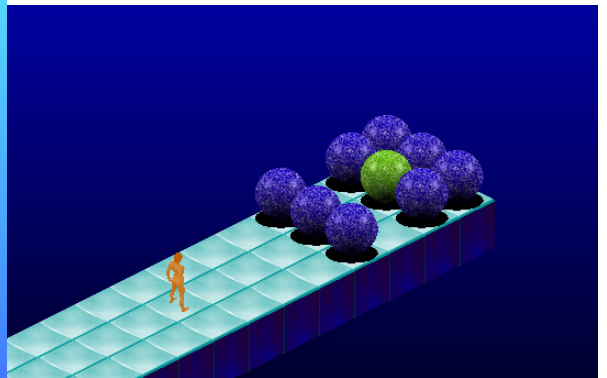
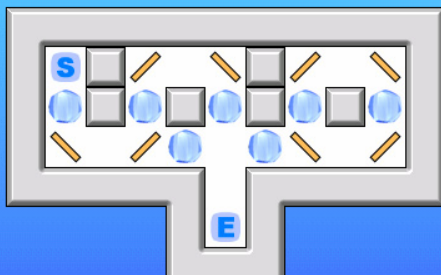


Game Maker 6.1

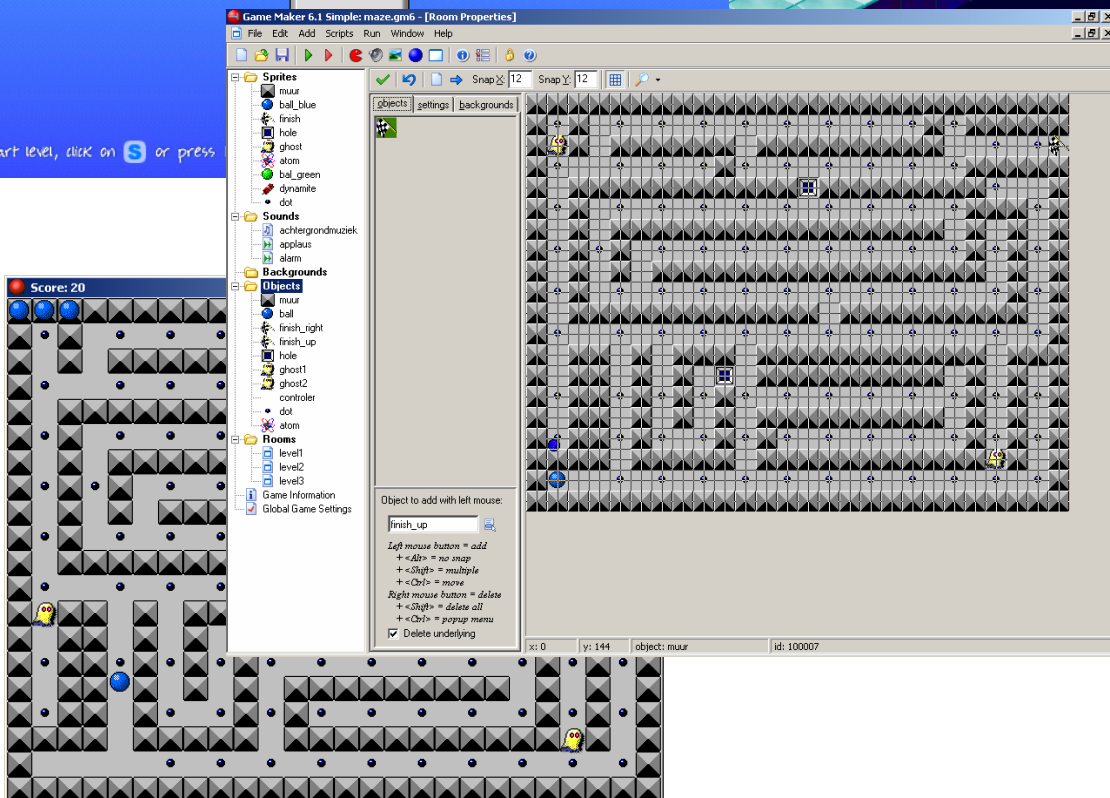
Life's like a maze

Hoofdstuk 3.1



to restart level, click on S or press

LEVEL: 01
RESERVE: 2



Life's like a maze

Het wordt echt tijd om wat je geleerd hebt nog eens te herhalen in een ander soort spel. Je gaat een doolhofspel maken. Eerst maken we weer een omschrijving.

Design document

Description:

In een doolhof van blokjes moet een balletje van begin naar finish gerold worden. Onderweg verzameld de speler zoveel mogelijk punten. Dots zijn 10 punten waard, de finishvlag is 100 punten waard. De speler moet echter oppassen voor spookjes die hem een leven kosten en gaten die al z'n punten laten verdwijnen.

Game objects

Er zijn 6 objecten: De bal, de muur, het gat, het spook, de punten en de vlag. Het muur object heeft een vierkante vorm, de muur staat om het gehele speelveld en vormt de doolhof. Het muur object doet niets, het is er alleen om de bal in het speelveld te houden.

Het bal object heeft de vorm van een bal, het beweegt door de doolhof. De speler bestuurt de bal met de cursortoetsen (pijltoetsen). Als de bal tegen een dot rolt krijgt hij 10 punten. Als hij tegen de finishvlag rolt krijgt hij 100 punten. Als de bal tegen een spook rolt verliest de speler een leven en springt de bal terug naar de begin positie. Als de bal in een gat rolt verliest de speler al zijn punten en springt de bal ook terug naar de begin positie.

Het spook heeft de afbeelding van een spookje. Het spookje beweegt zich op en neer door de doolhof.

De dot heeft geen functie, hij verdwijnt als de bal er tegen aan rolt.

De finishvlag heeft de vorm van een vlag met zwarte en witte blokjes. Als de bal er tegen aan rolt krijgt de speler 100 punten en kan hij zijn naam aan de highscorelijst toevoegen.

Sounds

Er is maar 1 geluid in het spel, het achtergrondgeluid.

Controls

De enige control die de speler heeft zijn de cursortoetsen waarmee hij de bal moet besturen.

Game flow

Als het spel begint staat de stand op 0. De speler heeft dan 3 levens. Het spel begint meteen. Als de speler op <Esc> druk beëindigt hij/zij het spel.


Levels

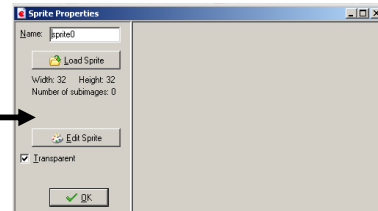
Er is maar een level.

Hoofdstuk 3.1

In de doolhof moet een balletje veilig van het begin naar het einde rollen. Onderweg moet het balletje obstakel, zoals gaten, bommen en spoken ontwijken.

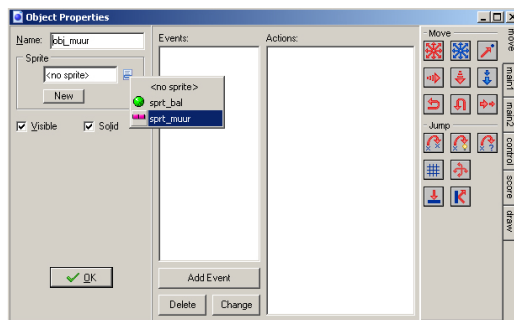
Gebruik voor dit spel de sprites uit het mapje *maze*.


- Start een nieuw spel.
- Sla je spel op als *maze*.
- Maak een sprite voor de muur
-  Geef de sprite de juiste naam.



Denk er aan dat een sprite de afkorting *sprt_* krijgt.

- Doe hetzelfde voor de bal , het spookje , het gat , de punten  en de finishvlag .




- Maak een  object voor de muur. Gebruik ook hier weer de afkorting *obj_*.
- Doe hetzelfde voor de andere sprites.
- Maak nog **geen** events of actions.



Sla je spel op.


Zo alle objecten zijn gemaakt. Je moet omschrijven wat de objecten gaan doen. Gebruik daarbij de volgende omschrijving.

Let op: Eerst komt de omschrijving en daarna de uitleg over hoe je alles moet programmeren. Lees eerst de omschrijving goed door.

Eerst de bal: 

1. De bal moet bewegen als de speler op de pijltjes toetsen drukt, → is dus naar rechts bewegen. Als de speler de pijltjestoets los laat moet het balletje stoppen.
2. De bal moet stoppen als hij tegen een muur aan rolt.
3. De bal heeft 3 levels.
4. De bal moet 10 punten krijgen voor ieder punt dat hij laat verdwijnen.

Hoofdstuk 3.1

Het gat: 

5. De bal moet verdwijnen als hij in een gat rolt en weer aan het begin van de doolhof verschijnen. De speler verliest al zijn punten.

Het spookje: 


6. De bal moet verdwijnen als hij een spookje raakt en weer aan het begin van de doolhof verschijnen. De bal (de speler) verliest daarbij een leven.
7. Het spookje moet van boven naar beneden bewegen.
8. Als het spookje een muur raakt moet hij de andere kant op gaan bewegen.

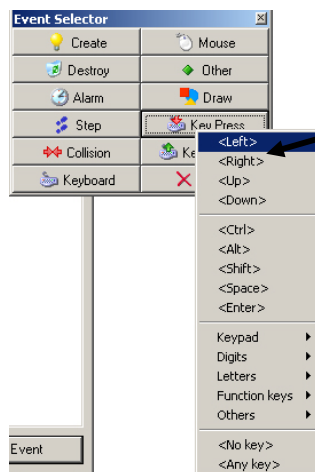
De finishvlag: 


9. De bal moet 100 punten krijgen als hij de finishvlag haalt.

De andere objecten hoeven geen omschrijving te krijgen, want die heb je allemaal al bij de bal genoemd.

We gaan eerst het balletje laten bewegen.

 Dubbelklik op het object *bal*.




 Maak voor iedere *key press* een event aan, dus voor links (left), rechts (right), boven (up) en onder (down).

Nu beweegt het balletje als de speler een pijltjestoets indrukt. Je wilt ook graag dat het balletje stopt met bewegen als de speler de toets los laat. Loslaten is in het engels *release*. Met het event *key release* kun je het balletje dus laten stoppen.

Maak voor alle vier de richtingen een *key release*.

Hoofdstuk 3.1

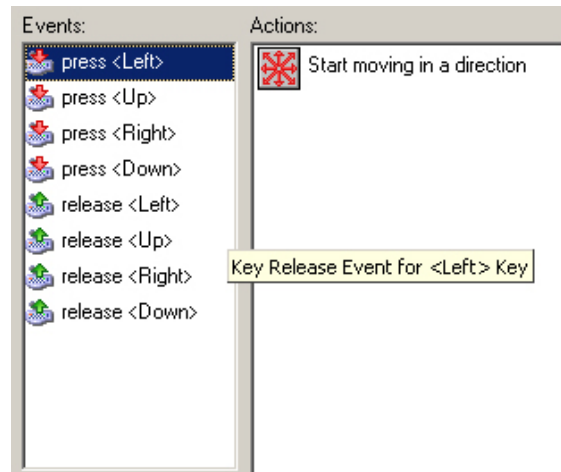
Bedenk goed met welke *action* de bal kunt laten bewegen.

- Sleep bij alle acht de events de  move action in het scherm.
- Selecteer bij iedere *key press* de juiste richting en geef het balletje de snelheid 3.
- Selecteer bij iedere *key release* het blokje in het midden en geef de snelheid 0.



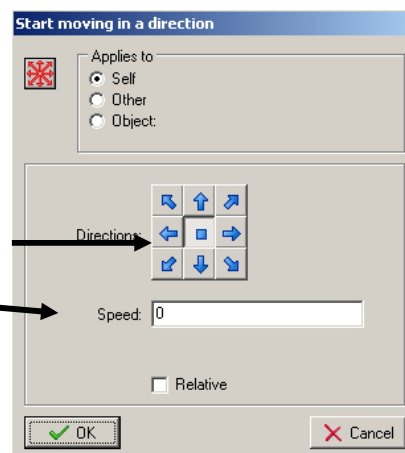
Sla je spel op.

Het eerste punt van de eigenschappen van de bal heb je geprogrammeerd. Ga nu door naar punt 2.



Het balletje moet stoppen als het tegen de muur botst. Dit is dezelfde action als het schuifje in het andere spel.

- Add event  *collision*, met de muur . Hij botst immers tegen de muur.
- Voeg de action  move toe.
- Klik alleen het blokje in het midden aan en kies voor speed 0.



Denk er aan dat de muur wel **solid** moet




zijn.

Het balletje moet stil gaan staan als het tegen de muur botst.

Punt 3, de bal heeft 3 levens. Weet je nog hoe dat in het vorige spel ging? Je had daar een speciale teller voor nodig.


- Maak een object en noem dit *obj_controler*. Je geeft aan dit object geen sprite, hij is immers niet zichtbaar.
- Plaats de *obj_controler* in de doolhof.

Hoofdstuk 3.1

- Maak een event  *collision with* .
- Kies action *set score relative to 100*.
- Kies action  *destroy the instant*. Welke instant moet vernietigd worden? *Self* of *other*

Met de bal zijn we op dit moment klaar, verder met de levens.

De *obj_controller* bepaald straks het aantal levens dat je aan het begin van het spel hebt en wat er gebeurt als al je levens (balletjes) op zijn.

- Add event  *other* → *game start*
- Kies action *set lives* in het tabblad **score**.



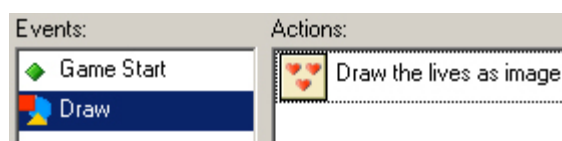
- Geef 3 lives aan het begin van het spel. Dit aantal is niet relative.

Je hebt drie levens aan het begin van het spel.






Natuurlijk wil de speler ook zien hoeveel leven hij heeft.

- Add event *draw*(tekenen).
- Kies action *draw the lives as een image*.
- Kies als image het balletje.

Voor ieder punt dat de speler laat verdwijnen door er tegen aan de rollen krijgt hij/zij 10 punten.



Punt 5 is het verdwijnen van het balletje als het in een gat rolt.

- Dubbelklik op  *hole*.
- Add event  *collision with*  *ball*.
- Kies action  *jump to start position*. Kies *self* of *other*.
- Kies action  *set score to 0*.

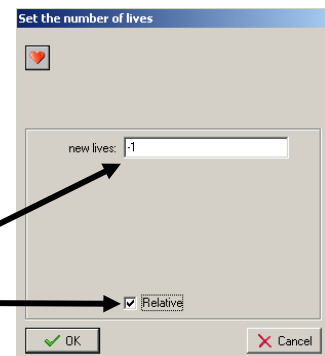
Hoofdstuk 3.1

De bal zal als hij het gat raakt naar het begin van de doolhof springen en de punten worden gereset.

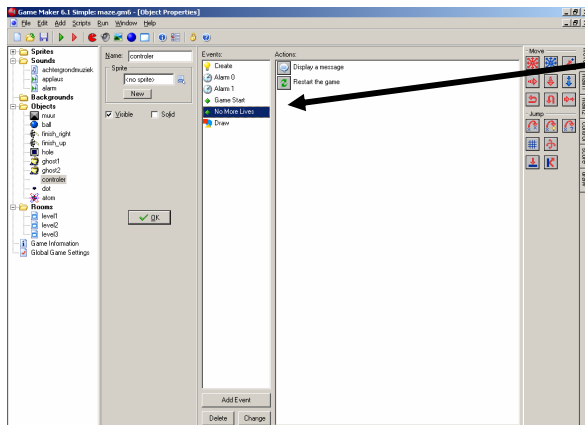
Punt 6, het balletje spring weer naar het begin, maar de speler verliest ook een leven als het spookje geraakt wordt.



Begin weer net als bij punt 5.

- Dubbelklik op  *ghost*.
- Add event  *collision with* .
- Kies event  *jump to start position*.
- Kies event *set live relative to -1*.



Nu kun je zelf de action voor **game over** gaan toevoegen aan de *obj_controler*.



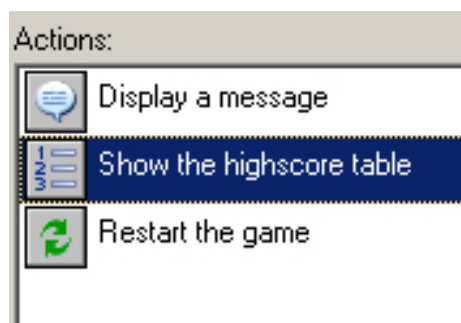
- Add event *no more lives*.
- Laat een bericht laten zien met bijv. "Game over" Dit doe je met de action  uit **main2**.
- Laat het spel opnieuw starten met de action  ook uit **main2**.

De highscore laten zien is natuurlijk ook leuk.

- Voeg uit het tabblad **score** *show highscore* toe aan het event *No more lives*. Plaats deze action tussen de twee eerder gemaakte actions.



Sla je spel op



Hoofdstuk 3.1

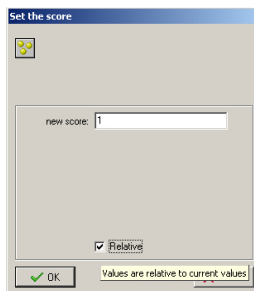
De spookjes in deze room bewegen alleen van boven naar beneden, omdat er later ook nog andere spookjes komen noem je deze spookjes *ghost_ver*. (ver=verticaal).




- Add event  *create*.
- Kies de action  *move* en laat het spookje naar boven of beneden bewegen.
- Maak daarna het event aan om het spookje te laten omkeren als hij tegen een muur botst.



De spookjes gaan bewegen vanaf het moment dat ze in beeld komen.

Als laatste moet de bal 100 punten krijgen als hij de finish haalt. De finish-vlag moet daarna verdwijnen. Dit gaat hetzelfde als met de punten.



- Add event  *collision with* .
- Kies action *set score relative to 100*.
- Kies action  *destroy the instant*. Welke instant moet vernietigd worden? *Self* of *other*.
- Als extra kun je de highscore laten zien.



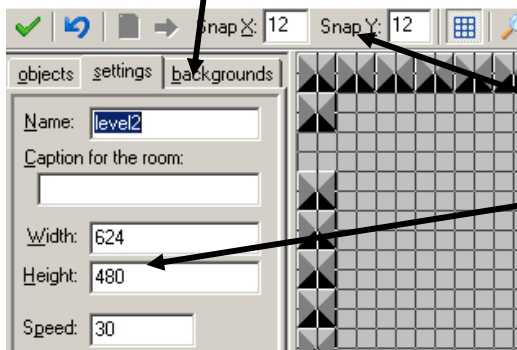
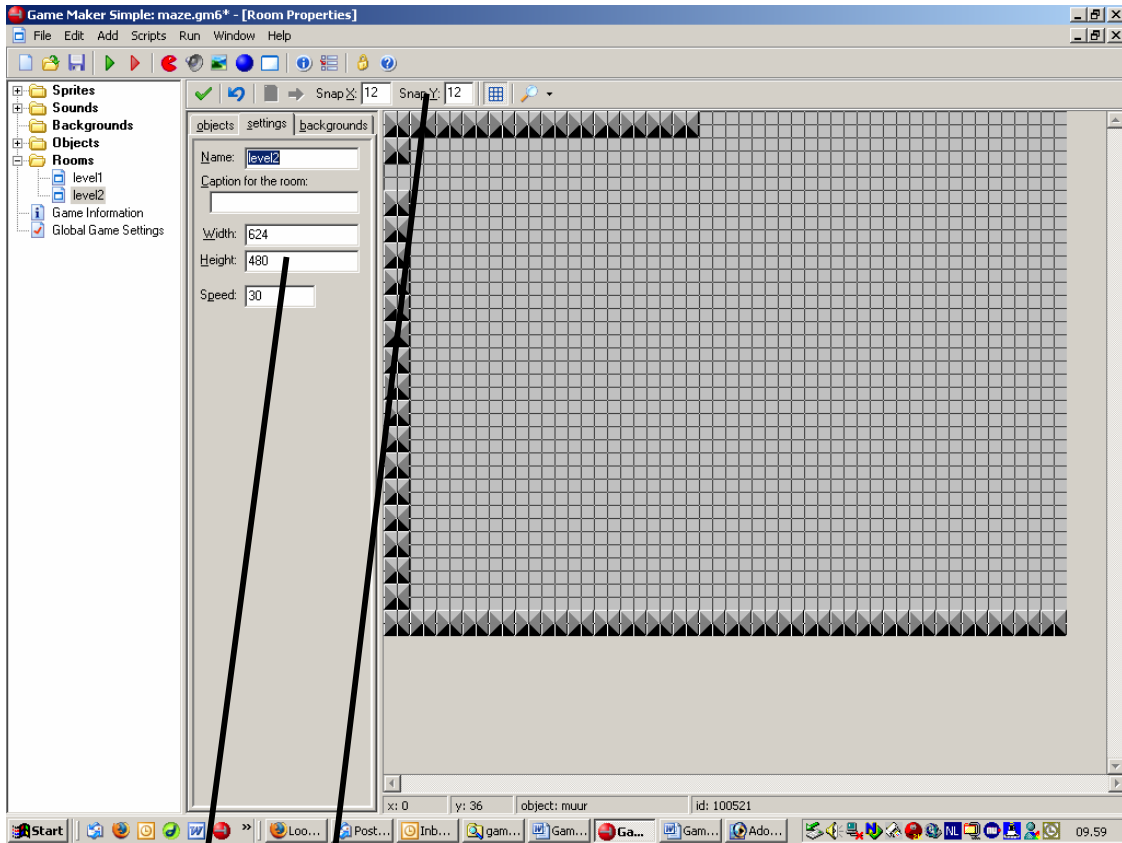
Sla je spel op.

Tijd om het spel in elkaar te zetten.

- Maak een  room.

In de room moet je eerst een aantal settings aanpassen, dit heeft te maken met de grootte van de sprites. De sprites van *maze* zijn namelijk 24 x 24 **pixels**. Het **grid** moet daar op aangepast worden, zodat de sprites er precies inpassen.

Hoofdstuk 3.1



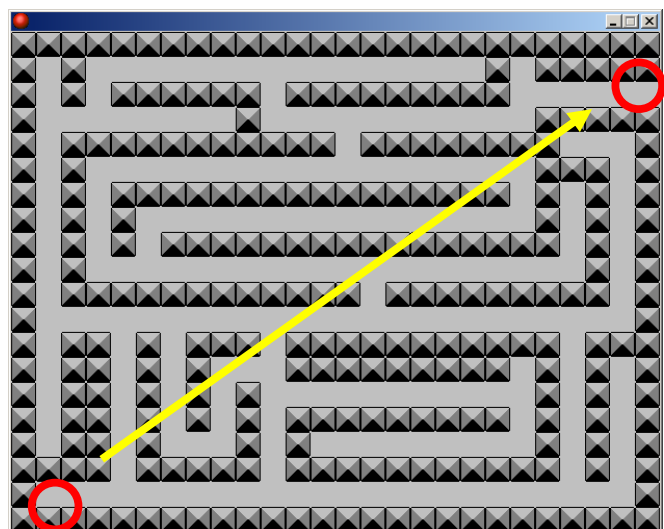
• Maak het grid 12 x 12 pixels.

• Maak de room 624 x 480 pixels.

Nu passen de sprites precies in de room.

• Maak een doolhof in de room. Zorg er voor dat je van het begin naar het einde kunt komen.

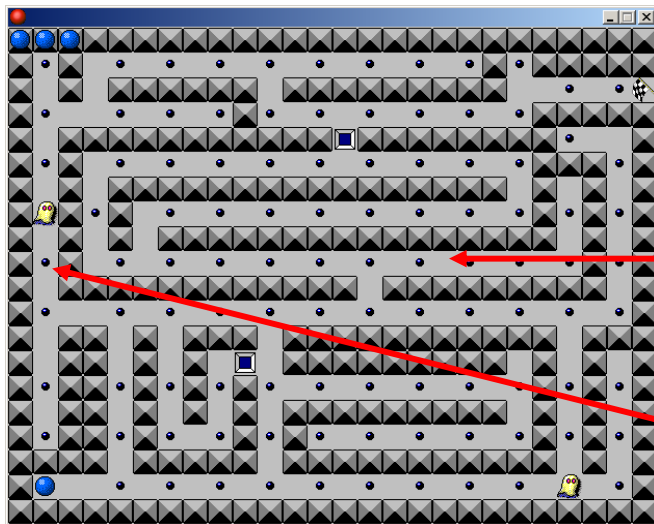
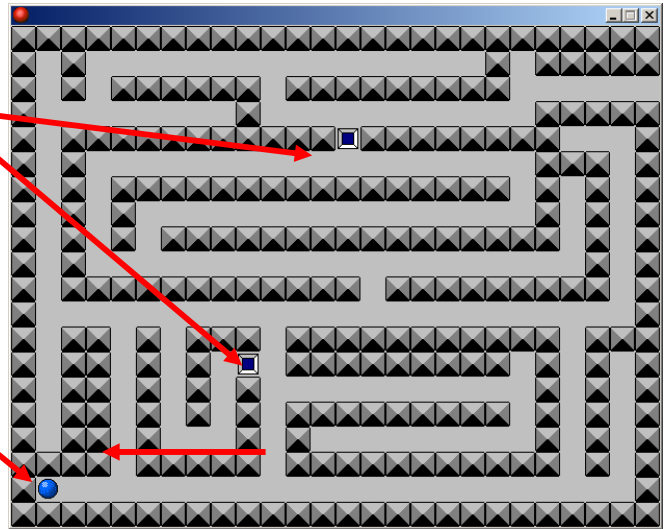
Je doolhof is klaar. Het wordt tijd om er objecten in te plaatsen.



Hoofdstuk 3.1

• Plaats 2 holes (gaten) in je doolhof.

• Plaats de bal aan het begin van de doolhof.



• Plaats een finish-vlag aan het einde van de doolhof.

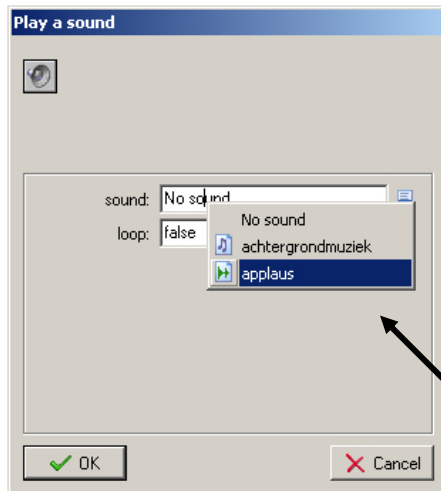
• Plaats de punten in de doolhof.


• Plaats 2 spookjes die van boven naar beneden kunnen bewegen.

Probeer je spel eens uit.

Het is nog wel een beetje stil.



Hoofdstuk 3.1



- Maak  geluidjes aan. Kies een geluidje voor
 - Het balletje als het tegen de muur botst.
 - Het balletje dat in een gat valt.
 - Het balletje dat door een spook geraakt wordt.
 - Het balletje dat de finish haalt.
- Koppel de juiste geluidjes aan het

juiste event.

Naast effectgeluiden is een achtergrondmuziekje ook leuk.

- Maak  een *sound* aan. Kies een .mid bestand voor.
- Koppel het achtergrondmuziekje aan het  *create* event van de *obj_controler*.

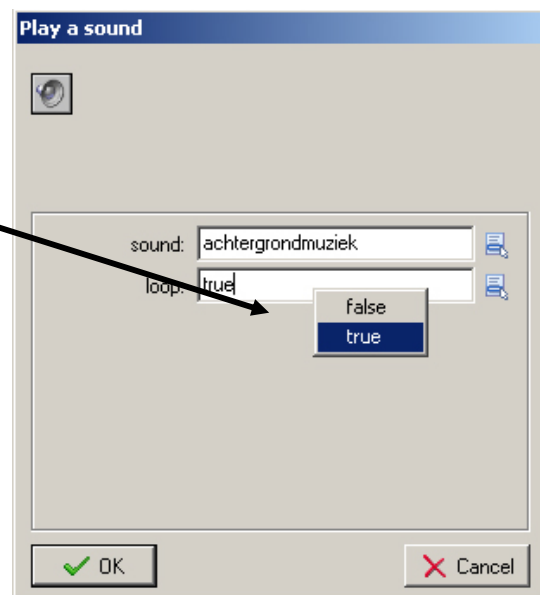
Het achtergrond muziekje moet natuurlijk niet zomaar ophouden als je midden in je spelletje zit.

- Kies daarom naast de juiste muziek voor **loop: *true***.

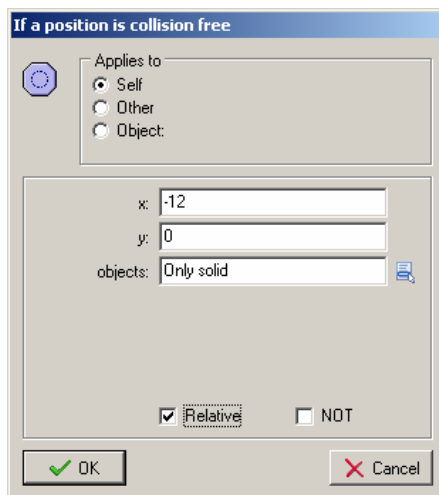
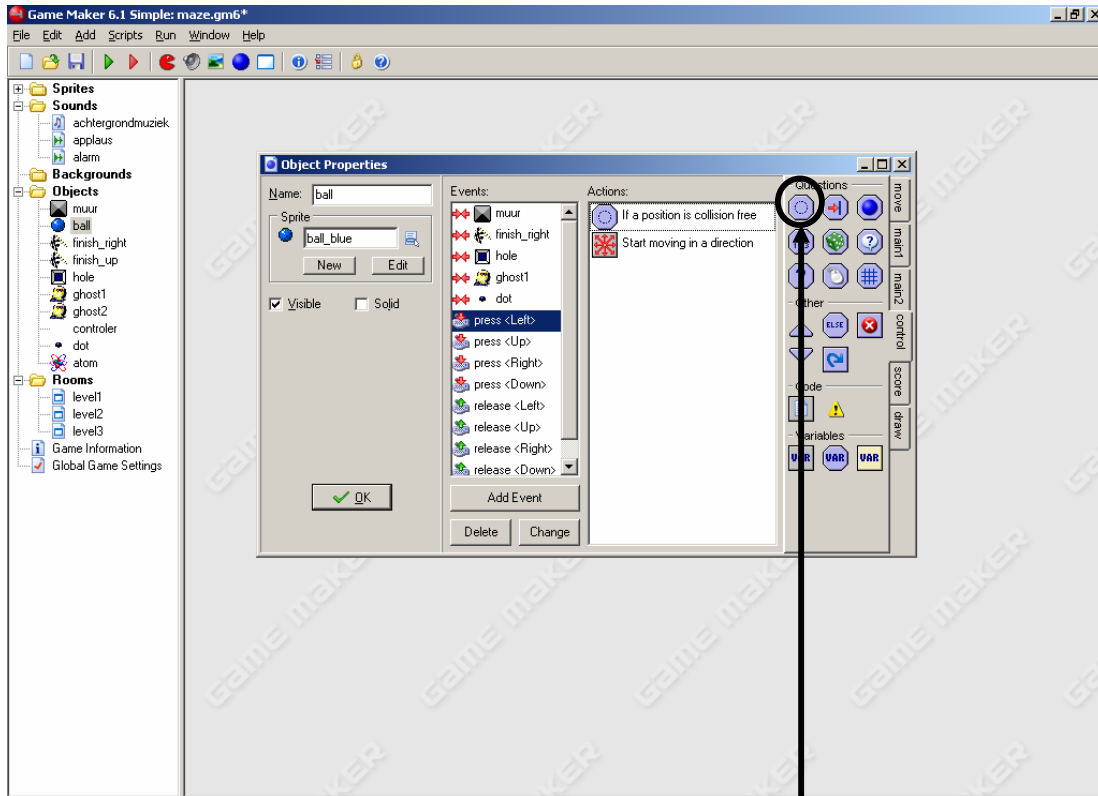


Sla je spel op.

Gefeliciteerd, het eerste level van je spel is al aardig compleet.
Om het balletje nog iets soepeler door het doolhof te laten bewegen voegen we nog een *action* aan het *event: press* toe.



Hoofdstuk 3.1





- Kies in het tabblad *control* voor *If position is collision free* (dat betekent in het Nederlands *als er op die plaats geen botsing plaats vindt...*)
- Plaats deze *action* boven de *action* van *move*.
- Type bij <left> x: -12 en maak het *relative*. Klik op OK
- Type bij <up> y: -12.
- Type bij <right> x: 12.
- Type bij <down> y: 12.
- Maak alles *relative*.


De bal kan nu alleen nog maar een andere kant op gaan als er een opening in de doolhof is.

Hoofdstuk 3.1

De speler moet ook nog weten waar hij aan toe is.

- Maak de  *game information*.
- Klik in het menu helemaal links op  *Game information*.

Je krijgt een blanco formulier waarin je de tekst kunt typen.

- Type de informatie over je spelletje. Je kunt deze informatie zo uit je design document halen. Type ook informatie over bepaalde functies, bijvoorbeeld dat je het spel kunt verlaten door op Esc. te drukken.
- Klik daarna op .

Je wilt dat iemand de informatie kan lezen als hij/zij het spel speelt.

- Kies object *obj_controler*.
- Add event Press Key → Function keys → F1
- Kies action *Show the game info* uit **main2**.
- Maak een executable van je game.
- Schrijf op wat er anders aan je spel is dan dat wat er in het **design document** beschreven is.



Sla je spel op.

Vragen:

1. Wat wordt bedoelt met de afkorting *obj_*?
 - A. *sound*
 - B. *sprite*
 - C. *object*
 - D. *action*
2. Welke uitkomst heeft een variable, true of false?
 - A. alleen **true**
 - B. alleen **false**
 - C. zowel **true** als **false**
 - D. geen van beide

Hoofdstuk 3.1

3. Wat vind je in de *game information*?

- A. Hoe het spelletje werk.
- B. Wie het spelletje gemaakt heeft.
- C. Naam van het spelletje.
- D. Alle bovenstaande informatie.

4. Wat is een pixel?

- A. een plaatje in het spel
- B. een beeldpunt op je scherm
- C. een achtergrondgeluid
- D. een action in het spel

5. Wat gebeurt er met de score als de action '*score is relative to +1*' is?
Iedere keer als deze action plaats vind...










- A. Wordt de score gelijk aan 1.
- B. Wordt er 1 punt van de score afgetrokken.
- C. Wordt er 1 punt bij de score opgeteld.
- D. Blijft de score gelijk.

6. Wat doet een *obj_controler*?

Een *obj_controler* ...

- A. Heeft controle over alle sprites.
- B. Heeft controle over alle objecten die geen action hebben.
- C. Heeft controle over actions die niet aan een object gebonden zijn.
- D. Heeft controle over het geluid.

7. In welke volgorde moeten de actions staan om de speler het bericht 'game over' te geven, de highscore te laten zien en daarna opnieuw met het spel te starten.

- A.   
- B.   
- C.   
- D. 