

Leren programmeren in het basisonderwijs

Een onderzoek naar de effecten van een leerarrangement rond programmeren op de computational thinking vaardigheden van basisschoolleerlingen



Pierre Gorissen

Marjoke Bakker

Nieske Coetsier

Rob Hölsgens

Wouter van den Berg

Marijke Kral

Colofon

Januari 2019

Auteurs

Pierre Gorissen, senior onderzoeker
Marjoke Bakker, onderzoeker
Nieske Coetsier, onderzoeker
Rob Hölsgens, onderzoeker
Wouter van den Berg, mediamentor
Marijke Kral, lector leren met ict

ixperium/Centre of Expertise Leren met ict

Kenniscentrum Kwaliteit van Leren
Faculteit Educatie,
Hogeschool van Arnhem en Nijmegen

www.ixperium.nl



Geselecteerde licentie

Naamsvermelding-NietCommercieel-GelijkDelen 4.0 Internationaal



Je bent vrij om:

het werk te delen – te kopiëren, te verspreiden en door te geven via elk medium of bestandsformaat

het werk te bewerken – te remixen, te veranderen en afgeleide werken te maken

De licentiegever kan deze toestemming niet intrekken zolang aan de licentievoorwaarden voldaan wordt.

Onder de volgende voorwaarden:

Naamsvermelding – De gebruiker dient de maker van het werk te vermelden, een link naar de licentie te plaatsen en aan te geven of het werk veranderd is. Je mag dat op redelijke wijze doen, maar niet zodanig dat de indruk gewekt wordt dat de licentiegever instemt met je werk of je gebruik van het werk.

NietCommercieel – Je mag het werk niet gebruiken voor commerciële doeleinden.

Geen aanvullende restricties – Je mag geen juridische voorwaarden of technologische voorzieningen toepassen die anderen er juridisch in beperken om iets te doen wat de licentie toestaat.

Let op:

Voor elementen van het materiaal die zich in het publieke domein bevinden, en voor vormen van gebruik die worden toegestaan via een uitzondering of beperking in de Auteurswet, hoef je je niet aan de voorwaarden van de licentie te houden.

Er worden geen garanties afgegeven. Het is mogelijk dat de licentie je niet alle gebruiksvoorwaarden geeft die nodig zijn voor het beoogde gebruik. Bijvoorbeeld, andere rechten zoals publiciteits-, privacy- en morele rechten kunnen het gebruik van een werk beperken.

INHOUD

Samenvatting	4
1. Inleiding	5
1.1. Praktijkvraag.....	5
1.2. Computational Thinking (CT)	5
1.3. Programmeren en CT	6
1.4. Gepersonaliseerd leren.....	6
1.5. Motivatie	7
1.6. Onderzoeksvragen.....	7
2. Methode van onderzoek	8
2.1. Het onderzoek	8
2.2. Het leerarrangement	9
2.3. Meetinstrumenten.....	11
2.4. Deelnemende scholen en leerlingen.....	17
3. Resultaten	19
3.1. Beginsituatie	19
3.2. Ontwikkeling van de CT-vaardigheden	21
3.3. Ontwikkeling van de motivatie voor programmeren.....	22
3.4. Factoren die een rol spelen bij de ontwikkeling van CT-vaardigheden	23
3.5. Factoren die een rol spelen bij de ontwikkeling van motivatie	23
4. Discussie en conclusie	25
4.1. Ontwikkeling van de CT-vaardigheden	25
4.2. Ontwikkeling van de motivatie voor programmeren.....	25
4.3. Robot vs. beeldschermomgeving.....	25
4.4. Gepersonaliseerd leren.....	26
4.5. Verschillen tussen leerlingen.....	26
4.6. Conclusies	26
Referenties	27
Bijlage: statistische details	28

SAMENVATTING

De ontwikkeling van computational thinking vaardigheden van leerlingen staat steeds meer in de belangstelling in het onderwijs. Bij computational thinking gaat het om het logisch denken en het opdelen van problemen in stappen die deels door een computer kunnen worden uitgevoerd. Ook in het basisonderwijs wordt hier steeds meer aandacht aan besteed, onder andere door het aanbieden van programmeeronderwijs. De vraag is echter wat de effecten hiervan zijn. Het iXperium/Centre of Expertise Leren met ict heeft in samenwerking met de stichting Conexus onderzoek gedaan naar de effecten van programmeeronderwijs in de bovenbouw van de basisschool. Zorgt programmeeronderwijs inderdaad voor betere computational thinking vaardigheden bij leerlingen? En bevordert het de motivatie van leerlingen voor programmeren?

Aan het onderzoek deden ruim 200 leerlingen van vijf basisscholen mee. Deze leerlingen volgden een leerarrangement over programmeren, bestaande uit vier lessen. De leerlingen leerden ofwel programmeren met een beeldschermomgeving (Scratch), ofwel met een robot (Edison). Sommige leerlingen kregen een gepersonaliseerde variant van het leerarrangement, waarbij zij konden kiezen uit verschillende maten van sturing; andere leerlingen hadden deze keuze niet. Bij de leerlingen werd een voormeting en een nameting afgenomen, waarin de computational thinking vaardigheden en de motivatie voor programmeren werden gemeten.

De resultaten laten zien dat leerlingen na de programmeerlessen hogere computational thinking vaardigheden hadden dan ervoor. Dit wijst erop het leerarrangement inderdaad bijdroeg aan de computational thinking vaardigheden van de leerlingen. Het maakte hierbij niet uit of leerlingen hadden geprogrammeerd met een beeldschermomgeving of met een robot. Ook maakte het niet uit of leerlingen de gepersonaliseerde variant van het leerarrangement hadden gevolgd (met keuze in mate van sturing) of de niet-gepersonaliseerde variant.

Naast de computational thinking vaardigheden ging ook de motivatie (interesse) voor programmeren vooruit. Leerlingen hadden na afloop van de lessen meer interesse voor programmeren dan ervoor. De toegenomen interesse kwam ook tot uiting in gedrag: Bijna veertig procent van de leerlingen gaf aan dat zij buiten de lessen om meer zijn gaan programmeren.

Het leerarrangement was het meest effectief voor leerlingen in de hogere leerjaren van de bovenbouw (groep 7-8). Voor leerlingen in lagere leerjaren van de bovenbouw (groep 5-6) zou een aangepaste leerarrangement mogelijk geschikter zijn, met meer ondersteuning of oefening. Verder bleek dat leerlingen die vooraf al ervaring hadden met programmeren en leerlingen met hogere rekenvaardigheden meer baat hadden bij het leerarrangement; zij pikten het programmeren sneller op. Leerlingen zonder programmeerervaring vooraf en leerlingen met lagere rekenvaardigheden hebben waarschijnlijk wat meer ondersteuning nodig.

1. INLEIDING

Computational Thinking (CT) en daaraan gerelateerde onderwerpen als programmeren, algoritmisch redeneren en robotisering, staan de laatste jaren in toenemende mate in de belangstelling binnen het onderwijs in Nederland en erbuiten (Bocconi, Chiocciariello, Dettori, Ferrari, & Engelhardt, 2016; Pijpers, 2015). Ook in het basisonderwijs is hier steeds meer aandacht voor. Bij CT gaat het om een manier van denken en probleemoplossen, waarbij problemen worden opgedeeld in stappen, die deels door een computer (of robot, etc.) kunnen worden uitgevoerd.

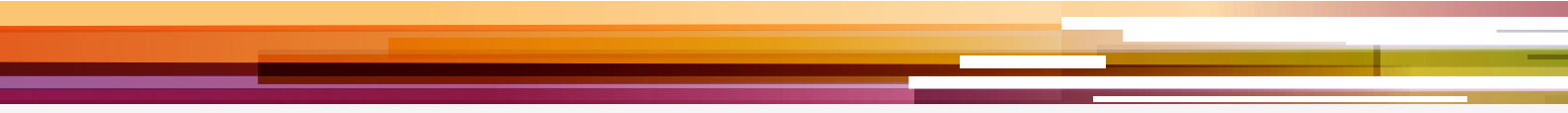
Binnen de stichting Conexus, een stichting bestaande uit 31 scholen voor basisonderwijs, speciaal basisonderwijs en speciaal onderwijs in de gemeenten Heumen en Nijmegen, is onderzoek gedaan naar het bevorderen van CT bij basisschoolleerlingen door middel van programmeeronderwijs. Dit rapport beschrijft het onderzoek en de resultaten. Het onderzoeksproject werd uitgevoerd door het iXperium/Centre of Expertise Leren met ict, in het schooljaar 2016/2017. Het onderzoek is mogelijk gemaakt door een subsidie in het kader van het landelijk onderzoeksproject Doorbraak Onderwijs & ICT.

1.1 Praktijkvraag

Het onderzoek is gestart naar aanleiding van een praktijkvraag van de stichting Conexus. De stichting wilde weten in hoeverre programmeren in het onderwijs bijdraagt aan de CT-vaardigheden van leerlingen, alsmede aan de motivatie van leerlingen voor programmeren. De stichting vroeg zich daarbij af of kinderen makkelijker leren programmeren met een robot dan op een scherm, vanwege de fysieke component van een robot. Vanuit de toenemende aandacht binnen de stichting voor gepersonaliseerd leren, was de stichting ook geïnteresseerd in de rol van gepersonaliseerd leren bij het leren programmeren. In het project is daarom gekeken naar de effecten van een leerarrangement rond programmeren op de CT-vaardigheden van leerlingen en hun motivatie voor programmeren, en is daarnaast onderzocht of deze effecten verschillen wanneer er met een robot, dan wel op een scherm wordt geprogrammeerd, en wanneer het leerarrangement al dan niet elementen van gepersonaliseerd leren bevat.

1.2 Computational Thinking (CT)

Computational Thinking (CT) wordt gezien als het denkproces dat nodig is voor het formuleren van problemen en oplossingen op een manier die het mogelijk maakt om de oplossingen op een efficiënte wijze uit te laten voeren door een informatie-verwerkende agent, zoals een computer of robot (Wing, 2010). Waar eerst de nadruk lag op het leren van procedureel denken door te programmeren (Papert & Harel, 1991) wordt CT nu vaak breder opgevat. Brennan en Resnick (2012) ontwikkelden een kader voor CT op basis van onderzoek naar activiteiten van jongeren die interactieve media ontwerpen. Het kader bestaat uit drie sleuteldimensies: *computational concepts*, *computational practices* en *computational perspectives*. Bij *computational concepts* gaat het met name over kennis van de verschillende mogelijkheden of onderdelen van programmeren, zoals opeenvolgingen, condities en herhalingen. Programmeren is een optelsom van computationele concepten. Om goed te kunnen programmeren heb je op zijn minst begrip nodig van deze concepten. Bij *computational practices* gaat het om het daadwerkelijk programmeren door gebruik te maken van de computationele concepten. Brennan en Resnick zien vier manieren die bij het programmeren worden gebruikt om de computationele concepten in



te zetten: incrementeel en iteratief, testen en debuggen, hergebruiken en samenvoegen, en abstraheren en moduleren. De laatste sleuteldimensie, *computational perspectives*, gaat over de verandering in perspectief die het programmeren met zich mee brengt: het programmeren draagt bij aan een andere kijk op de technologische wereld en biedt nieuwe mogelijkheden om ideeën uit te drukken en met anderen samen te werken.

1.3 Programmeren en CT

Computational thinking is gerelateerd aan programmeervaardigheden, maar de vertaalslag van concrete programmeeractiviteiten naar een aantoonbare bijdrage aan CT-vaardigheden blijkt niet eenvoudig, zo blijkt uit een recente inventarisatie van CT binnen het Europees onderwijs (Bocconi e.a., 2016). In ons onderzoek is onderzocht of een kort leerarrangement rond programmeren bijdraagt aan de ontwikkeling van CT-vaardigheden bij basisschoolleerlingen. Omdat basisschoolleerlingen vaak nog weinig ervaring hebben met programmeren, richtten we ons op de basiskennis van computational concepts en op enkele aspecten van computational practices.

Het zou kunnen dat het programmeren van fysieke robots bij kinderen meer tot de verbeelding spreekt dan het programmeren op een beeldscherm. De bewegingen die worden geprogrammeerd, worden fysiek door de robot uitgevoerd, in plaats van op het beeldscherm getoond, wat het begrip kan vergroten. De fysieke positie van de robot in de ruimte kan bijvoorbeeld helpen bij het koppelen van de bewegingen aan de code, zoals wanneer de robot in de kijkrichting van de leerling wordt geplaatst (een zogenoemde complementaire actie, Antle, 2013). We onderzoeken daarom of er verschillen optreden tussen leerlingen die leren programmeren met een robot en leerlingen die leren programmeren op een beeldscherm.

1.4 Gepersonaliseerd leren

Gepersonaliseerd leren heeft betrekking op leersituaties die tegemoetkomen aan individuele verschillen tussen leerlingen (Marquenie, Opsteen, Ten Brummelhuis, & Van der Waals, 2014). Hierbij kan enerzijds onderscheid gemaakt worden in de mate van uniformiteit versus flexibiliteit, bijvoorbeeld één niveau van instructie voor alle leerlingen versus verschillende niveaus van instructie. De andere dimensie waarop gevarieerd kan worden, is de mate van externe regie versus zelfregie, oftewel de mate waarin de keuze voor het niveau en de leerinhoud bij de leraar ligt of bij de leerling zelf (Van Loon, Van der Neut, De Ries, & Kral, 2016).

Door middel van gepersonaliseerd leren kan beter worden aangesloten bij de leerbehoeften van individuele leerlingen, en zelfregie kan bovendien bijdragen aan een hogere leermotivatie. We verwachten daarom dat een gepersonaliseerd leerarrangement rond programmeren leidt tot hogere leeruitkomsten dan een niet-gepersonaliseerd leerarrangement.

1.5 Motivatie

De intrinsieke motivatie van leerlingen voor een onderwerp bestaat uit verschillende deelaspecten, waaronder interesse/plezier, competentiegevoel, waargenomen inspanning, waargenomen keuzevrijheid en waargenomen nut (Ryan & Deci, 2000; Self-Determination Theory, z.d.). Interesse/plezier voor het onderwerp is de meest rechtstreekse maat voor intrinsieke motivatie. De andere deelaspecten zijn voorspellers van intrinsieke motivatie. Zo draagt bijvoorbeeld het gevoel goed te zijn in een bepaalde activiteit (competentiegevoel) bij aan de intrinsieke motivatie. Ook het toekennen van waarde of nut aan een onderwerp en het ervaren van keuzevrijheid hebben een positief effect op de intrinsieke motivatie. Leerlingen die intrinsiek gemotiveerd zijn, zetten zich vaak meer in voor een lesactiviteit.

1.6 Onderzoeksvragen

De onderzoeksvragen zijn:

1. Draagt een leerarrangement rond programmeren bij aan de ontwikkeling van CT-vaardigheden van leerlingen in de bovenbouw van het basisonderwijs?
2. Draagt een leerarrangement rond programmeren bij aan de motivatie van leerlingen met betrekking tot programmeren?
3. Is de ontwikkeling in CT-vaardigheden en in motivatie met betrekking tot programmeren gerelateerd aan:
 - a. het volgen van een leerarrangement waarbij geprogrammeerd wordt met een robot dan wel met een beeldschermomgeving?
 - b. het volgen van een gepersonaliseerd dan wel niet-gepersonaliseerd leerarrangement?
 - c. de mate van ervaring die leerlingen vooraf hadden met programmeren?
 - d. de voorkennis van leerlingen wat betreft CT en de initiële motivatie van leerlingen wat betreft programmeren?
 - e. de leerlingkenmerken geslacht, leerjaar, en vaardigheid in rekenen en begrijpend lezen?

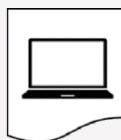
2. METHODE VAN ONDERZOEK

2.1 Het onderzoek

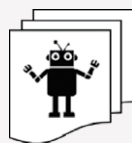
Om de onderzoeksvragen te kunnen beantwoorden, is gebruikgemaakt van een quasi-experimentele onderzoeksopzet. Leerlingen uit verschillende klassen van vijf basisscholen volgden een interventie bestaande uit een leerarrangement rond programmeren. Vier verschillende varianten van dit leerarrangement werden vergeleken, variërend wat betreft twee dimensies: a) gepersonaliseerd leren vs. niet-gepersonaliseerd leren, en b) programmeren met een beeldschermomgeving vs. programmeren met een robot (zie figuur 2.1). Elke leerling volgde één van deze varianten. Elk leerarrangement bestond uit vier lessen.



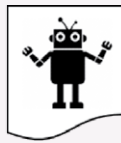
Gepersonaliseerd leren programmeren met een **beeldschermomgeving**



Niet-gepersonaliseerd leren programmeren met een **beeldschermomgeving**



Gepersonaliseerd leren programmeren met een **robot**



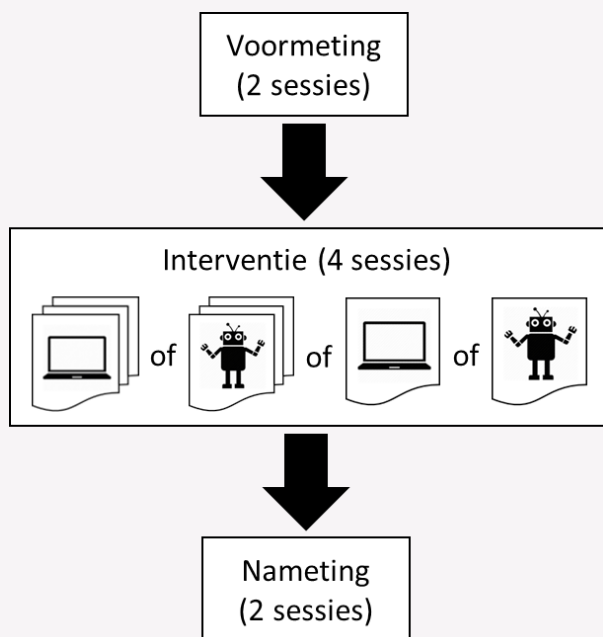
Niet-gepersonaliseerd leren programmeren met een **robot**

De vier varianten van het leerarrangement

Om te kunnen onderzoeken of deze leerarrangementen bijdragen aan de CT-vaardigheden van de leerlingen en hun motivatie wat betreft programmeren, werd gebruikgemaakt van een voormeting en een nameting van CT-vaardigheden en motivatie voor programmeren.¹ Om de invloed van eerdere programmeerervaring te kunnen onderzoeken, werd tijdens de voormeting tevens gevraagd hoe vaak leerlingen al geprogrammeerd hadden. De nameting bevatte naast de metingen van CT-vaardigheden en motivatie, de vraag of leerlingen sinds de voormeting meer zijn gaan programmeren buiten de lessen om. Hiermee werd gemeten of een eventuele toegenomen motivatie voor programmeren ook tot uiting komt in gedrag (het daadwerkelijk meer gaan programmeren). De voormeting en de nameting bestonden ieder uit twee sessies.

De onderzoeksopzet, bestaande uit de voormeting (2 sessies), interventie (4 sessies) en nameting (2 sessies) wordt weergegeven in figuur 2.2.

¹ Op sommige scholen werd op een later moment nog een retentiemeting afgenomen. Deze wordt in deze rapportage niet meegenomen, omdat het maar om enkele scholen ging.



De onderzoeksopzet

2.2 Het leerarrangement

Elke leerling in het onderzoek volgde een van de vier varianten van het leerarrangement variërend wat betreft a) gepersonaliseerd leren vs. niet-gepersonaliseerd leren, en b) programmeren met een beeldschermomgeving vs. programmeren met een robot (zie figuur 2.1). Alle varianten van het leerarrangement bestonden uit vier lessen van ca. 30 minuten, waarin gewerkt werd met werkboekjes. Leerlingen werkten zelfstandig met de werkboekjes; de rol van de leerkracht was minimaal. Zo werd de invloed van de verschillende leraren in het onderzoek zoveel mogelijk beperkt, waardoor de effecten van de verschillende varianten van het leerarrangement zo zuiver mogelijk konden worden gemeten. De leraren werden geïnstrueerd de leerlingen alleen indien nodig te ondersteunen.

2.2.1 Beeldschermomgeving vs. robot

Voor het programmeren met een beeldschermomgeving werd gebruikgemaakt van de online programmeeromgeving Scratch (<https://scratch.mit.edu/>). In Scratch kunnen leerlingen programmeren door middel van blokken die ze in een bepaalde volgorde moeten zetten en waarin ze waarden moeten vullen, zoals posities of aantallen stappen (zie figuur 2.3). In het leerarrangement leerden de leerlingen hoe ze, door het samenstellen van een Scratch code, voorwerpen (“sprites”) op het beeldscherm konden laten bewegen.



Voorbeeld van Scratch-code

Voor het programmeren met een robot werd gebruikgemaakt van de robot Edison (zie figuur 2.4). Deze robot kan worden geprogrammeerd in een grafische omgeving (op een beeldscherm) waarin, net als bij Scratch, blokken in een bepaalde volgorde moeten worden gezet en waarden moeten worden ingevuld (zie figuur 2.4). Waar bij Scratch het resultaat (het bewegende voorwerp) op het beeldscherm wordt getoond, is bij het programmeren van Edison het resultaat te zien in de fysieke bewegingen van de Edison-robot. Zowel bij Scratch als bij Edison gaat het dus om het programmeren van bewegingen, waarbij deze bewegingen dan wel op het scherm worden getoond (bij Scratch), dan wel door een fysieke robot worden uitgevoerd (bij Edison).



De Edison robot (boven) en de bijbehorende programmeeromgeving (onder)

2.2.2 Gepersonaliseerd vs. niet-gepersonaliseerd

De dimensie gepersonaliseerd vs. niet-gepersonaliseerd werd vormgegeven door leerlingen die een gepersonaliseerde variant van het leerarrangement volgden te laten kiezen uit verschillende niveaus van sturing in de instructie. Hiermee werden verschillende instructieniveaus voor verschillende leerlingen gerealiseerd (flexibiliteit in instructie), en werd de regie voor de keuze hiertussen bij de leerling gelegd (zelfregie). Leerlingen die een gepersonaliseerd leerarrangement volgden, hadden de keuze uit drie versies van de werkboekjes:

- *Versie A*: er wordt stap voor stap uitgelegd wat leerlingen moeten doen
- *Versie B*: de eerste stappen worden uitgelegd, waarna de leerling zelf (vergelijkbare) vervolgstappen moet afleiden
- *Versie C*: de eerste stappen worden uitgelegd, waarna de leerling zelf (vergelijkbare) vervolgstappen moet afleiden. Dit wordt aangevuld met een verdiepende opdracht.

De leerlingen die een niet-gepersonaliseerd leerarrangement volgden, hadden deze keuze niet: zij werkten altijd met versie A.

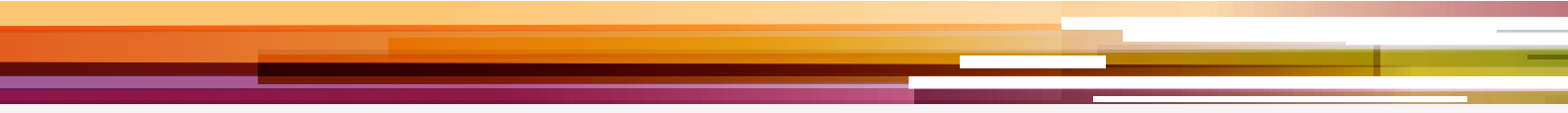
2.2.3 Opbouw van het leerarrangement

Het leerarrangement bestond uit vier lessen van ca. 30 minuten, waarin gewerkt werd met werkboekjes. In les 1 t/m 3 werden verschillende aspecten van het programmeren aangeleerd (bewegen, herhalen, als-dan); les 4 bestond uit een overkoepelende opdracht waarin al deze aspecten toegepast moesten worden. Er waren verschillende werkboekjes voor het programmeren met een beeldschermomgeving (Scratch) en het programmeren met een robot (Edison).² In les 1 t/m 3 waren er voor de gepersonaliseerde variant van het leerarrangement drie verschillende versies van de werkboekjes (versie A, B en C), waaruit de leerlingen mochten kiezen; in de niet-gepersonaliseerde varianten werd alleen gebruikgemaakt van versie A. In les 4 (overkoepelende opdracht) waren er geen verschillende versies van het werkboekje.

2.3 Meetinstrumenten

Om te kunnen onderzoeken of het leerarrangement bijdraagt aan de CT-vaardigheden van de leerlingen en hun motivatie wat betreft programmeren, werd gebruikgemaakt van een voormeting en een nameting. Zowel de voor- als de nameting bestond uit een toets van CT-vaardigheden en vragenlijst over motivatie voor programmeren. Daarnaast werd tijdens de voormeting gevraagd hoe vaak leerlingen al geprogrammeerd hadden, en tijdens de nameting werd gevraagd of leerlingen sinds de voormeting meer zijn gaan programmeren buiten de lessen om. De voor- en nameting werden digitaal afgenomen in de online omgeving van MWM2. Zowel de voor- als de nameting waren opgedeeld in twee sessies (ieder ca. 30 minuten). De eerste sessie bevatte de vragen over intrinsieke motivatie, de vraag over de mate waarin leerlingen zelf al programmeren en een deel van de opgaven over CT-vaardigheden; de tweede sessie bevatte de resterende opgaven over CT-vaardigheden.

² De instructies in de werkboekjes zijn gebaseerd op bestaand instructiemateriaal voor Scratch en Edison. Voor de werkboekjes voor Scratch is voor ideeën en teksten gebruikgemaakt van de MOOC van de TU Delft (www.edx.org/course/scratch-programmeren-voor-kinderen-8-delft-scratchx-o). Voor de werkboekjes voor Edison is gebruikgemaakt van ideeën en teksten uit de door Bert Grave vertaalde handleiding van Edison (<https://meet Edison.com/content/Lesson-plans/10%20Edisonlessen%20NL.pdf>).



Om de vaardigheid van leerlingen in rekenen en begrijpend lezen mee te kunnen nemen in het onderzoek, werd gebruikgemaakt van scores op de toetsen rekenen en begrijpend lezen uit het Cito Leerlingvolgsysteem.

2.3.1 Toets CT-vaardigheden

Omdat er geen bestaand meetinstrument was voor het meten van CT-vaardigheden bij basisschoolleerlingen, is voor dit onderzoek een toets van CT-vaardigheden ontwikkeld.³ De toets bestond uit 20 opgaven en richtte zich op drie categorieën van CT-vaardigheden:

- *Algoritmisch redeneren* (9 opgaven): Bij algoritmisch redeneren is een leerling in staat om een oplossing stapsgewijs te beschrijven in de vorm van een of meerdere algoritmen die geautomatiseerd uitgevoerd kunnen worden. Zie de voorbeeldopgaven in figuur 2.5 en 2.6.
- *Probleemdecompositie* (5 opgaven): Het opdelen van een probleem in eenvoudigere delen met als doel om het mogelijk te maken het probleem in algoritmen te beschrijven. Zie de voorbeeldopgave in figuur 2.7.
- *Debugging* (6 opgaven): Het opsporen en verhelpen van fouten in een algoritme. Zie de voorbeeldopgave in figuur 2.8.

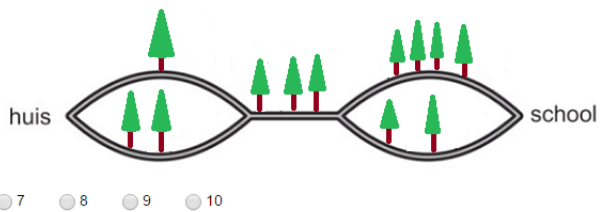
Per categorie en voor het totaal werd de score berekend als het percentage correct beantwoorde opgaven.

Bij de verschillende opgaven is geprobeerd enige variatie in de soort opgaven aan te brengen en de opgaven niet uitsluitend met programmeren te verbinden (zie figuur 2.5 en 2.7). Bij de opgaven die wel met programmeren te maken hadden, is gekozen voor een format anders dan de programmeeromgeving van Scratch of Edison (zie figuur 2.6 en 2.8), zodat geen van de groepen leerlingen voordeel had bij het gebruikte programmeerformat in de toets.

³ Het meetinstrument werd ontwikkeld op basis van opgaven uit een eerder project van RU en HAN in opdracht van het Platform Betatechniek ("Individuele verschillen in onderzoekend en ontwerpend leren: het voorspellend vermogen van het Toetsinstrumentarium Wetenschap & Technologie"), aangevuld met opgaven gebaseerd op de opdrachten zoals beschikbaar via www.code.org en vragen afkomstig van Cito-trainer.

Floris loopt van huis naar school. Hij telt het aantal bomen langs de weg waar hij loopt.

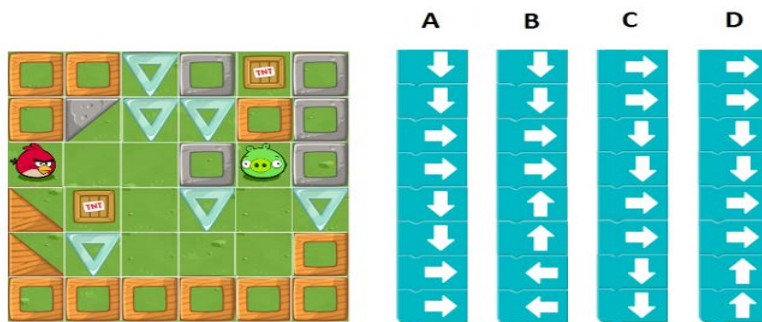
Welk aantal kan **niet** goed zijn?



Figuur 2.5 – Voorbeeldopgave algoritmisch redeneren

Welke route moet het rode vogeltje afleggen om op het groene varkentje te komen?

Kies een van de 4 programma's.



Figuur 2.6 – Voorbeeldopgave algoritmisch redeneren

Je moet het hok van het konijn verschonen. Hieronder staat een lijstje met dingen die je daarvoor moet doen. Kun je onderstaande stappen in de juiste volgorde slepen?

Te rangschikken antwoorden

- Plaats de voederbak en de drinkbeker terug in het hok
- Verwijder de voederbak en de drinkbeker en haal het oude stro uit het hok
- Open het hok en haal het konijn er uit
- Stop het nieuwe stro in het hok en vervang het water in de drinkbeker
- Plaats het konijn terug in het hok en sluit het hok weer af
- Zet het konijn voorzichtig in een kleinere kooi

Gerangschikte antwoorden

Figuur 2.7 – Voorbeeldopgave probleemdecompositie

Het rode vogeltje wil op het groene varkentje terecht komen. Van de vier getoonde programma's zijn er drie goed.

Klik het programma aan dat het vogeltje **niet** op het varkentje brengt.

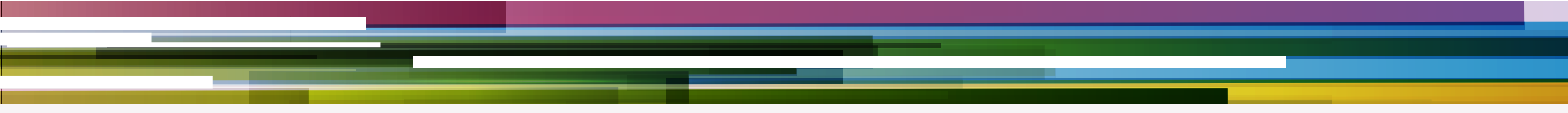
A

B

C

D

Figuur 2.8 – Voorbeeldopgave debugging



Omdat er bij de voormeting rekening mee werd gehouden dat leerlingen mogelijk nog helemaal geen kennis van programmeren hadden, werd bij elk nieuw onderdeel van de toets eerst een scherm met toelichting getoond.

De ontwikkelde toets is eerst uitgetoetst bij 20 leerlingen die niet deelnamen aan het onderzoek. Op basis hiervan is de formulering en vormgeving van de opgaven waar nodig bijgesteld. In de voor- en nameting werden in principe dezelfde opgaven gebruikt, maar de antwoordopties werden gewijzigd (qua inhoud en volgorde) en de afbeeldingen van sommige opgaven werden gespiegeld om de herkenbaarheid van de vragen te verkleinen. Omdat leerlingen na de voormeting geen feedback kregen op hun antwoorden, verwachten we dat het leereffect van het maken van vergelijkbare opgaven klein is.

2.3.2 Vragenlijst motivatie voor programmeren

De motivatie wat betreft programmeren werd in de voor- en nameting gemeten met een vragenlijst gebaseerd op de Intrinsic Motivation Inventory (IMI; zie Ryan & Deci, 2000; Self-Determination Theory, z.d.). In het onderzoek is gebruikgemaakt van de schalen *interesse*, *competentiegevoel* en *waargenomen nut* uit de IMI.⁴ Voor het onderwerp programmeren kunnen deze schalen als volgt omschreven worden:

- *Interesse*: Hoe leuk vinden leerlingen het om te programmeren? Of, als leerlingen voorafgaand aan de interventie nog niet hebben geprogrammeerd, hoe leuk lijkt het ze?
- *Competentiegevoel*: Hoe competent voelen leerlingen zich wat betreft programmeren?
- *Waargenomen nut*: In hoeverre zien leerlingen het nut van programmeren?

De schaal *interesse* is een directe maat van de intrinsieke motivatie van leerlingen. *Competentiegevoel* en *waargenomen nut* zijn factoren die nauw gerelateerd zijn aan de motivatie.

Bij elk van de drie schalen hoorde een aantal stellingen, waarbij leerlingen moesten aangeven in hoeverre deze op hen van toepassing zijn. Hierbij werd gebruikgemaakt van een vijfpuntsschaal van 1 “past helemaal niet bij mij” tot 5 “past helemaal bij mij” (zie figuur 2.9). Per schaal werd een gemiddelde schaa score berekend.

⁴ In de vragenlijst waren ook vragen van twee andere schalen van de IMI opgenomen: waargenomen inspanning en waargenomen keuzevrijheid. Deze schalen bleken tijdens de voormeting onvoldoende betrouwbaar (Cronbach's alpha < .6) en zijn daarom niet in de analyses meegenomen.

Vul in hoe deze uitspraken bij jou passen

	1 (past helemaal niet bij mij)	2	3 (past een beetje bij mij)	4	5 (past helemaal bij mij)
Ik denk dat ik best wel goed ben in programmeren	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ik vond het programmeren best wel leuk	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ik heb geprogrammeerd omdat ik geen keuze had.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ik had het gevoel dat ik programmeren moest doen.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ik denk dat programmeren gunstig is voor mij.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ik denk dat ik best wel vaardig was in het programmeren.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ik heb erg mijn best gedaan bij het programmeren.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Ik ben bereid om nog een keer te programmeren omdat ik er waarde aan hecht.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Figuur 2.9 – Voorbeelden van vragen uit de vragenlijst motivatie voor programmeren

2.3.3 Vragen over programmeerervaring

Om een indicatie te krijgen van de programmeerervaring van de leerlingen voorafgaand aan de interventie, bevatte de voormeting de vraag of de leerlingen al ervaring hadden met programmeren, en zo ja, hoe vaak zij programmeerden en in welke programmeeromgeving. Tijdens de nameting werd gevraagd of leerlingen sinds de voormeting buiten de lessen om (meer) zijn gaan programmeren.

2.3.4 Cito-scores rekenen en begrijpend lezen

Om te kunnen onderzoeken of achtergrondkennis en -vaardigheden van leerlingen mogelijk een rol spelen bij het leren van CT-vaardigheden, werd de vaardigheid van leerlingen in rekenen en begrijpend lezen meegenomen in het onderzoek. Hiervoor werd gebruikgemaakt van de scores voor rekenen en begrijpend lezen uit het Cito Leerlingvolgsysteem. Van bijna alle deelnemende leerlingen was de meest recente Cito-score voor rekenen en begrijpend lezen beschikbaar. In de analyses werd gebruikgemaakt van het leerrendementpercentage (LR%) uit de Cito-gegevens, dat een indicatie geeft van de relatieve vaardigheid van leerlingen. Een LR% van 100 geeft aan dat een leerling scoort op het niveau dat past bij het aantal maanden ontvangen onderwijs. Een LR% hoger dan 100 duidt op een hogere score dan verwacht op basis van aantal maanden ontvangen onderwijs, terwijl een LR% lager dan 100 betekent dat een leerling lager scoort dan verwacht op basis van het aantal maanden ontvangen onderwijs.


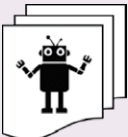

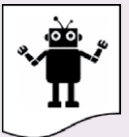
2.4 Deelnemende scholen en leerlingen

Het onderzoek is uitgevoerd bij vijf scholen van stichting Conexus in de regio Nijmegen. De scholen worden in deze rapportage aangeduid als school 1 t/m school 5. Van elke school namen minimaal twee klassen deel aan het onderzoek. Dit waren klassen uit de bovenbouw (leerjaren 5, 6, 7, 8).

Of de deelnemende klassen werkten met de gepersonaliseerde of de niet-gepersonaliseerde leerarrangementen werd bepaald op basis van de bestaande werkwijze in de school of klas: als de school of klas gewend was te werken met gepersonaliseerd leren dan werd de gepersonaliseerde variant van het leerarrangement gebruikt, anders werd gewerkt met de niet-gepersonaliseerde variant. Of leerlingen werkten met de beeldschermomgeving (Scratch) of met de robot (Edison) werd random bepaald. In elke klas werkte de helft van de leerlingen met de beeldschermomgeving en de andere helft met de robot.

In totaal zijn er 271 leerlingen met het onderzoek gestart. Door problemen met het inplannen van de metingen hebben niet alle leerlingen alle sessies van de voor- en nameting voltooid. Leerlingen die een sessie hebben gemist of vragen hebben overgeslagen, konden in sommige analyses niet worden meegenomen. In het bovenste deel van tabel 2.1 wordt weergegeven hoeveel leerlingen in ieder geval de eerste sessie van zowel de voormeting als de nameting hebben voltooid. Deze leerlingen konden worden meegenomen in de analyses van de ontwikkeling van motivatie (de motivatievragenlijst werd steeds in de eerste sessie afgenomen). Het onderste deel van de tabel toont de aantallen leerlingen die beide sessies van zowel de voormeting als de nameting hebben voltooid. Dit zijn de leerlingen die werden meegenomen in de analyses van de ontwikkeling van CT-vaardigheden (de CT-opgaven waren verdeeld over beide sessies van de metingen). Bij de resultaten wordt steeds het aantal leerlingen (*n*) vermeld dat in de betreffende analyses is meegenomen.

Tabel 2.1 – Aantal leerlingen die de eerste sessie van voor- en nameting hebben voltooid en aantal leerlingen die beide sessies van voor- en nameting hebben voltooid, per school en per variant van het leerarrangement

	variant van het leerarrangement				totaal
	 gepersonaliseerd, beeldscherm-omgeving	 gepersonaliseerd, robot	 niet-gepersonaliseerd, beeldschermomgeving	 niet-gepersonaliseerd, robot	
leerlingen die tenminste de eerste sessie van voor- en nameting hebben voltooid					
school 1	0	0	14	17	31
school 2	43	43	0	0	86
school 3	23	21	0	0	44
school 4	0	13	6	9	28
school 5	0	0	21	19	40
<i>totaal</i>	66	77	41	45	229
leerlingen die beide sessies van voor- en nameting hebben voltooid					
school 1 ^a	0	0	0	0	0
school 2	31	32	0	0	63
school 3	15	12	0	0	27
school 4	0	12	5	2	19
school 5	0	0	18	19	37
<i>totaal</i>	46	56	23	21	146

^aVan school 1 had geen van de leerlingen alle sessies (op tijd) voltooid.

3. RESULTATEN

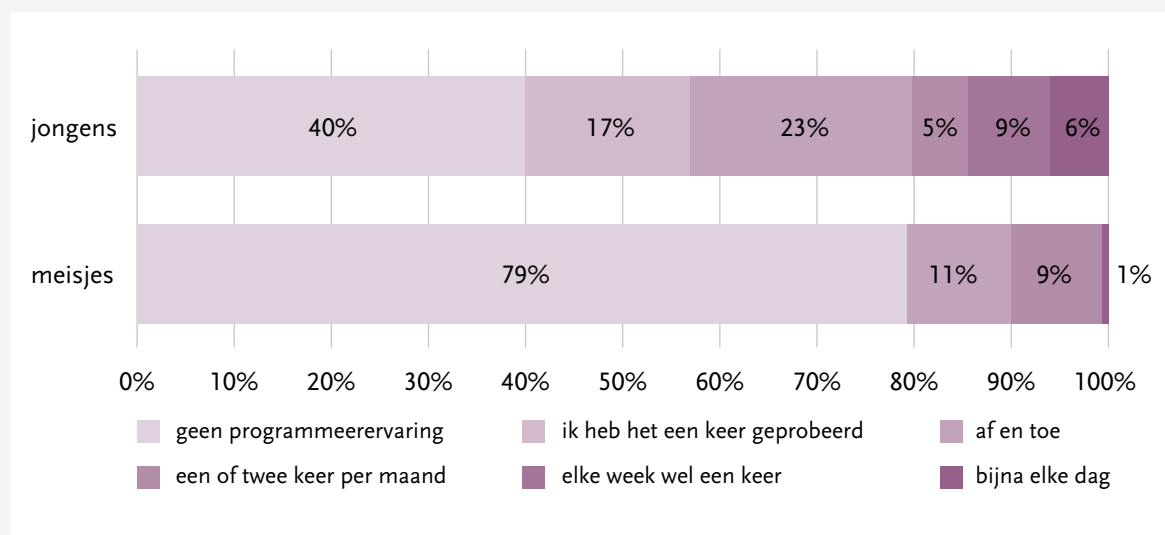
3.1 Beginsituatie

Voordat we de onderzoeksvragen beantwoorden, kijken we eerst naar de beginsituatie. Hadden leerlingen voorafgaand aan de interventie al ervaring met programmeren, hoe scoorden leerlingen vooraf op CT-vaardigheden, en hoe gemotiveerd waren leerlingen vooraf voor het programmeren?

3.1.1 Programmeerervaring

Ruim de helft van de leerlingen (57%) had voorafgaand aan de interventie nog geen ervaring met programmeren. Een deel van de leerlingen had al wel ervaring: met Scratch (19%), met het programmeren van robots (24%), of met andere programmeeromgevingen. Het percentage leerlingen dat al ervaring had met programmeren was bij jongens een stuk hoger (60%) dan bij meisjes (21%). Er was geen verschil in programmeerervaring tussen leerlingen van verschillende leerjaren. Wel waren er verschillen tussen leerlingen van de verschillende scholen.

Van de leerlingen die al ervaring hadden met programmeren, gaven de meesten aan dat ze slechts af en toe programmeerden of het alleen een keer hadden uitgeprobeerd (zie figuur 3.1). Van de jongens gaf zo'n twintig procent aan regelmatig te programmeren. Bij de meisjes was er slechts één leerling die aangaf regelmatig te programmeren.



Figuur 3.1 – Mate van programmeerervaring voorafgaand aan de interventie (n = 270)

3.1.2 CT-vaardigheden

In de voormeting beantwoordden de leerlingen gemiddeld 58 procent van de CT-opgaven goed. Dit geeft aan dat leerlingen voorafgaand aan de interventie al bepaalde competenties hadden op het gebied van CT. In tabel 3.1 worden per leerjaar de gemiddelde scores op de categorieën van CT-vaardigheden weergegeven. Voor alle categorieën vonden we een significant verschil tussen de leerjaren: leerlingen in groep 7 en 8 scoorden over het algemeen hoger dan leerlingen in groep 5 en 6 (zie tabel 3.1). Er was geen verschil tussen jongens en meisjes in hun CT-vaardigheden. Dit is opvallend, gezien de grote verschillen tussen jongens en meisjes in hun programmeerervaring.

Tabel 3.1 – Gemiddelde score CT-vaardigheden tijdens de voormeting, uitgesplitst naar leerjaar

categorie CT-vaardigheden	gemiddelde score voormeting (percentage correct)				
	groep 5 (n = 30)	groep 6 (n = 71)	groep 7 (n = 62)	groep 8 (n = 71)	totaal (n = 234)
algoritmisch redeneren	48,1	51,3	65,2	67,1	59,4
probleemdecompositie	54,7	46,5	64,5	60,3	56,5
debugging	49,4	43,7	62,1	70,0	57,3
totaal	50,2	47,8	64,1	66,3	58,0

3.1.1 Motivatie voor programmeren

In de voormeting werden verschillende aspecten van de motivatie van leerlingen voor programmeren gemeten: interesse (hoe leuk vinden leerlingen programmeren?), competentiegevoel (hoe competent voelen leerlingen zich wat betreft programmeren?) en waargenomen nut (in hoeverre zien leerlingen het nut van programmeren in?). De interesse voor programmeren was bij de leerlingen gemiddeld behoorlijk hoog (een gemiddelde van 3,8 op een schaal van 1 tot 5). Ook beschouwden leerlingen zich gemiddeld meer dan een beetje competent in programmeren (gemiddelde van 3,5), en vonden ze programmeren vrij nuttig (gemiddelde van 3,4). Op alle drie de aspecten scoorden jongens beduidend hoger dan meisjes (zie tabel 3.2). Voor competentiegevoel en waargenomen nut vonden we een lichte significante correlatie met leerjaar: leerlingen in hogere leerjaren schatten hun programmeercompetentie en het nut van programmeren iets lager in. Hoewel leerlingen uit hogere leerjaren gemiddeld hoger scores op CT-vaardigheden (zie tabel 3.1), schatten zij hun competenties juist wat lager in. Mogelijk hebben deze leerlingen een breder beeld van wat programmeren inhoudt en wat ze daarvan nog niet kunnen.

Tabel 3.2 – Gemiddelde score motivatie voor programmeren tijdens de voormeting, uitgesplitst naar geslacht

schaal motivatie voor programmeren	gemiddelde score voormeting		
	jongens (n = 150)	meisjes (n = 121)	totaal (n = 271)
interesse	4,0	3,6	3,8
competentiegevoel	3,8	3,2	3,5
waargenomen nut	3,6	3,3	3,4

3.2 Ontwikkeling van de CT-vaardigheden

Een belangrijke vraag in het onderzoek was of het leerarrangement rond programmeren heeft bijgedragen aan de ontwikkeling van de CT-vaardigheden van de leerlingen (onderzoeksvraag 1). Om deze vraag te beantwoorden hebben we onderzocht of de CT-vaardigheden van de leerlingen zijn toegenomen van voormeting naar nameting. Tabel 3.3 toont van de leerlingen die zowel de voormeting als de nameting hebben voltooid, de gemiddelde scores op CT-vaardigheden in beide metingen. We zien dat de totaalscore op CT-vaardigheden significant is toegenomen van voormeting naar nameting. Dit geldt ook voor de onderdelen algoritmisch redeneren en debugging. Het lijkt er dus op dat het leerarrangement inderdaad heeft bijgedragen aan de CT-vaardigheden van de leerlingen. Alleen bij het onderdeel probleemdecompositie zien we geen significant verschil tussen voormeting en nameting.

Tabel 3.3 – Verschil in score CT-vaardigheden nameting - voormeting (n = 146)

categorie CT-vaardigheden	voormeting gemiddelde score (% correct)	nameting gemiddelde score (% correct)	nameting - voormeting gemiddelde verschillscore
algoritmisch redeneren *	61,7	66,4	4,6
probleemdecompositie	58,8	57,7	-1,1
debugging *	59,8	64,2	4,3
totaal *	60,4	63,5	3,1

* significant verschil nameting - voormeting

Voor statistische details, zie tabel B.1 in de bijlage.

3.3 Ontwikkeling van de motivatie voor programmeren

We wilden ook weten of het leerarrangement heeft bijgedragen aan de motivatie van leerlingen met betrekking tot programmeren (onderzoeksvraag 2). Dit hebben we op twee manieren onderzocht. Ten eerste hebben we gekeken of de motivatie voor programmeren zoals gemeten met de motivatievragenlijst, is toegenomen van voormeting naar nameting (zie bovenste helft van tabel 3.4). Voor de schaal interesse vonden we geen significant verschil tussen voormeting en nameting. Voor de schalen competentiegevoel en waargenomen nut vonden we dat de gemiddelde score na afloop van de interventie significant lager was dan vóór de interventie. Bij nadere analyses bleek echter dat deze resultaten sterk zijn beïnvloed door één afwijkende school. Op deze school (school 2) zijn de motivatiescores op de nameting opvallend laag in vergelijking met de andere scholen. Dit komt mogelijk doordat er op deze school ten tijde van het onderzoek veel onrust was door leerkrachtwisselingen. Als we school 2 niet in de analyses meenemen zien de resultaten er anders uit (zie onderste helft van tabel 3.4). Nu zien we wat betreft de interesse voor programmeren een significante toename van voor- naar nameting. Leerlingen lijken door de interventie dus meer interesse te krijgen in programmeren. Wat betreft competentiegevoel en waargenomen nut is er nu geen verschil tussen voor- en nameting.

Tabel 3.4 – Verschil in score motivatie voor programmeren nameting - voormeting

schaal motivatie voor programmeren	voormeting gemiddelde score	nameting gemiddelde score	nameting - voormeting gemiddelde verschilscore
leerlingen van alle scholen (n = 229)			
interesse	3,86	3,77	- 0,10
competentiegevoel *	3,59	3,38	- 0,21
waargenomen nut *	3,50	3,38	- 0,12
leerlingen van alle scholen behalve school 2 (n = 143)			
interesse *	3,90	4,08	0,18
competentiegevoel	3,57	3,55	- 0,02
waargenomen nut	3,57	3,60	0,04

* significant verschil nameting - voormeting

Voor statistische details, zie tabel B.2 in de bijlage.

Ten tweede hebben we leerlingen tijdens de nameting gevraagd of zij meer zijn gaan programmeren. Van de 238 leerlingen die deze vraag hebben ingevuld, gaven 94 leerlingen (39%) aan dat zij buiten het leerarrangement om meer zijn gaan programmeren. Van deze leerlingen zijn er 60 meer gaan programmeren in Scratch, 26 leerlingen zijn meer met robots gaan programmeren, en 17 hebben (ook) andere programmeeromgevingen gebruikt. Van de leerlingen die aangaven meer te zijn gaan programmeren, was ongeveer twee derde (68%) een jongen en had bijna de helft (44%) nog geen programmeerervaring vooraf.

3.4 Factoren die een rol spelen bij de ontwikkeling van CT-vaardigheden

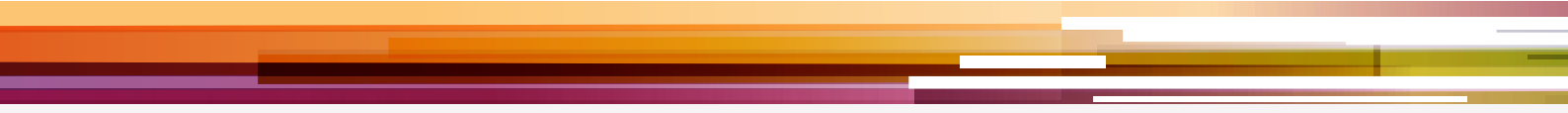
We zagen dat leerlingen na afloop van de interventie hoger scoorden op CT-vaardigheden dan vóór de interventie. We waren ook geïnteresseerd in mogelijke factoren die een rol spelen bij deze ontwikkeling van de CT-vaardigheden (onderzoeksvraag 3). Was de toename in CT-vaardigheden hoger bij leerlingen die met een robot (Edison) hebben leren programmeren dan bij leerlingen die met een beeldschermomgeving (Scratch) leerden programmeren? Hebben leerlingen die de gepersonaliseerde variant van het leerarrangement volgden meer geleerd dan leerlingen die de niet-gepersonaliseerde variant volgden? En wat is de invloed van programmeerervaring, bestaande CT-vaardigheden en motivatie voorafgaand aan de interventie, en van geslacht, leerjaar, en reken- en leesvaardigheid?

Met een regressieanalyse hebben we onderzocht welke van deze factoren voorspellers zijn voor de leerwinst in CT-vaardigheden van voormeting naar nameting (zie tabel B.3 in de bijlage). Uit de analyse blijkt dat de volgende factoren gezamenlijk de leerwinst voorspellen: CT-vaardigheden vooraf, leerjaar, programmeerervaring vooraf en rekenvaardigheid. Ten eerste zien we dat leerlingen die vooraf een lagere score op CT-vaardigheden hadden, meer vooruit gingen tijdens de interventie. Dit is niet verrassend: deze leerlingen hadden meer ruimte voor verbetering. Daarnaast blijkt dat leerlingen in hogere leerjaren van de bovenbouw gemiddeld meer vooruit gingen in hun CT-vaardigheden dan leerlingen in lagere leerjaren van de bovenbouw. Het lijkt erop dat het leerarrangement dus beter aansluit bij leerlingen in de hoogste leerjaren van de bovenbouw (groep 7-8). Ook leerlingen die vooraf al ervaring hadden met programmeren gingen tijdens de interventie meer vooruit in hun CT-vaardigheden. Dit geldt onder constanthouding van de CT-vaardigheden bij de voormeting: leerlingen met gelijke CT-vaardigheden bij de voormeting gingen meer vooruit als zij vooraf al ervaring hadden met programmeren. Door hun eerdere ervaringen pikten deze leerlingen het programmeren mogelijk sneller op. Tot slot blijkt dat ook leerlingen met een hogere rekenvaardigheid meer vooruit gingen in hun CT-vaardigheden (ook hier onder constanthouding van de CT-vaardigheden). Mogelijk helpen rekenvaardigheden bij het leren programmeren.

We vonden geen verschillen in ontwikkeling van CT-vaardigheden tussen leerlingen die met de robot of met de beeldschermomgeving hebben gewerkt, en ook geen verschillen tussen leerlingen die met een gepersonaliseerd of met een niet-gepersonaliseerd leerarrangement werkten. Ook vonden we geen significante verschillen in ontwikkeling van CT-vaardigheden tussen jongens en meisjes of tussen leerlingen met een hogere of lagere initiële motivatie voor programmeren.

3.5 Factoren die een rol spelen bij de ontwikkeling van motivatie

In paragraaf 3.3 zagen we dat de ontwikkeling van de motivatie voor programmeren bij school 2 afweek van de andere scholen. Als we school 2 niet in de analyses meenemen, vinden we van voor- naar nameting een toename in de interesse voor programmeren. We vroegen ons af of er naast de school waar een leerling op zit, nog andere factoren een rol spelen bij de ontwikkeling van de interesse voor programmeren.



Ook hier maakten we gebruik van een regressieanalyse, waarbij we keken welke factoren voorspellers zijn van de toename in interesse van voor- naar nameting (zie tabel B.4 in de bijlage). Uit de analyse blijkt dat naast de school waar een leerling op zit, ook de interesse voor programmeren tijdens de voormeting en de programmeerervaring vooraf voorspellende factoren zijn voor de toename in interesse voor programmeren. We zien dat bij leerlingen met een lagere interessescore op de voormeting de interesse voor programmeren meer toeneemt. Omdat hun interesse vooraf laag was, hadden deze leerlingen hierin meer ruimte voor groei. Verder lieten leerlingen die vooraf al ervaring hadden met programmeren een hogere toename in interesse voor programmeren zien dan leerlingen die nog geen programmeerervaring hadden. Mogelijk sloot de interventie beter aan bij deze leerlingen.

We vonden geen verschil in ontwikkeling van programmeerinteresse tussen leerlingen die een gepersonaliseerd dan wel niet-gepersonaliseerd leerarrangement volgden, of tussen leerlingen die met een robot dan wel met een beeldschermomgeving hebben leren programmeren.

4. DISCUSSIE EN CONCLUSIE

In dit onderzoek werd onderzocht of een leerarrangement rondom programmeren in het basisonderwijs bijdraagt aan de CT-vaardigheden van leerlingen en hun motivatie wat betreft programmeren. Ook werd een aantal factoren onderzocht die hierbij mogelijk een rol kunnen spelen.

4.1 Ontwikkeling van de CT-vaardigheden

De resultaten laten zien dat leerlingen na afloop van de interventie gemiddeld hoger scoorden op CT-vaardigheden dan voorafgaand aan de interventie. Er was een significante leerwinst voor twee van de drie gemeten aspecten van CT: algoritmisch redeneren en debugging. Een leerarrangement rond programmeren lijkt dus inderdaad te kunnen bijdragen aan de CT-vaardigheden van leerlingen. Dit geldt al voor een korte interventie; de interventie in dit onderzoek duurde bij elkaar slechts twee uur.

4.2 Ontwikkeling van de motivatie voor programmeren

De interesse die leerlingen hadden voor programmeren was na afloop van de interventie gemiddeld hoger dan voorafgaand aan de interventie (hierbij werd een afwijkende school uit de analyses weggelaten). Een leerarrangement rond programmeren lijkt dus bij te dragen aan de interesse van leerlingen voor programmeren. Op twee andere aspecten van motivatie voor programmeren – competentiegevoel en waargenomen nut – vonden we geen verschil tussen voormeting en nameting.

Bijna veertig procent van de leerlingen gaf aan dat zij door de interventie meer zijn gaan programmeren buiten de lessen om. De toename die we vonden in interesse voor programmeren, komt dus bij een deel van de leerlingen ook tot uiting in hun gedrag.

4.3 Robot vs. beeldschermomgeving

De leerwinst in CT-vaardigheden verschilde niet voor leerlingen die met een robot dan wel met een beeldschermomgeving hadden geprogrammeerd. Ook vonden we geen verschillen in motivatie tussen leerlingen die met een robot of een beeldschermomgeving hadden gewerkt. Een eventueel voordeel van het fysieke aspect bij leren met een robot kon in dit onderzoek dus niet worden aangetoond. Dit kan te maken hebben met het relatief kleine aantal leerlingen per variant van het leerarrangement. Ook is het mogelijk dat het werken met een robot meer voordeel heeft bij het ontwikkelen van meer complexe aspecten van CT. Dit zou vervolgonderzoek moeten uitwijzen.

4.4 Gepersonaliseerd leren

Leerlingen die met een gepersonaliseerd leerarrangement dan wel een niet-gepersonaliseerd leerarrangement hadden gewerkt, verschilden niet van elkaar in hun ontwikkeling in CT-vaardigheden of motivatie voor programmeren. Ook hier zou het relatief kleine aantal leerlingen per variant van het leerarrangement een rol kunnen spelen. Een andere mogelijke verklaring is dat er in dit onderzoek een relatief beperkte vorm van gepersonaliseerd leren werd gebruikt. Er was alleen sprake van een beperkte differentiatie naar werkvorm; er was geen differentiatie naar bijvoorbeeld niveau of leerinhoud. Ook was er slechts in beperkte mate sprake van zelfregie voor de leerling (keuze uit drie werkvormen). Daarnaast was de rol van de leraar in de interventie minimaal, omwille van de vergelijkbaarheid van de verschillende scholen en klassen. De resultaten van de regressieanalyses wijzen erop dat meer differentiatie naar het niveau van leerlingen een toegevoegde waarde kan hebben voor het leerarrangement.

4.5 Verschillen tussen leerlingen

De leerwinst in CT-vaardigheden en de ontwikkeling in motivatie voor programmeren bleken te verschillen voor verschillende leerlingen. Ten eerste vonden we dat leerlingen uit hogere leerjaren tijdens de interventie meer vooruitgingen in hun CT-vaardigheden dan leerlingen uit lagere leerjaren. Dit wijst erop dat het gebruikte leerarrangement vooral aansluit bij leerlingen uit de hoogste leerjaren in het basisonderwijs (groep 7-8). Voor leerlingen uit lagere leerjaren in de bovenbouw (groep 5-6) is mogelijk een aangepast leerarrangement nodig, met bijvoorbeeld meer instructie of ondersteuning door een leraar of meer oefening. Verder bleek dat leerlingen met een hogere rekenvaardigheid, en leerlingen die al ervaring hadden met programmeren, meer vooruitgingen in CT-vaardigheden. Wat betreft de ontwikkeling in motivatie voor programmeren vonden we ook een voordeel voor leerlingen die al programmeerervaring hadden. Deze bevindingen kunnen worden gezien als een indicatie dat het leerarrangement moeilijk(er) was voor leerlingen met een lagere rekenvaardigheid en leerlingen die nog niet eerder hadden geprogrammeerd. Het zou goed zijn het leerarrangement voor deze groepen leerlingen aan te passen, of bijvoorbeeld te verlengen.

4.6 Conclusies

Het onderzoek levert aanwijzingen dat een leerarrangement rond programmeren kan bijdragen aan de CT-vaardigheden van leerlingen in de bovenbouw van de basisschool. Ook zijn er aanwijzingen voor een positieve bijdrage aan de interesse van leerlingen voor programmeren. Een mogelijk voordeel van het programmeren met robots of van gepersonaliseerd leren kon niet worden aangetoond.

Het gebruikte leerarrangement kan mogelijk worden verbeterd door te zorgen voor een betere aansluiting op de verschillende beginniveaus van leerlingen, zodat leerlingen zonder programmeerervaring en leerlingen met lagere rekenvaardigheden er meer baat bij hebben.

REFERENTIES

- Antle, A. N. (2013). Research opportunities: Embodied child–computer interaction. *International Journal of Child-Computer Interaction*, 1(1), 30–36. <http://doi.org/10.1016/j.ijcci.2012.08.001>
- Bocconi, S., Chiocciariello, A., Dettori, G., Ferrari, A., & Engelhardt, K. (2016). *Developing Computational Thinking in Compulsory Education*. <https://ec.europa.eu/jrc/en/publication/eur-scientific-and-technical-research-reports/developing-computational-thinking-compulsory-education-implications-policy-and-practice>
- Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *AERA*. Geraadpleegd op http://web.media.mit.edu/~kbrennan/files/Brennan_Resnick_AERA2012_CT.pdf
- Marquenie, E., Opsteen, J., Ten Brummelhuis, A., & Van der Waals, J. (2014). *Elk talent een kans Inhoud. Verkenning van gepersonaliseerd leren met ICT. Onderzoeksnotitie voor de VO-raad*. (Vol. 25). http://www.vo2020.nl/wp-content/uploads/2015/02/Onderzoeksnotitie_gepersonaliseerd_leren.pdf
- Papert, S., & Harel, I. (1991). Situating Constructionism. <http://namodemello.com.br/pdf/tendencias/situatingconstrutivism.pdf>
- Pijpers, R. (2015). *Computing-onderwijs in de praktijk - Wat kunnen we leren van de Britten?* Zoetermeer. <https://www.kennisnet.nl/artikel/wat-wij-kunnen-leren-van-computing-onderwijs-in-de-britse-praktijk/>
- Ryan, R. M., & Deci, E. L. (2000). Intrinsic and extrinsic motivations: Classic definitions and new directions. *Contemporary Educational Psychology*, 25(1), 54-67.
- Self-Determination Theory (z.d.). *Intrinsic Motivation Inventory (IMI)*. <http://selfdeterminationtheory.org/intrinsic-motivation-inventory/>
- van Loon, A.-M., van der Neut, I., De Ries, K., & Kral, M. (2016). *Dimensies van gepersonaliseerd leren*. Nijmegen. <http://ixperium.nl/files/2014/08/dimensies-gepersonaliseerd-leren.pdf>
- Wing, J. M. (2010). Computational Thinking: What and Why? *The Link - The Magazine of the Carnegie Mellon University School of Computer Science*. <http://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why>

BIJLAGE: STATISTISCHE DETAILS

Tabel B.1 – Verschil in score CT-vaardigheden (percentage correct) nameting - voormeting (n = 146)

categorie CT-vaardigheden	score voormeting	score nameting	verschilscore nameting - voormeting		
	M (SD)	M (SD)	M (SD)	t	d
algoritmisch redeneren	61,7 (19,5)	66,4 (18,7)	4,6 (18,1)	3,10 **	-0,26
probleemdecompositie	58,8 (22,8)	57,7 (25,1)	-1,1 (24,7)	-0,54	-0,04
debugging	59,8 (28,3)	64,2 (28,6)	4,3 (26,0)	2,02 *	-0,17
totaal	60,4 (17,9)	63,5 (18,3)	3,1 (13,1)	2,88 **	-0,24

M = gemiddelde, SD = standaarddeviatie, t = waarde t-toets, d = effectgrootte

* significant verschil nameting - voormeting ($p < .05$, éénzijdig)

** significant verschil nameting - voormeting ($p < .01$, éénzijdig)

Tabel B.2 – Verschil in score motivatie voor programmeren nameting - voormeting

schaal motivatie voor programmeren	score voormeting	score nameting	verschilscore nameting - voormeting		
	M (SD)	M (SD)	M (SD)	t	d
leerlingen van alle scholen (n = 229)					
interesse	3,86 (0,81)	3,77 (1,05)	-0,10 (1,00)	-1,48	-0,10
competentiegevoel	3,59 (0,78)	3,38 (0,96)	-0,21 (0,89)	-3,63 **	-0,24
waargenomen nut	3,50 (0,84)	3,38 (0,99)	-0,12 (0,86)	-2,15 *	-0,14
leerlingen van alle scholen behalve school 2 (n = 143)					
interesse	3,90 (0,85)	4,08 (0,88)	0,18 (0,87)	-2,46 *	-0,21
competentiegevoel	3,57 (0,79)	3,55 (0,89)	-0,02 (0,86)	-0,28	-0,02
waargenomen nut	3,57 (0,90)	3,60 (0,89)	0,04 (0,86)	-0,51	-0,04

M = gemiddelde, SD = standaarddeviatie, t = waarde t-toets, d = effectgrootte

* significant verschil nameting - voormeting ($p < .05$, tweezijdig)

** significant verschil nameting - voormeting ($p < .01$, tweezijdig)

Tabel B.3 - Verklarend regressiemodel voor verschilscore CT-vaardigheden nameting - voormeting (n = 139)

	<i>B</i>	<i>Std. fout</i>	<i>t</i>
(intercept)	- 4,33	2,67	- 1,63
Score CT-vaardigheden voormeting	- 0,49	0,07	- 6,69 **
School 3 ^a	0,56	0,52	1,07
School 4 ^a	- 0,69	0,84	- 0,81
School 5 ^a	- 0,81	1,30	- 0,62
Meisje (vs. jongen)	0,58	0,43	1,33
Leerjaar	0,76	0,24	3,11 **
Programmeerervaring vooraf	0,99	0,43	2,30 *
Rekenen LR%	0,02	0,01	2,08 *
Begrijpend lezen LR%	0,01	0,01	1,51
Motivatiescore interesse voormeting ^b	0,37	0,29	1,28
Leerarrangement met robot (vs. beeldschermomgeving)	0,40	0,40	0,99
Gepersonaliseerd leerarrangement (vs. niet-gepersonaliseerd)	0,04	1,22	0,03

B = regressiecoëfficiënt, *Std. fout* = standaardfout, *t* = t-waarde.

$R^2 = 0,32$ (32% verklaarde variantie).

^aReferentiecategorie = School 2. School 1 kon niet worden meegenomen in deze analyse, omdat de leerlingen van deze school niet alle metingen (op tijd) hadden voltooid.

^bVanwege multicollineariteit van de verschillende schalen uit de motivatievragenlijst werd in de regressieanalyse alleen de schaal interesse meegenomen.

* significante voorspeller van verschilscore CT-vaardigheden ($p < .05$, tweezijdig)

** significante voorspeller van verschilscore CT-vaardigheden ($p < .01$, tweezijdig)

Tabel B.4 - Verklarend regressiemodel voor verschilscore interesse nameting - voormeting (n = 188)

	<i>B</i>	<i>Std. fout</i>	<i>t</i>
(intercept)	2,04	0,92	2,23 *
Motivatiescore interesse voormeting	-0,53	0,09	- 5,62 **
School 1 ^a	0,97	0,50	1,92
School 3 ^a	0,77	0,19	3,98 **
School 4 ^a	0,83	0,31	2,64 **
School 5 ^a	1,02	0,49	2,09 *
Meisje (vs. jongen)	-0,10	0,15	-0,67
Leerjaar	-0,09	0,08	-1,16
Programmeerervaring vooraf	0,29	0,15	2,00 *
Rekenen LR%	-0,00	0,00	-0,40
Begrijpend lezen LR%	0,00	0,00	0,25
Score CT-vaardigheden voormeting	-0,01	0,02	-0,34
Leerarrangement met robot (vs. beeldschermomgeving)	-0,00	0,14	-0,01
Gepersonaliseerd leerarrangement (vs. niet-gepersonaliseerd)	0,16	0,45	0,36

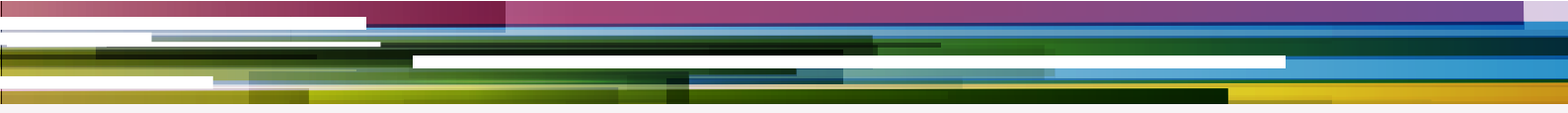
B = regressiecoëfficiënt, *Std. fout* = standaardfout, *t* = t-waarde.

$R^2 = 0,30$ (30% verklaarde variantie).

^aReferentiecategorie = School 2.

* significant voorspeller van verschilscore interesse ($p < .05$, tweezijdig)

** significant voorspeller van verschilscore interesse ($p < .01$, tweezijdig)



iXperium/Centre of Expertise Leren met ict

‘Het leren van morgen is een leven lang gepersonaliseerd leren in een door technologie ondersteunde sociale leeromgeving.’

Het iXperium/CoE is een netwerkorganisatie van het onderwijswerkveld (po, vo, mbo en hbo), lerarenopleidingen en onderzoek, op het gebied van leren en lesgeven met ict. We ontwikkelen en delen kennis, inspireren en professionaliseren leraren, docenten en leidinggevenden om ict in te zetten in het onderwijs. We doen praktijkgericht onderzoek om gepersonaliseerd leren met inzet van ict én ict geletterdheid in het onderwijs verder te ontwikkelen. We onderzoeken de rol van technologie in het onderwijs in drie onderzoekslijnen:

1. Leren met ict als middel, ten behoeve van gepersonaliseerd leren
2. De organisatie, op micro-, meso- en macroniveau, van gepersonaliseerd leren
3. Leren met ict als doel: ict-geletterde leerders en werkers opleiden voor de snel veranderende samenleving

Het iXperium/CoE is een netwerkorganisatie van het HAN lectoraat ‘Leren met ict’, schoolbesturen in de brede regio en de lerarenopleidingen van de HAN.

Kijk voor meer informatie op:

www.ixperium.nl

Volg ons op:

 [Facebook.com/ixperium](https://www.facebook.com/ixperium)

 [@ixperium](https://twitter.com/ixperium)