het vak kiezen en verwachten zonder moeite een zes te kunnen halen. Ze verwachten er helemaal niets van. En dan toch geef je eindige automaten en grammatica's op het eind, en begrijpen ze ineens hoe natuurlijke talen werken. Ze konden het koppelen aan Nederlands en Engels, waar ze dan net in de examenfase zaten. Ze begonnen ook te merken dat ze eindige automaten en grammatica's overal om zich heen konden zien.

Ik laat leerlingen grammatica's schrijven in EBNF en niet als BNF, dat helpt ontzettend. En ze begrijpen beter hoe parsers werken; waarom programmeren zo strikt is. Ik heb me ook lang afgevraagd welke volgorde we moesten hanteren: eerst eindige automaten en dan grammatica's of andersom? Het is ook moeilijk bij het vak informatica. Je hebt in een klas enorm veel niveauverschillen. Sommige docenten blijven van mening dat iedereen gelijk moet blijven; dezelfde stof moet doen. Ik ben overtuigd dat dat niet de bedoeling is. Het gaat in de bovenbouw om presteren op jouw niveau, niet eronder of erboven.

Interviewer: Het gaat erom dat je groeit. Waar je vandaan komt verschilt per leerling.

Respondent: Precies. Informatica leent zich als vak ideaal hiervoor. Tegenwoordig laat ik de leerlingen veel duidelijker zien dat informatica een soort verbinding vormt naar andere vakken. Ik laat ze dan via een stroomdiagram nadenken: hoe verbindt informatica met andere vakken? Op die manier pas ik het toe. We zijn nog steeds in ontwikkeling.

Interviewer: Laten we het concreter hebben over wat je met eindige automaten doet. Welke tool gebruik je om ze te tekenen? Welke tool gebruiken je leerlingen? Waarom?

Respondent: Ik gebruikte in het verleden JFLAP. Alleen die werkte op een bepaald moment niet meer. Onlangs is het me weer gelukt om hem op te starten. Maar het is een vervelend proces. Er is ook een online tekentool (madebyevan.com/fsm) die ik liever gebruik. Ik heb geen betere gevonden die makkelijk te bedienen is.

Interviewer: Hij is inderdaad makkelijk in gebruik. Je hoeft niets te installeren.

Respondent: Precies. Lokaal software installeren geeft altijd moeilijkheden bij leerlingen die iets verkeerd hebben staan of dat het niet lukt. Ik ben daar tegenwoordig vanaf gestapt. Ik doe zo veel mogelijk online. Iedereen ziet dan hetzelfde. En ik gebruik dus deze tool. Leerlingen vinden het moeilijk om de theorie in boeken, met a's en b's en zo, te begrijpen. Je moet het inzichtelijk maken. Zeker voor havo-leerlingen. Vwo-leerlingen pakken het soms nog wel op.

Interviewer: Je kunt eindige automaten heel theoretisch aanvliegen. Je kunt ze ook praktischer aanvliegen: je gaat praten over toestanden en pijltjes en je hoeft de automaten niet eens bij naam te noemen. Je presenteert het dan als instrument voor programmeren. In de context van voortgezet onderwijs, welke aanvliegroute zou je belangrijker vinden?

Respondent: Ik heb het ooit geprobeerd met de theoretische aanvliegroute. Ik was ze al na 30 seconden kwijt. Ik ben er meteen vanaf gestapt. Ik moest die les meteen improviseren om de praktische route te gebruiken. En dat was zelfs bij vwo-leerlingen. Later, wanneer ze het begrepen, gingen ze wel naar het theoretische kader. Daarvan begreep een derde deel het wel. Dat heb ik sowieso altijd in gedachten: een derde deel pakt het op, een derde deel zit op de helling – die pakken het op of niet – en een derde deel interesseert het niets.

Interviewer: Hoe toets je de kennis over eindige automaten? Je gaf aan dat je een theoretische toets op papier en een praktische toets met een digitaal product gebruikt. En bij grammatica's gebruik je een online Nearly-parser (omrelli.ug/nearly-playground).

Respondent: Klopt. We testen theoretisch even of ze het begrip van eindige automaten hebben. Of ze eindig of oneindig zijn maakt me niet uit. Je moet daar heel precies voor communiceren, en de helft van de leerlingen breng je juist daarmee in verwarring. Zulke discussies over de details kun je, vooral bij havo, beter uit de weg gaan.

Interviewer: Oké, dus je doet een kleine theoretische toets om te kijken of ze de eigenschappen een beetje onder de knie hebben?

Respondent: Ja, maar ik leg de nadruk bij grammatica's. Dat ligt veel dichter bij hun belevingswereld. Ze zien tegenwoordig ook verbanden tussen grammatica's en ChatGPT. De techniek heeft alles vrijer gemaakt, maar chatbots zullen in eerste instantie wel een grammatica hebben. Het doet er alleen minder toe of er een schrijffout of een spatie te veel is. Ze begrijpen ineens wel dat een orde van verschillende programma's gebruikmaken van eindige automaten en grammatica's. Op die manier zien ze die grote blokken en lijnen.

Ik kan dan ook een stap maken naar domein R (computational science) of F1 (usability) waar er een onderzoekscyclus of user design cyclus is. En die kun je ook zien als toestandsdiagram. Dan zien leerlingen: toestandsdiagrammen zijn eigenlijk om een overzicht te bewaren/krijgen. Ik probeer het aan hun dagelijks leven te koppelen. Ik vind het al leuk als ze zien: hé, dit lijkt op een toestandsdiagram.

Interviewer: Je toetst dus theoretisch. Je toetst ook praktisch met een digitaal product. Wat doen ze dan precies?

Respondent: Ze moeten dan een grammatica over iets willekeurigs maken. Ik heb een voorbeeld gegeven, die specificaties van computers genereert. Ze moeten die dan analyseren en markeren, en eventueel stukken overnemen. En dan maken ze een eigen grammatica. Sommige leerlingen gebruiken ChatGPT, maar die produceert alleen BNF, terwijl ze EBNF moeten maken. Sommige leerlingen

maken er een hele fantasiewereld omheen. Anderen maken een eigen taal. Leerlingen die een fantasiewereld hebben, koppelen hun grammatica ook aan generatieve AI. Hun grammatica maakt dan een prompt die ingevoerd kan worden in een AI-plaatjesmaker, waardoor ze in feite een grammatica hebben die plaatjes genereert. Ik vind dat je hun fantasie dan ook geheel de vrije loop moet kunnen laten gaan. Het kan alle kanten op, en ik stimuleer dat ook. Het moet bij hun passen.

Interviewer: Ik snap nog even niet goed wat ze nou precies moeten ontwikkelen.

Respondent: Ze moeten eerst een grammatica ontwikkelen. De logica ervan moeten ze omschrijven met een toestandsdiagram of flowchart. Ze blijven dus de hele tijd grondslagen toepassen. Het eindproduct is een vrije uitvoering van de basis die ze hebben gekregen. Degenen die echt heel erg geïnteresseerd zijn moeten doorgaan met Turingmachines. Daar kom ik bij de meesten niet aan toe: je kunt ze uitleggen wat het is, en dan is de tijd voorbij. Alleen de hele goede leerlingen komen eraan toe. Ik zeg dan: het kost net zo veel tijd als je huidige PO, dat ik met maximaal een 7 of 8 beoordeel. Alleen met een Turingmachine erbij krijg je een 10. Maar het is net zo veel werk. Dus je belooft alleen de goede leerlingen. Daar heb ik de laatste vijf jaar slechts vier of vijf leerlingen van gehad.

Interviewer: We hebben al gekeken naar het bewust behandelen van eindige automaten en grammatica's tegelijk. Die hebben een verband met elkaar. Heb je ook andere onderwerpen, buiten grondslagen, die je bewust voor of na eindige automaten en grammatica's behandelt?

Respondent: We doen eerst eindige automaten. Die koppel ik pas veel later aan de PO met grammatica's. Een deel van de stof betrekken we, zonder het te noemen, bij programmeren. We noemen het nog niet. Tegen de tijd dat we de verdieping zoeken, moet er een 'oh ja'-moment komen. Veel lesmethoden doen dat anders: ze leggen eerst de basis uit, en passen het toe in programmeren. Maar dat hebben ze nog niet nodig. Je kunt het beter omdraaien. Dan krijgen ze later een herkenning: oh, zo hangt het samen, er komt begrip. Dat wil ik eigenlijk.

Interviewer: Wat zou je in de toekomst willen veranderen aan jouw lessen over eindige automaten en grammatica's om ze nog beter te maken?

Respondent: We moeten het er met meer mensen over hebben. Om te kijken wat hun ervaringen zijn en zo tot een beter product te komen. Alleen is maar alleen. Dat is niet genoeg. Je moet het met veel meer mensen delen en dan verbeteren. Er zullen vast verbeterlagen zijn, maar ik zit nu op een dusdanige fase dat je in je eentje niet meer weet waar je moet zoeken en waar je moet beginnen. Het werkt, je ziet dingen langskomen, en je zit met een dilemma: het sluit nu goed aan bij de ontwikkeling van een leerling, een nieuwe versie misschien niet. Als je ze alles vertelt, ontdekken ze dingen zelf niet meer. Door ze dingen te vertellen, beperk je soms hun creativiteit. Dus moet je dan wel verbeteren? Dat is een spanningsboog.

Interviewer: Als ik het goed begrijp, dan zou je graag meer bronnen krijgen om beter te weten wat de mogelijkheden zijn over eindige automaten en grammatica's, en dan zeg je eigenlijk in zekere zin: ik heb hulp nodig van andere collega's en vakexperts om te kunnen verbeteren.

Respondent: Dat klopt. Maar het zou ook handig zijn als de boekmakers het theoretisch kader veel meer loslaten. Dat is het nadeel van het examenprogramma. Het is zo theoretisch opgezet, en daardoor is de eerste opzet van de boekmakers geweest om dicht bij de theorie en het programma te blijven. In havo heb je daar niets aan.

Interviewer: Ik heb al twee bijzondere ontdekkingen gedaan in dit onderzoek. Ik dook de boeken in om te kijken waarom deze onderwerpen in het programma zitten. Wat zag ik? De commissie heeft expliciet in het advies benoemd dat het niet de bedoeling is om alles theoretisch aan te vliegen met alle termen en verbanden en wiskundige eigenschappen. Ze moeten eindige automaten en grammatica's alleen als hulpmiddel gebruiken om projecten op te leveren.

Respondent: Dus eigenlijk zit ik op de goede weg?

Interviewer: Ja! En ik heb ook ontdekt, of eigenlijk bevestigd: geen enkele hbo-opleiding doet iets met eindige automaten of grammatica's. Alleen bij universiteiten.

Respondent: Ik denk dan: het te theoretische kader in de leerdoelen is dus eigenlijk verkeerd opgepakt. De kleine lettertjes daar leest iedereen overheen. Er moet een extra toelichting komen voor havo/vwo, hoe je dat moet oppakken.

Ik heb die lessen over grammatica's trouwens met een collega ontwikkeld, toen we met een groep docenten bezig waren met het maken van materiaal voor grondslagen. Iedereen hield zich toen aan de basis van wat je ermee kunt doen. Ik dacht: weet je wat, ik ga gewoon een grammatica schrijven die verschillende soorten computers kan genereren. Je krijgt dan rare computers, met bijvoorbeeld belachelijke hoeveelheid cores of geheugen. Leerlingen zeggen dan gelijk; dat kan helemaal niet. Het geeft de lol weer. Iedereen hield zich zo strikt aan de basis. Vervolgens heb ik geprobeerd om een Turingmachine te maken voor een draaischijftelefoon.

Interviewer: Dan nog een stukje grammatica's. Laten we daar in duiken. Je leerlingen maken hun eigen grammatica's, maar je bent daar heel creatief in. Het hoeft niet per se een taal te zijn; de grammatica's genereren iets, en wat dat is, bepalen ze zelf.

Respondent: Ja. Het kunnen zelfs auto's zijn. Het raakt aan hun interesses. Als een leerling heel veel hamburgers wil eten, moet hij maar creatief zijn in het maken van verschillende hamburgers.

Interviewer: Oh, dus dan zou de grammatica bijvoorbeeld zeggen: een hamburger bestaat uit een broodje, toppings, burger en nog een broodje. En dan een regel: toppings is tomaat, komkommer, augurk, en nog meer dingen. En de zinnen die de grammatica produceert zijn auto's in het autobedrijf en burgers in het burgerrestaurant.

Respondent: Ja. Met een sandwich hebben we leerlingen vervolgens de grootste lol gehad door de gegenereerde sandwiches door generatieve AI te laten maken. Dan ben je op een speelse manier bezig om leerlingen te laten begrijpen wat het is, zonder het te theoretisch te houden. Dit komt ook omdat ik vroeger veel heb geprogrammeerd, in FORTRAN. Ik weet: het interesseert leerlingen helemaal niks, ongeacht programmeertaal. Je moet het dan dus op een andere manier aanvliegen. Ik merkte dat ook toen ik als zij-instromer mijn eerstegraads bevoegdheid ging halen. Je denkt dan na: welke theorieën kan ik toepassen? De feedback en de lol die leerlingen hadden, waren ontzettend waardevol. Dan zie je dat ze het begrijpen en dat ze hun werk gedaan hebben. Ik vind dat leerlingen moeten leren en ontdekken. Of eigenlijk leren via ontdekken.

Interviewer: Dan de laatste afsluitende vraag. Ik vroeg hem eerder in de gesloten vragenlijst, maar ik ga je expres niet vertellen wat je toen als antwoord gaf. Wil je nog iets anders kwijt, waar we het nog niet over hebben gehad, maar dat wel relevant is voor het onderzoek?

Respondent: Ik hoop dat het onderzoek duidelijk naar voren laat komen dat we de academische insteek minder moeten doen. Er moet een toelichting komen voor zowel havo als vwo op het examenprogramma. Ik denk dat het op zich niet slecht is, maar er moet een betere toelichting komen. Je ziet ook in de keuzethema's dat de uitvoering over het algemeen veel te theoretisch is. De tijd die je kwijt bent aan de verschillende onderdelen matcht niet. Je bent te veel tijd kwijt en we moeten het realistischer invullen. Met wat er nu staat, haal je het gewoon niet. Je moet keuzes maken. En voor alle domeinen geldt: het gaat er niet per se om dat ze het einddoel halen, maar dat ze er naartoe werken.

Interviewer: En als ze het écht willen halen, dan gaan ze dat op een vervolgopleiding doen?

Respondent: Precies. En dat is wat je wil stimuleren. Dat ze nieuwsgierigheid krijgen.