

Computational thinking: dat klinkt naar eentjes en nulletjes en code. Maar wat als je het begrip los haalt van het programmeren en juist inzet op een creatieve manier? In dit artikel aandacht voor de 'kunst' van computational thinking.

Technologie & onderwijs

Technologie heeft een steeds grotere rol in onze samenleving, en het is dan ook niet gek dat hier in scholen meer aandacht voor komt. Mede door deze technologie zijn problemen steeds complexer geworden. Tegelijkertijd is diezelfde technologie steeds toegankelijker en kan het (bijna) iedereen in staat stellen om te helpen nog grotere problemen op te lossen. [Computational thinking](#) helpt hierbij als set strategieën om, met behulp van technologie, complexe problemen op te lossen.

Maar wat als je nu niet perse een probleem hebt, maar vooral iets wil maken? Hoe past computational thinking daar bij?

Kunst & computers

Het mooie van alle nieuwe mogelijkheden is dat er op creatieve wijze enorm veel mee gemaakt kan worden. Door gebruik te maken van digitale fabricage zoals een [3D printer](#) of [lasersnijder](#) kun je digitale ontwerpen maken, en die vervolgens vasthouden. Hetzelfde geldt voor [microcontrollers](#) zoals een [microbit](#): je begint met technologie, maar vervolgens doet het iets in de echte wereld. Ideaal dus om [eigen ideeën](#) van leerlingen vorm te geven, terwijl je wel op een complex niveau bezig bent met programmeren en kunst.

Computers en kunst kunnen zo goed samenwerken: je leert nadenken over wat je ziet, je leert dit beschouwen en daarnaast ook zelf nieuwe dingen creëren. En hoewel de digitale wereld en die van de kunst elkaar niet altijd lijken te vinden is het juist de kruisbestuiving waar veel nieuwe mogelijkheden zijn voor leerlingen.

Code als materiaal

Het gebruik van computers (of andere technologie) binnen kunst maakt niet direct dat je bezig bent met computational thinking. Dus waar ontstaat deze link? In een eerder artikel omschreven we al de 4 verschillende gebieden binnen computational thinking: decompositie, abstractie, algoritme en patronen herkennen. Door deze vaardigheden in te zetten binnen een creatief proces, gericht op het maken van iets wat je wil maken, dan uit zich dit al snel in een creatief product.

De vaardigheden worden dan niet gebruikt om een probleem op te lossen, maar juist om een idee helder te krijgen, en dit vervolgens met behulp van technologie uit te werken. Daarbij speelt code de rol van materiaal: het verbindt de digitale wereld aan het tastbare.

Wat is er nodig?

Hoewel je met alleen een laptop of PC al kunt beginnen, is het natuurlijk vooral interessant als er meer soorten materiaal zijn. Wanneer je een makerspace op school hebt is dit eenvoudig: er is dan al

veel voorhanden.

3D printers, lasersnijders, maar zeker ook snijplotters zijn leuk en praktisch om digitale ontwerpen fysiek te maken. Met microcontrollers kun je ook veel: iets maken wat iets doet is voor leerlingen bijna magisch. Zelfs met een eenvoudig programma kun je al snel een motortje of ledje programmeren met een microbit. Wil je iets verder, dan kun je met arduino's of andere microcontrollers aan de slag. En daar waar digitale fabricage en microcontrollers de digitale wereld naar de analoge werkelijkheid brengen, doet virtual reality doet dat precies andersom. Ook hierin kun je zelf dingen bedenken en maken.

En wat doe je dan?

Veel projecten die draaien om samenwerking tussen kunst en computational thinking gaan uit van twee dingen: leerlingen bedenken zelf ideeën en gebruiken hierbij code om deze ideeën werkelijkheid te laten worden. En dan niet omdat ze een probleem willen oplossen, maar omdat ze iets willen maken. Er zijn veel voorbeelden te vinden en te bedenken van hoe je dit kan aanpakken, en hieronder hebben we er een aantal op een rijtje gezet:

2D Ontwerpen genereren: Met behulp van 'TurtleArt' kun je, doormiddel van code, zelf ontwerpen genereren. Deze manier van programmeren is eigenlijk al best oud: Seymour Papert bedacht dit om leerlingen interactief met code te laten werken. Er zijn verschillende versies op internet te vinden die hetzelfde doen, [maar deze is misschien wel het makkelijkste](#) (en in het Nederlands!). De ontwerpen kunnen opgeslagen worden als 'SVG-bestand', welke je direct kunt lasersnijden, plotten of 3D printen.

3D ontwerpen genereren: Tinkercad wordt veel gebruikt om in 3D ontwerpen te maken, en sinds kort heeft het ook een functie om ontwerpen te genereren op basis van code. Het lijkt daarbij op scratch of turtle art, en je kunt zo [3D objecten bedenken](#) op basis van een stappenplan dat je zelf bedenkt.

Creatief programmeren: Programmeren niet creatief? De programmeertaal 'Processing' is speciaal bedacht voor kunstenaars en anderen die met zo min mogelijk programmeerkennis iets willen programmeren. Het is daarbij erg gericht op visualisaties, waardoor je snel resultaat hebt. De afgeleide versie 'P5js' is nog praktischer, en er zijn veel tutorials te vinden.

'Iets maken wat iets doet': combineer een microbit (of andere microcontroller) met allerlei andere materialen en er zijn enorm veel mogelijkheden. [Kijk hier eens voor 3 goede voorbeelden.](#)

Creatieve technologie

Genoeg voorbeelden om mee aan de slag te gaan dus! [Digitale geletterdheid](#), [programmeren](#), computational thinking, nieuwe technologie: allemaal kunnen ze toegepast worden in een [creatief proces](#) om daarmee een kunstzinnig project te maken. Sterker: juist die creativiteit helpt om de technologie beter te begrijpen en te gebruiken en andersom maakt de technologie nieuwe soorten

kunst mogelijk. Mogelijkheden genoeg dus voor interessante projecten met een mooie wisselwerking tussen technologie en kunst, waarbij leerlingen moderne middelen gebruiken om zichzelf creatief te uiten. Wat ga jij met je leerlingen maken?