

The background of the slide is a screenshot of the Scratch programming environment. It shows a stage with a grid and a background of binary code. Several Scratch blocks are visible, including 'CB' (Control) blocks with gear icons, '4' (Number) blocks, and '1,000' (Number) blocks. The text is overlaid on this background.

De ontwikkeling van Computational Thinking skills via SRA-programmeren

ORD, 28 juni 2019

Nardie Fanchamps MSc, PhD-candidate Open University Netherlands Welten TELI
Michelle Petrus, leerkracht primair onderwijs / Master student Onderwijswetenschappen OU

Programmeren in onderwijs

In Nederland en de ons omringende landen is veel belangstelling voor programmeren met leerlingen in het onderwijs.

Onderwerp van 21st century skills / de Human Capital Agenda (Delta, 2015) en relevant voor de ontwikkeling van kinderen met name voor wat betreft het ontwikkelen van denkvaardigheden die samengevat ook wel computational thinking genoemd worden (Wing, 2006).

Omvat een concrete reeks instructies die een computer uitvoert en die de gebruiker spiegelt aan de vooraf gestelde opbrengst en eisen om opdrachten uit te voeren of om problemen op te lossen.

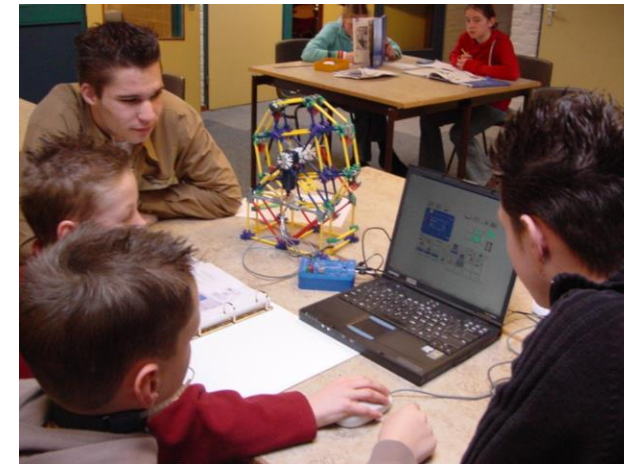


Programmeren in onderwijs

Basisscholen willen gestalte geven aan de ontwikkeling van het leren omgaan met deze vaardigheid vanuit authentieke leeromgevingen.

Echter, er is nog weinig kennis beschikbaar met betrekking tot het aanleren en begeleiden van vaardigheden rondom computational thinking en aan welke eisen een programmeer-leeromgeving moet voldoen.

In het kader van promotieonderzoek doen wij diverse deelstudies.



Focus onderzoek en deelstudies

Identificeren educatieve interventies die binnen een robotica SRA-programmeeromgevingen stimuleren om meetbaar hogere cognitieve competenties te bereiken.

Verhelderen impact van het leren programmeren van robots op de cognitieve ontwikkeling waarbij programmeren met sensoren andere authentieke leeromgevingen vereist die tevens flexibel zijn.

Illustreer aan welke eisen een SRA-programmeeromgeving moet voldoen, wat daarin mogelijke variaties zijn, welke interventies hoofdzakelijk de boventoon voeren, welke instructiestijl het meest effectief kan zijn om probleemoplossend te kunnen programmeren.

Uitgangspunten onderzoek

Het leren programmeren met sensoren **vereist veelzijdige, andere authentieke leeromgevingen** die flexibel zijn. In die richting; ook leerkrachten leren andere vaardigheden als ze in veranderende omgevingen komen.

Niet om een specifieke taal/omgeving te leren, maar om door het gebruiken van een of meerdere talen/omgevingen **overstijgende inzichten en vaardigheden** ook wel ***Computational Thinking skills*** genoemd te ontwikkelen.

Leerlingen al op jonge leeftijd laten **kennismaken met werkingsprincipes** van programmeerbare computertechnologie.

Dat leerlingen die toepassingen zoals, apps, simulaties, games en robots programmeren, vaardigheden en inzichten met betrekking tot 'computational thinking' ontwikkelen (SLO, 2016). **Maar nog (te) weinig wetenschappelijk bewijs.**

Het ontwikkelen van CT vaardigheden is echter geen vanzelfsprekendheid. Het vraagt dat leerlingen en leraar in onderlinge gesprekken samen expliciteren hoe en waarom ze bepaalde keuzes maken (Slangen, 2016). Met andere woorden de principes worden meer manifest gemaakt.

Computational Thinking

- Het procesmatig (her)formuleren van problemen op een zodanige manier dat het mogelijk wordt om met computertechnologie problemen op te lossen wordt binnen 21st Century Skills aangeduid als *Computational thinking* genoemd (Kennisnet, 2016).
- Omvat een verzameling van denkprocessen waarbij o.a. probleemformulering, gegevensorganisatie, -analyse en -representatie gebruikt worden voor het oplossen van problemen met behulp van ICT.
- Informatietechnologie en programmeren kan daarbij als interventiemiddel vanuit betekenisvolle contexten en omgevingen gebruikt worden.

Computational thinking skills

- **Problemen (her)formuleren:** Kan mogelijke oplossingen analyseren om de meest kansrijke richting te bepalen, Kan procesmatig relevante gegevens verzamelen
- **Gegevens verzamelen:** verzamelen van relevante informatie via verschillende bronnen zoals artikelen, interviews, enquêtes of literatuurstudie;
- **Gegevens analyseren:** logisch ordenen en analyseren van gegevens, begrijpen van gegevens, vinden van patronen, trekken van conclusies, het evalueren van grafieken of het toepassen van relevante statistische methodes;
- **Gegevens visualiseren:** weergeven van gegevens door middel van bijvoorbeeld modellen van de werkelijkheid, of door grafieken, tabellen, woorden en plaatjes en het selecteren van de effectiefste representatie;
- **Probleem decompositie:** opdelen van een taak in kleinere, overzichtelijke taken, zoals bijvoorbeeld het opdelen van een lange lijst met opdrachten in subcategorieën van typen opdrachten en het plannen van een project door middel van deelprojecten;
- **Abstractie:** reduceren van de complexiteit van een probleem om te komen tot de kern van het probleem;
- **Algoritmes en procedures:** gebruiken van een serie geordende stappen om een probleem op te lossen of een bepaald doel te bereiken;
- **Automatisering:** door een computer laten uitvoeren van zichzelf herhalende of eentonige taken totdat een oplossing is bereikt;
- **Simulatie en modellering:** weergeven van een model of een proces, of het uitvoeren van een experiment op basis van dat model of proces;
- **Parallellisatie:** zorgen voor gelijktijdige uitvoering van taken om een gezamenlijk doel te bereiken.

SRA-cyclus

- Een robot die op basis van sensorgebruik waarnemingen registreert (**sense**),
- deze waarnemingen vervolgens vergelijkt met interne waarden van de externe situatie en beslist welk pad gevolgd moet worden (**reason**),
- het daaropvolgende proces waarin het programma de robot, op basis van de sensorische input, moet “vertellen” hoe de robot moet reageren (**act**).

Onderwerp van (deel)onderzoek

Focus op gehanteerde instructiestijl, begeleidingsinterventie van de leerkracht, variatie in aanbod, eisen aan materialen en omgeving.

[Link](#)

Opzet en uitvoering onderzoek

- Leerkrachten basisonderwijs volgen professionaliseringsbijeenkomsten m.b.t. programmeren.
- Vervolgens bieden leerkrachten in diverse sessies de door leerlingen op te lossen programmeerproblemen aan.
- Leerlingen lossen in tweetallen de programmeerproblemen op. De leerkracht instrueert, begeleidt, biedt guidance en intervenueert vanuit de eigen identiteit.
- Perceived monitoring en scaffolding wordt bij zowel de leerkracht als bij de leerlingen gemeten (TAFL-Q / SAFL-Q).
- De gehanteerde instructiestijl en interventies worden gescoord en gecategoriseerd evenals CT-skills.
- De impact van de gehanteerde omgeving wordt in kaart gebracht.



Doel van het onderzoek

Adresseren of de gehanteerde instructievariant, de gebruikte omgeving en de gehanteerde begeleidingsinterventie van invloed is op het correct kunnen programmeren van Lego robotica waardoor leerlingen computational thinking skills ontwikkelen.



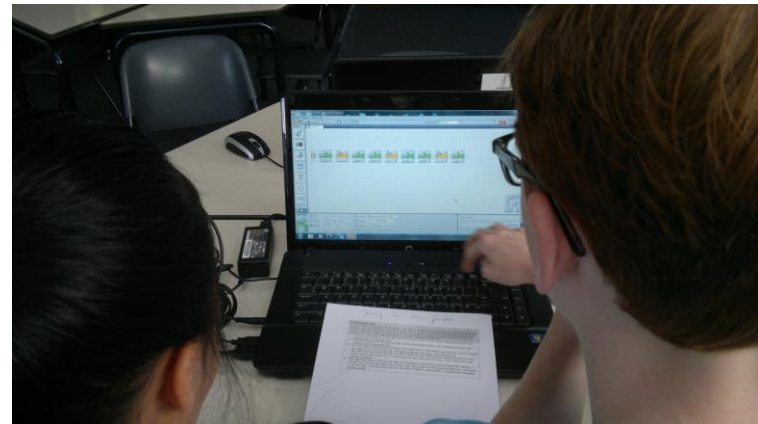
Doelgroep

- Leerkrachten primair onderwijs per cluster van diverse basisscholen.
- Leerlingen van alle jaargroepen werken in tweetallen.
- Schoolbrede ontwikkeling en professionalisering.



Variabelen in het onderzoek

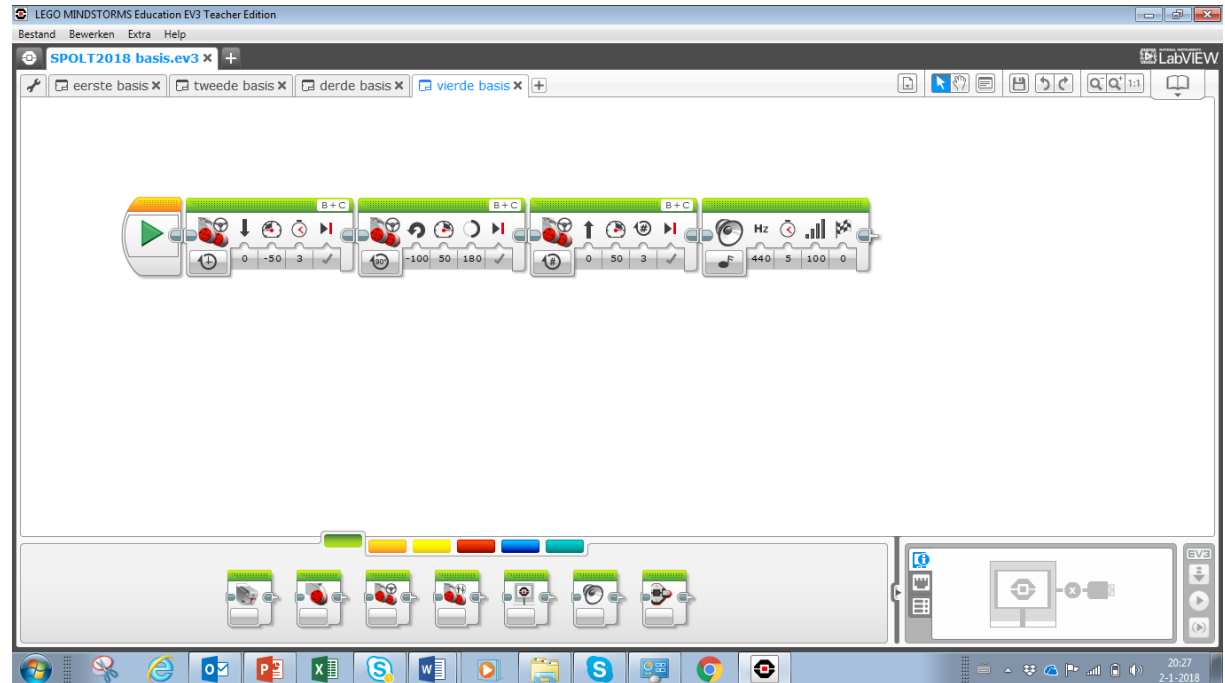
- Sense, Reason and Act programmeren (SRA)
- Gehanteerde begeleiding- en interventiesystemathiek
- Variatie in programmeeromgeving
- Percieved Monitoring & Scaffolding
- Computational thinking skills



Gebruikte materialen

- Lego Mindstorms EV-3

[Link](#)



Leren programmeren met SRA

- Basisinstructie om te komen tot lineair programmeren
- Diverse programmeeropdrachten
- Lego-challenge met 5 verschillende opdrachten



Eerste bevindingen

Eerste opbrengsten

- Leerkrachten schatten hun mate van “perceived monitoring” en “perceived scaffolding” bij programmeren te hoog in. Dit is een direct vergelijk met de beleving van leerlingen.
- Een gehanteerde instructie-systematiek volgens “scaffolding” brengt leerlingen verder in de ontwikkeling van computational thinking skills dan directe instructie.
- De boventoon in begeleidingsinterventie wordt gevormd door het stellen van vragen, het geven van hints en het voordoen/nadoen.
- Opvallend is het maar marginaal weinig voorkomt dat de leerkracht het daadwerkelijk “overneemt”.
- De omgeving lijkt van invloed te zijn op het SRA-denken. [Link](#)

Eerste opbrengsten

- Leerkrachten zijn verbaasd over de ontwikkeling en het potentieel dat leerlingen laten zien.
- SRA-programmeren is voor leerlingen van groep 1 t/m groep 8 functioneel mogelijk.
- Aspecten van sociaal/emotioneel leren en effecten van coöperatief leren blijken spontaan te ontstaan.
- Opvallend dat de betrokkenheid, motivatie en beleving uitzonderlijk hoog is; ook gedurende langere tijd bij alle jaargroepen.
- Geen waarneembare verschillen tussen meisjes en jongens. [Link](#)

Zelf proberen

Workshop Lego EV-3

Zelf programmeren met Lego EV-3

- Nu de gelegenheid om zelf te programmeren met Lego EV-3
- Plenair verkennen van de mogelijkheden van deze omgeving
- Gezamenlijk werken aan uitdagende programmeeropdrachten

Afronding en vragen?

Discussie?

Programmeren draagt absoluut bij aan de ontwikkeling van computational thinking skills

[Link](#)

Dank voor uw aandacht!

Nardie Fanchamps MSc, Promovendus Open Universiteit Welten TELI
Michelle Petrus, Masterstudente Open Universiteit