

Scratch – programmeren (5)

René Suiker

Aflevering vijf over het programmeren in Scratch.
Doe wel mee, want op 28 september kunt u uw kunsten laten zien tijdens de MegaCompUfair!

Inleiding

Vorige keer zei ik dat het misschien wel even stil zou zijn vanuit het Scratch-gebeuren, maar intussen hebben we natuurlijk onze ALV en CompUfair gehad en is er (door Gerard Vriens van HCC!AI) een presentatie over Scratch gehouden. En dat geeft altijd aanleiding voor een vervolg.

Naast het feit dat het voor onze lezers nuttig was, was het dat ook voor mij, want in mijn vorige nummer liepen de nummeringen van de plaatjes niet helemaal synchroon met de tekst. Daar lag een technisch probleem aan ten grondslag, deels al in Word, dat weigerde mijn afbeeldingen netjes te nummeren. Onze eindredactie heeft het liefst tekst in Word of Writer, eventueel met kopjes, maar in een eenvoudige opmaak. De eindopmaak wordt verzorgd in Scribus, het freeware DTP-programma. Verder waren mijn plaatjes blijkbaar nogal laag van resolutie. Maar dat is met wat kunstgrepen opgelost.

We gaan naar <https://scratch.mit.edu> en inloggen maar. Weten we het nog? De opgaven van de vorige keer:

1. Teken twee spiralen, in elkaar, die elkaar niet mogen raken.
2. Teken drie van links boven naar rechtsonder lopende diagonalen, beginnend in de linkerbenedenhoek en doorlopend naar de rechterbovenhoek. Er moeten dus telkens drie lijnen zichtbaar zijn; 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.

Nog even

Vorige keer had ik er, meen ik, al even bij stilgestaan: de nieuwe versie van Scratch ziet er iets anders uit. Ik toon even het nieuwe scherm en beschrijf weer de hoofdpunten; dit om het iets eenvoudiger te maken de zaken terug te kunnen vinden.

De indeling is dus iets anders geworden, maar geen zorgen, in grote lijnen is het idee nog steeds hetzelfde. Onder '1' zie je de gereedschapskist, waarbij je drie tabbladen ziet. Deze keer houden we het nog even bij het 'code'-tabblad, maar een volgende keer gaan we in op de 'Costumes' (ik had 'm' nog niet op Nederlands gezet, na de komst van de nieuwe computer: uiterlijkheden) en 'Sounds' (geluiden dus). Bij '2' zie je het wereldbolletje, net rechts van het Scratch-logo; daarmee stel je de gewenste taal in.

Vak '3' is het werkgebied waar je je scripts maakt door blokken uit de gereedschapskist erheen te slepen en aan elkaar te plakken. Vak '4' is dan het speelveld, waarin e.e.a. zich afspeelt. Je ziet hier nu een kat staan, dat is de standaard (Engels: default) 'Sprites'. Sprites zijn figuren die over het scherm bewegen.

Zijstapje

Voor de oudjes onder ons, die rond de begintijd van de HCC zelf aan het programmeren waren - toen had je al systemen met Sprites, maar de meesten hadden dat niet. Als je dan een figuurtje wilde laten bewegen over het scherm, dan moest je dat over de achtergrond heen tekenen, maar als het figuurtje dan bewoog, dan moest je de achtergrond weer terugtekenen. Met Sprites gaat dat automatisch, die bewegen over de achtergrond heen.

Over Sprites gesproken, bij vakje '5' zie je dus de Sprites die je in dit project gebruikt. In dit geval hebben we net een nieuw project geopend en starten we met de standaard waarden en dat is met één Sprites.

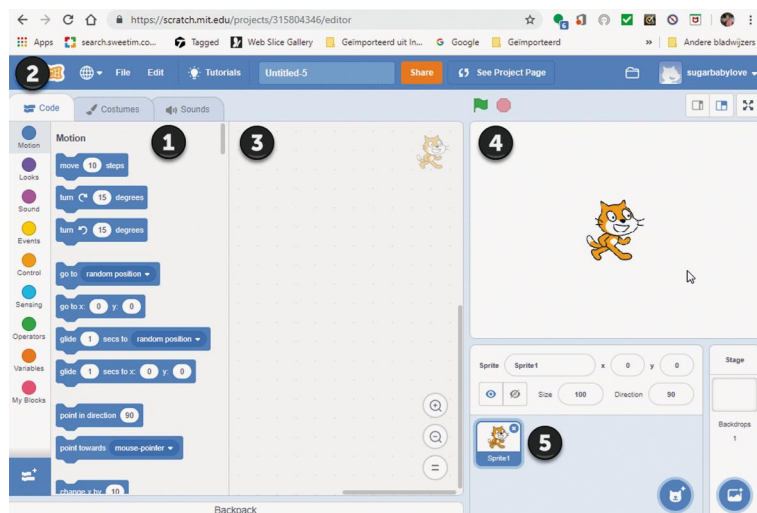
Let even op, het codeblok (blok '3') dus, is gekoppeld aan een Sprites. Als je een nieuwe Sprites introduceert, dan komt er ook een nieuw codeblok. Zodoende kunnen de Sprites onafhankelijk van elkaar bewegen. Je kunt ze ook afhankelijk van elkaar laten bewegen, maar dat is voor later. Dan is het nu tijd voor het echte werk en gaan we over naar Opgave 1.

Opgave 1

Er moest natuurlijk een beetje uitdaging inzitten, dus daar heb ik voor gezorgd. Met wat we tot dusver behandeld hebben was deze opgave in principe nog wel te maken, maar erg omslachtig. Je moest dus een beetje verder gaan kijken en ik hoop dat dat dan ook gelukt is.

Hoe teken je een spiraal? Laten we even aannemen dat je van binnen naar buiten tekent, dan begin je een cirkel te tekenen, alleen met elke stap maak je de straal net iets groter. En als je van buiten begint, dan maak je uiteraard de straal net iets kleiner.

Tot nu toe hebben we steeds met vaste waarden in onze instructies gewerkt. Daarmee kun je een spiraal tekenen, door elke beweging als instructie op te geven. Maar dat is best een hele opgave. Eenvoudiger is het om een 'variabele' te gebruiken.



Een variabele is een van de essentiële elementen van het programmeren. In een variabele kun je een waarde opslaan, die je later in je programma kunt gebruiken. Maar, de naam zegt het al, de waarde kan variëren in de loop van het programma. Dit in tegenstelling tot de constanten die we tot dusver gebruikt hebben. En hoewel Murphy beweert ‘constants aren’t, variables won’t’, nemen we dit toch maar even als waarheid aan.

Dus, als je een spiraal wilt maken, dan zet je de pen ergens neer, je zet een stapje, je draait een beetje, je zet een iets groter stapje, je draait weer een beetje, je zet weer een iets groter stapje, enzovoort.

En dat draaitje, dat kan steeds hetzelfde blijven, maar je ziet al in de instructie, steeds iets groter, dat smeekt dan om een variabele. En gelukkig kan Scratch daar ook goed mee omgaan. Je kunt je eigen variabele definiëren en deze door het programma heen gebruiken. Je kunt variabelen definiëren binnen de context van één Sprite, of binnen je hele project. En misschien zijn er nog wel meer varianten, want ik loop niet zo heel ver voor op wat ik beschrijf.

Eén spiraal

Om eens te kijken wat dit nu betekent, gaan we eerst eens één spiraal uitproberen. Als dit niet goed gaat, dan moeten we de aanpak al aanpassen, maar met één spiraal introduceren we al de variabele, dus laat eens kijken wat er gebeurt. Kijk nog eens in de gereedschapskist, daar zie je de te gebruiken elementen, gegroepeerd per categorie.

In een van de eerste afleveringen heb ik dit concept al uitgelegd, al stond dit toen dus in het midden. Intussen is er ook wel iets veranderd aan de indeling: zo zie je hier ineens de ‘pen’ niet meer staan. Maar dat heb ik de vorige keer al uitgelegd, die voegen we gewoon toe via ‘uitbreidingen’ en dan is die weer beschikbaar. Maar nu dus even goed kijken naar de ‘variabelen’. Als je daar op klikt, dan zie je dat de gereedschapskist een ander vakje toont, namelijk dat van de variabelen:



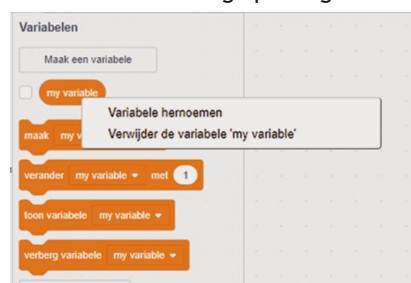
Je ziet hier een heel beperkt aantal blokken, maar dit blok kan in principe oneindig groot worden naarmate je meer variabelen gaat introduceren. Elke variabele wordt apart getoond. Standaard zit er één variabele in, en wel ‘my variable’, zoals je ziet bij ‘1’.

Dan zie je twee blokken onder ‘2’ waarmee we de waarde van de variabele kunnen bepalen. Met het eerste blok zet je een absolute waarde neer. Met het tweede blok modificeer je de bestaande waarde van een variabele. Je kunt er iets bij optellen of iets van aftrekken. Maar er is veel meer mogelijk, want je kunt weer functies gebruiken in het blokje waarmee je de variabele instelt. En die functies kunnen weer gebruik maken van deze en andere variabelen. Scratch is op zich een volwaardige programmeertaal, die door de vormgeving aantrekkelijk is, maar daarmee niet noodzakelijkerwijs altijd maar makkelijk.

Onder ‘3’ zie je de mogelijkheid om variabelen te tonen in het speelveld. Dat doe je normaal gesproken vooral als je nog aan het foutzoeken bent (‘debuggen’). Maar je kunt ook andere redenen hebben om de waarde van variabelen te tonen.

Ten slotte zie je een aantal blokjes in het wit, zoals bij ‘4’: ‘Maak een variabele’. Zolang we één spiraal maken, hebben we ook maar één variabele nodig, maar dan zou je die wel een andere naam willen geven.

Op ‘lijst’ gaan we een volgende keer nader in. Voorlopig houden we het nog even eenvoudig. Het stukje ‘mijn blokken’ hoort niet meer bij variabelen, de gereedschapskist is één geheel, je kunt er ook doorheen scrollen en dan ga je gewoon van ‘beweging’ bovenin helemaal door naar beneden tot en met ‘mijn blokken’. En de blokken, daar komen we in de toekomst ook nog op terug.



Als je met de rechtermuisklik op ‘my variable’ klikt, dan kun je deze een andere naam geven:

Als je op ‘variabele herbenoemen’ klikt, dan krijg je nevenstaand scherm:



Er staat ‘hernoem ALLE...’. Dit betekent, dat de naamwijziging door het hele project heen doorgevoerd wordt. Want deze variabele wordt mogelijk al op veel plaatsen gebruikt en je wilt niet dat het programma niet meer werkt omdat je een variabele een andere naam hebt gegeven. We noemen de variabele maar even voor het gemak ‘straal_2’. Hoezo is dat gemakkelijk? Nou, ik heb de opgave eerst uitgewerkt voor ik het artikel schreef en daar heb ik wel twee stralen nodig gehad. En ik gebruik de code van Sprites 2 ter illustratie. Het ‘voor het gemak’ slaat dus op mijn gemak.

Links onderin staat een blauw blokje, met twee witte lijntjes en een plusje. Hiermee kun je nieuwe codeblokken toevoegen. Als je de pen bijvoorbeeld nog niet hebt toegevoegd, dan kun je dat daar doen. Maar er zijn meerdere blokjes die niet standaard al zijn ingeschakeld. Als je op het blokje klikt kom je op een nieuwe pagina, waar je verschillende blokken tegenkomt:

- Muziek
- Pen
- Video
- Tekst naar spraak
- Vertaal
- Makey makey
- Micro:bit
- Lego Mindstorms EV3
- Lego Education WeDo 2.0
- Lego Boost
- Go Direct Force & Acceleration

Ik denk dat deze lijst in de toekomst nog wel uitgebreid zal worden, want de ontwikkelingen rondom Scratch staan niet stil.

Voorlopig laten we die blokken nog even voor wat ze zijn, maar: zet wel even het pen-blok aan als je de opdrachten wilt uitvoeren.

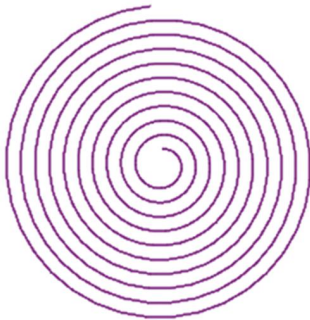
Om een spiraal te maken zoals zojuist beschreven, heb je dus de volgende code nodig:



Wat je hier dus ziet, we gaan ongeveer naar het midden van het scherm. Je kunt ook exact het midden nemen, maar, zoals gezegd, ik begon de uitleg op basis van de tweede spiraal, misschien niet handig, maar ik ben in elk geval consequent. Dan richt ik mijzelf. En dan stel ik de variabele in. Dan zet ik mijn pen neer en ik stel een kleur in. Het was geen deel van de opgave dat de twee spiraal

ralen een andere kleur moesten hebben, maar je kunt dit zeker in het begin beter wel doen. Als je dan gekke dingen ziet gebeuren, kun je even kijken of dat voor beide spiralen geldt of voor maar één van de spiralen en dan weet je ook gelijk welke.

De vorige keer ben ik heel diep ingegaan op het foutzoeken, dus nu wil ik het even hierbij laten. Vervolgens kom ik in een herhaal-lus, in dit geval ga ik de lus 360 keer uitvoeren. Omdat je daaronder ziet dat ik telkens tien graden draai, weet

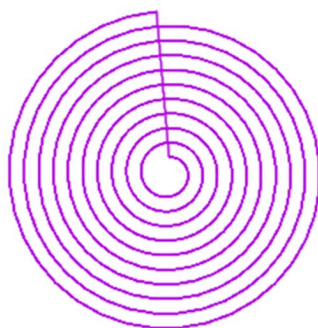


je dus dat er in totaal 3600 graden gedraaid wordt, ofwel we zullen straks een spiraal zien van tien 'cirkels'.

Als ik nu op de groene vlag klik gebeurt er nog niets, want we hebben nog geen activering ingebouwd. Maar als ik op het bovenste blauwe blokje klik, dan krijg ik een beeld zoals hier weergegeven.

Druk je nog een keer op het blauwe blokje bovenin, dan is het resultaat iets anders:

Vergeet even de omvang van de afbeelding, dat is een kwestie van het uitsnijden van het scherm, daar ben ik niet zo'n ster in. Maar het grote verschil is dat er een streep staat tussen begin en eindpunt. Probeer eens te verklaren waar dat nu door komt. Je hoeft het me niet te mailen, ik kom er de volgende keer wel op terug, maar je zou het zelf moeten kunnen beredeneren. Intussen hebben we gezien dat het blokje code in staat is om een spiraal te produceren. Dus dat hebben we bereikt.



Terug naar de opgave

Je kunt ook hier weer het probleem oplossen op verschillende manieren. Je kunt een spiraal tekenen en daarna de tweede spiraal tekenen, sequentieel dus. Je kunt ook twee Sprites ieder een spiraal laten tekenen, zodat je de spiralen gelijktijdig opgebouwd ziet worden. Er zijn natuurlijk ook nog voldoende andere mogelijkheden.

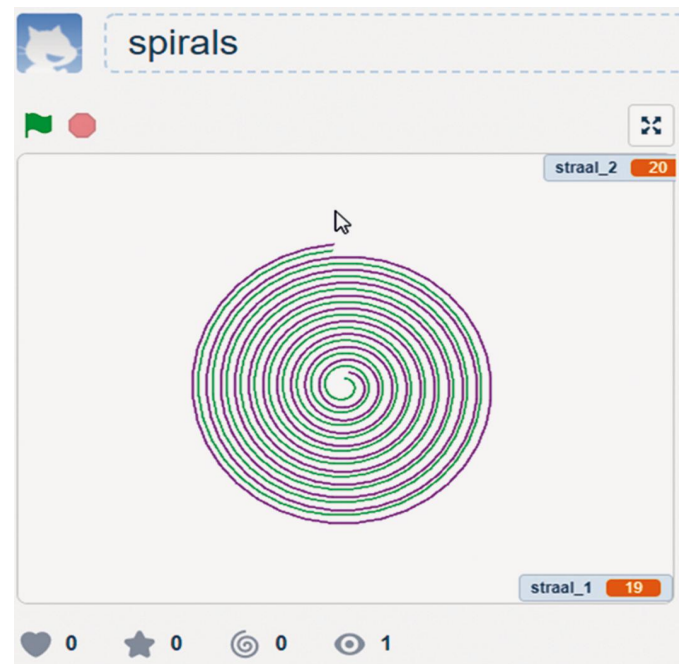
Zo kun je, voor een leuk grafisch effect, natuurlijk één spiraal van binnenuit opbouwen en je tweede spiraal van buiten naar binnen opbouwen. Om dit goed te doen, is het zaak te kijken wat je startwaarden moeten zijn, want vanaf één beginnen met tellen is makkelijk, maar om te weten waar je moet beginnen als je terug gaat bewegen is wat complexer. Ik zou beginnen met van binnenuit, kijken waar je uitkomt (weet je nog, je kunt dat in het Sprites-blok aflezen) en daar je beginsituatie maken.

In het variabelenblok, links bij de gereedschappen dus, kun je bij elke variabele met een vinkje aangeven of je ook de waarde op je scherm wilt zien. Dat heeft voor onze opdracht weinig zin, maar als je een probleem hebt met de opdracht kan het wel heel nuttig zijn om te kijken wat de variabele doet. En als het allemaal te snel gaat, kun je natuurlijk elke keer een korte wachttijd inbouwen. Dat moet je natuurlijk wel binnen de lus doen, anders merk je er nog niet veel van. We hebben gezien hoe we een spiraal kunnen tekenen, nu willen we natuurlijk even kijken hoe we er twee tekenen. De makkelijkste manier is om te beginnen met wat grotere afstanden tussen de lijnen, maar het is natuurlijk wel mooier om het niet te grof te maken. Daarom moet je ofwel een beetje rekenen, ofwel gewoon langzaam naar een werkbare waarde toewerken. Dat klinkt misschien wel erg flauw, maar zo wordt vaak wel gewerkt en op zich is er niets mis mee: het resultaat telt.

Wat we wel hadden afgesproken is, dat beide spiralen getekend werden door de druk op de groene vlag. Weten we het nog? Het blokje onder de categorie 'Gebeurtenissen', met de bolle bovenkant: 'wanneer op de groene vlag wordt geklikt'. Alles daaronder wordt dan uitgevoerd. Daaronder in die zin dat het wel eraan vastgeplakt moet zijn, simpelweg eronder positioneren in het scriptgebied haalt niets uit. Twee spiralen dus, dus twee Sprites. Je kunt het ook door één Sprite laten doen, maar het is gemakkelijker met twee. We hebben namelijk de code al om één spiraal te tekenen. Deze code kun je gewoon kopiëren naar een andere Sprite. Daarna pas je de beginpositie en misschien de richting en de begin-'straal' aan en dan kunnen ze tegelijk gaan 'lopen'. Wel nog even zorgen dat je een heldere uitgangspunt inneemt voordat je de herhaling ingaat.

Een oplossing voor het spiraalprobleem kunt u terugvinden op: <https://scratch.mit.edu/projects/301217909/>

Weet je het nog? Je opent het project en dan klik je op de blauwe knop rechts boven 'bekijk van binnen'. Het resultaat:



Je ziet dat de variabelen aan de rechterkant worden weergegeven. Dat kun je dus uitzetten in het blok 'variabelen' van de gereedschapskist.

Als je na het afronden in het Spritesgebied naar de waarden kijkt, dan zou je dus genoeg informatie moeten hebben, om de tweede spiraal van buiten naar binnen te laten draaien. Ik denk dat we deze opdracht meenemen als 'de makkelijke opdracht' voor de workshop tijdens de MegaCompUfair van 28 september aanstaande. In het volgende nummer van de SoftwareBus gaan we alvast uitgebreid in op wat je daar te wachten staat, maar we hebben al beloofd dat we dan een workshop Scratch organiseren. En ik hoop dat ik dan een zaal vol enthousiaste Scratchers aantref, met wie we samen leuke dingen kunnen doen.

Even voor alle duidelijkheid, de beschikbare ruimte deze keer is niet toereikend om opgave 2 helemaal uit te voeren, laat staan opgave 3. Maar ik wil wel wat hints geven, want opgave 2 lijkt me ook een leuke opgave voor de workshop. En ten slotte wil ik ook nog iets doen met de mieren uit de presentatie van Gerard Vriens.

De hints

Je hebt een lijst nodig, vergelijkbaar met een array in andere programmeertalen. Je tekent namelijk drie lijnen, waarna je de laatste weer wist. Je moet dus wel onthouden waar die stond.

Je zou kunnen beginnen met alleen lijnen te tekenen op de manier zoals in de opgave beschreven, en niets weghalen, maar het scherm wissen en dan de laatste lijnen weer tekenen. Dat kan uiteindelijk eenvoudiger uitpakken. Misschien is dit zichtbaar, maar uiteindelijk voldoet het resultaat toch aan de eisen. 'Perfect is de vijand van goed', zeggen we wel eens.

Voor wie het even vergeten is, we onderscheiden tijdens onze (Mega)CompUfairs workshops en lezingen. Lezingen worden gegeven door de inleider, waarbij er soms (nagenoeg altijd) gedemonstreerd wordt. De zaal luistert en mag ook vragen stellen. Bij een workshop zijn er ook pc's beschikbaar voor de bezoekers. In principe ga je, onder begeleiding, in groepjes van twee zelf aan de slag achter een pc. De begeleider houdt meestal wel een soort inleiding, doet eventueel iets voor, maar daarna wordt er verwacht dat de bezoeker aan het werk gaat. De ervaring leert dat je daar veel meer van opsteekt, want door vallen en opstaan leer je uiteindelijk het meest.



Ik denk dat Scratch zich heel goed leent voor een workshop en verwacht eerlijk gezegd best een grote opkomst. Ik sta open voor suggesties met betrekking tot de opdrachten, maar raad iedereen alvast aan een scratch account te openen, zodat je daarmee in kunt loggen en hoeven we daar geen tijd aan te verspillen. En de grote aanbeveling, neem je (klein)kind mee en kom als familieteam aan de pc, dat zou ik echt heel leuk vinden. En, even voor alle duidelijkheid, als uw (klein)kind dit allemaal veel beter kan dan ik, dan voel ik mij allermindst bezwaard.