

Programmeren en robots in de klas:

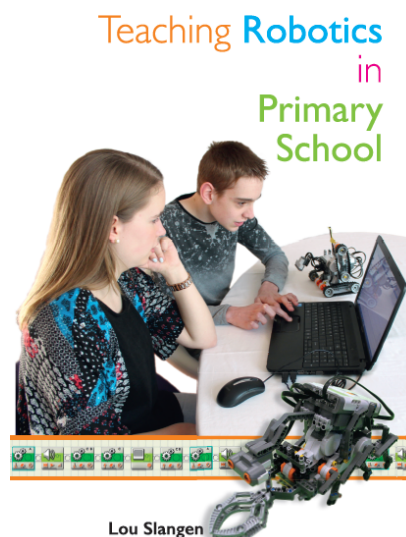
N de
nieuwste
pabo
FONTYS & ZUYD

Wat kinderen er van leren!

Dé NWT-conferentie voor het basisonderwijs

Lunteren 16-11-2018

Lou Slangen PhD
Associate lector Science and Technology education in the primary.
de Nieuwste Pabo.



In basisonderwijs zien we

N de
nieuwste
pabo
FONTYS & ZUYD

toenemende interesse voor:

- Onderwijzen en leren van STEM (Science, Technology, Engineering, Math) topics.
- Toepassingen zoals robots en programmeren zijn populaire thema's.



Maar waarom robotica en programmeren?



Waarom zou je hieraan in de basisschool zoveel aandacht besteden als het onderwijsprogramma al zo overvol is?

Welke argumenten, redenen, voorbeelden zijn aan te dragen?

Noteer voor jezelf twee drie aspecten



Maar waarom robotica en programmeren?



Waarom zou je hieraan in de basisschool zoveel aandacht besteden als het onderwijsprogramma al zo overvol is?

De verwachting is dat leerlingen door het ontwerpen, maken en programmeren van toepassingen zoals, apps, simulaties, games en robots, overstijgende inzichten en vaardigheden ontwikkelen.

Dat zijn toepassing overstijgende inzichten of vaardigheden zoals:

Technische concepten (Slangen, 2016)

Computational thinking skills (SLO, 2016).



Waarom programmeren, wat heeft voor zin?



- *“Everybody in this country should learn how to program a computer... because it teaches you how to think”*
Steve Jobs
- *“Programmeren is de vaardigheid van de toekomst: het bevordert creativiteit, leert je samenwerken met mensen over de hele wereld en laat je communiceren in een universele taal!”* **Neelie Kroes ‘Young Advisors’**
- *Programmeren leert kinderen creëren in plaats van consumeren. Het is dus niet per definitie bedoeld om kinderen op te leiden tot programmeur, maar om ze vaardigheden te leren waarvan ze in elk beroep profijt hebben.* Website **www.mediawijsheid.nl**
- *.... computational thinking, komt er in het kort op neer dat je kunt nadenken over de vraag hoe je een probleem met een computer kunt oplossen. Programmeren is daar een middel voor.* **PO-raad 2017**
- **Welke programmeertaal of programmeeromgeving moet je dan leren? Er zijn er duizenden in soorten en functies.**

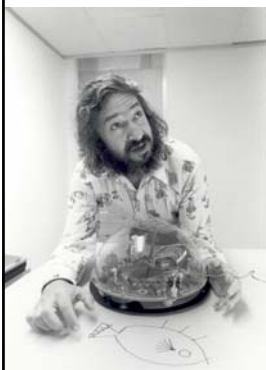
Robots, talen/omgevingen, sites



- | | | |
|---|---|--|
| <ul style="list-style-type: none"> • Coderups • Beebot • Bluebot (Tactile reader) • Probot • Dash&Dot • Lego WeDo (1&2) • Lego Mindstorms (RCX, NXT, EV3) • Arduino • Micro:Bit • Raspberry Pi • Fischertechnik Robotics • Artec Robotica • ROBOTRON Robotica • Intelligent | <ul style="list-style-type: none"> • Jimu Robot • Ozobot (Bit, EVO) • K'nex Robotics • Little Bits • MaKey MaKey • OhBot • Primo-Cubetto • BB-8™ by Sphero • Thymio • UARO • Engino Robotics • Makeblock Robotica • Meccano Meccanoid • VEX | <ul style="list-style-type: none"> • Scratch ,Scratch junior • Blockly, OzoBlockly • Osmo coding • Python • Javascript • C • Ruby • Specifieke talen, omgevingen specifiek voor de robot <p> http://teach2code.nl www.codeuur.nl/ https://code.org/ www.coderdojo.nl http://microbit.org/nl/code/ http://www.coderminds.nl/ http://hetlabrotterdam.nl/robotica/ http://ozoblockly.com/ www.bomberbot.nl </p> |
|---|---|--|

Seymour Papert - Turtle Logo - 1980

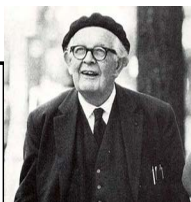
N de
nieuwste
pabo
FONTYS & ZUYD



Papert was van 1967 tot 1981 co-directeur bij het Artificial Intelligence Laboratory van het MIT (Massachusetts Institute of Technology).

Grondlegger programmeren
door kinderen op school

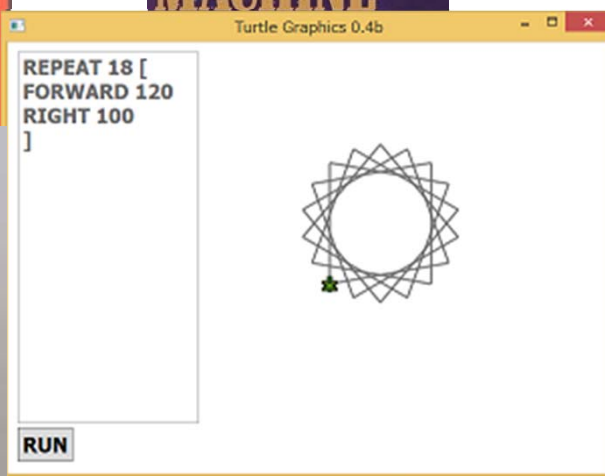
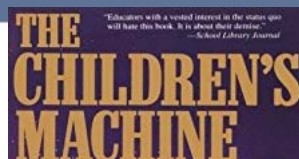
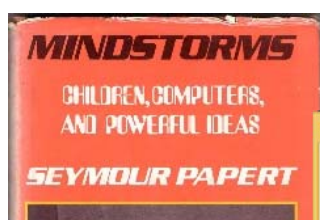
Papert, werkte in Geneve samen met Piaget die hem inspireerde met het constructivistisme.



Papert ontwikkelde deze theorie verder in het zogenaamde *constructionisme*, waarbij de notie werd toegevoegd dat leren beter wordt wanneer leerlingen een "betekenisvol product construeren" bv. met behulp van computers of robots.

Seymour Papert - Turtle Logo - 1980

N de
nieuwste
pabo
FONTYS & ZUYD



Technische concepten mbt. robots (Slangen, 2016)



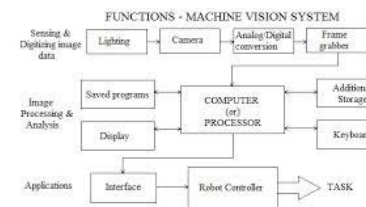
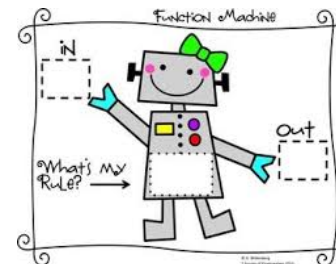
N de
nieuwste
pabo
FONTYS & ZUYD

Functie: een robot is een apparaat dat is bedoeld om bepaalde activiteiten uit te voeren die zijn afgeleid van problemen, behoeften of andere uitdagingen - (doel-middel / vorm-functie).

Systeem: een robot is een groep onderling verbonden delen die als collectief zijn ontworpen om een doel te bereiken op basis van een fundamentele structuur van het realiseren van invoer, processen en uitvoer - (interactie en interdependentie).

Besturing: een robot wordt bestuurd door middel van software die algoritmen gebruikt om taken uit te voeren binnen de mogelijkheden en beperkingen van de materiële (digitale) constructie - (controle hebben).

Sense-reason-act loop: de prestaties van een robot zijn gebaseerd op de drie basisvermogens waarnemen (S), redeneren (R) en acteren (A), die achtereenvolgens worden herhaald en de zogenaamde S-R-A-lus vormen - (sequentieel en parallel programmeren).



Computational thinking skills (zie lijst)

N de
nieuwste
pabo
FONTYS & ZUYD

- Problemen (her)formuleren
- Gegevens verzamelen
- Gegevens analyseren
- Gegevens visualiseren
- Probleem decompositie
- Abstracties maken
- Algoritmes en procedures
- Automatisering
- Simulatie en modellering
- Parallelisatie



Ozobot



Scratch



Lego Mindstorms



Coderups



Lego WeDo



Bee-Bot

Computational thinking skills (zie lijst)



- **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.

Computational thinking skills



- Problemen (her)formuleren
- Gegevens verzamelen
- Gegevens analyseren
- Gegevens visualiseren
- Probleem decompositie
- Abstracties maken
- Algoritmes en procedures
- Automatisering
- Simulatie en modellering
- Parallellisatie



Ozobot



Scratch



Lego Mindstorms



Coderups



Lego WeDo



Bee-Bot

Deze vaardigheden in veel verschillende situaties inzetten, toepassen en oefenen maakt ze wendbaar. Leerlingen gaan ze geleidelijk aan steeds beter hanteren.

Programmeren van robots



- Het oplossen van programmeerproblemen doet een beroep op het kritisch en creatief denken.
- Het zoeken naar oplossingen betekent dat leerlingen inzicht krijgen hoe een reeks en volgorde van instructies eruit moet zien om het probleem op te lossen.
- Leerlingen leren denken in stappen, sequentiële en parallelle processen, variabelen en parameters, verkortingen (herhalingen en functies) en volgordes, ze redeneren in oorzaak en gevolg, in vergelijken en beslissen.

Welke vaardigheden



Uitgangspunt inzichten en vaardigheden leren via robots en programmeren voor alle leerjaren

Overdenken wat leerlingen leren bij de verschillende producten

| | |
|----------------------|---|
| Groep 1-2-3 | Bee-bot Coderups |
| Groep 2-3-4 | Scratch junior Blue-bot |
| Groep 3-4-5 | Ozobot evo Lego we do (krokodil) |
| Groep 5-6-7-8 | Scratch Lego mind storm EV 3 Ozoblockly.com |

De verwachting



Het ontwikkelen van zulke inzichten en vaardigheden is geen vanzelfsprekendheid.

Het vraagt dat leerlingen en leraar in onderlinge (vraag)gesprekken samen **expliciteren hoe en waarom** ze bepaalde keuzes maken.

Met andere woorden de **principes worden meer manifest** gemaakt.

Waarbij het enerzijds gaat ook over **coderen en programmeren** maar anderzijds ook over **technische concepten**.

De leraar moet daarvoor zelf over kennis, vaardigheden en begeleidingstaal beschikken



begeleiden van het proces



Rol van de leerkracht is in gesprek met leerlingen technische concepten en computational skills bewust toe te passen in veel verschillende situaties.

- *Vragen naar persoonlijk oordeel:* "Vind je dit een efficiënte programmering?"
- *Luisteren en begrijpen stimuleren:* "Kun je uitleggen waarom je voor richting 180 kiest?"
- *Laten aanvullen:* "Kun je het probleem in nog meer stapjes verdelen?"
- Vragen naar tegenstrijdigheden of overeenkomsten. "Wie heeft een andere aanpak die ook werkt?"
- *Leerprocessen modelleren via het zichtbaar maken van eigen denken:* "Ik zou me afvragen hoe ik de Bee-Bots tegelijk kan laten draaien?"
- *Het denkproces coachen via hints:* "Wanneer raakt de bal de olifant?"
- *De aanpak een ander perspectief of focus geven:* "Hebben we al gekeken of we de bal naar beneden kunnen laten bewegen?"
- *Bepaalde reacties van leerlingen benadrukken:* "We kunnen ook het voorstel van Jos testen en instellen dat de bal terugkaatst."
- *Zorgen voor steeds nieuwe uitdagingen die de vaardigheden activeren;* "Kunnen we ook het volgende probleem met de Bee-Bot oplossen."

Bee-Bot



ter voorbeeld



De Bee-Bot is een “robot” die een voorgeprogrammeerde instructie-sequentie kan onthouden en op commando deze kan uitvoeren door in stapjes naar een plek op een speelveld te gaan.

- Bewust worden van het bestaan van algoritmes en procedures om een beoogd doel te bereiken
- En passant leren kinderen bepaalde inhouds of laten ze zien dat ze deze beheersen (nevendoeel)



Bee-Bot



CT skills



De Bee-Bot is een “robot” die een voorgeprogrammeerde instructie-sequentie kan onthouden en op commando deze kan uitvoeren door in stapjes naar een plek op een speelveld te gaan.

- Bewust worden van het bestaan van algoritmes en procedures om een beoogd doel te bereiken
 - En passant leren kinderen bepaalde inhouds of laten ze zien dat ze deze beheersen (nevendoeel)
- Problemen (her)formuleren
 - Gegevens verzamelen
 - Gegevens analyseren
 - Gegevens visualiseren
 - **Probleem decompositie**
 - Abstracties maken
 - **Algoritmes en procedures**
 - Automatisering
 - Simulatie en modellering
 - Parallellisatie

Welke computational skills / technische concepten worden aangesproken?

N de
nieuwste
pabo
FONTYS & ZUYD

Computational skills

- Problemen (her)formuleren
- Gegevens verzamelen
- Gegevens analyseren
- Gegevens visualiseren
- Probleem decompositie
- Abstracties maken
- Algoritmes en procedures
- Automatisering
- Simulatie en modellering
- Parallellisatie



Technische concepten

- Functie
- Systeem
- Besturing
- Sense Reason Act

Lego Mindstorms



Coderups



Scratch

Bee-Bot



Ozobot



Lego WeDo

Bee-Bot



verdieping

N de
nieuwste
pabo
FONTYS & ZUYD

De Bee-Bot is een "robot" die een voorgeprogrammeerde instructie-sequentie kan onthouden en op commando deze kan uitvoeren door in stapjes naar een plek op een speelveld te gaan.

- Bewust worden van het bestaan van algoritmes en procedures om een beoogd doel te bereiken
- En passant leren kinderen bepaalde inhoud of laten ze zien dat ze deze beheersen (nevendoel)
- Via welke opdracht zou je de Bee-Bot kunnen gebruiken om leerlingen andere computational skills laten bereiken
- Enkele voorbeelden uitgewerkt
- Problemen (her)formuleren
- Gegevens verzamelen
- Gegevens analyseren
- Gegevens visualiseren
- **Probleem decompositie**
- Abstracties maken
- **Algoritmes en procedures**
- Automatisering
- Simulatie en modellering
- Parallellisatie

- Twee kinderen laten ieder een eigen Bee-Bot op A2 en op A3 tegelijk starten.
- A2 gaat via de kortste weg naar E4 en A3 gaat via de kortste weg naar E1.
- Ze mogen elkaar niet raken en moeten tegelijk aankomen.
- Hoe kunnen ze dat het beste aanpakken en wat is de route?



| | 1 | 2 | 3 | 4 |
|---|----------|---------------|---------------|----------|
| A | | Start naar E4 | Start naar E1 | |
| B | | | | |
| C | | | | |
| D | | | | |
| E | Einde A3 | | | Einde A2 |

Welke vaardigheden vraagt de opdracht?

- Twee kinderen laten ieder een eigen Bee-Bot op A2 en op A3 tegelijk starten.
- A2 gaat via de kortste weg naar E4 en A3 gaat via de kortste weg naar E1.
- Ze mogen elkaar niet raken en moeten tegelijk aankomen.
- Hoe kun je dat aanpakken?



- **Probleem herformulering**
- **Probleem decompositie**
- **Abstraheren**
- **Parallellisatie**
- **Algoritmisch denken**
- **Samenwerken**



| | 1 | 2 | 3 | 4 |
|---|----------|---------------|---------------|----------|
| A | | Start naar E4 | Start naar E1 | |
| B | | | | |
| C | | | | |
| D | | | | |
| E | Einde A3 | | | Einde A2 |

Bedenk nieuwe opdrachten?



Laat vijf kinderen samen de Bee-Bots zo programmeren dat deze een lijndans uitvoeren

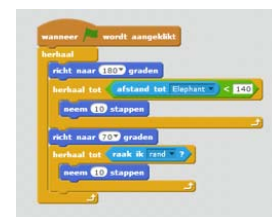
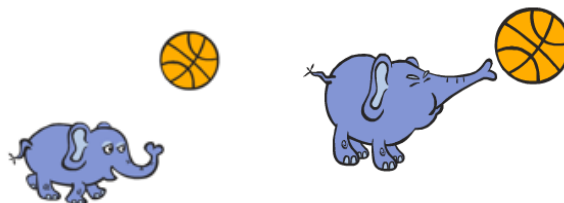


| | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| A | |  |  | |
| B | | | | |
| C | | | | |
| D | | | | |
| E | | | | |

Er zijn veel geschikte producten

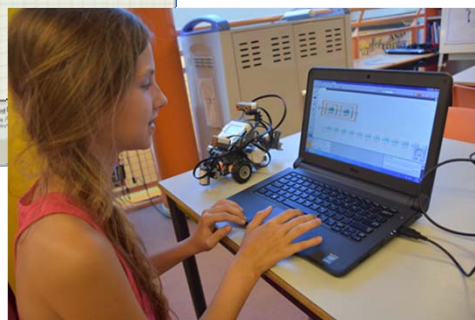
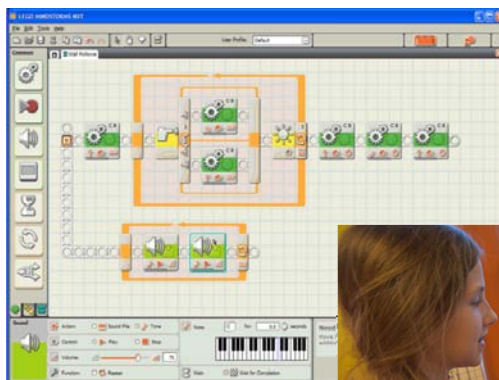


- Als ze leerlingen aanzetten tot ontwikkelen van een of meer van Computational Thinking Skills
- Als de leerkracht de leerlingen kan begeleiden met het expliciteren (conceptualiseren) van de ontwikkelde inzichten



Speed introductie Lego Mindstorms

N de
nieuwste
pabo
FONTYS & ZUYD



Stellingen:

N de
nieuwste
pabo
FONTYS & ZUYD

1. Leerkrachten hebben voldoende inzicht in de essenties van robotica en programmeren om leerlingen hierbij te begeleiden.
2. Het heeft geen zin om robotica en programmeren in school te introduceren zolang leerkrachten geen zicht hebben op de achterliggende leerdoelen.
3. Wanneer leerkrachten zichzelf programmeren en robotica op een onderzoekende en ontwerpende manier eigen maken zijn ze in staat leerlingen adequaat te begeleiden.
4. Het begeleiden van het leren van robotica of programmeren vraagt coachende vaardigheden die leerkrachten vreemd zijn.

Vragen?



Literatuur:

- Slangen, L. (2016). *Teaching Robotics in Primary School*. (PhD), Eindhoven University of Technology, Eindhoven.
- Slangen, L., & Rohaan, E. (2018). Programmeren en robotica. In Tycho Malmberg, Ellen Rohaan, Sara van Duijn, & R. Klapwijk (Eds.), *Onderzoekend en ontwerpend de wereld ontdekken, Natuur & Techniek in het basisonderwijs* (pp. 257-273). Groningen/Houten: Noordhoff Uitgevers.SLO. (2016).
- SLO Curriculum van de toekomst; Leerlijn programmeren stoomt kinderen klaar voor de toekomst. Retrieved from <http://curriculumvandetoekomst.slo.nl/nieuws-overzicht/leerlijn-programmeren-stoomt-kinderen-klaar-voor-de-toekomst>

