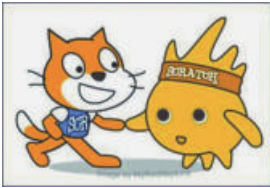


Scratch (1)

René Suiker



Scratch een verrassend (eenvoudige) manier om met programmeren bekend te raken.

De artikelen verschenen in de Softwarebus, een uitgave van HCC!CompUsers ig, en werden welwillend beschikbaar gesteld als syllabus bij het HCC-seizoensthema 'Robotica - Drones - Programmeren'

Geïnteresseerd geraakt wat HCC!CompUsers nog meer te bieden heeft? Neem dan eens een kijkje op www.compusers.nl of abonneer je op de SoftwareBus

Als CompUsers zijn we de interessegroep van 'toepassingen', 'applicaties'. We hebben nooit zo heel veel met programmeren gehad, hoewel er natuurlijk onder de leden wel degelijk een en ander geprogrammeerd is.

De jeugd heeft de toekomst

En zo kon het gebeuren dat, in het kader van het betrekken van een nieuwe generatie bij onze activiteiten, onze voorzitter op de gedachte kwam dat veel van onze leden intussen grootouder zijn en misschien wel eens de leuke dingen van de computerhobby op hun kleinkinderen willen overdragen. De generatie ertussen moeten we niet gelijk vergeten, maar met Scratch willen we ons vooral richten op de jonge jeugd en op de grootouders die hierbij willen helpen.

Krassen

Scratch, dat is Engels voor 'krassen', 'krabbelen'. Vroeger noemden we Cobol-programmeurs wel eens gekscherend 'Cobol-krassers' maar nu is scratcher, de gebruiker van Scratch, een soort geuzennaam.

Feuilleton

De bedoeling is dat er, in navolging van de reeks over Word-Press, weer een soort feuilleton van mijn hand verschijnt, waarna iedereen zich gerust geroepen mag voelen bij te dragen door mij aan te vullen of zo nodig te corrigeren. Ik denk dat we wel een blad of zes kunnen vullen, voordat een en ander echt uit de verf komt, maar dat hoeft geen probleem te zijn. Vanaf dit artikel zijn we al direct bezig en kunnen we de computer al dingen laten doen. Het leuke is dus, we leren programmeren en tegelijk op een object-georiënteerde manier.

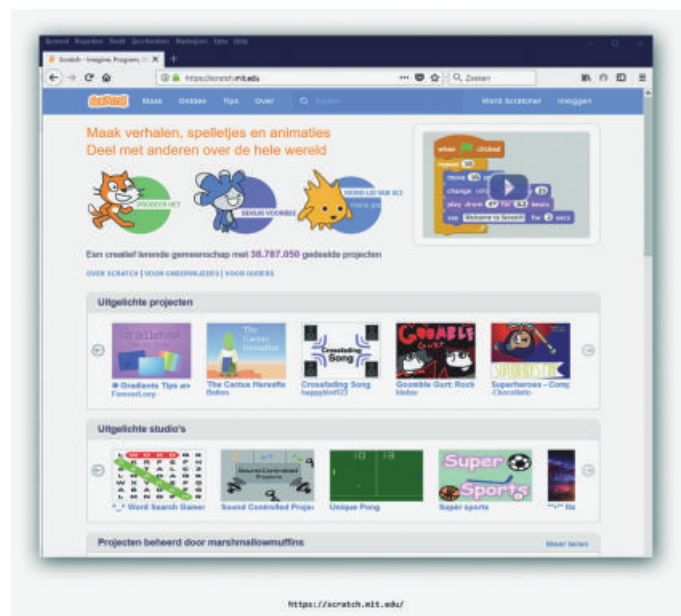
Even nog een opmerking vooraf, Scratch werkt niet op een tablet of telefoon. Je moet gewoon met een laptop of desktop aan de slag. Windows, Linux of Mac, dat maakt niet uit.

Aan de slag

Ons avontuur begint op <https://scratch.mit.edu/> en ik raad je aan gewoon met me mee te gaan. Je komt dan op de site van een van de beroemdste universiteiten van Amerika, misschien wel van de wereld, de *Massachusetts Institute for Technology*, kortweg dus *mit.edu*.

Je komt dan terecht op een kleurig ogende pagina, die gelijk wel aanduidt dat de doelgroep de jeugd is. En met name de nog wat jongere kinderen, die graag met opa of oma op ontdekkingsreis gaan.

Rechts bovenin zie je de optie om in te loggen, daarnaast de knop om scratcher te worden. Nou is het niet verplicht om een account te hebben en te gebruiken, je kunt scratch ook als gast gebruiken en ontdekken, maar ik raad aan om gewoon een account aan te maken. Voor de prijs hoeft je het niet te laten, het kost namelijk niets.



Figuur 1- Scratch home page

Verder is MIT nou niet bepaald een instituut dat je met spam bekogelt. Als je een account aanmaakt, wordt één keer om een bevestiging van je e-mailadres gevraagd. Heb je dat gedaan, dan krijg je een welkomstmil, met daarin wat links, waarover straks meer.

Na de aanmelding, op 20 september jl., heb ik geen mail meer van ze gekregen en het is nu 6 oktober. Ofwel, je mailbox loopt niet vol, MIT is een gerenommeerd instituut, de kosten zijn nihil: wij maken een account aan. Klik dus op 'Word Scratchter' (oftewel: ga lekker scratchen!).

Scratcher worden

Je krijgt dan een pop-up-venster te zien, zoals in de figuur hierna. Hier vul je een gebruikersnaam in, die je wel even creatief moet kiezen. Maar als je een naam kiest die al bestaat, dan krijg je daar een melding van. Zodra je een goede gebruikersnaam hebt gekozen, kun je een wachtwoord bedenken en dat twee keer op identieke wijze invullen, iets



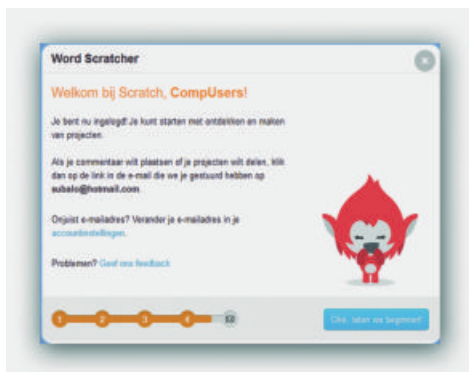
Figuur 2- Word Scratcher

Dan klik je op 'Volgende', het blauwe knopje rechts onderin.

Op de volgende pagina binnen de pop-up worden je geboortedatum, geslacht en land gevraagd. Deze gegevens zijn bedoeld voor statistische doeleinden bij MIT, maar worden verder niet gedeeld.

Geen reden dus om hier geheimzinnig te doen, bovendien helpt het als je ooit nog eens contact moet opnemen, bijvoorbeeld als je je wachtwoord vergeten bent. Overigens gaat het alleen over geboortemaand en jaar. Naast man of vrouw kun je ook nog iets anders opgeven, maar dat is in het kader van dit artikel niet zo van belang. Klik op 'volgende'.

Hierna vul je je e-mailadres in. Overigens kun je op één e-mailadres meerdere accounts aanmaken. Ook je e-mailadres voer je twee keer in, om typfouten te vermijden. Verder heb je nog de keuze om al dan niet updates van het Scratchteam te ontvangen. Dat raad ik aan, want je kunt je altijd weer afmelden, als je het toch niet zo leuk vindt. Aanmelden als je niet weet waar, kan wel eens ingewikkelder zijn. Klik weer op 'Volgende'. Gelukt? Dan volgt dit scherm:



Figuur 3 - Gelukt

Niet online

Ik vermoed dat bij de meeste lezers de computer continu met Internet is verbonden, maar er bestaat ook de mogelijkheid om Scratch lokaal te installeren, bijvoorbeeld op een laptop, zodat je ook met Scratch aan de gang kan als er eens geen internet beschikbaar is, zoals op een vakantiepark. Hier kunnen we in de toekomst nog wel eens op ingaan. Nu gaan we echter aan de slag via het net.

De omgeving

Goed, we hebben ons aangemeld, we kunnen dus beginnen met ontdekken. Klik rechtsonder op de blauwe knop 'Oké, laten we beginnen'. We komen weer op het startscherm zoals in *Figuur 1- Scratch home page*, alleen zie je nu rechtsboven je gebruikersnaam. Laten we nog eens goed naar dat scherm kijken.

wat je ongetwijfeld nog kent van alle overige sites waar je je op moet geven. Zorg dat je het wachtwoord dat je in gedachten hebt ook goed intypt, of anders toch minstens consequent fout.

Onder de navigatiebalk van de browser zien we een blauwe menubalk van Scratch. Daarover straks meer.

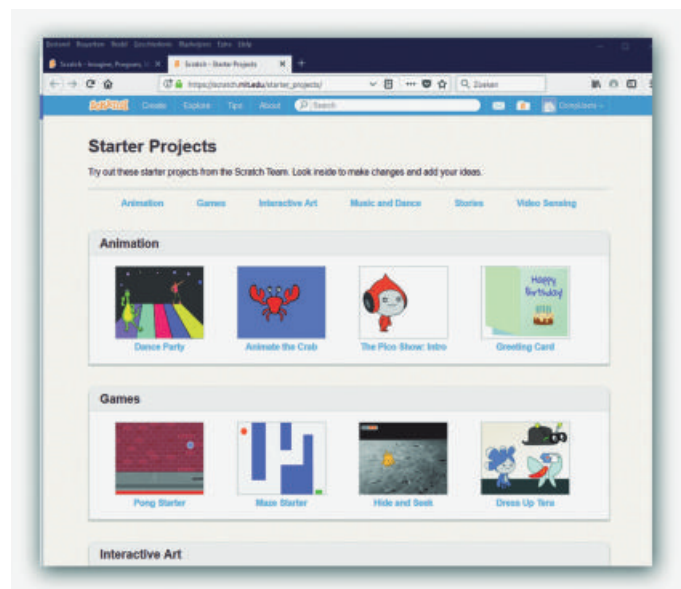
Onder de blauwe balk kun je een gele balk aantreffen, dat is de berichtenbalk. Ik denk dat hij niet altijd zichtbaar is, maar hier zie je bijvoorbeeld staan dat je je e-mailadres nog moet bevestigen, of er staan aankondigingen van de makers. Op dit moment staat er bij mij een melding om Scratch 3.0 alvast uit te proberen. Dat is iets voor een vervolg.

Hieronder zie je een aantal 'blokken' in het werkgedeelte. Zo begint het met 'Welkom bij Scratch', met daaronder weer drie blokjes, in feite grote knoppen, om je op weg te helpen.

Het eerste blokje heet 'Leer hoe je een project maakt in Scratch'. In feite moet ik niet aanraden om hierop te klikken, want dit maakt deze artikelenreeks in feite overbodig.

Maar dat is natuurlijk flauw, het is heel leuk om alvast een kijkje te nemen. En de ene aanpak en de andere om iets te leren kunnen elkaar natuurlijk ook aanvullen. Als je hierop klikt, kom je in een Scratch-scherm, dat ik straks ga uitleggen, met aan de rechterkant stap voor stap instructies om je eerste eigen programma te maken. Je kunt hier dus ook beginnen, misschien wijst het zich vanzelf, maar ik laat het nog even voor wat het is.

Het tweede blokje heet 'Probeer de starterprojecten'. Dan kom je op een pagina met startersprojecten, die er ongeveer zo uitziet:



Figuur 4 - Starterprojecten

Hier vind je een aantal projecten, die bepaalde aspecten van Scratch nader verduidelijken. Ik kom er straks nog op terug. Het derde blokje is 'Maak contact met andere Scratchers'. Je komt dan op de pagina van het 'Scratch Welcoming Committee', ofwel het welkomstcomité van Scratch. Daar gaan we een volgende keer uitgebreid op in, maar eerst maar eens terugkomen op wat ik al beloofd heb.

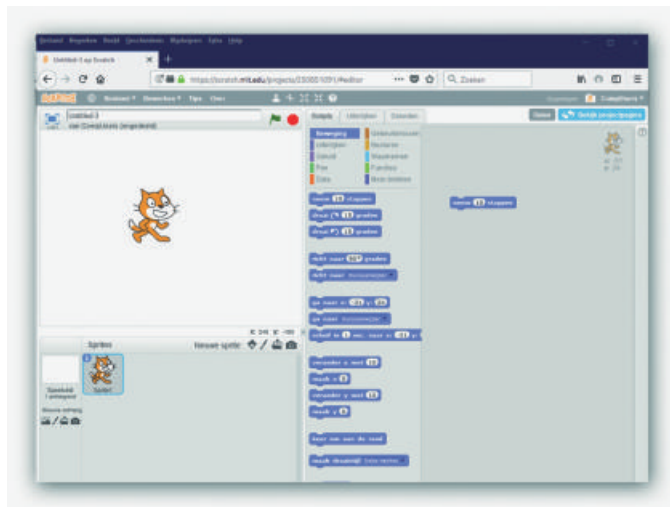
Naast het eerste blok, 'Welkom bij Scratch!', vind je een blok 'Scratchnieuws'. Daar vind je berichten vanuit de makers van Scratch, als een soort blog, dus het laatste nieuws eerst. Overigens, als je intussen mij gevolgd hebt en dus op een andere pagina in je browser bent: je kunt altijd weer terug naar de 'home'page, door links bovenin op 'Scratch' te klikken.

Hieronder vind je nog een aantal blokken, zoals 'Uitgelichte projecten', 'Uitgelichte studio's', enzovoort. Helemaal onderin vind je de gebruikelijke voetnoten, met informatie over de site, over Scratch, en helemaal onderin

Is je trouwens iets opgevallend aan die blokjes met de instructies? We pakken het eerste blokje onder 'Beweging', het blauwe blokje 'neem 10 stappen' en slepen dat naar de scriptzone.

De eerste instructie

Het scherm ziet er nu als volgt uit:



Figuur 6 - De eerste instructie

En let eens goed op dat instructieblokje (figuur 7):



Figuur 7

In het blauwe blokje zie je de instructie, in het witte gedeelte zie je een parameter die bij de instructie hoort. En let ook eens op de vorm. Met de palletjes schuif je straks de blokjes die bij elkaar horen ook in elkaar, zodat je een set instructies straks als één geheel kunt behandelen.

Als je nu op het groene vlaggetje klikt gebeurt er nog steeds niets. Dat komt allemaal nog, alles op zijn tijd. Maar als je op de instructie in de scriptzone klikt, dan zie je de kat een stukje naar rechts bewegen. Er staat immers 'tien stappen'. De kat kijkt naar rechts en gaat in die richting tien stappen vooruit. Dat doet hij in één keer. En elke keer als je op de instructie klikt doet hij dat weer, tot hij bijna uit het speelveld verdwenen is.

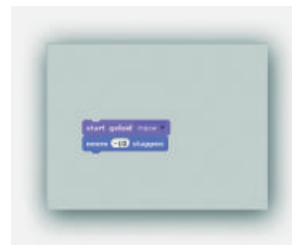
Nu kun je de kat gewoon oppakken (met de muis, het is een omgekeerde wereld hier) en hem ergens anders neerzetten en dan kun je weer verder gaan. Je kunt ook een keer op de '10' in de instructie klikken en dan kun je er bijvoorbeeld '-10' in typen. En zowaar, de kat gaat dan achteruit als je op de instructie klikt.

We hebben nu dus onze eerste instructie in Scratch gebruikt, uitgevoerd, aangepast en opnieuw uitgevoerd. We zijn nog geen programmeurs, maar hebben de kat op ons commando laten bewegen en dit is het begin van de ontdekkingsreis.

Meer instructies

Klik nu in het blokkenpallet eens op geluid en sleep de instructie 'start geluid miauw' naar de scriptzone en wel zo, dat dit blokje aan de bovenkant van het blauwe blokje aansluit, zoals te zien in figuur 8.

Klik nu weer op een van de blokjes (maar niet op de '-10' en ook niet op de 'meow'). U hoort nu de kat miauwen en vervolgens doet de kat een stap naar rechts. Alsof u op zijn teen heeft gestaan. U ziet nu dus twee instructies die bij elkaar horen. De onderste is blauw, die zorgt voor beweging, de bovenste is lila, die zorgt voor geluid.



Figuur 8 - Twee instructies

Het witte deel in de blauwe instructie liet toe de waarde te veranderen. Als je hier een grote waarde invult, beweegt de kat heel snel. Probeer het gerust. In het geluidsblokje zie je een andere manier om de instructie aan te passen, je ziet een driehoekje naast 'miauw' en als je daarop klikt, krijg je een aantal keuzes voorgeschoteld, waarvan je er één kunt kiezen. In dit geval is er nog niet veel keuze, het is óf miauw óf iets nieuws opnemen, want we hebben nog niet veel geluid in onze bibliotheek. Maar als u hier al een 'blaf' had staan, weet u nu hoe u een kat kunt laten blaffen.

Er is nog wat ruimte op deze pagina, dus we gaan nog een klein beetje verder. We hebben nu twee instructies, we zetten er nog een derde vóór, weer uit de beweging, weer de tien stappen, maar nu passen we de waarde 10 niet aan. Daarna (dus die 'wringen' we ertussen), nemen we ook de instructie 'draai rechtsom 90 graden' en dan weer 'neem 10 stappen'.

Besturen

Vervolgens kiezen we de instructieset 'Besturen' en slepen de set 'Herhaal 10 keer' naar onze instructieset toe. Zodra je de instructie naar de bestaande set toe sleept, zie je 'de bek' van de instructie open gaan. Wat we nu hebben gezegd, is, dat als de instructie moet worden uitgevoerd, dit tien keer moet worden gedaan. Alles ziet er nu zo uit:



Als je nu op een van de instructies klikt, hoor je een zielig gejam en de kat gaat een beetje zielig tekeer, misschien niet helemaal wat je verwachtte, maar dat leg ik uit.

In essentie wordt tien keer achter elkaar de set van vijf instructies afgespeeld. Echter, zonder te wachten; dus met een beetje snelle computer zie je niet wat er gebeurt maar wel is

de kat intussen tien keer negentig graden gedraaid, dus twee keer helemaal rond en één keer half, en intussen is de ene miauw nog niet uitgespeeld of de volgende wordt gestart. Daar is dus nog wat fine-tuning voor nodig en dat gaan we zeker ook nog doen. Alleen, niet meer vandaag.

Scratchen

Dit is natuurlijk nog maar het begin, maar zie het als de stap naar 'hello world' bij de traditionele programmeertalen. Het begint simpel, maar op enig moment kun je het jezelf zo moeilijk maken als je wilt. We willen hier wel een vervolg aan geven, bij gebleken belangstelling. Maar niet alleen dat, we willen dit in samenwerking doen met andere groeperingen.

We hebben al de toezegging dat HCC!AI ig mee wil doen, die zijn al actief met Scratch. We denken dat de Senioren-Academie hier ook wel brood in ziet. En we denken aan gezamenlijke projecten, waarbij AI al concreet bezig is, en we denken ook aan workshops tijdens onze dagen.

We willen jullie op alle mogelijke manieren begeleiden, maar het is vooral ook zaak om zelf te gaan ontdekken.

Scratch (2)

René Suiker



Deel 2 van de vervolgserie over het programma Scratch, een objectgeïoriënteerde programmeer-app. waarmee je heerlijk creatief bezig kunt zijn op computer of tablet. René Suiker neemt u aan de hand mee naar de leeromgeving van Scratch en ScratchJr van het MIT in Massachusetts (USA).

Inleiding

Tja, wie a zegt, moet ook b zeggen. En zo is hier dan deel 2 van Scratch, een feuilleton over programmeren. In de SoftwareBus, programmeren? Ja dus, dat kan zowaar. In het kader van samenwerking, in het kader van het aantrekken van nieuwe generaties, maar vooral ... omdat het leuk is.

Voor de duidelijkheid: we zitten hier niet bij de interessegroep 'programmeren', die inderdaad programmeren als 'core activiteit' heeft staan, maar bij CompUsers zijn er ook mensen die dit leuk vinden, die een technisch hart hebben dat sneller gaat kloppen bij dit idee. En zo zijn we dus vandaag bij deel 2 beland.

Allereerst moet ik even iets rechtzetten. Want hoewel ik niet heel veel onzin heb verteld in mijn vorige artikel, is mijn inleiding niet helemaal waarheidsgetrouw. Niet bewust, maar ik werd er fijntjes op gewezen dat we als CompUsers inderdaad zelf een boekhoudpakket HABO hebben uitgebracht, en dat dit niet door Henk van Andel, maar voornamelijk door Hans Luning was geschreven. En dat dit, op basis van het programma FIBO, dan weer door Wim Souren en Joep Bär was geschreven. Waarvan akte.

Verder heeft niemand anders me op fouten gewezen, maar er zijn nog wel enkele suggesties gedaan, waar ik toch even aandacht aan wil besteden.

Leeftijd

Zoals ik al aangaf, er waren wel wat opmerkingen op mijn eerste artikel, zeker op de proefversie, want zo gaat dat op een redactie. Maar een deel van de suggesties heb ik niet direct overgenomen, ook al waren ze toch zinvol. Ik wilde ze echter deze keer aan de orde stellen.

Ik gaf al aan, we zien dit vooral als een middel om ook de jeugd bij onze vereniging te betrekken. Misschien slaan we zo wel een generatie over, is dit iets voor opa (en/of oma, uiteraard) en kleinkind. Maar natuurlijk kan ook papa of mama met de kinderen iets doen. Zoals ook volwassenen met hun kinderen met Lego helpen, maar soms ook gewoon meedoen omdat ze het zelf ook nog leuk vinden. En daar is niets mis mee, althans, niet in mijn ogen. Maar goed, ik lees ook de Donald Duck nog wekelijks.

De vraag was voor welke leeftijdsgroep Scratch bedoeld is. Daar is een officieel antwoord op: van 8 tot 16 jaar, maar ik denk dat dat wat krap is. Er is ook een ScratchJr op tablet, voor vijf tot acht jaar, dat uitgaat van een kind dat nog niet kan lezen. Ik weet niet hoe het uw (klein)kinderen vergaat, met zes jaar las ik zelf de krant. Ik wil niet zeggen dat ik inhoudelijk alles al begreep, maar ik kon het toch wel al lezen. Ik neem aan dat ik niet heel uniek ben hierin, hoewel ik ook wel besef dat het niet heel gebruikelijk is. Maar met acht jaar zou je toch al enigszins moeten kunnen lezen. En voor wie is Lego bedoeld? Daar zit geen maximumleeftijd aan.

Wat kun je ermee?

Een andere vraag was 'wat kun je ermee?' Dat is een heel interessante vraag, kort gesteld, maar daar kan ik een heel artikel aan besteden. Wat kun je met Lego? Je kunt het gebruiken om eenvoudige autootjes te maken, je kunt er ook kastelen mee bouwen. En intussen graafmachines, ruimtestations, noem maar op. Het eerste is makkelijker, zeker als de stap-voor-stap-handleiding gevolgd wordt. Het tweede is moeilijker, maar daarom voor sommigen ook weer interessanter. Ik weet niet of je er alles mee kunt maken; het is ongetwijfeld niet de eerste programmeertaal die bij je opkomt als je een serieuze business-applicatie als een ordermanagementsysteem wilt maken. Maar het is in feite een redelijk complete programmeertaal, die veel structurelementen van andere talen ook heeft. Alleen, wat makkelijker toe te passen.

Een derde vraag ligt in het verlengde hiervan, wat is de beloning voor het kind? Het programma is visueel ingesteld, het kind ziet al heel snel een resultaat dat er grafisch interessant uitziet en dat nodigt uit om verder te gaan. En net als het schaak-initiatief van de gemeente Rotterdam helpt dit programma het logisch denken te stimuleren. Is dit een beloning? Ja, ongetwijfeld. Ziet het kind dat ook zo? Dat is nog maar de vraag.

Maar, naar de analogie met Lego: er is wel een heel groot voordeel. Als je Lego leuk vindt en steeds meer en moeilijker wilt bouwen, moet je als kind ook wel veel geduld hebben. Stel dat je een basisdoos hebt gekregen en de goede Sint (of Zwarte Piet, maar laten we die discussie hier maar niet overdoen) heeft je ook nog een mooie extra doos gegeven, dan kun je leuke dingen maken. Maar als je iets groters wilt, dan moet je weer meer steentjes hebben, soms ook andere steentjes. Op een gegeven moment wil je er wel een motor-tje bij hebben, of zelfs elektronica.



Uit aflevering 1: welkom bij Scratch!

je dat hebt, dan kun je dat éne steentje net zo vaak gebruiken als je wilt.

En zo kun je dus, terwijl je jezelf ontwikkelt, zonder extra kosten je resultaten steeds mooier maken. En als je er heel handig in wordt, kun je je eigen steentjes maken.

Alleen, als je zo'n jaar of tien bent heb je nog geen krantenwijk en de tijd gaat nog niet zo snel als bij de ouderen. Het duurt heel lang voor je weer jarig bent. En met Scratch, daar heb je een Lego-steentje en als

Tablet

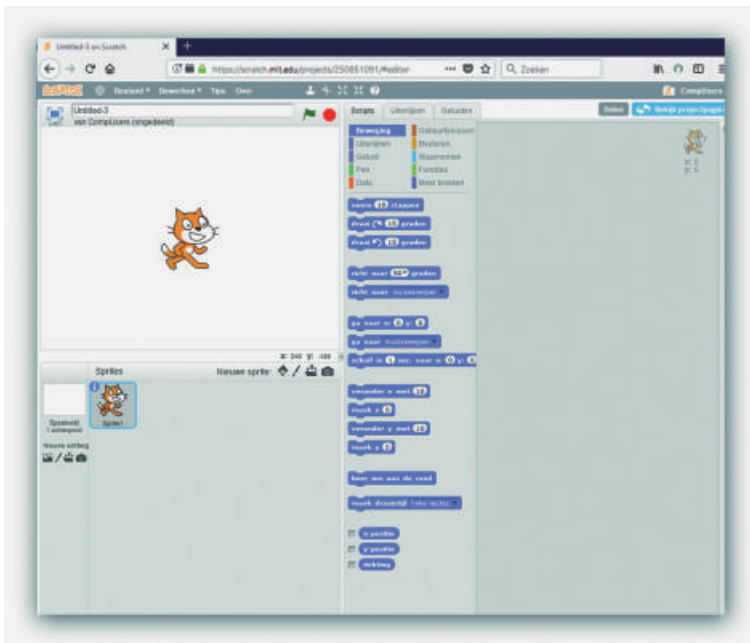
Zoals ik al zei, er is blijkbaar een juniorversie die op een tablet werkt, maar ik kreeg van een oplettende lezer ook nog te horen dat de volgende versie van (gewoon) Scratch (3.0) niet meer op Flash, maar op HTML5 is gebaseerd en dat die dan ook op een tablet kan werken. De bètaversie is blijkbaar al beschikbaar, maar officieel komt hij in januari uit.

Terugblik

Nog even terugkijken wat we al geleerd hebben en dan gaan we gauw weer aan de slag. Vorige keer heb ik uitgelegd hoe je scratcher kan worden en ik hoop dat dit advies al veel is opgevolgd. Ik heb de omgeving uitgelegd en ook wat je zoal kunt verwachten als je je aangemeld hebt. Vervolgens heb ik de instructiegroepen summier uitgelegd.

Ten slotte heb ik beloofd om nog eens nader in te gaan op het Scratch-welkomstcomité, maar dat is nu nog niet aan de orde. Deze keer gaan we wat nader op de instructies in en gaan we eens wat meer programmeren. Want het zielig jankende katje waar we vorige keer mee eindigden, dat heeft intussen wel iets beters te doen.

We starten Scratch weer op via <https://scratch.mit.edu/>. We zien dan weer het intussen bekende startscherm, zoals hieronder nogmaals weergegeven. Ziet u iets anders, klik dan in de menubalk op 'maak'.



Figuur 1 - Scratch editor

Ik zal het niet opnieuw helemaal beschrijven, maar wel even heel kort. Onder het menu zie je links het 'speelveld' met daarin een 'sprite', in dit geval een kat. Een sprite is het voornaamste basisobject in Scratch, hetgeen allemaal lekker visueel maakt wat we aan het doen zijn.

In de loop der tijd zullen we ook andere sprites zien, ze misschien wel zelf maken en beweging simuleren door sprites te wisselen tijdens de loop van het programma, een beetje zoals dat in een tekenfilm ook gebeurt.

Rechts naast het speelveld (dat je overigens tijdens de 'run'-fase kunt maximeren zodat je een beeldvullend speelscherm hebt) zie je het 'blokkenpalet', waarin je de codeblokken kunt vinden, waarmee je je programma's maakt. Zeg maar: de verzameldoos met de legosteentjes. Alleen, als je hier een steentje uithaalt, blijft het steentje ook nog gewoon in de doos liggen. Je kunt de steentjes blijven gebruiken. Rechts in het scherm zie je de 'scriptzone'. Dit is in feite de

bouwplaat, waar je je programma's maakt. Ten slotte zie je links onder het speelveld nog het spritesgebied. Hier zie je de sprites die je kunt gebruiken, maar hier kun je ook kiezen voor sprites bewerken, nieuwe creëren, weggooien, etc.

En nu gauw aan de slag

Wat ik al aangaf, de jankende kat van de vorige keer laten we even voor wat hij is, we beginnen met een schone lei. Dus als je al een project hebt opgeslagen, prima, maar we starten nu 'van scratch'.

Als de kat nog niet in het midden van het speelveld staat, pak hem dan op met de muis en sleep hem naar het midden. Dan zie je onder in het speelveld de coördinaten van de kat. Je zou de kat op die manier heel precies in het midden kunnen zetten. Heel precies geven de coördinaten de positie van de cursor weer. Het is goed om te weten, dat de positie in het midden bepaald wordt door $x=y=0$. Meer naar links, dan wordt de x negatief, meer naar beneden wordt de y negatief. Als jullie op school wiskunde hebben gehad, dan zal je dit wel herkennen.

Je kunt in de legodoos op een blokje klikken en dan wordt de actie uitgevoerd. Als je in de legodoos op tabblad 'scripts' onder beweging kijkt, dan zie je daar ongeveer halverwege een blokje met de tekst 'ga naar $x: .. y: ..$ ' waarbij op de .. al waarden ingevuld zijn, maar het zijn overduidelijk invulvelden.

Ik weet dat sommigen van jullie heel precies hebben geprobeerd de kat op precies $x=0 y=0$ te krijgen. Als je in dit blokje nu voor x en y de waarde 0 invult en je klikt erop, dan zie je dat de kat precies naar het midden van het speelveld verplaatst wordt. Ook als je hem helemaal aan de rand hebt staan. Probeer het maar.

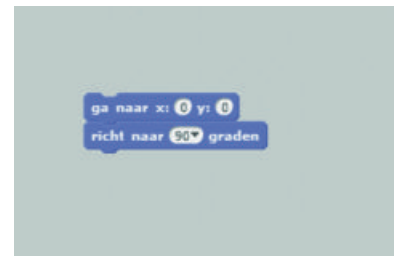
Elementaire bewegingen

Vervolgens schuif je dat blokje naar de scriptzone als eerste instructie van ons programma. Klik nu op het blokje 'draai naar rechts '30' graden'. Als er geen 30 in staat, maar een andere waarde, dan zet je er 30 in. Dertig graden is precies één uur op de klok, althans, de beweging van de kleine wijzer. Als je vervolgens 12 keer op dat knopje klikt, zie je de kat precies één keer helemaal rond gaan. Doe maar. Klik er vervolgens nog één keer op, zodat de kat 30 graden gedraaid is.

Zoek vervolgens het blokje 'neem 10 stappen', bovenin de Legodoos en klik daar een paar keer op. Je ziet de kat zich dan bewegen in de richting waarin hij staat, in dit geval dus 30 graden gedraaid t.o.v. volledig horizontaal. Als je de kat weer terug wilt laten lopen, kun je in hetzelfde blokje '-10' invullen en dan zie je hem terug bewegen als je erop klikt.

Overigens, de kat was dus 30 graden gedraaid t.o.v. de horizontaal, maar dat was in de dimensies van Scratch een looprichting van 120 graden. Een richting van 0 graden is precies verticaal, omhoog.

Zoek in de legodoos naar het blokje 'Richt naar .. graden' en vul op de .. '90' in. Klik er vervolgens op. Sleep dit blokje ook naar de scriptzone en klik het onder het andere blokje, waarmee de kat naar het midden wordt bewogen. Zorg dat de twee blokjes in elkaar klikken, zoals in de figuur hiernaast.



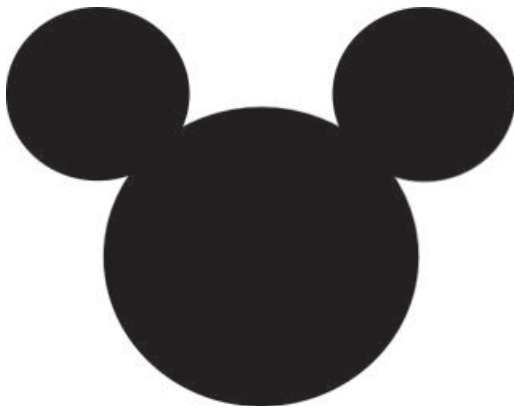
We hebben er nu in feite één gecombineerde instructie van gemaakt. Als je in de Legodoos nog even met de knopjes speelt, de kat alle kanten op laat draaien en ergens

neerzet, met een druk op de gecombineerde instructie in de scriptzone, kun je de kat weer in zijn gewenste uitgangspositie brengen. Daarom noem ik het plaatje ook 'startpositie'.

Het is in het programmeren een goede gewoonte de uitgangssituatie expliciet vast te leggen. We zouden dat achterwege kunnen laten, maar dan kunnen tien mensen hetzelfde programma draaien en met telkens een andere uitkomst komen. Dit maakt het testen van programma's erg lastig.

Nog een keer... wat kunnen we ermee?

We hebben nu de kat wat elementaire bewegingen laten uitvoeren, te weten stappen, verplaatsen, draaien. Als we dit nou handig doen, zouden we de kat ook rondjes kunnen laten draaien. En als we dat dan zo doen dat de kat een spoor achterlaat, dan zouden we een figuur kunnen tekenen.



Voorwaar, niet echt iets om heel warm voor te lopen, maar dan hebben we een begin van een tekening. En als je de Donald Duck leest, dan weet je dat je niet veel meer dan drie cirkels nodig hebt om Mickey Mouse te tekenen. Dus laten we daar maar eens mee gaan beginnen.

Maar eerst even dit

We hebben nu de instructies uitgevoerd door op de blokjes te klikken. De instructie wordt dan uitgevoerd zoals die daar staat, ook gecombineerde instructies. Maar ik heb in het vorig nummer over de groene vlag gesproken waarmee je je programma zou kunnen starten. Nu hebben we een gecombineerde instructie in de scriptzone staan, maar dit is nog niet een programma dat door de vlag wordt bestuurd. Probeer het maar, met de scriptzone zoals hierboven weergegeven, draai en verplaats de kat en druk op het groene vlaggetje in ons speelveld. Je ziet, er gebeurt niets. Probeer het gerust nog een paar keer, als het goed is zie je geen verschil.

Waarom nog een paar keer proberen? Om het goed te laten bezinken, dat een programma normaal gesproken elke keer op dezelfde wijze reageert. En nu nog even dit. Als je een in elkaar geklikte reeks van instructies wilt verplaatsen, pak je het bovenste blokje.

Daarmee sleep je het geheel alle kanten op. Maar wil je een blokje eruit halen, dan moet je dat blokje even wegschrijven. Daarmee schuift alles eronder even mee. Daarna schuif je alles wat je wilde behouden weer terug onder het eerste blokje. Je schuift dus steeds al het onderliggende mee, maar alles wat erboven staat blijft staan.

Als je een overbodig blokje in je scriptzone over hebt, dan kun je dat gewoon in de Legodoos terugzetten. Je sleept het naar de Legodoos en de ingebouwde butler zorgt dat het blokje weer in het juiste vakje komt te liggen; je hoeft je dus niet druk te maken om het weer op de juiste plek te leggen, dat komt allemaal goed. Opruimen van Lego is een heel gedoe, opruimen van Scratch is heel eenvoudig. Bovenin de Legodoos zie je een aantal blokjes met een kleurcode; we hebben al die tijd gewerkt met 'blauw - beweging'. We gaan nu de reactie op de groene vlag maken. Klik nu op de groep 'gebeurtenissen', kleur is een beetje donker oranje/bruin. Bovenaan staat 'Wanneer de groene vlag wordt aangeklikt'. Plak dat blokje boven de code die we al hadden staan, de 'startpositie'. Dat ziet er uit als volgt:



Figuur 3 - startpositie met trigger

Nu hebben we in de scriptzone een trigger ingebouwd die op het vlaggetje reageert. Zet nu de kat ergens weg, draai hem in een bepaalde richting en klik dan op het groene vlaggetje in het speelveld. En zie, de instructie zoals hierboven weerge-

geven, de startpositie, wordt ingenomen door de kat. Hiermee hebben we een begin gemaakt voor ons echte programma. Dit is een leuke basis. Alleen, ik zou nog willen, als we straks een cirkel gaan tekenen, dat die niet het scherm afloopt. We gaan dus de startpositie nog even aanpassen. Klik op de y-positie in het 'ga naar' blokje en vul hier 100 in. Klik vervolgens op de groene vlag en de kat staat ergens bovenin het veld.

Opslaan

Nu is het tijd om ons eerste project op te slaan, zodat we hier altijd naar terug kunnen grijpen. Ik ga er even vanuit dat jullie de online versie gebruiken. Net boven het speelveld zie je een invulvakje waar nu iets als 'untitled' in staat. Daar vul je nu een naam in die je later kunt herkennen, ik noem hem zelf even 'startpositie'.

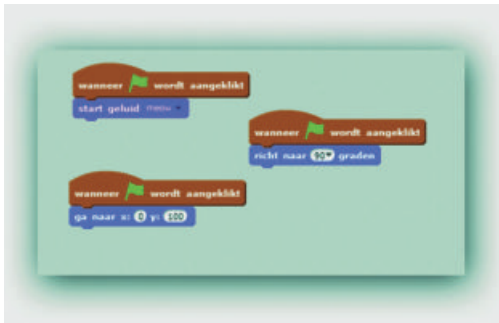
Vervolgens klik je in het menu 'Bestand' op 'Nu opslaan' en je 'project' is later weer oproepbaar. Omdat je alles opslaat bij MIT is je project ook weer beschikbaar als je op een andere computer, of misschien later een tablet of telefoon, verdegat. Voor de oudere generatie is de 'cloud' misschien nog iets onwettelijks, maar met (klein)kinderen werk je met Scratch hier al.

Een kleine noot nog. Tijd voor koffie, maar los daarvan, ik had het over definiëren van startpositie. Maar daar zijn we uiteraard niet helemaal compleet in geweest; we hebben nu de standaard kat, een standaard wit speelveld, enz. In goede programma's wordt de startpositie veel uitgebreider beschreven, maar vooralsnog volstaat dit wel en kunnen we deze startpositie steeds weer oproepen als we een nieuw programma gaan maken.

Moderne technieken

Vorige keer heb ik al laten zien dat er in Scratch ook opdrachten voor geluid aanwezig zijn. Daarmee wil ik gelijk iets anders laten zien. We gaan daar later een keer uitgebreider op in, als we met meerdere sprites gaan werken. Zo ver zijn we nog niet, maar alvast een concept. Klik eens op het thema 'Geluid', de paarse commando's. En daar staat bovenaan 'start geluid ..' en erbij is 'meow' ingevuld. Dat is een miauw en het enige geluid dat standaard meegeleverd wordt. Klik er maar eens op. Je hoort dan de kat, alsof je 'm op zijn staart hebt getrapt. Schuif dit blokje onderaan onze commando's in de script-area.

Klik weer op het groene vlaggetje nadat je de kat weer hebt laten draaien, bewegen. Je merkt dan, dat de kat miauwt, terugdraait en zich weer naar zijn startpositie begeeft. Je mag aannemen, dat hij dat in de genoemde volgorde doet, maar met een beetje snelle computer is dat niet uit te maken, het lijkt allemaal parallel te gebeuren. Wat je ook kunt doen om het nog meer parallel te laten gebeuren, is deze drie opdrachten uit elkaar trekken (onderaan beginnen, uiteraard, anders schuift alles mee) en dan bovenaan elk blokje uit de gebeurtenissen het blokje 'wanneer



Figuur 4 - parallelle verwerking

acties worden weer uitgevoerd. Wat je in feite nu hebt gedaan, hoewel je het verschil nog niet merkt, is drie 'threats' gestart, die elk één taak voor zijn rekening neemt. Nu merk je het verschil nog niet, het is nu nog een nutteloze toevoeging, maar het biedt alvast zicht op wat er straks zoal mogelijk is.

Met de druk op de groene vlag kunnen we meerdere activiteiten tegelijk laten starten. En elk van die activiteiten kunnen we met de benodigde controlemiddelen straks door elkaar laten lopen. En dat biedt mogelijkheden om bijvoorbeeld spelletjes te spelen, waarbij de speler één figuur bedient en de computer een andere figuur. En die hoeven dan niet op elkaar te wachten, ze kunnen beide onafhankelijk van elkaar bewegen.

In feite is hier natuurlijk een limiet aan de mogelijkheden van de computer, maar vooralsnog, met de meeste computers en de eenvoud van onze programma's nu, lijkt dat toch echt op puur parallelle verwerking. En dat is iets wat we indertijd, toen we nog met Basic of Pascal aan het spelen waren, niet zomaar voorhanden hadden. En hier zit het gewoon in de Legodoos, alsof je twee motortjes ter beschikking hebt, of drie, of honderd.

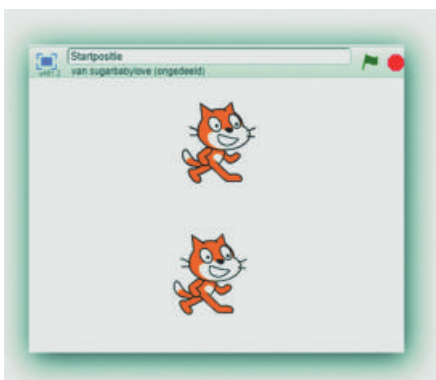
We halen weer even de twee triggers weg, of laten ze ergens liggen en hangen weer de drie opdrachten onder de overgebleven trigger zoals in figuur 3, maar dan nog met de miauw eronder (of ertussen). We slaan ons project weer op, we hebben een uitgebreidere startpositie nu. Het begin van de run wordt gemarkeerd door een miauw.

Stempelen

We kijken nu even naar de categorie 'Pen', in het groen. En we klikken op 'stempel'. Zie je wat er gebeurt? Als je niets ziet, probeer het gerust nog een keer. Het lijkt erop, alsof dit niets doet. Maar schuif nu de kat eens ergens heen. Er blijft een afbeelding van de kat staan waar hij stond, maar de kat gaat ook ergens anders heen, daar waar je hem heen sleept. De 'sprite' kun je bewegen, de stempel is een afdruk van de kat op het speelveld, waar je verder nog niets mee kunt. Als je op het groene vlaggetje drukt, dan zie je nog maar één kat, op de startpositie, maar dat komt, omdat de katten (de sprite en de stempel) exact over elkaar heen vallen, tenminste, als je vanuit de startpositie op 'stempel' hebt gedrukt.

Beweeg nu de kat naar beneden, zodat je twee katten op je scherm hebt staan. Het ziet er nu uit als in figuur 5.

Druk nu weer op 'stempel', verplaats de kat weer en druk nog een keer op



Figuur 5 - twee katten

de groene vlag wordt aangeklikt' plaatsen. Het ziet er dan ongeveer uit als in figuur 4: Vervolgens rommel je weer wat met de kat, maar daarna klik je op de groene vlag en alle

'stempel'. Klik dan weer op het groene vlaggetje. We kunnen dus overall een afbeelding maken van de kat. Klik nu op 'wis alles' en je zult zien, je hebt één kat op je scherm. Verplaats hem en er zijn geen afbeeldingen meer over, je hebt alleen de sprite nog, die je overall neer kunt zetten.

Dit 'wis alles' is een goede actie om toe te voegen aan onze startpositie, zodat we die langzaam maar zeker uitbreiden tot een redelijk compleet geheel. Dit is van belang opdat we straks de uitkomsten van een programmeren goed met elkaar kunnen vergelijken.

Sla het project vervolgens weer op. Je kunt je blokje gewoon onder de andere blokjes plaatsen, je kunt het er ook tussen persen. Wat je niet kunt is, het bovenop de trigger zetten. Logisch, want wat heeft dat voor zin, de trigger activeert alles eronder! Daarom ook biedt de trigger niet zo'n mooi vormpje waar een andere opdracht in past, aan de bovenkant is de trigger glad en rond. Je kunt er aan de bovenkant niets aan vastmaken.

Klik nu op 'pen neer'. Let er even op dat de penkleur niet wit is. Anders klik je op het blokje 'maak penkleur ..' nadat je er bijvoorbeeld '5' in geschreven hebt. Dan is de waarde in elk geval niet wit.

Sleep nu de kat ergens heen, gewoon door 'm op te pakken en ergens neer te zetten. Je ziet nog niets ongewoons. Laat nu de kat eens bewegen met de stapjes, en je ziet dat de kat een spoor trekt in de kleur van de pen. Als je, terwijl de pen 'neer' is op het groene vlaggetje drukt, dan zie je de kat weer naar de uitgangspositie gaan, maar hij trekt een lijn van waar hij begon. Probeer het maar. Als je daarna de kat weer oppakt en hem ergens anders neerzet en het groene vlaggetje activeert, dan zie je een nieuwe lijn. De oude lijn is weg, want in onze uitgangspositie hebben we alles gewist. Dat wil zeggen, alle stempels, maar ook de getekende lijntjes.

Zoals ik eerder al aangaf, het is handig om in je startpositie zoveel mogelijk zaken op te nemen die later verrassingen kunnen voorkomen. Daarom beginnen we onze startpositie ook met 'pen op'. En de volgorde maakt nu wel uit, we willen de pen omhoog voordat we bewegen, of we willen alles wissen als we op onze plek zijn. Als je hier de volgorde niet goed doet, heb je kans dat er toch nog een lijntje op het scherm blijft staan als we met de besturing gaan beginnen.

Tekenen

Aan het eind van dit tweede artikel willen we, nadat we de uitgangspositie hebben opgeslagen, een tekening maken. Heel eenvoudig, gewoon een figuur, zoals een vierkant of een driehoek, of zeskant. Verzin maar iets. Ik ga iets proberen dat een cirkel zou moeten benaderen.

Om dit te doen, moet ik een lus inbouwen om een set van instructies herhaaldelijk uit te voeren. Daarvoor ga ik naar 'besturen', de lichtbruine kleur, en kies 'herhaal .. keer'. Ik wil een cirkel tekenen, althans, benaderen, dus ik kies voor telkens een heel kleine draaiing, zeg 1 graad.

Binnen de lus van 'herhaal' neem ik twee instructies op, te weten 'draai rechtsom .. graden' met voor mij dus 1 graad. Daarna de instructie 'neem .. stappen' maar om een cirkel te maken moeten die stappen ook heel klein zijn, minimum is 1. En voordat ik dit kan laten tekenen, moet ik aan het begin nog even de 'pen neer' uitvoeren. Dit kan net boven de lus, dat hoeft er niet in, want dat kost verwerkingstijd. Daar gaan we de volgende keer dieper op in.

Succes met je eerste tekening!

Scratch (3)

René Suiker

Wat vooraf ging...

En zo is het alweer 2019 en is het tijd voor aflevering 3. Een korte recapitulatie.

In deel 1 gaf ik een inleiding in scratch. In deel 2 gaf ik daar een aanvulling op en begonnen we met programmeren. In deze aflevering gaan we hiermee verder, maar ga ik ook nog even in op wat andere overwegingen.

We gingen in op sprites, op events, op lussen, op pen op en neer, in feite op de basiselementen om een programma te maken dat reageert op prikkels van de omgeving en dan vervolgens enige logica hanteert.

En nu gaan we ook eens tijd maken voor 'anderen'. We hadden het al aangekondigd, Scratch was leuk om met kinderen te doen. Liefst niet te grote kinderen, want die zullen het misschien weer te kinderachtig vinden, maar leeftijd basisschool is zo'n beetje waar je aan kunt denken. Hoewel er natuurlijk ook altijd kinderen buiten die categorie zijn die het wel leuk vinden. En binnen de categorie heb je ook kinderen die er niets aan vinden. Het blijft natuurlijk iets wat u het best voor uw (klein)kinderen kunt aanvoelen, er is geen universele waarheid.

Met anderen

Zo ving ik op het werk wat op in een gesprek tussen collega's over Scratch, van wie er een enthousiast over aan het vertellen was, dat kinderen het heel leuk vonden. En dat er bij hem in de buurt ook bijeenkomsten waren waar het uitgelegd werd, waar kinderen konden oefenen met computers die al klaar stonden en dat ze met vragen geholpen werden. En zo maakte ik kennis met het fenomeen 'Coder(Dojo)'. Om eens een kijkje te nemen ga ja naar:

<https://coderdojo.nl>

Misschien is dit iets om als CompUsers mee samen te werken, want dan staat de nieuwe generatie doelgroep gelijk klaar. Maar dat moeten we dus nog even uitzoeken. Ze zijn in elk geval al verbonden met wat andere instanties, dus waarom niet CompUsers, of HCC. We zouden zelf een Dojo kunnen opstarten, als één van de mogelijkheden.

Ik ga natuurlijk niet dit hele artikel aan CoderDojo besteden, en CoderDojo is niet alleen Scratch. Maar het is toch wel interessant genoeg om er even bij stil te staan. Want we zien in de vereniging de veroudering toeslaan, maar jongelui worden heel enthousiast van zelf programmeren.

Van hun site citeer ik:

CoderDojo is een open source, door vrijwilligers opgezette, beweging, die staat voor het houden van gratis, non-profit development clubs en reguliere sessies voor jonge mensen. Omdat CoderDojo open source is kan elke Dojo anders zijn en volledig los staan van andere Dojo's.

Bij een Dojo leren jonge mensen, tussen de 5 en 17 jaar, programmeren, websites bouwen, apps ontwikkelen, programma's maken, games maken, en meer. Dojo's worden opgezet en georganiseerd door vrijwilligers. Naast het leren programmeren, ontmoeten deelnemers ook andere, like-minded¹ mensen, delen deelnemers waaraan ze aan het werken zijn en maken ze nieuwe dingen. CoderDojo maakt development en leren programmeren een leuke, sociale en 'kick ass'-ervaring². CoderDojo legt ook veel nadruk op open source en gratis software en heeft wereldwijd een breed

netwerk aan vrijwilligers. CoderDojo heeft slechts één regel: 'Boven Alles: Wees Cool!', pesten, liegen en mensen hun tijd verdoen is niet cool. Om aan een Dojo mee te doen kun je er een bij jou in de buurt zoeken of er zelf een opstarten.

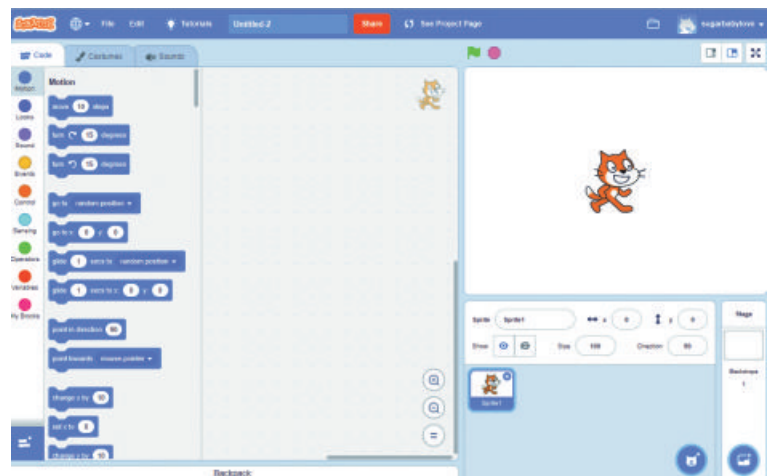
Oftewel, wat wij ook doen, met vrijwilligers, gratis leuke dingen doen met de computer. Ze werken met Scratch, dat doen wij ook, ze werken met HTML, JAVA, zaken waar je ook bij ons voor terecht kunt als het Platform WebOntwerp weer opgericht is, ze werken met Arduino, dat je bij ons bij Domatica aantreft. Enfin, hoezo nou vergrijzing? Kijk eens wat er daar gebeurt, kijk ook eens op <https://coderdojo.com/nl-NL> en neem je (klein)kind bij de hand en ga leuke dingen doen. En, niet onbelangrijk, de kennis en vaardigheden die ze hierbij opdoen zijn mogelijk heel belangrijk voor hun toekomst.

Goed, genoeg reclame, we gaan zelf weer aan de slag, want daar is het toch om begonnen.

Scratch zit niet stil

We starten Scratch weer op via <https://scratch.mit.edu/>

We worden meteen geconfronteerd met het nieuws dat er een nieuwe versie is uitgekomen: versie 3.0. En dat heeft wat te betekenen, want zo is bijvoorbeeld de hele werkomgeving omgegooid.



Afbeelding 1 - Vernieuwde werkomgeving

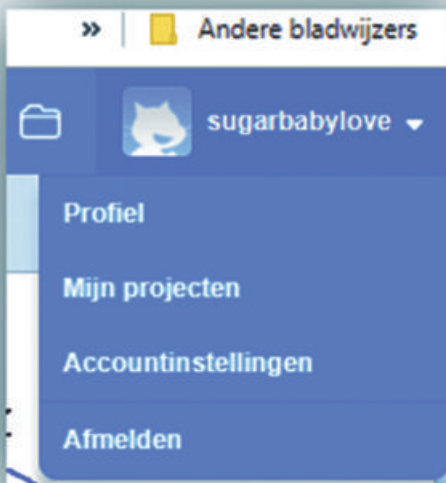
Het idee is nog wel hetzelfde, maar de positionering van de onderdelen is gewijzigd. De bibliotheek van alle commando's bevindt zich nu links. En het schijnt dat er de nodige blokken bij zijn gekomen in deze nieuwe versie. En helemaal links zie je dan de categorieën, daarnaast de actuele opdrachten. Dan zie je in het midden het werkgebied. Daarboven zie je de tabbladen, in dit geval voor 'code', 'uiterlijken' en 'geluiden'. Normaal gesproken staat hij op 'code' en kun je in je werkgebied je programma bewerken. Maar je kunt dus ook de geluiden bewerken en de sprites of achtergronden bewerken. Overigens, als je een ander tabblad kiest, dan verandert ook het gedeelte links. De hierboven beschreven opdrachten zijn aan het werkblad 'code' gekoppeld. Kom je in de grafische editor of ik de geluidseditor, dan zie je links de geluiden of grafische elementen die je kunt bewerken. Rechts daarvan is het uitvoeringsgebied, ofwel speelveld, waar de acties uitgevoerd worden. Zoals jullie ongetwijfeld nog weten, kun je dat gebied ook maximaliseren, dan wordt het beeldvullend.

Onder het speelveld zie je een gebied waar je alle sprites kunt opslaan die je wilt gebruiken. Daarnaast nog een stukje met de naam 'speelveld', waar je je achtergronden bewaart.

Wat me de vorige keer niet opviel, dus misschien was het er ook niet, maar dat weet ik niet zeker, is dit: in het werkgebied, nu weer terug op 'code', zie je rechts bovenin een gebleekte afbeelding van de geselecteerde sprite. Dat geeft aan dat de ingevulde code betrekking heeft op die specifieke sprite. Daar gaan we straks verder op in.

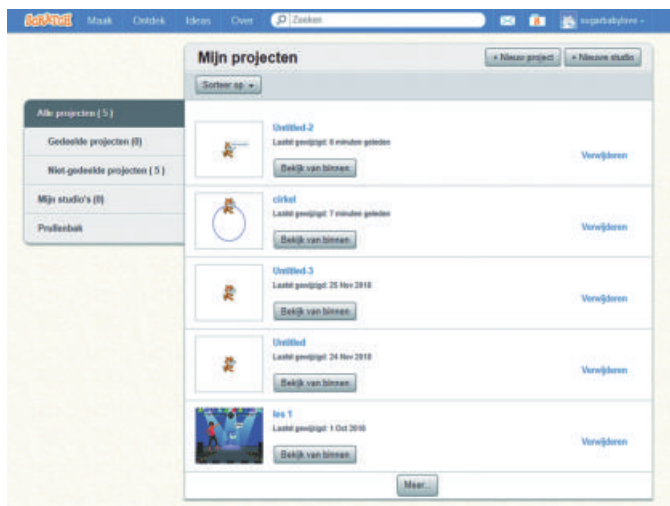
Mijn project

Vorige keer ben ik geëindigd met mijn projectje, een programma dat een cirkel ging tekenen. Dat was nog in Scratch 2.0. Een mooie proef; hoe zou die nu laden, in Scratch 3.0? Welnu, dat viel reuze mee, want ik kon mijn programma gewoon openen. Rechts bovenin klik ik op het driehoekje naast mijn gebruikersnaam en vervolgens klik ik dan op 'Mijn projecten'.



Afbeelding 2 - Mijn projecten

Je krijgt dan een overzicht van je projecten en daaruit kun je er dan één selecteren. Let wel, in principe zijn deze projecten in de cloud van MIT opgeslagen, je kunt er dus ook bij als je, bijvoorbeeld op de CompUfair, hier verder aan wilt werken.



Afbeelding 3 - Mijn projecten voor CompUsers

Ik zie hier de projecten die ik in het kader van dit artikel heb gemaakt, of waar ik aan begonnen ben en ik open project cirkel, gemaakt in Scratch 2.0, maar dat probleemloos opent in 3.0. Overigens hoeft dat niet voor al uw projecten te gelden, er zijn wat problemen bekend, maar daarvoor verwijs ik naar de website. Overigens krijg je bovenstaand overzicht ook als je op het mapjes-icoon klikt. Uiteraard ziet het er nu wel iets anders uit, want de wijziging in de omgeving heeft zijn invloed:

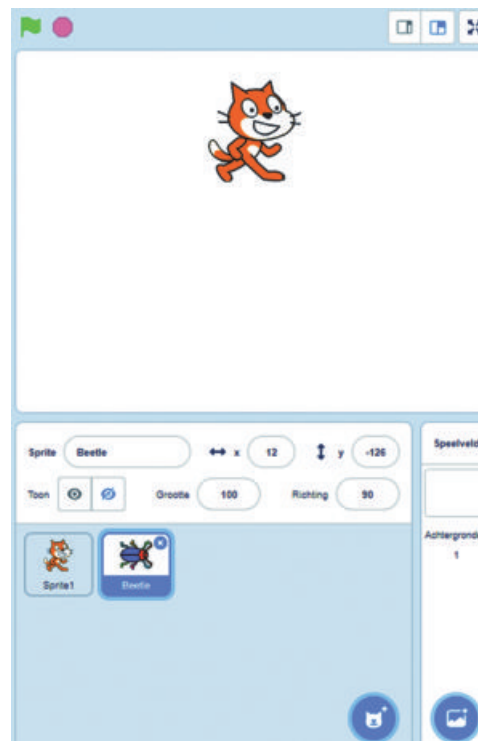


Afbeelding 4 - Project 'cirkel'

U ziet geen meldingen en als ik op het groene vlaggetje klik hoor ik een miauw en gaat de kat een rondje lopen, ondertussen een cirkel tekenend. Tot zover geen probleem. De oplettende lezer die nog goede ogen heeft is misschien iets opgevallen? Een verschil tussen figuur 1 en figuur 4? Ik heb tussendoor de taal weer even op Nederlands gezet. Ik weet niet of het met versie 3 of gewoon met verstreken tijd of iets anders te maken heeft, maar hij was weer naar Engels teruggesprongen. Weet u het nog? Het wereldbolletje naast het SCRATCH-logo links bovenin. Daar kunt u de taal selecteren.

Wat misschien ook opvalt is dat in het speelveld de kat staat, maar eronder, in het sprite veld, staan twee figuren opgesteld. Eén ervan staat zichtbaar in het speelveld, de kat dus, de ander staat nog niet zichtbaar in het speelveld, maar is wel beschikbaar.

We kijken eens naar het sprite-gedeelte, want dat was mijn bedoeling.



We kijken er eens goed naar. We zien nu twee sprites, waarvan er één geselecteerd is: die in het blauw. Daarboven zie je de naam van de sprite. In dit geval staat er bij de kat 'Sprite 1' of zoiets en bij het insect zie je 'Beetle' staan. Boven de figuurtjes zelf zie je de details van de geselecteerde figuur.

Afbeelding 5 - Sprite-veld bij het speelveld

Hier kun je details ook aanpassen. Zo kun je de naam van de kat veranderen, maar je kunt ook zijn positie veranderen, door andere x- en y-coördinaten op te geven. Verder kun je de loopricting ook nog aanpassen. Als je op het veldje 'richting' gaat staan, dan wordt er een cirkel getoond met daarin de loopricting aangegeven.



Afbeelding 6 - Looprichting visueel

Standaard staat de loopricting op 90 graden. Zoals vorige keer besproken: dit is de richting waarin de sprite beweegt als er stappen genomen worden.

Onder de naam van de sprite kun je aangeven of hij zichtbaar moet zijn in het speelveld, of juist onzichtbaar. Daarnaast kun je de grootte nog aanpassen. Als je dit doet, dan zie je gelijk het effect in het speelveld. Als je de kat bijvoorbeeld in plaats van 100 de waarde 10 geeft, dan wordt het echt een speldenpuntje.

Onderin het veld is nog een knop om extra sprites toe te voegen.

In mijn originele cirkelprogramma had ik de kat een rondje laten lopen, terwijl hij een cirkel tekende. Als je nu de kat onzichtbaar maakt, dan kan het programma nog steeds lopen, maar dan zie je alleen maar de cirkel getekend worden, en niet de sprite.

Vorige keer had ik een aantal zaken geregeld aan het begin van de programma-uitvoering, waaronder de startpositie. Hoe meer je leert tijdens het ontdekken van Scratch, des te uitgebreider wordt ook je set van commando's aan de start van je programma. Want als ik per se zeker wil weten dat de kat ook zichtbaar is, dan moet ik in mijn programma zorgen dat hij verschijnt, ook als de gebruiker op dat moment de sprite niet zichtbaar heeft staan. Gelukkig hebben we daar ook een commando voor, in het blok 'uiterlijkheden', het paarse blok.

Ik plaats dus het blokje 'verschijn' in mijn code en wel ergens bovenin, zodat hij in elk geval niet binnen de lus telkens aangeroepen wordt. Want elke instructie kost tijd. Het is dus zaak goed na te denken over wat je binnen een lus zet en wat niet.

Samenwerken

Ik heb mijn programma nu zo aangepast dat de twee sprites ieder een halve cirkel gaan tekenen. Daartoe kopieerde ik het codeblok van de kat naar het codeblok van het insect. Daarna paste ik de beginpositie aan en de aanvangsloopricting. In beide stukjes code stelde ik het aantal herhalingen in op de helft, omdat ze nu nog maar ieder een halve cirkel hoeven te tekenen. De eerste keer mislukte natuurlijk, omdat ik niet goed had opgelet hoe groot de cirkel uiteindelijk werd, dus sloten de delen niet goed aan. De straal

was wel precies hetzelfde, maar ik moest de positie in de code aanpassen.

Iets meer over de blokken

De code voor de kat ziet er nu zo uit:



Afbeelding 7- Code voor de kat

Je kunt voor jezelf ongetwijfeld nagaan wat al die blokken precies doen, die kennis moeten jullie intussen hebben. En als je de startpositie van onze tweede sprite wilt weten, dan moet je kijken waar de kat ophoudt. Als die twee posities overeenkomen, dan gaat het precies goed.

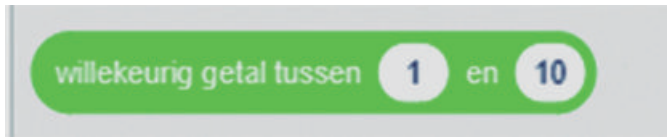
Nog even over de blokken zelf. Naast het kleurverschil valt vast nog iets op. Er zijn verschillende vormen te zien. De meest voorkomende vorm is het blokje met bovenin een kuiltje en onderin een uitstulping. Deze passen in elkaar. De blokken in deze vorm worden wel de 'stapelblokken' genoemd.

Daarnaast zie je hierboven ook een bijzondere soort stapelblok, dat een blok boven en onder kan hebben, net als een normaal stapelblok, maar dat ook een blokje (of een set blokken) er tussenin kan hebben. Dit is de lus.

Verder zie je hier in de blokken de witte velden, waarin je waarden kunt invullen. Als dat een afgerond vakje is, een cirkel of een ovaal, dan wordt een numerieke waarde verwacht. Is het een rechthoek, dan kun je ook een tekst invoeren. Je kunt dan ook een nummer invoeren, maar dat wordt in principe als tekst geïnterpreteerd.

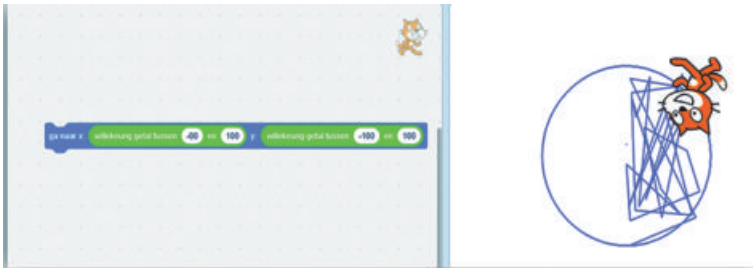
Dan zie je bovenaan nog het blokje met een uitstulping naar boven en naar beneden. Hier kunnen dus wel blokken onder worden gezet, maar niet erboven. Deze blokken starten de uitvoering van de onderliggende code, als de gebeurtenis

welke het blok benoemd wordt uitgevoerd. Overigens is de kracht van programma's natuurlijk dat je niet altijd alle waarden van tevoren invult. Variabelen gaan we de volgende keer bespreken, maar je hebt nog een paar andere vormen, en één ervan is een afgerond blok, zoals dit:



Afbeelding 8 - Afgerond blok

Zo'n afgerond blok kun je plaatsen binnen zo'n wit cirkeltje. Het kan er bijvoorbeeld bij een startpositie zo uitzien:



Afbeelding 9 - Willekeurige startpositie

Ik weet niet of het plaatje heel leesbaar wordt, maar ik heb dus even een blokje gepakt, zoals in figuur 7 wel duidelijk leesbaar is, met een 'Ga naar x: ... y: ...' invullen (blauw blokje - beweging). Daarbinnen heb ik voor x en voor y het groene blokje van figuur 8 ingevoerd en dit heb ik als zelfstandige instructie uitgevoerd. Omdat de pen nog naar beneden stond, kun je in het speelveld mooi het spoor volgen dat de kat heeft afgelegd tussen de diverse punten. Ik had de willekeurige waarden laten variëren tussen -100 en 100.

U had misschien al opgemerkt dat er ook een blauw blokje bestaat met de opdracht 'ga naar willekeurige positie'. Dat heeft niet helemaal hetzelfde effect, probeer het maar eens uit. Het effect is, dat willekeurig ook echt willekeurig is, en met de willekeurige getallen kun je de willekeurige begrenzen. Er zijn nog meer varianten die we in de loop van de tijd gaan bekijken.

Maar eerst nog even een klein zijspiongetje. Ik zei net zo makkelijk: 'kopieer die code van de kat naar ons insectje'. Om dat eenvoudig te doen hebben we helemaal onder in beeld een rugzak. Die kun je gebruiken om stukken code op te slaan en op een andere plek weer te gebruiken. Had ik al verteld, dat je oneindig veel blokjes hebt, dus instanties van hetzelfde blokje? Als je iets uit je rugzak gebruikt, blijft het er ook nog eens gewoon in zitten. Erg handig. Ik spring natuurlijk van de hak op de tak, maar dat komt omdat hier geen expert aan het woord is die even uitlegt hoe het werkt, maar iemand die jullie deelgenoot maakt van een ontdekkingsreis door Scratch. Ik weet nog lang niet alles, maar ik word wel steeds enthousiaster om ermee door te gaan. En ik nodig jullie uit om hier ook zelf in te duiken, misschien eerst aan mijn hand, maar toch steeds meer op eigen benen.

De eerste stappen

De eerste stappen zijn misschien wel het moeilijkst, want je begint weer aan iets nieuws, en daar voel je je misschien te oud voor. Maar zie het als terug naar je jeugd, met Lego, Meccano of Fischertechnik spelen, maar dan op je computer. Je wordt er weer jong van, de computer doet weer dingen die je hem opdraagt.

We zijn natuurlijk van de applicaties, de toepassingen. Maar met domotica sturen we in huis van alles aan, via zo'n Ardui-

no-apparaat. Die schijnen ook met Scratch bestuurd te kunnen worden, in plaats van met alleen maar ingewikkelde talen als C++.

Huiswerk

Intussen heb ik zonder dat je het misschien helemaal door had, al een aardige inleiding Scratch bijelkaar geschreven. En daar ga ik ook mee door, maar ik wil niet elke keer alles moeten herhalen, dus ga eens proberen of je het nu ook kunt.

Ik wil op een komende CompUfair een keer een Scratch-workshop verzorgen, waar we zelf kunnen gaan spelen. Ik zal daar wel e.e.a. moeten toelichten, maar het zou fijn zijn als dan sommige lezers al een beetje verder zijn en mij bij kunnen staan.

Daarom een paar eenvoudige opdrachten, om eens te kijken of er nog iets is blijven hangen.

Vorbereiding

Regel een account bij MIT, waardoor je je projecten ook op kunt slaan. Dit kost niets en is in een mum van tijd geregeld.

Opdracht

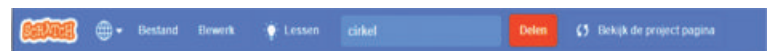
Maak een programma dat zonder zichtbare sprites ergens in het midden van het scherm drie concentrische cirkels tekent.

Punten van aandacht

- Programma wordt gestart door het groene vlaggetje
- Startpositie voor het programma, wat moet er allemaal aan- en uitgezet worden?
- Hoe teken ik een cirkel?
- Wanneer ben ik klaar?
- Werkt het programma, doet het wat het moet doen?
- Hoeveel regels code heb je nodig gehad? Kan het met minder?
- Hoe lang duurt de uitvoering? Kan het sneller?
- Zijn aanpassingen op (6) en (7) aanvullend, of werken ze elkaar tegen?

Ik hoop dat jullie hier een uitdaging in zien en ik zie graag wat stukken code uit jullie handen komen.

En om nu te voorkomen, dat we allemaal screenshots met elkaar gaan delen, nog als toegift het volgende. Bovenin je Scratch-venster zit een knopje 'Bekijk de projectpagina'. Als je daarop klikt, dan krijg je de mogelijkheid je project te delen. Het maakt niet uit dat je een heel eenvoudig project hebt gemaakt, er staan miljoenen heel eenvoudige projecten online, en niemand die daar aanstoot aan neemt.



Afbeelding 10 - Naar de projecten pagina

Via <https://scratch.mit.edu/projects/264866296/> kunnen jullie mijn cirkel zien. Gewoon de URL kopiëren, dan kun je die aan anderen beschikbaar stellen. Dat is nog iets anders dan dat een ander het ook kan wijzigen, maar je kunt ze wel naar de inhoud laten kijken. Dat heet dan 'Bekijk van binnen'.

Veel plezier weer met Scratchen.

1. like-minded = gelijkgestemde
2. ass-kicked = ruig, wild, baldadig (Am.-Eng. slang)

Scratch (4)

René Suiker

Alweer de vierde aflevering, maar gaan we er mee verder? De vraag is natuurlijk of deze reeks in een behoefte voorziet. Heel veel terugkoppeling heb ik niet gekregen. Het was wel een grote tegenvaller dat ik helemaal geen huiswerk heb ontvangen, terwijl de oproep toch onmiskenbaar was.

Geen terugkoppeling krijgen, dat gebeurt vaker, uiteindelijk schrijf je wel voor anderen, maar toch vooral omdat je het zelf leuk vindt. En of anderen het waarderen, daar kom je alleen maar achter als je je werk probeert te 'verkoppen'.

Huiswerk

De reden om huiswerk op te geven was om enige interactie mogelijk te maken, maar toch ook als inleiding om nu echt te gaan programmeren: problemen oplossen d.m.v. instructies aan een computer.

Nu is dit niet de officiële definitie misschien, maar wel één die ik hanteer, omdat het een praktische is.

De bedoeling is om nu met behulp van Scratch echt te gaan programmeren en de eenvoudige opdracht in het huiswerk leek me een goed begin.

Maar, ofwel het was te simpel, ofwel het was te hoog gegrepen, ofwel u had echt geen tijd. Of, niemand heeft het gelezen. Wat de echte reden ook is, ik probeer het gewoon om jullie toch mee te nemen op mijn reis door Scratch.

Vraagstelling

Het huiswerk was vrij eenvoudig, althans de opgave:

- Het programma start met de vlag;
- Het programma tekent drie concentrische cirkels, ongeveer midden in het beeld;
- De sprite die het tekent is onzichtbaar.

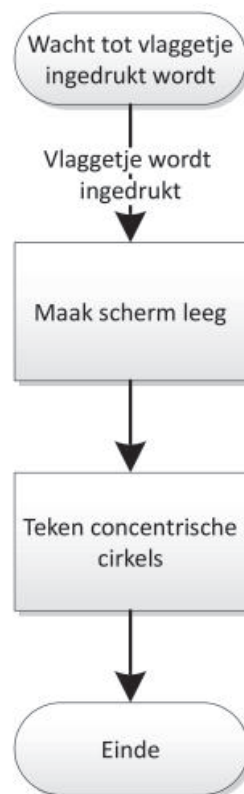
Verder gaf ik nog wat tips en aandachtspunten mee om jullie op weg te helpen.

We gaan nu niet een hele cursus programmeren opnemen, maar het is altijd verstandig het probleem dat te groot lijkt om in één keer op te lossen, te verdelen in deelproblemen die wel oplosbaar zijn.

Vroeger leerden we dit met stroomdiagrammen, maar die zie ik niet zo vaak meer. Toch is het een handige manier

om grafisch uit te leggen wat er moet gebeuren. Zo iets zou er ongeveer zo uit kunnen zien:

En als je dit nog niet in Scratch kunt uitwerken, dan kijk je of je een blokje weer kunt onderverdelen in kleinere elementen, die wel te maken zijn.



Figuur 1 - Stroomschema huiswerk

Zo kan ik me voorstellen dat hier alleen een uitdaging zit in blokje 3, het tekenen van de cirkels. Want: we hebben in de legodoos die Scratch heet niet een opdracht om een cirkel te tekenen. Laat staan drie. En wat betekent concentrisch nu helemaal?

Welnu, concentrisch betekent dat de cirkels allemaal hetzelfde middelpunt hebben. Verder wordt er wel expliciet bij gedacht dat ze een verschillende straal hebben, anders vallen ze precies over elkaar heen en zie je niet dat het er meer dan een zijn.

Een mogelijke (lineaire) oplossing is hieronder weergegeven in figuur 2.



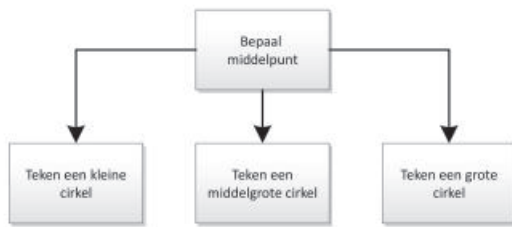
Figuur 2 - Teken concentrische cirkels - lineair

Intussen hebben we gezien dat je in Scratch ook meerdere processen tegelijk kunt laten lopen.

Dus een alternatieve manier om de cirkels te tekenen kan zijn als weergegeven in figuur 3.

Uiteraard mag je een voorkeur hebben, maar normaal gesproken vinden beginners de lineaire methode eenvoudiger. En de parallelle methode kan sneller zijn als je meerdere processors tegelijk aan zo'n opdracht zou kunnen zetten, maar bij elementaire programma's als dit zal dat geen meetbaar verschil opleveren.

In het kader van dit artikel ga ik beide manieren uitwerken. En als je dit ook doet, dan kan je kijken of je een meetbaar verschil hebt.



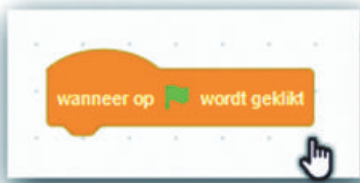
Figuur 3 - Concentrische cirkels - parallel

Op zich hoogstwaarschijnlijk niet, maar je kunt natuurlijk kijken of je, als je het geheel 10.000.000 keer herhaalt, wel een verschil kunt meten. Want dat is vrij makkelijk te regelen: herhalen is iets wat een computer heel goed kan. Kijk nog eens terug naar figuur 1, het stroomschema. En start intussen ook Scratch maar weer op. Als het goed is, staat die intussen in 'favorieten' of 'bookmarks' van de browser, maar voor alle zekerheid geef ik 'm hier nog maar een keer op:

<https://scratch.mit.edu>

Het eerste blokje in figuur 1 geeft aan, dat er pas iets moet gebeuren als het groene vlaggetje ingedrukt wordt. Daar hebben we al een keer eerder kennis mee gemaakt, in een voorgaande aflevering. Weet je het nog?

Voor wie het niet meer weet: er is een leuk blokje binnen de categorie 'Gebeurtenissen' dat aan de bovenkant bolt, zodat er niets op geplaatst kan worden, maar dat aan de onderkant wel de bekende uitstulping heeft, zodat eronder wel blokken geplaatst kunnen worden. Binnen de categorie is het overigens ook het eerste blokje, dus gauw gevonden. Zet dit blokje bovenin het werkgebied.



Figuur 4 - Wacht op de vlag

Hieronder gaan we aan de slag om de rest van het programma uit te voeren. Dat is in dit geval 'maak scherm leeg'. We moeten nog uitmaken of we het pad van figuur 2 of van figuur 3 willen doorlopen. Het eenvoudigst lijkt het om te beginnen met de lineaire oplossing. Dit houdt in, dat we de drie cirkels na elkaar gaan tekenen. Dit houdt ook in, dat we met één sprite toe kunnen.

Het vakje 'Maak scherm leeg' uit figuur 1 betekent, dat we alles wissen en ook de sprite verbergen. Daarvoor hebben we dus twee opdrachten nodig. Alles wissen betekent, dat alles wat op het scherm geschreven of gestempeld is

wordt gewist, maar sprites en achtergronden blijven staan. Nu hebben we het nog niet over achtergronden gehad, maar wel over sprites. Om het scherm leeg te maken plakken we dus de volgende opdrachten onder de trigger van figuur 4.



Figuur 5 - Maak scherm leeg

De blokjes zijn al aan elkaar geplakt, en je kunt deze al activeren in je werkgebied. Je zult dan zien, dat het speelgebied leeg raakt.

Nu de activiteiten uit figuur 2. Het middelpunt is heel eenvoudig, want dat is $x=0$ en $y=0$. Daar kun je je sprite alvast heen sturen. Daarvoor hebben we een opdracht in de categorie 'beweging'.

Voor alle zekerheid zetten we eerst de pen uit, anders beginnen we strepen te trekken die we straks weer moeten zien kwijt te raken. Eigenlijk kun je die nog wel bij 'maak scherm leeg' plaatsen, maar je hoofd natuurlijk niet.

Cirkels teken je niet vanuit het middelpunt, maar we moeten wel cirkels rondom hetzelfde middelpunt tekenen.

Dat zal nog wel wat bijsturing behoeven; dat gaat misschien niet in één keer goed. Er is tenslotte niet een opdracht 'teken een cirkel'. Nog niet. Die kun je wel maken, maar daar zijn we nog niet aan toe.

Overigens, ik heb al eens eerder over de hulp bij Scratch geschreven, wat ik hier nog maar eens wil onderstrepen. Als je op de site van Scratch komt, zie je allerlei voorbeelden van programma's die met Scratch gemaakt zijn.



Figuur 6 - Ander project

Als je zo'n voorbeeld bekijkt en weten wilt hoe ze dat nu gedaan hebben, dan kun je eenvoudig op 'bekijk van binnen' klikken. Dat is die grote blauwe knop rechts bovenin.

Als je dat doet wordt het project in editor-mode geopend en kun je dus zien hoe het programma is opgebouwd. En dat is heel leerzaam. Op dezelfde wijze heb ik ooit HTML geleerd en het helpt me nu op mijn reis door Scratch. En bij Scratch mag het ook, het wordt gestimuleerd om van elkaar te leren, om je kennis te delen.

Het voorbeeld hierboven doet er verder niet zo heel veel toe, je ziet in dit project hoe een tekening gemaakt wordt door de computer, maar uiteraard is elke stap geprogrammeerd, met zo veel details is het dus een uitgebreid programma.

Nu terug naar de opgave, want daar zijn we nu mee bezig. Ik heb een vorige keer al laten zien hoe we een cirkel konden maken, maar mocht je dat vergeten zijn en de SoftwareBus niet meer kunnen vinden, dan kun je dus altijd een project opzoeken dat een cirkel tekent en dan 'van binnen' bekijken hoe dat gedaan wordt. Vandaar dit kleine uitstapje.

Nu gaan we dus eerst één cirkel maken en dan nog twee. Volgens het blok-schema beginnen we met de kleinste.

Hoe maak je een cirkel:

- Pen omhoog
- Je gaat naar de startpositie
- Pen omlaag
- Je zet een stapje, draait een beetje en dat herhaalt je totdat je op het beginpunt bent.

Dus zo doen we dat met deze drie blokken. Je kunt ze onder elkaar schuiven voor de seriële bewerking:



Figuur 7 - Drie concentrische cirkels

Hoe bepaal je nu het startpunt? Heel simpel, ik begon met startpunt (0, 0), ofwel $x=0$ en $y=0$. Weten we het nog, de coördinaten die we ooit met wiskunde leerden? Het middelpunt was dus (0, 0). In het blokje van de sprite, rechtsonder, staan de coördinaten van de sprite. Als je op 0 begint en je laat 'm tot halverwege lopen, dan lees je bij de y-waarde de diameter. Zoals je weet is de straal de helft daarvan.

Dus, als je de startwaarde van de y nu op de halve diameter zet, dan is het middelpunt nog steeds $y=0$.

Om puzzel op te lossen volgens de lineaire methode hoeft u alleen de blokken van figuur 4, 5 en 7 onder elkaar te plakken. Zodra je dan op het groene vlaggetje drukt krijg je dit als resultaat:



Figuur 8 - Concentrische cirkels - output

Wil je het parallel oplossen, dan heb je in principe dezelfde code nodig, alleen dan niet onder elkaar, maar als drie stromen, die met hetzelfde vlaggetje gestart worden.

In principe kun je de blokjes van figuur 4 en 5 achter elkaar zetten en deze bovenop elk blokje van figuur 7 plaatsen. Dit lijkt misschien wel dubbel werk, maar je weet niet van tevoren welke stroom als eerste uitgevoerd wordt.

Als je dus maar in één van de drie stromen de pen omhoog doet, heb je kans dat de andere stromen al aan het verplaatsen zijn en dus al een lijn tekenen. Of dat, als je maar in één van de stromen een 'alles wissen' doet, een andere stroom dan al een deel getekend had, dat dan ook gewist wordt en dan wordt de cirkel nooit gecompleteerd.

Dat kun je natuurlijk oplossen door de cirkel nog wat langer door te tekenen, over de bestaande cirkel heen. Je kunt ook na het beginstukje overal even wachten, zeg één seconde, dan weet je wel zeker dat alle initialisatie achter de rug is voor je begint te tekenen.

Ik probeer het eerst rechttoe rechtaan:



Figuur 9 - Parallele verwerking

Als je dit zo binnen één blokje probeert gaat dat niet goed. Probeer het maar eens uit. Ik krijg er iets uit als:



Figuur 10 - Wat gaat er mis?

gaan debuggen, oftewel onderzoeken wat er fout gaat en dan de fout herstellen.

Het oplossen van fouten, daar leer je meer van dan van menige les, is mijn ervaring.

Wat zie je al in het plaatje hierboven, wat valt jullie op?

Als je nog eens naar figuur 8 kijkt, zie je een resultaat dat we voor ogen hadden. Figuur 10 geeft een heel ander beeld. Er is niets concentrisch meer aan, en de derde cirkel wordt niet afgemaakt. Dat is natuurlijk raar, want we geven aan drie cirkels te willen tekenen en we geven ook nog aan waar we die willen hebben.

Iets wat ook opvalt, voor de oplettende lezer, is dat de kleur nu ineens heel anders is. Is dat spontaan? Dat lijkt me niet, in de code hebben we niets aan de kleur veranderd, tenminste niet in de code die ik in dit artikel heb gebruikt. En ook voor mij was de uitkomst enigszins verrassend, maar intussen wel helemaal verklaarbaar.

Wat heb ik gedaan? Ik heb in elk van de stromen de penkleur ingesteld. En natuurlijk alle drie verschillend, zodat ik kon zien wat er misging.

Nu, ik heb geen drie concentrische cirkels in drie kleuren gezien. We moeten ons onze sprite maar eens even indenken als een arme soldaat die drie commando's tegelijk moet uitvoeren.

De sergeant zegt links, de majoor zegt rechts, de kolonel zegt rechtdoor. Maar, hij kan dat niet allemaal tegelijk. Hij kan al helemaal niet op drie plaatsen tegelijk zijn. Ik zie dus dat die arme soldaat probeert iedereen van dienst te zijn, dus hij verdwijnt, hij zet de pen omhoog, gaat ergens heen, gaat ergens anders heen, gaat weer ergens anders heen en zet dan de pen neer.

Hij verandert de kleur van de pen drie keer en vervolgens gaat hij in een aantal herhalingen, per 'opdrachtgever' nog in een ander aantal herhalingen, zet een aantal stapjes, draait een beetje en blijft dat een aantal keren herhalen.

Aan het eind zijn de sergeant, de majoor en de kolonel boos, want de soldaat heeft niet gedaan wat hem opgedragen werd.

Het gaat dus mis, dit is niet wat we beoogd hebben. Is dat erg? Nee, dat is helemaal niet erg, het betekent dat we nog niet klaar zijn; we moeten

Wat is de oplossing?

Toch ten minste drie soldaten. En dat gaan we dus doen!

In het spriteblok, rechtsonderin, zorgen we dat er nu drie sprites komen te staan. Een extra sprite toevoegen doe je door op het katje met de plus te drukken.



Figuur 11 - Sprite blok

Het maakt niet uit welke figuur je kiest, want het eerste wat we doen is de figuur verbergen. Kies er dus maar een.

Vervolgens pakken we één 'kolom' uit de code van de eerste sprite en verplaatsen we die naar de tweede. Dan pakken we een andere kolom uit de eerste sprite en verplaatsen die naar de derde. We hebben dus voor elke sprite een eigen werkveld, maar nog steeds maar één speelveld.

Je kunt een kolom selecteren door op het blok te staan, met rechts te klikken en dan op kopiëren te klikken. Daarmee is hij nog niet weg bij de eerste sprite, maar dat komt nog wel. Op dezelfde wijze kun je 'm namelijk ook verwijderen. Je kunt ook gewoon een heel blok als geheel naar de 'legadoos' slepen. De blokjes sorteren zichzelf weer uit, heerlijk is dat.

Als we nu elk van de reeksen in een eigen werkveld hebben staan, dus één voor elke sprite, dan zul je zien dat het wel werkt.

Helaas, toch ging er nog wat mis en dat kan ik niet verklaren. Ik hoop dat er in ieder geval één lezer is die me dit uit kan leggen!

Als ik de code uitvoer zoals hierboven ingegeven (figuur 9, maar dan gesplitst per sprite) en ik voer ze stap voor stap uit, dus ik ga op een sprite staan en voer de code uit door op het bovenste blok te klikken, dan wordt er één cirkel getekend, gewoon, volledig, zoals de bedoeling was.

Als ik op de groene vlag klik lijkt dat ook te gebeuren, maar worden twee van de drie cirkels niet afgemaakt, het laatste stapje ontbreekt. Geen idee hoe dat komt, maar wel wat ik er aan kan doen, namelijk de herhalingen van de betreffende cirkels met één verhogen, dan gaat het allemaal goed.

Dus ook parallel krijgen we dan een goed resultaat. Door het foutzoeken heb ik nog steeds drie kleuren, daarover zegt de opgave niets, dus is het ook goed:



Figuur 12 - Resultaat

Samenvattend: We hebben dus twee programma's gemaakt die doen wat er gevraagd wordt, het tekenen van drie concentrische cirkels, ergens in het midden.

We hebben hiervoor precies het midden genomen, dat is wel zo makkelijk, vanwege de (0, 0) coördinaten. We hebben dit eerst op de lineaire wijze gedaan, zeg maar 'klassiek', en daarna hebben we dit op de parallelle wijze gedaan, zeg maar 'object georiënteerd'.

We hebben het programma laten starten met de groene vlag. We hebben naar de startpositie gekeken, we hebben het scherm leeggemaakt voor we begonnen, zowel door alle lijnen weg te halen als door de sprites te verbergen.

We hebben eenvoudige cirkels getekend, waar misschien wel wat op af te dingen valt, maar het is duidelijk wat de bedoeling is.

Kijken we nog naar de resterende vragen, dan is duidelijk dat we klaar zijn als de derde cirkel getekend is. Dat ging in de parallelle verwerking om een of andere reden mis, hopelijk weet iemand waarom. Ik ben benieuwd.

Verder is de uitvoering vlot, maar dat komt door de aanpak. Vorige keer deed ik in heel kleine stapjes met 360 herhalingen. Dan duurt het langer.

De kleine cirkel is nu wat grof, met stapjes van 36 graden, 10 stappen, dus snel klaar. Als je goed kijkt, zie je bij de uitvoering dan ook dat de kleinste cirkel als eerste klaar is, de middelste daarna en de grote cirkel, met 30 herhalingen, doet er nog iets langer over.

Overigens, niet zo relevant als je aan het stoeien bent, maar als je eenmaal serieus met Scratch aan de slag bent, zit er ook nog een optie 'Zet turbo-modus aan' in het Scratch-menu, onder 'Bewerk'. Dan gaat alles echt heel snel en kun je niets waarnemen over de opbouw, maar zo kun je natuurlijk wel een spelletje spelen dat een beetje vlot reageert.

CompUfair op 20 april 2019

Elders in dit nummer vindt u een uitgebreide voorbeschuiving op onze CompU-

fair van 20 april aanstaande. Daar kunt u ook een lezing bezoeken over Scratch. Daarvoor of daarna kunt u natuurlijk altijd even langskomen bij de spreker, maar ook aan de tafel van ons Platform WebOntwerp bent u welkom om over Scratch te praten. En als ik toevallig bij de redactietafel sta, dan mag dat daar uiteraard ook.

Scratch is intussen een van de speerpunten van CompUsers aan het worden. Bij gebleken belangstelling kan hiervoor ook een eigen Platform komen, naast of in plaats van het Platform WebOntwerp. Zover zijn we nog niet, maar wat niet is kan nog komen. Vooralsnog zitten we bij WebOntwerp goed.

Wat gaan we doen dit najaar?

Tijdens de komende MegaCompUfair in september gaan we een heuse workshop opzetten rondom Scratch. Ik houd daar een korte inleiding en dan kunnen de deelnemers zelf aan de slag.

We gaan dan een account aanmaken en onze eerste stappen zetten. Dat zal voor de een sneller gaan dan voor de ander, maar ik zal zorgen dat er voor iedereen een uitdaging bij zit.

Als je dus al bekend bent met het huiswerk hier, het zelf al een keer hebt gedaan, dan begin je met een voorsprong.

Het zou trouwens HEEEEEEEL leuk zijn, als de workshop gevuld werd met teams van (groot)ouder met (klein)kind. Zoals jullie weten organiseren wij onze workshops altijd met één laptop per twee deelnemers, en op deze manier een team vormen is helemaal de bedoeling van ons en Scratch.

Het hele 'project'

<https://scratch.mit.edu/projects/296645638>

Wat kunnen we nog meer doen?

We kunnen verder gaan met programmeren, maar de basis is nu toch wel gelegd. Verder komt het er nu op aan om leuke uitdagingen te verzinnen.

Ik zou wat dieper kunnen ingaan op Scratch en de Arduino-projecten, maar daar moet ik me zelf eerst nog in verdiepen. Dat kan in de toekomst wel, maar misschien dat iemand anders dit al kan of wil oppakken?

Intussen heb ik tijd nodig om me verder te verdiepen in Scratch, want er zitten nog wat leuke dingen in, die ik nu nog niet onder de knie heb. Zo weet ik al - maar nog niet precies hoe - dat we ons probleem van hierboven nog slimmer zouden kunnen aanpakken, door toch

maar één keer de initialisatie te doen en dan een bericht uit te sturen. In plaats van het groene vlaggetje om de drie stromen te activeren, zou ik dat bericht kunnen gebruiken. Maar ook daar moet ik me eerst in verdiepen.

Wat óók kan, heb ik ook nog niet gedaan, is eigen codeblokken maken. En eigen variabelen definiëren. Die zou je misschien ook weer in je eigen codeblokken kunnen gebruiken, daar ga ik eigenlijk wel van uit. Oftewel, nog genoeg te ontdekken!

En ten slotte: interactie. Ik heb wel wat ideeën gezien door de diverse projecten te bekijken. En het moet gezegd, de werkwijze met de verschillende stromen om zaken te activeren, dat is iets wat programmeren in deze tijd toch makkelijker maakt dan toen ik nog leerde programmeren.

Ik weet niet of ik voor volgend nummer al weer voldoende bijgeleerd heb. Misschien dat ik een nummer moet overslaan, of alleen maar met wat leuke opdrachten kom. En ja, dat is alleen zinvol als er ook iets mee gebeurt.

Ik zal bij de lezing over Scratch in elk geval goed opletten wie er meedoet en wat er zoal leeft. En tijdens de workshop in september kunnen we zien of er nog belangstelling is en wat we daarmee kunnen doen.

Om de mensen tegemoet te komen die wel belangstelling hebben, maar dit nog niet geuit hebben, volgen hierbij nog wat opgaven.

Opgaven

1. Teken twee spiralen, in elkaar, die elkaar niet mogen raken.
2. Teken drie van linksboven naar rechtsonder lopende diagonalen, beginnend in de linkerbenedenhoek en doorlopend naar de rechterbovenhoek. Er moeten dus telkens drie lijnen zichtbaar zijn, en als de vierde getekend wordt, moet de eerste weer weggehaald worden. Het lijkt dus, alsof de lijnen lopen.
3. Probeer de opgave van dit artikel uit te voeren met behulp van berichten, dus in de gehele code mag maar één keer op het drukken van de groene vlag gecontroleerd worden.

Samenwerking jeugd en ouderen

Ik gaf het al aan, tijdens de workshop zou dit heel goed kunnen worden vormgegeven.

En als mensen samenkomen bij de WebOntwerp-tafel, kunnen we misschien wel even een laptop vrijmaken om hier samen mee aan de slag te gaan.