

Scratch (7)

René Suiker

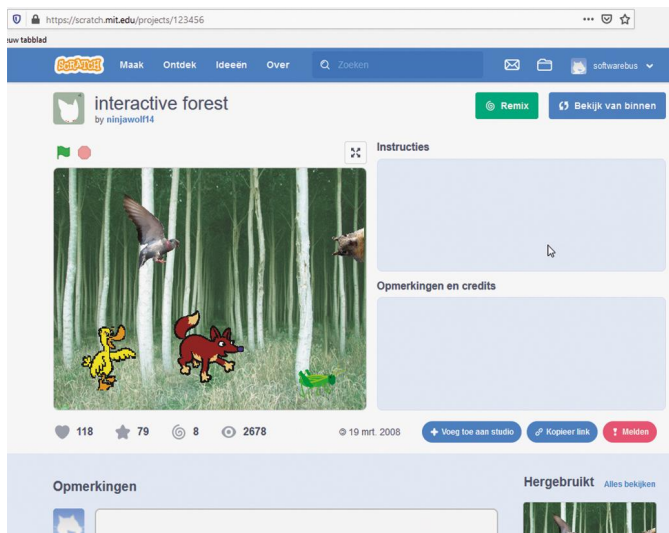
Inleiding

Beste Wim Boon, ik ben je nog niet vergeten! Door praktische omstandigheden heb ik je het boek nog niet kunnen toesturen, maar dat gaat dit jaar nog plaatsvinden. Waarschijnlijk heb je het intussen ook gekregen. Zo, dit moest er even uit. In het vorige nummer haalde ik de oplossingen van Wim Boon aan bij het beschrijven van het huiswerk. Omdat hij alle opgaven goed had beantwoord, hebben we in het bestuur besloten hem een beloning toe te kennen.

Nu even een heel korte terugblik. Vorige keer vertelde ik over de workshop tijdens de CompUfair en dat we die nog een keer gaan doen, waarschijnlijk al in februari a.s. (2020 dus). Ik ga proberen of ik dan ook iets meer tijd kan vrijmaken, dan kan ik net iets meer doen. Vervolgens ging ik uitgebreid in op het huiswerk van de keer ervoor en nam ik jullie mee langs de oplossingen van Wim Boon.

Daarbij stelde ik dan ook nog wat vragen, waarop ik helaas geen enkele inzending binnen heb gekregen. Misschien, omdat de nummers 5 en 6 elkaar wel erg snel opvolgen, maar misschien omdat het zachtjes aan tijd wordt om de reeks wat af te bouwen. Voor volgend jaar ga ik dan ook wat minderen met Scratch, de bedoeling is dan dat ik nog drie nummers over Scratch ga schrijven. De bedoeling is ook dat die dan wel wat meer voor gevorderden zijn, want de reeks is natuurlijk via de website www.compusers.nl te volgen, en na 7 artikelen ben je intussen geen echte beginner meer.

Goed, we gaan weer Scratchen vandaag. Ik doe eens gek en tik wat willekeurigs in: <https://scratch.mit.edu/projects/123456>.



Figuur 1 - Project 123456

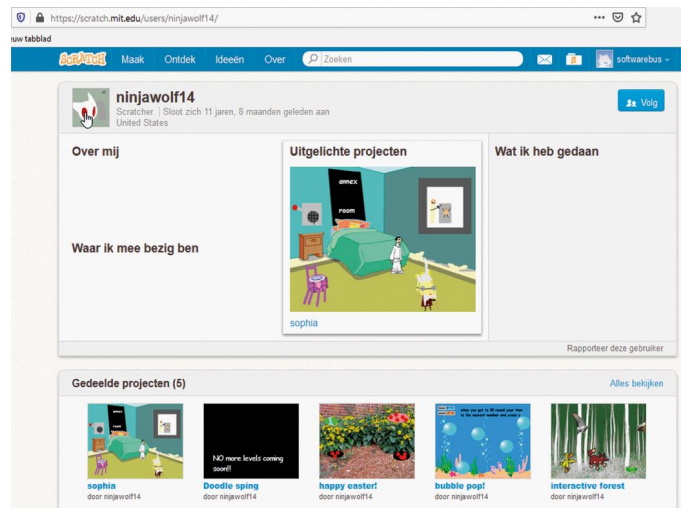
Waarom dit project? Geen bijzondere reden, maar het geeft wel aan dat er op de site van Scratch al de nodige projecten te vinden zijn. Ik heb ook nog 1234567 geprobeerd, maar die kon hij niet vinden. Intussen zijn we al veel verder, want mijn nieuwste project kreeg een nummer met 8 cijfers. Het gaat dus hard.

Hieruit kunnen we afleiden, dat project 1234567 blijkbaar nog niet is 'gedeeld'. Je kunt op elk moment besluiten je project te delen, en zodra je het deelt kunnen anderen het zien. En waarom zou je dat willen? Daar kunnen een aantal redenen voor zijn:

1. Het 'CompUsers' idee: je deelt je kennis met gelijkgezinden
2. Het 'narcisme'-idee: kijk eens hoe goed ik ben
3. Het 'samen leren'-idee: ik loop hier tegenaan, wie helpt mij?

Zelf vind ik delen interessant, maar nog niet als ik net met een project begin, het moet wel al enige substantie hebben. Als ik nog aan het rommelen ben, dan vul ik ook de 'instructies' en 'opmerkingen en credits' nog niet in. Als je gaat publiceren, 'delen' dus, is het aan te raden dat wel te doen. Het helpt je bezoekers en is misschien bij later terugkijken ook voor jezelf interessant. Even voor alle zekerheid, die 'instructies' en zo zie je rechts in beeld als je op de projectpagina staat (zie figuur 1).

Is dit project 123456 bijster interessant? Niet echt, maar ik heb wel een paar punten gezien waar ik het in dit artikel over kan hebben. Je ziet een paar dieren in een bos, je ziet nog een keer een aantal aspecten die we al besproken hebben, maar ik wil nog een keer voor alle zekerheid door het scherm. Onder de titel van het project zien we de maker. In dit geval 'ninjawolf14'. Ik ken hem of haar niet, maar je kunt op het linkje klikken en dan kom je op de pagina van deze Scratcher. Het is geen Facebook, er worden niet heel veel details verstrekt over de gebruiker, maar wel waar hij of zij vandaan komt en hoe lang hij of zij al Scratcher is. En wat ruimte om wel iets over zichzelf te vertellen, maar het hoeft dus niet. Verder zie je daar ook de projecten die door deze Scratcher zijn gedeeld:



Figuur 2 - Scratcher Ninjawolf14

Zo te zien is deze Ninja dus al ruim elf jaar Scratcher en heeft hij vijf projecten gepubliceerd; het lijkt erop alsof deze Scratcher net als ik alleen maar een beetje probeert wat te experimenteren. Misschien zijn er al duizenden projecten gemaakt, maar er zijn er in al die jaren slechts vijf gedeeld.

Verder is wel zichtbaar wie hij of zij volgt en door wie hij of zij gevolgd wordt. En naar die Scratchers kun je ook weer doorklikken. Welkom in de wereld van Scratch, waar we allemaal met elkaar verbonden zijn, maar privacy toch wel behoorlijk gewaarborgd is. Overigens is Scratch dus verder ook een 'veilige' omgeving. Als je iets ziet wat echt niet door de beugel kan, dan kun je dit rapporteren en wordt het verwijderd. Denk hierbij aan geweld of porno.

Je ziet boven de projecten aan de rechterkant een linkje 'alles bekijken', maar in dit geval haalt dat niet veel uit, omdat alle vijf de projecten al in beeld zijn. Maar je kunt je voorstellen dat er actieve Scratchers zijn die pagina's kunnen vullen met hun projecten. Met behulp van de link kom je op de projectenpagina van deze Scratcher, maar daar zie je dus niets anders dan deze vijf projecten.

Nog even terug naar figuur 1. Je ziet daar naast de knop 'bekijk van binnen' nog een knop, in het groen, met 'Remix' daarop. Als je op die knop 'Remix' klikt, dan wordt er een kopie van dit project gemaakt onder jouw account en kom je gelijk in de editor terecht. Hier mag je dan gaan Scratchen, het programma aanpassen dus, maar je kunt natuurlijk niet het origineel van iemand anders aanpassen. Je kunt wel iets aanpassen in het origineel, maar dan moet je het onder je eigen account opslaan. Met Remix wordt dus de basis gelegd om alvast een vliegende start voor je eigen project te hebben. Dat lijkt een leuke optie om zelf een soort basis aan te leggen, waarmee je dan telkens kunt beginnen, maar ik heb nog geen optie gezien om je eigen projecten te remixen. Als je zoiets wilt, dan moet je gewoon een basisproject maken en openen en dan als kopie opslaan, waarna je met die basis je project verder ontwikkelt.

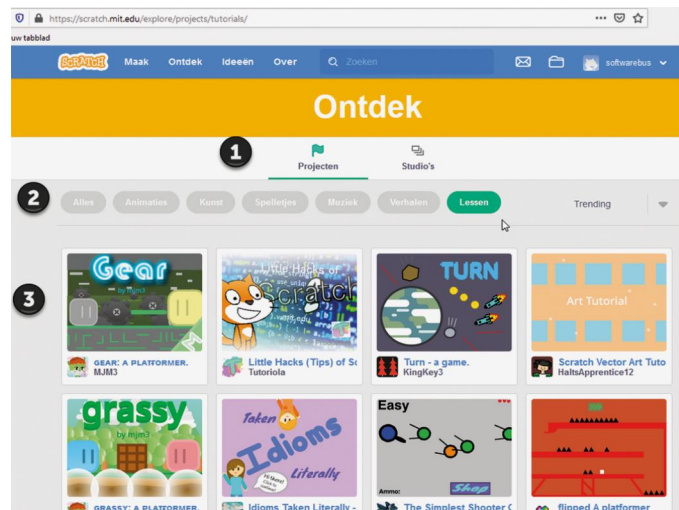
Het project 'interactive forest' waar we mee begonnen, was ik nog verder aan het beschrijven. Je ziet een plaatje van een bos, met daarop een aantal dieren geprojecteerd. Zonder nu al in de code te kijken, zie je dus dat de dieren deels tekeningen zijn en deels foto's, bijzonder 'hard' gesneden. Als je op de groene vlag klikt gebeurt er niets. Maar als je op één van de dieren klikt, dan wordt een geluid afgespeeld, dat het geluid van het dier moet uitbeelden. Door van binnen te kijken zien we dan ook niet heel veel bijzonders, het lijkt een project om iets met geluiden te proberen. Daar is op zich natuurlijk helemaal niets mis mee. Het is mij nu met name te doen om het feit dat je natuurlijk complete spellen online kunt zetten, maar dat er ook niets mis mee is om een experiment waar je mee klaar bent ook te delen, zodat anderen daar ook hun leermoment mee kunnen hebben.

Tot zover de Ninja. We gaan nu eens kijken wat er allemaal voorhanden is om ons verder op weg te helpen. Dat is namelijk zo leuk aan Scratch, het is allemaal beschikbaar, het is in feite een heel goedkope hobby. Je hebt een pc nodig, die hebben we allemaal. Een internetverbinding, die hebben we ook allemaal. En verder kun je aan de slag, want al het 'speelgoed' is voorhanden en je hebt een oneindige doos met handleidingen.

In het verleden, toen ik mijn eerste schreden zette op het gebied van HTML, kon je ook veel leren door naar mooie sites te kijken en dan de code te bekijken om te zien hoe ze dat hadden gedaan. En soms moest je dan nog wat erbij zoeken voordat je het helemaal snapte. In Scratch is in feite de mogelijkheid om bij iemand achter de schermen te kijken geïnstitutionaliseerd en alle handleidingen zijn beschikbaar binnen de omgeving. Voor HTML is natuurlijk ook alles voorhanden zonder dat je de boekhandel hoeft op te zoeken, maar bij Scratch is het allemaal in de omgeving geïntegreerd. Wat natuurlijk niet wil zeggen dat er daarbuiten niets voorhanden is. Want ook buiten de directe Scratch-omgeving kun je informatie over Scratch vinden.

Als je nog even naar figuur 1 kijkt, zie je in de menubalk de link 'Ontdek'. Dit klinkt als leerzaam en interessant en daarom neem ik jullie even mee.

Bij knopje 1 zie je twee icoontjes, één voor 'Projecten' en één voor 'Studio's'. Een Studio is in feite een verzameling van projecten. En zoals eerder gezegd, een project is gewoon één programma dat je maakt. Als je echt met Scratch bezig wilt zijn, is het gewoon aan te raden af en toe eens op ontdekking te gaan, je kunt hier een hoop inspiratie uithalen. Je kunt in je studio's je eigen projecten onderbrengen,



Figuur 3 - Ontdek

maar het is ook mogelijk om projecten van anderen in jouw studio onder te brengen. Op die manier kun je veel gebruikte projecten dus groeperen.

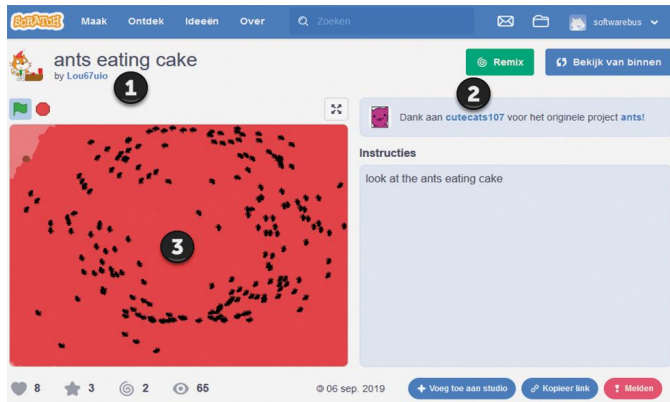
Bij knopje 2 zie je een aantal opties om de projecten (dan wel studio's) te filteren. Je hebt hier de mogelijkheid van 'alles', dus zonder filter, maar je kunt ook op soort zoeken, in dit geval 'Animaties', 'Kunst', 'Spelletjes', 'Muziek', 'Verhalen' en 'Lessen'. Daarnaast zit nog een knopje waarmee je kan kiezen of je de 'Trending', 'Populaire' of 'Recente' bijdragen wilt zien. Ik ben er nog niet uit wat het verschil is tussen 'Trending' en 'Populair' maar ik denk dat 'Trending' hier 'actueel populair' is.

Ik meen dat ik al uitgelegd heb dat je andere Scratchers kunt 'volgen'. Dat houdt dan in dat je snel toegang hebt tot de gedeelde projecten van een andere Scratcher. Dat is weleens praktisch, als je nog even snel terug wilt kijken hoe iemand iets gedaan heeft. Je kunt geen projecten volgen, dat heeft ook niet zo veel zin. Je kunt wel 'studio's' volgen. Als 'SoftwareBus' volg ik bijvoorbeeld Scratcher Wim Boon en Scratcher 'René Suiker' maar ook de studio 'Tutorials!'. Want ik wil met Scratch wel verder, ik wil in de toekomst ook met Scratch en Arduino wat gaan doen en als ik echt nog eens veel tijd over heb, ook kijken hoe ik Scratch kan gebruiken om mijn drone te laten vliegen. Er is dus nog genoeg te leren.

Bij knopje 3 zie je een aantal projecten waar je naar kunt kijken. Je ziet, dit is een selectie 'Projecten' uit de categorie 'Lessen' die 'Trending' zijn. Stuk voor stuk zouden deze dus leerzaam kunnen zijn. Ik ga ze uiteraard niet allemaal hier bespreken, je kunt gerust zelf gaan kijken. En als je iets leuks ziet, dan kun je er gebruik van maken. Even een waarschuwing tussendoor over waar ik tegenaan liep bij het schrijven van dit artikel. Ik werk met twee browsers, Firefox en Chrome, maar in Firefox lukt het me niet om een nieuw project aan te maken, hij blijft maar bezig om dingen aan te maken, maar het komt maar niet af. In Chrome gaat het allemaal wel. Volgens de site van Scratch lukt het in Internet Explorer helemaal niet, dus dat heb ik niet eens geprobeerd. Het kan natuurlijk aan mijn pc liggen, maar lukt het u ook niet: probeer eens een andere browser.

Ik weet nog dat we tijdens de CompUfair van april jl. een presentatie over Scratch hadden door Gerard Vriens van onze collega interessegroep 'artificial intelligence' oftewel 'kunstmatige intelligentie'. Hij hield een leuk betoog over een simulatie van mieren, en ik denk dat we daar eens op verder moeten bouwen. Ik ben dus eens op zoek gegaan, ik kon zijn pagina (project/studio) niet direct vinden, maar ik ben gewoon eens op zoek gegaan naar projecten over mieren en kwam een leuk project tegen, waar we wat uit gaan leren.

<https://scratch.mit.edu/projects/326902212/> is het adres, en de Scratcher die dit deelt is Lou67uio. Hij is een basisschool leerling uit Maleisië, die nog geen jaar met Scratch werkt, dus zo'n beetje tegelijk met jullie en mijzelf is begonnen en hij heeft al ruim 100 projecten gedeeld. Ik ben, ook namens jullie, begonnen hem te volgen en als ik zie hoe productief hij is geweest, dan kunnen we er mogelijk nog veel meer van verwachten. Overigens ook door zaken van anderen opnieuw te gebruiken. Je hoeft in Scratch niet alles zelf uit te vinden om een mooi project te maken.



Figuur 4 - Mieren

Zoals ik al eerder schreef, bovenin (bij knopje 1) vind je de titel en de auteur. Bij knopje 2 zie je wat hij als basis heeft gebruikt. Dat was het project 'ants' van de Scratcher 'cute-cats107'. Dat is iemand uit Canada die al zo'n 3 jaar actief is op Scratch en intussen al meer dan 500 projecten gedeeld heeft. Dat project is een stelletje rondrennende mieren, waarbij je een vergrootglas boven de mieren kunt houden en dan zie je de zaken onder het vergrootglas wat groter. Maar we gaan het niet over het origineel hebben, we gaan eens kijken wat onze jonge Scratcher uit Maleisië ons kan leren.

Bij knopje 3 zie je het speelveld, een aantal rondlopende mieren, die bijna allemaal ongeveer hetzelfde rondje lopen. Je ziet twee kleuren achtergrond, de mieren zijn een stuk cake aan het eten en de donkere kleur is wat ze al op hebben. Kunstmatige intelligentie is het dus niet echt, want in het echt zouden de mieren ook het laatste restje gaan opzoeken en opeten, maar deze mieren hebben een meer 'random' beweging, ze zijn niet erg doelgericht. Desondanks is het een heel leuk én leerzaam project, dus we gaan eens binnen kijken. Het is trouwens leuk om te weten dat je naar binnen kunt terwijl het programma nog loopt, en ook weer terug. Het programma gaat gewoon door, net zoals je tijdens de loop van het programma bepaalde acties ge-

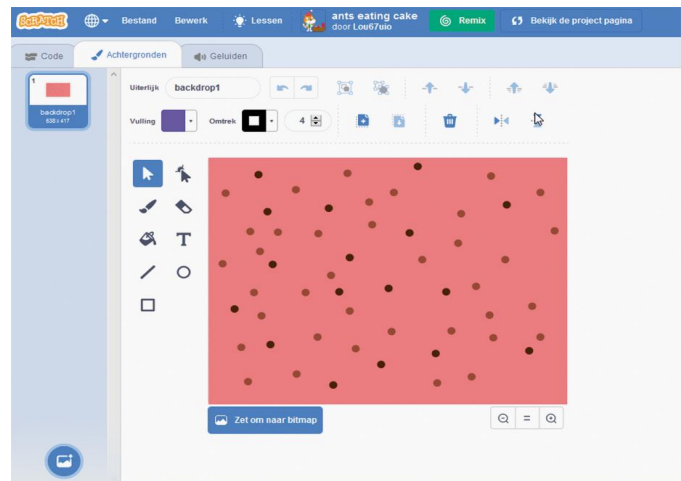


Figuur 5 - Mieren van binnen

woon kunt blijven uitvoeren, zoals de turbomode aanzetten of sprites verplaatsen, enz.

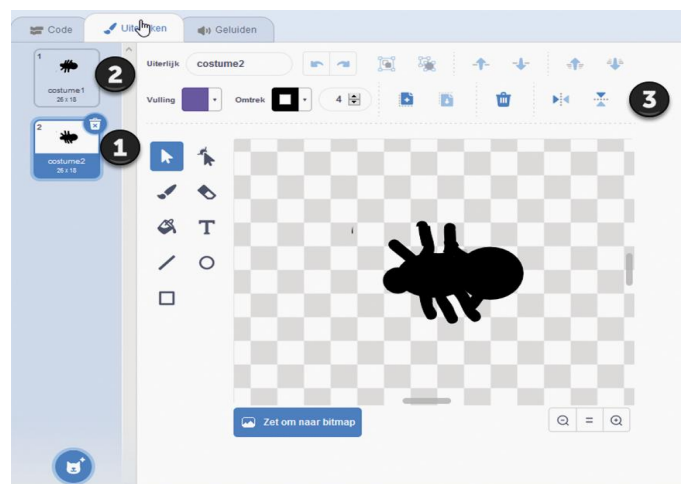
Je ziet hier vier codeblokken. Als je naar het originele project gaat, dan ziet het er iets anders uit, maar ik heb alleen de blokken een beetje verschoven, zodat alles in beeld kwam. En ik moet zeggen, ik ben onder de indruk, want dit hele blok hoort bij één sprite. Als ik in het sprite-blok in de editor kijk, zie ik slechts één sprite. Maar dat is zo mooi, dat maakt het zo krachtig, want hiermee doet deze Scratcher echt alles. Ik ga zo uitleggen wat ik er van snap.

Als we nog verder kijken in de editor, dan zien we ook maar één speelveld, weet u nog hoe dat ging? We zitten met figuur 5 naar het blok 'code' te kijken, maar we gaan nu even naar een ander blok, 'uiterlijkheden'. En dan kies ik in eerste instantie voor het speelveld.



Figuur 6 - Speelveld

Je ziet hier een roze speelveld met wat kleurtjes erin; dit stelt de taart voor waar de mieren zich straks mee voeden. We kunnen ook nog even naar de uiterlijken van de mieren gaan kijken, omdat ik daar straks ook iets anders over wil vertellen, nog naar aanleiding van de lezing van Gerard Vriens, waar ik zojuist aan refereerde. Overigens zie je hier ook, dat 'Uiterlijkheden' nu 'Achtergronden' heet.



Figuur 7 - Mierenkostuums

Hier zie je bij knopje 1 het uiterlijk dat geselecteerd is. Bij knopje 2 kun je het tweede uiterlijk van de sprite bekijken. In het grote veld zie je hoe dit uiterlijk is opgebouwd en hier kan je dus zaken aan veranderen. Je kunt niet alleen alles bewerken, je kunt ook een kopie maken en dan die bewerken. Gerard deed dat voor tijdens de lezing, hij maakte een kopie en spiegelde deze. Door tijdens de beweging van de mier tussen 2 (of meer) uiterlijken te wisselen, lijkt de beweging iets meer op lopen, dan wanneer de mier alleen maar beweegt.

Goed, we gaan nog even naar figuur 5 terug. We zien daar dus een boel code en die is opgedeeld in vier blokken. Blok 1 is een definitieblok, waarin in dit geval het eten is gedefinieerd. Ik moet eerlijk bekennen dat ik niet helemaal de logica doorzie, maar het lijkt mij als volgt. Als de mier in kwestie op een eetbaar stukje komt (dat is alles wat nog niet reeds opgegeten is, dan loopt hij een stukje door, dan zet hij de pen neer en loopt een klein stukje terug. Daarna tilt hij de pen weer op. In blokje 4 zien we dat de pen de kleur heeft van het donkere gedeelte uit figuur 4. Op die manier verdwijnt de cake zogenaamd in de maag van de mier. Spitsvondig!

In blokje 2 zien we de initialisatie van het programma. Dat is netjes gedaan, heel kort en helder. Hij gaat naar een startpositie, ergens rechtsboven (x en y zijn positief, het midden is 0,0). De sprite verdwijnt en alles wordt gewist. Daarna maakt hij dertig klonen van zichzelf (die dus allemaal deze code kennen, deze uiterlijkheden kennen en ook de gedefinieerde geluiden kennen. Vervolgens gaan deze klonen in één richting staan.

Blokken 3 en 4 worden beide getriggerd wanneer ze als kloon worden gestart, dus deze worden dertig keer ‘parallel’ gestart. Dat is zo mooi aan Scratch: er zijn dertig objecten die hun eigen code draaien, maar de code is slechts éénmaal opgenomen, daarna tijdens het uitvoeren wordt dit gekloond. Je kunt je dus voorstellen dat je hier ook echt heel serieuze simulaties mee kunt draaien.

Ik moet toegeven dat ik blokje 3 niet helemaal snap, want hoezo kan de kloon sprite 3 raken? Er is geen sprite 3. En als het om kloon 3 zou gaan, waarom is die specifieke kloon dan relevant? Dus ik denk dat deze ‘als-constructie’ altijd naar costume1 blijft verwijzen.

In blokje 4 wordt geregeld, in een eeuwige lus, dat de mier blijft lopen. Komt hij aan de rand, dan keert hij om. Komt hij iets eetbaars tegen, dan eet hij het op.

Dan komen we nu aan het huiswerk, gebaseerd op dit project.

Opgave 7.1. Kijk wat blok 3 doet en kijk wat er gebeurt als je dit blok weglaat. Wat is het effect op het programma?

Opgave 7.2. Stel dat je de mier tijdens het bewegen telkens van uiterlijk wilt laten veranderen. In welk blok en waar in het programma moet je dan ingrijpen? Hint: je hebt de instructie ‘volgend uiterlijk’ nodig, dan wisselt Scratch automatisch tussen de beschikbare uiterlijken.

Opgave 7.3. We hebben nu dertig klonen rondlopen die de taart opsnoepen. Die starten ook allemaal op dezelfde plek. Pas het programma zo aan dat er tussen de 25 en 35 mieren rond gaan lopen, die ook nog eens niet precies op dezelfde plek starten, maar, zeg ergens in het blokje met een spreiding van evenveel punten als er mieren zijn, ten opzichte van het originele startpunt. Dus, als er dertig mieren zijn, ergens starten met een x tussen 193-15 en 193+15 en een y waarde tussen 51-15 en 51+15.

Opgave 7.4. Voor de echte cracks, pas de beweging van de mieren dusdanig aan, dat ze ‘ruiken’ waar de cake is en dus wat doelgerichter richting de cake lopen en hem dus effectiever opeten. Dan maken we echt een stap op weg richting kunstmatige intelligentie, dus ik verwacht van Gerard Friends wel een goede oplossing

Dan kijk ik nog even terug naar de opgaven van de vorige keer:

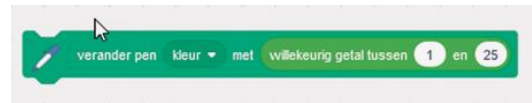
Opgave 6.1. Waarom worden de lijnen niet goed gewist als je de dikte van wissen gelijk maakt aan de dikte van de getekende lijn?

Dat is eigenlijk een merkwaardig fenomeen. Het kan ermee te maken hebben, dat de punten in Scratch niet helemaal parallel lopen aan de punten op je beeldscherm en dat de afwijking per kleur iets anders is. Wat trouwens nog opvalt is, dat als je achtergrond niet effen is, je nog niet kunt wis-

sen op deze manier. Er zou met versie 3 van Scratch de mogelijkheid moeten zijn om ‘transparante’ lijnen te tekenen, maar ik heb het nog niet gevonden. Ik sta open voor suggesties.

Opgave 6.2. Pas het programma zo aan dat de getekende lijnen in willekeurige kleuren zijn. Let op, dit doe je dus in het hoofdprogramma. Het ‘eigen blok’ hoeft hiervoor niet aangepast te worden.

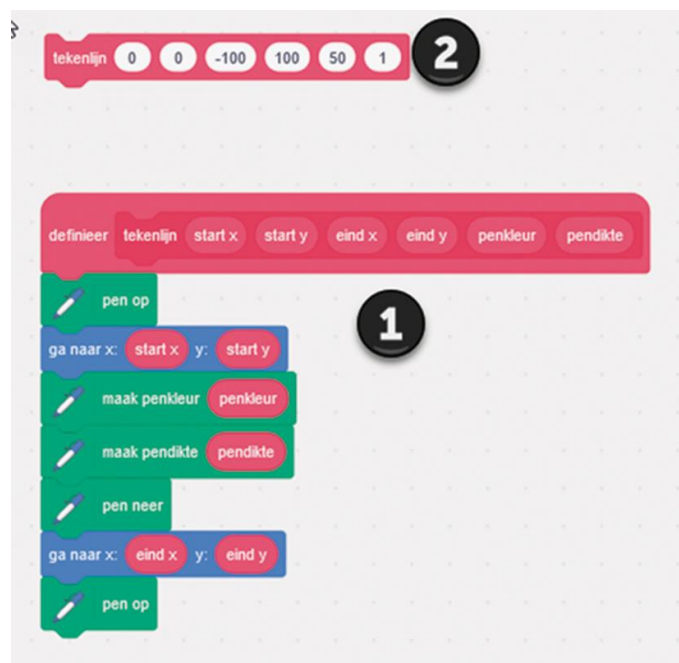
Hiertoe pas je de instructie voor de lijnkleur als volgt aan:



Figuur 8 - Lijn in willekeurige kleur

Opgave 6.3. Maak een eigen blokje dat één lijn tekent, met zes parameters: startpositie x en y, eindpositie x en y en de kleur en de dikte.

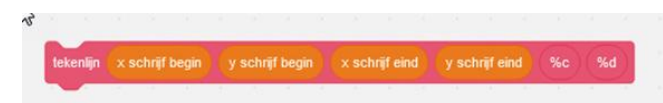
Figuur 9 - Eigen functie ‘tekenlijn’



In blokje 1 zie je de definitie. Je maakt dus een eigen blok, met zes parameters die je tijdens de definitiefase al namen geeft zoals hierboven. De volgorde mag je natuurlijk zelf weten, maar deze lijkt mij vrij logisch.

Met blokje 2 kun je vervolgens je eigen functie even testen, dit was geen onderdeel van de opgave, maar het is altijd handig om je eigen werk te testen voordat je publiceert.

Opgave 6.4. Pas het blok ‘tekenlijn2’ zo aan dat dit ons nieuwe blokje aanroept in plaats van de reeks instructies bij (4). Daarvoor moet je dus even terug naar het vorige nummer, naar figuur 4. Het blokje met nummer 4 kan dan vervangen worden door slechts één instructie:



Figuur 10 - Alternatief voor lijn tekenen

Opgave 6.5. Maak een functie die een lijn wist, dus ongeacht de achtergrond. Maak dan de hele opgave nog een keer compleet, maar dan met een achtergrond die niet effen is, maar bijvoorbeeld de standaard achtergrond ‘Arctic’.

Deze opgave heb ik nog niet uitgewerkt. Ik hoop dat iemand de uitwerking kan indienen, dan controleer ik hem. Ik ga zelf ook op zoek, maar ik heb het nog niet gevonden.