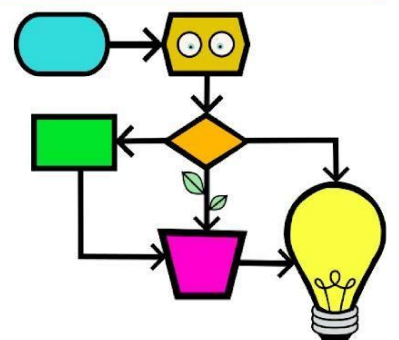
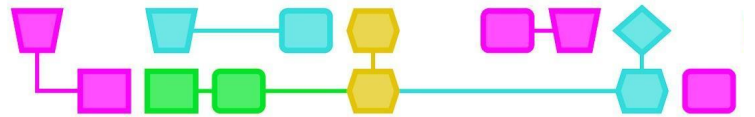


Introductie
Abstractie





Samenvatting

Met deze activiteit wordt het begrip abstractie uitgelegd, een van de vier grondbeginselen van *computational thinking*. Abstractie betekent in CT vooral dat je hoofd- en bijzaken van elkaar kunt scheiden. Wat is er echt belangrijk om een probleem op te lossen?

Doelgroep: Alle leerlingen vanaf groep 3 kunnen deze activiteit doen.

Duur: 30-50 minuten

Leerdoelen: Op een praktische manier het concept abstractie, zoals dat in de computerwetenschappen wordt gebruikt, leren kennen.

Online/offline: offline

Computational thinking:

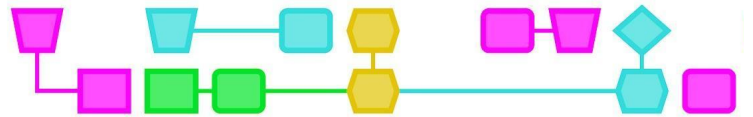
- Algemene vaardigheden: samenwerking, creativiteit
- CT-foundations: abstractie
- CT-concepten: -

Materialen:

- Potlood en papier
- NL CTPrimED fotokaarten

Vorbereiding

Verdeel de klas in groepen van 4 tot 8 leerlingen rond een tafel of bord. Elke groep heeft potloden en een plek nodig om te tekenen (bijv. papier, schoolbord, enz.).



Algemene introductie bij de vier CT-grondbeginselen van de lessen:

Vraag de leerlingen:

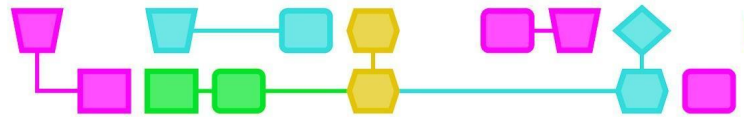
- Wat weet je over hoe een computer en telefoon werkt?
- Kan een computer zelf nadenken? (Waarom wel, waarom niet?)
- Wie bedenkt hoe een computer werkt?

Vertel de leerlingen: We gaan vandaag aan de slag met *computational thinking*. Simpel gezegd is dat leren hoe je een computer voor jou een probleem kunt laten oplossen. Het is niet alleen programmeren, maar bijvoorbeeld ook leren hoe je een probleem in stukjes op kunt delen, of het herkennen van patronen zodat je een probleem beter kunt oplossen. Er zijn vier belangrijke onderdelen van CT:

- **Decompositie** → een probleem opdelen in kleine stukjes.
- **Patroonherkenning** → zoeken naar overeenkomsten of patronen binnen die kleine stukjes die je kunnen helpen het probleem op te lossen.
- **Abstractie** → hoofd- en bijzaken van elkaar onderscheiden; wat is er echt belangrijk om het probleem op te lossen?
- **Algoritmes** → stapsgewijze instructies bedenken om een probleem op te lossen.

In deze les maak je kennis met abstractie.

<p>Decomposition</p> <p>Take ideas and problems apart</p>	<p>Pattern recognition</p> <p>Look for similarities or trends</p>
<p>Abstraction</p> <p>Removing unnecessary information</p>	<p>Algorithms</p> <p>Create step by step instructions</p>



Raad de afbeelding op de kaart

Introductie (5 min)

Eén leerling neemt een kaart van de stapel en probeert, zonder dat de anderen de kaart zien, een abstractie van de afbeelding te tekenen. De rest van het groepje moet raden welke afbeelding er getekend wordt. Elke ronde pakt een andere leerling uit het groepje een kaart en tekent de afbeelding volgens de speciale instructies na.

Lesbeschrijving (10-20 min)

Leg de opdracht uit:

- Trek een kaart (houd deze geheim)
- Teken de afbeelding na
 - Je mag in totaal vijf vormen gebruiken, en alleen de volgende vormen: driehoek, vierkant, rechthoek, cirkel en ellips.
- De rest van het groepje raadt wat de tekening voorstelt.

Laat onderstaand voorbeeld zien bij de uitleg. De activiteit stopt als iedereen in de groep minstens één kaart heeft getekend.

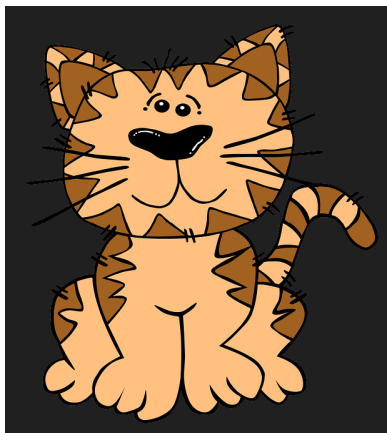


Fig. De kaart

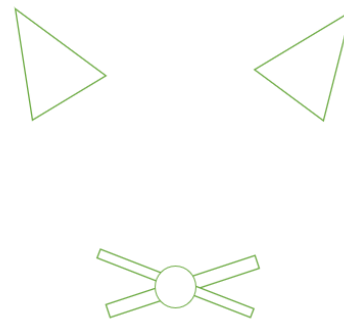
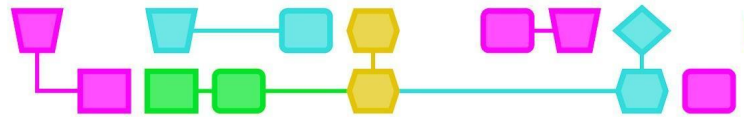


Fig. Getekende afbeelding in max. vijf vormen.

Extra: Om het moeilijker te maken, beperk je het aantal vormen dat de leerling mag tekenen tot drie (herhaling mogelijk).

Conclusie (15-25 min)

Denk met de leerlingen na over het begrip abstractie. Leg uit dat je wordt gedwongen na te denken over wat de belangrijkste kenmerken van de afbeelding zijn doordat je alleen basisvormen mag gebruiken. Daardoor laat je alle details weg. Dit noem je abstractie. Abstractie wordt ook gebruikt bij het oplossen van problemen met behulp van computational thinking. Dan moet bijvoorbeeld worden bekeken welke variabelen of gegevens het belangrijkste zijn, welke bijzaak en welke juist specifiek voor een bepaald geval. Soms moet er ook abstractie worden toegepast op een reeks gegevens omdat er beperkingen zitten aan wat het computersysteem kan opslaan. Bij het ontwerpen van computerprogramma's is abstractie essentieel om algoritmen te ontwerpen die meerdere of algemene problemen kunnen oplossen.



EXTRA ACTIVITEIT: ABSTRAHEREN VAN KENMERKEN UIT EEN GEDETAILEERDE BESCHRIJVING

Als de leerlingen het moeilijk vinden om abstractie te begrijpen, kan deze activiteit helpen bij het leren van de basis van programmeren.

UITDAGING 1

Vraag de leerlingen om kenmerken te bedenken die horen bij een leerling. Geef de leerlingen een paar minuten de tijd om er individueel over na te denken, laat ze daarna discussiëren in kleine groepjes en deel ten slotte de bevindingen met de hele groep.

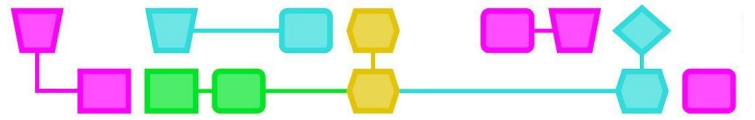
Mogelijke kenmerken die door de leerlingen worden genoemd:

1. Naam
2. Achternaam
3. Geboortedatum
4. Groep
5. Aantal broers/zussen
6. Favoriete kleur

Denk met de leerlingen na over de noodzaak van de kenmerken om een leerling te definiëren. Zijn ze allemaal belangrijk? Dat hangt af van het gebruik van de gegevens. Voor administratief personeel op school zijn punt 5 en 6 bijvoorbeeld niet van belang, maar andere kenmerken zijn dat weer wel (bijv. wie de ouders of wettelijke voogd van een leerling zijn). Hetzelfde gebeurt wanneer men in de informatica gegevensstructuren bedenkt: welke gegevens nodig zijn, hangt af van de context en de omvang van het probleem.

UITDAGING 2

Vraag de leerlingen te beschrijven hoe ze voorwerpen in een kamer ordenen (bijv. hun slaapkamer, het klaslokaal, enz.). Ze moeten het eens zijn over de stappen die ze daarvoor moeten volgen en beseffen dat ze veel details weglaten (het maakt bijvoorbeeld niet uit of het speelgoed een hond of een kat is, het gaat meer om de grootte, het materiaal waarvan het gemaakt is, het gewicht, enz.). Ze abstraheren van de concrete kenmerken om zich te concentreren op de kenmerken die belangrijk zijn voor het sorteren.



Colofon

© CTPrimED

This publication is a product of CTPrimED (2021-1-NL01-KA210-SCH-000031319), funded with support from the Erasmus+ Programme of the European Union. This publication reflects the views only of the authors, and the Commission cannot be held responsible for any use that may be made of the information contained therein.

Project Coordinator

NEMO Science Museum, The Netherlands

Partners

Universidad de la Iglesia de Deusto Entidad Religiosa, Spain
Stichting Children's Science Museum Curacao, Curacao



Co-funded by the
Erasmus+ Programme
of the European Union



Universidad de Deusto
University of Deusto

Deusto

