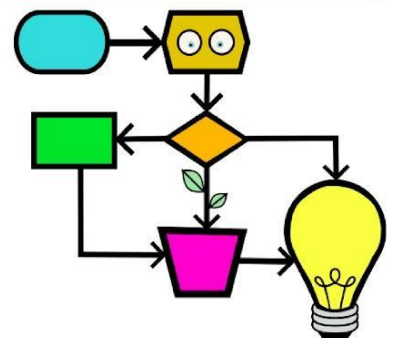
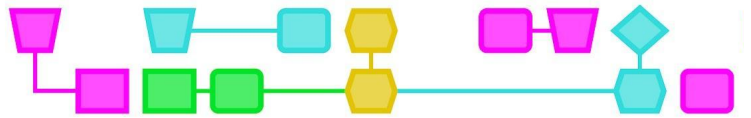


Algemene
vaardigheden





Introductie

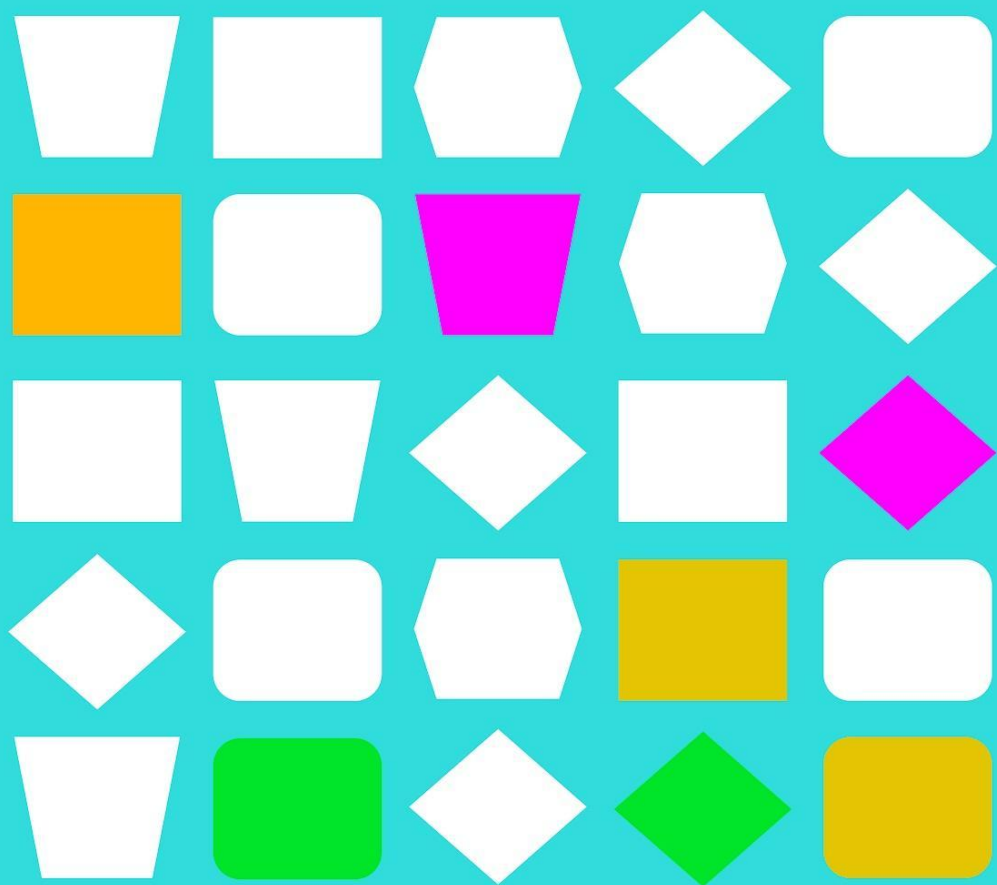
Algemene vaardigheden oefenen, zoals communicatie, probleemoplossing en planning, kunnen leerlingen helpen om te gaan met problemen en vragen waar ze in hun dagelijks leven tegenaan lopen.

Bij het uitvoeren van *computational thinking*-activiteiten gebruik en ontwikkel je algemene vaardigheden. Deze zes offline activiteiten laten manieren zien waarop je dit soort algemene vaardigheden zou kunnen oefenen. Je kunt de opdrachten bijvoorbeeld gebruiken als *energizer* of als afsluiting na een zware dag. De activiteiten kunnen ook gebruikt worden om woorden als creativiteit en brainstorm uit te leggen, zodat deze termen later makkelijk gebruikt kunnen worden in de CT- lessen.

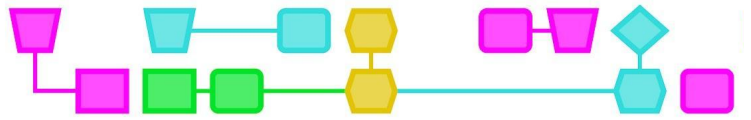
De zes activiteiten gaan over samenwerking (2), brainstormsessies (1), creativiteit (1), en een combinatie van vaardigheden (2).

De opbouw van de activiteiten is voor elke activiteit hetzelfde:

1. Titel van de activiteit
2. Samenvatting
3. Doelgroep: voorkennis/niveau dat de leerlingen nodig hebben voor de activiteit.
4. Duur: hoelang het duurt om deze activiteit uit te voeren.
5. Leerdoelen: wat zijn de belangrijkste doelen die bereikt moeten worden met de uitvoering van deze activiteit
6. Bijzonderheden
7. Materialen: lijst van materialen die nodig zijn voor deze activiteit
8. Voorbereiding: dingen die een leerkracht vooraf moet voorbereiden om de activiteit goed uit te kunnen voeren
9. Beschrijving: hierin worden de activiteit en de didactiek van de les beschreven.



Samenwerken



Stapel de bekers

Samenvatting: De leerlingen werken samen om een beker te verplaatsen zonder deze aan te raken. Elk team moet een manier vinden om van touw en elastiek iets te maken waarmee ze de bekers kunnen oppakken en stapelen. Bij het stapelen worden er drie bekers op de onderste rij gezet, met daarop een rij van twee bekers. De creatie om de bekers op te pakken wordt door twee leerlingen bestuurd. Aan elke kant staat een leerling. Zij werken samen om de bekers op te pakken en te stapelen.

Doelgroep: Vanaf 7 jaar (de leerlingen moeten beschikken over een goed ontwikkelde fijne motoriek en een goed evenwichtsgevoel).

Duur: 15-20 minuten

Leerdoelen: Leerlingen leren als team samen te werken en samen het doel te bereiken.

Online/offline: offline

Computational thinking:

- Algemene vaardigheden: plannen, samenwerken, communiceren
- CT-foundations: algoritme en patroonherkenning
- CT-concepten: loops

Bijzonderheden: Deze activiteit kan het best binnen worden uitgevoerd, of op een plaats met minder of geen wind.

Materialen: Voor elke groep: vijf niet-breekbare bekers, een elastiekje en een touwtje.

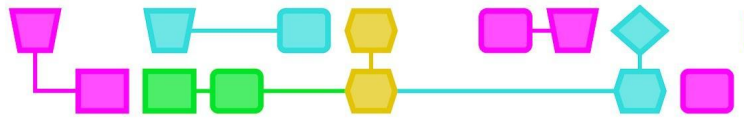
Vorbereiding:

Verdeel de klas in tweetallen. Elk duo heeft een eigen werkplek (bureau) waar ze de bekers kunnen pakken en een bureau om hun stapel van vijf bekers op te bouwen.

Beschrijving:

Deel de materialen uit, vertel de leerlingen dat ze van het touwtje en elastiekje iets moeten maken waarmee ze de beker kunnen oppakken zonder dat ze de beker aanraken. Als elk groepje dat heeft gedaan en even heeft geoefend met het oppakken van de bekers kan het stapelen beginnen. Op het teken van de docent moet elk tweetal één beker oppakken met de creatie en op hun stapeltafel neerzetten. Het tweetal dat als eerste klaar is, wint. Als een beker tijdens het verplaatsen valt, moet het tweetal opnieuw beginnen.

Begin met een stapel van drie rijen voor de jongere leerlingen en voeg meer rijen toe naarmate de leeftijd toeneemt.



Schouderwandeling-challenge

Samenvatting: Bij deze activiteit moeten de leerlingen naast elkaar staan en tegen elkaars schouders leunen. Hierdoor dragen ze elkaars gewicht tijdens het lopen, en zo moeten ze de finish bereiken.

Doelgroep: Vanaf 6 jaar (de leerlingen moeten beschikken over een goed ontwikkelde balans).

Duur: 15-20 minuten

Leerdoelen: Leerlingen leren samen te werken en elkaar te vertrouwen.

Online/offline : offline

Bijzonderheden: Deze activiteit moet buiten of in een gymzaal worden uitgevoerd omdat er ruimte nodig is om met z'n tweeën de benodigde afstand te kunnen afleggen.

Materialen: Krijt om een start- en finishlijn te markeren.

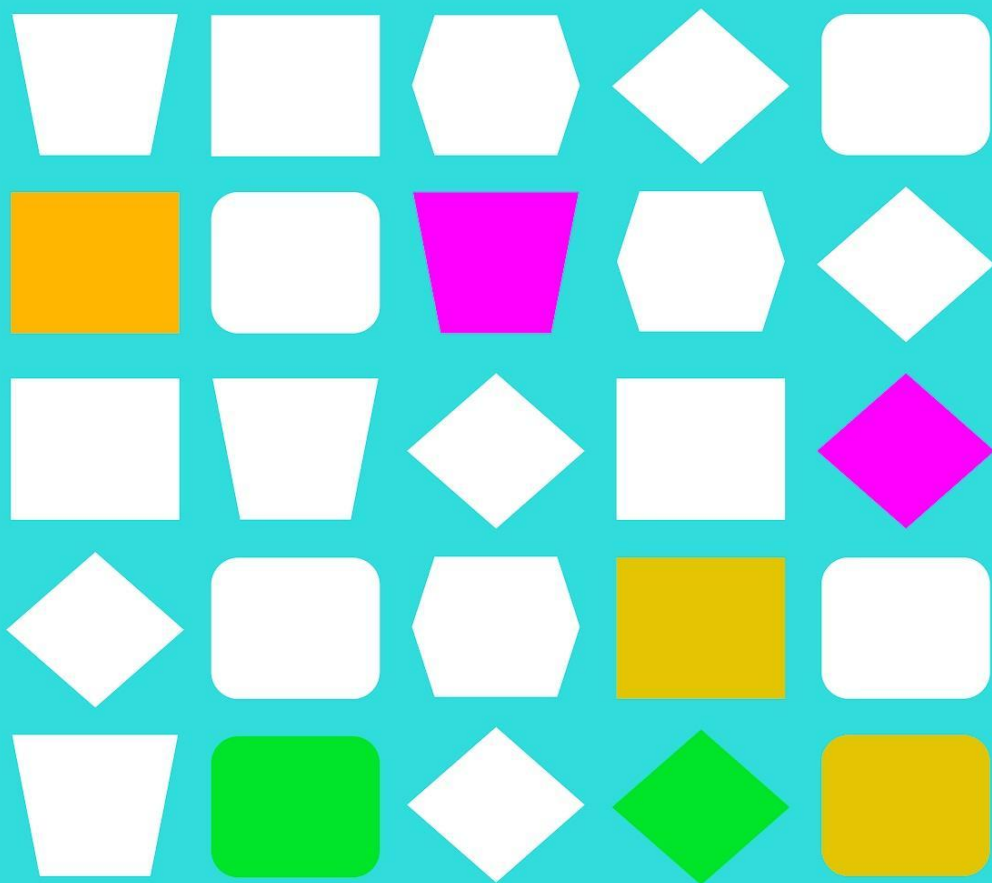
Vorbereiding: Verdeel de klas in twee groepen. Elke groep vormt paren om de schouderwandeling te doen.

Beschrijving: Vraag de leerlingen om tweetallen te vormen, met iemand van ongeveer dezelfde lengte. Elk tweetal staat naast elkaar en leunt zijwaarts naar elkaar toe, zodat ze op elkaars schouders steunen. Vraag de tweetallen om hun eigen voeten zo ver van hun partner af te zetten als voor hen fijn voelt.

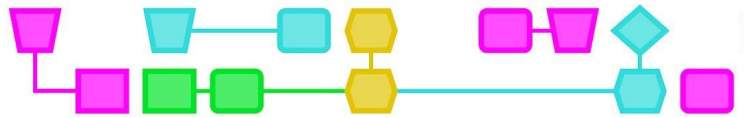
Als ze klaar zijn, leg dan uit dat elk paar deze leunende houding moet aanhouden terwijl ze samen naar de finish lopen.

Elke groep heeft evenveel paren die moeten lopen en over de finish moeten gaan om te winnen. Als iedereen de eindstreep heeft bereikt, wisselen de partners.

Voor dit spel is vertrouwen nodig, wees voorbereid om de leerlingen gerust te stellen en te begeleiden.



Brainstormen



Carrousel

Samenvatting: Er staat een (uitdagende) stelling op het bord van de klas. De leerlingen schrijven in korte tijd zoveel mogelijk oplossingen op voor de stelling. Het idee van de opdracht is om de leerlingen uit te dagen zoveel mogelijk out of the box ideeën, creatieve en algemene ideeën te laten bedenken.

Doelgroep: 8 jaar en ouder (lees- en schrijfvaardigheid en algemene kennis zijn vereist).

Duur: 15-20 minuten

Leerdoelen: Leerlingen leren nadenken over een onderwerp, ideeën bedenken en verwoorden, snel denken en niet bang zijn om rare oplossingen op te schrijven.

Bijzonderheden: De activiteit kan het best binnen worden uitgevoerd.

Materialen: Vijf flip-over vellen , vijf verschillende kleuren markers, plakband.

Vorbereiding: Verdeel de klas in groepjes van vier of vijf leerlingen. Bedenk en schrijf een paar uitspraken.

Voorbeelden vragen en stellingen (afhankelijk van de leeftijd):

Vrolijke vragen

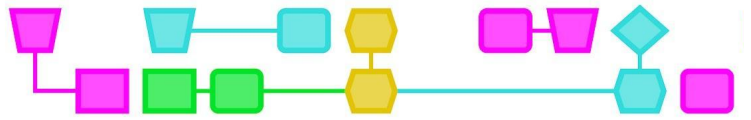
1. Wat zouden we nodig hebben om een leuker/actiever schoolplein te maken?
2. Hoe kunnen we eenzaamheid onder ouderen tegengaan?
3. Hoe zouden we een schooldag leuker kunnen maken?

Serieuze stellingen

1. De opwarming van de aarde is zorgwekkend.
2. Te veel plastic in de zee doodt vissen.
3. Door junkfood krijgen meer kinderen overgewicht.

Lesbeschrijving

Schrijf vijf 'stellingen' op vijf verschillende flip-overvellen en plak deze rondom de klas op de muren. Verdeel de leerlingen in groepjes van vier of vijf leerlingen. Elke groep gaat bij een flip-over staan om te beginnen. Elke groep krijgt een andere kleur stiften om hun oplossingen mee op te schrijven. Zet de timer op 40 seconden. Elke groep moet de probleemstelling lezen en zo veel mogelijk oplossingen opschrijven die ze kunnen bedenken. Als de timer afgaat, wisselen ze klokgewijs naar de volgende flip-over. Wanneer alle groepen elke flip-over hebben gehad, worden de oplossingen van elke stelling klassikaal besproken. Leg de nadruk hierbij niet op goed of fout maar op de veelzijdigheid aan ideeën die ze hebben bedacht.



Doodle art

Samenvatting: Met een zwarte of donkerblauwe stift moet iedere leerling op wit papier lijnen tekenen. Als het papier gevuld is met lijnen moet de leerling de witte vlakken nauwkeurig inkleuren. Door het inkleuren probeert de leerling te ontdekken of er een bekend of creatief voorwerp ontstaat.

Doelgroep: 6 jaar en ouder.

Duur: 15-20 minuten

Leerdoelen: Leerlingen leren hun creativiteit te uiten en zich te concentreren.

Bijzonderheden: Deze opdracht moet binnen aan een bureau worden uitgevoerd.

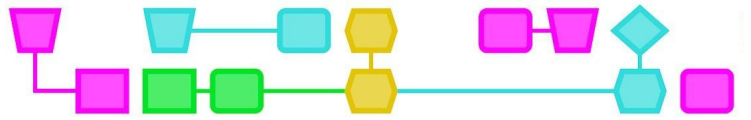
Materialen: Wit A4-papier voor elke leerling, zwarte of donkerblauwe marker, kleurpotloden met fijne lijnen of uitwisbare markers met fijne punten (voor kinderen).

Vorbereiding: Elke leerling krijgt een wit A4-papier, een zwarte of donkerblauwe stift en een pakje kleurpotloden met fijne lijntjes of uitwisbare stiften.

Beschrijving

Elke leerling begint met het tekenen van willekeurige lijnen op het witte papier met de zwarte of donkerblauwe stift. Ze kleuren nog niet. Nadat het papier gevuld is met lijnen, kleurt de leerling de witte vlakken zorgvuldig in om te proberen een bekend of creatief voorwerp te ontdekken/maken dat zou kunnen ontstaan door bepaalde vlakken in te kleuren.

Let op dat de leerlingen bij het tekenen van de lijnen kleine vlakjes over laten om in te kleuren!



Wat is het verhaal???

Samenvatting: De leerlingen krijgen drie minuten om een begin te bedenken van een verhaal waarvan ze alleen de titel weten. Na deze drie minuten schrijft een andere groep verder aan het verhaal. Samen maken ze een volledig verhaal.

Doelgroep: 8 jaar en ouder.

Duur: 15-20 minuten

Leerdoelen: Leerlingen leren creatief denken en zinnen formuleren.

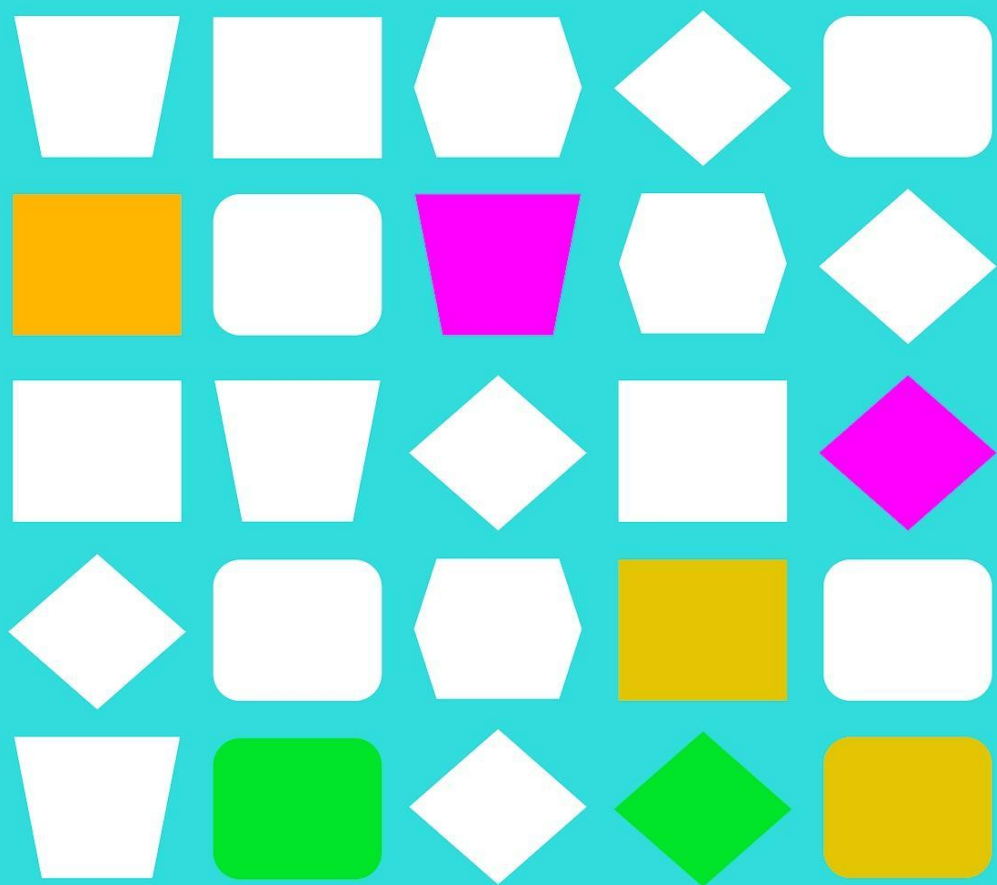
Bijzonderheden: Deze opdracht kan zowel binnen als buiten worden uitgevoerd.

Materialen: Grote witte papieren, potloden, timer

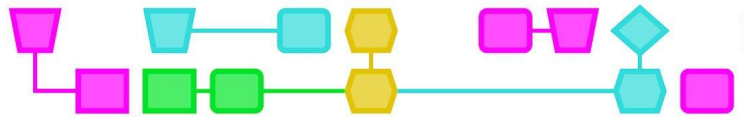
Vorbereiding: Hang vijf grote witte papieren aan de muren en verdeel de klas in groepjes van zes leerlingen. Elke groep gaat rondom zijn eigen witte papier staan. Elke leerling krijgt een potlood.

Lesbeschrijving

Schrijf titels op wit papier als startpunt van het verhaal. Bijvoorbeeld: 'De gekke vogel' of: 'Een mooie tuin'. Zet een timer op drie minuten. De leerlingen schrijven het begin van het verhaal. Elke leerling mag slechts twee woorden aan het verhaal toevoegen. Elke volgende leerling uit het groepje zal twee woorden aan het verhaal toevoegen tot de timer afloopt. Het is belangrijk dat de woorden een zin vormen die logisch is. De leerlingen kunnen zelf bepalen hoe lang een zin is en wanneer een zin eindigt. Als de timer afgaat, mag van elk groepje de leerling die het verhaal begon hardop voorlezen wat ze tot dan toe hebben geschreven. Daarna wisselen de groepjes van vel papier en gaat elk groepje verder met het verhaal van het vorige groepje.



Verschillende vaardigheden



Bouw of maak een helpende hand

Samenvatting: In groepjes ontwerpen en bouwen de leerlingen samen een helpende hand om iets te pakken of op te rapen.

Doelgroep: 7 jaar en ouder.

Duur: 30 minuten

Leerdoelen: Verkennen, ontwerpen, creëren.

Online/offline: offline

Algemene vaardigheden: Samenwerking, communicatie, planning.

Materialen: karton, stro, fijn touw, lijm, touwtjes, ringen, flessen, enz.

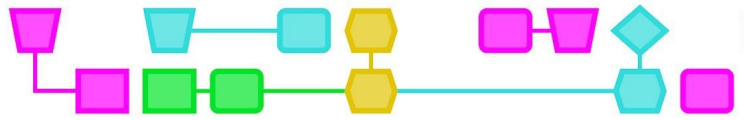
Vorbereiding: De klas moet worden verdeeld in groepen van vijf leerlingen.

Beschrijving

In groepjes ontwerpen en bouwen de leerlingen samen een helpende hand om een taak uit te voeren: een hulpmiddel om iets te pakken of op te pakken.

Laat de leerlingen een voorbeeld van een helpende hand zien en toon de materialen die ze kunnen gebruiken. Geef de leerlingen 20 minuten om aan deze activiteit te werken.

Zorg voor een voorbeeld van een helpende hand om de leerlingen te inspireren.



Een auto die wordt aangedreven door lucht

Samenvatting: Elke groep leerlingen moet met de beschikbare materialen een auto bouwen die beweegt door lucht. De auto moet rijden op de kracht van de wind in de ballon en mag niet met de hand geduwd of gesleept worden.

Doelgroep: 8 jaar en ouder.

Duur: 45 minuten

Leerdoelen: Leerlingen leren onderzoeken, ontwerpen, creëren en experimenteren met de kracht van lucht.

Algemene vaardigheden: Samenwerken, communiceren, plannen, creëren/bouwen

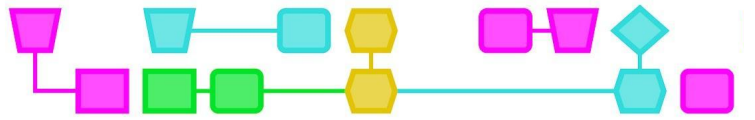
Materialen: Gerecyclede producten, zoals ijsstokjes, oude dvd's, karton, tandenstokers, touwtjes, lijm, touwen, ballonnen, plastic zakken, motoren, touwtjes van draad.

Vorbereiding: Verdeel de klas in groepen van vijf leerlingen.

Beschrijving: Elke groep leerlingen bouwt met de beschikbare materialen een auto die kan bewegen door luchtkracht. De auto moet rijden op de kracht van de wind in de ballon en niet met de hand geduwd of gesleept worden.

Zet een afgezet parcours klaar, met een startlijn en een finishlijn. Het belangrijkste doel is dat de auto over de lijn wordt geduwd door lucht. De leraar bepaalt de afstand tussen de start- en de finishlijn.

Laat de leerlingen een voorbeeld zien van een auto die beweegt door lucht, en toon de leerlingen de materialen die ze kunnen gebruiken. De leerlingen mogen alle beschikbare materiaal gebruiken. Geef de leerlingen 40 minuten om hun auto te ontwikkelen.



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

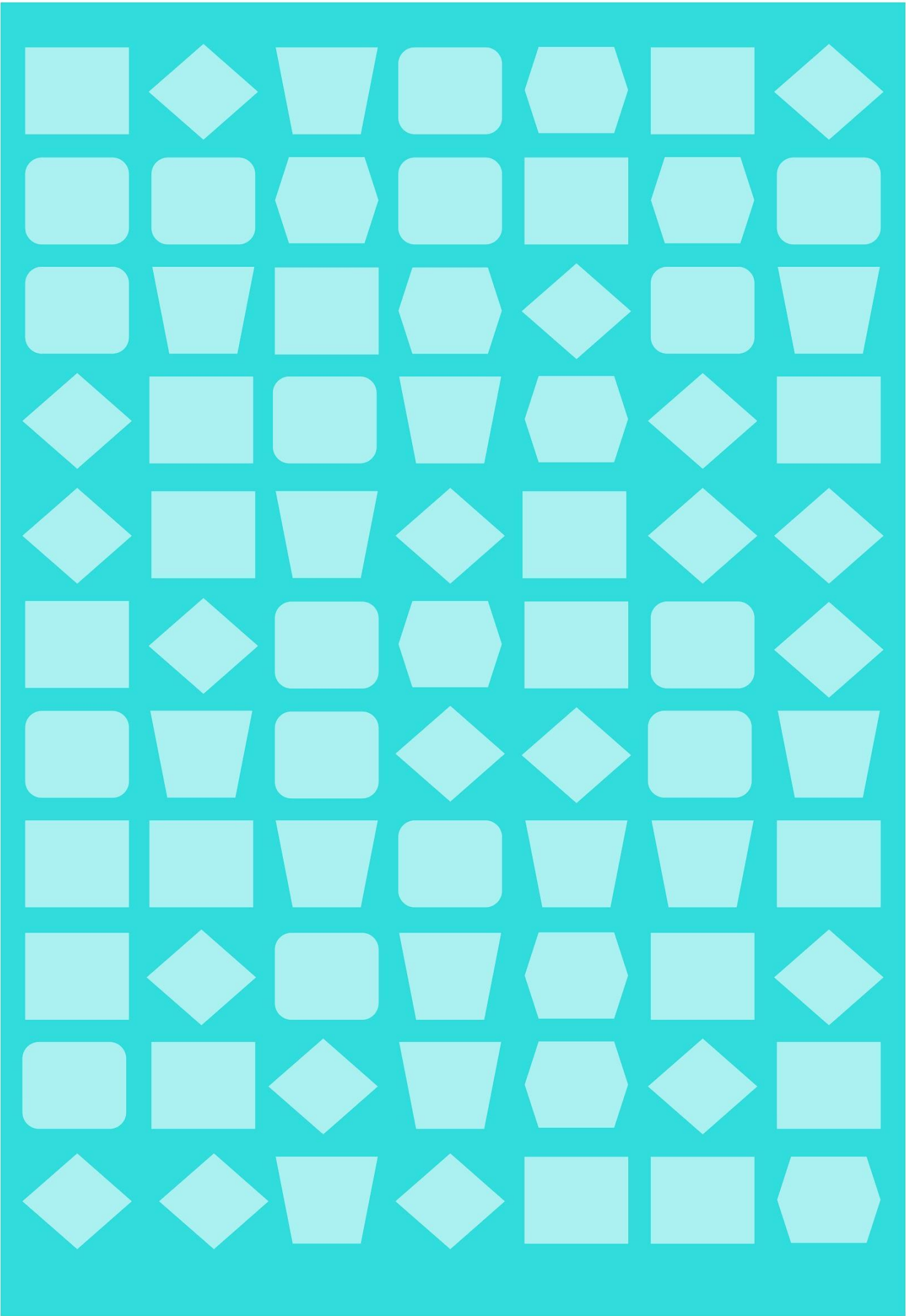


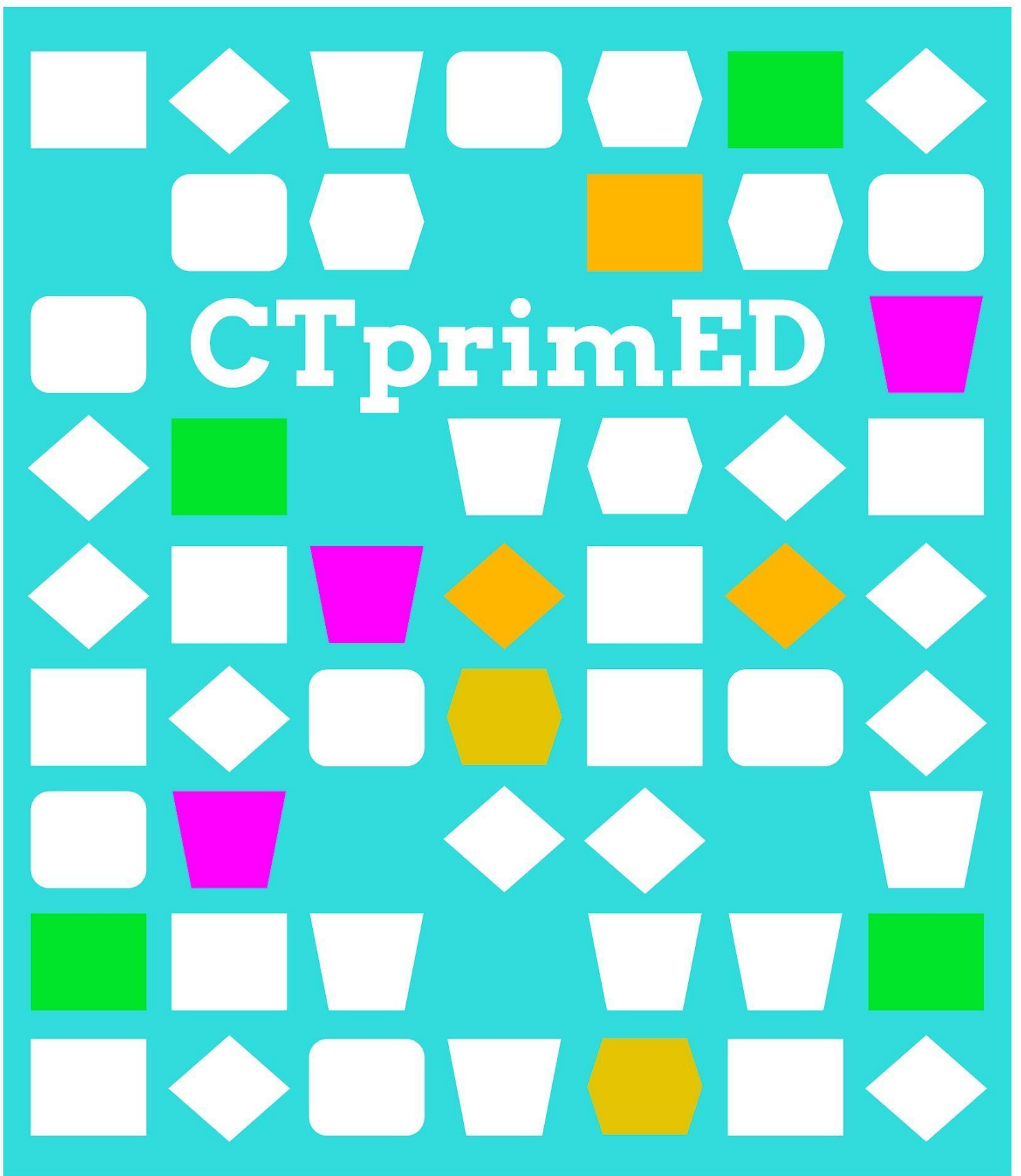
Co-funded by the
Erasmus+ Programme
of the European Union



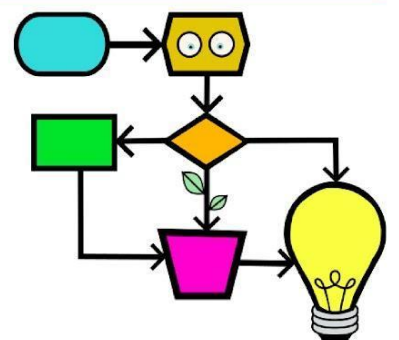
Universidad de Deusto
University of Deusto

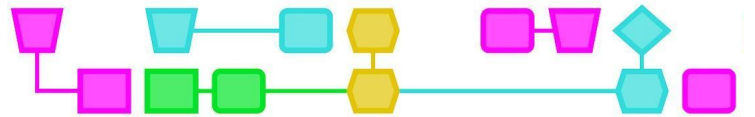
Deusto





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:

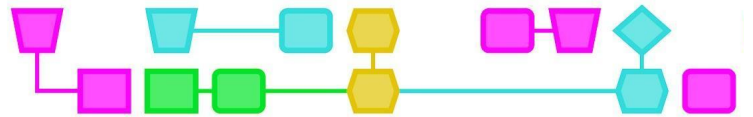
- Algemene vaardigheden: samenwerking, creativiteit
- CT-foundations: abstractie
- CT-concepten: -

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:

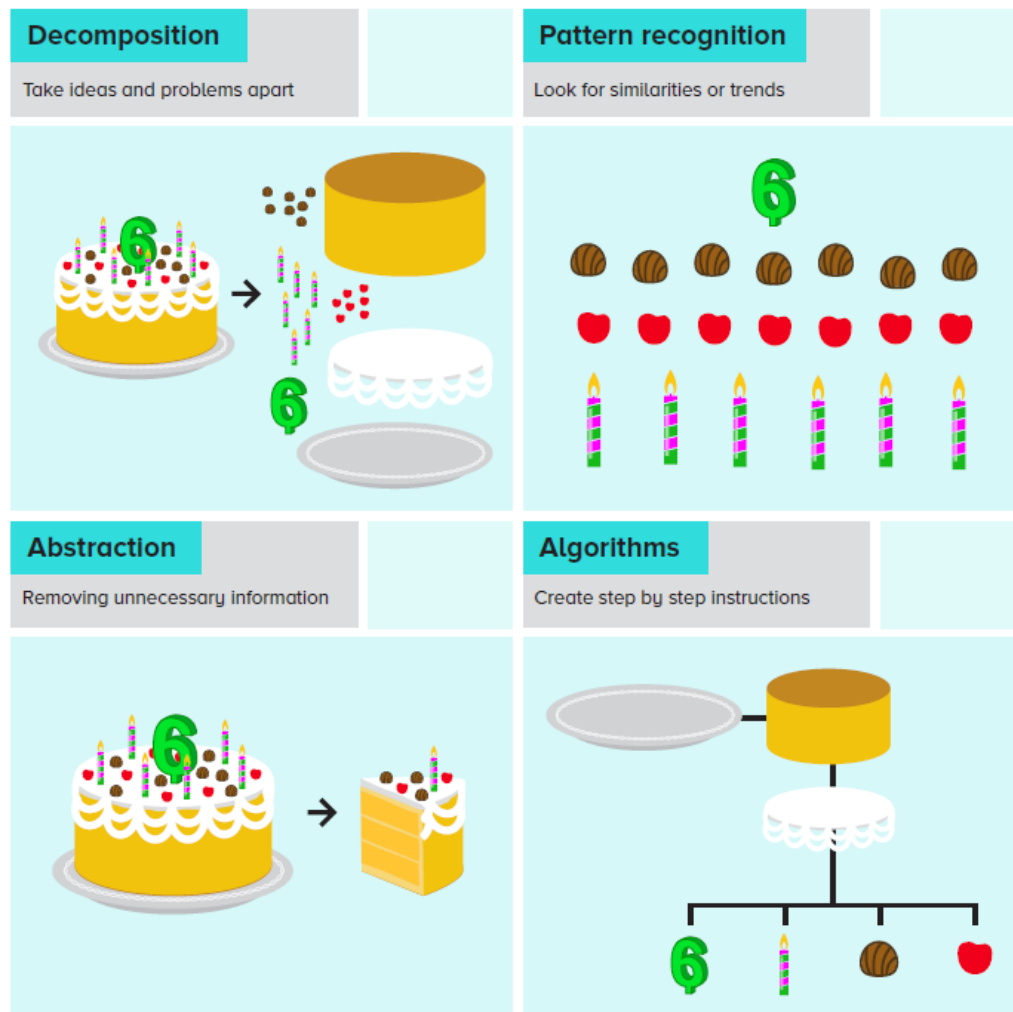
Vraag de leerlingen:

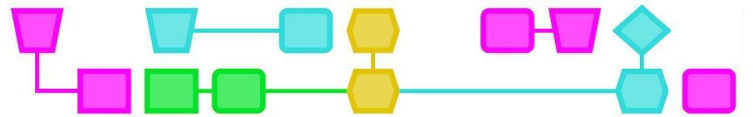
- Wat weet je over hoe een computer en telefoon werkt?
- Kan een computer zelf nadenken? (Waarom wel, waarom niet?)
- Wie bedenkt hoe een computer werkt?

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.





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.

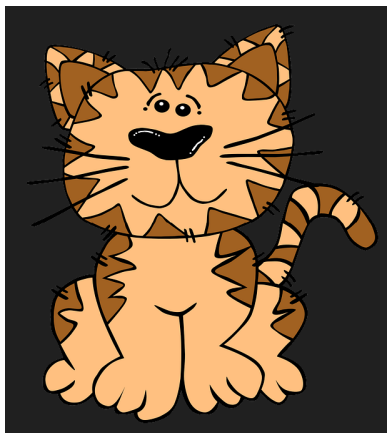


Fig. De kaart

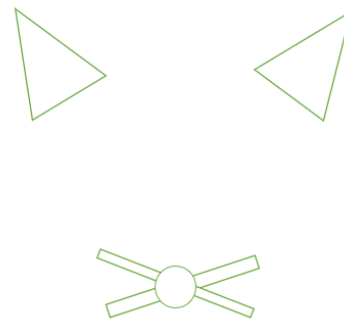
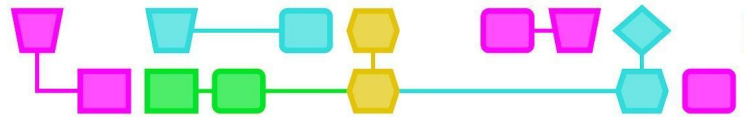


Fig. Getekende afbeelding in max. vijf vormen.

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 GEDETAILLEERDE 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.

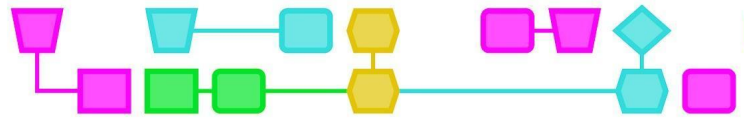
Mogelijke kenmerken die door de leerlingen worden genoemd:

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

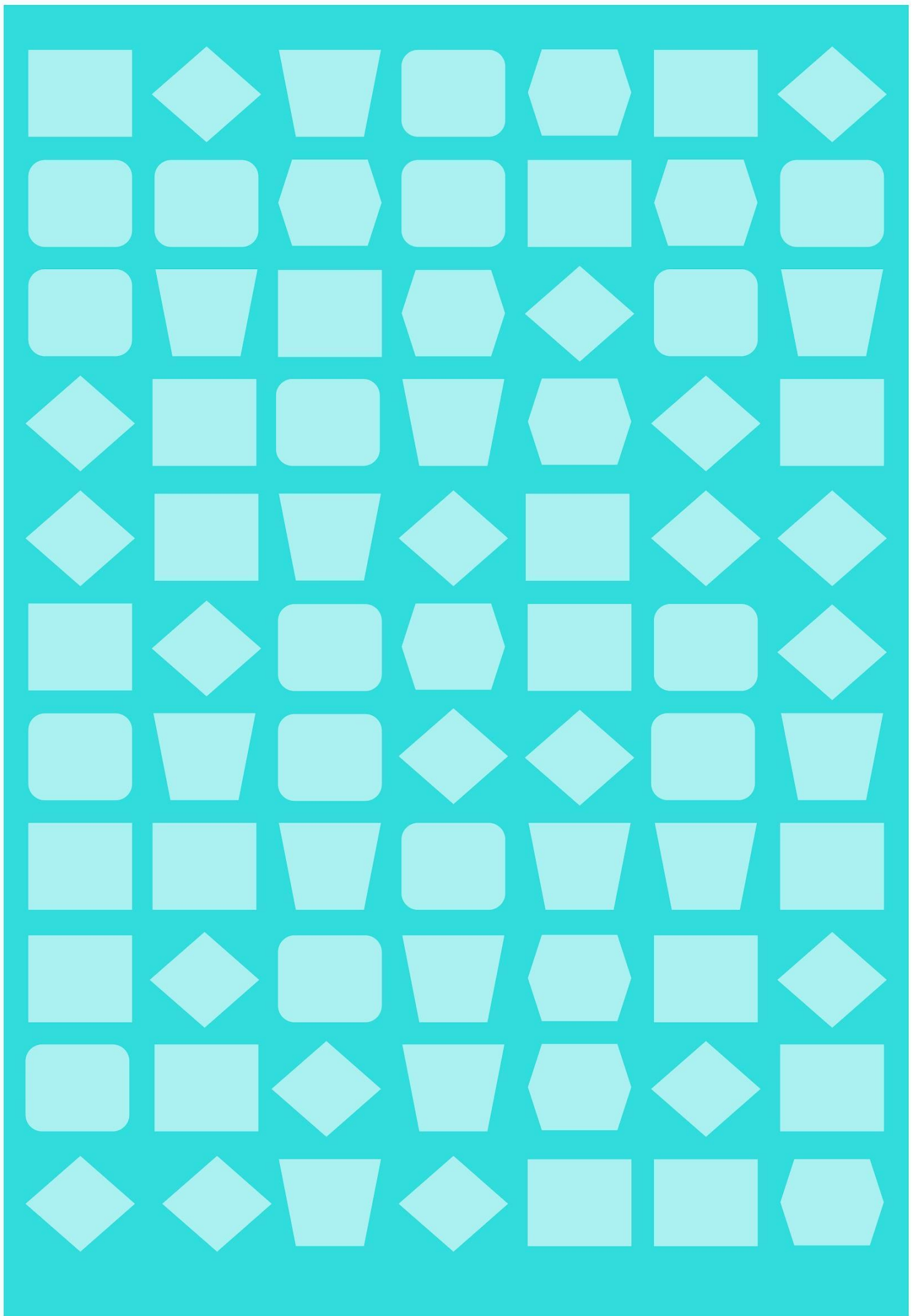


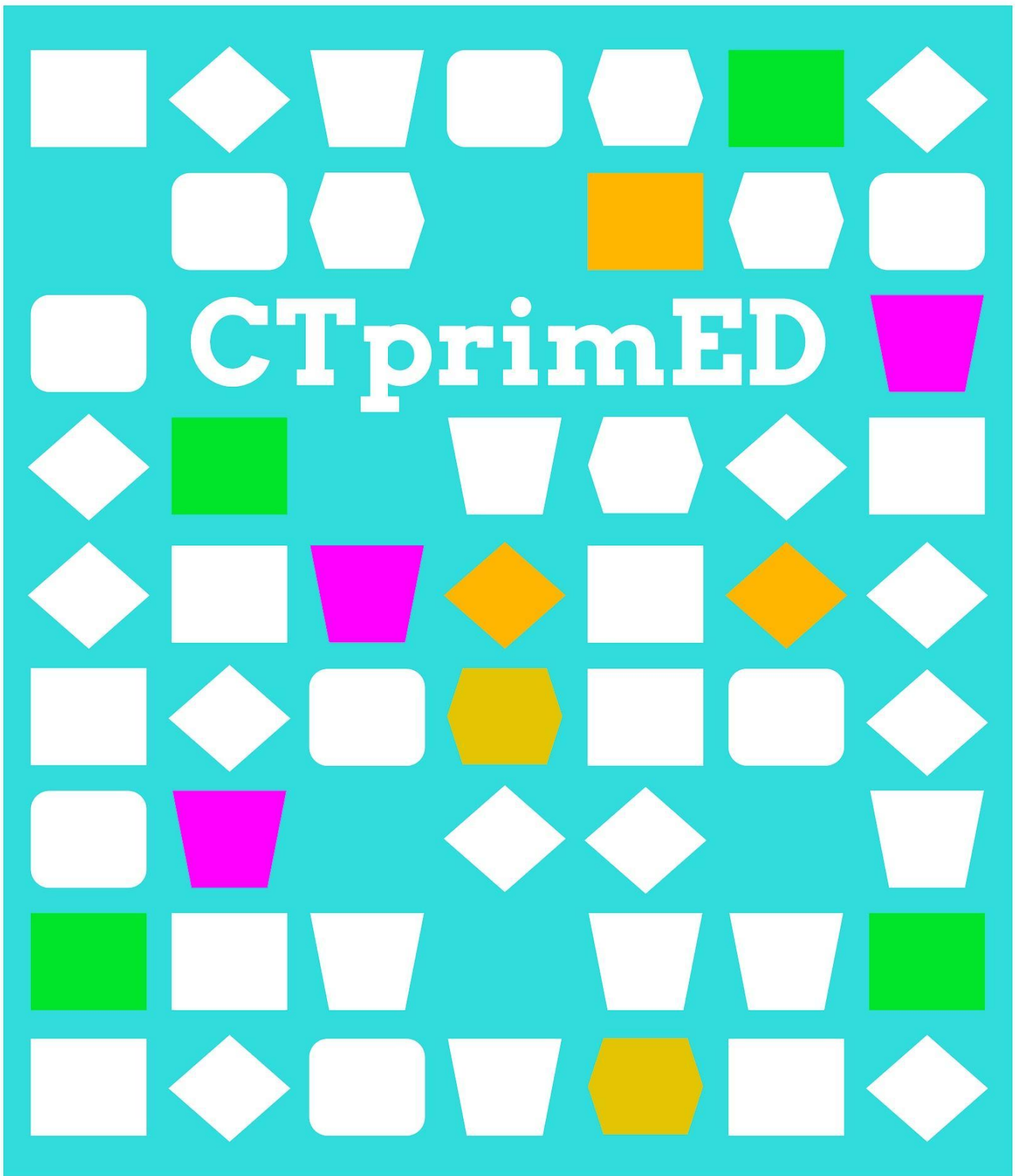
Co-funded by the
Erasmus+ Programme
of the European Union



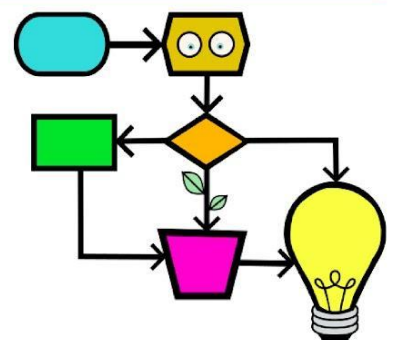
Universidad de Deusto
University of Deusto

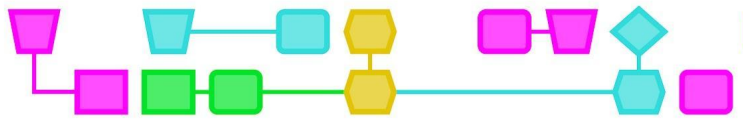
Deusto





Introductie Algoritmes





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

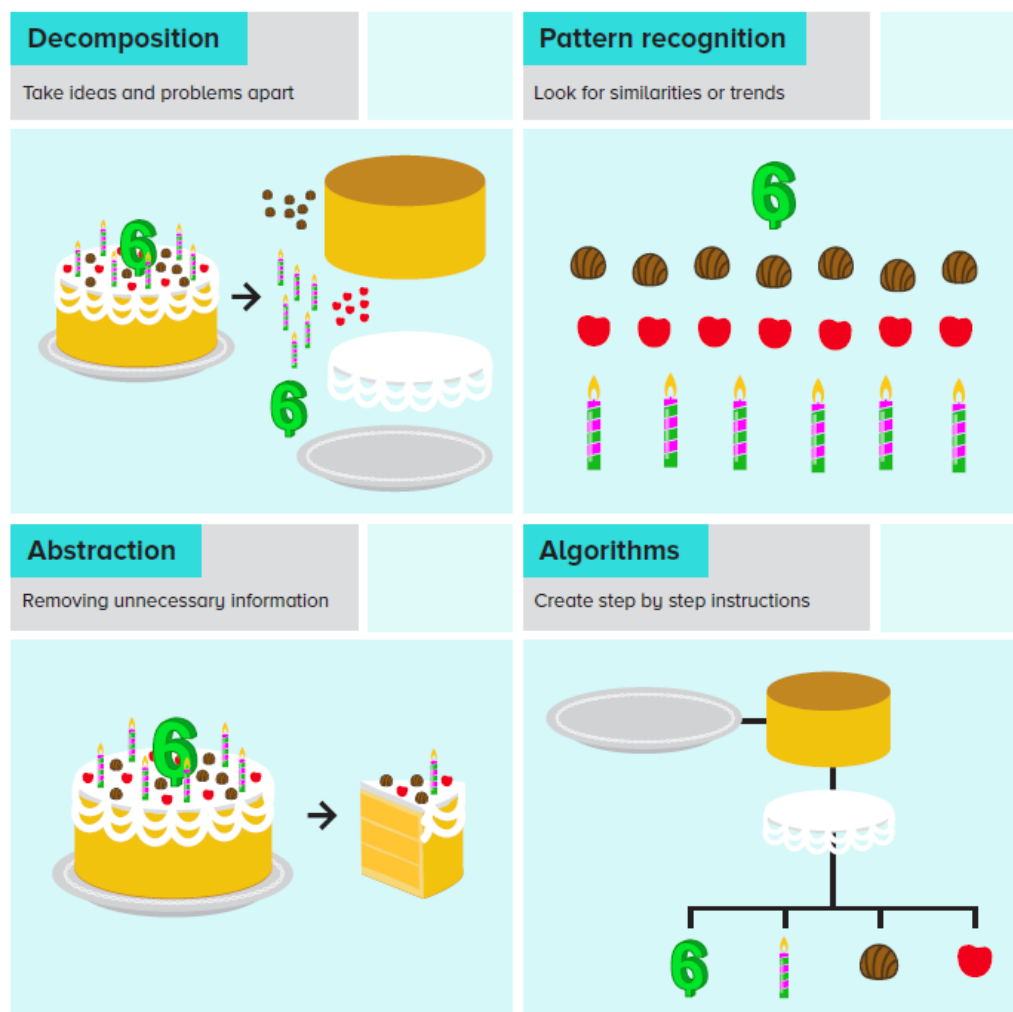
Vraag de leerlingen:

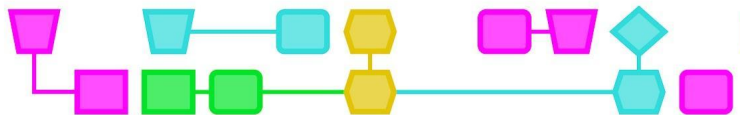
- Wat weet je over hoe een computer en telefoon werkt?
- Kan een computer zelf nadenken? (Waarom wel, waarom niet?)
- Wie bedenkt hoe een computer werkt?

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 algoritmes.





Samenvatting

Doelgroep: Leerlingen vanaf 7 jaar. Er is geen voorkennis nodig.

Duur: 50 minuten

Leerdoelen:

De leerlingen leren

- specifieke opdrachten op te volgen (zoals een computer zou doen);
- hun eigen kaart te maken (programmeren).

Online or offline: offline

Computational thinking:

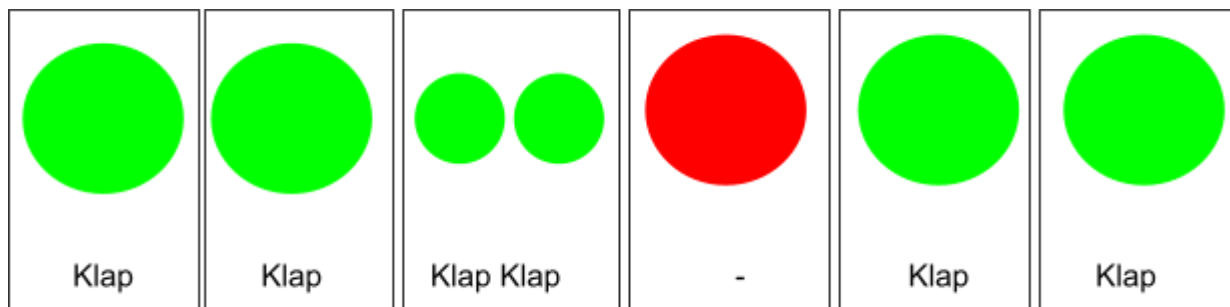
- Algemene vaardigheden: samenwerken, creativiteit
- CT-foundations: algoritme, patroonherkenning, decompositie, abstractie

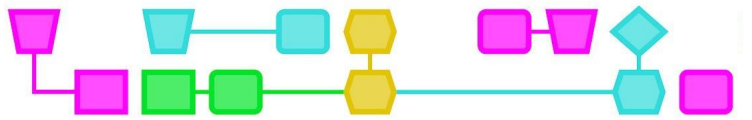
Materialen:

- Stroken wit karton
- Rode en groene stiften
- Potlood om opdrachten mee uit te werken

Vorbereiding

Verdeel de klas in groepjes van 4. Print voor elk groepje onderstaande basiskaart.

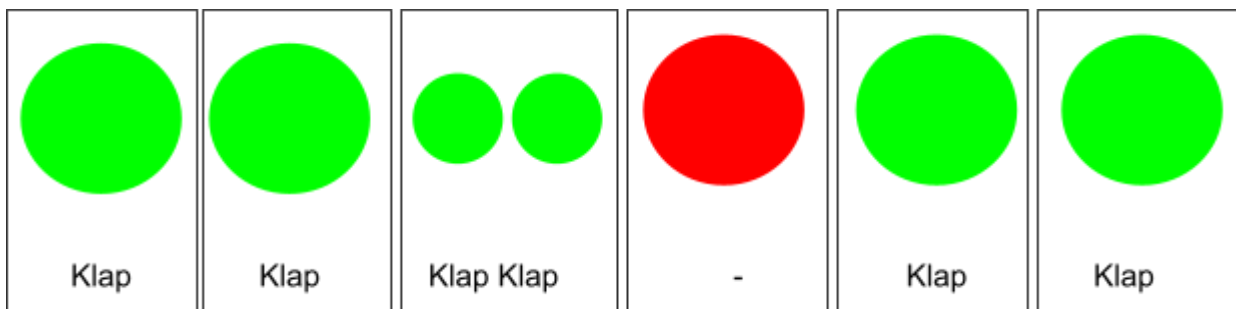




Lesbeschrijving

Introductie (10 min)

Laat onderstaande afbeelding aan de leerlingen zien. Leg uit wat de symbolen betekenen en klap samen met de leerlingen het ritme. Vertel dat de leerlingen nu een programma, oftewel een algoritme, hebben uitgevoerd. Een algoritme is een reeks regels die je kunt gebruiken om een probleem op te lossen en wordt vaak gebruikt door een computer. Maar ze kunnen ook voor andere dingen gebruikt worden, zoals het strikken van je veters, een recept volgen of een choreografie. Nadat onderstaand algoritme gevolgd is gaan de leerlingen hun eigen algoritme maken.



Schrijf je eigen algoritme (15 min)

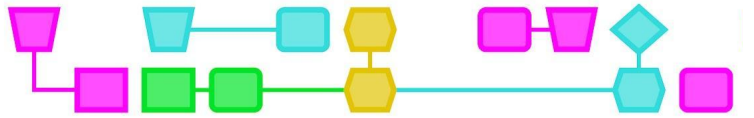
Verdeel de materialen die nodig zijn voor elk groepje van vier: stroken wit karton, gekleurde stiften, potlood of pen om de verschillende opdrachten mee uit te werken. Elk groepje maakt een eigen algoritme. Samen kiezen ze welk geluid ze koppelen aan bovenstaande pictogrammen en in welke volgorde ze de pictogrammen zetten. Ze mogen maximaal tien pictogrammen gebruiken.

Nadat ze het algoritme hebben getekend gaan ze elkaars algoritmes uitvoeren.

Tip: Om het wat uitdagender te maken voor de leerlingen kun je ze ook een *loop* laten maken (hierbij moeten een aantal pictogrammen herhaald worden) of een 'als... dan...' regel laten toevoegen.

Conclusie (10 min)

Sluit de activiteit af met de vraag hoe het voelt om algoritmes te volgen. Bespreek ook waar de leerlingen tegenaan liepen tijdens het testen van hun algoritme. Vertel dat het belangrijk is om duidelijke opdrachten te geven in een algoritme, en dat dit bij computers ook belangrijk is.



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

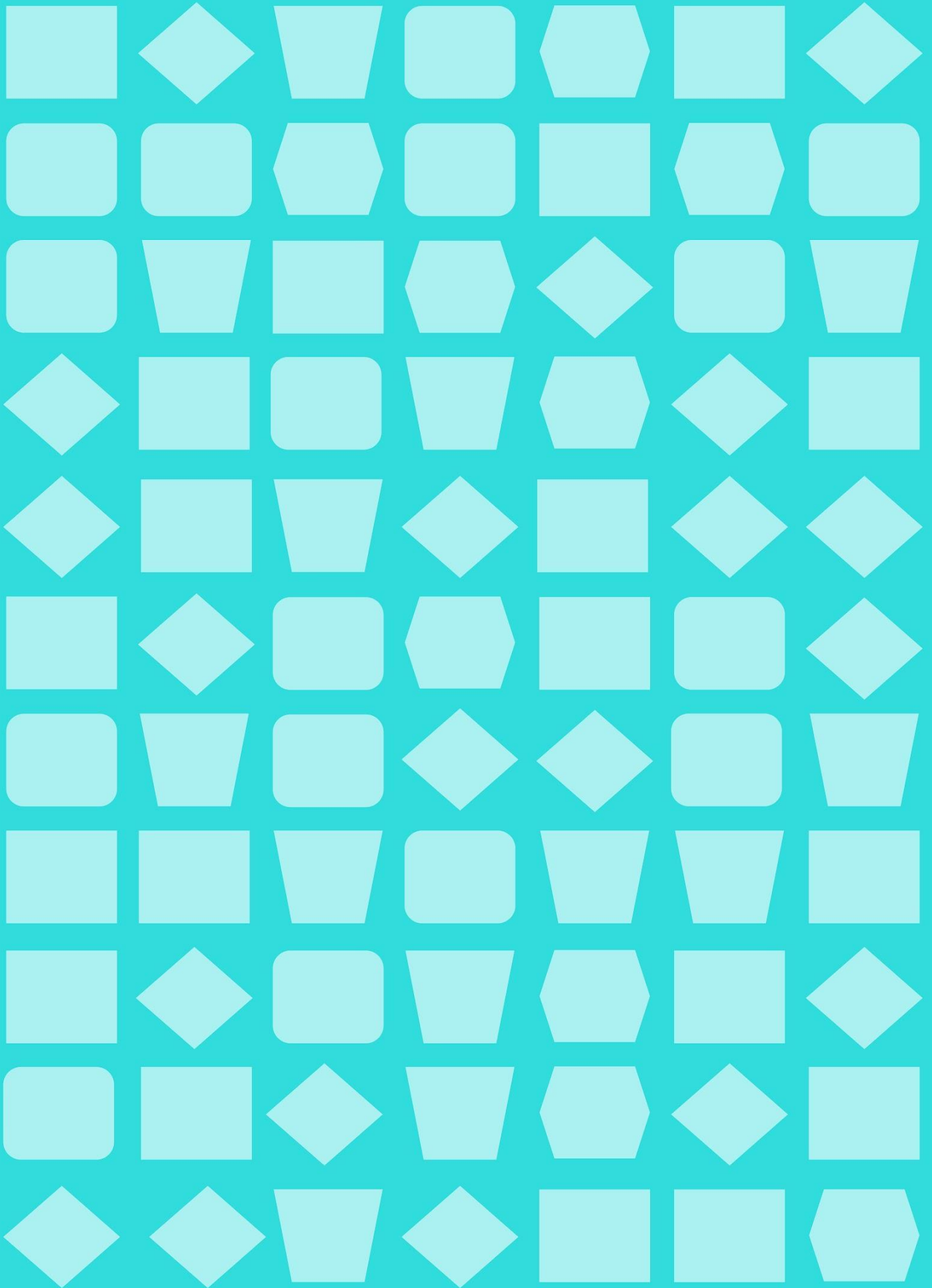


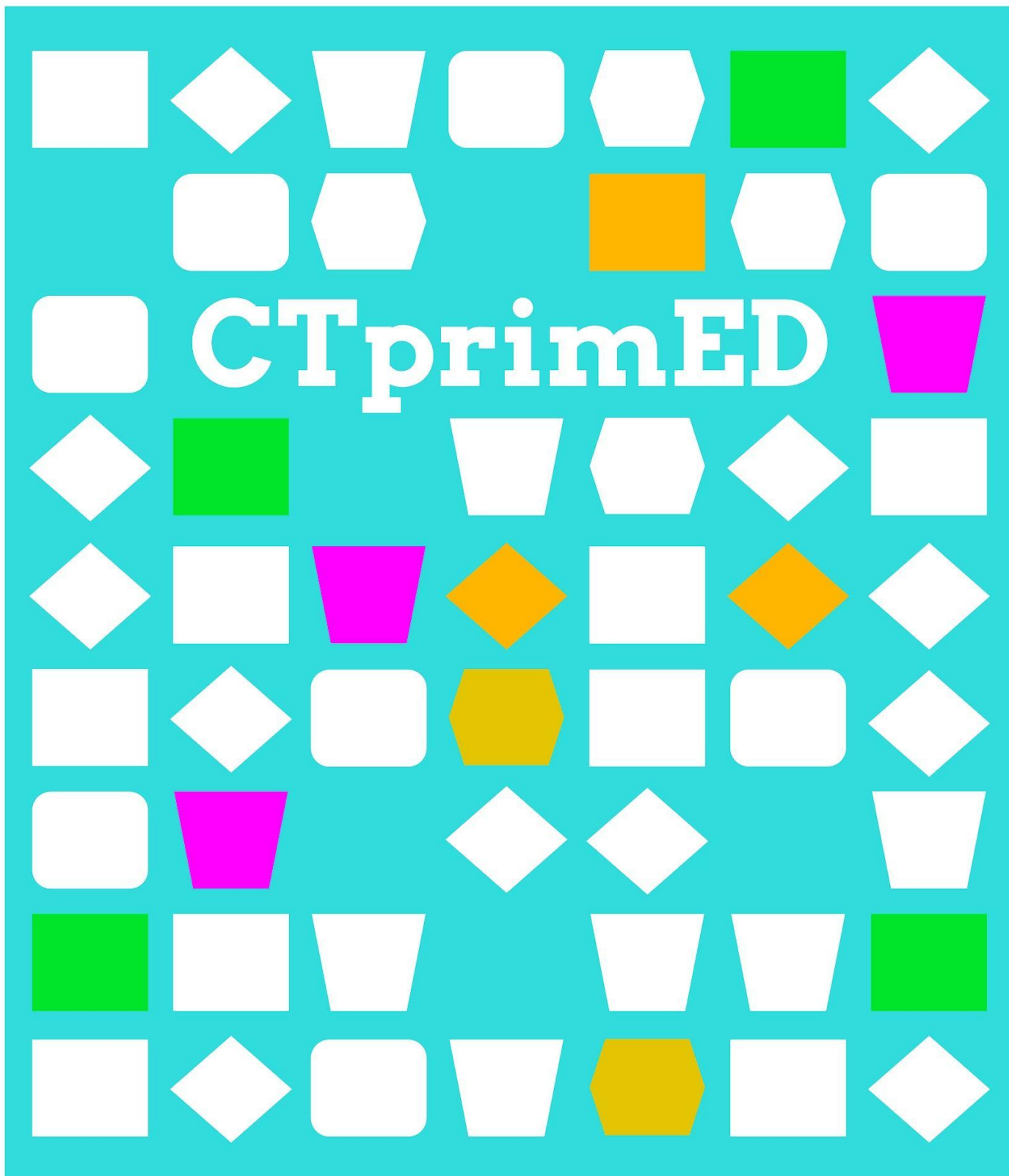
Co-funded by the
Erasmus+ Programme
of the European Union



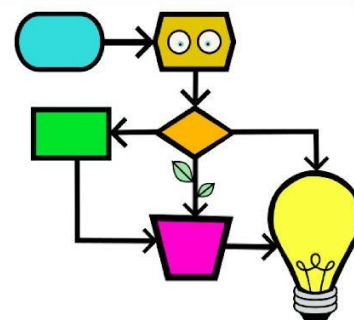
Universidad de Deusto
University of Deusto

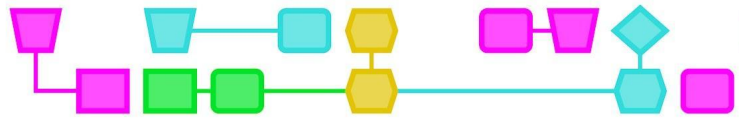
Deusto





Introductie
 Decompositie





Samenvatting

In deze activiteit werken de leerlingen aan het concept decompositie, een van de vier grondbeginselen van *computational thinking*. Decompositie is het proces waarbij een probleem wordt opgesplitst in een aantal kleinere problemen die beter te begrijpen en op te lossen zijn.

Doelgroep: De deelnemers moeten de speelkaarten kunnen begrijpen (de waarden en hoe ze te ordenen) en weten hoe ze moeten optellen.

Duur: 30-50 minuten

Leerdoelen: Het doel is om op een praktische manier het begrip **decompositie** te leren (zoals dat in de computerwetenschap wordt gebruikt). Het gaat uit van een activiteit die ver van de computer afstaat en brengt deze vervolgens in verband met probleemoplossing, programmeren of datastructuren.

Online/offline: offline

Computational thinking:

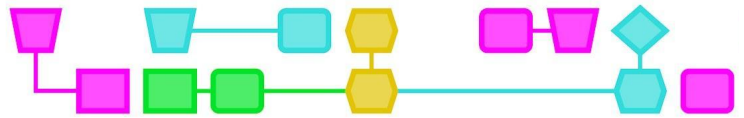
- Algemene vaardigheden: samenwerken, logisch denken, problemen oplossen, besluiten nemen
- CT-foundations: decompositie.

Materialen:

- Twee kaartspellen per groep (pokerkaarten, 80-100 kaarten per groep)
- Een stopwatch (bijvoorbeeld een timer op een mobiele telefoon)

Vorbereiding

Verdeel de klas in groepjes van 4 tot 8 leerlingen rond een tafel. Elke groep heeft twee kaartspellen nodig.



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

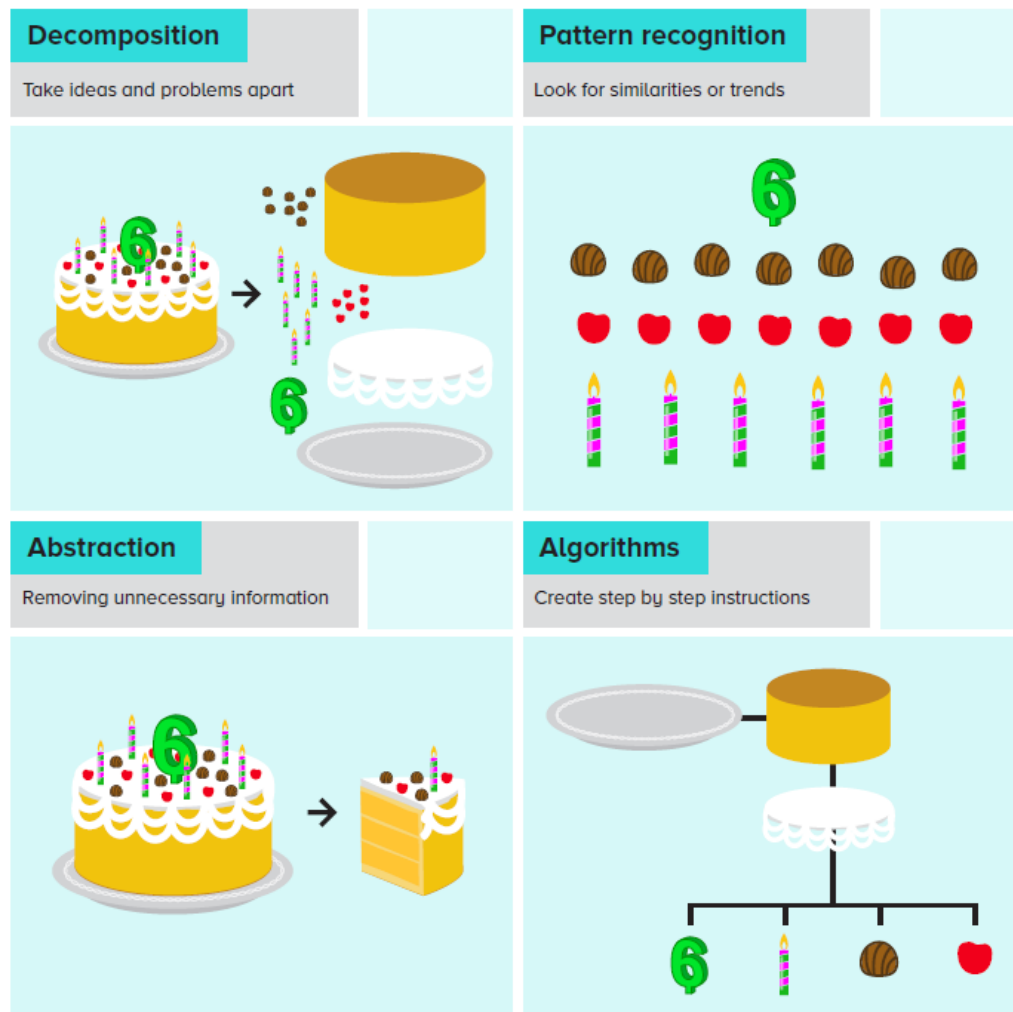
Vraag de leerlingen:

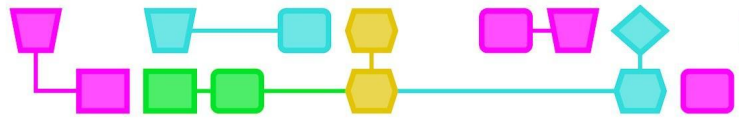
- Wat weet je over hoe een computer en telefoon werkt?
- Kan een computer zelf nadenken? (Waarom wel, waarom niet?)
- Wie bedenkt hoe een computer werkt?

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 decompositie.





Kaartspel Uitdagingen

Introductie (5 min)

Elk groepje van 6-8 leerlingen gebruikt twee kaartspellen voor de opdrachten. De leerlingen gaan de volgende drie uitdagingen oplossen.

1) Zoeken

Doel: zo snel mogelijk de acht azen van het kaartspel vinden.

Schud het kaartspel en leg de gedekte stapel op tafel. Zet de timer en probeer de acht azen zo snel mogelijk te vinden.

2) Sorteert

Doel: de twee kaartspellen op volgorde leggen; van aas tot koning en van klaveren, ruiten, harten tot schoppen.

Schud het kaartspel en leg de gedekte stapel op tafel. Zet de timer en leg de kaartspellen zo snel mogelijk in de goede volgorde.

3) Optellen

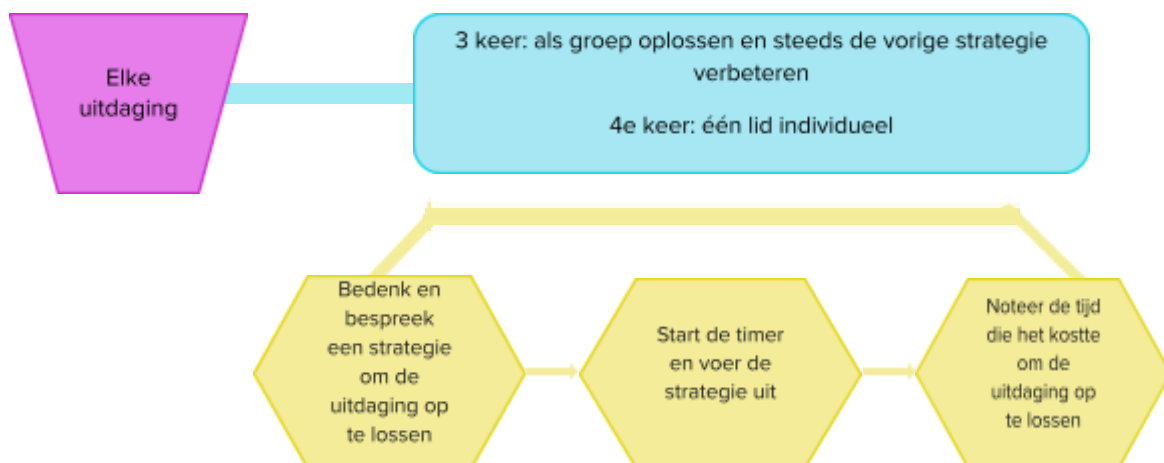
Doel: het optellen van de numerieke waarde van de kaarten.

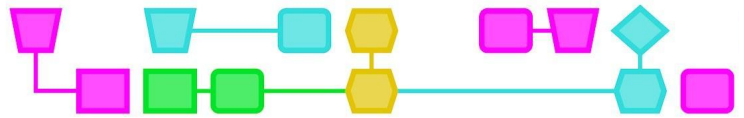
Schud het kaartspel en leg de gedekte stapel op tafel. Haal de eerste 20 kaarten uit de stapel en leg ze weg. Start de timer en tel zo snel mogelijk de numerieke waarde van de kaarten op. (Gebruik de 20 opzij gelegde kaarten om te controleren of het antwoord correct was).

Lesbeschrijving (45 min - 15 minuten per uitdaging of olopende duur: 10-15-20)

De leerlingen doen elke uitdaging vier keer, ze proberen de opdracht in zo kort mogelijke tijd op te lossen door verschillende strategieën te volgen.

Elke uitdaging wordt vier keer opgelost: drie keer als groep (ze kunnen nieuwe strategieën uitproberen of hun gebruikte strategie verfijnen) en één keer individueel door een lid van de groep. Voordat ze elke uitdaging oplossen, spreekt de groep af hoe ze het gaan aanpakken. Bij elke poging wordt de tijd in seconden bijgehouden. Zo kunnen ze bekijken welke strategie het beste werkt. Bespreek na de vierde keer hoe de verschillende groepjes elke uitdaging hebben aangepakt en wat hun strategieën waren.





Conclusie (15-25 min)

Denk met de leerlingen na over het begrip decompositie. Bespreek met de leerlingen dat decompositie inhoudt dat een probleem in kleinere en eenvoudigere delen wordt opgedeeld, waarbij gebruik wordt gemaakt van de capaciteit van verschillende actoren (processors) die het probleem oplossen. De uitdagingen nemen toe in complexiteit, het laat zien wanneer het ontleden van een probleem in subproblemen een aanzienlijk voordeel oplevert en wanneer niet. Dit is een fundamentele competentie bij pogingen om problemen op te lossen door middel van CT.

In de eerste uitdaging wordt duidelijk dat vergelijkingen tussen de tijd die nodig is om de uitdaging in een groep of individueel op te lossen vaak niet heel verschillend zijn: soms gaat het oplossen van de uitdaging individueel bijna even snel als in een groep. Het is zo'n eenvoudige uitdaging dat de tijdswinst door het werk te verdelen niet altijd het tijdsverlies van het verdelen van het werk compenseert.

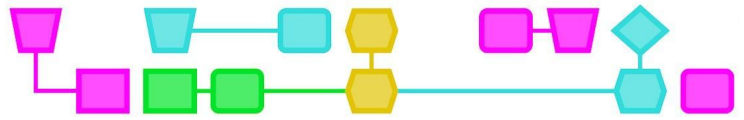
In de tweede uitdaging zijn de voordelen van het feit dat er meerdere mensen in de groep zitten beter merkbaar. Hier ontstaan gevarieerdere strategieën tussen verschillende groepen die zorgen voor discussie over de voor- en nadelen. Veel groepen verdelen deze sorteerausdaging in twee fasen: sorteren en dan nog eens sorteren. Groepen die geen veelvoud van vier leerlingen hebben, moeten minder voor de hand liggende strategieën bedenken, omdat de kaarten vier kleuren hebben.

De derde uitdaging is zo lastig dat elke uitsplitsing en verdeling van het werk door de groepen het proces versnelt. De subtaken worden hierdoor heel foutgevoelig, omdat er veel mensen aan werken. Hierdoor hebben maar weinig groepen het juiste antwoord aan het eind.

Leg de leerlingen uit dat het in de informatica vaak nodig is een probleem in kleinere delen op te splitsen en die afzonderlijk op te lossen en te integreren. Op die manier is de analyse, het ontwerp, programmeren en testen gemakkelijker en efficiënter. Hetzelfde geldt voor complexe gegevens en gegevensstructuren; het ontleden ervan en het analyseren van de onderdelen is gemakkelijker, evenals het onderhouden en opsporen van fouten.

Bespreek verschillende voorbeelden uit hun dagelijks leven, waarin ze decompositie gebruiken om een uitdaging aan te gaan, zoals

- bij een grote taak (bijv. een ingewikkeld kookrecept, een experiment, een toneelstuk, een groepswerk, enz.), waarbij je de taak in kleinere delen verdeelt, die samen tot de algemene oplossing leiden.
- als je een feest organiseert, waarbij de taken onder de aanwezigen worden verdeeld zodat alles goed verloopt: het drinken en het eten kopen, de verschillende gerechten koken, de tafel klaarmaken en afhalen, de muziek kiezen, de versieringen kiezen en kopen, enz.
- bij het opruimen thuis, dat zo wordt verdeeld dat iedereen helpt.



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

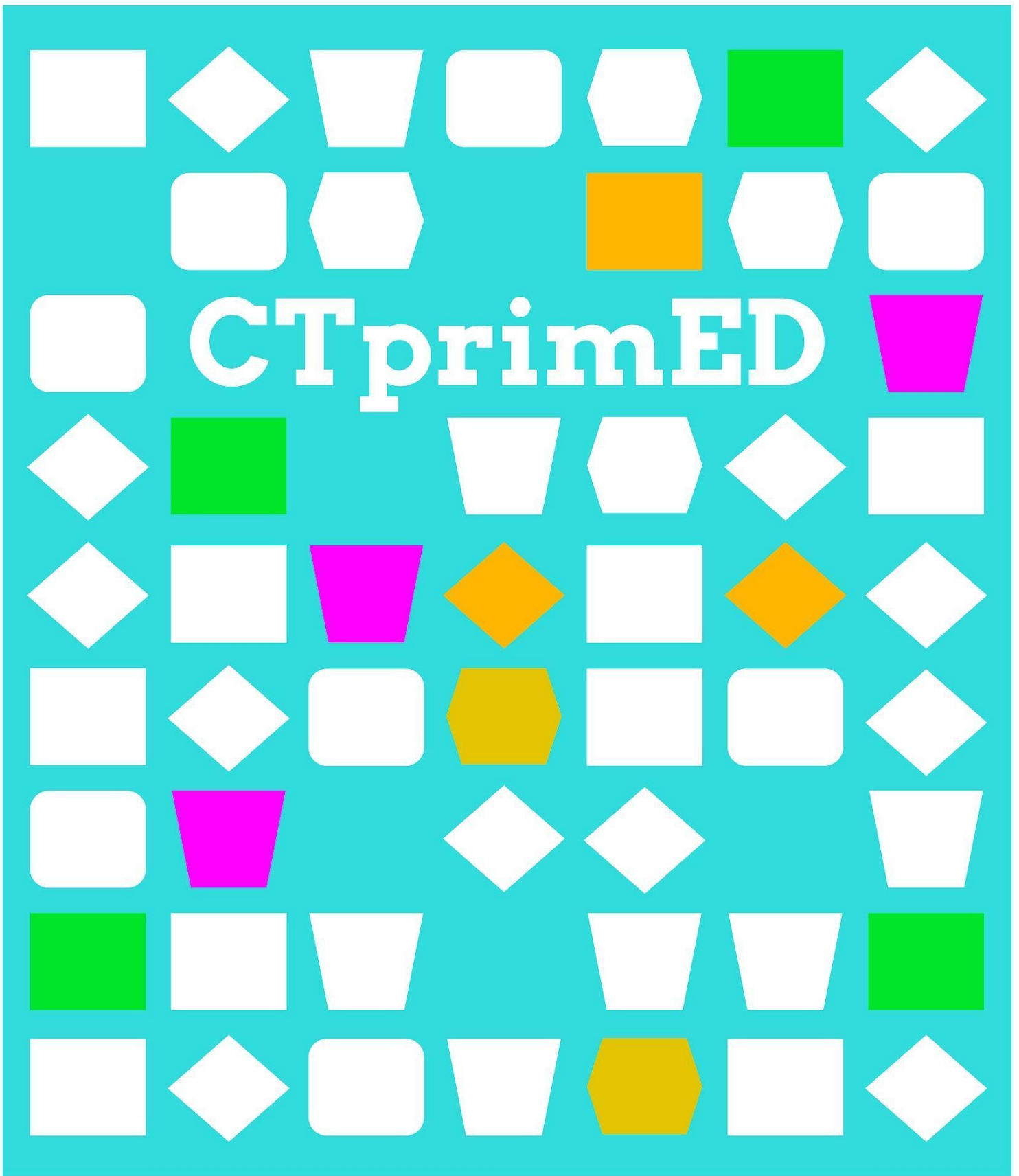


Co-funded by the
Erasmus+ Programme
of the European Union



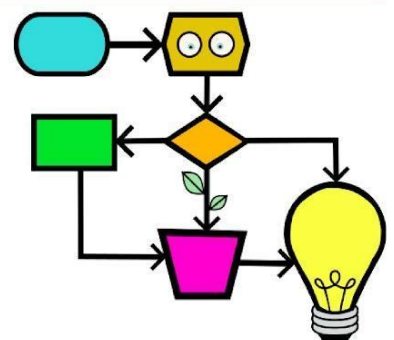
Universidad de Deusto
University of Deusto

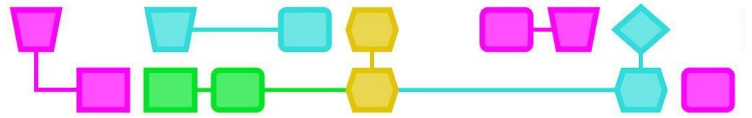
Deusto



Introductie

Patroonherkenning





Samenvatting

Patroonherkenning helpt bij het herkennen van overeenkomsten tussen problemen en systemen. Als je patronen kunt herkennen, helpt dat ook om een oplossing opnieuw te gebruiken voor een ander probleem. In deze les gaan de leerlingen aan de slag met patroonherkenning door kaartjes op te delen in verschillende categorieën. Als extra opdracht is er de mogelijkheid om een (klein) algoritme te schrijven, zodat een computer de kaartjes ook zou kunnen sorteren.

Doelgroep: 6-12 jaar.

Duur: 25 min - extra opdracht 25 min

Leerdoelen:

De leerlingen

- kunnen patronen herkennen in een (data)set;
- maken kennis met patronen uit het dagelijks leven;
- kunnen oplossingen hergebruiken voor een ander probleem.

Online/offline: offline

Computational thinking:

- General skills: creativiteit, samenwerken, beslissingen nemen
- CT-foundations: patroonherkenning, abstractie, algoritmes
- CT-concepten: sorteren, classificeren

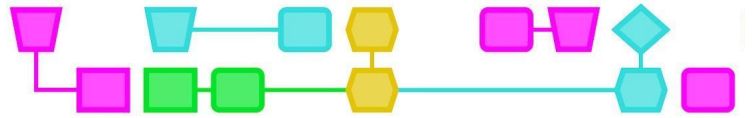
Materialen:

- NL CTPrimED: fotoKaartjes
- Timer op smartboard of telefoon
- (Kleine) bakjes - minimaal 4 per groepje van 3-4 leerlingen
- Werkblad bij extra opdracht

Tip: Plastificeer de kaartjes zodat je ze vaker kunt gebruiken.

Vorbereiding

Print en knip de kaartjes. Verdeel de klas in groepjes van 3-4 leerlingen en zorg dat ieder groepje een eigen oppervlakte (tafel o.i.d.) heeft om de kaartjes op te leggen. Geef bij aanvang van de les elk groepje een set met kaartjes.



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

Vraag de leerlingen:

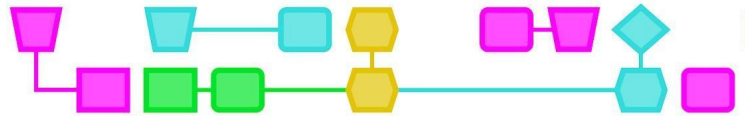
- Wat weet je over hoe een computer en telefoon werkt?
- Kan een computer zelf nadenken? (Waarom wel, waarom niet?)
- Wie bedenkt hoe een computer werkt?

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 patroonherkenning.

| | |
|---|--|
| <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> |
| | |



Lesomschrijving - patroonherkenning

Opdracht 1 - Sorteren (25 min)

Geef de leerlingen de opdracht om in drie minuten de kaartjes te sorteren. Leg, indien nodig, het woord 'sorteren' uit. Geef verder geen uitleg en laat de leerlingen zelf bepalen in welke categorieën ze sorteren.

Zet de timer op drie minuten. Bespreek na de drie minuten wat de leerlingen gedaan hebben.

- Welke categorieën hebben ze gekozen? (De groepjes zullen allemaal verschillende categorieën hebben gemaakt.)
- Hoe hebben ze de categorieën bepaald?
- Wat vonden ze lastig?

VanDale: sor-te-ren
soort bij soort leggen =
uitzoeken.

Ga door naar opdracht 2.

Opdracht 2 - Aantal categorieën maken (10 min)

Geef de leerlingen de opdracht om de kaartjes nu in **vier** categorieën te sorteren. De groepjes bepalen zelf welke categorieën dat zijn. Waarschijnlijk vinden leerlingen het lastig om de kaartjes in vier categorieën te sorteren omdat ze de categorieën die ze in de vorige opdracht hebben gebruikt moeten samenvoegen. De leerlingen zouden bijvoorbeeld de categorieën 'objecten', 'dieren', 'mensen', 'eten', 'vormen' en 'kleuren' kunnen hebben bedacht. Voor deze opdracht moeten ze bepalen in welke vier groepen ze de kaartjes nu gaan verdelen.

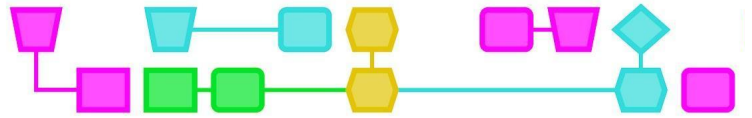
Zet de timer op drie minuten. Bespreek na de drie minuten wat de leerlingen gedaan hebben.

- Welke categorieën hebben ze gekozen? (De groepjes zullen allemaal verschillende categorieën hebben gemaakt, er is geen goed of fout.)
- Hoe hebben ze de categorieën bepaald?
- Wat vonden ze lastig?

Leg uit dat ze tijdens het sorteren hebben gezocht naar patronen: ze probeerden te ontdekken welke dingen ongeveer hetzelfde zijn en welke niet. Ze hebben een snelle blik op de kaartjes geworpen en zagen bijvoorbeeld dat er eten en dieren tussen zaten. Bij het maken van de categorieën waren ze steeds op zoek naar patronen: welke overeenkomsten en welke verschillen zijn er? En op basis daarvan hebben ze heel snel besloten welk kaartje in welke categorie moest.

Tip: Bij heel veel spellen maak je gebruik van patroonherkenning. Denk aan Minecraft, waarin je bepaalde 'recepten' (patronen) nodig hebt om gereedschap te bouwen, of Roblox, waarin je leert hoe de verschillende spellen werken door de patronen te herkennen.

Afsluiting (5 min)

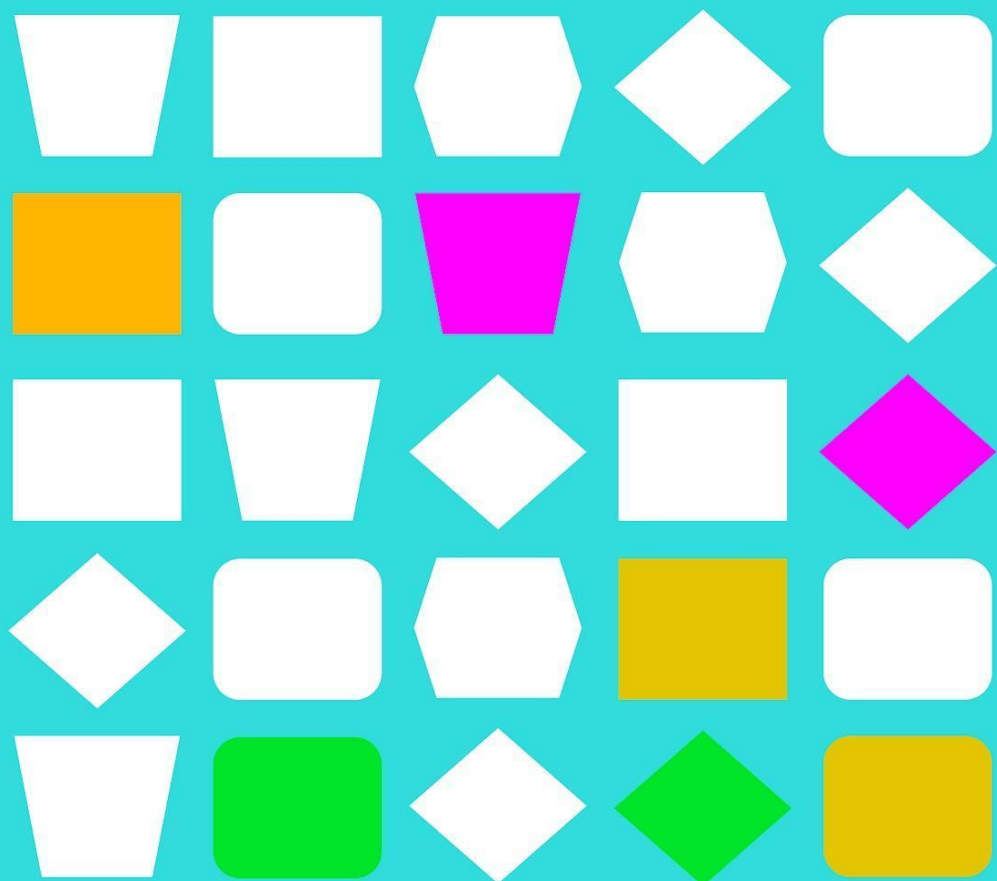


Al deze kaartjes bij elkaar kun je ook data noemen. Computers werken vaak met heel veel data. Om een computer te kunnen vertellen hoe die met al deze data om moet gaan, is het belangrijk om patronen in een dataset te herkennen. Als je bijvoorbeeld een app over voedsel ontwerpt, hoeft de computer niet na te denken over elk los ingrediënt, maar alleen over de algemene kenmerken die ze hebben als categorie (bijvoorbeeld: type, gewicht, kleur of verpakking). Op deze manier is de datastructuur in een model gezet en kan het ingevuld worden met specifieke data. Deze datastructuur is erg specifiek: wanneer je bijvoorbeeld een app ontwerpt voor in de supermarkt zijn de categorieën die net werden genoemd niet belangrijk. Dan zijn bijvoorbeeld de prijs, korting en kwaliteit belangrijker. Patronen zijn daarom erg specifiek, waardoor het lastig kan zijn om te bepalen naar welk specifiek patroon je moet zoeken in de data.

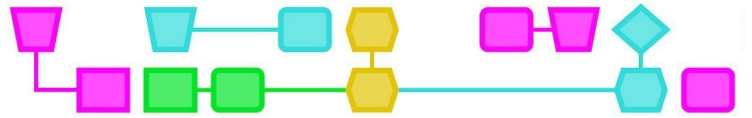
In het dagelijks leven gebruik je ook vaak patronen. Bijvoorbeeld bij het oplossen van een rekensom. Denk aan het patroon van een reeks keersommen: je kunt zes keer vijf makkelijker oplossen als je al weet wat vijf keer vijf is. Je kunt ook makkelijk dieren herkennen omdat je weet in welk patroon ze passen (heeft het poten, een vacht, snorharen, enz.). Patronen maken het leven net iets makkelijker.

Tip: Je kunt de antwoorden van leerlingen gebruiken om het begrip 'decompositie' uit te leggen. Dit kun je doen door het volgende te vertellen:

Decompositie is wanneer je de specifieke details kan negeren en de belangrijke grote lijnen kan zien. Dat doe je ook als je bijvoorbeeld deze kaartjes 'scant' om de categorieën te bepalen. Je gaat niet kijken naar de details van ieder ding, maar welke dingen ongeveer bij elkaar passen. Als je een probleem moet oplossen kun je ook kijken naar de grote lijnen, in plaats van naar ieder detail. Je zult zien dat je dan sneller een oplossing kan bedenken.



Bijlagen



Bijlage 1: Extra opdracht - Sorteeralgoritme voor dierenkaartjes (25 min)

Voorwaardes en regels (5 min)

Geef elk groepje 4 bakjes en laat de leerlingen alle kaartjes waar geen dieren op staan uit de set halen. Leg aan de leerlingen uit dat je ook een computer kunt gebruiken om de kaartjes (de data) te sorteren. In deze opdracht bedenken ze een algoritme dat de kaartjes sorteert.

Om deze instructie zo duidelijk mogelijk te maken kun je gebruikmaken van voorwaarden. Dit is een 'als..., dan...'-regel die stelt dat er iets moet gebeuren voordat het andere gebeurt. Een voorbeeld van een voorwaarde voor het sorteren zou kunnen zijn: 'Als het poten heeft, is het een dier en **dan** moet het in de eerste bak' of: 'Als het veren heeft, **dan** moet het in de tweede bak.'

Algoritmes (15 min)

Deel het werkblad uit. De leerlingen bedenken eerst vier (of meer) categorieën en proberen 'als..., dan...'-regels (voorwaarden) te bedenken waarmee de computer deze categorieën kan sorteren.

Zet de timer op tien minuten. Na tien minuten stoppen de leerlingen met de opdracht. Kies een paar groepjes waarvan je het algoritme dat ze geschreven hebben probeert. Jij bent dan de computer. Laat het groepje hun voorwaarden vertellen en ga aan de slag. Zijn de instructies duidelijk genoeg?

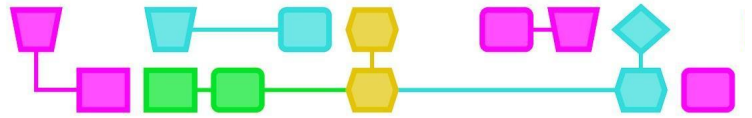
Leerlingen hebben waarschijnlijk geen perfect sorteeralgoritme geschreven. Dit komt doordat het erg lastig is om de voorwaarden duidelijk genoeg te maken. Leerlingen kunnen bijvoorbeeld de volgende voorwaarde hebben: 'Als het veren heeft, is het een vogel en moet het in de bak met vogels.' Maar als ze ook een bak met boerderijdieren hebben, kan een kip in allebei de bakken.

Of als ze de volgende voorwaarde hebben: 'Als het poten heeft, is het een dier en moet het in de bak met dieren.' Maar een stoel heeft ook poten, dus is het onduidelijk waar die gesorteerd zou moeten worden. Een computer kan niet nadenken, waardoor sorteeralgoritmes moeilijk zijn om te schrijven, zelfs als je alleen dierenkaarten sorteert.

Afsluiting (5 min)

De leerlingen hebben 'als..., dan...'-regels gemaakt waarmee een computer de dierenkaartjes kan sorteren. Omdat een computer niet voor zichzelf kan nadenken, moet je bij het maken van een sorteeralgoritme heel precies zijn en veel uitdenken en dat is waarom ze vaak lastig zijn om te schrijven.

Leerlingen voeren zelf ook algoritmes uit! Denk aan het poetsen van hun tanden, of aan het doen van een dansje. Dit zijn allemaal stap-voor-stapinstructie die ze kunnen volgen om een bepaalde taak makkelijker of soepeler te laten verlopen.



Sorteer de kaarten - Algoritme

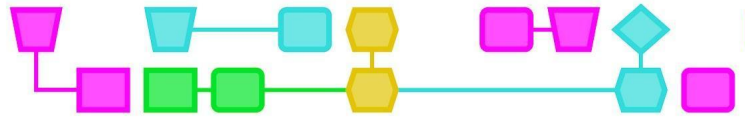
Sorteer de dierenkaartjes in vier categorieën.

Onze categorieën zijn:

-
-
-
-

De regels voor de computer zijn:

- Als..... → Dan.....
- Als..... → Dan.....
- Als..... → Dan.....
- Als..... → Dan.....
- Als..... → Dan.....
- Als..... → Dan.....
- Als..... → Dan.....



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

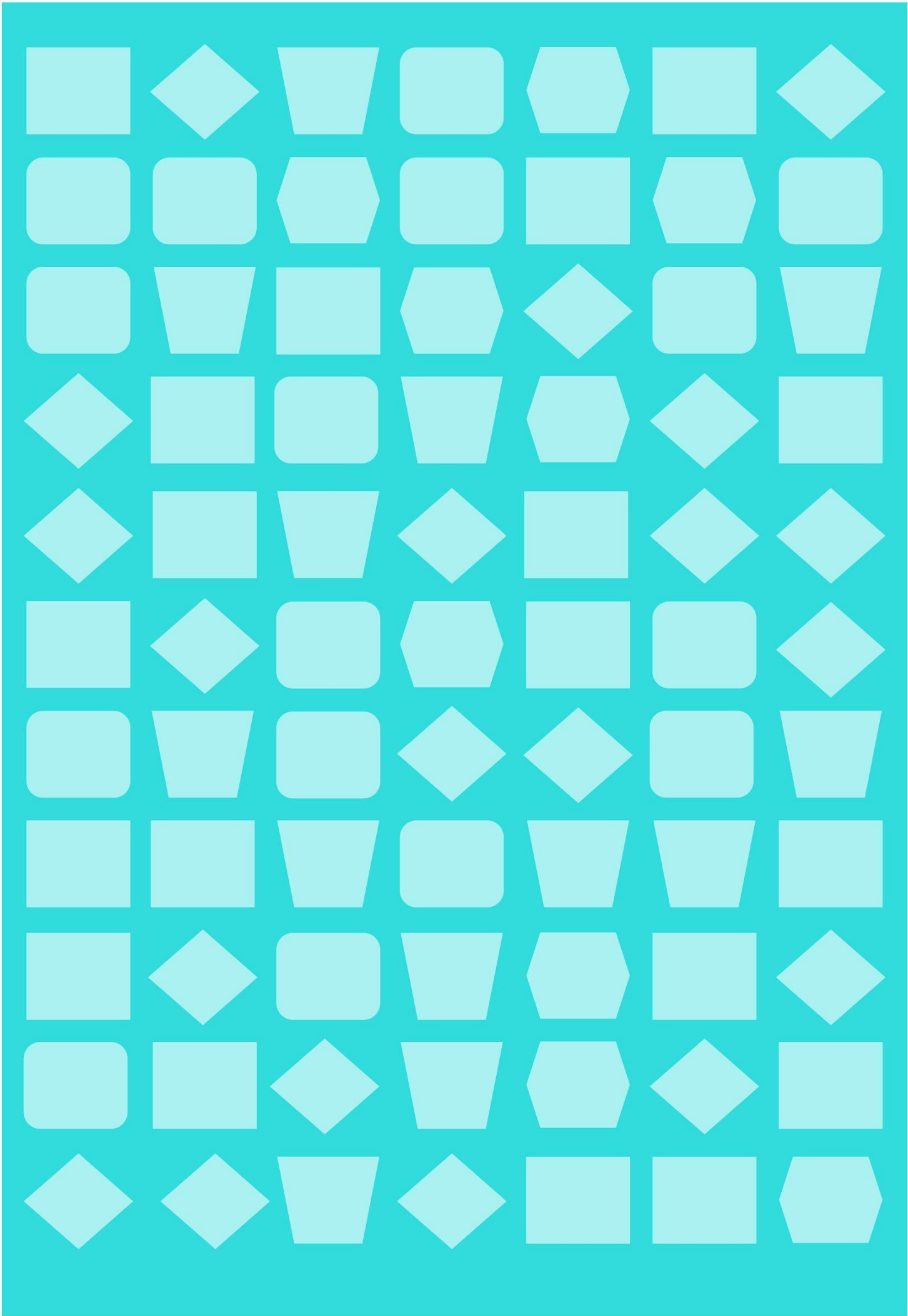


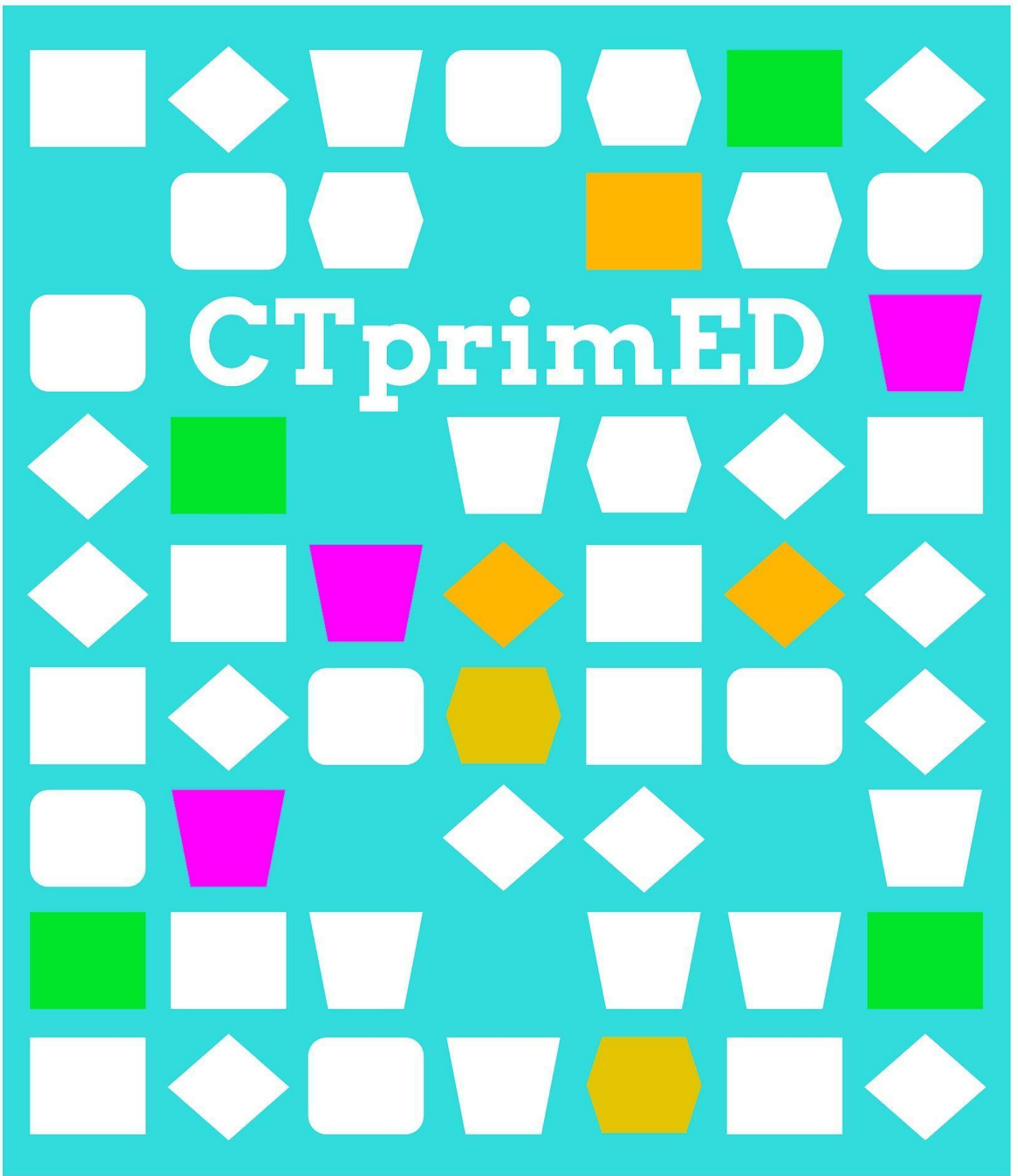
Co-funded by the
Erasmus+ Programme
of the European Union



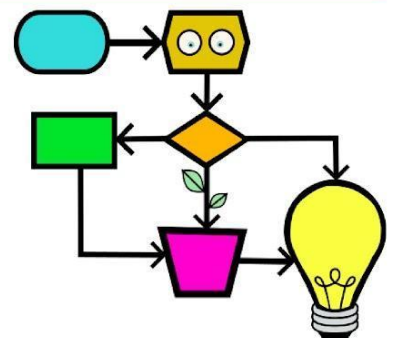
Universidad de Deusto
University of Deusto

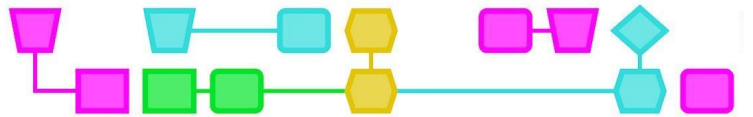
Deusto





Offline
programmeren





Samenvatting

Een algoritme is een stapsgewijze instructie om een probleem op te lossen. Denk aan een recept, een script voor een toneelstuk of een programma voor de computer. In deze les leren de leerlingen wat een algoritme is. Het begint met een spel dat de leerling laat zien welke instructies een computer kan begrijpen. Hierna programmeren ze elkaar als danscomputer.

Doelgroep: 6-12 jaar (voor kinderen die niet gemakkelijk lezen en schrijven hebben we een differentiatie toegevoegd).

Duur: 55 minuten

Leerdoelen:

De leerlingen leren

- wat een algoritme is;
- hoe je een algoritme in het dagelijks leven kunt gebruiken;
- wat een lus en een voorwaarde zijn en hoe je die in een programma kunt gebruiken.

Online/offline: offline

Computational thinking:

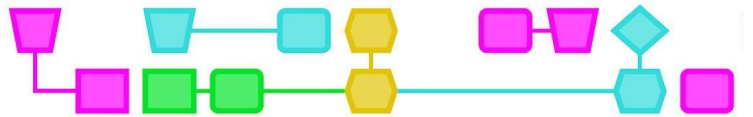
- General skills: creativiteit, samenwerken, logisch nadenken
- CT-foundations: algoritmen, patroonherkenning
- CT-concepten: variabelen, reeksen, lussen en voorwaarden

Materialen:

- Presentatie: De Danscomputer
- Dans Pictogrammen
- Pen
- Papier

Vorbereiding

Lees de les door en zet de presentatie klaar. Bedenk voor de eerste opdracht waar plek A en plek B in de klas zijn.



Lesbeschrijving

Introductie - Van A naar B (10 min)

Leg aan de leerlingen uit dat ze hun eigen algoritme gaan maken. Een algoritme is een proces of een reeks regels die gevolgd moeten worden bij berekeningen of andere probleemoplossende handelingen, voornamelijk door een computer.

Vertel de leerlingen dat jij in deze opdracht de computer bent en dat ze jou moeten 'programmeren' om van punt A naar punt B te komen in de klas. Kies een plek A, het beginpunt, en het eindpunt B. Zorg dat er een paar obstakels in de weg staan, zoals een tafel waar je onderdoor of overheen moet.

Kies 2-3 leerlingen die om de beurt proberen om jou naar punt B te krijgen. Ze moeten instructies geven zoals: 'Doe twee stappen vooruit' en: 'Draai naar rechts.' Vat de instructies extreem letterlijk op. Als leerlingen bijvoorbeeld zeggen: 'Ga vooruit', blijf dan naar voren lopen totdat je ergens tegenaan botst en blijf zelfs dan doen alsof je naar voren loopt. Zo leren de leerlingen dat ze ook 'Stop' moeten zeggen. Je zult zien dat leerlingen moeite hebben alles zo letterlijk mogelijk uit te leggen.

Vertel de leerlingen na afloop dat ze een algoritme hebben gemaakt: een proces of een reeks regels die gevolgd moeten worden bij berekeningen of andere probleemoplossende handelingen, om van punt A naar punt B te gaan. Ze hebben jou stap voor stap instructies gegeven en iedere stap heel letterlijk uitgelegd.

Choreografie - Algoritme (20 min)

Doel: algoritme, duidelijke instructie

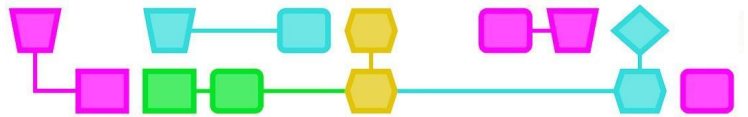
Leg uit dat algoritmes voor verschillende dingen gebruikt kunnen worden, zoals je veters strikken, een recept volgen of voor een danschoreografie. Tijdens deze opdracht bedenken de leerlingen een algoritme voor een dans die een ander groepje gaat uitvoeren. Verdeel de leerlingen in groepjes van 3-4 leerlingen. Elk groepje maakt een dansje (choreografie) van 1-2 minuten.

Stap 1: De leerlingen denken na over welke bewegingen ze willen gebruiken. Dit kan een bestaande dans zijn, zoals 'hoofd, schouders, knie en teen', of een zelfverzonnen dans.

Stap 2: De leerlingen schrijven de dans heel precies uit op papier, zodat een ander groepje hun dans kan uitvoeren als ze de instructies lezen.

Differentiatie: gebruik de danspictogrammen voor leerlingen die nog niet goed kunnen lezen of schrijven.

Stap 3: Leerlingen wisselen de choreografieën uit met een ander groepje. De groepjes krijgen vijf minuten om de choreografie te lezen en te repeteren en vervolgens voor de hele groep te dansen. Benadruk dat de leerlingen alleen mogen uitvoeren wat er op het briefje staat.



Stap 4: Bekijk elkaars dansen en bespreek de volgende vragen met de leerlingen:

- Voerde het groepje de dans precies hetzelfde uit?
- Wat ging er wel goed en wat niet?

De leerlingen hebben het dansje waarschijnlijk niet precies hetzelfde uitgevoerd. Benoem dit en leg uit dat algoritmes stap-voor-stapinstructies zijn die je kunt gebruiken om een probleem op te lossen. Je kunt een algoritme ook gebruiken om een computer iets voor jou te laten doen. In dit geval was het algoritme de beschrijving van de dans en het groepje dat de dans uitvoerde was de computer. Bij de beschrijving van de stappen is precisie belangrijk, omdat de onduidelijkheden fouten in het programma kunnen veroorzaken (= andere uitvoering van de dans). Precieze instructies geven is erg lastig, omdat je alles heel duidelijk uit moet schrijven. Dat is waarschijnlijk waarom de dansjes net iets anders gingen dan hoe ze bedacht waren.

Tip: Zet een muziekje aan bij de uitvoering van de choreografie.

Choreografie - Lussen en voorwaarden (20 min)

Doel: Leren wat lussen en voorwaarden zijn.

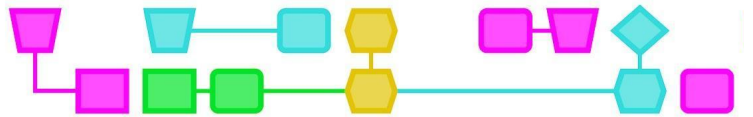
Bespreek met de leerlingen:

- Wat zouden jullie anders doen nu je de uitvoering van de dans hebt gezien?
- Hoe zouden we ervoor kunnen zorgen dat de dans precies zo wordt uitgevoerd als de bedoeling is?
 - Bewegingen versimpelen
 - Duidelijker omschrijven
 - Afspreken welke beweging je maakt
 - Dingen herhalen

Start de presentatie *De Danscomputer*. Laat de eerste dia zien en leg de leerlingen uit dat ze de opdracht nog een keer gaan doen, maar dan met behulp van pictogrammen. Leg uit dat elk pictogram een code is voor een beweging die de computer kan uitvoeren.

- Laat dia 2 zien en leg de pictogrammen uit.
- Laat dia 3 zien en voer dit dansje samen met de klas uit. Bespreek hoe je kunt laten zien dat ze het dansje drie keer moeten herhalen.
- Laat op dia 4 zien dat je de stappen allemaal kunt herhalen, maar dat dat heel onduidelijk zou worden. Bespreek hoe je het korter zou kunnen opschrijven.
- Laat dia 5 zien. Dit noem je een lus, een herhaling. Bespreek of ze nog een plek zien om een lus toe te passen (bij het klappen van de handen).
- Laat hierna dia 6 zien. Dit noem je een voorwaarde, een 'als..., dan...'-regel. Als iets gebeurt, dan stopt iets anders. *Als* de muziek aan is, *dan* wordt het dansje herhaald. *Als* de muziek uitgaat, *dan* stopt het programma.

De leerlingen gaan nu in groepjes een choreografie maken met behulp van de pictogrammen. Zorg dat ze ten minste twee lussen en een voorwaarde gebruiken.



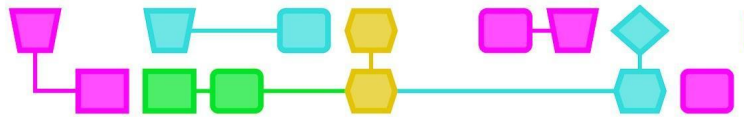
Bespreek na de opdracht wat de leerlingen vandaag hebben geleerd. Ze kunnen nu een danscomputer programmeren met een functie, lus en voorwaarden. Bij de danscomputer hebben ze gebruikgemaakt van pictogrammen als codetaal. Op de computer kunnen ze andere codetalen gebruiken om dingen te programmeren, zoals Scratch, MakeCode of Kodetu. In deze codetalen kunnen ze ook functies, lussen en voorwaarden gebruiken. In de informatica ('computerwetenschap') zijn algoritmes nodig om computerprogramma's te coderen.

| | |
|------------|--|
| Algoritme | Een reeks logische instructies voor het uitvoeren van een taak. In de informatica zijn algoritmen nodig om computerprogramma's te ontwerpen. |
| Programma | Opeenvolging van instructies voor een computer. |
| Functie | Een blok georganiseerde, herbruikbare code die wordt gebruikt om een enkele, verwante actie uit te voeren. |
| Instructie | Een specifieke opdracht die door een computer uitgevoerd kan worden. |
| Lus | Een herhaling van een (reeks) instructie(s). |
| Voorwaarde | Een 'als..., dan...'-regel die stelt dat er iets moet gebeuren voordat het andere gebeurt. |

Tip: Laat in Scratch, Kodetu of MakeCode zien hoe een functie, lus en voorwaarde er in codetaal uitzien.

Afsluiting (5 min)

Vertel de leerlingen dat ze deze les zelf een algoritme hebben gemaakt! Leg uit dat er heel veel algoritmes in de wereld zijn. Bijvoorbeeld bij stoplichten, of op sociale media zoals TikTok. Maar ook bij tandenpoetsen, de tafel dekken en bij rekensommen gebruik je stap-voor-stapinstructies die dingen makkelijker laten gaan. Ook lussen en voorwaarden zie je terug in het dagelijks leven, bijvoorbeeld in een recept waar je iets moet herhalen, maar ook bij de ongeschreven regels dat je als je naar de wc bent geweest je handen moet wassen, of dat je als je op reis gaat een koffer moet inpakken. Algoritmes, lussen en voorwaarden zijn overal!



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

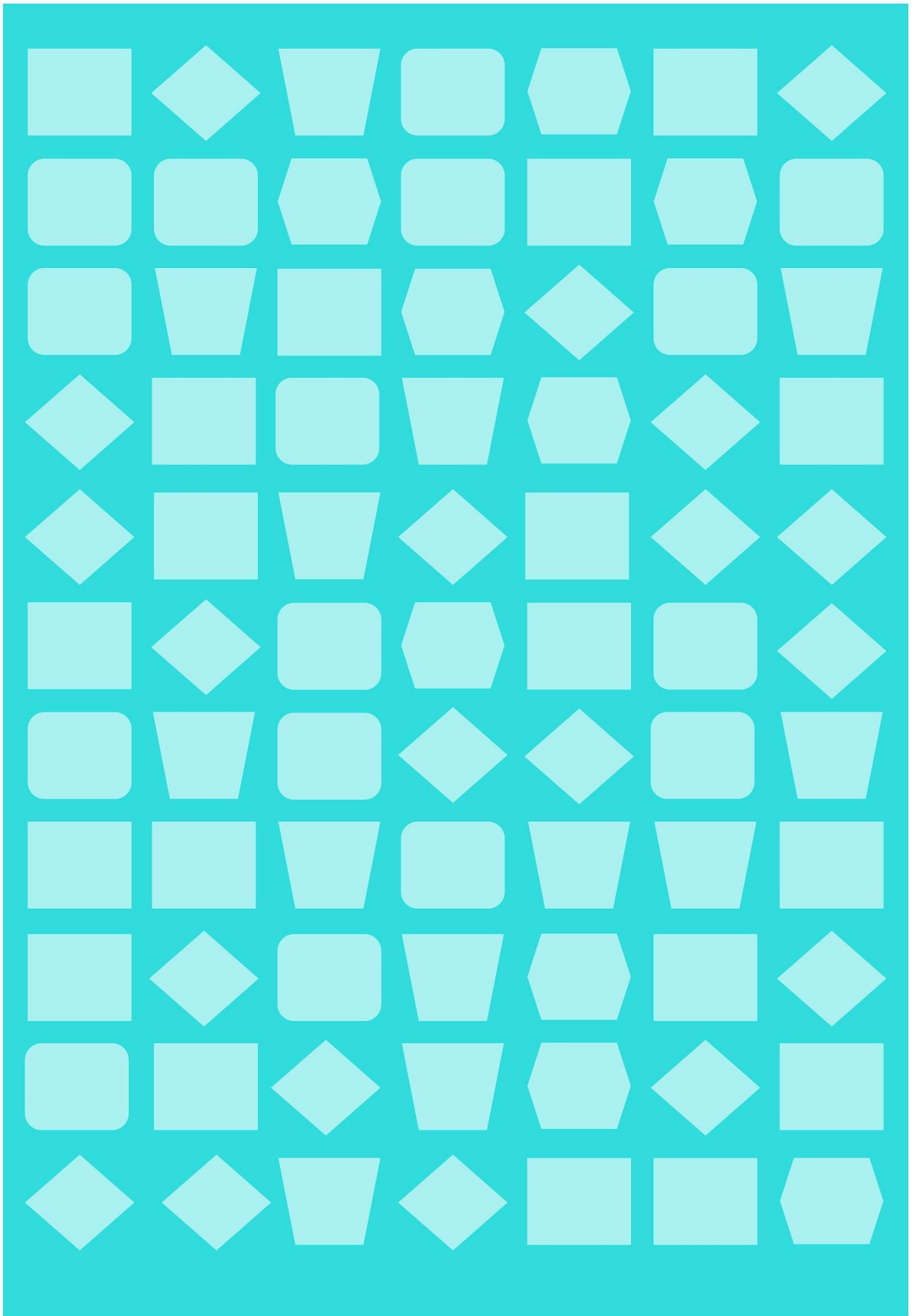


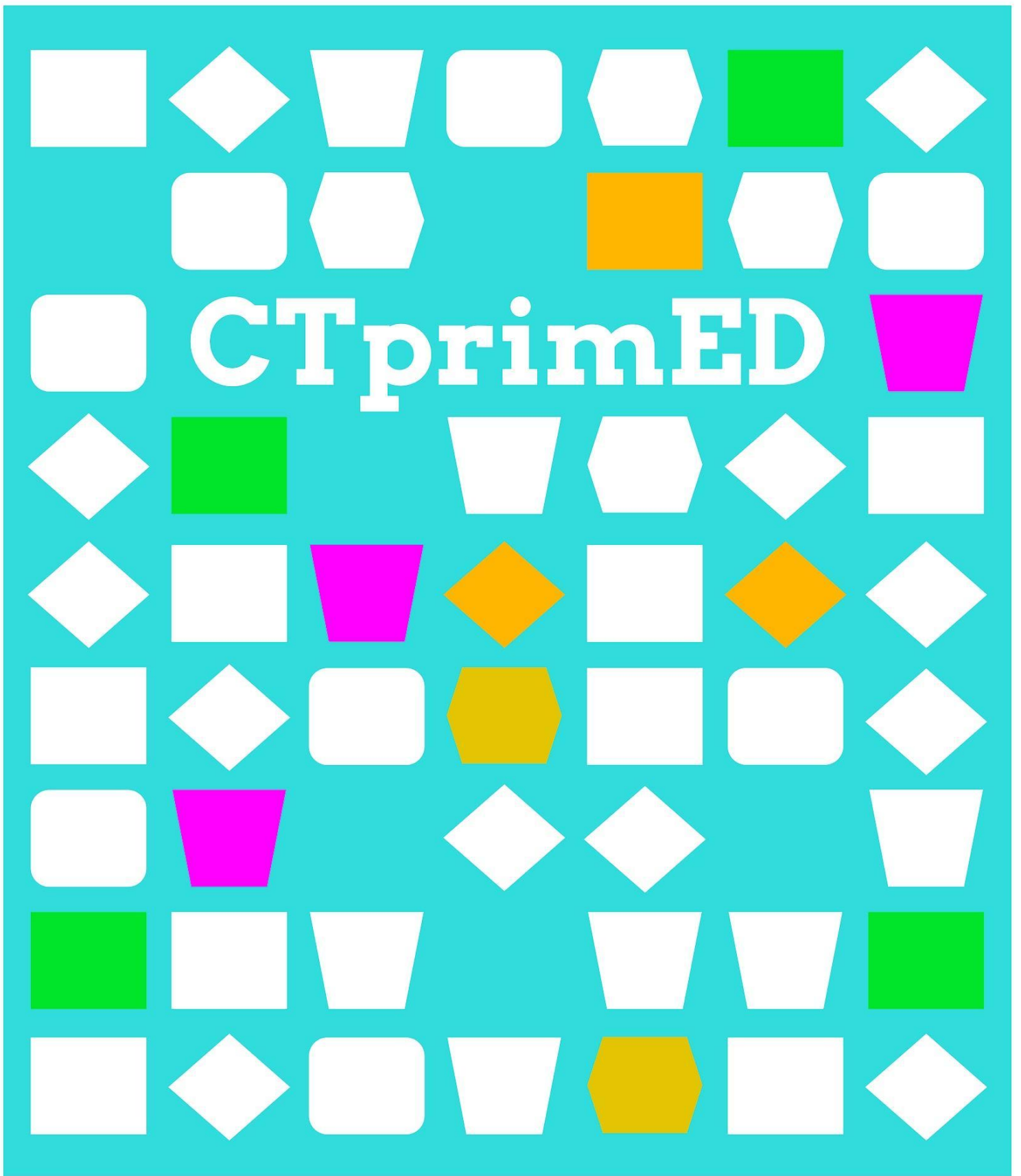
Co-funded by the
Erasmus+ Programme
of the European Union



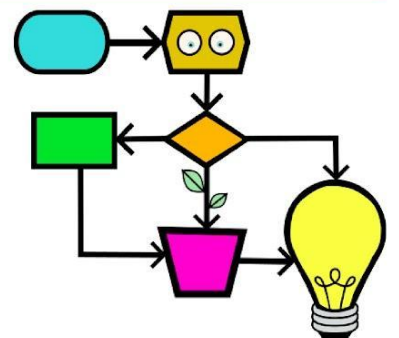
Universidad de Deusto
University of Deusto

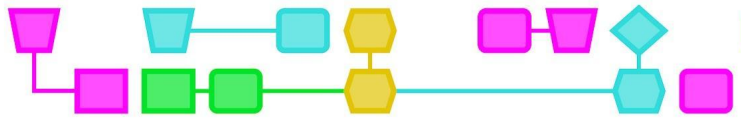
Deusto





Online programmeren





Samenvatting

Tijdens deze activiteit maken de leerlingen kennis met het concept algoritmen en de eerste stappen om te programmeren met Scratch. Het is een set van vijf opeenvolgende activiteiten die helpen de basisfuncties van programmeren te begrijpen.

Doelgroep: Leerlingen vanaf 7 jaar met basiskennis van lezen, schrijven en rekenen.

Duur: 120 minuten

Leerdoelen:

- Algoritmen begrijpen, interpreteren en ontwerpen.
- De instructies van een programma definiëren met behulp van pseudocode.
- Van pseudocode naar Scratch-code gaan.
- Programma's van gemiddelde complexiteit schrijven.

Online/offline: offline

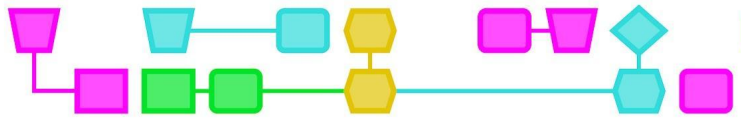
Computational thinking:

- Algemene vaardigheden: samenwerking, creativiteit, algoritmisch denken, logisch denken
- CT-foundations: algoritme, patroonherkenning
- CT-concepten:
 - Basisprogrammeerstructuren
 - Sequentieel
 - Iteratief (herhalend, loops)
 - Conditioneel (selectie)
 - Procedures
 - Functies
 - Coördinatie en synchronisatie

Materialen:

- Schoolbord, potlood, papier en gum
- Gedrukte werkbladen 1-4 voor elke leerling
- Indien mogelijk: toegang tot een computer met internetverbinding

Vorbereiding: Print werkblad 1-4 voor elke leerling



Programmeren voor beginners

Introductie (5 min)

Hieronder volgt een reeks van vijf opeenvolgende activiteiten die helpen om de basisfuncties van programmeren te begrijpen. Ze leggen de basis om te beginnen met programmeren in Scratch of vergelijkbare programmeerprogramma's. De activiteiten gaan van makkelijk naar moeilijk en eindigen met een activiteit op gevorderd niveau. Voor deze activiteit verwachten we een minimale kennis van Scratch.

Lesbeschrijving (120 min)

UITDAGING 1 - EEN VIERKANT TEKENEN

De leerlingen proberen de stappen te beschrijven die nodig zijn om een vierkant te tekenen. Vraag twee leerlingen naar voren. Eén leerling tekent, de andere leerling geeft instructies om een vierkant op het bord/papier te tekenen. (De leerling die tekent mag van tevoren niet weten wat hij of zij gaat tekenen.) Laat de leerlingen het ook proberen met een ingewikkelder figuur dan een vierkant.

Bespreek de opdracht met de klas:

- Hoe is het gegaan?
- Is de informatie volledig?
- Zijn de commando's juist?
- Welke instructies hebben we gemist?
- Hoe zouden we de instructies kunnen verbeteren?

De leerlingen krijgen nu een reeks ongeordende instructies (Werkblad 1) en zetten die op volgorde om een vierkant op het bord/papier te tekenen.

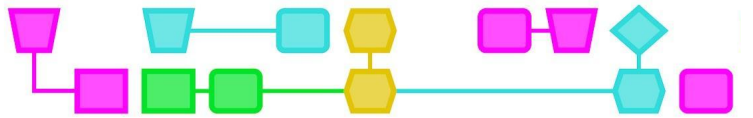
Set ongeordende instructies:

- Trek een rechte lijn naar rechts van 20 cm.
- Zet het schilderij op het bord.
- Draai 90 graden naar rechts.
- Trek een rechte lijn naar rechts van 20 cm.
- Trek een rechte lijn van 20 cm naar rechts.
- Draai 90 graden naar rechts.
- Teken een rechte lijn van 20 cm naar rechts.
- Ga in het midden van het bord staan.
- Pak een penseel/potlood.
- Draai 90 graden naar rechts.

Controleer met de leerlingen de juiste volgorde. Zie voor de oplossing van de activiteiten Bijlage 3.

Vraag de leerlingen naar de lijst met instructies te kijken. Stel aan hen de volgende vragen:

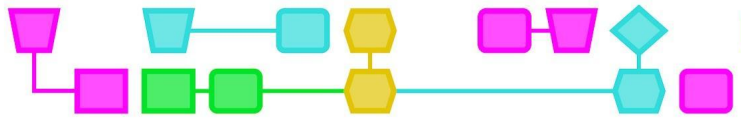
- Is het je opgevallen dat er herhaalde sets instructies zijn?
- Kun je zien welke reeks instructies wordt herhaald en hoe vaak?
- Hoe zouden we dezelfde instructies kunnen uitleggen met minder regels?



Mogelijke oplossing:

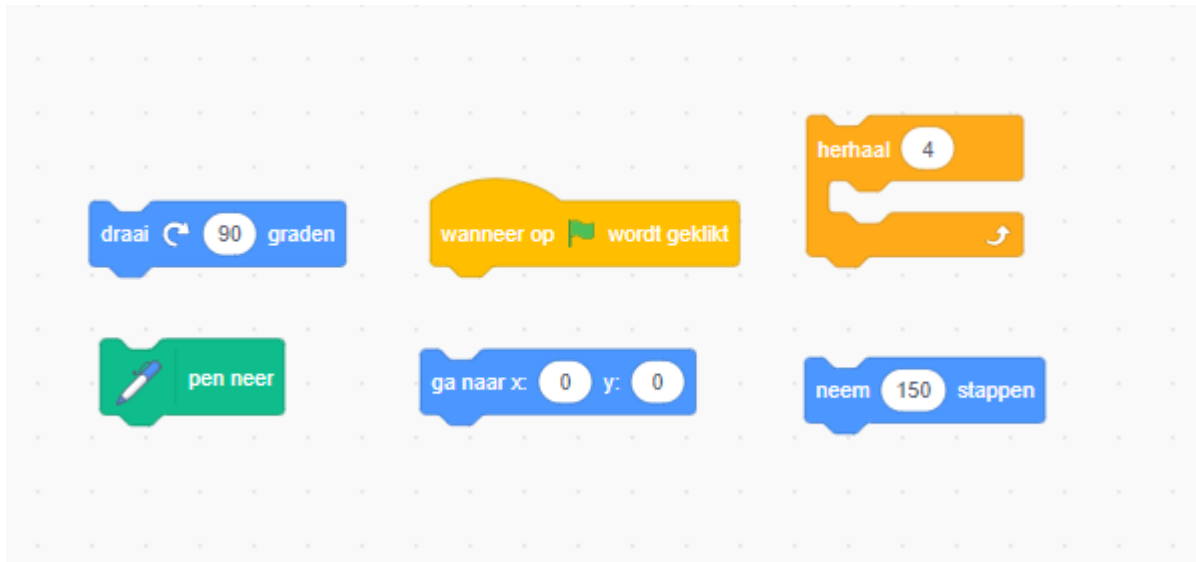
- Neem een schilderij.
- Ga in het midden van het bord staan.
- Leg het schilderij op het bord.
- Trek een rechte lijn naar rechts van 20 cm.
- Herhaal de onderstaande punten vier keer:
 - Trek een rechte lijn naar rechts van 20 cm.
 - Draai 90 graden naar rechts.

Door deze activiteit ontdekken de leerlingen hoe belangrijk het is om instructies in de juiste volgorde en op een precieze manier te geven, en leren ze zelfs hoe ze efficiëntere programma's kunnen maken door herhalingen of loops te gebruiken.



UITDAGING 2 - EEN VIERKANT TEKENEN MET BEHULP VAN SCRATCH

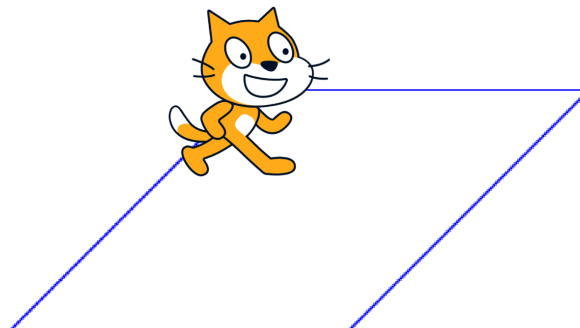
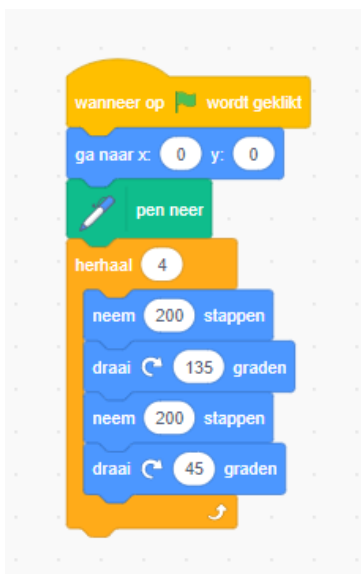
Deel Werkblad 2 uit aan de leerlingen. Vraag de leerlingen hetzelfde vierkant te maken, maar dan met behulp van Scratch-programmeerblokken. Deze activiteit kan online en offline worden gedaan. In de offline versie knippen de leerlingen de blokken uit zodat ze ze op de juiste manier kunnen groeperen.

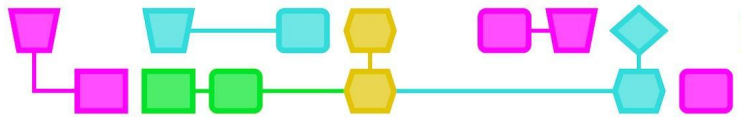


Voor de online versie van Uitdaging 2 en 3 gaan de leerlingen naar: <https://scratch.mit.edu/> en maken ze de geometrische vormen met behulp van de avatar.

UITDAGING 3 - ANDERE VORMEN

Deel Werkblad 3 uit. Laat de leerlingen de juiste code met de juiste vorm verbinden.

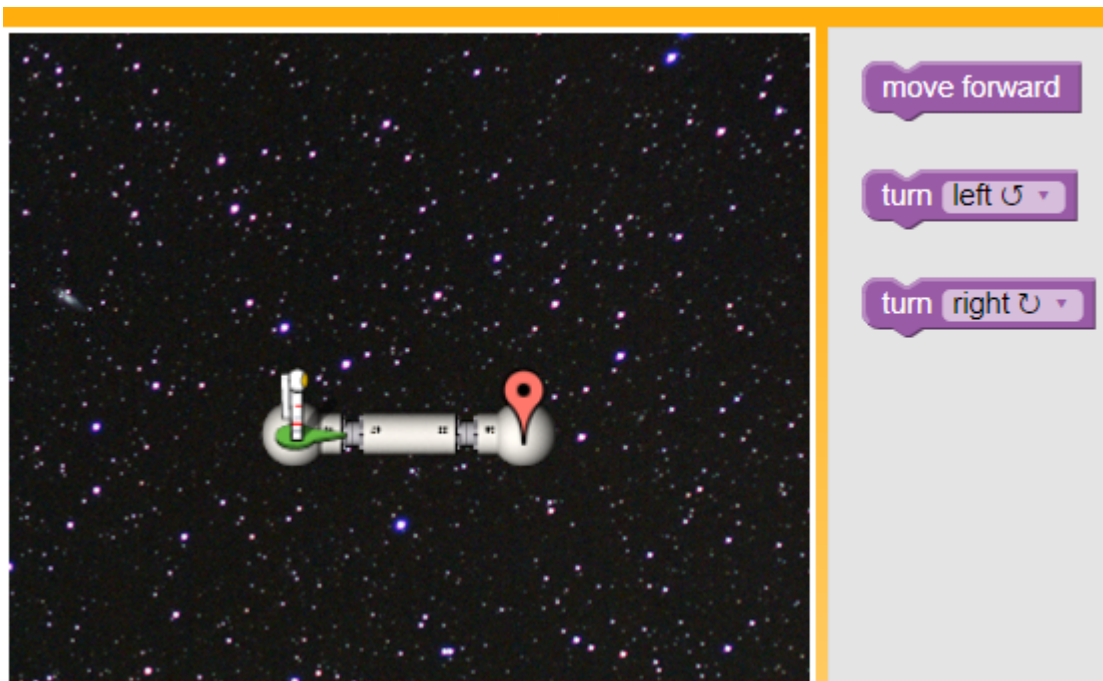


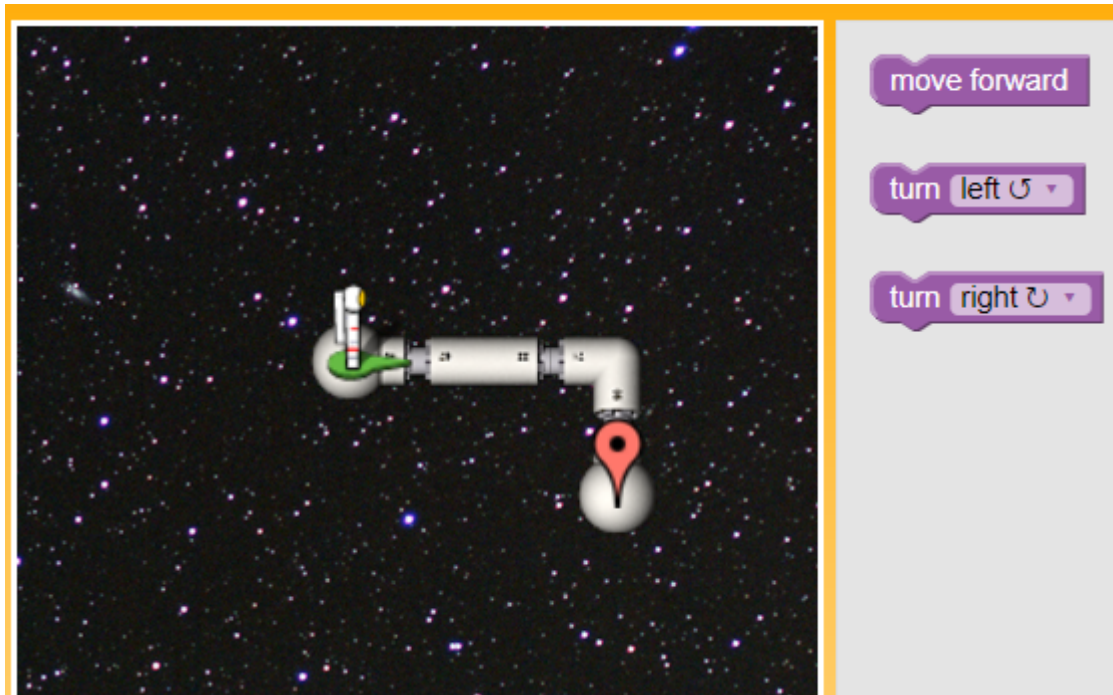
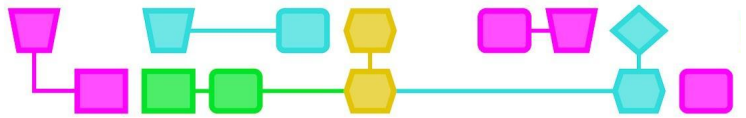


EXTRA - HELP DE ASTRONAUT OM DE MISSIE TE VOLBRENGEN (ENGELS)

Als de leerlingen Scratch begrijpen, kunnen ze het principe van deze programmeertaal ook gebruiken in andere omgevingen, zoals Blockly in de Kodetu-omgeving. Helaas is Kodetu nog niet in het Nederlands beschikbaar, maar leerlingen die beschikken over basiskennis van het Engels kunnen deze opdracht ook uitvoeren.

In Kodetu moeten de leerlingen de astronaut helpen de missie te volbrengen. Ze kunnen deze activiteit online doen (<http://kodetu.org/>).





ACTIVITEIT VOOR GEVORDERDEN - MAAK JE EIGEN INTERACTIEVE ANIMATIE!

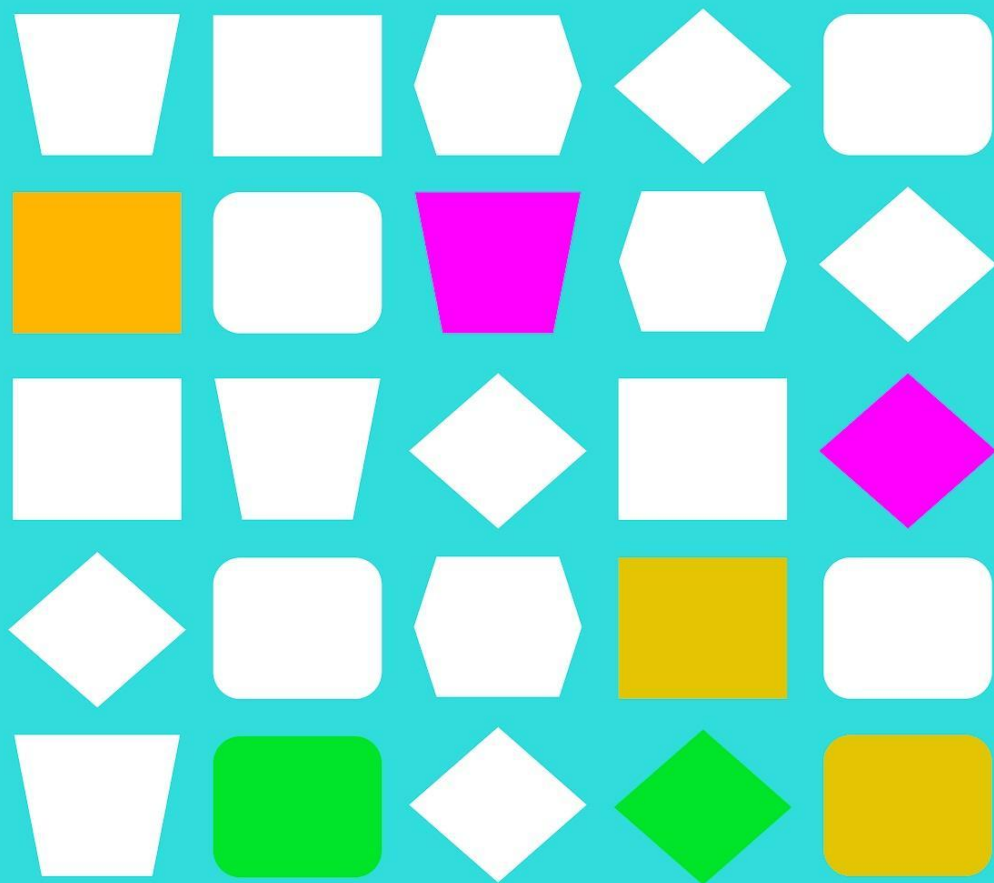
Nu de leerlingen enige kennis hebben van hoe Scratch werkt, kunnen ze een eigen creatieve opdracht doen in Scratch. Vraag de leerlingen een animatie om zichzelf te maken. Ze moeten beelden en geluiden combineren tot een interactieve collage. Moedig de leerlingen aan om te experimenteren met personages, kostuums, achtergronden, verschijning blokken en geluiden om een interactief project met Scratch te maken.

Geef de leerlingen de volgende startinstructies:

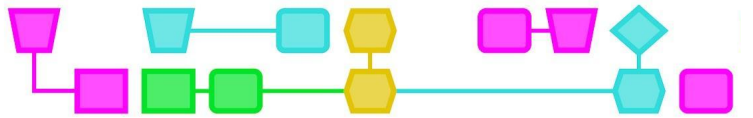
- Maak een karakter.
- Maak het interactief.
- Maak je karakter interactief door programma's toe te voegen die het karakter laten reageren op klikken, toetsaanslagen en andere dingen.
- Gebruik kostuums om het uiterlijk van het personage te veranderen.
- Maak verschillende achtergronden.
- Probeer geluiden toe te voegen aan je project.
- Probeer beweging toe te voegen aan je collage.

Conclusie (10 min)

Stel de leerlingen voor om de programma's en animaties die ze hebben ontworpen te laten zien aan anderen.



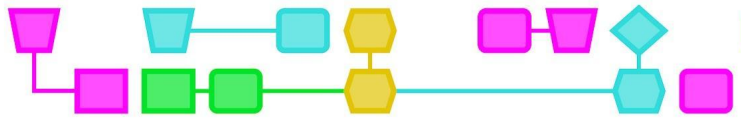
Teamwork



Werkblad - Uitdaging 1

Orden de volgende instructies voor het tekenen van een vierkant:

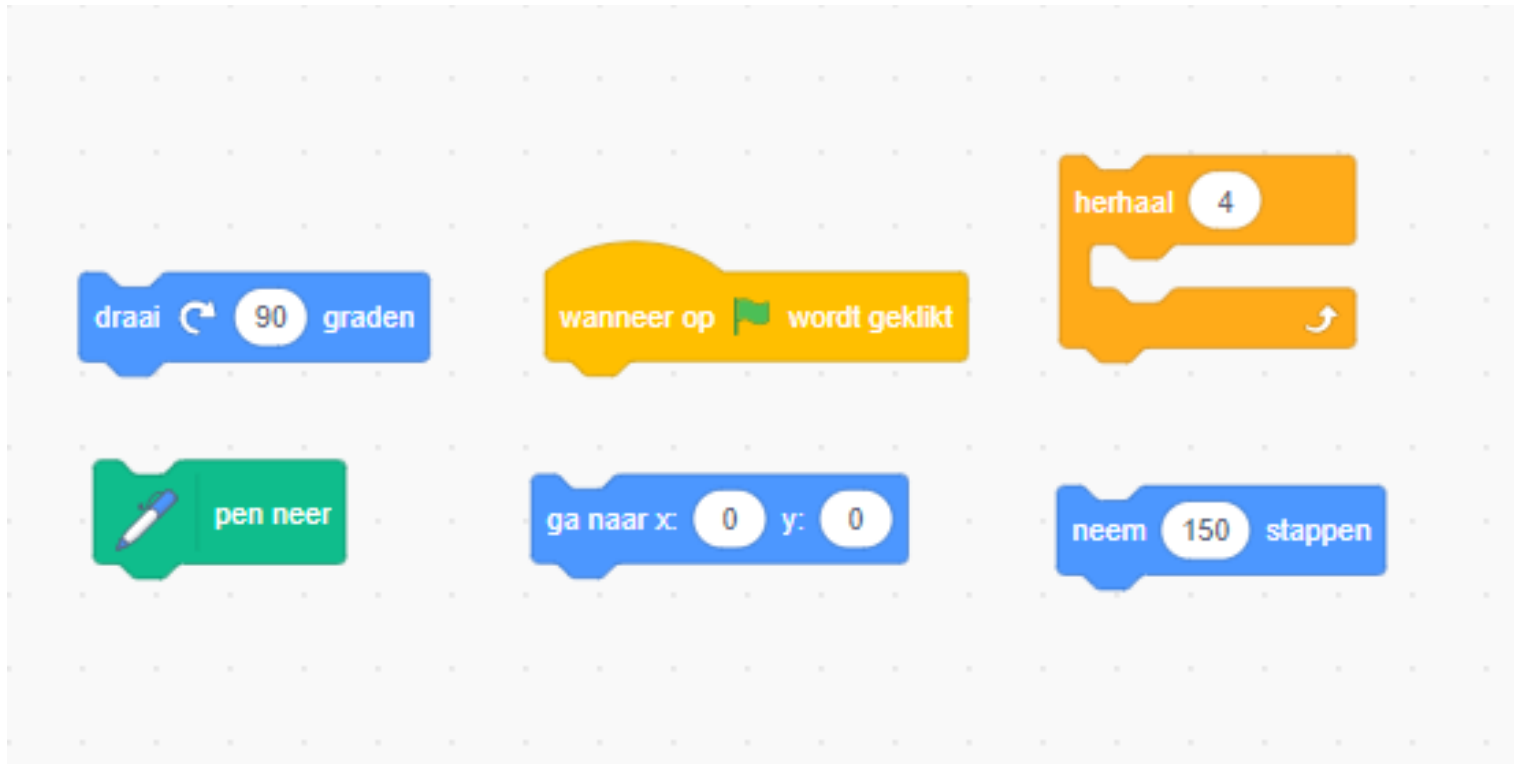
- Trek een rechte lijn naar rechts van 20 cm.
- Zet het schilderij op het bord.
- Draai 90 graden naar rechts.
- Trek een rechte lijn naar rechts van 20 cm.
- Trek een rechte lijn van 20 cm naar rechts.
- Draai 90 graden naar rechts.
- Trek een rechte lijn naar rechts van 20 cm.
- Ga in het midden van het bord staan.
- Pak een kwast/potlood.
- Draai 90 graden naar rechts.

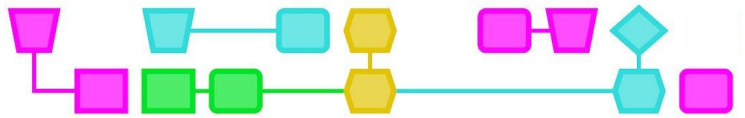


Werkblad - Uitdaging 2

Kun je een code schrijven om een vierkant te tekenen met Scratch met behulp van de volgende blokken?

Dat kan offline met de onderstaande blokken, of online (<https://scratch.mit.edu/>).





Werkblad - Uitdaging 3

Trek een lijn tussen de codeblokken en de juiste vorm:

```

    whenever op vlag wordt geklikt
    ga naar x: 0 y: 0
    pen neer
    herhaal 360
    neem 1 stappen
    draai 1 graden
  
```

```

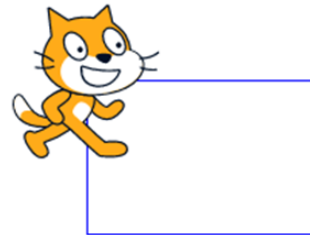
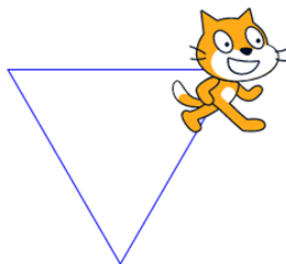
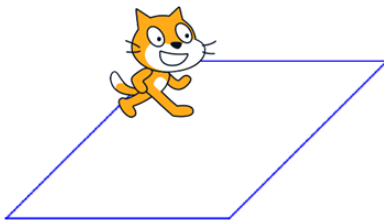
    whenever op vlag wordt geklikt
    ga naar x: 0 y: 0
    pen neer
    herhaal 4
    neem 200 stappen
    draai 135 graden
    neem 200 stappen
    draai 45 graden
  
```

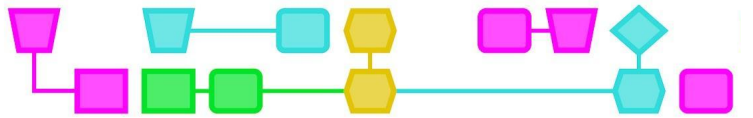
```

    whenever op vlag wordt geklikt
    ga naar x: 0 y: 0
    pen neer
    herhaal 2
    neem 150 stappen
    draai 90 graden
    neem 100 stappen
    draai 90 graden
  
```

```

    whenever op vlag wordt geklikt
    ga naar x: 0 y: 0
    pen neer
    herhaal 3
    draai 120 graden
    neem 180 stappen
  
```





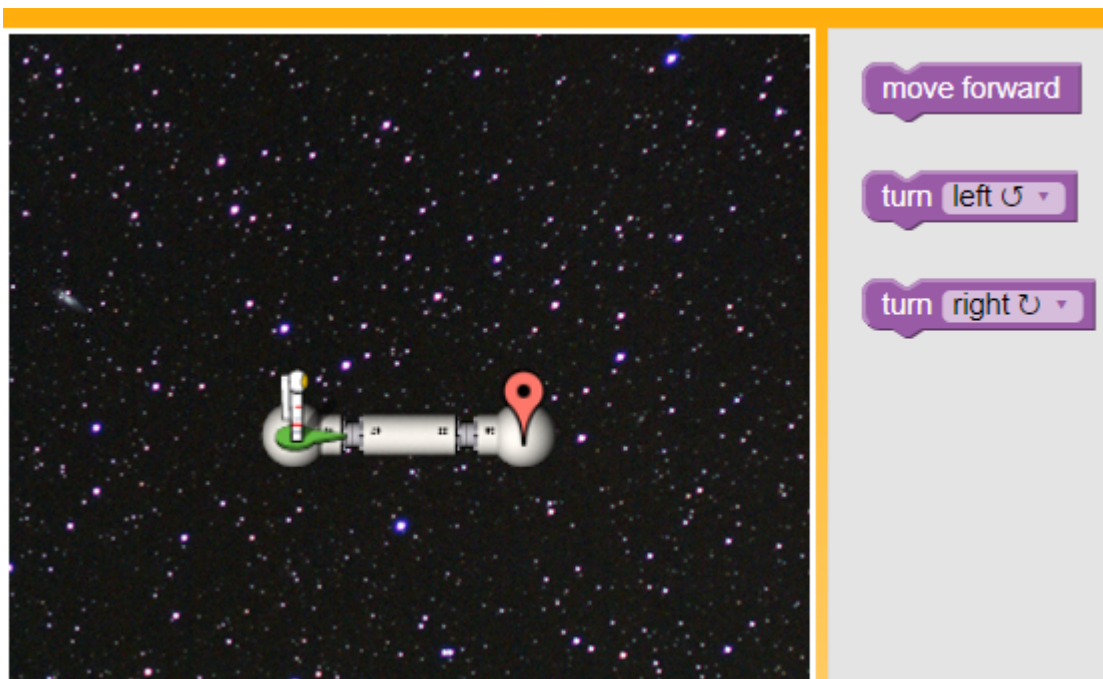
Werkblad - EXTRA

Help de astronaut de missie te volbrengen.

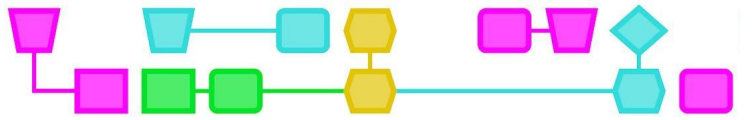
In de volgende uitdagingen ga je de astronaut helpen om de missie te voltooien.

Ga naar de volgende website: <http://kodetu.org/>.

Verzamel de blokken om de juiste instructies te geven aan de astronaut om het doel (het rode punt) te bereiken. Je mag alleen de blokken gebruiken die op het scherm verschijnen.



Missie 1 - Jouw antwoord:

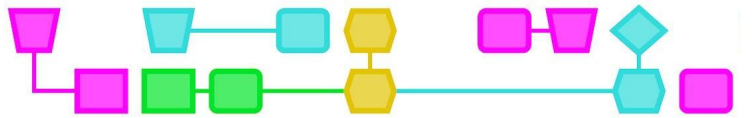


move forward

turn left 90

turn right 90

Missie 2 - Jouw antwoord:



Bijlage 3: oplossingen voor de uitdagingen

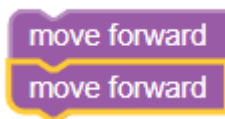
UITDAGING 1

Juiste volgorde van instructies:

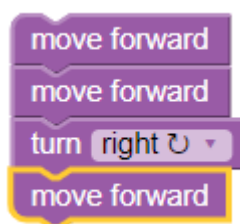
- Pak een schilderij op.
- Ga in het midden van het bord staan.
- Leg het schilderij op het bord.
- Trek een rechte lijn naar rechts van 20 cm.
- Draai 90 graden naar rechts.
- Trek een rechte lijn naar rechts van 20 cm.
- Draai 90 graden naar rechts.
- Trek een rechte lijn naar rechts van 20 cm.
- Draai 90 graden naar rechts.

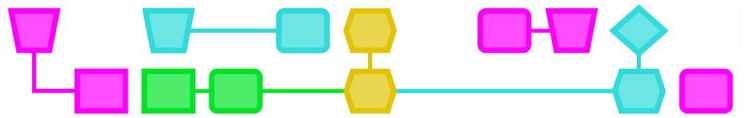
UITDAGING 2

Missie 1:



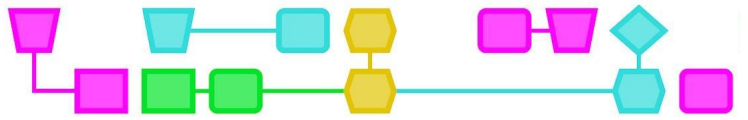
Missie 2:



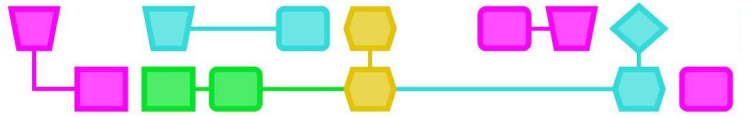


UITDAGING 3

| | |
|--|--|
| <pre>wanneer op vlag wordt geklikt ga naar x: 0 y: 0 pen neer herhaal 360 neem 1 stappen draai 1 graden</pre> | |
| <pre>wanneer op vlag wordt geklikt ga naar x: 0 y: 0 pen neer herhaal 4 neem 200 stappen draai 135 graden neem 200 stappen draai 45 graden</pre> | |



| | |
|---|--|
| <pre> wanneer op vlag wordt geklikt ga naar x: 0 y: 0 pen neer herhaal 3 draai 120 graden neem 180 stappen </pre> | |
| <pre> wanneer op vlag wordt geklikt ga naar x: 0 y: 0 pen neer herhaal 2 neem 150 stappen draai 90 graden neem 100 stappen draai 90 graden </pre> | |



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

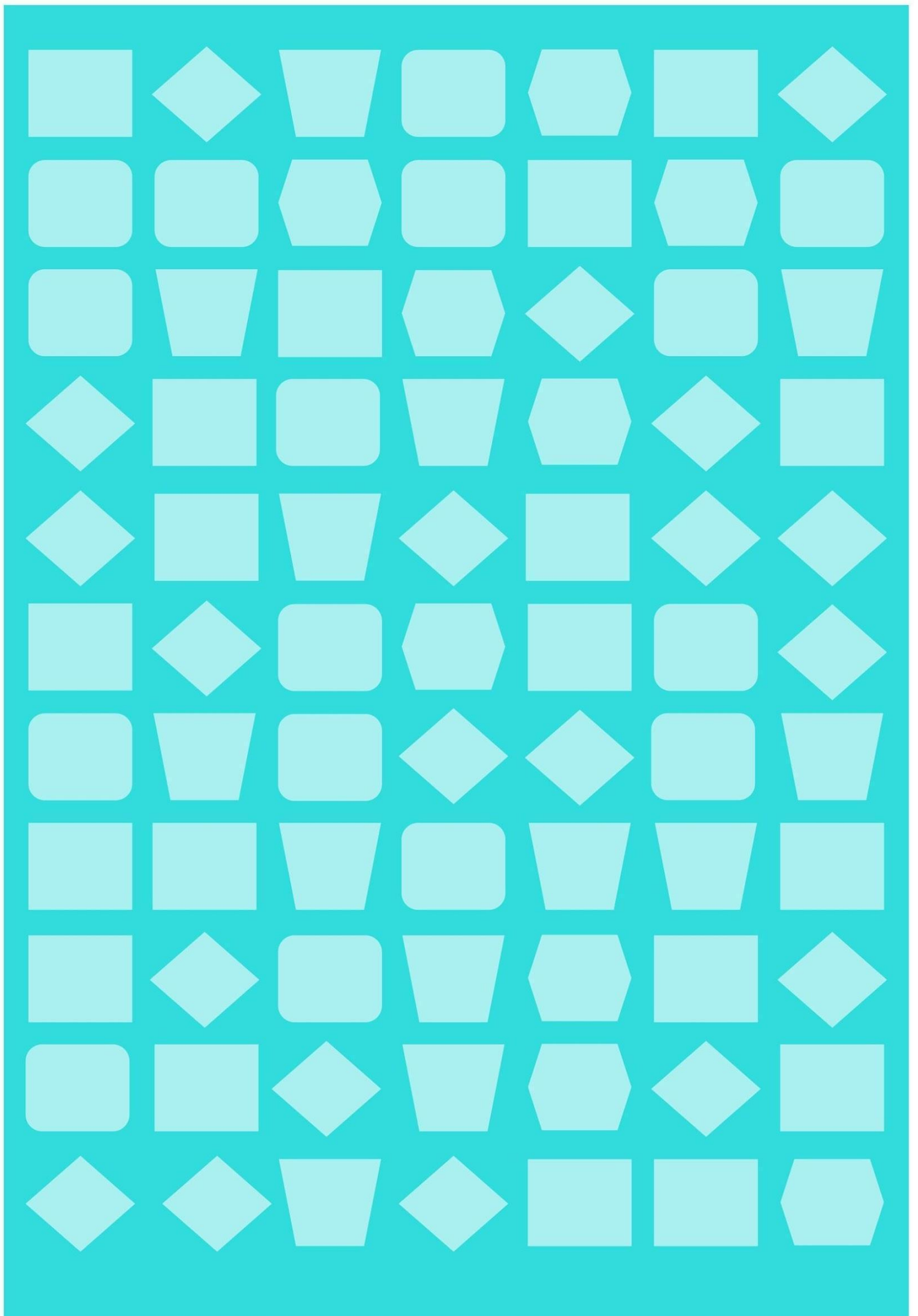


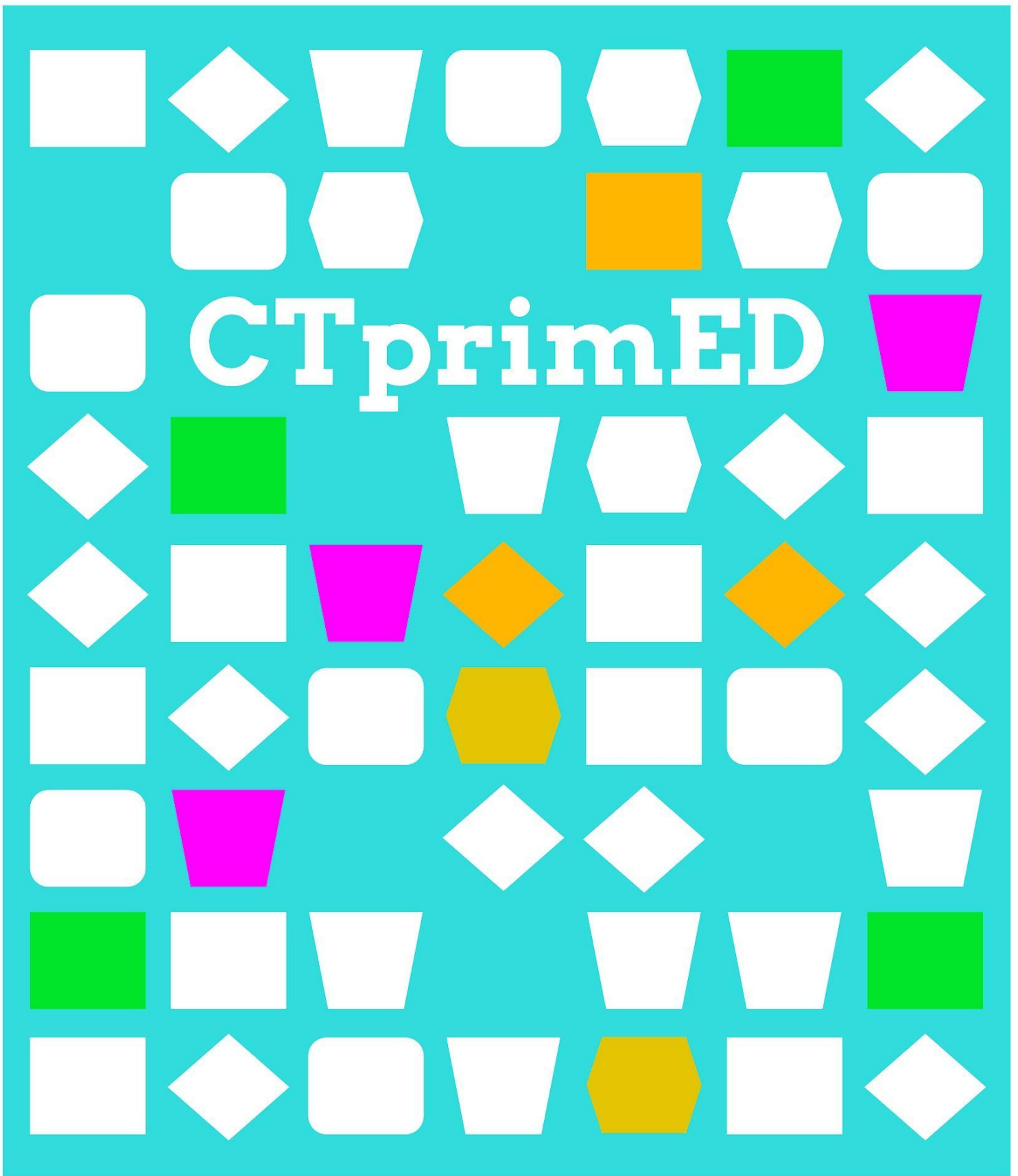
Co-funded by the
Erasmus+ Programme
of the European Union



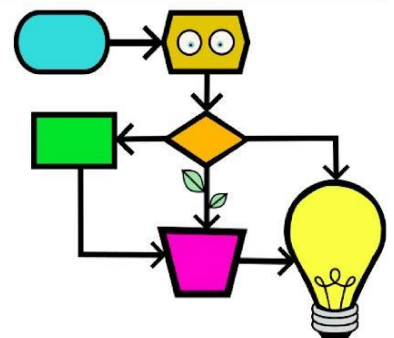
Universidad de Deusto
University of Deusto

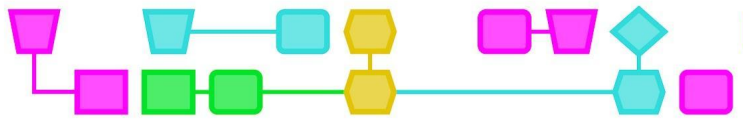
Deusto





Basisconcepten





Samenvatting

Bij deze activiteit worden de concepten algoritmes en binaire code geïntroduceerd. Je gebruikt spelkaarten om te begrijpen hoe pixels werken en om geheime boodschappen in binaire code te schrijven.

Doelgroep: Leerlingen vanaf 7 jaar met basiskennis van lezen, schrijven en rekenen.

Duur: Een sessie, maximaal 60 - 90 minuten.

Leerdoelen:

- Het binaire getallenstelsel leren kennen en gebruiken.
- De weergave van decimale getallen in het binaire getallenstelsel (binaire code) begrijpen.
- Algoritmen begrijpen, interpreteren en ontwerpen.
- De weergave van afbeeldingen door middel van pixels begrijpen.
- Afbeeldingen met pixels maken met behulp van de binaire code.

Online/offline: offline

Computational thinking:

- Algemene vaardigheden: samenwerking, creativiteit, logisch denken, algoritmisch denken.
- CT-foundations: algoritmes
- CT-concepten: binaire code, rekenkundige en logische bewerkingen, weergave van informatie in computers.

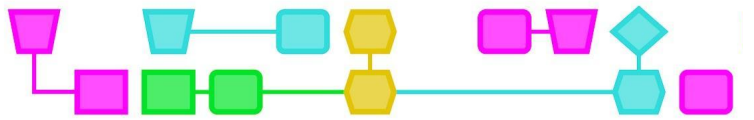
Materialen:

Per groepje:

- 10 kartonnen bekercs
- Een spel pokerkaarten per 4 leerlingen/groepje
- Papier
- Potlood
- Gum

Vorbereiding

- Verzamel de kaarten met de nummers 1, 2, 4 en 8 uit de stapel kaarten en maak vier sets. Alleen deze kaarten worden gebruikt in de activiteit. Elk groepje gebruikt een van de sets.
- Print de pixel- en geheime-boodschap-activiteitenbladen voor elke leerling.



Basisconcepten

Introductie (5 min)

Leg de leerlingen uit dat ze een aantal opdrachten gaan doen die te maken hebben met binaire code, beeldweergave en algoritmen.

Lesbeschrijving (45-60 min)

UITDAGING 1 - PROGRAMMEREN MET BEKERTJES

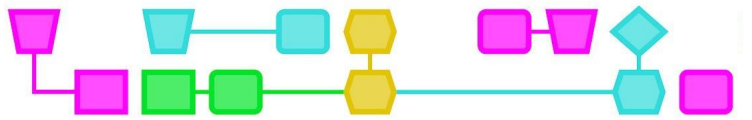
Een algoritme is een reeks instructies waarmee je een probleem op kunt lossen. Leg de leerlingen uit dat ze in tweetallen gaan werken, waarbij één leerling de programmeur is en instructies geeft aan de andere leerling. Deze leerling stelt de computer voor en verplaatst de bekertjes volgens de instructies van de programmeur.

De programmeur moet zorgen dat de computer de bekertjes opstapelt zoals in het onderstaande voorbeeld.



Dit zijn de instructies die de programmeur kan gebruiken:

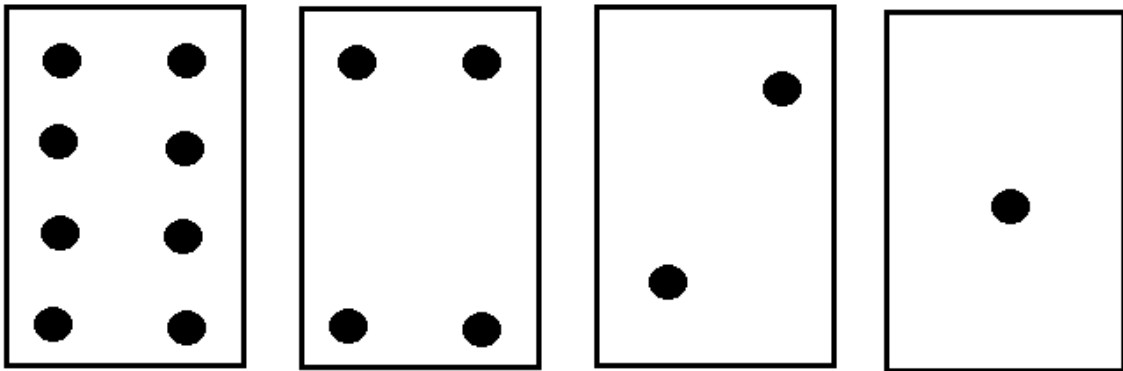
- ↑ Til de beker omhoog
- ↓ Laat de beker zakken
- Beweeg de beker 1/2 breedte naar voren
- ← Beweeg de beker 1/2 breedte naar achteren
- ⌚ Draai de beker 90 graden naar rechts
- ⌚ Draai de beker 90 graden naar links



In deze activiteit leren de leerlingen hoe belangrijk het is om nauwkeurig te zijn bij het geven van instructies, en in te zien hoe moeilijk het is om deze instructies op te volgen of er een algoritme mee te ontwerpen.

UITDAGING 2 - BINAIR TELLEN

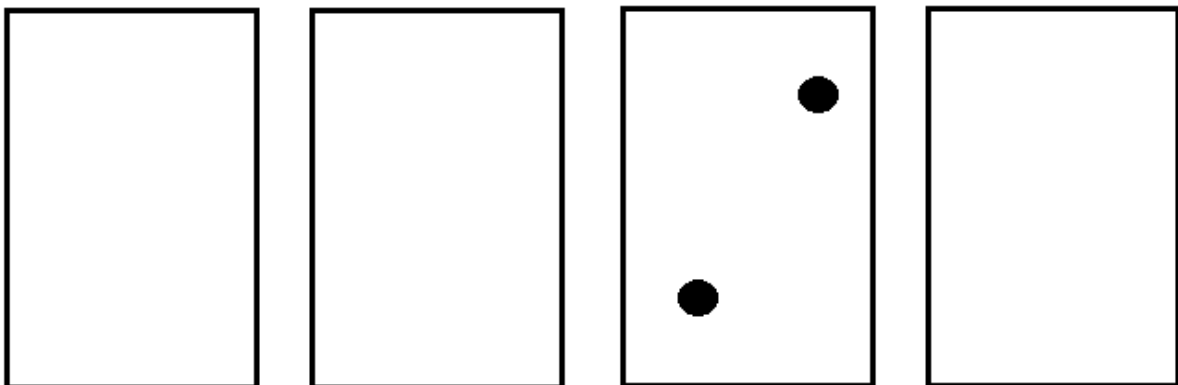
Vorbereiding: Maak of print onderstaande kaarten. De leerlingen gaan ze gebruiken om binair te tellen.



Deel de set kaarten uit aan de groepjes leerlingen. Vraag ze alle kaarten voor zich neer te leggen zoals in de bovenstaande afbeelding. Leg uit dat een gesloten kaart een 0 voorstelt en een open kaart een 1.

Vertel de leerlingen dat ze binair gaan tellen. Om de decimale waarde te achterhalen tellen ze de punten van de kaarten die open liggen bij elkaar op. Neem onderstaand voorbeeld met de leerlingen door.

Begin met: 0010. Volg samen de regels van de binaire code: 0010 betekent: kaarten 8-gesloten, 4-gesloten, 2-open, 0-gesloten. De decimale waarde is dan dus 2.



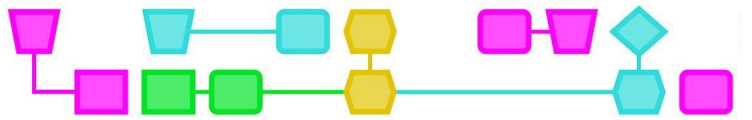
0 (Gesloten)

0 (Gesloten)

1 (Open)

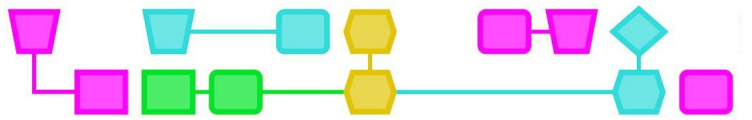
0 (Gesloten)

Probeer nu 0110. (Kaart 8 gesloten, kaart 4 open, kaart 2 open, kaart 1 gesloten.) Vraag de leerlingen de kaarten bij elkaar op te tellen die nu openliggen ($4 + 2 = 6$). Wat is de decimale waarde van 0110? De decimale waarde is 6. Concludeer: 0110 (binair) = 6 (decimaal).



Blijf oefenen met de leerlingen tot ze begrijpen hoe ze binair moeten tellen. Vraag de leerlingen vervolgens de binaire getallen van 0 tot 15 op te schrijven.

Binaire code is een taal gebaseerd op de getallen 1 en 0, waarvan de combinatie leidt tot een reeks instructies. Dit systeem wordt door computers gebruikt om te communiceren en alle informatie die zij opslaan te verwerken. Binaire code heeft vele toepassingen, zoals programmeren, gegevensoverdracht, digitale en elektronische communicatie.



UITDAGING 3 - EEN BERICHT CODEREN/DECODEREN

Nu de leerlingen kunnen tellen in binaire code, kunnen ze deze geheime boodschap oplossen.

Vertel de leerlingen dat ze een geheime boodschap moeten vinden. Deel werkblad 1 uit en leg de opdracht uit. Eerst schrijven ze het gelijkwaardige decimale getal naast het binaire getal. Dan gebruiken ze de tabel hieronder om te zien met welke letter het decimale getal overeenkomt en lezen ze de verborgen boodschap.

| Binair systeem | | | | | Decimaal systeem | Letter |
|----------------|---|---|---|---|------------------|--------|
| 16 | 8 | 4 | 2 | 1 | | |
| 0 | 0 | 0 | 1 | 1 | | |
| 1 | 0 | 0 | 1 | 0 | | |
| 0 | 0 | 1 | 0 | 1 | | |
| 0 | 0 | 1 | 1 | 0 | | |
| 0 | 1 | 0 | 1 | 0 | | |
| 1 | 0 | 1 | 1 | 0 | | |
| 0 | 1 | 1 | 0 | 1 | | |
| 0 | 0 | 1 | 1 | 0 | | |
| 1 | 1 | 0 | 0 | 0 | | |
| 0 | 1 | 1 | 0 | 0 | | |

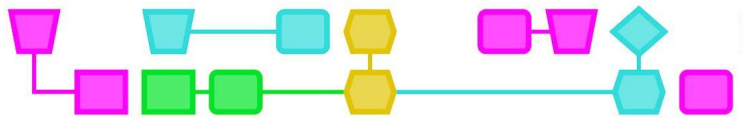
Decoderingstabel:

| | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| a | b | c | ch | d | e | f | g | h | i | j | k | l | ll | m |
| 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | |
| n | ñ | o | p | q | r | s | t | u | v | w | x | y | z | |

Extra:

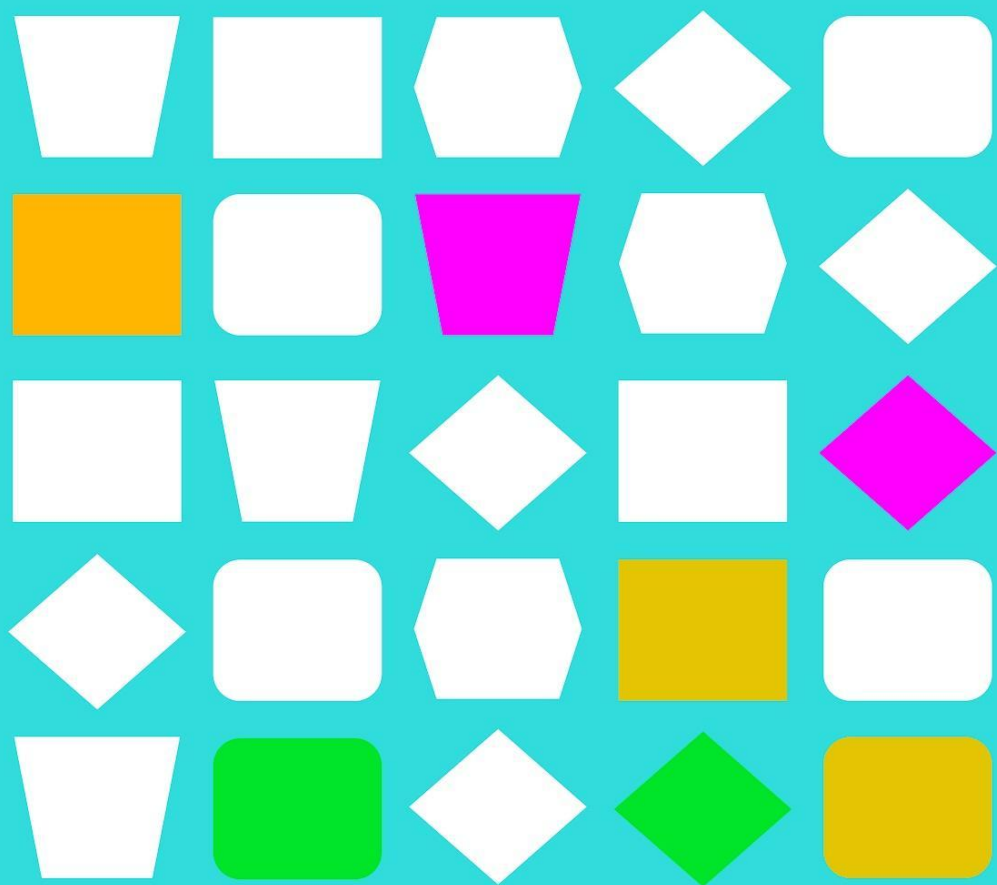
Vraag de leerlingen een geheime boodschap aan een andere leerling te schrijven. Daarvoor moeten ze het tegenovergestelde doen:

1. Schrijf het bericht in letters
2. Schrijf het decimale getal naast elke letter
3. Zet het decimale getal om in het bijbehorende binaire getal
4. Geef de code aan een andere leerling

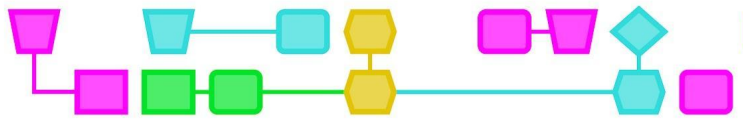


Conclusie (10 min)

Besprek met de leerlingen wat ze hebben geleerd van de activiteit. Herinner hen eraan waarom binaire code belangrijk is bij het weergeven en overbrengen van informatie, en vertel dat het belangrijk is heel nauwkeurig en precies te zijn bij het maken van instructies of algoritmen.



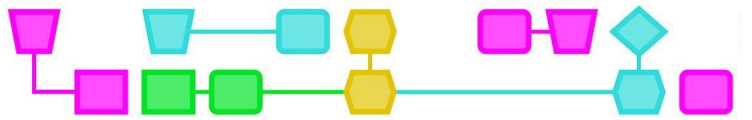
Bijlagen



Werkblad - Uitdaging 3

| Binair systeem | | | | | Decimaal systeem | Letter |
|----------------|---|---|---|---|------------------|--------|
| 16 | 8 | 4 | 2 | 1 | | |
| 0 | 0 | 0 | 1 | 1 | | |
| 1 | 0 | 0 | 1 | 0 | | |
| 0 | 0 | 1 | 0 | 1 | | |
| 0 | 0 | 1 | 1 | 0 | | |
| 0 | 0 | 1 | 1 | 1 | | |
| 1 | 0 | 0 | 1 | 0 | | |
| 1 | 0 | 1 | 0 | 1 | | |
| 0 | 0 | 1 | 1 | 1 | | |
| 1 | 1 | 0 | 0 | 0 | | |
| 1 | 0 | 0 | 0 | 0 | | |

| | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| a | b | c | ch | d | e | f | g | h | i | j | k | l | ll | m |
| 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | |
| n | ñ | o | p | q | r | s | t | u | v | w | x | y | z | |



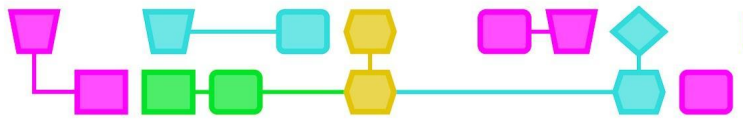
Bijlage: Oplossingen

UITDAGING 3

| Binair systeem | | | | | Decimaal systeem | Letter |
|----------------|---|---|---|---|------------------|--------|
| 16 | 8 | 4 | 2 | 1 | | |
| 0 | 0 | 0 | 1 | 1 | 3 | C |
| 1 | 0 | 0 | 1 | 0 | 18 | O |
| 0 | 0 | 1 | 0 | 1 | 5 | D |
| 0 | 0 | 1 | 1 | 0 | 6 | E |
| 0 | 0 | 1 | 1 | 1 | 10 | I |
| 1 | 0 | 1 | 1 | 0 | 22 | S |
| 0 | 1 | 1 | 0 | 1 | 13 | L |
| 0 | 0 | 1 | 1 | 0 | 6 | E |
| 1 | 1 | 0 | 0 | 0 | 24 | U |
| 0 | 1 | 1 | 1 | 0 | 12 | K |

CODE IS LEUK

| | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| a | b | c | ch | d | e | f | g | h | i | j | k | l | ll | m |
| 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | |
| n | ñ | o | p | q | r | s | t | u | v | w | x | y | z | |



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

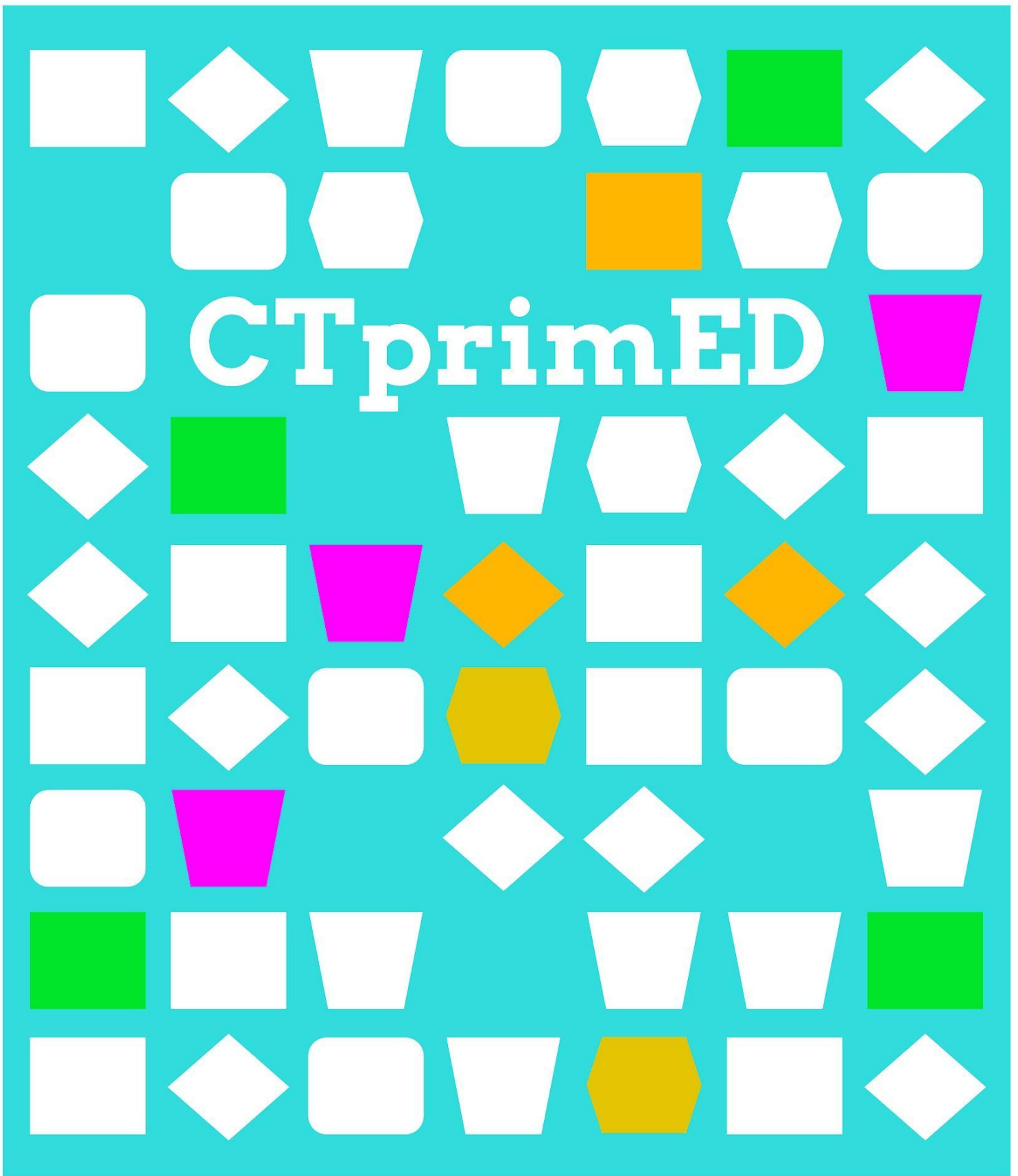


Co-funded by the
Erasmus+ Programme
of the European Union

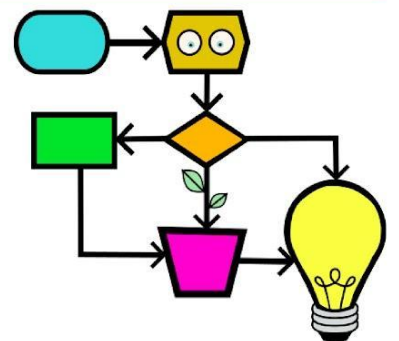


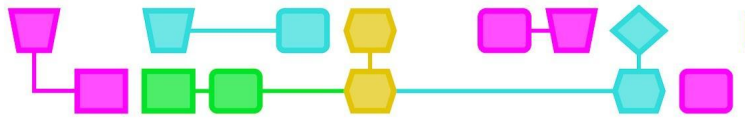
Universidad de Deusto
University of Deusto

Deusto



Roadrunner en Coyote





Samenvatting

De leerlingen spelen een spel waarin ze met behulp van vooraf bepaalde instructies door een doolhof moeten bewegen. Er zijn twee groepen: de Road Runner-groep, die moet proberen het einde van de doolhof te bereiken zonder gepakt te worden, en de Coyote-groep, die probeert de Roadrunner te 'pakken' door zijn pad te kruisen. De bewegingen in de doolhof worden opgeschreven voordat het spel begint.

Doelgroep: 9-12 jaar.

Duur: 60 minuten

Leerdoelen:

De leerlingen leren

- heel precies te zijn met instructies;
- hoe je voorwaarden kan gebruiken om door de doolhof te bewegen;
- wat programmeren is.

Online/offline: offline

Computational thinking:

- Algemene vaardigheden: plannen, samenwerking, problemen oplossen
- CT-foundations: algoritme, patroonherkenning, decompositie, abstractie
- CT-concepten: programmeren, functies, loops, codes, voorwaarden

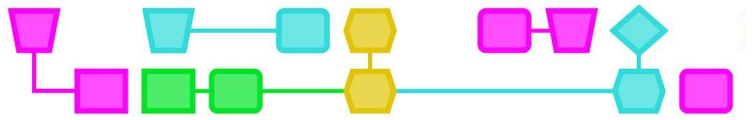
Materialen:

- Groot papier (flip-over of A3) of het voorbeeldrooster, zie bijlage
- Figuurtes die de Coyote en de Road Runner voorstellen (bijv. van lego)
- Tokens (bijv. muntjes, gekleurde vormen, etc. (zie voorbeeld)
- Dobbelstenen

Tip: Gebruik een flip-overvel met een rooster als basis voor het doolhof.

Vorbereiding

- Maak voldoende doolhoven voor de klas.
- Verzamel de benodigde materialen.



Roadrunner en Coyote

Introductie Roadrunner en Coyote (15 min)

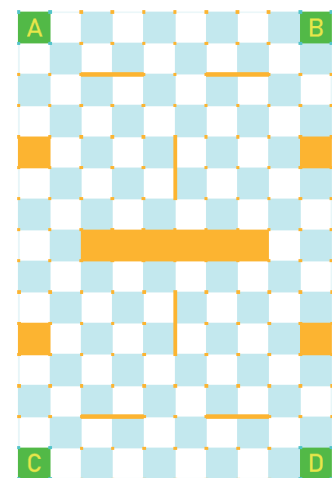
In deze introductieactiviteit zijn de leerlingen Roadrunner of Coyote. Net als in de tekenfilm is het hoofddoel van Coyote het vangen van Roadrunner, terwijl Roadrunner het einde van de doolhof probeert te bereiken zonder gepakt te worden. De bewegingen die de Roadrunner en Coyote maken in de doolhof worden van tevoren bepaald.

De leerlingen worden verdeeld in groepjes van vier. Ze vormen een duo en plannen samen welke bewegingen ze willen gaan maken op de doolhof. Er zijn dus twee duo's die in één doolhof spelen.

Zie het voorbeeld van de doolhof in Bijlage 2.

Begin- en eindpunt van elke groep

- De Roadrunnergroep start op punt A en probeert punt C te bereiken.
- De Coyote-groep start op punt C en probeert de Roadrunnergroep te vangen.



Bewegen door de doolhof (15 min)

Elk groepje schrijft met behulp van pijlen voor vooruit, achteruit, links, rechts ($\uparrow \leftarrow \rightarrow \downarrow$) welke weg ze door de doolhof gaan afleggen. Laat de leerlingen een route van 30 stappen uitschrijven, waarbij elk pijltje gelijkstaat aan 1 blokje. Als iedereen de route heeft uitgeschreven kan het spel beginnen.

Tip: Print het doolhof voor elk groepje een keer extra, dan kunnen ze de vooraf bedachte route op de doolhof intekenen.

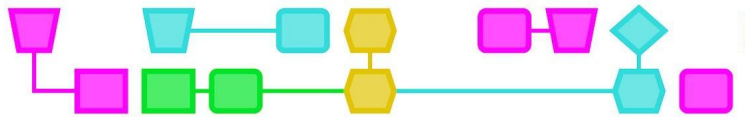
Om te bepalen wie met de eerste stap begint, gooien ze met de dobbelsteen. De groep met de hoogste worp gaat eerst en de groep met de een-na-hoogste worp gaat als tweede.

Het spel spelen (20 min)

De leerlingen lopen door de doolhof en leggen daarbij hun eigen route af. Bij elke beurt mag je een blokjes door het rooster bewegen. Herhaal de opdracht een paar keer zodat ze hun strategie kunnen verbeteren.

Afsluiten en evalueren (10 min)

Bespreek met de leerlingen hoe het ging. Leg uit dat ze een programma (algoritme) hebben geschreven om Coyote en Roadrunner door de doolhof te bewegen. Zouden ze het programma ook korter kunnen opschrijven met behulp van een loop of voorwaarde?



Het Doolhof

Maak je eigen doolhofspel (50 min)

De leerlingen gaan nu in groepjes van 4-8 hun eigen spel bedenken. De basis voor het spel is de doolhof of het open raster.

Stap 1: Doel van het spel

De leerlingen bedenken samen wat het doel van het spel wordt. Leg uit dat ze het bestaande raster mogen gebruiken of zelf een raster in het lege voorbeeld mogen maken.

De leerlingen kunnen bijv. een spel bedenken waarin ze

- elkaar moeten vangen;
- als eerste een bepaald punt moeten bereiken;
- de meeste munten of punten moeten verzamelen;
- zo goed mogelijk moeten samenwerken;
- elkaar moeten tegenwerken.

Stap 2: De doolhof

Het spelbord is de doolhof. In deze stap bedenken de leerlingen wat er past bij hun spelidee.

Bijvoorbeeld:

- Een leeg raster
- Obstakels tekenen in de doolhof
- Dezelfde obstakels of obstakels die juist verschillen aan elke kant van de doolhof
- Vakjes met een bepaalde functie, zoals bij ganzenborden

Stap 3: Regels

Bedenk hoe het spel moet worden uitgevoerd.

- Gooi een dobbelsteen voor het startpunt.
- Bij elke beurt loop je twee stappen.
- Het aantal stappen minimaliseren. Bijvoorbeeld: je mag maximaal 20 stappen zetten en moet zoveel mogelijk muntjes verzamelen.
- Probeer loops en voorwaarden te gebruiken. Bijvoorbeeld: 'Ga naar links of rechts als je bij een muur komt.'

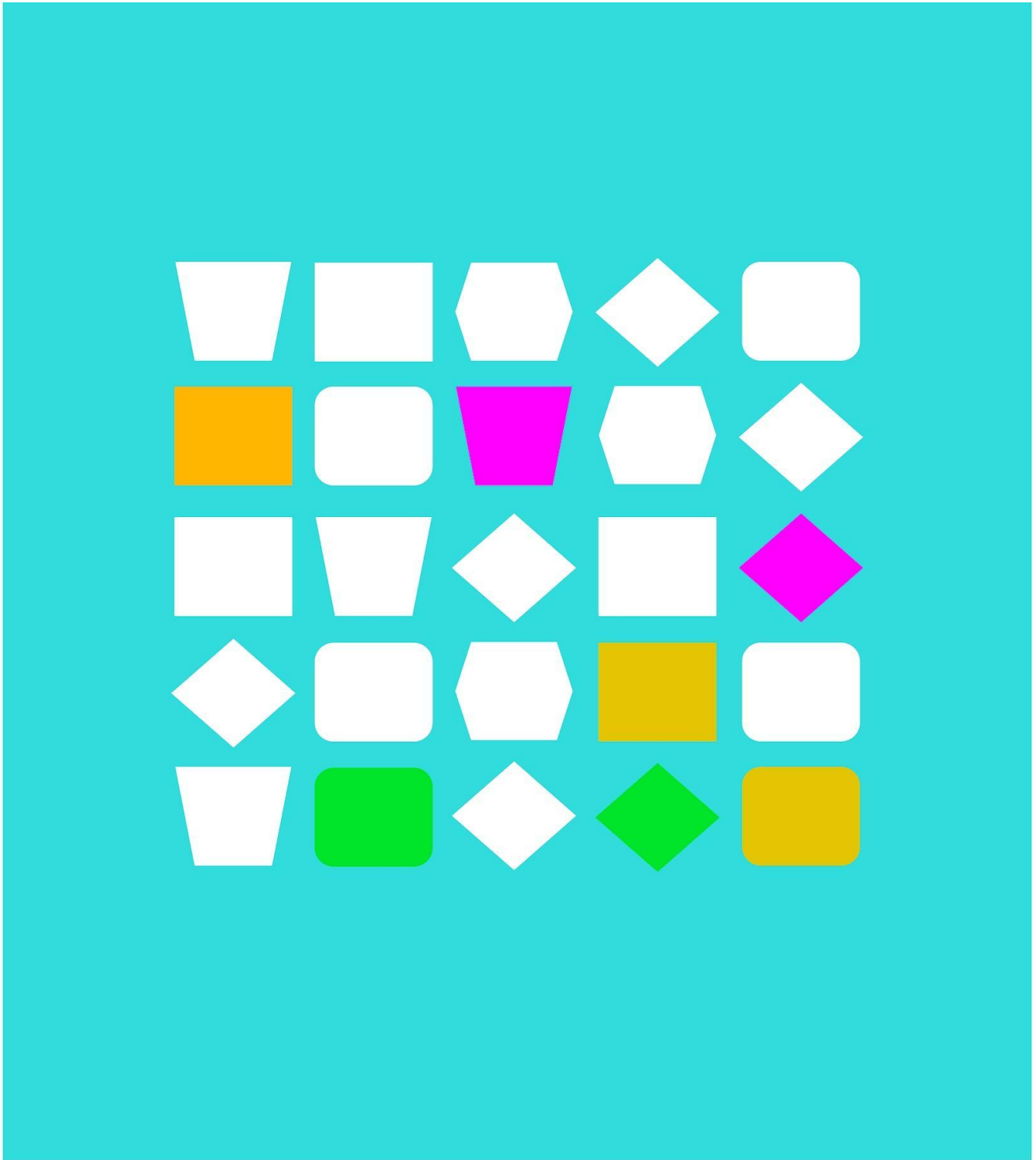
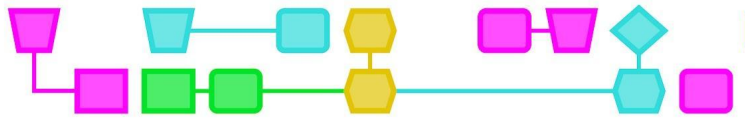
Stap 4: Uitproberen en aanpassen

Leerlingen schrijven de regels op en proberen het spel uit. Tijdens het uitproberen komen ze dingen tegen die niet of juist wel goed werken. Moedig de leerlingen aan om de regels aan te passen totdat het spel goed werkt.

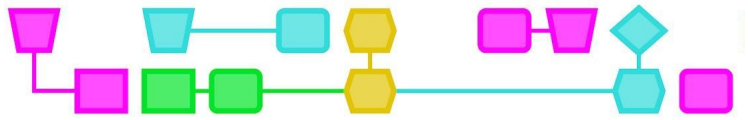
Conclusie (15 min)

Wissel de spelletjes uit tussen de leerlingen en schrijf verbeterpunten op. Bespreek met de leerlingen hoe ze het vonden en wat ze hebben geleerd.

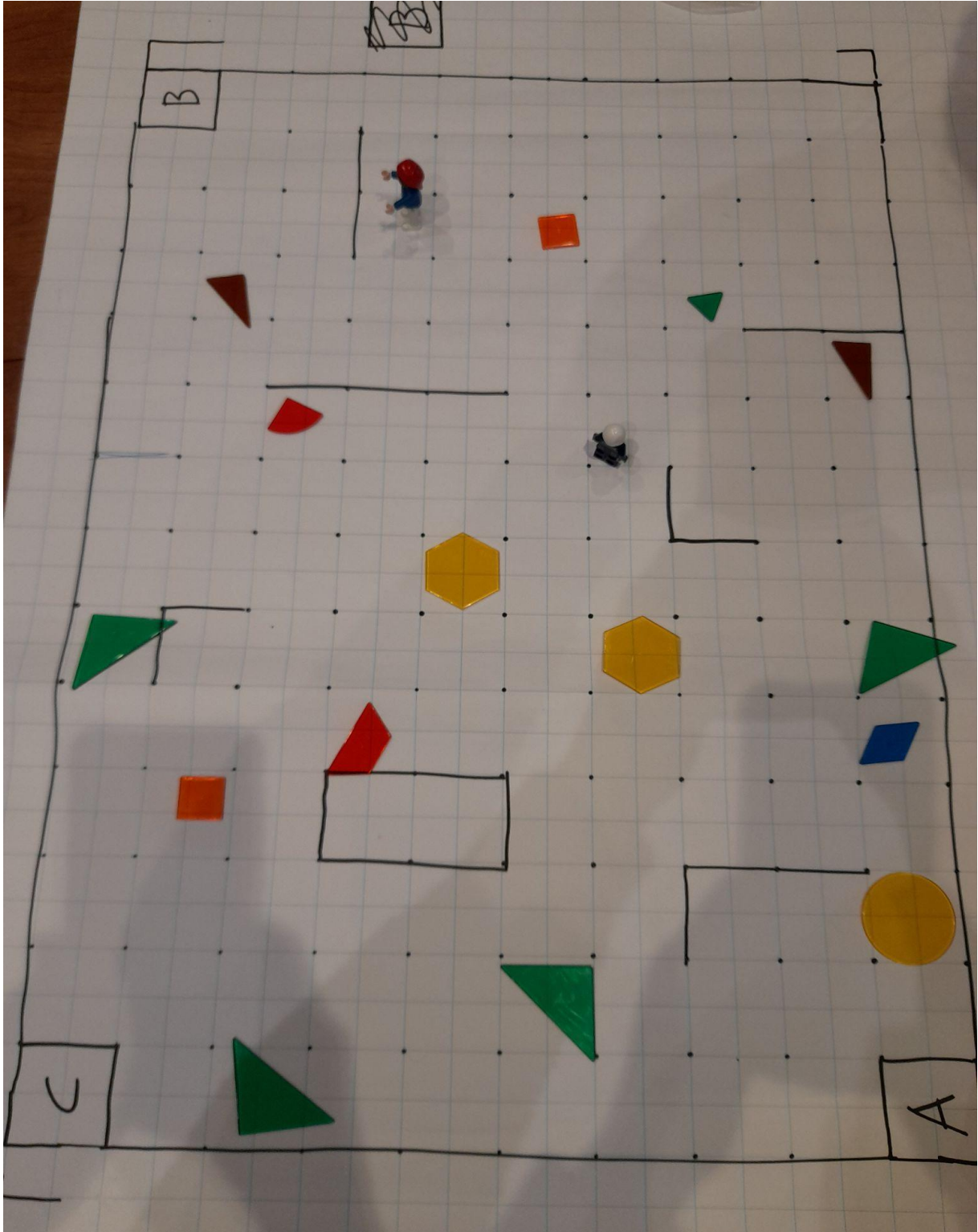
Leg het verband met programmeren uit: je moet heel precies zijn in je commando's, anders voert de computer je commando niet uit zoals je bedoeling was.

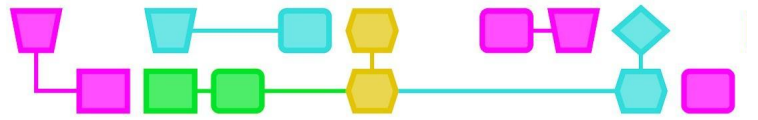


Bijlagen



Bijlage: Voorbeeld van een doolhof





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

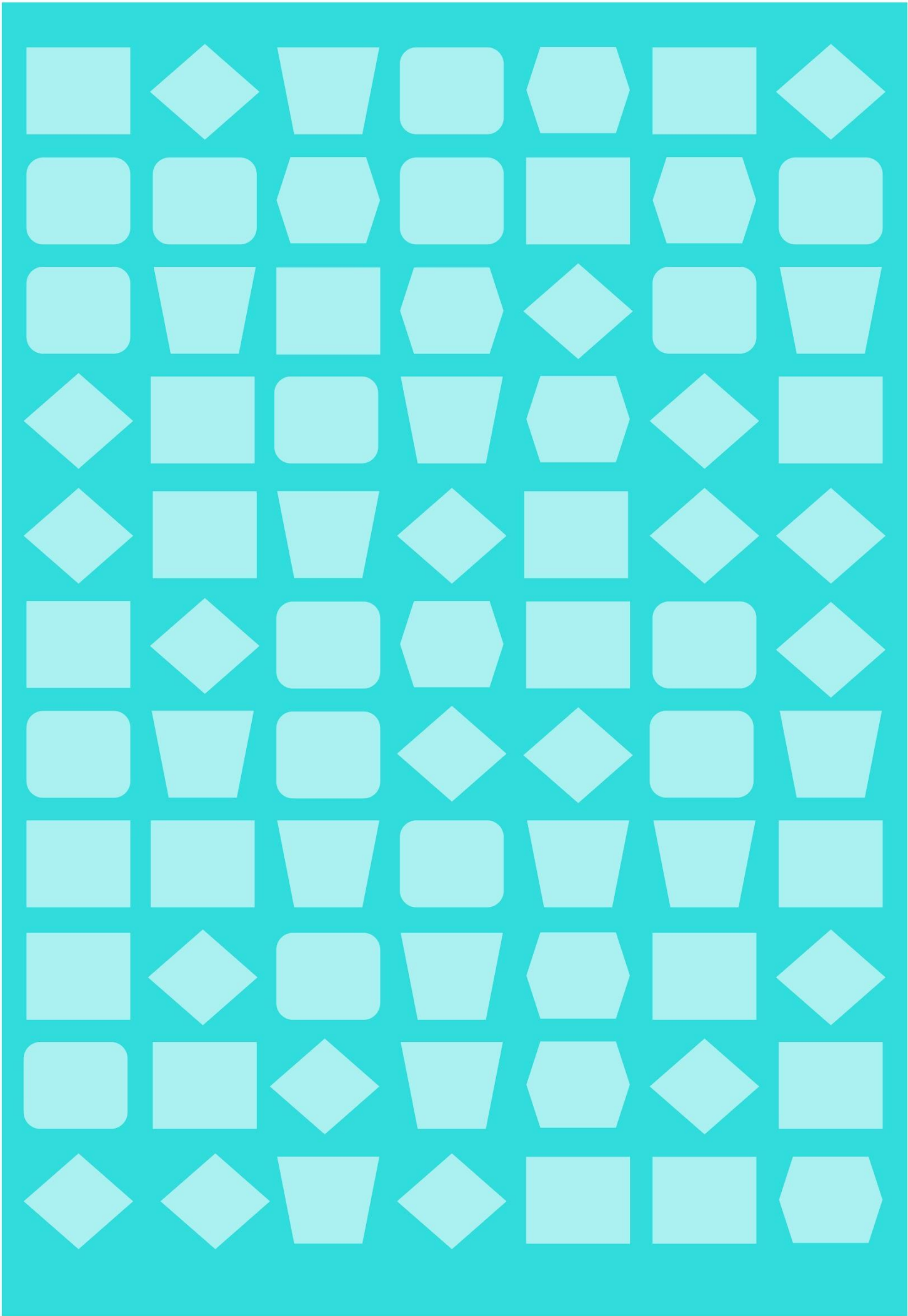


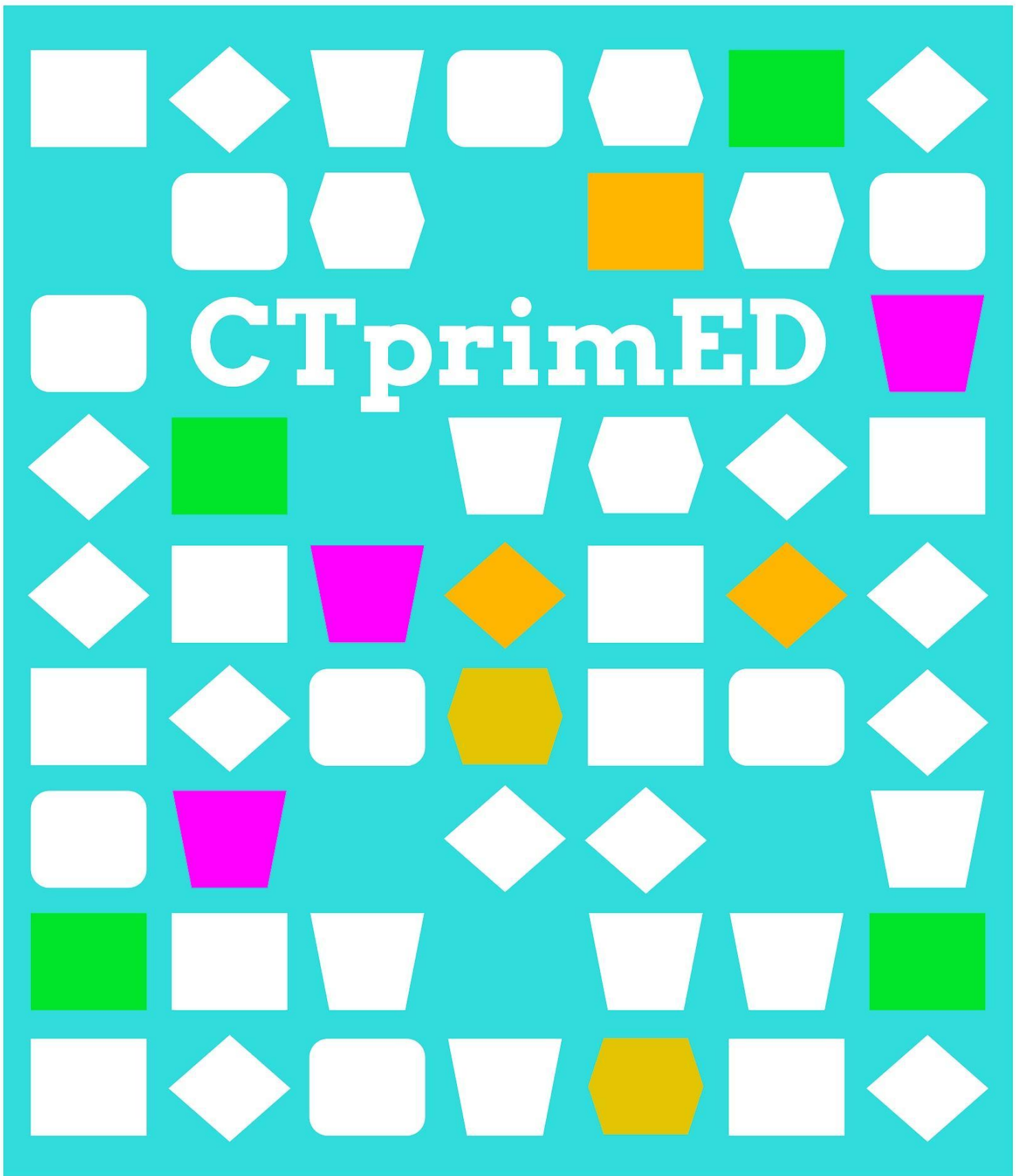
Co-funded by the
Erasmus+ Programme
of the European Union



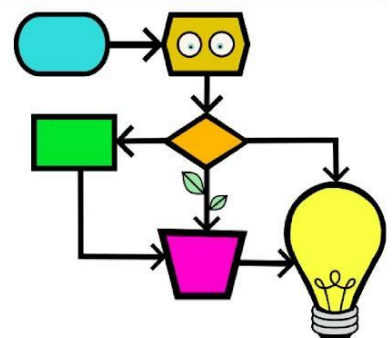
Universidad de Deusto
University of Deusto

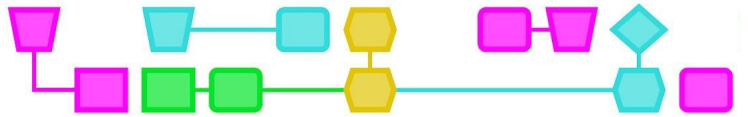
Deusto





Tinkeren met CT -
Zeg het met karton





Samenvatting

De leerlingen maken een kunstwerk waarmee ze een boodschap overbrengen. De leerlingen bepalen welke boodschap ze willen overbrengen; bijvoorbeeld een emotie zoals blijheid of woede, een waarschuwing, of iets totaal anders. Met deze activiteit kunnen leerlingen op een laagdrempelige manier leren programmeren door een micro:bit als deel van het kunstwerk te gebruiken. Een micro:bit is een kleine computer met een led-display en twee knoppen. De micro:bit kan worden geprogrammeerd met Scratch of MakeCode.

Doelgroep: 6-12 jaar (leerlingen moeten weten hoe ze MakeCode moeten gebruiken).

Duur: 60-120 minuten (voor de online versie: afhankelijk van de vraag of de leerlingen micro:bit en MakeCode kunnen gebruiken)

Leerdoelen:

- Op een creatieve manier een boodschap overbrengen aan anderen.
- Het doorlopen van stappen om een idee om te zetten in een product. Maken, uitproberen, en ontdekken staan centraal.

Online/offline: activiteit kan on- en offline gedaan worden. In de lesbeschrijving vind je tips om de les offline te doen.

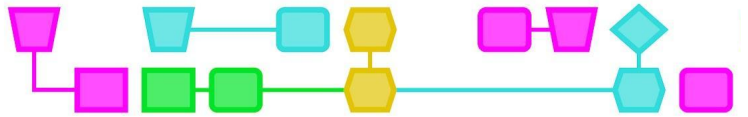
Computational thinking:

- Algemene vaardigheden: creatief denken, kritisch denken, samenwerken, plannen
- CT-foundations: algoritme, decompositie, abstractie
- CT-concepten: programma, functie, loop, code

Bijzonderheden: Voor de online versie is het handig als de leerlingen de micro:bit en MakeCode al kennen. Zie de extra micro:bit-instructie voor een kennismaking met de micro:bit.

Materialen:

- Karton
- Papier
- Satéprikkers of ijslollystokjes
- Papieren tape
- Lijm
- Schaar
- Paperclip
- Viltstiften
- Splitpennen
- Dikke of dunne elastiekjes
- Een bal touw
- Gerecyclede kartonnen dozen
- Knutselmaterialen naar keuze
- Ledlampen, 1,6-2,4 V
- Krokodillenklemmen of stroomdraad

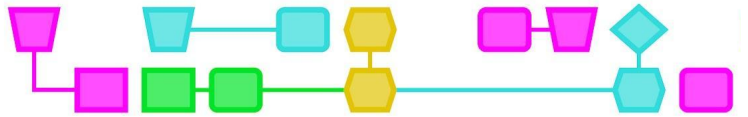


Alleen online:

- Computer met usb-output en internetverbinding
- Micro:bit (+ snoer en batterij)

Vorbereitung

Zorg ervoor dat er genoeg tafels en stoelen in het lokaal zijn. Zet minimaal twee materiaaltafels op verschillende plekken in de ruimte zodat de leerlingen door het lokaal moeten lopen om de materialen te verzamelen. Plaats (wanneer mogelijk) meerdere tweetallen aan elke tafel zodat de leerlingen zien wat anderen maken en dit met elkaar kunnen bespreken. Maak een voorbeeld (zie de foto's in de bijlage).



Zeg het met karton

Introductie (5 min)

Laat zien welke materialen er zijn en leg uit wat de leerlingen gaan maken. De opdracht is:

- Offline: maak een (kunst)werk dat een boodschap overbrengt, gebruik een lampje en de andere beschikbare materialen.
- Online: maak een (kunst)werk dat een boodschap overbrengt, gebruik een micro:bit en de andere beschikbare materialen.

Leg uit dat de studenten zelf mogen kiezen welke boodschap ze willen overbrengen, bijvoorbeeld een emotie zoals blijheid of woede, een waarschuwing, of iets totaal anders. Laat bedenken wie ze een boodschap willen sturen. Het kan bijvoorbeeld een politieke, grappige of persoonlijke boodschap zijn, bedoeld voor social media of voor de buurman aan de overkant van de straat.

Moedig de leerlingen aan een micro:bit in hun kunstwerk te gebruiken. Afhankelijk van de groep en hun kennis van de micro:bit kun je ervoor kiezen om een korte instructie over de micro:bit te geven (zie bijlage).

Laat een aantal afwisselende voorbeelden zien. Hierna kunnen de leerlingen aan de slag.

Mogelijke aanvullingen:

- Bespreek wat het woord 'boodschap' betekent en wat voor soort boodschappen je kunt overbrengen. Voor niet-moedertaalsprekers is dit een heel leuke toevoeging.
- Laat een aantal voorbeelden zien van tekens die iedereen kent om uit te leggen wat een boodschap kan zijn. Denk bijvoorbeeld aan een stopbord, of je vinger op je lippen leggen om aan te geven dat iemand stil moet zijn.

Beschrijving van de les (40 min)

De begeleider loopt door de ruimte, observeert en vraagt de leerlingen wat ze aan het doen zijn.

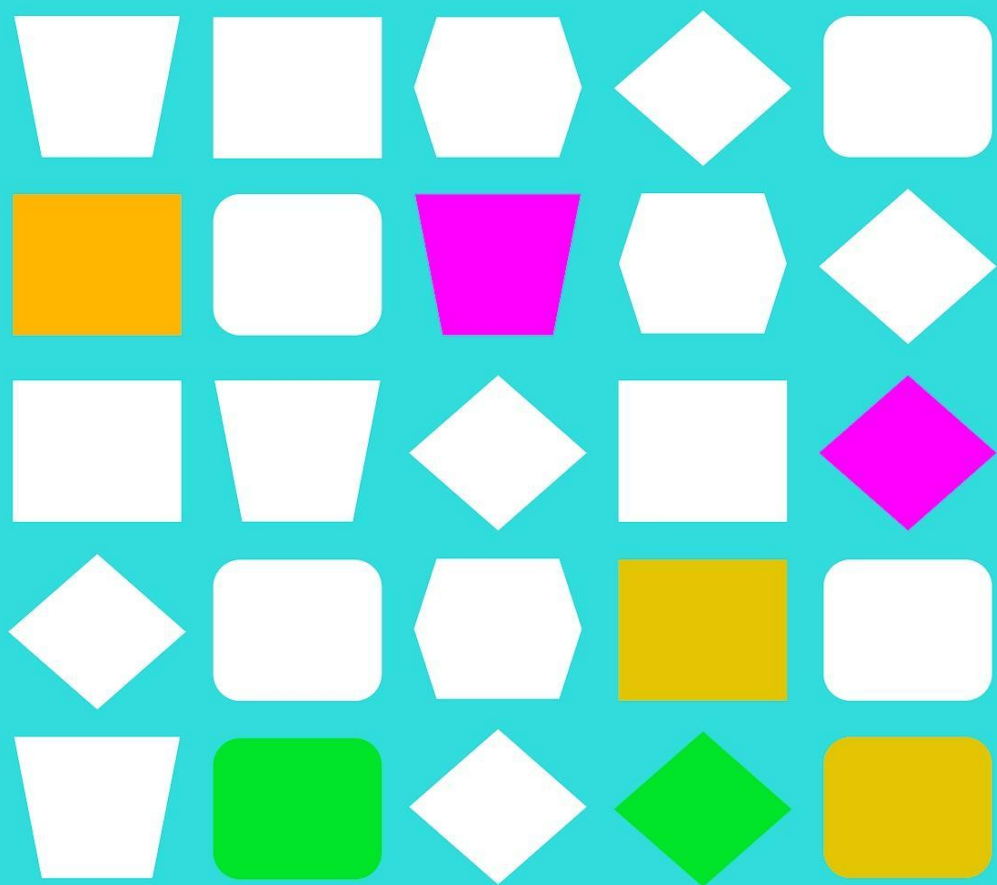
Moedig leerlingen aan om verschillende dingen uit te proberen met de micro:bit. Bijvoorbeeld:

- Daag ze uit om een icoon op het led-display te laten zien
- Daag ze uit om een reactie te programmeren wanneer knop A wordt ingedrukt
- Daag ze uit om licht te gebruiken in hun werk

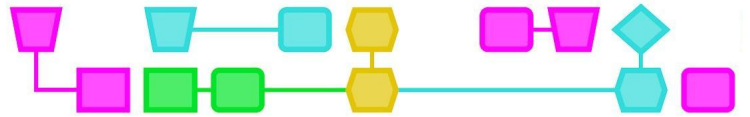
Laat het de leerlingen weten als de tijd om te bouwen bijna voorbij is.

Conclusie (10 min)

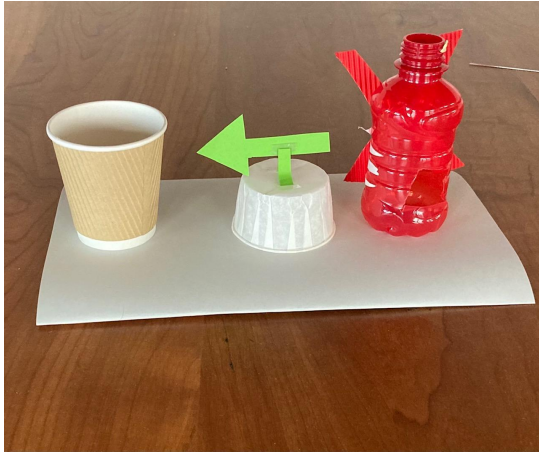
De leerlingen laten de groep zien wat ze hebben gemaakt en vertellen hoe ze dat hebben aangepakt. De begeleider deelt per groepje wat hij of zij heeft gezien. De leerlingen kunnen bijvoorbeeld vertellen over de wetenschappelijke of technische problemen die ze tegenkwamen tijdens het maken en hoe ze deze problemen hebben opgelost. Of ze bepaalde materialen op een creatieve manier hebben gebruikt. Hoe de samenwerking verliep en of ze doorzettingsvermogen nodig hadden.



Bijlagen



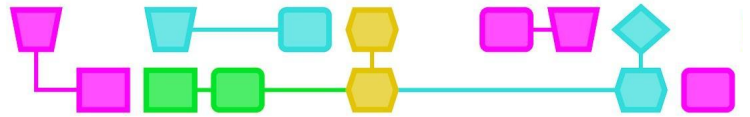
Bijlage 1: Voorbeelden



Boodschap: gebruik geen plastic maar karton!



Boodschap: wees lief voor elkaar! (Twee flessen die naar elkaar toe bewegen voor een knuffel.)



Colofon

Dit document is eerder ontwikkeld in het kader van het Project Tinkering EUROPE 2, mede mogelijk gemaakt door Erasmus+.

© 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

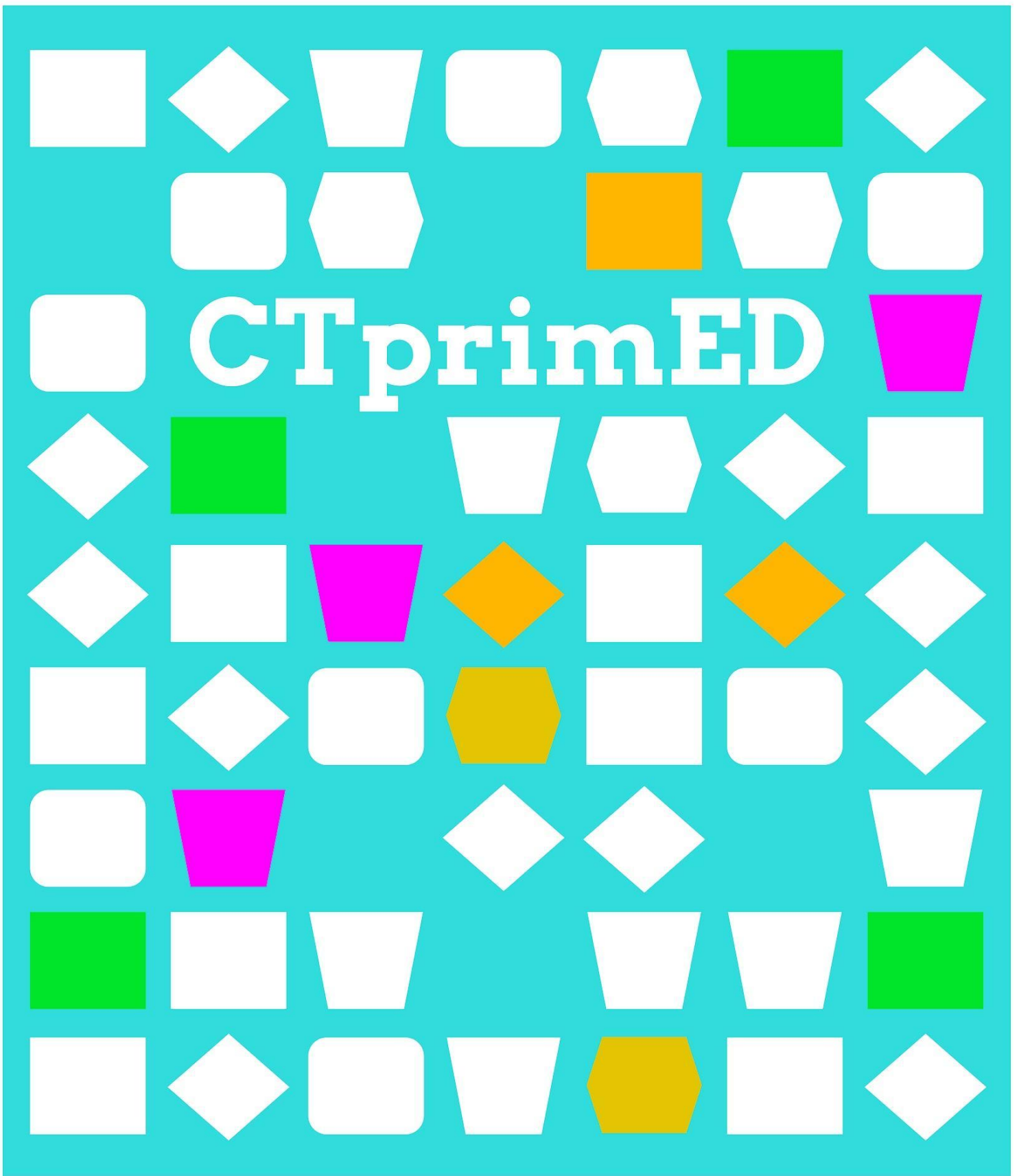


Co-funded by the
Erasmus+ Programme
of the European Union

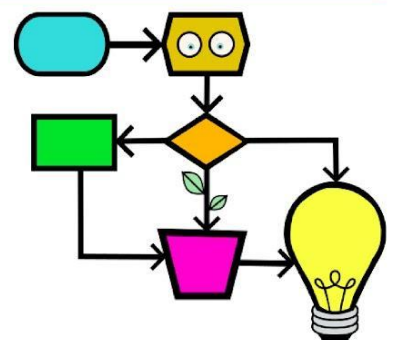


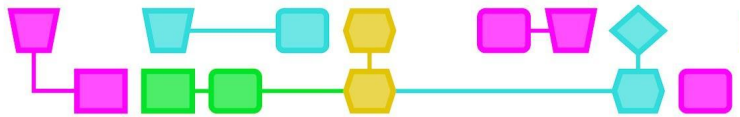
Universidad de Deusto
University of Deusto

Deusto



Maak X





Samenvatting

Met dit lesformat kun je een *computational thinking*-les (CT-les) maken die aansluit op het curriculum. Denk bijvoorbeeld aan de eindpresentatie van een project, een samenvatting van een hoofdstuk of een losstaand project of opdracht over een bepaald onderwerp. In deze les staat een format, verschillende ideeën en tips om zelf een CT-les te bedenken.

Over Maak X

Met Maak X kun je extra aandacht besteden aan CT door dit te integreren in al een bestaand programma. Het format vind je op bladzijde 3. Hieronder volgen een aantal voorbeelden van hoe je de X van Maak X zou kunnen inrichten:

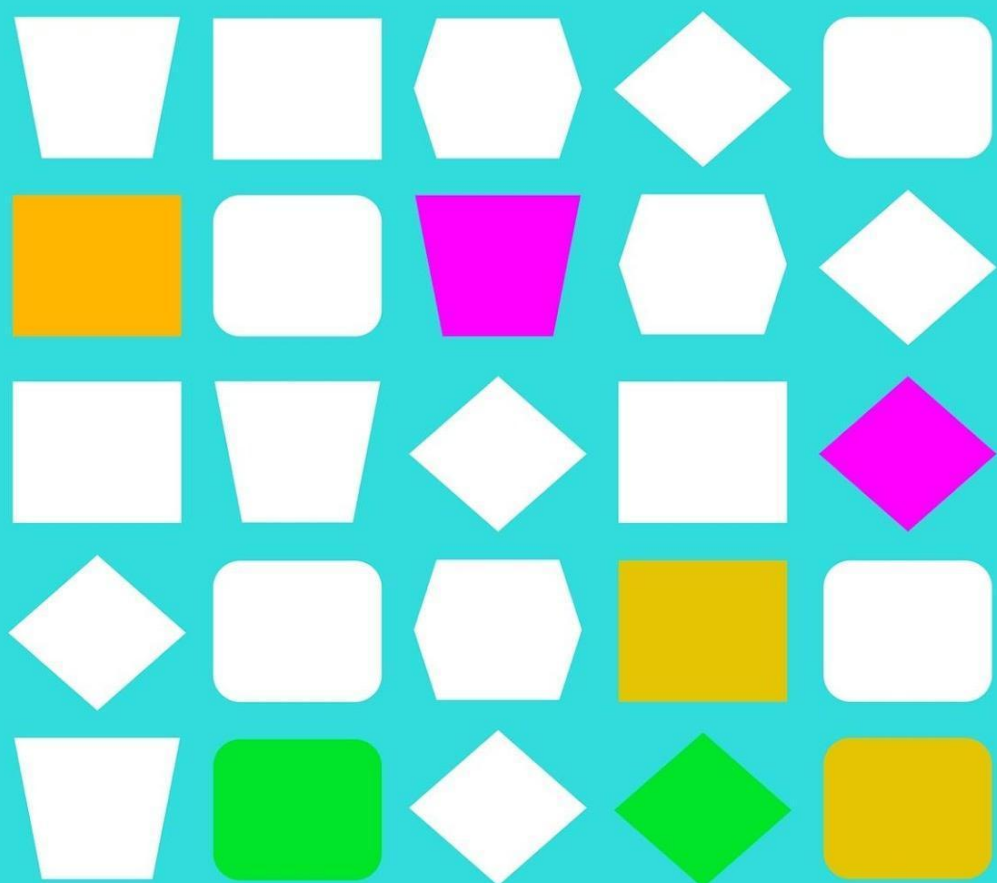
CT integreren in de les

- Eindpresentatie van projecten
 - Laat de leerlingen als presentatie een tekenfilm in het programma Scratch, een flip-overboek of een TikTok-filmpje maken.
- Als vervanging van uitleg:
 - Rekenen: Laat leerlingen een rekenmachine maken op de Micro:bit.
 - Spelling: Laat leerlingen een spel maken waarmee je kinderen van een andere groep spelling kunt uitleggen.
 - Taal: Laat leerlingen hun eigen codetaal maken, een opstel schrijven in de vorm van een stripverhaal in Scratch of de micro:bit gebruiken om een spelletje te maken waarmee je woordjes kunt leren.

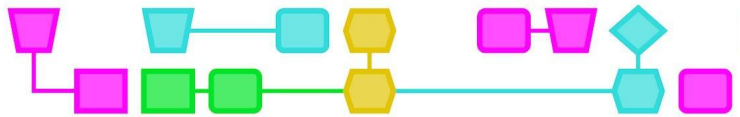
CT in een losstaand project/opdracht

- Als creatieve opdracht
 - Laat de leerlingen gebruikmaken van interactieve tools om kunst te maken. Denk aan Scratch, Lego Education spike, Lego WeDo. Bijvoorbeeld een interactief kunstwerk, verhaal of landschap.
 - Als voorbeeld van een creatieve opdracht zijn er twee voorbeeldlessen gemaakt: 'Maak een (mini)lichtshow (met micro:bit)' en 'Maak een (mini)lichtshow'.

Zie Bijlage 2 voor meer voorbeelden van 'X'.



Bijlagen



Bijlage 1: Format Maak X

Samenvatting

In deze Maak X-les heb je als docent besloten wat 'X' wordt, maar mogen leerlingen zelf de voorwaarden bedenken. Leerlingen gebruiken bij het nadenken over de onderdelen van een opdracht en om hun voorwaarden te kiezen CT-concepten als abstractie en decompositie.

De leerlingen krijgen eerst een introductie over wat 'X' inhoudt en gaan daarna aan de slag met experimenteren en onderzoeken. Ze gaan in groepsverband of individueel aan de slag met materiaal of ideeën om 'X' vorm te geven. Daarna worden er in klassikaal verband voorwaarden opgesteld voor 'X'. Leerlingen maken een plan voor 'X' en gaan aan de slag. Hierna presenteren ze 'X' aan de klas.

Leerdoelen (afhankelijk van de activiteit)

- Leerlingen kunnen CT gebruiken tijdens het uitvoeren van 'X'
- Leerlingen kunnen de voorwaarden voor 'X' klassikaal vaststellen

Online/offline: kan zowel online als offline

Computational thinking: Afhankelijk van de activiteit.

Algemene vaardigheden: Samenwerken, creatief denken.

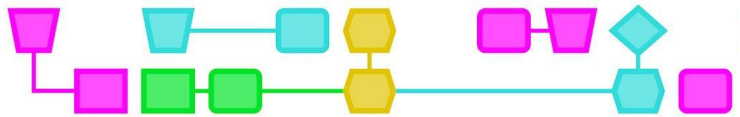
Materialen

Het benodigde materiaal is afhankelijk van 'X' als activiteit. Als leerlingen een bepaald soort materiaal/middel moeten gebruiken, kun je dit verzamelen en klaarleggen. Eventueel kun je ook leerlingen laten bepalen wat voor materiaal ze willen gebruiken voor de opdracht. Dit kun je doen door het eerste deel ('Experimenteren en voorwaarden (35 min)') ongeveer een week eerder te geven dan het tweede deel ('X' maken en presenteren (40 min)'). Je kunt leerlingen dan ook bijvoorbeeld thuis materiaal laten verzamelen om aan de slag te gaan met 'X'.

Vorbereiding

De voorbereiding hangt af van de activiteit.

De tijdsplanning in deze lesomschrijving is een inschatting van de minimale tijd die nodig is voor ieder onderdeel – de duur kan verkort of verlengd worden als dat nodig is.



Deel 1: Experimenteren en voorwaarden (35 min)

Introductie (5 min)

Leg aan de leerlingen uit wat de opdracht is en wat ze gaan maken, vertel dat ze zelf de voorwaarden mogen bepalen. Voorwaarden zijn eisen waar 'X' aan moet voldoen. D ze experimenteren en onderzoeken naar hoe ze 'X' willen vormgeven zodat ze een idee van 'X' kunnen krijgen.

Experimenteer en onderzoek (20 min)

Verdeel de klas in groepjes van 2-4 leerlingen. De leerlingen kunnen nu 20 minuten experimenteren met beschikbaar materiaal en/of onderling overleggen over wat ze zouden willen maken. Als je de leerlingen zelf materiaal laat verzamelen, kunnen ze nu ook bijvoorbeeld een materiaallijst maken voor hun 'X'-ideeën.

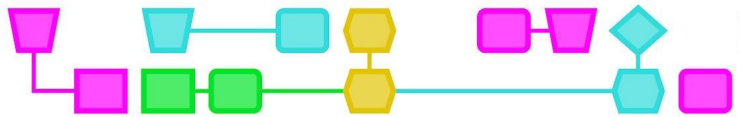
Voorwaarden bepalen (10 min)

Vertel de leerlingen dat ze klassikaal de voorwaarden van 'X' gaan bepalen. Vraag de leerlingen welke voorwaarden ze kunnen bedenken. Laat ze een voor een dingen opnoemen.

Als het grootste gedeelte van de klas het eens is met de voorwaarden, kun je deze op het bord schrijven. Zorg dat er uiteindelijk een duidelijke (en korte) lijst voorwaarden is voor de leerlingen.

Vragen om leerlingen op weg te helpen:

- Welk materiaal moet er gebruikt worden?
- Hoelang moet 'X' duren?
- Wat ga je uiteindelijk presenteren?



Deel 2: 'X' maken en presenteren (40 min)

Plan maken en uitvoeren (20 min)

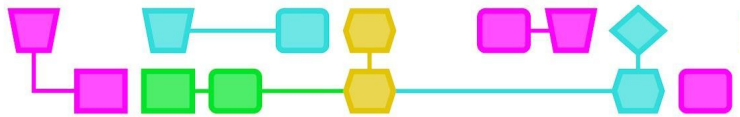
Vertel de leerlingen dat het belangrijk is om van tevoren te overleggen over hoe ze 'X' gaan uitvoeren. Wanneer ze een redelijk plan hebben opgesteld, kunnen ze aan de slag gaan met het maken van 'X'. Om de leerlingen te ondersteunen kun je rondlopen en meedenken over oplossingen voor problemen waar ze tegen aanlopen.

'X' presenteren (15 min)

Roep de leerlingen weer bijeen en vraag hoe het is gegaan. Wat vonden ze leuk om te doen en waar liepen ze tegenaan? Roep daarna een voor een de groepjes naar voren om 'X' te presenteren. Vraag naar hoe ze het hebben aangepakt en waarom. Vraag ook waar ze trots op zijn.

Afsluiting van de les (5 min)

Bespreek de opdracht en hoe het bepalen van de voorwaarden verlopen is. Vraag wat ze van de les vonden. Laat de leerlingen opruimen en nabespreken.



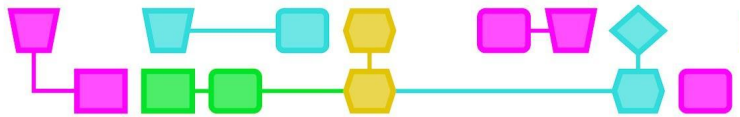
Bijlage 2: Voorbeelden

CT in lessen integreren

- Maak een presentatie
 - Een tekenfilm in Scratch
 - Een flip-overboek
 - Een klasgenoot 'programmeren' om de presentatie te geven
 - Leg een bepaald hoofdstuk uit het boek uit alsof je lesgeeft aan baby's
- Vakspecifiek
 - Aardrijkskunde: een digitale uitleg van aardverschuivingen
 - Biologie: ga op zoek naar patronen in de natuur
 - Natuurkunde: leg energie uit door middel van beweging
 - Topografie: kook het meest populaire eten van een land
 - Rekenen: maak een rekenmachine op de micro:bit
 - Spelling: maak een spel waarmee je kinderen van een andere groep spelling kunt uitleggen
 - Taal: maak je eigen codetaal

CT in een losstaand project/een losstaande opdracht gebruiken

- Maak een schilderij/tekening
 - Een boslandschap
 - Een onderwaterwereld
 - Het klaslokaal
 - De ruimte
- Maak een plan
 - Voor een snoepjesfabriek
 - Voor een raketlancering
 - Voor een moestuin
 - Voor een recept
- Maak iets van een bepaald materiaal
 - Iets van piepschuim
 - Iets van karton
 - Iets van aluminiumfolie
 - Iets in een (schoenen)doos
- Maak een knutselwerk
 - Een slinger
 - Een poster
 - Een boom
 - Een dier



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

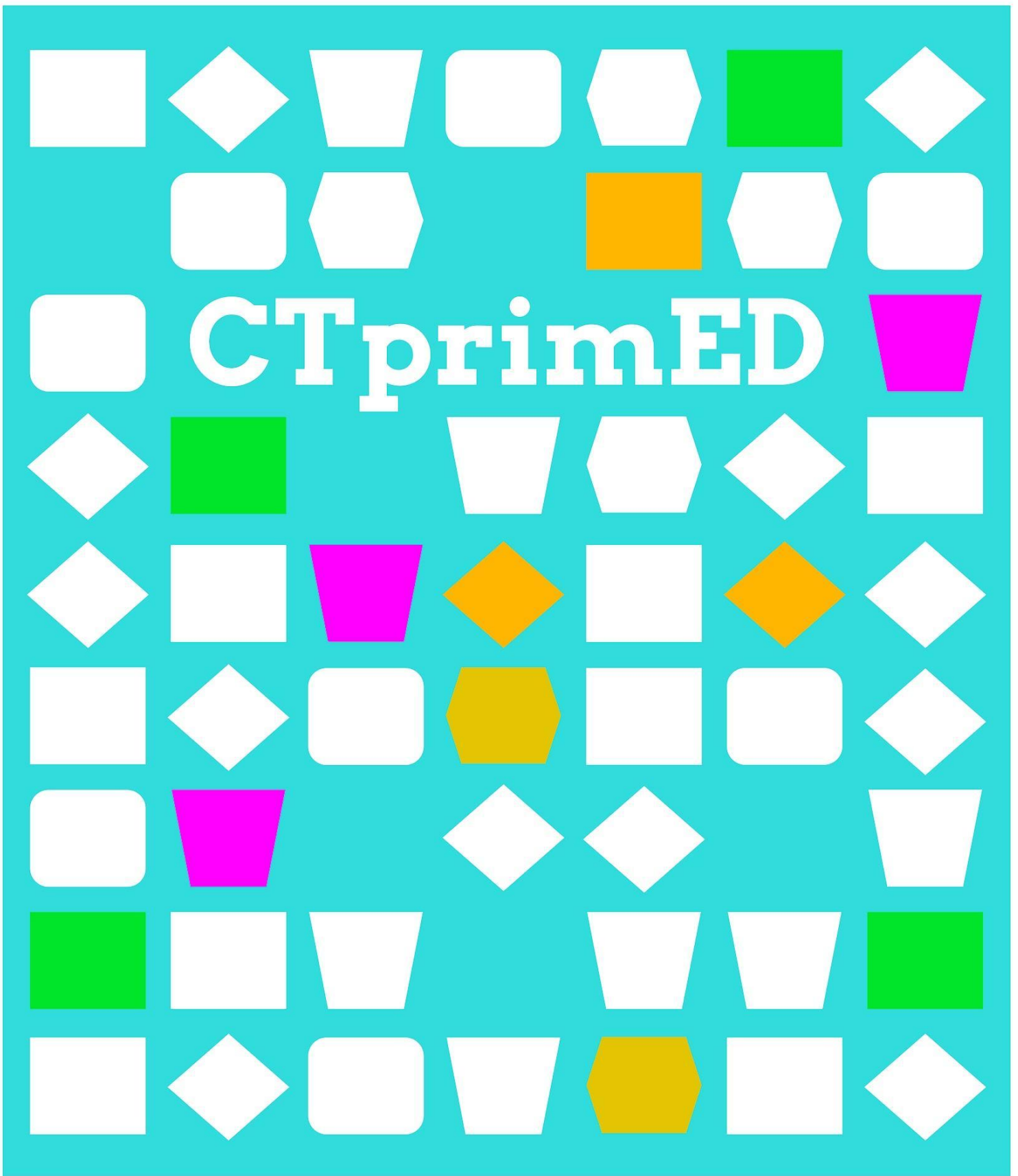


Co-funded by the
Erasmus+ Programme
of the European Union

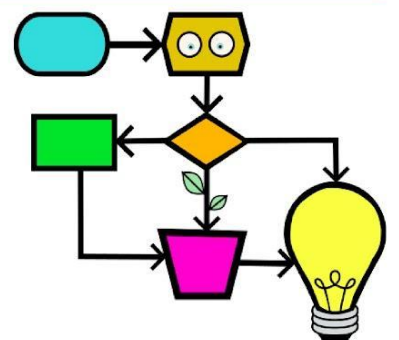


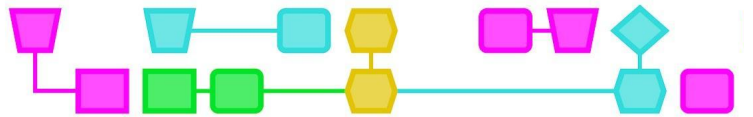
Universidad de Deusto
University of Deusto

Deusto



CT Escaperoom!





Samenvatting

De leerlingen gaan in deze les vijf uitdagingen aan en als ze die hebben opgelost, krijgen ze een deel van de code om aan het eind met elkaar een opvoering te doen. De opdracht is dus gestructureerd als een escaperoom en behandelt naast de CT-gerelateerde concepten ook soft skills zoals logisch denken, teamwork en timemanagement. Tijdens de laatste uitdagingen kunnen de deelnemers experimenteren met de relatie tussen CT en kunst, in dit geval door middel van een voorstelling.

Doelgroep: Leerlingen uit de laatste cyclus van het basisonderwijs, 10-12 jaar.

Duur: 120 minuten in twee sessies. De activiteit kan ook worden opgedeeld in kleinere sessies waarin per sessie 1-2 uitdagingen worden behandeld.

Leerdoelen: op een leuke manier verschillende concepten met betrekking tot *computational thinking* te leren.

Online/offline: offline

Computational thinking:

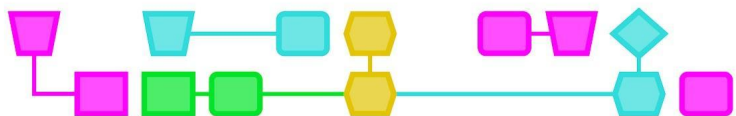
- Algemene vaardigheden: samenwerking, logisch denken, probleemoplossing, timemanagement
- CT-foundations: algoritme, decompositie, abstractie, patroonherkenning
- CT-concepten: binaire code, programmeren, debuggen, alternatieve zin, repeterende zin, sequenties, coderen en decoderen

Materialen:

- Sjablonen bij de activiteiten, één op elk blad, zodat de uitdagingen een voor een aan de groep kunnen worden uitgedeeld
- Potlood en gum
- Timer om de resterende tijd voor het oplossen van de uitdagingen weer te geven

Vorbereiding

Verdeel de klas in groepjes van 4 tot 8 deelnemers. Elke groep moet in een aparte ruimte spelen met een kopie van al het materiaal dat nodig is voor de sessie.



Verzamel de code... en ontdek de verrassing!

Introductie (5 min)

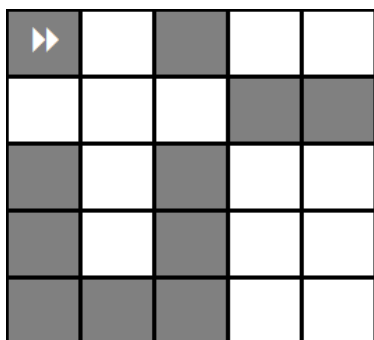
Leg de leerlingen uit dat ze zich in een escaperoom bevinden, en dat ze vijf uitdagingen moeten oplossen, die te maken hebben met computational thinking.

Na het oplossen van elke uitdaging krijgen de leerlingen een coderegel van de laatste 'puzzel'. Er zijn drie puzzels, die verdeeld kunnen worden over drie escaperooms. Als er meer of minder kamers zijn, kan de docent kiezen voor een andere verdeling, zodat aan het eind alle coderegels onder de deelnemers zijn verdeeld.

Lesbeschrijving (45-60 min)

UITDAGING 1 - PROGRAMMERING OM EEN GRAFIEK TE SCHRIJVEN

Maak met behulp van de pijlen in de tabel het juiste programma om deze grafiek te krijgen.



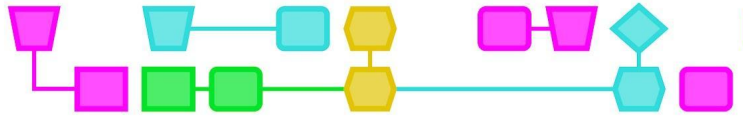
| Verplaats een vakje naar rechts | Verplaats een vakje naar links | Verplaats een vakje omhoog | Verplaats een vakje omlaag | Verf |
|---------------------------------|--------------------------------|----------------------------|----------------------------|---|
| → | ← | ↑ | ↓ |  |



Dit symbool geeft het begin en de richting van het programma aan.

Schrijf hier de programma-instructies in de juiste volgorde (van links naar rechts, en van boven naar beneden):






| | | | | | | | | | |
|--|--|--|--|--|--|--|--|--|--|
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |

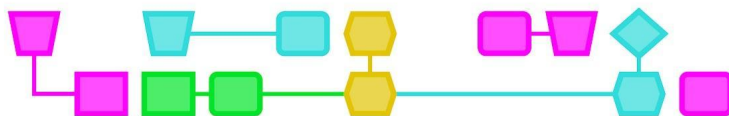


Een tijdslimiet vaststellen voor elke uitdaging en die weergegeven met een timer verhoogt de spanning. Pas de tijd aan aan het aantal deelnemers in de groep, hun leeftijd en vaardigheden.

UITDAGING 2 - BINAIR TELLEN


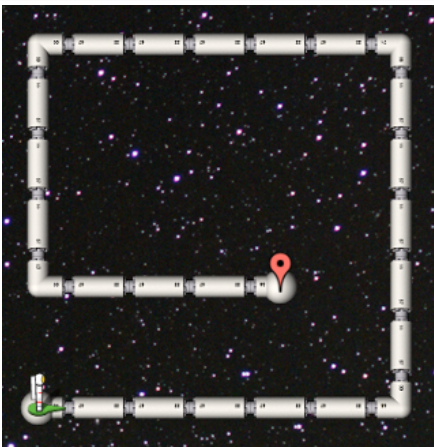
Deze tabel laat zien hoe binaire code werkt. Vul de laatste rijen in en gebruik de waarden in de bovenste rijen als voorbeeld:

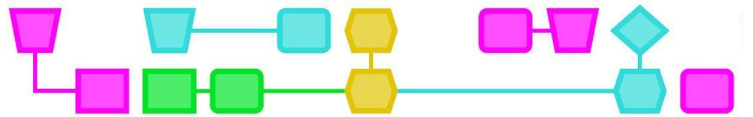
|  |  |  |  |  | Binair getal | Decimaal getal |
|---|---|---|---|--|--------------|----------------|
| 0 | 0 | 0 | 1 | 1 | 00011 | 3 |
| 0 | 1 | 0 | 0 | 1 | 01001 | 9 |
| 1 | 1 | 0 | 1 | 0 | 11010 | 26 |
| 0 | 0 | 0 | 0 | 1 | 00001 | 1 |
| | | | | | | 5 |
| | | | | | 01101 | |
| | | | | | | 11 |
| | | | | | 10101 | |



UITDAGING 3 - PROGRAMMERING OM DE ASTRONAUT NAAR HET DOEL TE KRIJGEN

Schrijf het programma dat de astronaut helpt het doel te bereiken.

| Beginpositie | Beschikbare instructies | Eerste instructies Je mag slechts 8 regels gebruiken (op de puntjes uit de laatste regel bij beschikbare instructies kun je een andere regel plaatsen, dit telt als twee regels) |
|---|--|---|
|  | <p>Ga vooruit Draai naar Links Draai naar Rechts Herhaal tot eindpunt</p> | |
|  | <p>Ga vooruit Draai naar Links Draai naar Rechts Herhaal tot eindpunt Als weg doe dan anders</p> | |



UITDAGING 4 - BERICHTEN DECODEREN

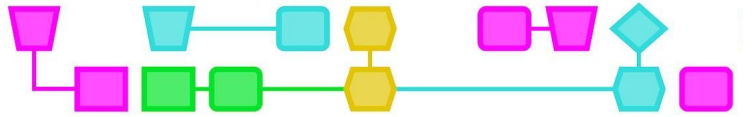
Decodeer het volgende bericht. Een rij staat voor één letter.
Stel het bericht samen met alle letters.

Bericht:

| Binair systeem | | | | | Decimaal systeem | Letter |
|----------------|---|---|---|---|------------------|--------|
| 16 | 8 | 4 | 2 | 1 | | |
| 0 | 1 | 0 | 0 | 0 | | |
| 1 | 0 | 0 | 1 | 0 | | |
| 0 | 0 | 1 | 1 | 0 | | |
| 0 | 0 | 1 | 0 | 1 | | |
| 0 | 1 | 0 | 0 | 0 | | |
| 0 | 0 | 1 | 1 | 0 | | |
| 0 | 0 | 1 | 0 | 1 | | |
| 0 | 0 | 0 | 0 | 1 | | |
| 0 | 0 | 0 | 0 | 1 | | |
| 1 | 0 | 0 | 0 | 0 | | |

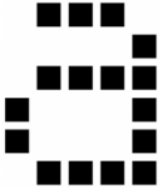
Decoderingstabel:

| | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | |
| a | b | c | ch | d | e | f | g | h | i | j | k | l | ll | |
| 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 |
| m | n | ñ | o | p | q | r | s | t | u | v | w | x | y | z |



UITDAGING 5 - AFBEELDINGEN CODEREN

De letter **a** in een computer, ingezoomd op de pixels, kan er zo uitzien:

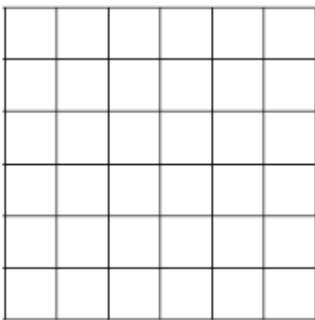


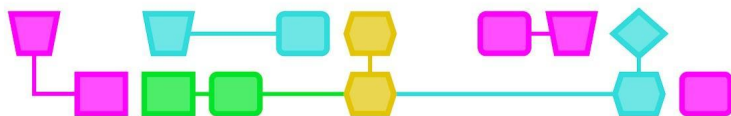
Dit kan worden gecodeerd met nummers:

- Het eerste getal geeft het aantal witte vakjes in een rij aan.
- Het volgende getal geeft het aantal zwarte vakjes in een rij aan.
- Herhaal dit tot je het einde van de reeks in elke rij bereikt.
- Uitzondering: wanneer een rij begint met een zwart vakje, begint die met een nul (0) en dan geeft het getal aan hoeveel zwarte vakjes er in een rij staan.

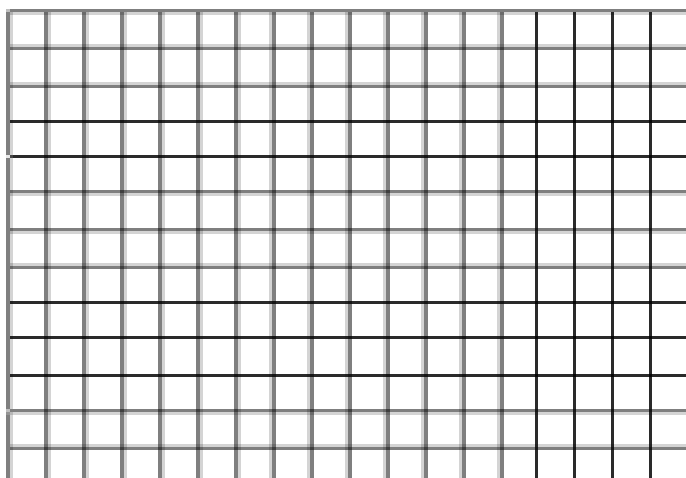
| | | | | | |
|---|---|---|---|---|------------|
| | ■ | ■ | ■ | | 1, 3, 1 |
| | | | | ■ | 4, 1 |
| | ■ | ■ | ■ | ■ | 1, 4 |
| ■ | | | | ■ | 0, 1, 3, 1 |
| ■ | | | | ■ | 0, 1, 3, 1 |
| | ■ | ■ | ■ | ■ | 1, 4 |

Schrijf een letter (bijvoorbeeld R) en codeer die met cijfers ?





Doe nu de omgekeerde oefening, vul het raster in aan de hand van de gegeven getallenreeks:



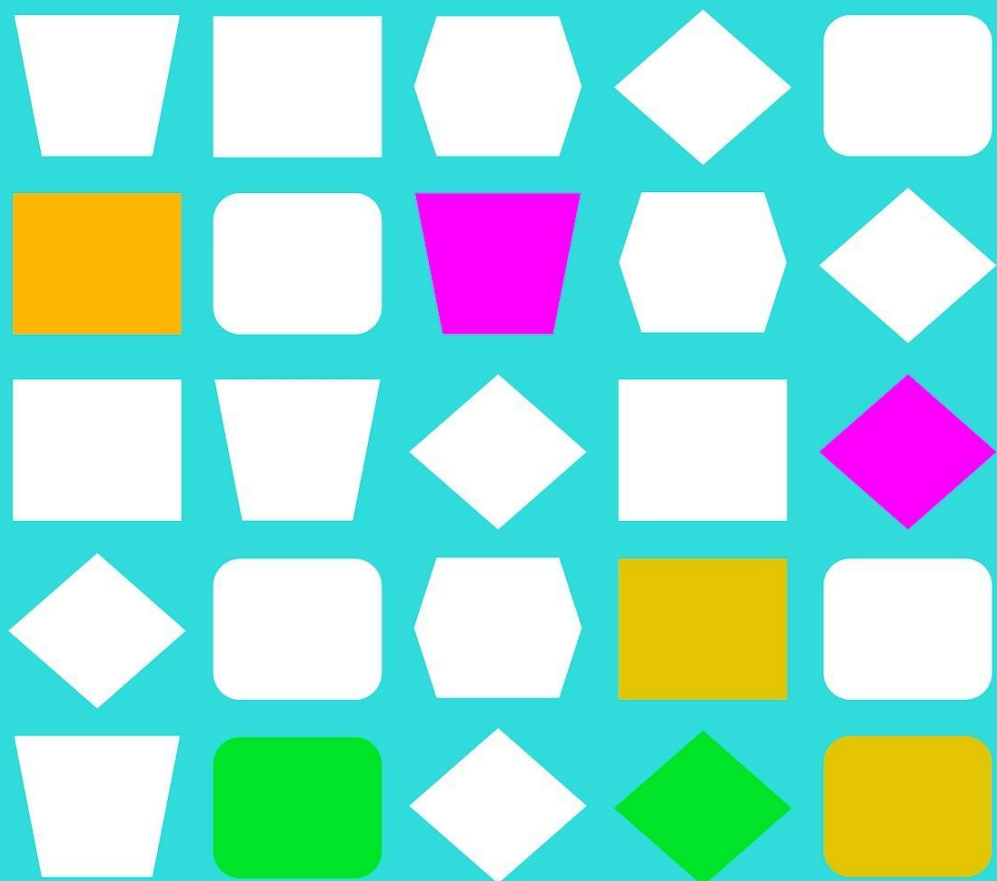
- 6, 5, 2, 3
- 4, 2, 5, 2, 3, 1
- 3, 1, 9, 1, 2, 1
- 3, 1, 9, 1, 1, 1
- 2, 1, 11, 1
- 2, 1, 10, 2
- 2, 1, 9, 1, 1, 1
- 2, 1, 8, 1, 2, 1
- 2, 1, 7, 1, 3, 1
- 1, 1, 1, 1, 4, 2, 3, 1
- 0, 1, 2, 1, 2, 2, 5, 1
- 0, 1, 3, 2, 5, 2
- 1, 3, 2, 5

Conclusie (10 min)

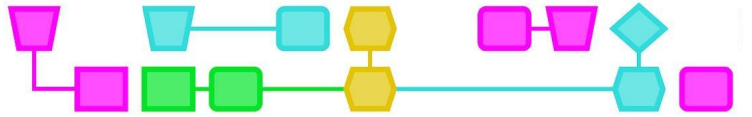
In deze activiteit zijn verschillende concepten van computational thinking op een speelse manier aan bod gekomen en ook in verband gebracht met kunst. Met de verzamelde puzzelstukjes kunnen de leerlingen een choreografie uitvoeren die als een algoritme beschreven is.

Bespreek met de leerlingen welke CT-concepten ze hebben geleerd, welke CT-fundamenten ze hebben toegepast en welke vaardigheden nodig zijn om alle uitdagingen in de activiteit aan te gaan.

Om de activiteit uit te breiden, kan je ook een nieuwe uitdaging aan de klas voorstellen: bedenk een kort toneelstuk en ontwerp een algoritme voor elk profiel (bijvoorbeeld verschillende personages, belichting, muziek, decorontwerper, enz.).

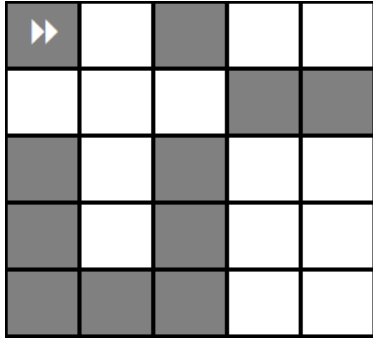


Bijlagen



Bijlage 1: Werkblad - Uitdaging 1

De uitdaging:

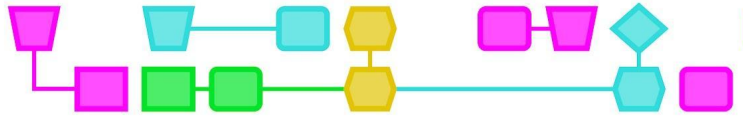


De mogelijke instructies:






| Verplaats een vakje naar rechts | Verplaats een vakje naar links | Verplaats een vakje omhoog | Verplaats een vakje omlaag | Verf |
|---------------------------------|--------------------------------|----------------------------|----------------------------|------|
| → | ← | ↑ | ↓ | ■ |

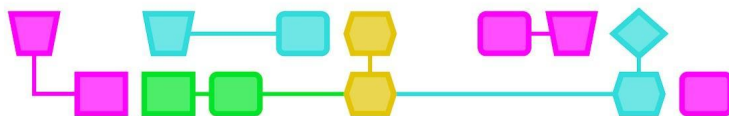
Schrijf hier je programmering (volgorde van instructies), van links naar rechts, en van boven naar beneden:

| | | | | | | | | | |
|--|--|--|--|--|--|--|--|--|--|
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |


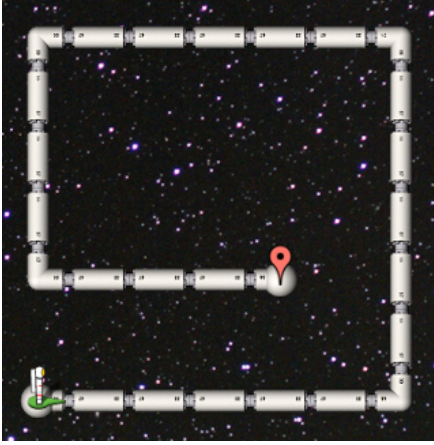


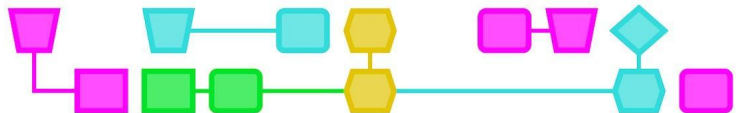
Werkblad - Uitdaging 2

|  |  |  |  |  | Binair getal | Decimaal getal |
|---|---|---|---|--|-----------------|-------------------|
| 0 | 0 | 0 | 1 | 1 | 00011 | 3 |
| 0 | 1 | 0 | 0 | 1 | 01001 | 9 |
| 1 | 1 | 0 | 1 | 0 | 11010 | 26 |
| 0 | 0 | 0 | 0 | 1 | 00001 | 1 |
| | | | | | | 5 |
| | | | | | 01101 | |
| | | | | | | 11 |
| | | | | | 10101 | |



Werkblad - Uitdaging 3

| Beginpositie | beschikbare instructies | Eerste instructies Je kunt slechts 8 blokken gebruiken |
|---|--|---|
|  | <p>Ga vooruit Draai naar links Draai naar rechts Herhaal tot eindpunt</p> | |
|  | <p>Ga vooruit Draai naar links Draai naar rechts Herhaal tot eindpunt Als weg doe dan anders</p> | |



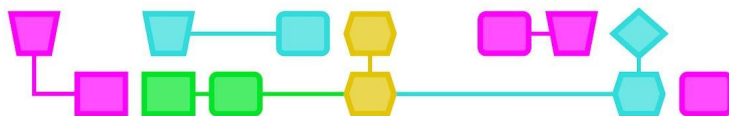
Werkblad - Uitdaging 4

Het bericht:

| Binair systeem | | | | | Decimaal systeem | Letter |
|----------------|---|---|---|---|------------------|--------|
| 16 | 8 | 4 | 2 | 1 | | |
| 0 | 1 | 0 | 0 | 0 | | |
| 1 | 0 | 0 | 1 | 0 | | |
| 0 | 0 | 1 | 1 | 0 | | |
| 0 | 0 | 1 | 0 | 1 | | |
| 0 | 1 | 0 | 0 | 0 | | |
| 0 | 0 | 1 | 1 | 0 | | |
| 0 | 0 | 1 | 0 | 1 | | |
| 0 | 0 | 0 | 0 | 1 | | |
| 0 | 0 | 0 | 0 | 1 | | |
| 1 | 0 | 0 | 0 | 0 | | |

Decoderingstabel:

| | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | |
| a | b | c | ch | d | e | f | g | h | i | j | k | l | ll | |
| 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 |
| m | n | ñ | o | p | q | r | s | t | u | v | w | x | y | z |



Bijlage 2: Coderegels voor leerlingen na het oplossen van elke uitdaging.

Klappen (elke rij na het oplossen van een uitdaging in de escaperoom):

| | | | | | | | |
|---|-------------|----|-------------|-----|----------|----------|---|
| 1 | X | xx | X | xx | X | xx | - |
| 2 | xxx | XX | xxx | X | xxx | XX | X |
| 3 | X | xx | xxx | X | xx | xxx | X |
| 4 | xx | X | Rep 3 maal: | X | xx | Eind-rep | X |
| 5 | Rep 2 maal: | X | xx | xxx | Eind-rep | XX | X |

X – grote klap

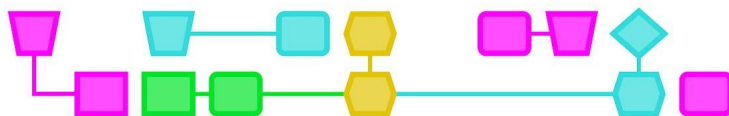
x – kleine klap

Beweging (elke rij na het oplossen van een uitdaging in de escaperoom):

| | | | | | | | |
|---|-------------|--|-------------|--|----------|----------|--|
| 1 | | | | | | | |
| 2 | | | | | | | |
| 3 | | | | | | | |
| 4 | | | Rep 3 maal: | | | Eind-rep | |
| 5 | Rep 2 maal: | | | | Eind-rep | | |

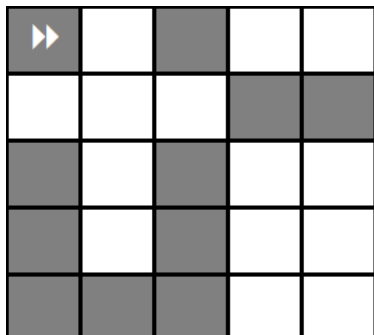
Verlichting (elke rij na het oplossen van een uitdaging in de escaperoom):

| | | | | | | | |
|---|-------------|--|-------------|--|----------|----------|--|
| 1 | | | | | | | |
| 2 | | | --- | | | --- | |
| 3 | | | | | | | |
| 4 | | | Rep 3 maal: | | --- | Eind-rep | |
| 5 | Rep 2 maal: | | | | Eind-rep | | |



Bijlage 3: Oplossingen

UITDAGING 1



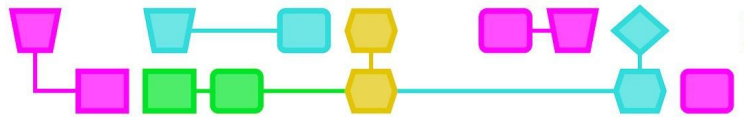
| Verplaats een vakje naar rechts | Verplaats een vakje naar links | Verplaats een vakje omhoog | Verplaats een vakje omlaag | Verf |
|---------------------------------|--------------------------------|----------------------------|----------------------------|------|
| → | ← | ↑ | ↓ | ■ |

Schrijf hier je programmeerinstructies in de juiste volgorde (van links naar rechts, en van boven naar beneden). Er zijn meerdere oplossingen, een ervan zou deze kunnen zijn:


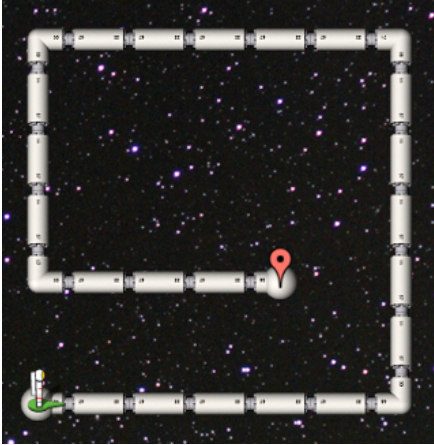
| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| → | → | ■ | → | → | ↓ | ■ | ← | ■ | ← |
| ↓ | ■ | ↓ | ■ | ↓ | ■ | ← | ■ | ← | ■ |
| ↑ | ■ | ↑ | ■ | | | | | | |

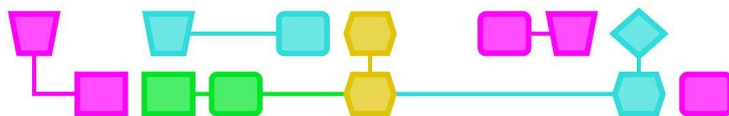
UITDAGING 2

| | | | | | Binair getal | Decimaal getal |
|---|---|---|---|---|--------------|----------------|
| 0 | 0 | 0 | 1 | 1 | 00011 | 3 |
| 0 | 1 | 0 | 0 | 1 | 01001 | 9 |
| 1 | 1 | 0 | 1 | 0 | 11010 | 26 |
| 0 | 0 | 0 | 0 | 1 | 00001 | 1 |
| 0 | 0 | 1 | 0 | 1 | 00101 | 5 |
| 0 | 1 | 1 | 0 | 1 | 01101 | 13 |
| 0 | 1 | 0 | 1 | 1 | 01011 | 11 |
| 1 | 0 | 1 | 0 | 1 | 10101 | 21 |



UITDAGING 3

| Beginpositie | beschikbare instructies | Eerste instructies Je kunt slechts 8 blokken gebruiken |
|--|---|--|
|  | <p>Ga vooruit Draai naar links Draai naar rechts Herhaal tot eindpunt</p> | <p>Herhaal Ga vooruit Draai naar links Ga vooruit Draai naar rechts tot eindpunt.</p> |
|  | <p>Ga vooruit Draai naar links Draai naar rechts Herhaal tot eindpunt Als weg doe dananders</p> | <p>Herhaal</p> <p>Als pad naar voren, doe dan Ga vooruit, Anders Ga naar links ... tot eindpunt.</p> |

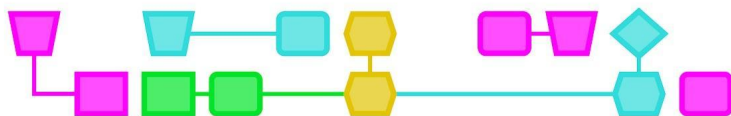


UITDAGING 4

Bericht:

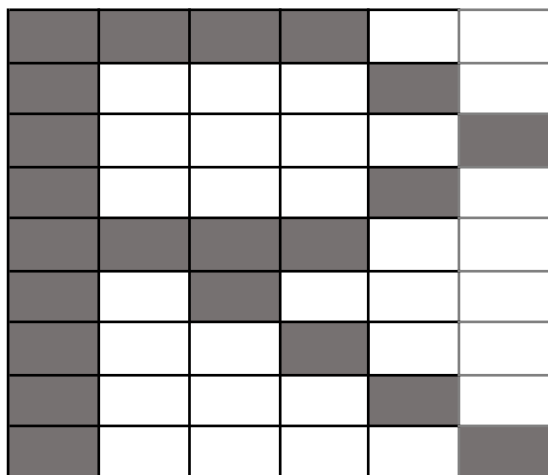
| Binair systeem | | | | | Decimaal systeem | Letter |
|----------------|---|---|---|---|------------------|--------|
| 16 | 8 | 4 | 2 | 1 | | |
| 0 | 1 | 0 | 0 | 0 | 8 | G |
| 1 | 0 | 0 | 1 | 0 | 18 | o |
| 0 | 0 | 1 | 1 | 0 | 6 | e |
| 0 | 0 | 1 | 0 | 1 | 5 | d |
| 0 | 1 | 0 | 0 | 0 | 8 | G |
| 0 | 0 | 1 | 1 | 0 | 6 | e |
| 0 | 0 | 1 | 0 | 1 | 5 | d |
| 0 | 0 | 0 | 0 | 1 | 1 | a |
| 0 | 0 | 0 | 0 | 1 | 1 | a |
| 1 | 0 | 0 | 0 | 0 | 16 | n |

| | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | |
| a | b | c | ch | d | e | f | g | h | i | j | k | l | ll | |
| 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 |
| m | n | ñ | o | p | q | r | s | t | u | v | w | x | y | z |



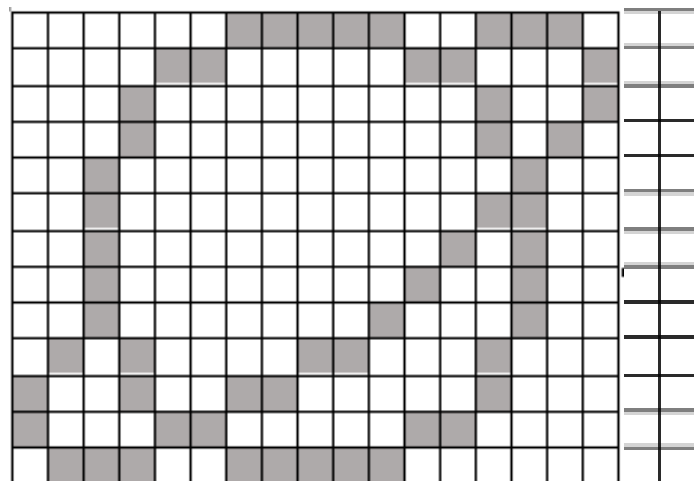
UITDAGING 5

Schrijf een letter (bijvoorbeeld R) en codeer die met cijfers:

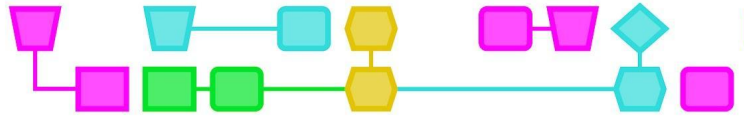


| |
|-----------|
| 0 4 2 |
| 0 1 3 1 1 |
| 0 1 4 1 |
| 0 1 3 1 1 |
| 0 4 2 |
| 0 1 1 1 3 |
| 0 1 2 1 2 |
| 0 1 3 1 1 |
| 0 1 4 1 |

Doe nu de omgekeerde oefening, vul het raster in aan de hand van de gegeven getallenreeks:



6, 5, 2, 3
 4, 2, 5, 2, 3, 1
 3, 1, 9, 1, 2, 1
 3, 1, 9, 1, 1, 1
 2, 1, 11, 1
 2, 1, 10, 2
 2, 1, 9, 1, 1, 1
 2, 1, 8, 1, 2, 1
 2, 1, 7, 1, 3, 1
 1, 1, 1, 1, 4, 2, 3, 1
 0, 1, 2, 1, 2, 2, 5, 1
 0, 1, 3, 2, 5, 2
 1, 3, 2, 5



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

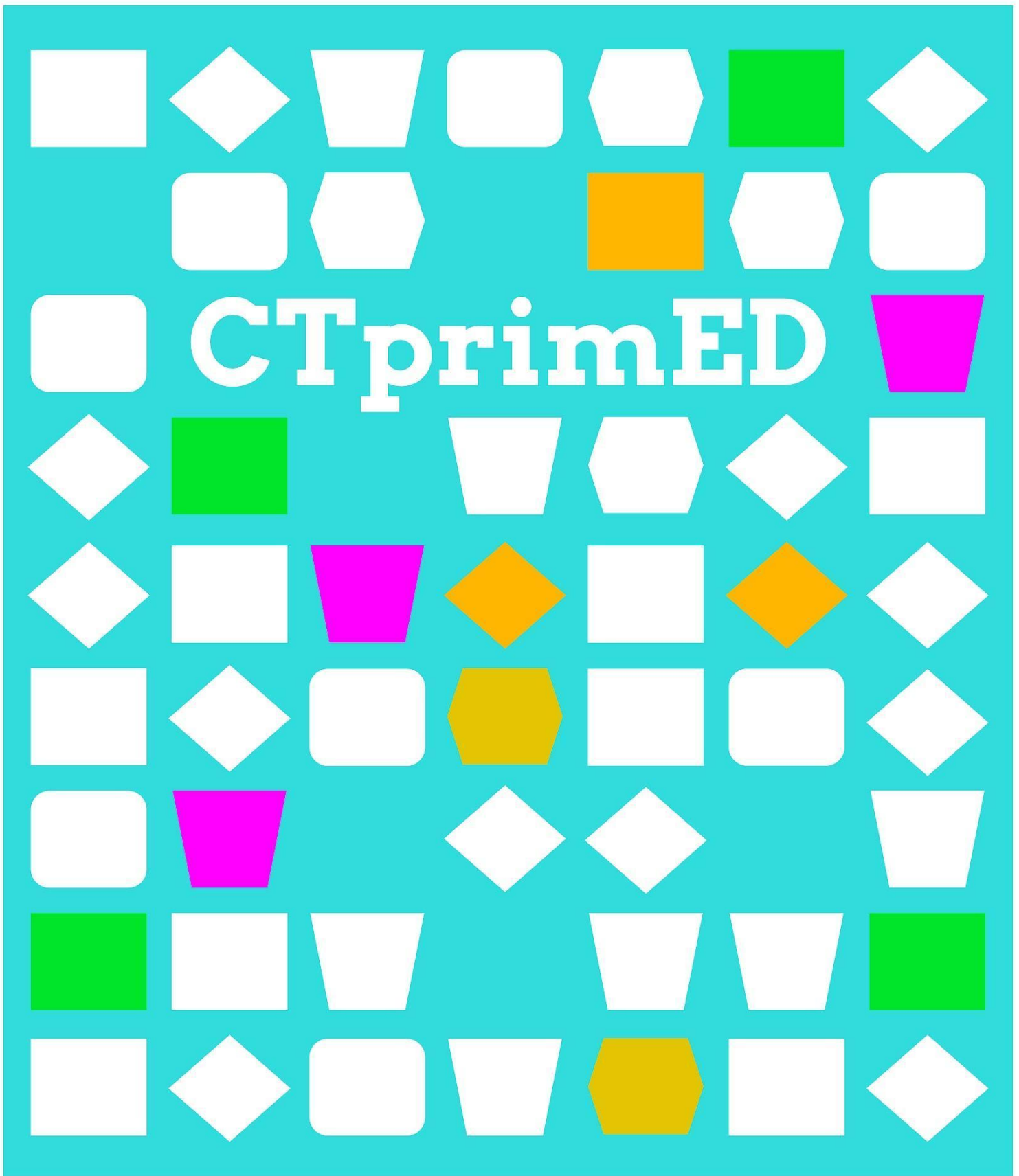


Co-funded by the
Erasmus+ Programme
of the European Union

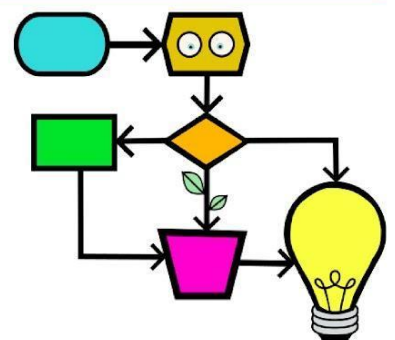


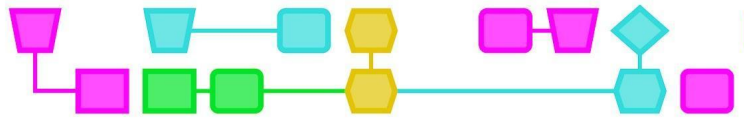
Universidad de Deusto
University of Deusto

Deusto



Maak een (mini) lichtshow
(met micro:bit)





Samenvatting

De leerlingen maken in deze workshop een lichtshow met behulp van een micro:bit en ledlampjes. De leerlingen krijgen klassikaal uitleg over hoe ze ledlampjes aansluiten op de micro:bit en hoe ze deze kunnen programmeren. Daarna sluiten de leerlingen in groepjes zelf een lampje aan, ontdekken wat de micro:bit nog meer kan en experimenteren met materiaal aan de hand van een werkblad. Daarna worden klassikaal de voorwaarden voor de lichtshow bepaald. Als ze een plan hebben opgesteld dat aan de voorwaarden voldoet, kunnen ze aan de slag. Wanneer iedereen klaar is, presenteren de leerlingen de lichtshow aan de rest van de klas.

Doelgroep: Ongeveer 9-12 jaar. De leerlingen moeten voldoende kunnen lezen en schrijven en moeten kunnen werken met micro:bit en in MakeCode.

Duur: 100 min

Deel 1: Micro:bit en materialen verkennen (40 min)

Deel 2: De (mini)lichtshow (60 min)

De delen kunnen als aparte lessen gegeven worden.

Leerdoelen:

De leerlingen leren

- voorwaarden voor een opdracht in groepsverband vast te stellen;
- creatieve materiaalkeuzes te maken;
- hoe ze met een micro:bit ledlampjes aan kunnen sturen;
- hoe ze een (mini)lichtshow kunnen maken;
- algoritme te maken.

Online/offline: online (micro:bit en computer)

Computational thinking:

- Algemene vaardigheden: lezen, samenwerken, creatief denken
- CT-foundations: patroonherkenning, algoritmes, abstractie, decompositie
- CT-concepten: het gebruik van een micro:bit, coderen in MakeCode, een elektrisch circuit maken

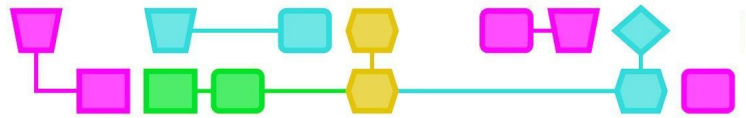
Bijzonderheden: In deze lesbeschrijving gaan we ervan uit dat de leerlingen bekend zijn met de micro:bit. Zie: Extra - Micro:bit handleiding. Deze activiteit moet in een (gedeeltelijk) verduisterde ruimte uitgevoerd worden. Anders zijn de ledlampjes moeilijk te zien.

Materialen:

Per groepje (2-4 leerlingen)

- Micro:bit incl. batterij en kabel (1 per groepje)
- Een computer/laptop (1 per groepje)
- Krokodillenbekjes (minimaal 6 per groepje)
- Ledlampjes (ongeveer 5 per groepje)





Onderstaande materialen zijn een suggestie. Gebruik vooral ook materialen die al in de klas aanwezig zijn, of materialen waar je niet zo snel aan zou denken. Zorg dat elk groepje voldoende materiaal heeft om te experimenteren.

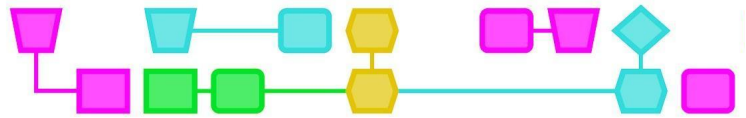
- Algemeen
 - (schoenen)dozen (1 per groepje als de ruimte niet verduisterd kan worden)
 - Wit en gekleurd papier
 - Schaar
 - Lijm
 - Stiften
 - Kleurpotloden
 - Plakband
 - A3-papier
- Reflecterend materiaal (2-3 items per groepje)
 - Aluminiumfolie
 - Reflecterend papier
 - Glitterpapier
 - (Oude) cd's
 - Spiegels
 - Discoballen
 - Prisma's
 - (Camera)lenzen
 - Glazen
- Materiaal om schaduwen mee te creëren (2-3 items per groepje)
 - Garde
 - Vergiet
 - Poppetjes
 - Crêpepapier
 - Cellofaan
 - Glasvezel
 - Gekleurd plastic

Vorbereiding

- Zorg dat alle materialen verzameld zijn en dat er genoeg spullen aanwezig zijn voor leerlingen om creatief aan de slag te gaan met hun lichtshow.
- Maak een voorbeeld van een micro:bit met aangesloten ledlampje (zie bijlage: Werkblad ledlampje aansluiten op de micro:bit). Programmeer deze zo dat het lampje gaat knipperen als je op knop A drukt.
- Bereid de uitleg voor over het coderen van de ledlampjes.

Tip: Gebruik gerecycled materiaal! Bijvoorbeeld oude CD's, oude schoendozen, gekleurd plastic, enz. Vraag de leerlingen een week van te voren thuis te zoeken naar materiaal. Of haal bijvoorbeeld afval op bij een maaklab!

In deze les gebruiken we de micro:bit i.c.m. ledlampjes als programmeerbare lichtbronnen. In plaats hiervan kun je ook gebruik maken van bijvoorbeeld Little Bits, Lego Spike, Raspberry pi, LED-strips, Neopixel strips, lichtknoppen, zaklampen, etc. Zie voor een analoge versie van deze les: 'Maak een (mini) lichtshow'.



Deel 1: Micro:bit en materialen verkennen

Introductie (5 min)

Vertel de leerlingen dat ze een minilichtshow gaan maken met behulp van een micro:bit en ledlampjes. Ze gaan eerst experimenteren met de materialen en daarna klassikaal de voorwaarden bepalen voor de lichtshow.

Uitleg micro:bit (5 min)

Doe het licht uit en laat zien dat je met de micro:bit een ledlampje kunt programmeren. Laat zien hoe je de krokodillenbekjes moet bevestigen en leg uit dat het lange pootje de pluskant is en het korte pootje de minkant. In het uitlegkader hieronder vind je een uitgebreidere uitleg over elektriciteit.

Het is belangrijk dat de leerlingen weten dat de pluskant aangesloten moet worden op de positieve pin (0, 1 of 2), en de minkant op de negatieve GND van de micro:bit, anders is het circuit niet gesloten en zal er geen lampje gaan branden. Je kunt lampjes aansluiten op Pin0, Pin1 en Pin2. **Sluit geen lampje aan op de derde pin, '3V'!** Als je dit wel doet, kun je de micro:bit te veel elektriciteit laten rondsturen waardoor die overbelast kan raken.

Uitleg stroomkring met en micro:bit

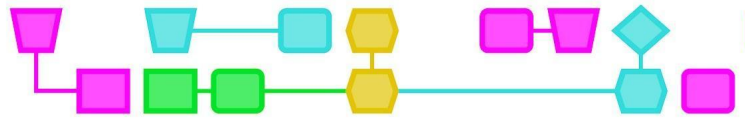
Elektriciteit is een stroom van elektronen. Elektronen zijn kleine deeltjes met een negatieve elektrische lading. Als een elektrisch apparaat aanstaat dan vormen de elektronen een stroomkring. De elektronen kunnen door de draden van het apparaat stromen. Dit noem je een gesloten circuit. Een stroomkring heeft altijd een elektriciteitsbron bijvoorbeeld een batterij, stekker of accu. De micro:bit heeft een batterij. Elke elektriciteitsbron heeft een minkant en een pluskant. Bij de minkant, ook wel minpool, zitten veel negatief geladen elektronen. De minpool is daarom negatief geladen. Bij de pluskant, de pluspool, zitten weinig elektronen. De pluspool is positief geladen. Elektronen stromen altijd van de minpool naar de pluspool, van de plek met veel elektronen naar de plek met weinig elektronen. Als de stroomkring ergens onderbroken wordt kan de stroom niet rondgaan. Daarom kan een elektrisch apparaat of ledlampje alleen maar werken als de stroomkring gesloten is.

Aan de slag met de micro:bit (10 min)

Verdeel de klas in groepjes van 2-4 leerlingen. Geef ieder groepje een micro:bit (incl. batterij en kabel), 2-3 ledlampjes, minimaal 6 krokodillenbekjes en een computer/laptop. Deel het werkblad 'Ledlampje aansluiten op de micro:bit' uit (zie Bijlage) en laat de leerlingen aan de hand van het werkblad een ledlampje aansluiten op hun micro:bit.

Eerst bevestigen de leerlingen het lampje met krokodillenbekjes aan de micro:bit, daarna programmeren ze het lampje. Loop rond en zorg dat ieder groepje ten minste één ledlampje aan heeft gekregen. Controleer of de clips op de goede pootjes zitten en of de leerlingen de MakeCode-instructies goed begrepen hebben.

Tip: Vinden de leerlingen het lastig? Laat de MakeCode code die je hebt gemaakt om het voorbeeldlampje te laten branden op het bord zien.



Experimenteer en onderzoek met materiaal (20 min)

Als een groepje een lampje heeft aangesloten, deel je het werkblad 'Maak een (mini)lichtshow met micro:bit - deel 1' uit (zie Bijlage). Met dit werkblad gaan de leerlingen verder experimenteren met de micro:bit. Ze kijken bijvoorbeeld naar andere functies in MakeCode en ontdekken wat ze verder met de micro:bit kunnen doen. Kunnen ze bijvoorbeeld pictogrammen op het scherm laten zien? Of meerdere ledlampjes aansluiten?

Bij de laatste stap op het werkblad gaan de leerlingen experimenteren met materiaal. Geef aan dat de leerlingen 15 minuten de tijd hebben om het materiaal te onderzoeken en ermee te experimenteren. De leerlingen denken na over hoe ze een cool lichteffect zouden kunnen creëren en welk materiaal ze willen gebruiken voor hun eigen lichtshow.

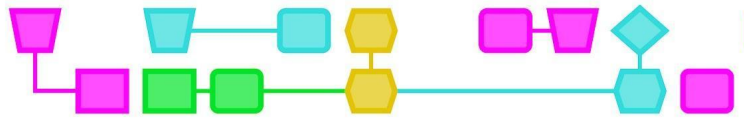
Tip: Zet een digitale visuele timer op het bord, bijvoorbeeld een cirkel. Zo kunnen leerlingen zelf de tijd in de gaten houden.

Tip: Als leerlingen moeite hebben met experimenteren kun je de voorbeelden uit Bijlage 1 laten zien. Laat de leerlingen daarna nog 5-10 minuten experimenteren.

Als je de les in twee delen geeft is dit het einde van de eerste les.

Deel 2: De (mini)lichtshow (60 min)

Herhaal kort wat de leerlingen in het vorige deel gedaan hebben.



Voorwaarden bepalen (5 min)

Vertel de leerlingen dat ze klassikaal de voorwaarden voor de lichtshow gaan bepalen. Voorwaarden zijn eisen waar de lichtshow aan moet voldoen, bijvoorbeeld welk doel de lichtshow heeft, hoelang hij moet duren en hoeveel lichteffecten erin moeten zitten. Vraag de leerlingen welke voorwaarden ze kunnen bedenken. Laat ze een voor een dingen opnoemen.

Vragen om leerlingen op weg te helpen:

- Wat kan het doel van de lichtshow zijn?
 - Denk hierbij aan: een verhaal vertellen, een cool TikTok-filmpje op het ritme van een (zelfgekozen) liedje maken, een bepaalde omgeving namaken (bijvoorbeeld een bos, zee of strand), een sfeer neerzetten (een enge, sprookjesachtige of relaxte sfeer), enz.
- Waar moet de lichtshow aan voldoen?
 - Welke kleuren moet je gebruiken? Hoeveel knoppen mag je indrukken? Welk materiaal moet gebruikt worden?
 - Denk hierbij aan projecteren op het plafond, een schaduwspel maken, een discobal gebruiken, enz. Voorbeelden van lichtshows kun je vinden in Bijlage 1.

Als het grootste gedeelte van de klas het eens is met de voorwaarden, kun je deze op het bord schrijven. Zorg dat er uiteindelijk een duidelijke (en korte) lijst voorwaarden is voor de leerlingen.

Plan maken en uitvoeren (30 min)

Als de klas gezamenlijk de voorwaarden heeft bepaald, kunnen de leerlingen verder met het werkblad. Op het werkblad staan een aantal vragen die de leerlingen helpen bij het ontwerpen van hun lichtshow. Bijvoorbeeld welke kleuren en welk materiaal ze gaan gebruiken of hoe ze de lichtshow willen presenteren. Hierna gaan ze aan de slag met het maken van de lichtshow.

Loop rond terwijl de leerlingen de lichtshow maken en zorg dat ze de lampjes tussendoor af en toe testen. Soms zit bijvoorbeeld een krokodillenbekje niet meer stevig vast, of hebben ze de clip aan het verkeerde pootje vastgemaakt. Zorg ook dat leerlingen tijdig beginnen met het bouwen met materiaal, en dat ze niet blijven hangen in het programmeren van hun lampjes. Geef na 15 min aan dat ze op de helft van de tijd zitten.

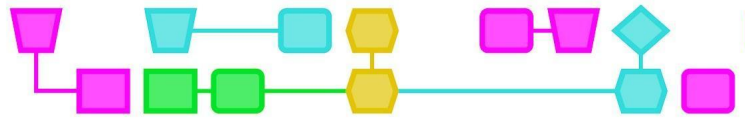
Tip: Zet bij deze stap ook een visuele digitale timer op het bord. Dit helpt de leerlingen met inschatten hoelang ze nog hebben.

Het presenteren van de lichtshow (15 min)

Besprek klassikaal hoe het programmeren is gegaan. Wat vonden ze leuk om te doen en waar liepen ze tegenaan? Roep daarna een voor een de groepjes naar voren om hun lichtshow te presenteren, of ga met de klas rond de tafel staan als een lichtshow niet verplaatst kan worden. Vraag aan de leerlingen hoe ze het maken en programmeren van de lichtshow hebben aangepakt en waarom. Vraag ook waar ze trots op zijn.

Afsluiting van de les (5 min)

Laat de leerlingen opruimen en bespreek de les na. Benadruk dat iedereen dezelfde opdracht kreeg, maar er toch heel diverse lichtshows uit zijn gekomen!



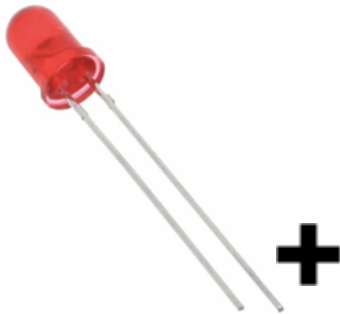
Werkblad - Ledlampje aansluiten op de micro:bit

Wat heb je nodig?

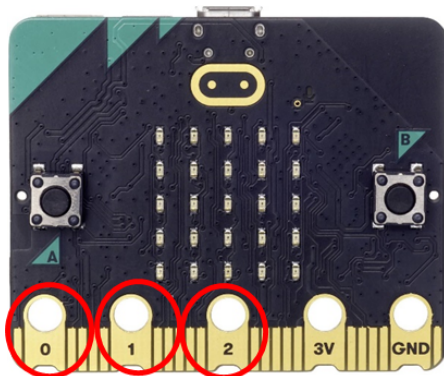
- Een micro:bit
 - Batterij
 - USB-kabel
- Computer/laptop
- Ledlampjes
- Alligatorclips

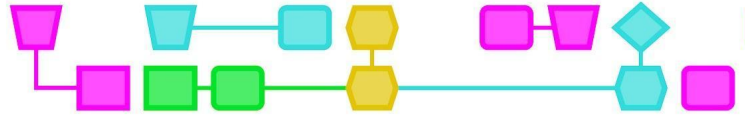
Wat moet je weten?

1. Ledlampjes zijn kleine lampjes waarin de stroom maar één richting op kan gaan. Daarom is het belangrijk om ze goed aan te sluiten. De lampjes hebben een langer pootje (+) en een korter pootje (-).



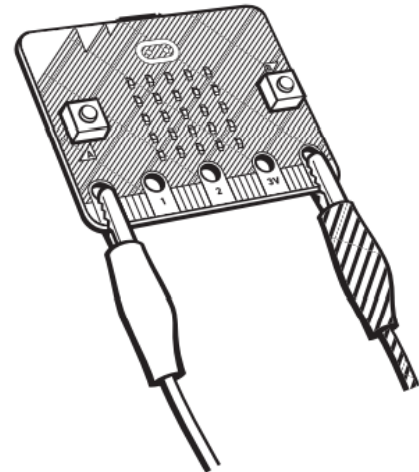
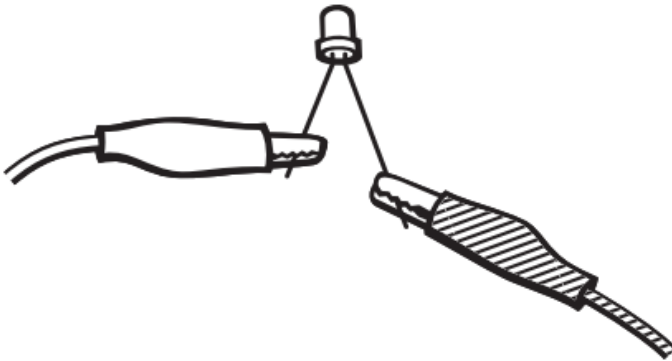
2. Op de micro:bit zitten pins waar stroom doorheen gestuurd kan worden. Als je het ledlampje hierop aansluit, kun je deze laten branden.





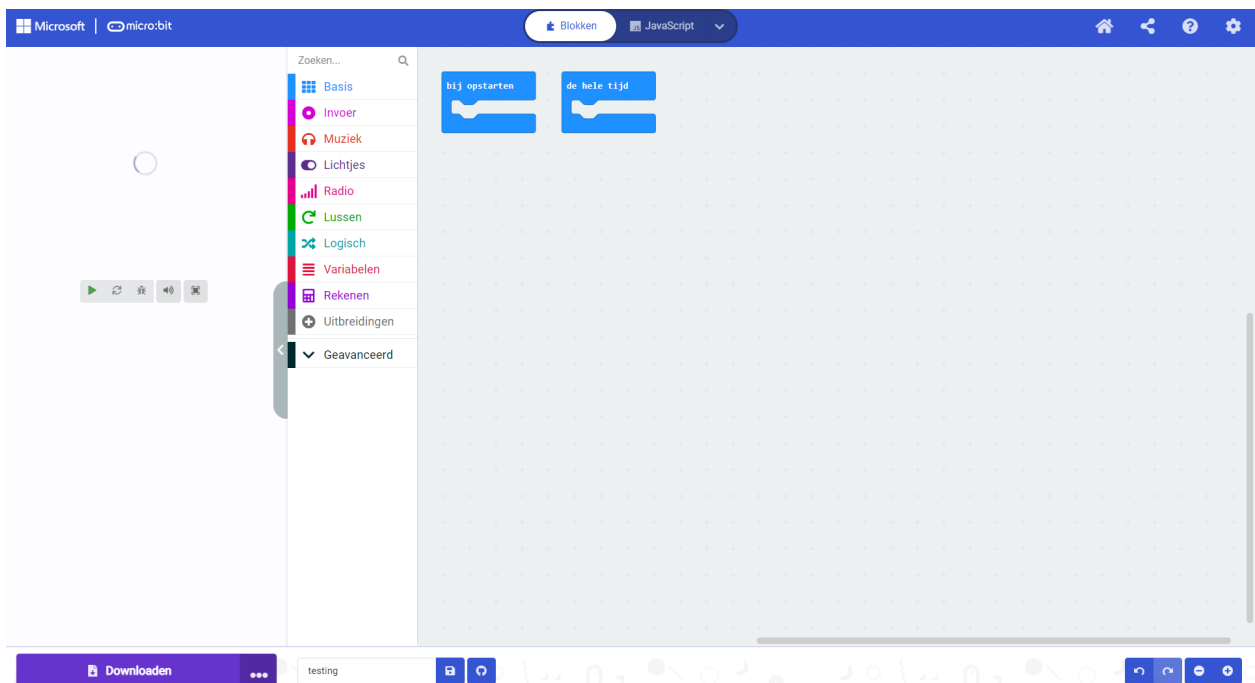
Aan de slag

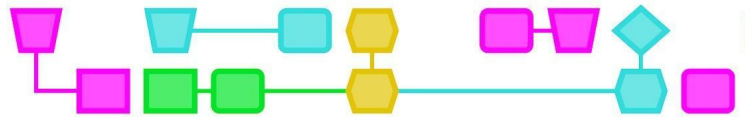
1. Pak een krokodillenbekje en sluit het aan het lange pootje (+) van het ledlampje. Sluit de andere kant aan op Pin0.
2. Pak het andere krokodillenbekje en sluit dit aan het korte pootje (-) van het ledlampje. Sluit de andere kant aan op GND.



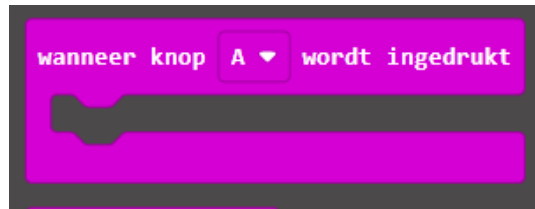
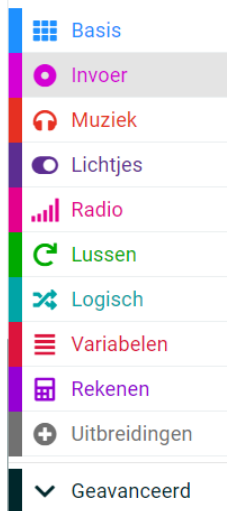
Het ledlampje programmeren

3. Sluit de micro:bit met het ledlampje aan op de computer.
4. Open MakeCode (<https://makecode.microbit.org/#>) en klik op 'Nieuw Project'.
5. Geef je project een naam en klik op 'Aanmaken'. Je krijgt nu dit scherm te zien.

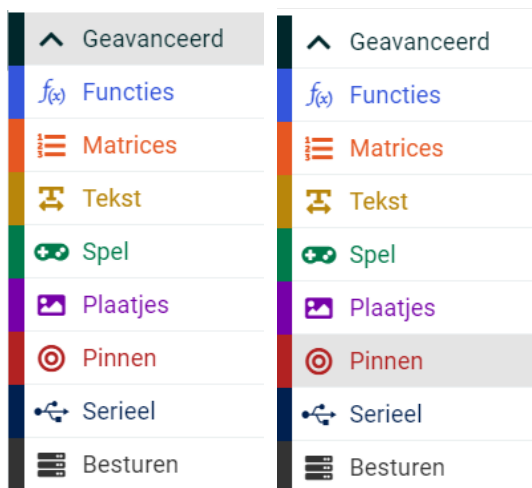




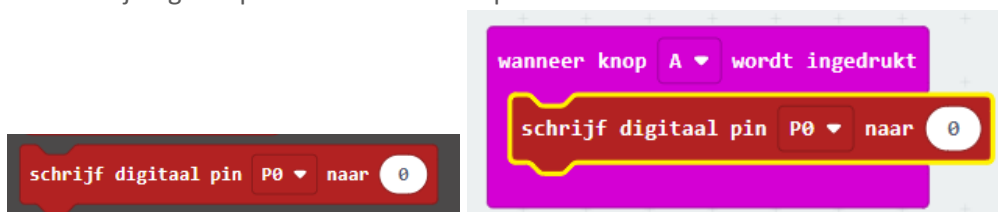
- Ga links naar 'Invoer' en kies 'Wanneer knop A wordt ingedrukt'. Sleep deze naar het werkveld in het midden van het scherm.

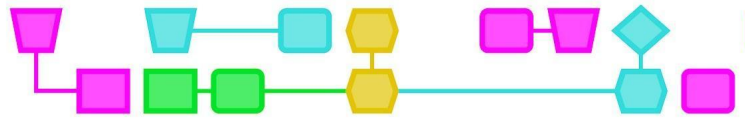


- Klik op 'Geavanceerd' en vervolgens op 'Pinnen'

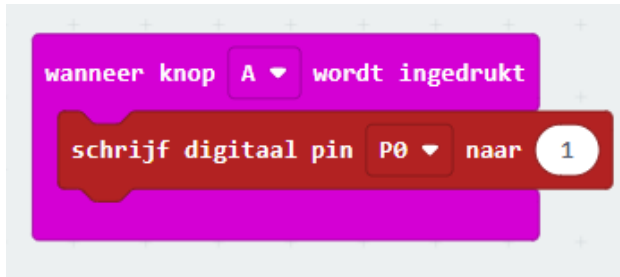


- Kies 'Schrijf digitaal pin P0 naar 0' en sleep deze het roze blok in.





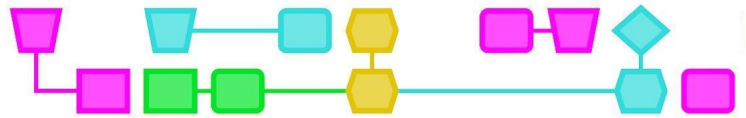
9. Klik op het witte veld met '0' erin en verander deze 0 in 1.



10. Download de micro:bit-code en kijk of je nu met knop A het lampje aan krijgt!
11. Probeer meerdere ledlampjes aan te sluiten. Elk lampje krijgt een krokodillenbekje naar Pin0, Pin1 of Pin2 en GND.

Let op!

De micro:bit heeft bij de derde pin '3V' staan. Sluit deze niet aan op GND! Als je dit wel doet kan de micro:bit teveel elektriciteit rond sturen en overbelast raken.



Werkblad - Maak een (mini)lichtshow met micro:bit - deel 1

In deze les ga je een (mini)lichtshow maken met behulp van de micro:bit. Om met dit werkblad aan de slag te kunnen gaan, moet je al weten hoe je een ledlampje aansluit op de micro:bit. Heb je dit zelf geprobeerd en is het gelukt? Dan kun je verder met dit werkblad.

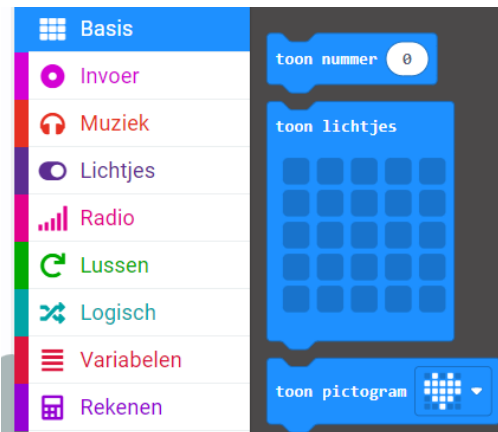
Wat heb je nodig?

- Een micro:bit
- Batterij
- USB-kabel
- Computer/laptop
- Ledlampjes
- Alligatorclips
- Materiaal voor de lichtshow

Experimenteer met de micro:bit

Je kunt een lampje laten branden, maar wat kan de micro:bit nog meer?

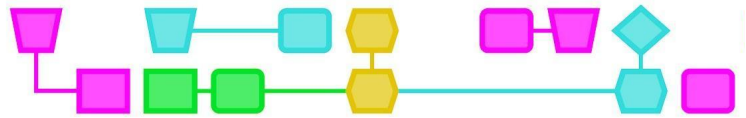
Kun je de micro:bit bijvoorbeeld een leuk pictogram laten weergeven? Dit kun je doen door bij 'Basis' op de blokken 'Toon lichtjes' of 'Toon pictogram' te klikken.



Kun je door de micro:bit te schudden dit pictogram ook weer wissen? Dit kun je doen door bij 'Invoer' op het blok 'bij schudden' te klikken, en bij 'Basis' het blok 'Wis scherm' erbij te zetten.

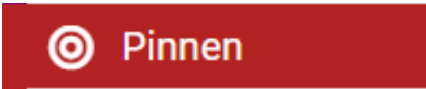


Ontdek wat je nog meer met de micro:bit kunt doen.



Experimenteer met de lampjes

Als je 'Pinnen' aanklikt, staan daar de blokken die de lampjes besturen. Wat kun je hier allemaal mee? Zijn er meer manieren om de lampjes te laten branden? Hoeveel lampjes kunnen er tegelijkertijd aan staan?



Er zijn nog andere opties onder '... meer', maar deze heb je niet nodig voor de lichtshow.

Experimenteer met materiaal

Welk materiaal kun je gebruiken voor de lichtshow? Wat zou een cool lichteffect creëren? Je kunt dit proberen met je geprogrammeerde ledlampje.

Beantwoord de volgende vragen:

1. Wat vinden jullie belangrijk voor de lichtshow? (Bijvoorbeeld: het lampje moet aan- en uitgaan, er moeten lichteffecten in zitten of het moet er mooi uitzien.)

.....

.....

.....

.....

.....

.....

.....

.....

.....

2. Wat is iets wat jullie graag willen verwerken in je lichtshow? (Denk aan een bepaald materiaal, een bepaald effect of een bepaalde sfeer.)

.....

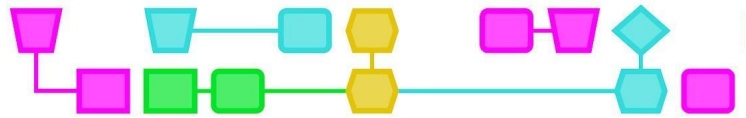
.....

.....

.....

.....

.....



.....
.....

Als iedereen klaar is met deze stap ga je samen met de klas de voorwaarden bepalen voor de lichtshow. Daarna krijg je een tweede werkblad.

Werkblad - Maak een (mini)lichtshow met micro:bit - deel 2

Je hebt samen met de klas bepaald waar de lichtshow aan moet voldoen. Met dit werkblad ga je de lichtshow plannen en uitvoeren.

Maak een plan voor je lichtshow

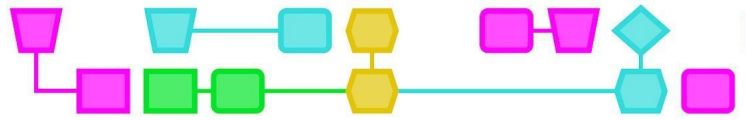
Gebruik de onderstaande vragen om te bepalen hoe jullie lichtshow eruit komt te zien.

1. Wat is het doel van de lichtshow? Als je dit met de klas hebt afgesproken heb je allemaal hetzelfde doel.

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

2. Hoe ga je dat doel bereiken? Denk aan lichteffecten en kleuren.

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



3. Welk materiaal heb je daarvoor nodig? Denk ook aan hoeveel lampjes je wilt gebruiken en of je bijvoorbeeld het scherm gaat gebruiken.

.....

.....

.....

.....

.....

.....

4. Hoelang wil je dat jouw lichtshow duurt?

.....

.....

.....

.....

.....

5. Hoe wil je de lichtshow presenteren? Denk aan op de muur, in een doos, onder de tafel of op een heel andere plek.

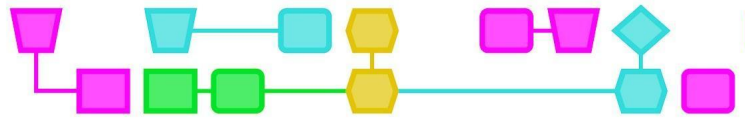
.....

.....

.....

.....

.....

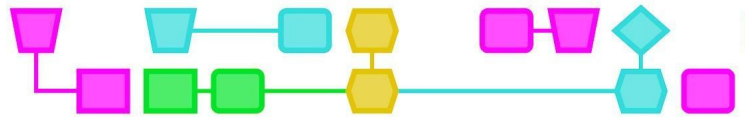


Programmeer en maak de lichtshow

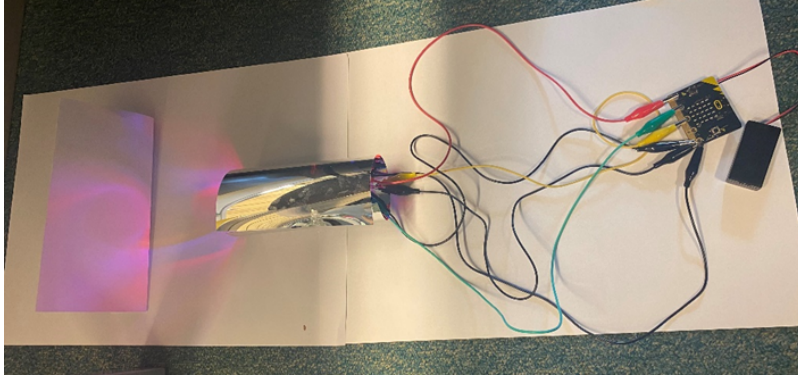
Ga in MakeCode aan de slag met het programmeren van je lichtshow. Zorg dat je tussendoor de micro:bit test, zodat je kunt zien of alle lampjes nog goed aangesloten zijn. Verzamel materiaal bij elkaar en ga aan de slag!

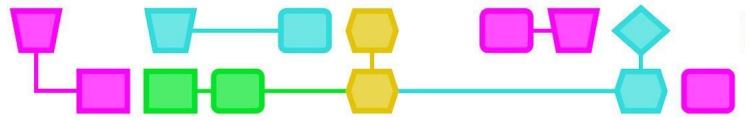
Presenteren!

Als de tijd om is, kun je je lichtshow presenteren aan de klas!



Bijlage 1: Voorbeeld lichtshows





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

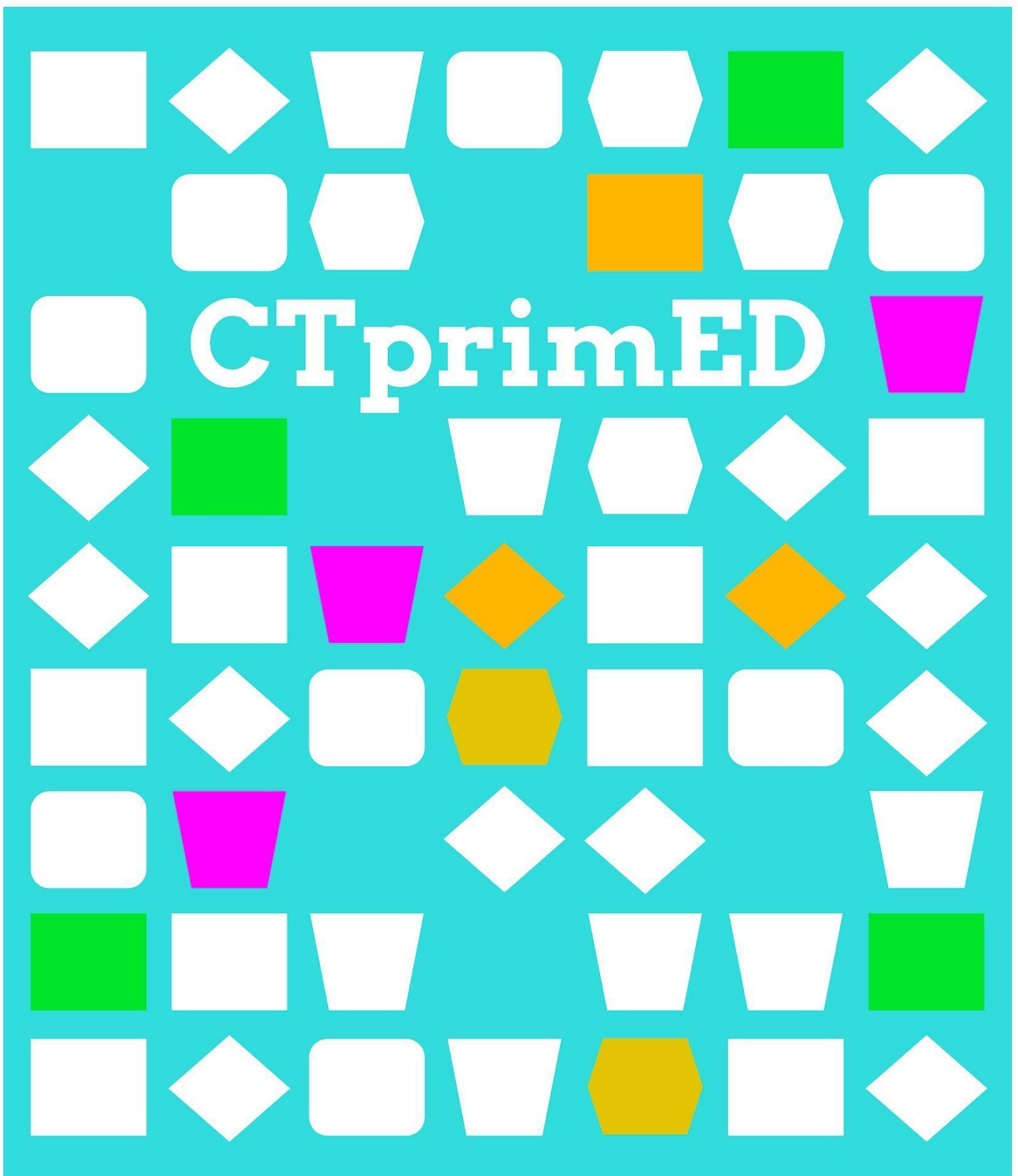


Co-funded by the
Erasmus+ Programme
of the European Union

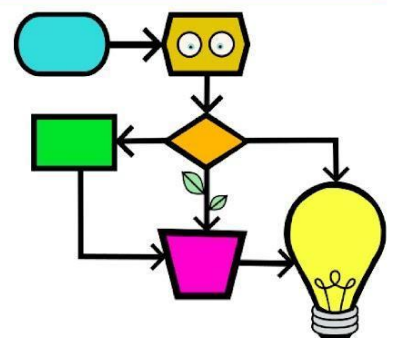


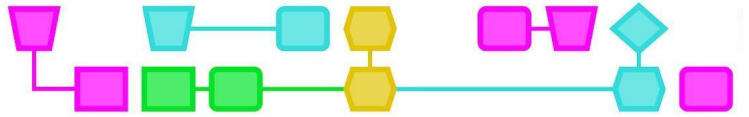
Universidad de Deusto
University of Deusto

Deusto



Maak een
(mini)lichtshow





Samenvatting

De leerlingen maken in deze workshop een (analoge) lichtshow. Ze experimenteren met divers materiaal en verschillende lichtbronnen om te onderzoeken hoe ze lichteffecten kunnen creëren. Daarna bepalen de leerlingen klassikaal de voorwaarden, waarna ze aan de slag gaan met het materiaal en hun eigen lichtshow maken. Als de lichtshow klaar is presenteren ze die aan de rest van de klas.

Doelgroep: 6-12 jaar.

Duur: 80 min

Deel 1: Introductie en materialen verkennen (25 min)

Deel 2: De lichtshow (55 min)

De delen kunnen als aparte lessen gegeven worden.

Leerdoelen:

Leerlingen leren

- voorwaarden voor een opdracht in groepsverband vast te stellen;
- hoe ze kunnen experimenteren met licht en hoe schaduw, lichtbreking en reflectie daarbij werken;
- hoe ze met behulp van pictogrammen een lichtshow kunnen programmeren;
- wat een algoritme is.

Online/offline: offline

Computational thinking: decompositie, abstractie en algoritmes

- Algemene vaardigheden: samenwerken, creatief denken, onderzoeken
- CT-foundations: -
- CT-concepten: programma, functie, loop, code

Bijzonderheden: Deze les moet in een (gedeeltelijk) verduisterde ruimte gegeven worden, zodat de lichtshows goed te zien zijn.

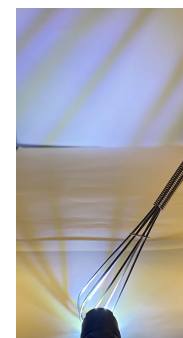
Materialen:

Onderstaande materialen zijn voorbeelden. Gebruik materialen die al in de klas aanwezig zijn, of materialen waar je niet zo snel aan zou denken. Zorg dat elk groepje voldoende materiaal heeft om te experimenteren.

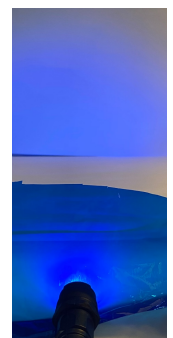
- Verschillende lichtbronnen (in ieder geval 1 lichtbron per leerling)
 - Zaklampen, fietslampen, oude lampen, ledstrips met afstandsbediening, laserlampjes
- Knutselmateriaal
 - Papier, karton, schaar, lijm, schilderstape, stiften, kleurpotloden, plakband, A3-papier
- Reflecterend materiaal (ong. 2-3 items per groepje)
 - Aluminiumfolie, reflecterend papier, glitterpapier, (oude) cd's, spiegels, discoballen, prisma's, lenzen, glazen, kristallen
- Materiaal om schaduwen mee te creëren (ong. 2-3 items per groepje)
 - Garde, vergiet, schuimspaan, poppetjes, crêpepapier



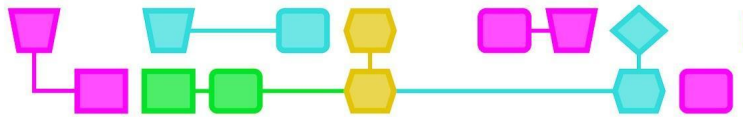
Reflectie van een glas



Schaduw van een garde

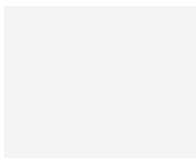


Blauw licht door cellofaan



- Lichtbreking en kleur (ong. 2-3 items per groepje)
 - Cellofaan, glasvezel, plexiglazen staafjes, gekleurd plastic
- Overig
 - NL CTPrimED: pictogrammen. Ongeveer 20 'aan', 15 'draaien' en 5 'lege' per groepje.
 - Als er te weinig ruimte is om leerlingen te laten projecten op de muur/het plafond kunnen ze dit ook op wit papier op de tafel doen. Zorg dan voor voldoende wit papier per groepje.

Tip: Gebruik gerecycled materiaal. Bijvoorbeeld oude 's, schoenendozen, gekleurd plastic, etc. Vraag de leerlingen dit materiaal van huis mee te nemen. Of vraag bij een *Makerspace* of ze bruikbaar materiaal/afval overhebben!



'Vrije invulling'



'Lampje aan'

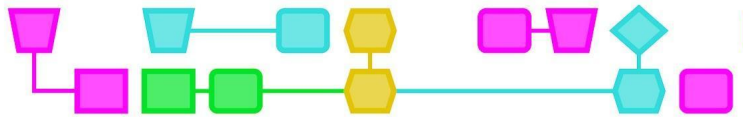


'Lampje draaien'

Vorbereiding

Verzamel de benodigde materialen. Print voor elk groepje een rooster en voldoende pictogrammen (NL CTPrimED: Pictogrammen). Als de leerlingen goed kunnen knippen, kun je ervoor kiezen om ze de pictogrammen zelf uit te laten knippen. Maak een voorbeeldrooster dat je kan gebruiken bij de introductie, zie voor een voorbeeld bladzijde 4.

Tip: Zie voor een digitale versie van deze les: 'Maak een (mini)lichtshow (met micro:bit)'.



Deel 1: Introductie en materialen verkennen (25 min)

Introductie (10 min)

Leg de leerlingen het volgende uit:

Vandaag gaan we een lichtshow 'programmeren', alleen doen we dat niet met computers maar op papier. De lichtshow maak je in groepjes met verschillende lampjes en materiaal. We doen het eerst een keer met de hele klas.

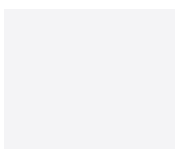
Klassikaal lichtshow programmeren

Verdeel de klas in vier groepen. Geef iedere leerling een lampje. Oefen met de leerlingen zodat ze het lampje snel aan en uit kunnen zetten. Mocht dit lastig gaan/niet snel genoeg kunnen, dan mogen de leerlingen hun hand op het lampje leggen om het licht te dimmen.

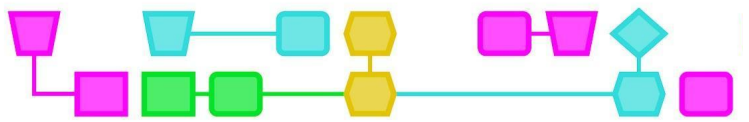
Laat het voorbeeld programmeerrooster zien. Leg uit:

- Ieder cijfer bovenin staat voor een seconde/tel.
- De regels zijn de vier groepen, oftewel vier lampjes.
- Het pictogram 'Lampje aan', betekent dat de lamp aan moet staan.
- Een leeg vlak betekent dat de lamp uitgezet moet worden.
- Een spiraal betekent dat het lampje gedraaid moet worden.
- Groepje 1 leest dus regel 1: lamp aan, lamp aan, lamp uit, lamp uit, lamp aan, lamp aan, etc.

| | | Actie + seconde | | | | | | | |
|------------|--------|-----------------|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| Taak/kleur | Lampje | | | | | | | | |
| groen | 1 | | | | | | | | |
| oranje | 2 | | | | | | | | |
| roos | 3 | | | | | | | | |
| paars | 4 | | | | | | | | |



'Vrije invulling' 'Lampje aan' 'Lampje draaien'



Als het voor iedereen duidelijk is, tel je een paar keer achter elkaar van 1 tot 8.

Leg uit dat dit rooster eigenlijk een programma is. Dit wordt nu door een menselijke 'computer' uitgevoerd, maar zou ook door een echte computer uitgevoerd kunnen worden. De pictogrammen zijn functies: een blokje code dat staat voor een taak. De computer kan dit aflezen en weet dan wat die moet doen en wanneer.

De leerlingen hebben de lampjes nodig voor de volgende opdracht.

Tip: Voor leerlingen die al wat verder zijn met coderen kun je het rooster weglaten. Laat ze zelf bedenken hoe ze met de pictogrammen de computer zouden kunnen programmeren. Ze kunnen ook gebruikmaken van variabelen (als...dan) en loops.

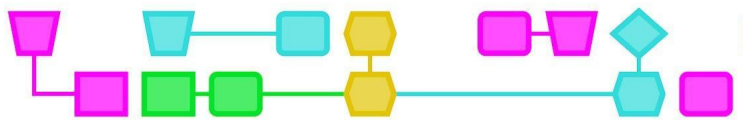
Experimenteer en onderzoek (15 min)

Geef aan dat de leerlingen nu mogen experimenteren met materiaal. Deel de klas op in groepjes van 2-4 leerlingen en laat ze in groepsverband experimenteren met materiaal, kleur en lichtbronnen. Ze kunnen alle verschillende lichtbronnen uitproberen, kijken hoe ze verschillende kleuren kunnen maken en welk materiaal interessante lichteffecten creëert.

Tip: Zet een digitale visuele timer op het bord, bijvoorbeeld een cirkel. Zo kunnen leerlingen zelf de tijd in de gaten houden.

Als de tijd om is, moeten de leerlingen gaan zitten. Vraag naar hun ervaringen: welk materiaal liet goed licht door en welk materiaal juist niet? Hebben ze materiaal gevonden waarmee ze leuke effecten kunnen creëren of materiaal dat het licht een andere kleur geeft? Wijs een paar leerlingen aan die een voorbeeld mogen laten zien.

Tip: Als leerlingen moeite hebben met experimenteren, kun je de voorbeelden uit bijlage 1 laten zien. Laat de leerlingen daarna nog 5-10 minuten experimenteren.



Deel 2: De lichtshow (55 min)

Voorwaarden bepalen (5 min)

Vertel de leerlingen dat ze klassikaal de voorwaarden voor de lichtshow gaan bepalen. Voorwaarden zijn eisen waar de lichtshow aan moet voldoen, bijvoorbeeld welk doel de lichtshow heeft, hoelang hij moet duren en hoeveel lichteffecten erin moeten zitten. Vraag de leerlingen welke voorwaarden ze kunnen bedenken. Laat ze een voor een dingen opnoemen.

Vragen om leerlingen op weg te helpen:

- Wat kan het doel van de lichtshow zijn? Bijvoorbeeld:
 - Een verhaal vertellen
 - Een cool TikTok-filmpje maken op het ritme van een (zelfgekozen) liedje
 - Een bepaalde omgeving namaken (bijvoorbeeld een bos, zee of strand)
 - Een sfeer neerzetten (een enge, sprookjesachtige of relaxte sfeer)
- Waar moet de lichtshow aan voldoen?
 - Welke kleuren moet je gebruiken?
 - Hoe vaak moeten de 8 tellen herhaald worden?
 - Hoeveel lampjes moet je minimaal gebruiken?
 - Welk materiaal moet gebruikt worden?
 - In welke ruimte of op welke plek vindt de lichtshow plaats?
 - Denk hierbij aan projecteren op het plafond, een schaduwspel maken, een discobal gebruiken, enz. Voorbeelden van de lichtshow kun je vinden in Bijlage 2.

Als het grootste gedeelte van de klas het eens is met de voorwaarden, kun je deze op het bord schrijven. Zorg dat er een duidelijke (en korte) lijst voorwaarden is voor de leerlingen.

Plan maken en uitvoeren (30 min)

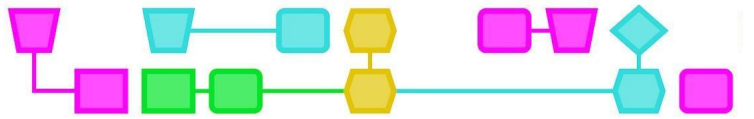
Deel de pictogrammen uit en zorg dat iedere groep een A3-rooster en lijm heeft. Vertel dat de leerlingen de lichtshow gaan bedenken, plannen en uitvoeren. Op het rooster kunnen ze door middel van de pictogrammen de verschillende commando's voor de lampjes plakken. De leerlingen kunnen in het taak-/kleurvak ook nog speciale instructies zetten. Zorg dat de leerlingen de voorwaarden niet vergeten door deze op het bord te laten staan. Ondersteun de leerlingen door rond te lopen en mee te denken over materiaal en kleur als ze er zelf niet uit komen.

Geef na 15 minuten aan dat de leerlingen op de helft van de tijd zitten en dat ze moeten beginnen met het programmeren van de pictogrammen.

Tip: Zet bij deze stap ook een visuele digitale timer op het bord. Dit helpt de leerlingen met inschatten hoelang ze nog hebben.

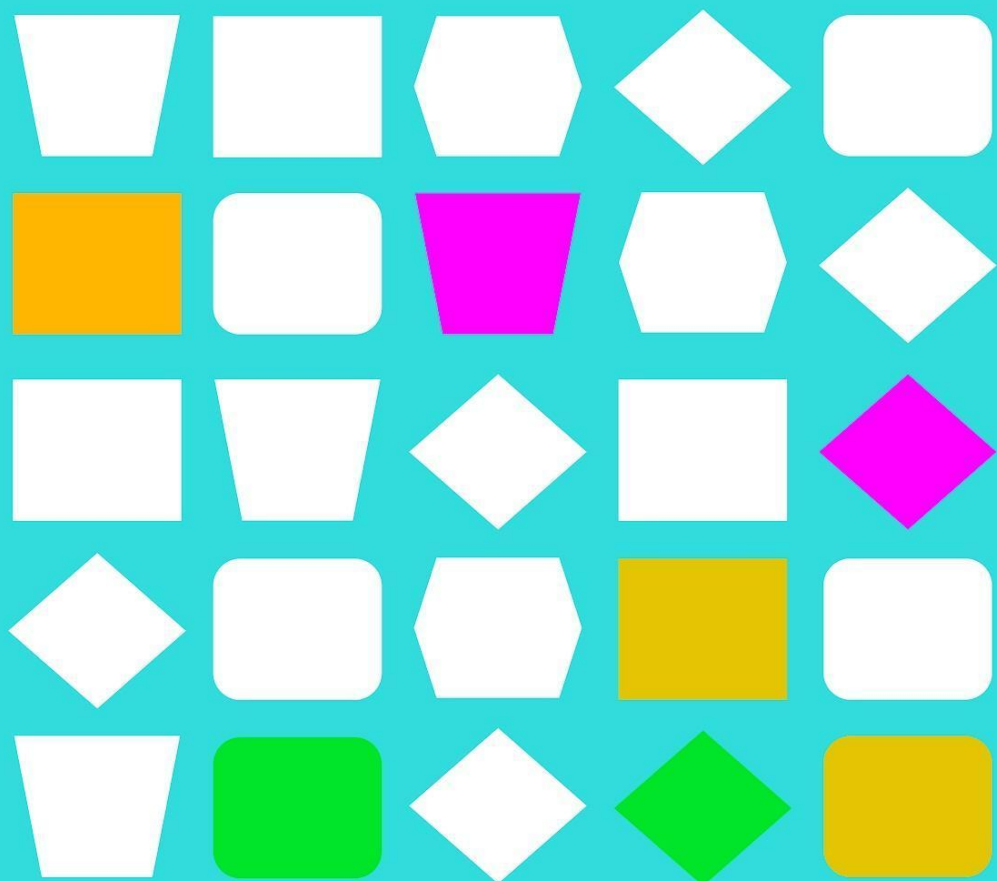
Het presenteren van de lichtshow (15 min)

Besprek klassikaal hoe het programmeren is gegaan. Wat vonden ze leuk om te doen en waar liepen ze tegenaan? Laat de groepjes daarna een voor een hun lichtshow presenteren. Vraag ze eerst even het plan te laten zien en daarna om het plan uit te voeren. Vraag hoe ze het maken en programmeren van de lichtshow hebben aangepakt en waarom. Vraag ook waar ze trots op zijn.

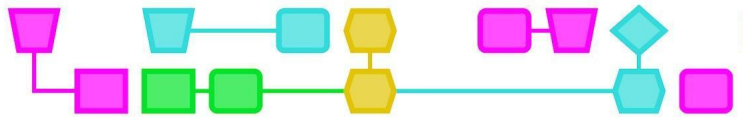


Afsluiting van de les (5 min)

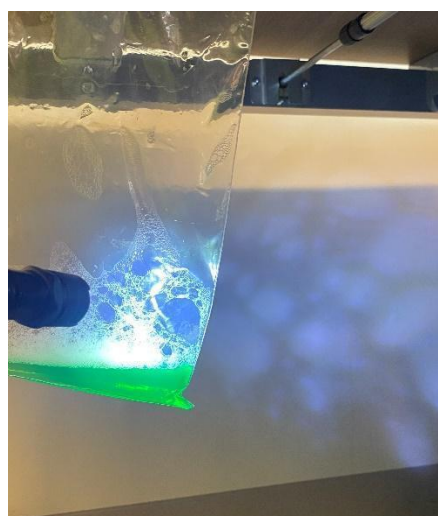
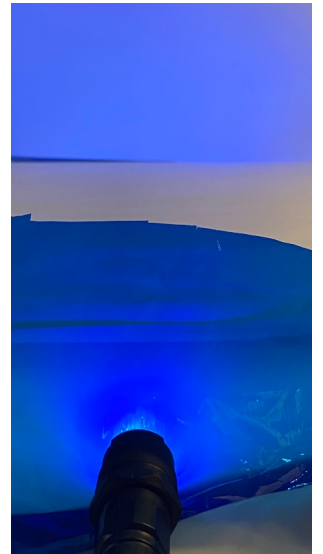
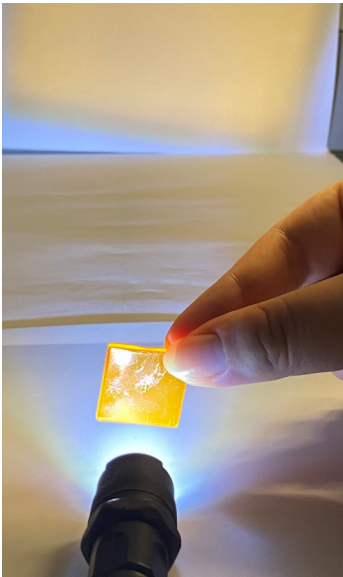
Laat de leerlingen opruimen en bespreek de les na. Benadruk dat iedereen dezelfde opdracht kreeg maar er toch heel diverse lichtshows uit zijn gekomen! Vertel dat de leerlingen met behulp van de pictogrammen een programma hebben geschreven.

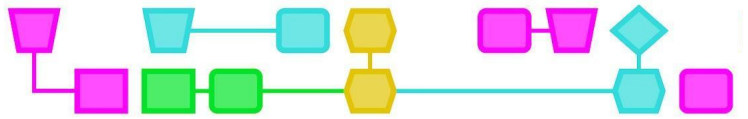


Bijlagen



Bijlage 1: Voorbeelden met materialen





Bijlage 2: Voorbeelden van lichtshows

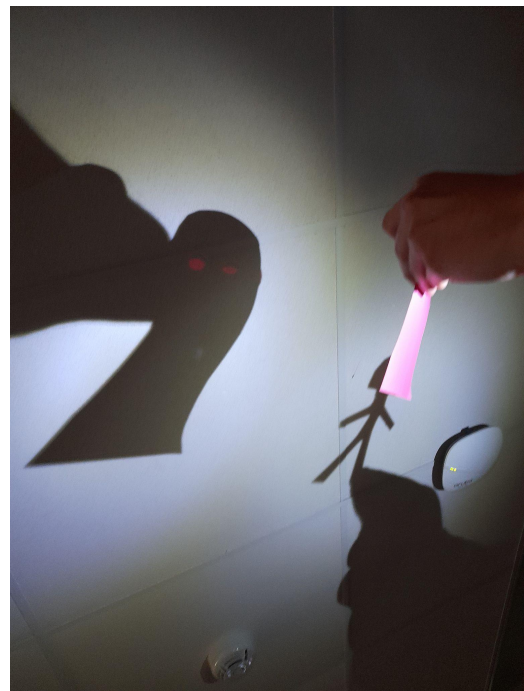


Eisen: Meerdere kleuren, verder vrijgelaten

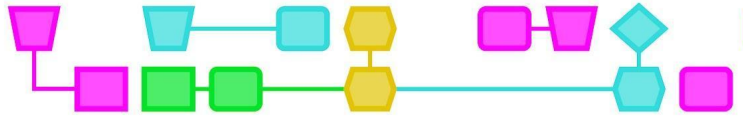
Eisen: Meerdere kleuren, verder vrijgelaten



Eisen: Op de muur, sprookje



Eisen: Op het plafond, horror



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

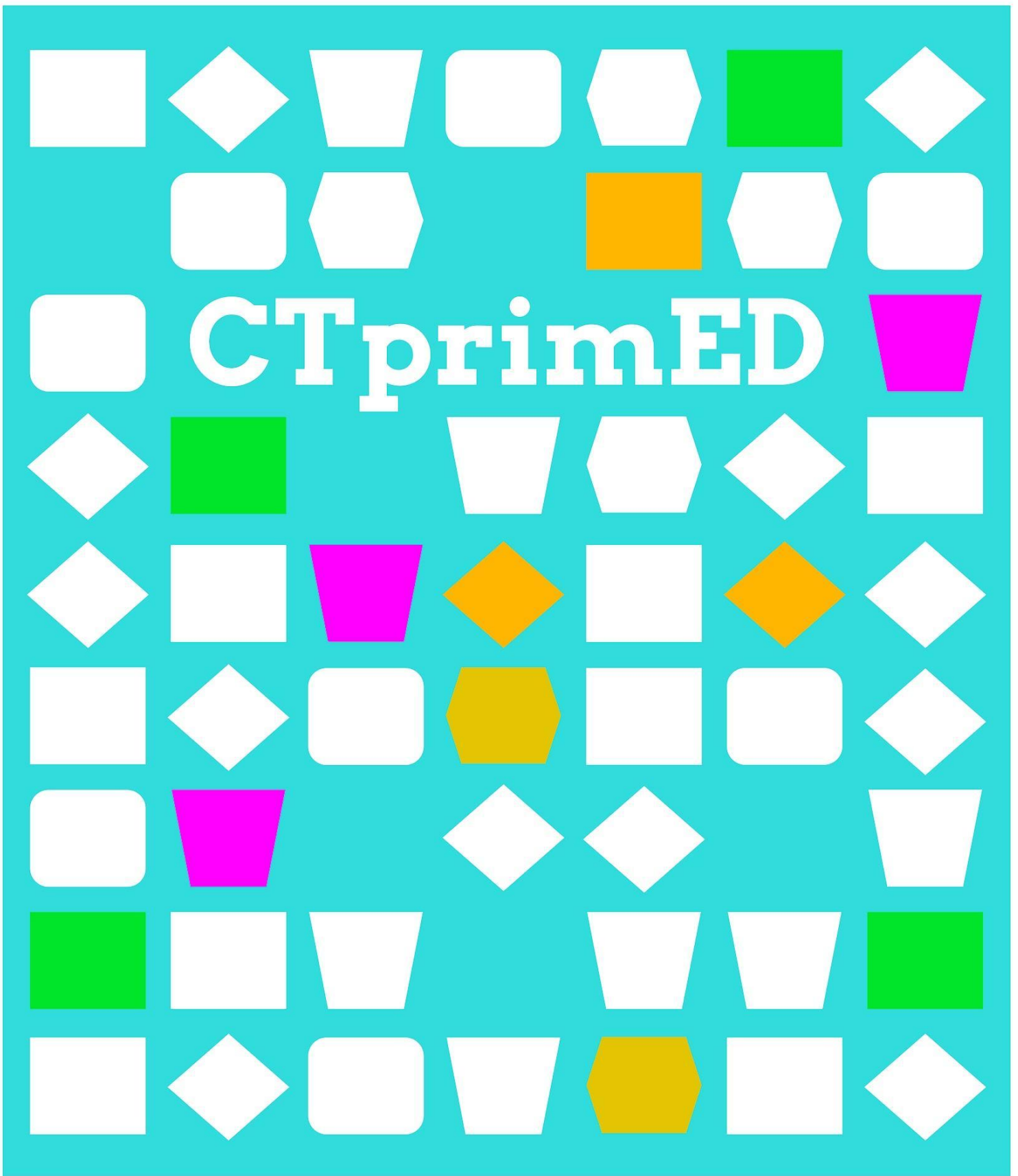


Co-funded by the
Erasmus+ Programme
of the European Union

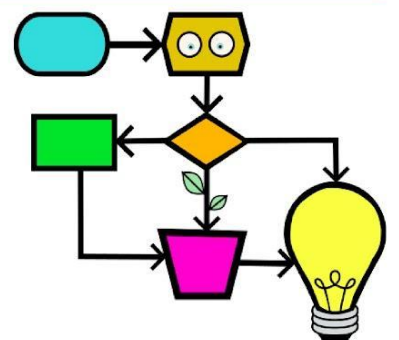


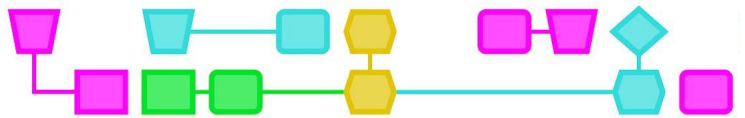
Universidad de Deusto
University of Deusto

Deusto



Micro:bit handleiding



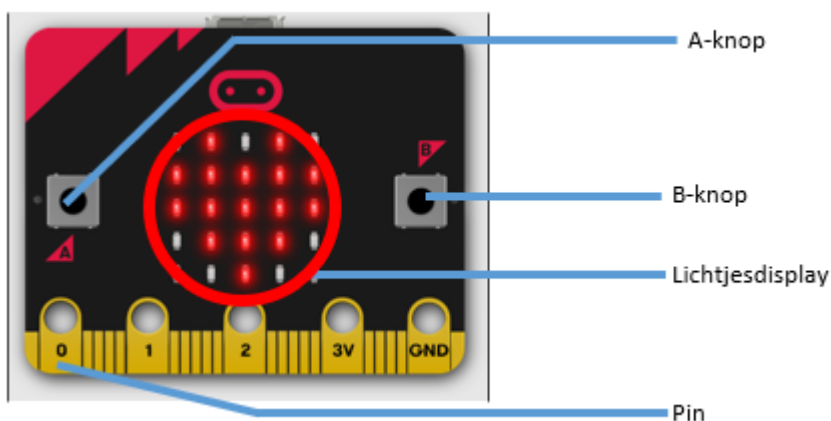


De micro:bit

Een micro:bit is een klein computertje met een display van 5 x 5 ledlampjes, waarop je boodschappen en icoontjes kunt programmeren (zie de afbeelding). Met knopjes en een bewegingssensor kan de micro:bit interactie hebben met de omgeving. Bijvoorbeeld: na het indrukken van de A-knop laat de micro:bit een icoon zien op het lichtjesdisplay. Daarnaast kun je er onderdelen op aansluiten die worden aangestuurd door de micro:bit, zoals een ledlampje of een minispeaker. Om de micro:bit te programmeren, bestaan er laagdrempelige online programma's, zoals Scratch en MakeCode.

Toelichting per onderdeel

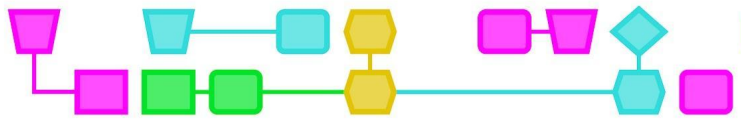
- Lichtjesdisplay: 25 ledlampjes vormen samen een display dat je kunt gebruiken om letters, cijfers en iconen weer te geven.
- A- en B-knop: twee knoppen die je apart of tegelijkertijd kunt indrukken. De micro:bit reageert hierop, bijvoorbeeld door iets te laten zien op het lichtjesdisplay.
- Pins: de pins 0, 1, 2 en GND kun je onder andere gebruiken om onderdelen op aan te sluiten, zoals een los ledlampje of een speaker. De 3V-pin is bedoeld om een extra spanningsbron aan te sluiten, maar dat is bij deze activiteiten niet nodig.
- Bewegingssensor (niet zichtbaar op afbeelding): dankzij de bewegingssensor kan de micro:bit reageren op schudden en andere bewegingen.



De micro:bit heeft spanning nodig om te kunnen werken. Zolang die is aangesloten op de laptop, ontvangt de micro:bit spanning van de computer. Als je de micro:bit los van de laptop wil gebruiken, kun je twee AAA-batterijen met de bijgeleverde batterijhouder aansluiten op de spanningspoort van de micro:bit.

De micro:bit aansluiten

- Sluit de micro:bit met het bijgeleverde kabeltje aan op de USB-poort van de laptop. Het oranje lampje op de achterkant van de micro:bit gaat knipperen als deze goed is aangesloten.
- Ga naar <https://makecode.microbit.org/>.
- Verander zo nodig de schermtaal in Nederlands: klik in de rechterbovenhoek op het icoon voor instellingen en kies in het menu bij Taal (Language) voor Nederlands.
- Klik op Nieuw project, geef het project een naam en klik op Aanmaken.
- Je komt in een nieuw scherm terecht. Klik linksonder op de drie puntjes naast de knop Downloaden.
- Kies in het menu voor Connect device, en daarna tweemaal op Volgende.
- Klik in de pop-up op BBC micro:bit CMSIS-DAP of DAPLink CMSIS-DAP en kies Verbinding maken.
- De micro:bit is nu goed aangesloten! Zo lang de micro:bit met het kabeltje verbonden blijft aan de laptop, kun je met één klik op de knop Downloaden nieuwe informatie op de micro:bit zetten.



Onderdelen aansluiten op de micro:bit

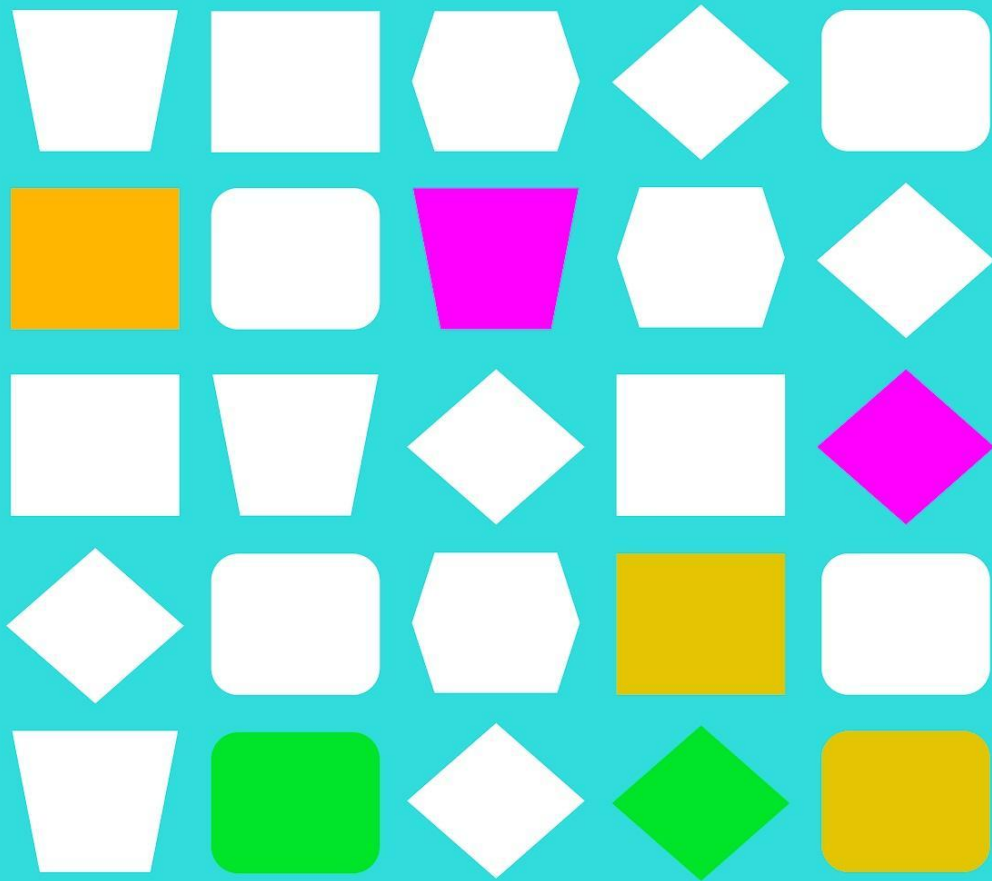
Om een ledlampje en minispeaker aan te sluiten gebruik je stroomdraden met een krokodillenbek. De kleur van de stroomdraden maakt op zich niet uit, maar in de werkbladen worden een rode en een zwarte stroomdraad genoemd. Dit is gedaan om de instructie voor de leerlingen duidelijker te maken. Om verwarring te voorkomen kun je ervoor kiezen om tijdens de lessen ook alleen gebruik te maken van zwarte en rode stroomdraden.

Let er bij het vastmaken van de krokodillenbek op dat deze niet over het gat in de pins van de micro:bit uitsteekt. Als er drie willekeurige lichtjes op het display van de micro:bit oplichten, is de krokodillenbek niet goed aangesloten.

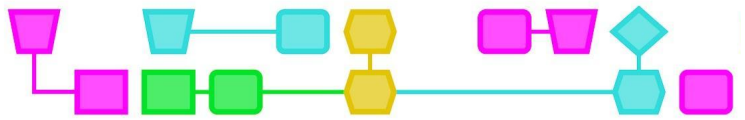
Ledlampje

Een ledlampje laat de stroom maar in één richting door. Als het ledlampje niet brandt, draai het dan om. Doe voorzichtig met de pootjes van de ledlampjes, want die kunnen afbreken als ze vaak worden gebogen.

1

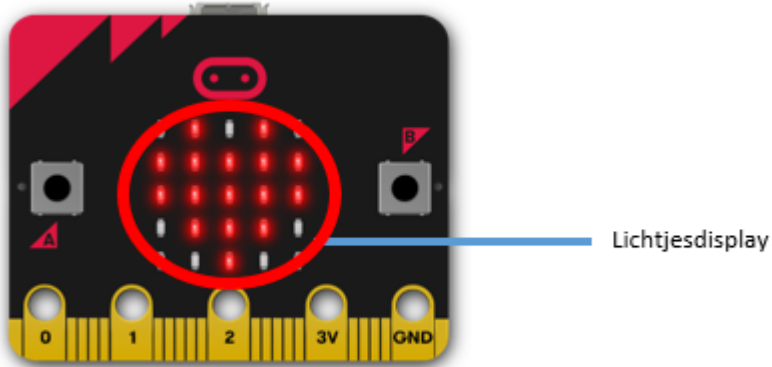


Werkbladen



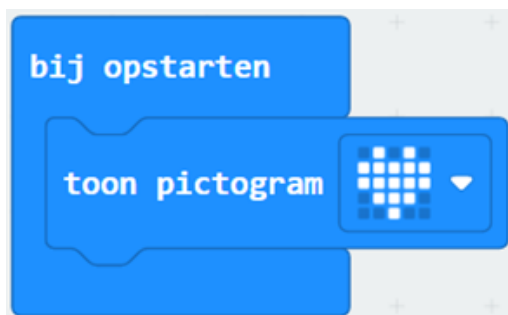
Werkblad 1: het lichtjesdisplay op de micro:bit besturen

Dit is het belangrijkste onderdeel dat je bij deze opdracht gebruikt:



Opdracht 1: Een pictogram laten zien

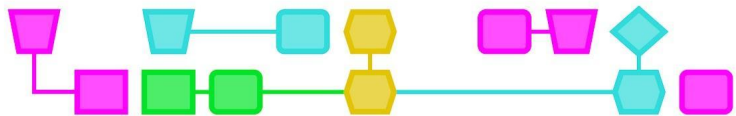
1. Staan er nog blokken in het werkgebied? Klik dan met je rechtermuisknop ergens in een leeg deel van het werkgebied en kies Alle blokken verwijderen.
2. Klik op **Basis** in het menu.
3. Kies het **bij opstarten**-blok en sleep dat naar het werkgebied.
4. Kies het **toon pictogram**-blok en sleep dat in het **bij opstarten**-blok.
Het zou er nu zo uit moeten zien:



Tip

Als je op het pijltje naast het hartpictogram klikt, kun je een ander pictogram kiezen.

5. Klik op **Downloaden** om je code op de micro:bit te zetten. Zie je het pictogram op de micro:bit verschijnen?



Opdracht 2: Zelf een pictogram maken

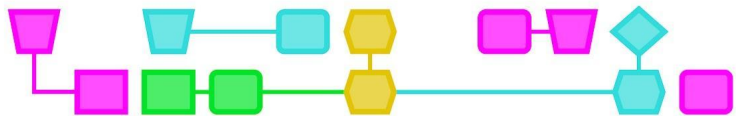
1. Sleep het **toon pictogram**-blok uit het **bij opstarten**-blok.
2. Klik op **Basis** in het menu, kies het **de hele tijd**-blok en sleep dat naar het werkgebied.
3. Kies bij **Basis** het **toon lichtjes**-blok en sleep dat in het **de hele tijd**-blok.

Het zou er nu zo uit moeten zien



In het **toon lichtjes**-blok kun je zelf een pictogram maken door lichtjes aan te zetten of uit te laten. Een lichtje staat aan als het blokje wit is en uit als het blokje donkerblauw is. Door op het blokje te klikken, kun je het lichtje aan- of uitzetten.

4. Maak zelf een pictogram en **download** je code op de micro:bit. Zie je je zelfgemaakte pictogram verschijnen?



Opdracht 3: Een tekst laten zien op de micro:bit

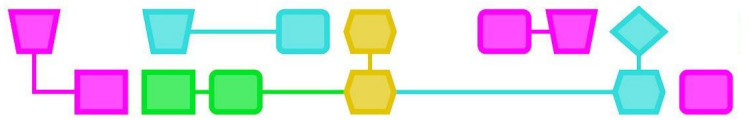
1. Sleep het **toon lichtjes**-blok uit het **de hele tijd**-blok.
2. Kies bij **Basis** het blok **toon tekens** en sleep dat in het **de hele tijd**-blok.
3. Het zou er nu zo uit moeten zien:



4. Klik op "Hello!" en typ hier je eigen tekst.
5. **Download** je code op de micro:bit. Zie je jouw tekst verschijnen?

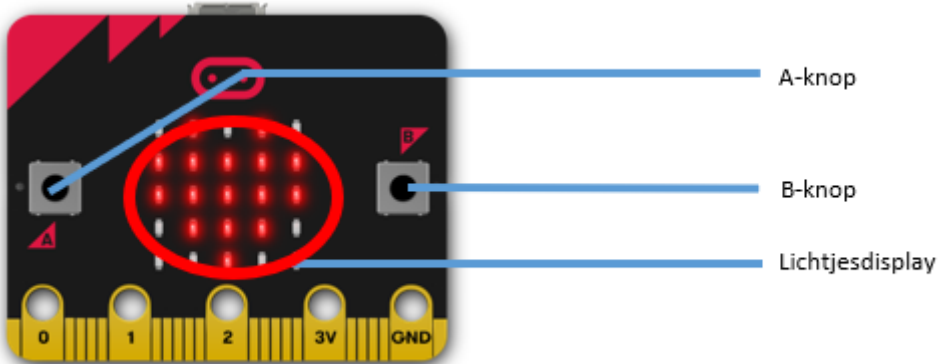
Opdracht 4: Een tekst en pictogram tegelijk laten zien

Je kunt ook meerdere blokken in het **bij opstarten**- of **de hele tijd**-blok zetten. Probeer bijvoorbeeld wat er gebeurt als je er tegelijk een **toon lichtjes**-blok en een **toon tekens**-blok in zet. Vergeet niet steeds je nieuwe code op de micro:bit te **downloaden**!



Werkblad 2: De micro:bit op je laten reageren

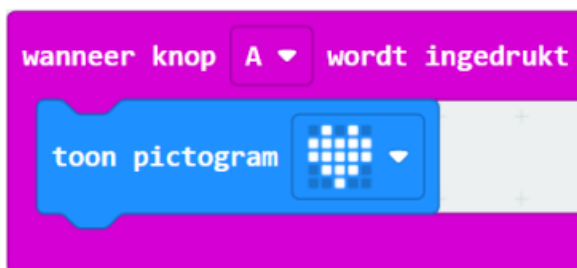
Dit zijn de belangrijkste onderdelen die je bij deze opdracht gebruikt:



Opdracht 1: Iets doen als je de A-knop, B-knop of beide tegelijk indrukt

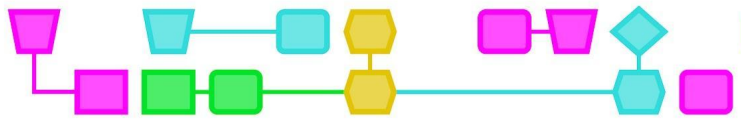
1. Staan er nog blokken in het werkgebied? Klik dan met je rechtermuisknop ergens in een leeg deel van het werkgebied en kies Alle blokken verwijderen.
2. Klik op **Invoer** in het menu.
3. Kies het blok **wanneer knop A wordt ingedrukt** en sleep het naar het werkgebied.
4. Klik op **Basis** in het menu en kies een blok uit, bijvoorbeeld **toon lichtjes**, **toon pictogram** of **toon tekens**. Sleep het blok van jouw keuze in het blok **wanneer knop A wordt ingedrukt**.

Het zou er nu zo uit kunnen zien:



Tip Als je klikt op het pijltje naast A in het **wanneer knop A wordt ingedrukt**-blok, kun je B of A + B kiezen.

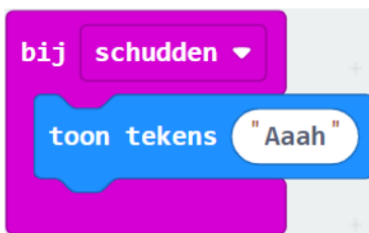
5. Klik op **Downloaden** om je code op de micro:bit te zetten.
6. Druk op de A-knop. Wat zie je gebeuren?
7. Programmeer de micro:bit nu zelf, zodat er iets gebeurt als je de B-knop indrukt of als je de A- en B-knoppen tegelijk indrukt.



Opdracht 2: Iets doen wanneer de micro:bit door elkaar wordt geschud

1. Klik met je rechtermuisknop ergens in een leeg deel van het werkgebied en kies Alle blokken verwijderen.
2. Klik op **Invoer** in het menu.
3. Kies het blok **bij schudden** en sleep het naar het werkgebied.
4. Klik op **Basis** in het menu en kies een blok uit, bijvoorbeeld een **toon lichtjes**, **toon pictogram** of **toon tekens**-blok. Sleep het blok van jouw keuze in het **bij schudden**-blok.

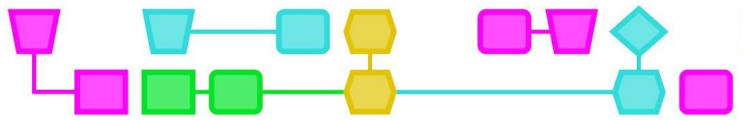
Het zou er nu zo uit kunnen zien:



Tip

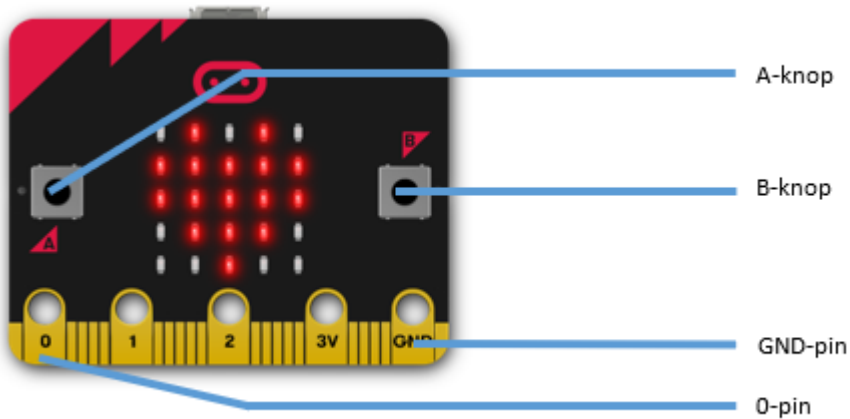
Wil je meer uitleg over hoe de **Invoer** werkt? Die vind je in werkblad 1 over hoe je het lichtjesdisplay op de micro:bit bestuurt.

5. **Download** je code op de micro:bit.
6. Schud de micro:bit goed door elkaar. Wat zie je gebeuren?



Werkblad 3: Een los lampje aan- en uitzetten en laten knipperen

Dit zijn de belangrijkste onderdelen die je bij deze opdracht gebruikt:



Vorbereiding

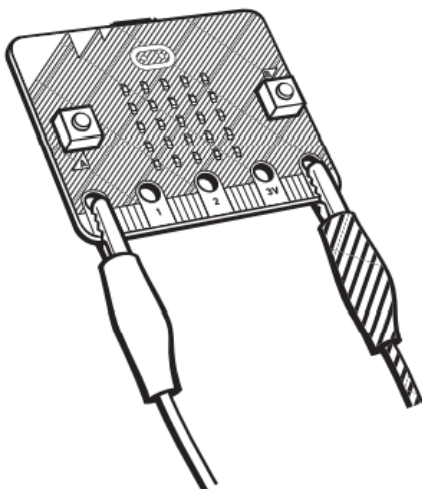
Voor de opdrachten in dit werkblad heb je deze materialen nodig:

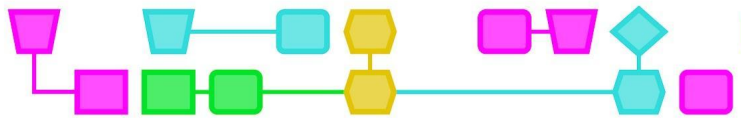
- Twee stroomdraden met krokodillenbek
- Een ledlampje

Opdracht 1: Het ledlampje aan- en uitzetten

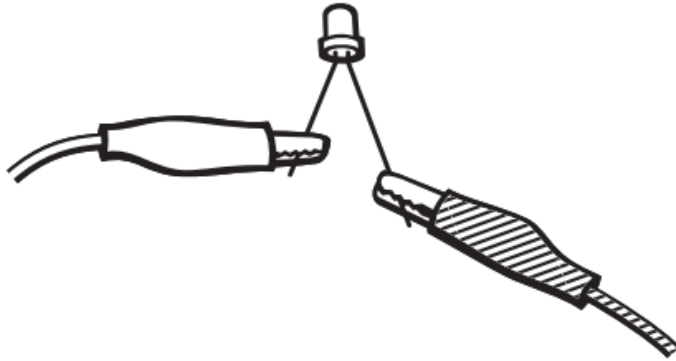
1. Maak één kant van de zwarte stroomdraad vast aan de GND-pin.
2. Maak één kant van de rode stroomdraad vast aan de 0-pin.

Het zou er nu zo uit moeten zien:

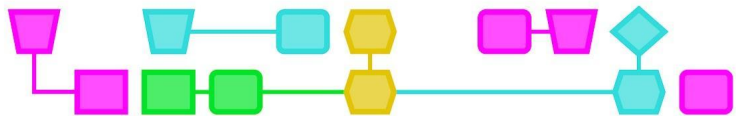




3. Klem de andere bek van de zwarte stroomdraad vast aan het korte pootje van het ledlampje.
4. Klem de andere bek van de rode stroomdraad vast aan het lange pootje van het ledlampje.
5. Het zou er zo uit moeten zien:



6. Staan er nog blokken in het werkgebied? Klik dan met je rechtermuisknop ergens in een leeg deel van het werkgebied en kies Alle blokken verwijderen.
7. Klik op **Invoer** in het menu.
8. Kies het blok **wanneer knop A wordt ingedrukt** en sleep het naar het werkgebied.
9. Sleep nog een **wanneer knop A wordt ingedrukt**-blok naar het werkgebied. Klik op het pijltje naast A en kies hier B.
10. Klik op **Geavanceerd** in het menu. Je ziet nog meer menu onderdelen verschijnen.
11. Klik op **Pinnen** in het menu en sleep het **schrijf digitaal pin P0 naar**-blok in het blok **wanneer knop A wordt ingedrukt**. Klik met je muis op 0 in het blok en verander dat getal in 1.
12. Klik op **Pinnen** in het menu en sleep het **schrijf digitaal pin P0 naar**-blok in het blok **wanneer knop B wordt ingedrukt**.

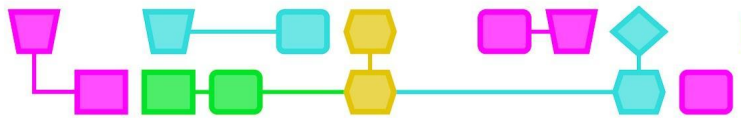


Het zou er nu zo uit moeten zien:

```
wanneer knop A ▼ wordt ingedrukt
  schrijf digitaal pin P0 ▼ naar 1

wanneer knop B ▼ wordt ingedrukt
  schrijf digitaal pin P0 ▼ naar 0
```

12. Klik op **Downloaden** om je code op de micro:bit te zetten.
13. Druk op de A-knop. Gaat het ledlampje aan?
14. Druk op de B-knop. Gaat het ledlampje weer uit?



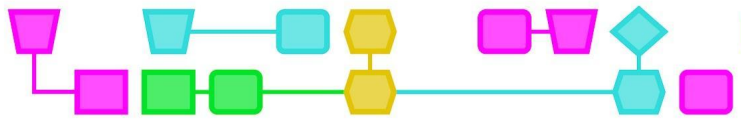
Opdracht 2: Het ledlampje laten knipperen

1. Klik met je rechtermuisknop ergens in een leeg deel van het werkgebied en kies Alle blokken verwijderen.
2. Klik op **Basis** in het menu.
3. Kies het blok **de hele tijd** en sleep het naar het werkgebied.
4. Klik op **Pinnen** in het menu en sleep het **schrijf digitaal pin P0 naar**-blok in het blok **de hele tijd**. Klik op 0 in het blok en verander dat getal in 1.
5. Klik op **Basis** in het menu.
6. Kies het blok **pauzeer (ms)** en sleep het onder het **schrijf digitaal pin P0 naar**-blok in het blok **de hele tijd**. Verander 100 in 1000.
7. Klik op **Pinnen** in het menu en sleep het **schrijf digitaal pin P0 naar**-blok in het blok **de hele tijd**.
8. Klik op **Basis** in het menu, kies het blok **pauzeer (ms)** en sleep het onder het tweede **schrijf digitaal pin P0 naar**-blok. Verander 100 in 1000.

Het zou er zo uit moeten zien:



9. **Download** je code op de micro:bit.
10. Het ledlampje gaat meteen knipperen! Wat gebeurt er als je het getal 1000 in een van de **pauzeer (ms)**-blokken of in beide **pauzeer (ms)**-blokken verandert in 100? En wat gebeurt er als je het getal verandert in 10.000? Vergeet niet steeds je code te **downloaden** op de micro:bit!



Colofon

Dit document is eerder ontwikkeld in het kader van het project Creativiteit als startpunt voor de verbinding van artistiek vermogen en technologie, mede mogelijk gemaakt door het Fonds voor cultuurparticipatie. De illustraties zijn gemaakt door Henk Stolker.

© 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

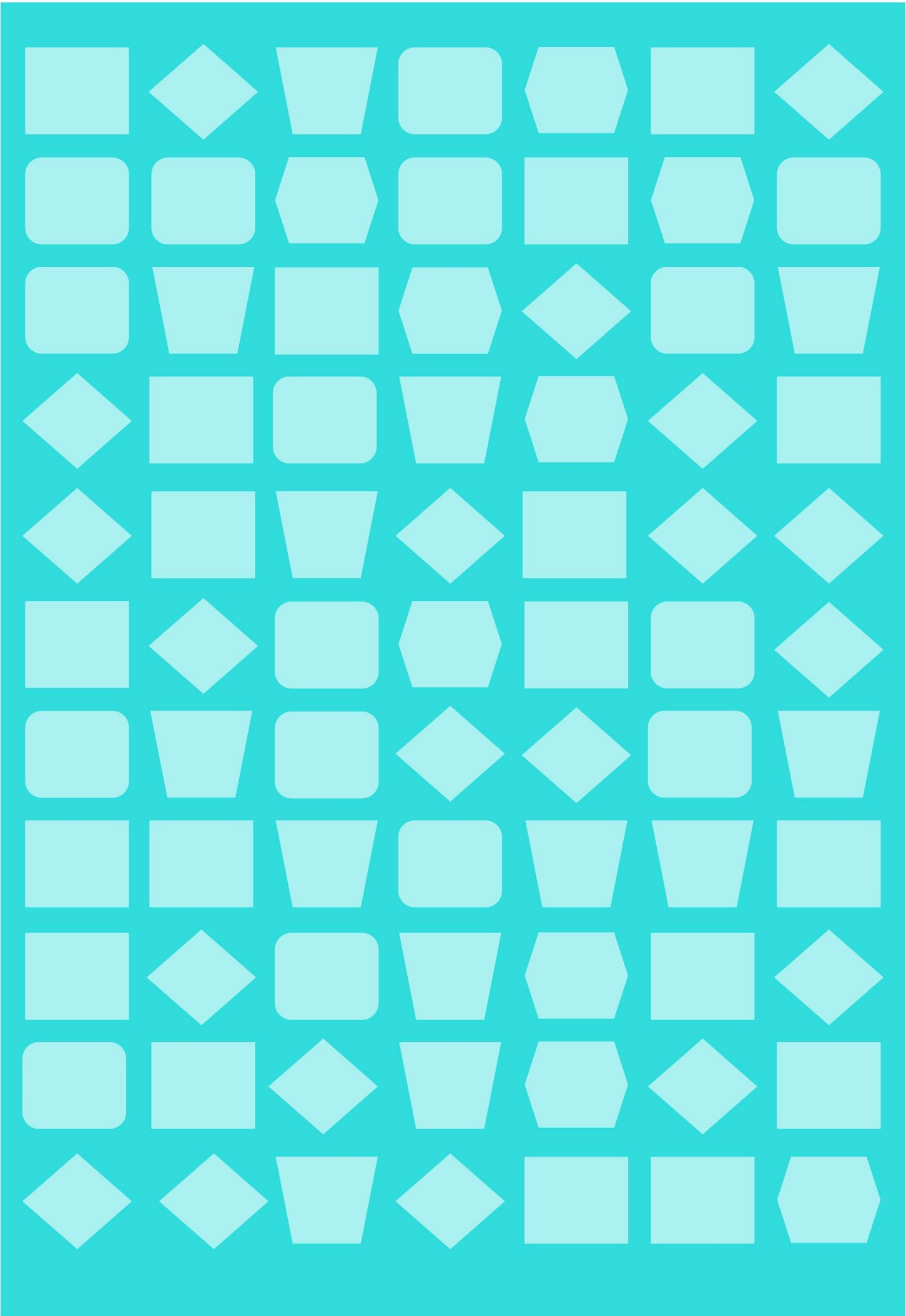


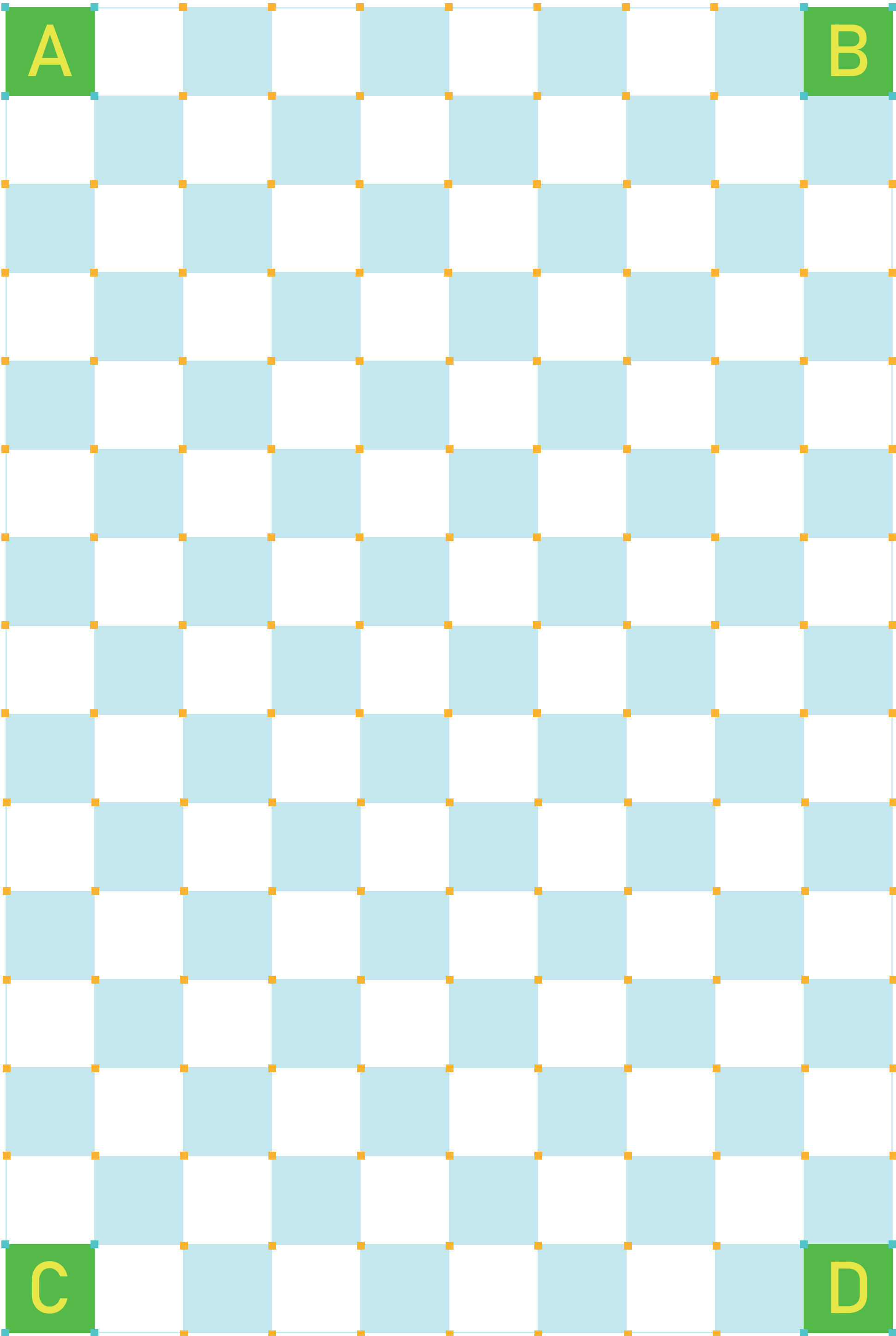
Co-funded by the
Erasmus+ Programme
of the European Union

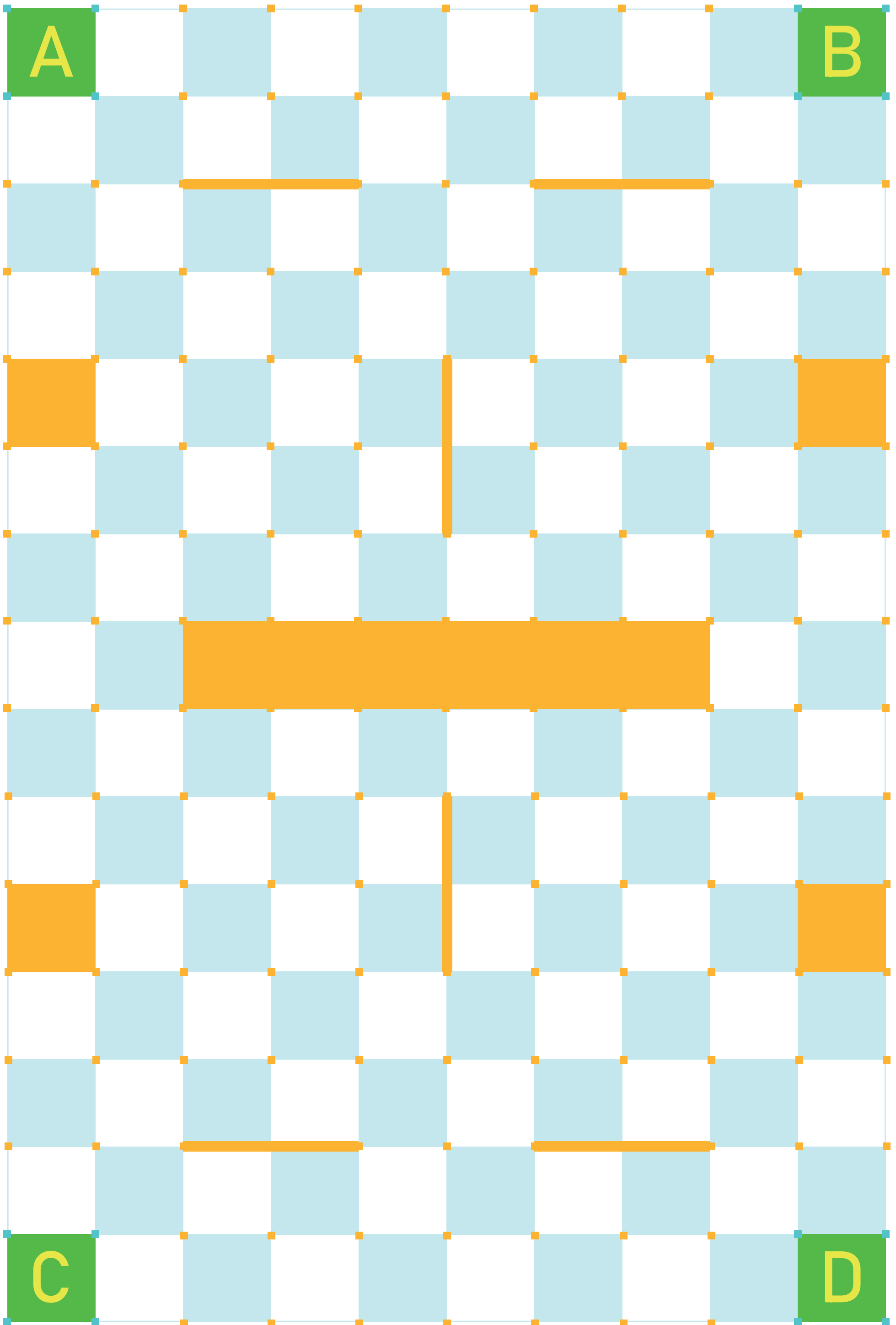


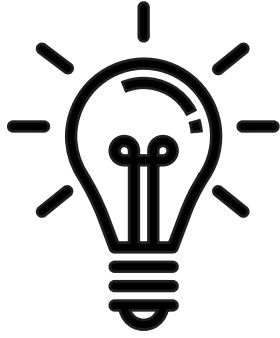
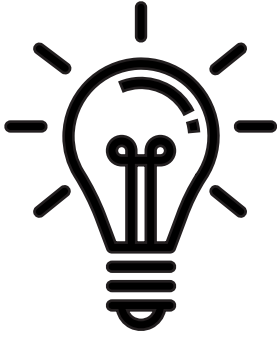
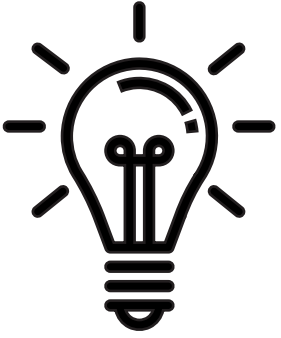
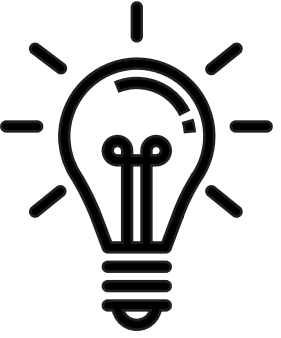
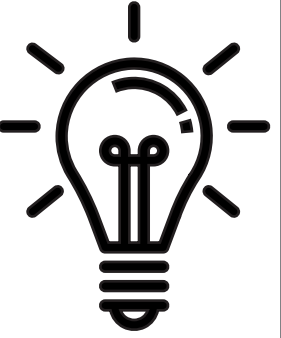





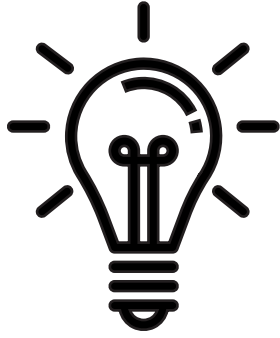
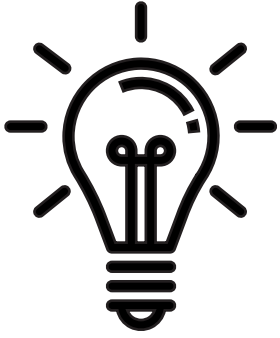
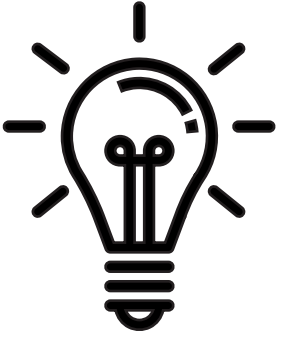
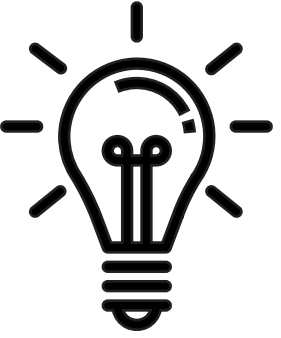
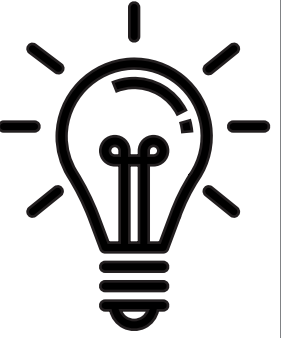





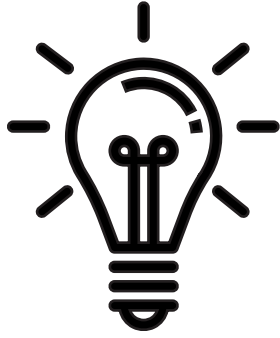
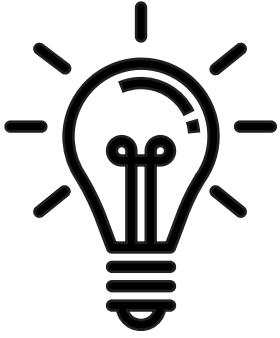
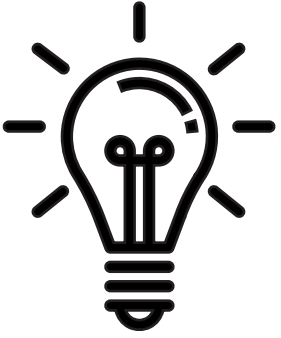
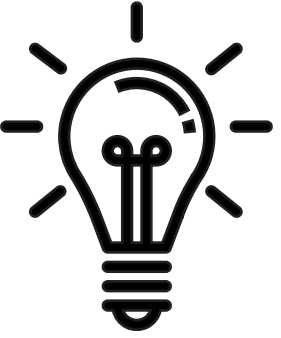
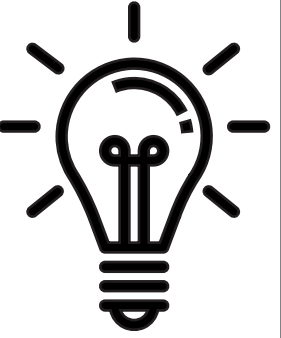
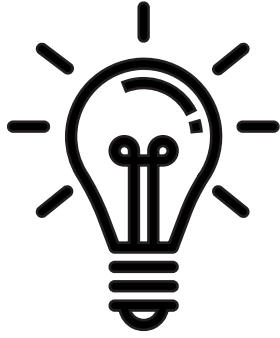
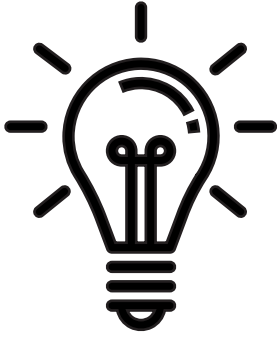
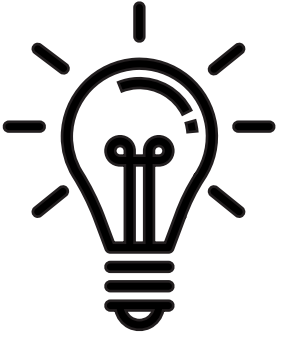
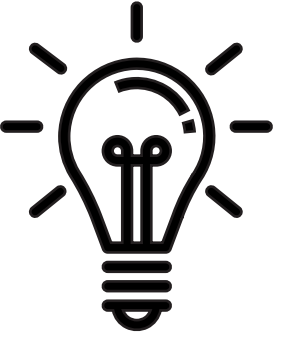
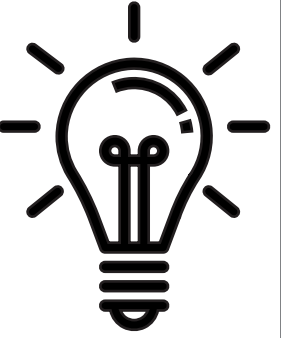
Universidad de Deusto
University of Deusto

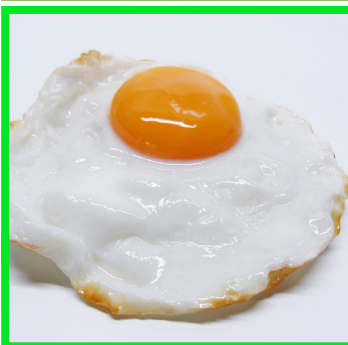
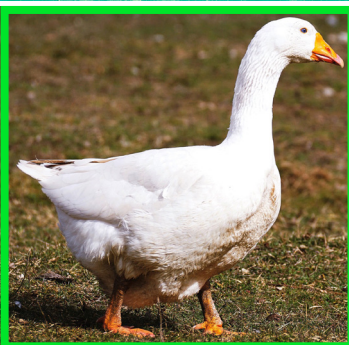
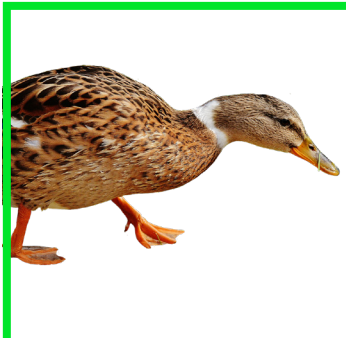
Deusto

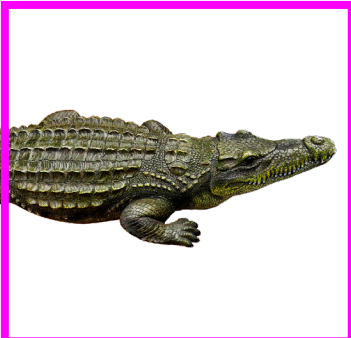


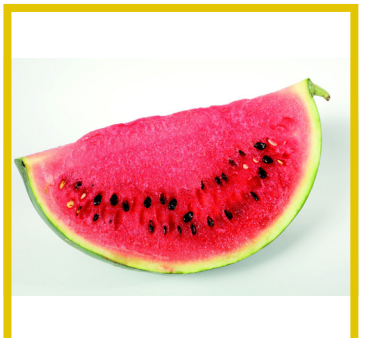
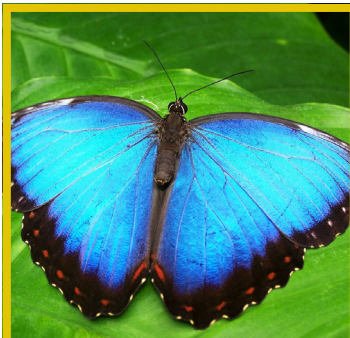
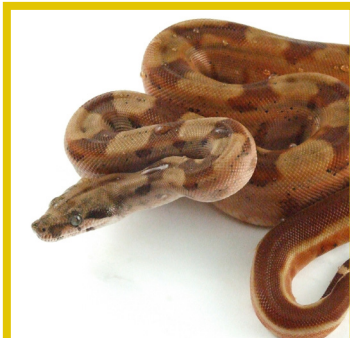
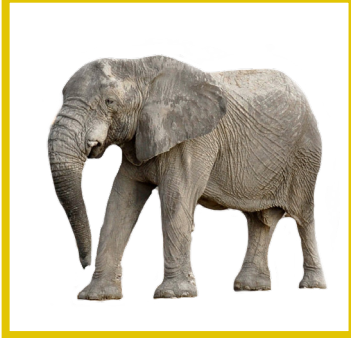


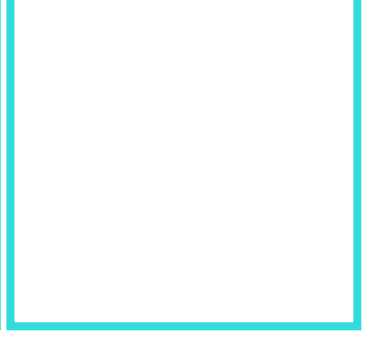
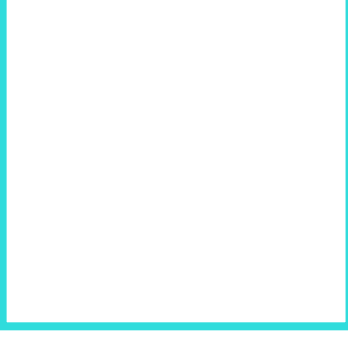
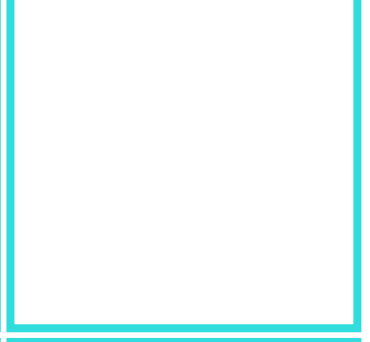
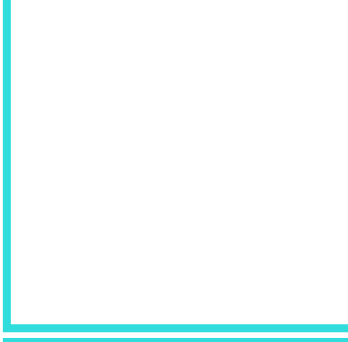
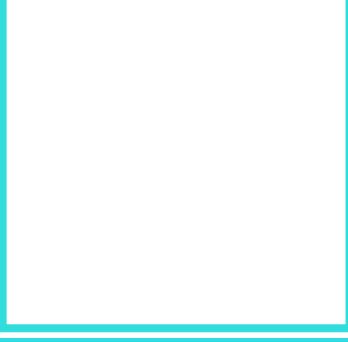
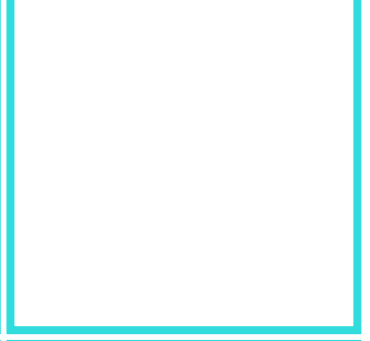
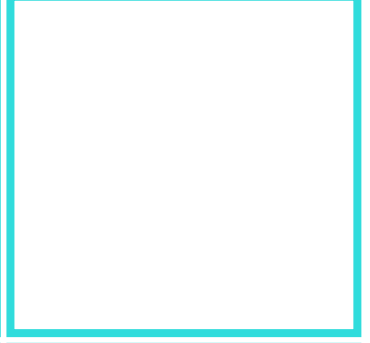
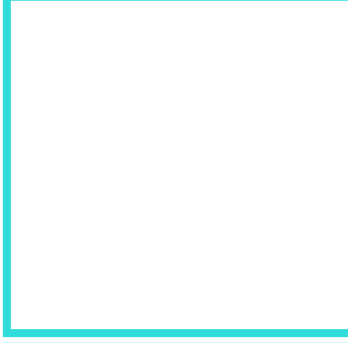
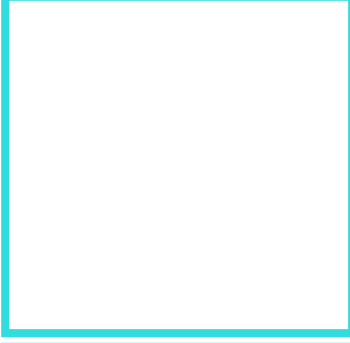
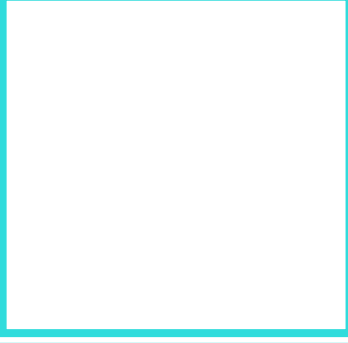
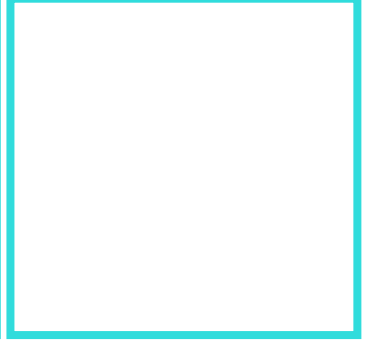
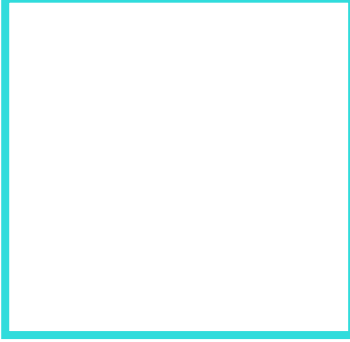
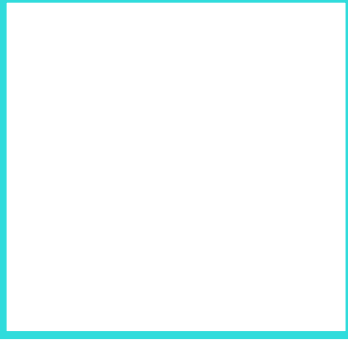
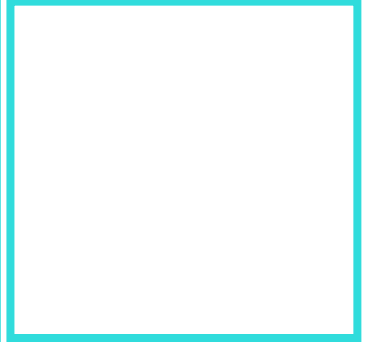


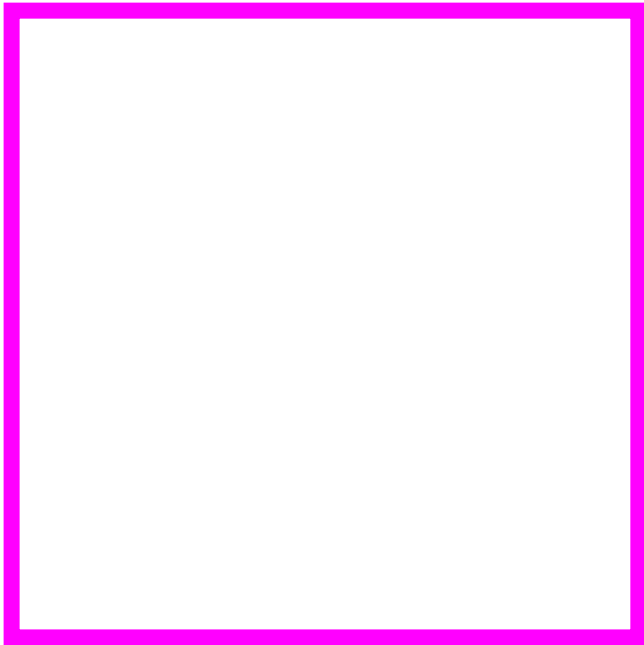
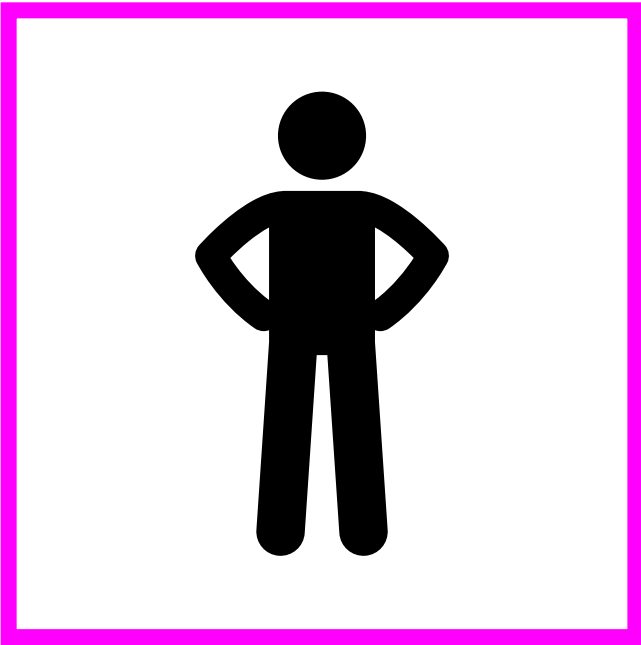
| | | | | | | | | | |
|--|---|---|--|---|---|---|---|---|---|
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  | | | | | |
|  |  |  |  |  | | | | | |











1x

2x

3x

4x

Als

.....

dan

.....

If

.....

then

.....

Baldin

.....

orduan

.....

Si

.....

entonces

.....

Si

.....

mester

.....