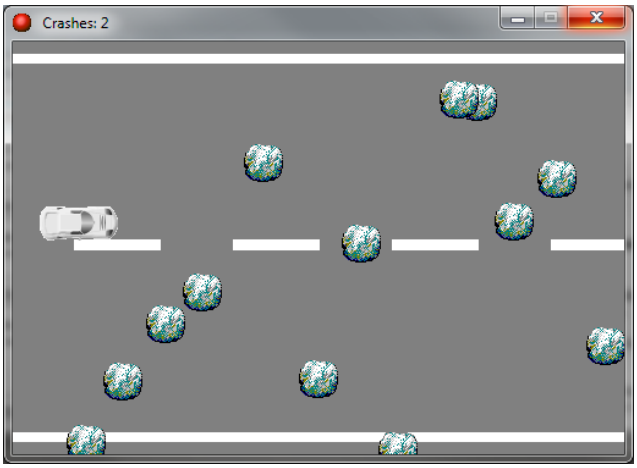


We gaan nu een totaal nieuw soort spel bouwen met GameMaker. Tot nu toe hebben we twee doolhofgames gebouwd. Hierin was een doolhof het speelveld, en liep je met het beertje door het doolhof. Dit spel wordt totaal anders. De bedoeling is dat de speler met een auto over de weg rijdt, en dat de weg is bezaaid met rotsblokken die hij moet zien te ontwijken. Als de speler meer dan 2 keer gecrasht is, verliest hij. Als hij het een minuut volhoudt zonder te verliezen, wint hij. Doordat de achtergrond (de weg) beweegt, wekken we de suggestie dat de auto over de weg rijdt. Doordat de rotsblokken net zo hard als de weg bewegen, en ook nog dezelfde kant op, wekken we de suggestie dat de rotsblokken op de weg liggen.
De crashes houden we bij met de Score-functie.

Speelveld maken

Maak een nieuw speelveld aan. Noem het speelveld Auto's en Rotsen. Maak de breedte 500 pixels. De hoogte van het speelveld hangt samen met de hoogte van het achtergrondplaatje, namelijk 338.
Nu moeten we de bewegende weg als achtergrond instellen. Klik helemaal bovenin het scherm op het knopje **Create a background** (een icoontje van een groen-blauw landschap). Klik op Load Background, en klik éénmaal op het bestand "weg.png". Zoals je in het voorbeeld ziet, stelt dit bestand helemaal niet zo veel voor. Het is een heel kort stukje van de weg, met maar één witte middenstreep. Doordat GameMaker echter beschikt over de mogelijkheid om het achtergrondbestand heel vaak te herhalen, creeëren we zo toch de illusie van een lange weg. Klik op "Open" om de achtergrond te laden. Type bij Name de naam in ("weg") en bevestig met een klik op OK. Ga vervolgens bij de instellingen van de Room naar de tab "backgrounds". Laat alle dingen in dit schermpje staan zoals ze stonden, maar klik op <no background> om de achtergrond "weg" actief te maken. Je ziet dat GameMaker automatisch het plaatje herhaalt, zodat het al op een heuse weg lijkt.
Start je spel maar eens op. Ondanks dat je een fraaie weg hebt gemaakt, ziet het er nog saai en onbeweeglijk uit. De weg moet ook bewegen, want je stapt immers niet in de auto om stil te blijven staan. Onderaan in de tab Backgrounds staan dan ook twee vakjes, "Hor. Speed" en "Vert. Speed". Aangezien de weg naar links moet bewegen, stellen we de horizontale snelheid in op -5 (positieve getallen = naar rechts, negatieve getallen = naar links). Test nog maar eens of je achtergrond nu naar links beweegt.

Speelveld vullen met objecten

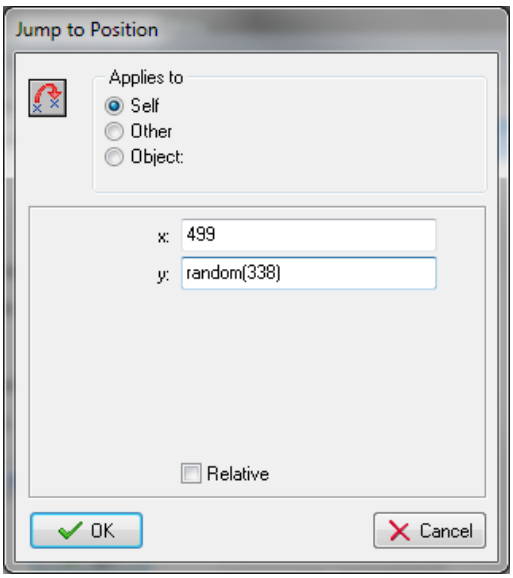
Maak drie sprites. De auto, de auto als hij gecrasht is (het bestand heet car_crash, noem de sprite ook zo) en de rots. Maak vervolgens twee objecten aan, eentje voor de witte auto (noem deze 'car') en eentje voor de rots. Vink bij de rots ook 'Solid' aan.

Plaats de auto ergens in het speelveld, en bezaai de weg met rotsen. Let erop dat over de hele breedte van de weg in elk geval rotsen te vinden zijn.

De rots

Test het spel maar weer eens. De weg beweegt weliswaar, maar de rotsen blijven op hun plek staan. Dat is natuurlijk niet de bedoeling. We moeten twee dingen instellen aan het object Rots. Ten eerste moeten de rotsen bewegen. Doe dit door bij het object een event Create te maken. Alle acties worden dan uitgevoerd zodra het object rots is gecreëerd (in dit geval dus gewoon aan het begin van het spel). Zorg er bij de kolom Actions voor dat de rots met snelheid 5 naar links gaat bewegen. Test je spel nu opnieuw. De rotsen bewegen mooi met de weg mee, het lijkt net alsof ze echt op de weg liggen. Maar… ze verdwijnen buiten beeld, en er komen geen nieuwe rotsen voor in de plaats!

Daarom moet je ten tweede instellen wat er gebeurt als de rots van het scherm verdwenen is. Gelukkig is hier ook een event voor. Klik op Add Event, dan op Other, en dan op Outside Room. Voeg Jump To Position to in de kolom Actions. Waar moet een rots verschijnen als hij links buiten beeld is gerold? Helemaal rechts natuurlijk. Vul daarom bij ‘x’ het getal 499 in (de ‘room’ is 500 breed, dus dan is de rots nét een klein beetje in beeld). Verder willen we dat het onbekend is waar de rots op de weg verschijnt. Als de rots gedurende het hele spel op dezelfde plek op de weg ligt, is de spanning er gauw vanaf. Gelukkig heeft GameMaker hier een functie voor. Die heet ‘random’. Random is een soort dobbelsteen waarbij je kunt instellen hoeveel zijden de dobbelsteen heeft. Elke keer gooit GameMaker weer met de dobbelsteen, en komt er een getal tussen 1 en het ingestelde aantal zijden uit. We willen dat de rots érgens op de weg verschijnt, boven, beneden of in het midden. Het aantal zijden van de dobbelsteen moet dus gelijk zijn aan de ‘hoogte’ van de Room. Als het goed is heb je de hoogte van het speeveld ingesteld op 338. Vul dus bij ‘y’ het volgende in: “random(338)”.



Het is verstandig om weer even te testen of dit nu naar behoren werkt. Als je het goed hebt gedaan verschijnen er nu telken weer ‘nieuwe’ rotsen op de weg, en de plek waar ze verschijnen is niet te voorspellen.

De auto

De rotsen doen nu wat ze moeten doen. De auto kun je echter nog niets mee. Hier gaan we verandering in brengen. Programmeer de vier pijltoetsen zo, dat de speler de auto kan bewegen (snelheid 5). Bedenk echter het volgende: Ook als de auto stilstaat in het beeld, lijkt het voor de speler alsof hij rijdt. Daarom is het verstandig om de snelheid naar rechts op 3 te zetten, zodat de auto maar “een klein beetje harder” gaat als je op de rechter pijltoets drukt. De snelheid naar links daarentegen mag extra hard. Als de auto met snelheid 5 naar links zou bewegen, zou hij net zo hard gaan als de weg en de rotsen. Voor het gevoel van de speler staat de auto dan gewoon stil op de weg. Stel de snelheid naar links dus in op 6. Vergeet ook niet te programmeren dat de auto stopt met bewegen als er geen pijltoets is ingedrukt. Probeer of dit naar behoren werkt.

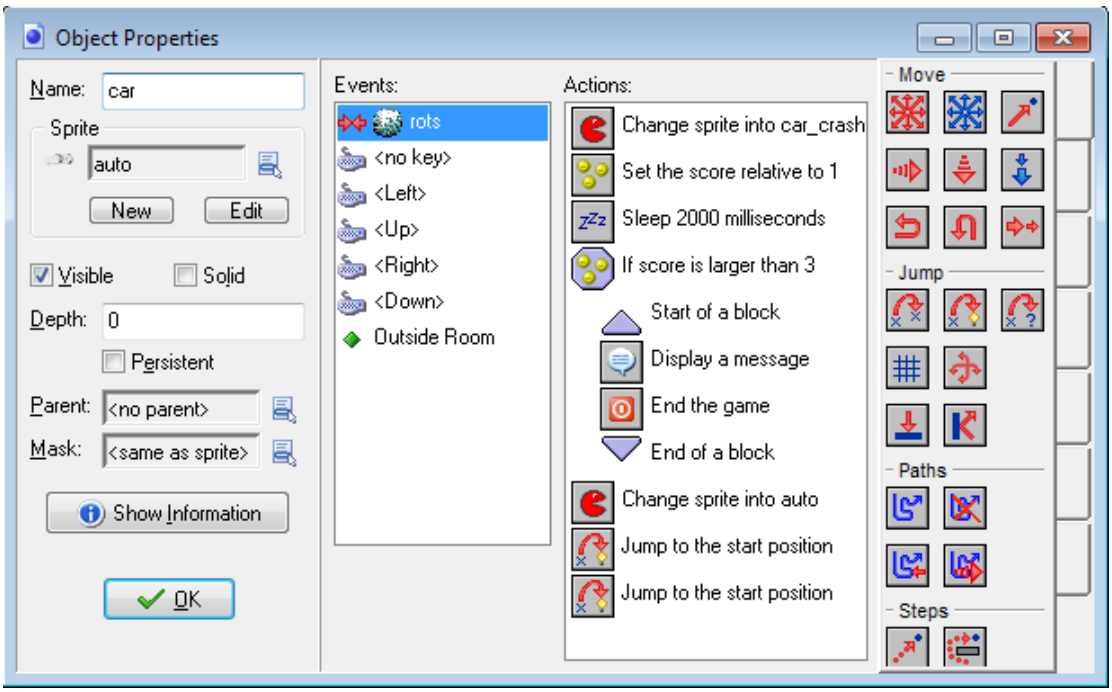
We zijn nog lang niet klaar met programmeren. Er moet ook vanalles gebeuren als de auto tegen een rots aan botst. Als eerste moet het plaatje van de auto veranderen in de sprite ‘car_crash’. Doe dit met het icoontje Change Sprite, dat je vindt in het tabblad main1. Klik hier bij ‘sprite’ op car_crash. Laat de andere instellingen onaangeroerd, en klik op OK. Vervolgens moet de score (daarmee hielden we de crashes bij) één omhoog. Zoek eventueel in de eerdere instructies op hoe dit moest.

Nu is het handig als het spel even 2 seconden stil staat. De speler is zich dan bewust van zijn fout, voordat hij – na twee seconden dus – weet verder kan spelen. Hiervoor is de actie “Sleep” te vinden in het tabblad main2. Hiermee pauzeert het hele spel gedurende een ingestelde tijd. Stel deze tijd in op 2000 milliseconden, en laat de overige instellingen onaangeroerd. Bevestig wederom met OK.
Na deze twee seconden moet het spel weer opnieuw beginnen. Hiervoor moet de auto weer terugveranderen in z’n normale sprite, en moeten de auto en alle stenen weer terugspringen naar de plek waar ze begonnen. Zorg dat als eerste de auto weer terugverandert. Je doet dit weer met het icoontje Change Sprite. Vervolgens moeten de auto en alle rotsen weer op hun startpositie worden geplaatst. Dit doe je met het icoontje “Jump to Start” (te vinden op tabblad move). Sleep dit icoontje naar de kolom Actions. Om de auto terug te plaatsen kun je het vinkje bij Self laten staan. We zitten op dit moment immers in de eigenschappen van het object car. Sleep hierna nogmaals dezelfde actie naar de kolom Actions. Dit keer moeten alle rotsen teruggezet worden. Klik daarom op Object, en verander het object in ‘rots’.

Zo zo, wat een werk! Tijd om het spel maar eens te spelen! Test of je de auto goed kunt rondbewegen, en kijk wat er gebeurt als je op een rots botst. Test ook of het aantal crashes goed wordt bijgehouden (dat zie je in de titelbalk van je spel).

Verliezen

Toch hebben we nog wat dingetjes te veranderen. Als eerste willen we dat je het spel verliest zodra je meer dan twee keer gecrasht bent. Hiervoor moeten we wat acties toevoegen aan de botsing tussen de auto en de rots. Ga naar de eigenschappen van het object car en klik op de event van de botsing met de rots. We plaatsen tussen het Sleep-commando en het tweede Change Sprite-commando wat extra acties, die dus alleen uitgevoerd worden als het aantal crashes (de score dus) groter is dan 2. Als eerste moeten we checken hoe groot de score is. Hiervoor gebruiken we een achthoekig check-commando (ook wel test-commando genoemd) waarmee je al kennis hebt gemaakt tijdens het maken van het tweede spel. Dit keer gebruiken we het “Test Score”-icoontje, te vinden in het tabblad score. Sleep het maar naar de kolom Actions, tussen het Sleep-commando en het tweede Change Sprite-commando. Verander de ‘operation’ in ‘larger than’, en typ bij waarde het getal 2. Bevestig met OK.



Zoals je ook bij het vorige spel hebt gezien, horen bij een check-commando ook altijd een “Start Block”-icoontje en een “End Block”-icoontje. Plaats deze twee direct onder het “Test Score”-icoontje dat je zojuist gemaakt hebt. Alles wat je nu tussen deze twee pijltjes plaatst, wordt alléén uitgevoerd als de score hoger is dan 2. We willen dat dan het spel beëindigd wordt, maar eerst moet er nog een bericht verschijnen om de speler duidelijk te maken dat hij toch écht niet goed genoeg is. Bedenk zelf hoe je dit gaat doen. Kijk eventueel in de instructie van spel 2. Test tenslotte of de toegevoegde acties naar behoren werken.

Het control-object

Zoals je waarschijnlijk wel is opgevallen, staat er in de titelbalk van je spel “Score: 2” als je bijvoorbeeld twee keer gecrasht bent. Eigenlijk is dat raar, want je wilt juist zo weinig mogelijk crashen, en nu lijkt het net alsof je daar juist punten mee verdient! Daarom moet je ervoor zorgen dat – meteen bij het opstarten van het spel – het woord “Score” wordt veranderd in “Crashes”.
Om iets te laten gebeuren aan het begin van het spel moet je een nieuw object maken. Geef dit object de naam ‘control’. Als je geen sprite instelt voor het object, is hij onzichtbaar, en dat is ook precies de bedoeling. Dit control-object kun je gebruiken voor extra dingen die moeten gebeuren maar die niet specifiek met één van de andere objecten te maken hebben. Eén van de voordelen van het control-object is bijvoorbeeld het event “Create” (te herkennen aan een gloeilampje), want alle acties die je toevoegt aan dit event worden aan het begin van het spel uitgevoerd.

Voeg dus een Create-event toe bij het object ‘control’. Sleep vervolgens – uit het tabblad score – het icoontje “Score Caption” naar de Actions. Laat alle instellingen onveranderd behalve de tweede regel, genaamd ‘score caption’. Vul hier de tekst “Crashes: ” in (let op de spatie aan het einde). Nu heb je ingesteld dat meteen aan het begin van het spel de tekst “Score: ” wordt veranderd in de tekst “Crashes: ”. Wanneer je het spel nu uitprobeert zul je echter merken dat het nog niet werkt, de tekst “Score” staat nog steeds in de titelbalk. Dit komt omdat je het control-object nog niet in het speelveld hebt gezet, objecten zijn pas nuttig in je spel als ze ook in het speelveld staan. Zet het control-object dus ergens in je speelveld, bijvoorbeeld helemaal linksboven. Hij is te herkennen aan een klein vraagtekentje, maar deze zie je gelukkig niet tijdens het uitvoeren van je spel. Als je het goed gedaan hebt staat er nu wel “Crashes” in plaats van “Score” in de titelbalk.

Winnen

Het is een beetje jammer als je wel kunt verliezen maar niet kunt winnen. Daarom moet je programmeren dat de speler ook kan winnen. Het zou mooi zijn als de speler zou winnen nadat hij één minuut lang het spel gespeeld heeft (en dus ook niet meer dan 2 keer is gecrasht).
Hiervoor kun je een alarm gebruiken. Dat werkt als een soort kookwekker: Met behulp van een actie in de kolom Actions kun je de kookwekker instellen voor een bepaalde tijd, en met een Event kun je instellen wat er gebeurt als de wekker afloopt.
De bedoeling is dat het alarm 1 minuut na de start van het spel afloopt. Daarom moet het alarm ingesteld worden aan het begin van het spel. Dit doe je met het icoontje “Set Alarm” van het tabblad main2. Je hebt al een event (in het control-object) gemaakt voor het begin van het spel, deze kun je nu ook mooi gebruiken. Het alarm werkt in GameMaker met stappen van 100 milliseconden. Reken zelf uit wat voor getal je dus moet invoeren om het overeen te laten komen met één minuut.
Nu moet je een Alarm-event aanmaken (dat doen we ook in het control-object). Kies hetzelfde alarmnummer als je hierboven gebruikt hebt. Alle acties die je aan dit event toevoegt worden uitgevoerd zodra het alarm afgaat. Bedenk zelf wat hier moet komen te staan, gebruik hiervoor de kennis die je eerder hebt opgedaan. Test of je inderdaad wint als je na één minuut nog in het spel bent!

Blijf op de weg

Je zou denken dat je spel nu echt helemaal af is. Eigenlijk is dat ook zo. Toch moeten we nog één klein dingetje toevoegen. Het is nu namelijk heel makkelijk om het spel te winnen door vals te spelen. Als de speler de auto simpelweg van de weg af stuurt (buiten beeld), gaat het spel nog gewoon door, maar kan hij niet meer geraakt worden door rotsen. Probeer dit maar eens! Hier moeten we nog wel wat aan doen. De makkelijkste manier is om te programmeren dat de auto – zodra hij buiten beeld verdwijnt – naar een willekeurige plek op het speelbord wordt verplaatst. Je hebt al geleerd hoe je kunt controleren of een object buiten beeld is. Gebruik het actie-icoontje "Jump to Random" (op het tabblad move) om de auto naar een willekeurige plek te laten springen. Als je deze naar de Actions sleept, komt er een venster voor de dag. Hier hoef je niets te veranderen, bevestig simpelweg met een klik op OK.

Game Information

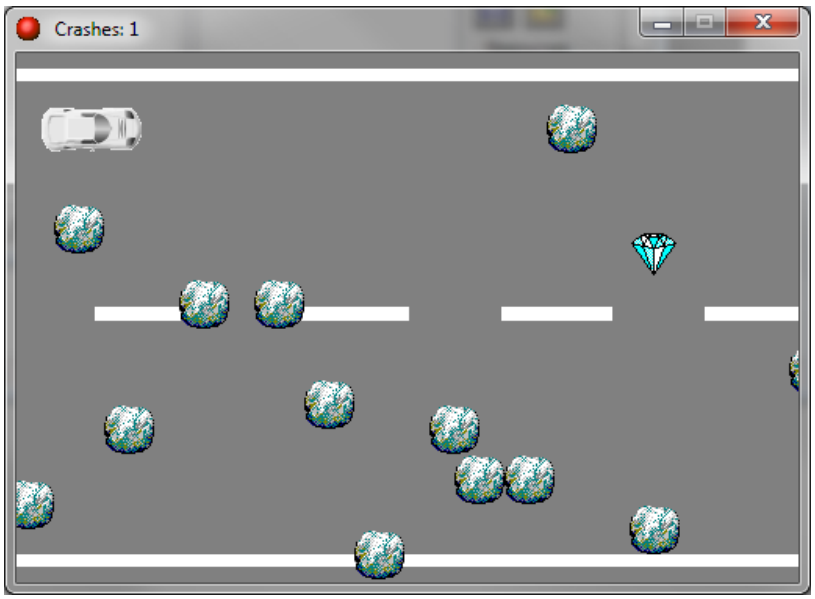
Als je het spel zelf hebt gemaakt ken je de spelregels natuurlijk precies. Daarom is het een goede gewoonte om voorafgaand aan het spelen een korte omschrijving van de spelregels te geven. Klik hiertoe links in het menu op "Game Information". Er verschijnt een tekstscherm, typ bijvoorbeeld in:

> *Hallo, welkom bij dit geweldige spel! Je speelt vandaag met de auto,*
> *en die kun je bewegen met de pijltoetsen. Kijk uit voor de rotsen! Als*
> *je vaker dan 2 keer tegen een rots botst heb je verloren.*
> *Veel speelplezier!*

Sluit af met het groene vinkje in de linkerbovenhoek van het venster.
De Game Information moet verschijnen aan het begin van het spel. Dat betekent dat we nog een actie toevoegen aan het Create-event van het control-object. Dit is het icoontje "Show Info" uit het tabblad main2.

Goed, je hebt nu een compleet spel gebouwd. Het is belangrijk om nog eenmaal goed te testen dat je niets fout hebt gedaan. Test of het rondbewegen van de auto goed werkt, of de rotsen goed werken, of de speler inderdaad verliest als hij te vaak gecrasht is, en of het spel na één minuut inderdaad gewonnen is (gebruik eventueel een stopwatch). Als alles goed werkt: Gefeliciteerd! Je hebt wederom een mooi spel gebouwd! Je hebt eerder al geleerd hoe je een zelfstandig exe-bestand kunt maken, dat je ook thuis (zonder GameMaker) kunt opstarten en spelen.

Bonusopdracht



Heb je alle drie de spellen gebouwd, en is alles klaar? Dan is hier nog een bonusopdracht. Voeg nog een extra functionaliteit toe aan het spel: Een diamant, die elke 15 seconden ergens (willekeurig) op de weg ligt. Als de auto deze diamant raakt, verdwijnt de diamant en gaat het aantal crashes 1 omlaag. Als je dit alles hebt geprogrammeerd is het spel mogelijk te simpel. Compenseer dit door bijvoorbeeld het maximaal aantal crashes te verkleinen, of de spelduur te verlengen. Of gebruik je creativiteit, en verzin zelf een oplossing!