Creating a Pac Man Game

Adapted from Basic Projects: Game Maker by David Waller

The Design

Before programming any game, all of the objects, events and actions should be planned and designed. The chart below lists the plan for Pacman.

Object	Events	Actions
Pacman	Right cursor pressed	- Pacman moves right - Sprite changed to SPR_PacmanRight
	Left cursor pressed	- Pacman moves left - Sprite changed to SPR_PacmanLeft
	Up cursor key pressed	- Pacman moves up - Sprite changed to SPR_PacmanUp
	Down cursor pressed	- Pacman moves down - Sprite changed to SPR_PacmanDown
	Creation	- Score set to zero - Score displayed - Lives set to three - Lives displayed
	Collision with wall	Movement Stopped
	Collision with food	- Score increases by 1 - Score variable tested - If score equals 25, display a message and move to next level - Food is destroyed
	Collision with monster	- Lives decreased by 1 - Lives variable toted - If lives equals zero then display a message and restart
Monsters 1 to 5	Creation	- Start moving in a vertical or horizontal direction
	Collision with Pacman	- Direction reversed
	Collision with wall	- Direction reversed (without this reversal, collision would continue and take all of the players lives with just one collision)
Food	None	None
Wall	None	None

Most of the concepts needed to create Pacman were previously covered in part one and two of Breakout so the steps for this game will not be as descriptive as the first two.

Let's get to it!

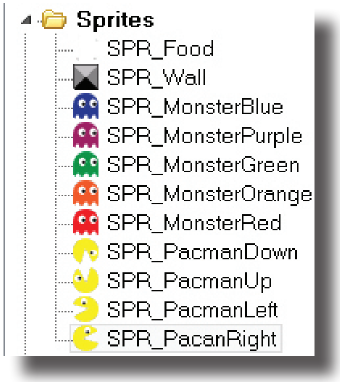
1. Launch Game Maker Lite (NOT Studio)

2. Create a new game. Choose File>New

3. Create sprites for the game. Load these sprites and name them as listed.

Graphics to load	Name of Sprite
Food	SPR_Food
Wall	SPR_Wall
Red Monster	SPR_MonsterRed
Orange Monster	SPR_MonsterOrange
Green Monster	SPR_MonsterGreen
Blue Monster	SPR_MonsterBlue
Purple Monster	SPR_MonsterPurple
Pacman Down	SPR_PacmanDown
Pacman Up	SPR_PacmanUp
Pacman Right	SPR_PacmanRight
Pacman Left	SPR_PacmanLeft

These sprites can be found on the out folder of the shared drive. HSTechC>Period>Out>Pacman. To create a new sprite, click on the red pacman icon. For the SPR_Food, check the center icon.

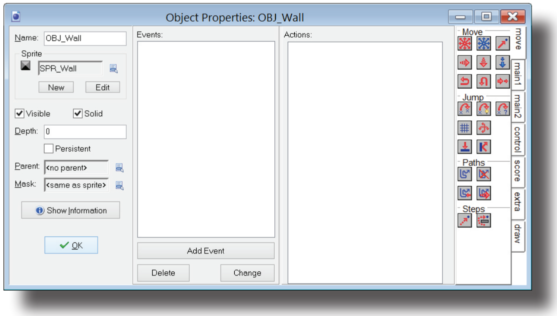


Sprites
- SPR_Food
- SPR_Wall
- SPR_MonsterBlue
- SPR_MonsterPurple
- SPR_MonsterGreen
- SPR_MonsterOrange
- SPR_MonsterRed
- SPR_PacmanDown
- SPR_PacmanUp
- SPR_PacmanLeft
- SPR_PacanRight

4. With all 11 sprites loaded, it's time to create the objects. Create sprites and name them as listed below.

Object	Name of Sprite
OBJ_Food	SPR_Food
OBJ_Wall	SPR_Wall
OBJ_Red Monster	SPR_MonsterRed
OBJ_Orange Monster	SPR_MonsterOrange
OBJ_Green Monster	SPR_MonsterGreen
OBJ_Blue Monster	SPR_MonsterBlue
OBJ_Purple Monster	SPR_MonsterPurple
OBJ_Pacman	SPR_PacmanRight

To create a new object, click on the blue ball icon. Remember to make each monster, the wall, and Pacman solid.



Object Properties: OBJ_Wall

Name: OBJ_Wall

Sprite: SPR_Wall

☑ Visible ☑ Solid

Depth: 0

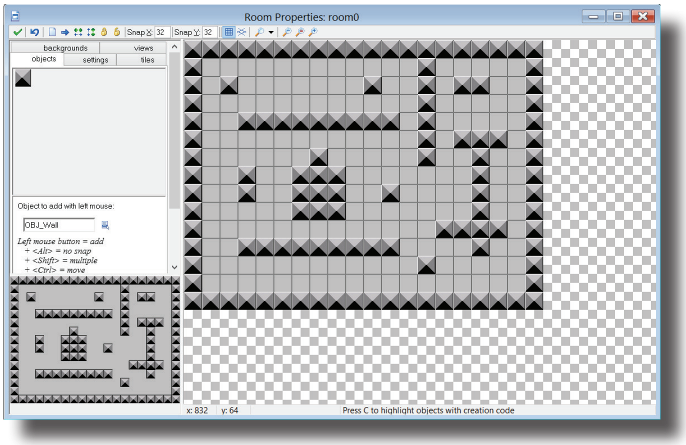
☐ Persistent

Parent: <no parent>

Mask: <same as sprite>

5. With the objects now made, create a new room (click on the icon represented by a white box with the blue bar at the top of it). Most of the sprites in this game a 32 pixels, so change the x and y grid to 32 if it isn't already at 32.

6. Change the 'Object to Add' to OBJ_Wall and create wall around and in the room to create a maze.

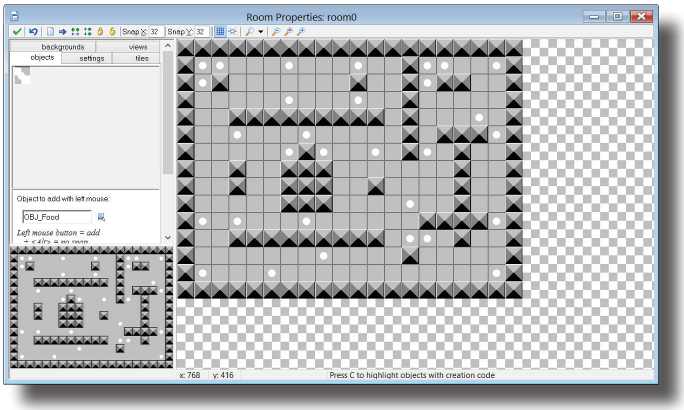


Snap X: 32 Snap Y: 32

Object to add with left mouse: OBJ_Wall

Left mouse button = add
+ <Alt> = no snap
+ <Shift> = multiple
+ <Ctrl> = move

x: 832 y: 64 Press C to highlight objects with creation code



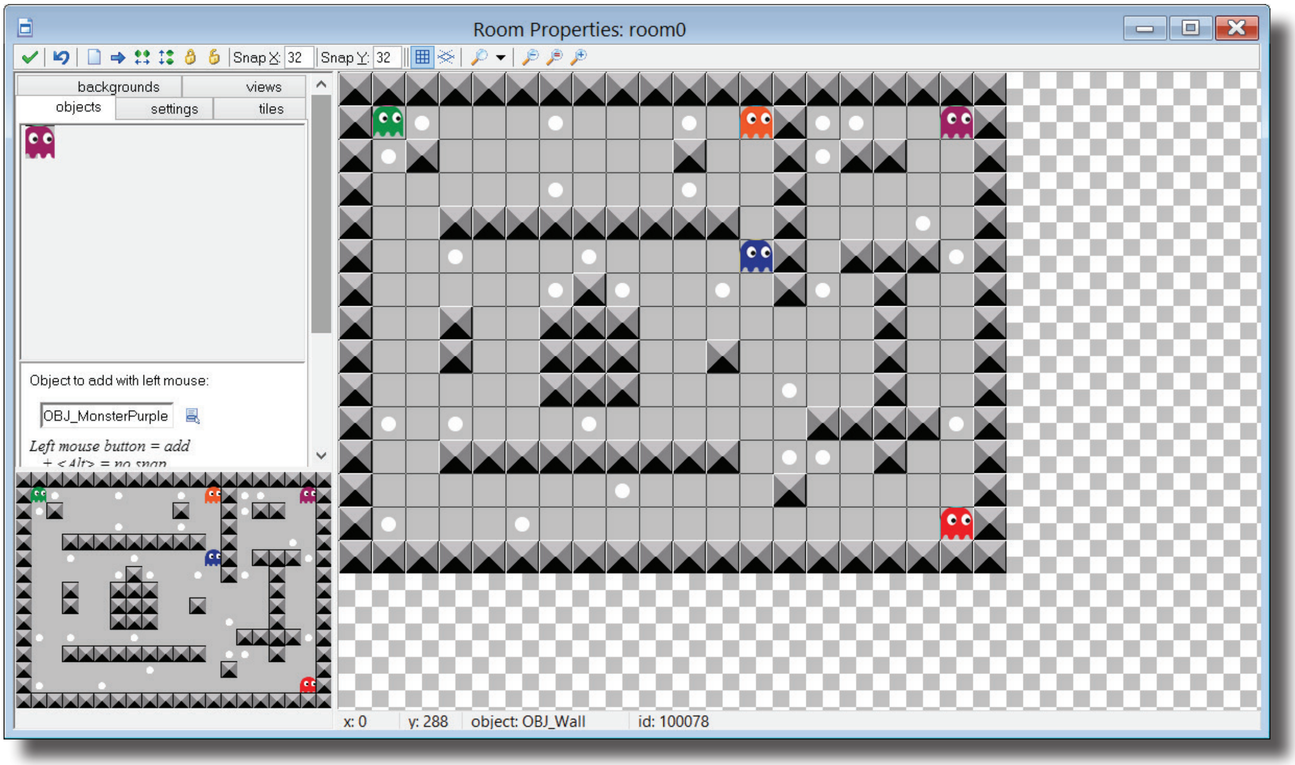
Snap X: 32 Snap Y: 32

Object to add with left mouse: OBJ_Food

Left mouse button = add
+ <Alt> = no snap

x: 768 y: 416 Press C to highlight objects with creation code

7. Now, change the "Object to Add' to OBJ_Food and place 30 instances of food in the room. Beating the first level of Pacman will require 25 points, so place 30 instances of food at various locations within the room.



Snap X: 32 Snap Y: 32

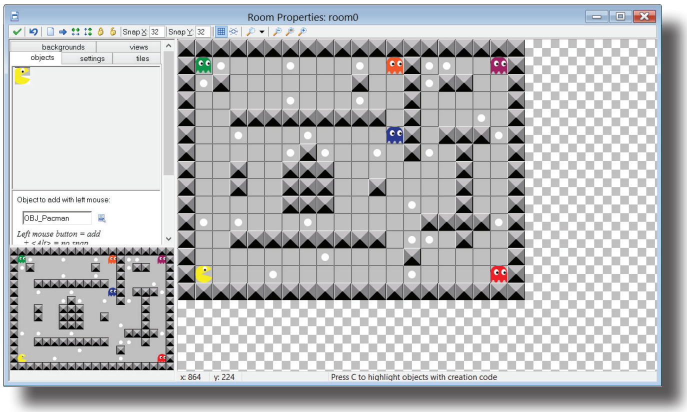
Object to add with left mouse: OBJ_MonsterPurple

Left mouse button = add
+ <Alt> = no snap

x: 0 y: 288 object: OBJ_Wall id: 100078

8. Next, arrange the monster as show in the example. The placement of the monsters is important. The red, orange and blue monsters are set to move horizontally and the purple and green monsters will move vertically. These actions will have to be programmed.

9. Now place one Pacman object in the bottom left corner of the room. With Pacman added, the room is arranged and com-



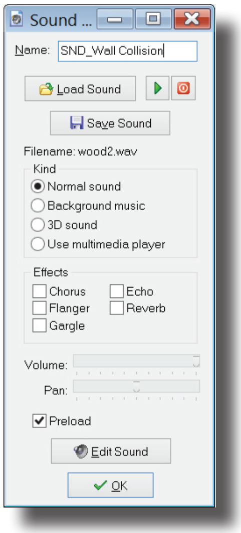
Snap X: 32 Snap Y: 32

Object to add with left mouse: OBJ_Pacman

Left mouse button = add
+ <Alt> = no snap

x: 864 y: 224 Press C to highlight objects with creation code plete. Click on the green check in the top left corner to save the changes and exit the room properties dialogue box.

10. The game needs three collisions between OBJ_Pacman: one with the wall, one with the food and one with the monsters. Each of these collisions needs a sound. Load an appropriate sound for each. To load a sound, click on the load sound icon (the speaker) in the options bar. Remember to check the preload sound icon for each sound.



Name: SND_Wall Collision

Filename: wood2.wav

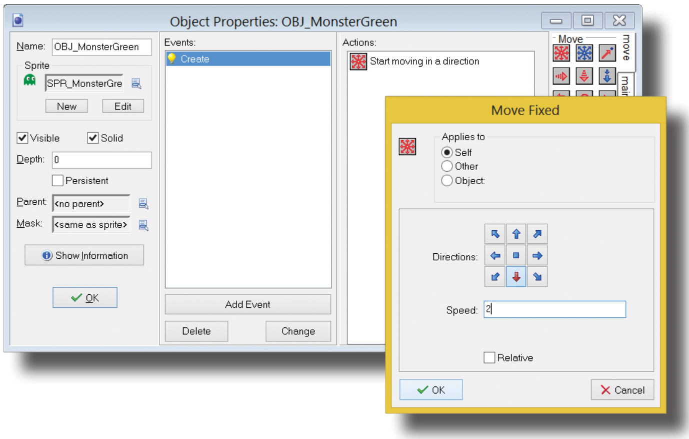
Kind
● Normal sound
○ Background music
○ 3D sound
○ Use multimedia player

Effects
☐ Chorus ☐ Echo
☐ Flanger ☐ Reverb
☐ Gargle

Volume:
Pan:

☑ Preload

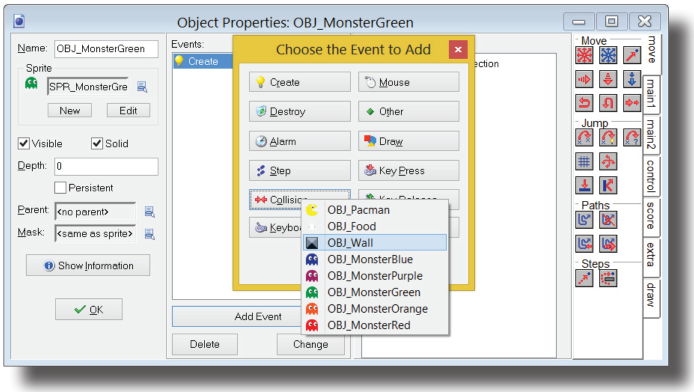
11. Next the actions for the game need to be programmed. Monsters, once created, need to start moving in either a horizontal or vertical direction. When the monster collides with the wall, it should begin moving in the opposite direction. When a monster collides with Pacman it must reverse its direction (otherwise it will take all the lives immediately). Remember that the red, orange and blue monsters must move horizontally and the purple and green monsters will move vertically.



Name: OBJ_MonsterGreen
Sprite: SPR_MonsterGre
☑ Visible ☑ Solid
Depth: 0
☐ Persistent
Parent: <no parent>
Mask: <same as sprite>
Events: Create
Actions: Start moving in a direction

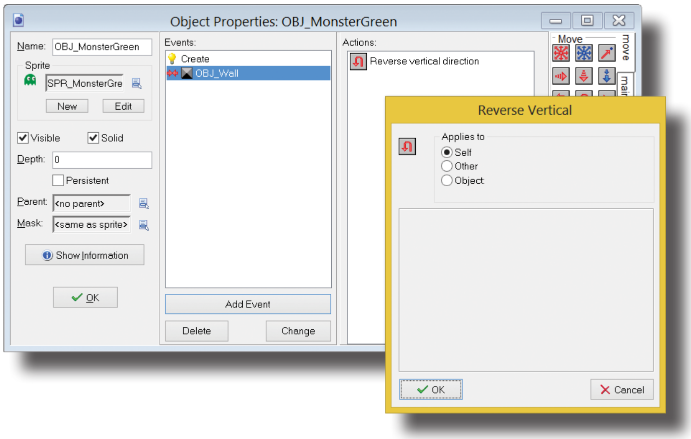
Applies to
● Self
○ Other
○ Object
Speed: 2
☐ Relative

12. Double click on the OBJ_MonserGreen to open its properties. Click the Add Event button and choose Create. Next, drag the move fixed icon (top left icon in the move tab) into the actions area. The green monster should move down when the game starts, so when the new dialogue box appears, select the down arrow. Before closing the dialogue box, change the speed to 2, then click ok.



Name: OBJ_MonsterGreen
Sprite: SPR_MonsterGre
☑ Visible ☑ Solid
Depth: 0
☐ Persistent
Parent: <no parent>
Mask: <same as sprite>
Events: Create

OBJ_Pacman
OBJ_Food
OBJ_Wall
OBJ_MonsterBlue
OBJ_MonsterPurple
OBJ_MonsterGreen
OBJ_MonsterOrange
OBJ_MonsterRed



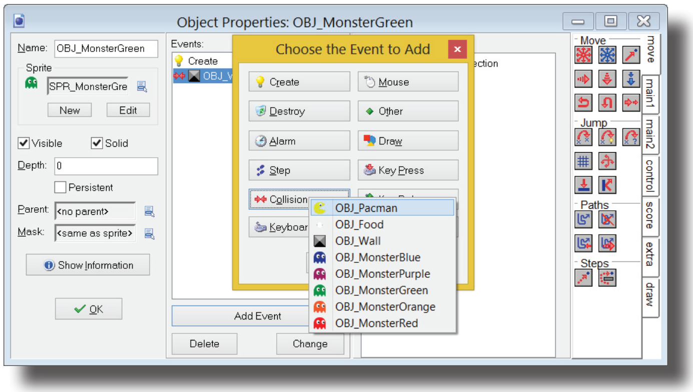
Name: OBJ_MonsterGreen
Sprite: SPR_MonsterGre
☑ Visible ☑ Solid
Depth: 0
☐ Persistent
Parent: <no parent>
Mask: <same as sprite>
Events: Create, OBJ_Wall
Actions: Reverse vertical direction

Applies to
● Self
○ Other
○ Object

13.

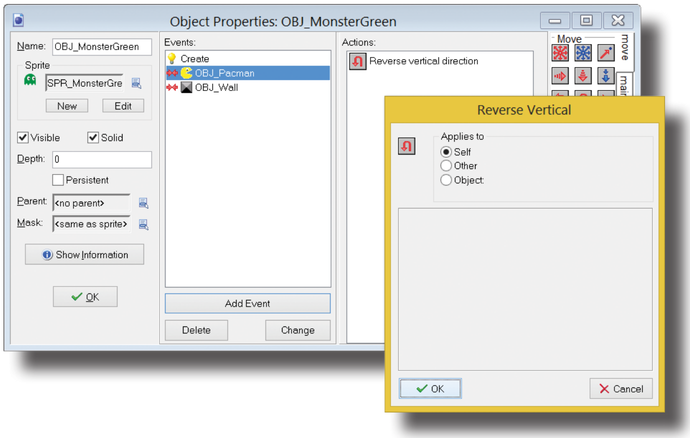
When the monster hits the wall, it has to move the opposite direction. To program this action, choose the Add Event button, click Collision, and when the next dialogue box appears choose OBJ_Wall. While still in the move tab, drag the reverse vertical icon into the actions area. A dialogue box will pop up, here, check to see if self is selected and if it is, choose ok.

14. When Pacman collides with the monster, the same action



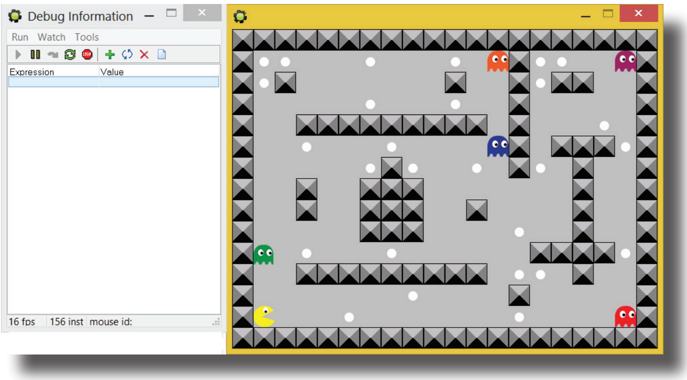
Name: OBJ_MonsterGreen
Sprite: SPR_MonsterGre
☑ Visible ☑ Solid
Depth: 0
☐ Persistent
Parent: <no parent>
Mask: <same as sprite>
Events: Create, OBJ_Wall

OBJ_Pacman
OBJ_Food
OBJ_Wall
OBJ_MonsterBlue
OBJ_MonsterPurple
OBJ_MonsterGreen
OBJ_MonsterOrange
OBJ_MonsterRed



Name: OBJ_MonsterGreen
Sprite: SPR_MonsterGre
☑ Visible ☑ Solid
Depth: 0
☐ Persistent
Parent: <no parent>
Mask: <same as sprite>
Events: Create, OBJ_Pacman, OBJ_Wall
Actions: Reverse vertical direction

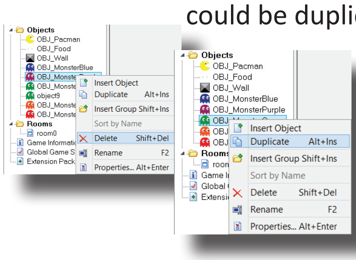
Applies to
● Self
○ Other
○ Object

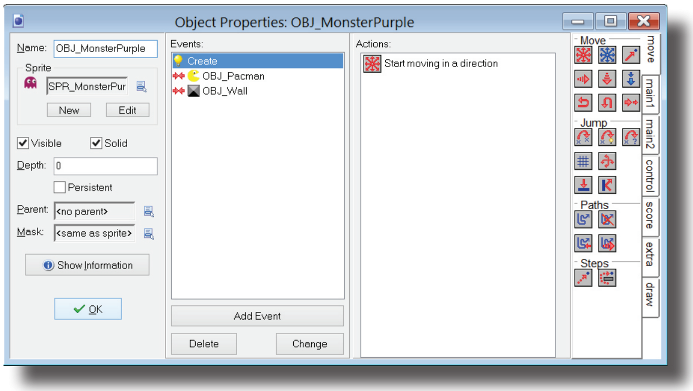


needs to occur. While still in the OBJ_MonsterGreen properties box (if not, double click on OBJ_MonsterGreen), click on the Add Event button, choose collision, and then pick Pacman. Drag in the Reverse Vertical icon, check self and click ok. Click ok to close the OBJ_MonsterGreen Dialogue box.

15. Save the game and click the Run in Debug Mode icon (the red play button) to test it. Does the monster reverse its direction when it collides with Pacman and the wall? If so, OBJ_MonsterGreen is fully programmed.

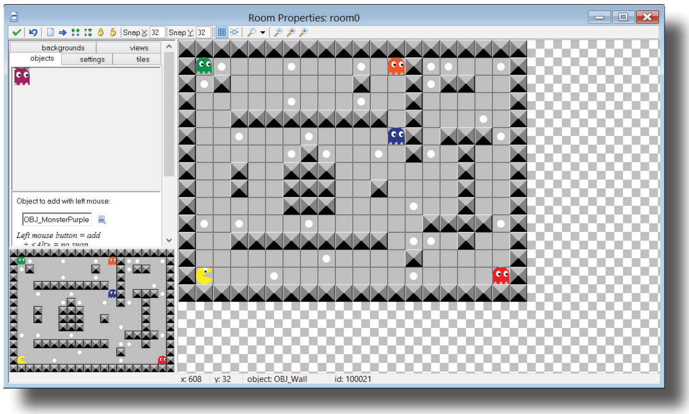
16. Looking at the chart of actions, the OBJ_MonsterPurple needs to perform the same tasks: Move down when the game starts, and reverse its course when it collides with the wall and/or Pacman. The same actions as performed for the OBJ_MonsterGreen could be performed, or OBJ_MonsterGreen could be duplicated, and then the sprite for the duplicate could be changed to the SPR_MonsterPurple. This second option will save some time and work.





Name: OBJ_MonsterPurple
Sprite: SPR_MonsterPur
☑ Visible ☑ Solid
Depth: 0
☐ Persistent
Parent: <no parent>
Mask: <same as sprite>
Events: Create, OBJ_Pacman, OBJ_Wall
Actions: Start moving in a direction

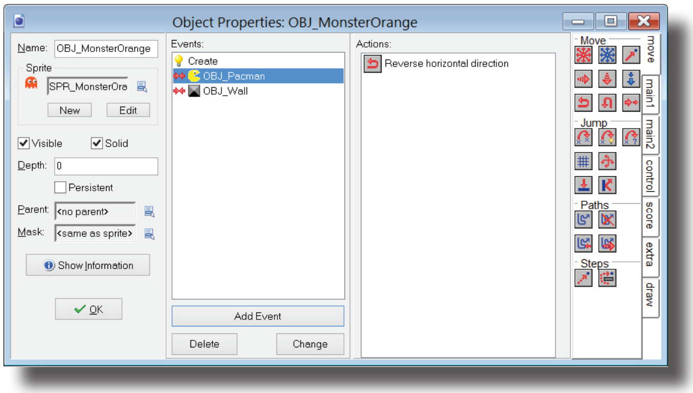
17. Right click on the OBJ_MonsterPurple in the resource tree and choose delete. A new dialogue box will appear asking if the object should be deleted. Choose yes. Now, right click on the OBJ_MonsterGreen and click duplicate. A copy of the monster will be made and it's properties window will appear. Change the name of the copy to OBJ_MonsterPurple and the sprite to SPR_MonsterPurple, then click ok to close the properties window.



Object to add with left mouse:
OBJ_MonsterPurple
Left mouse button = add
x: 608 y: 32 object: OBJ_Wall id: 100021

18. From the resource tree along the left, open the room. The purple monster disappeared from the top right corner of the room when it was originally deleted. It will have to be added to the room again. Change the object to add to OBJ_MonsterPurple and click in the top right corner of the room to add the purple monster to the room. With the monster now in place, click the green check in the top left corner of the room properties box to close it.

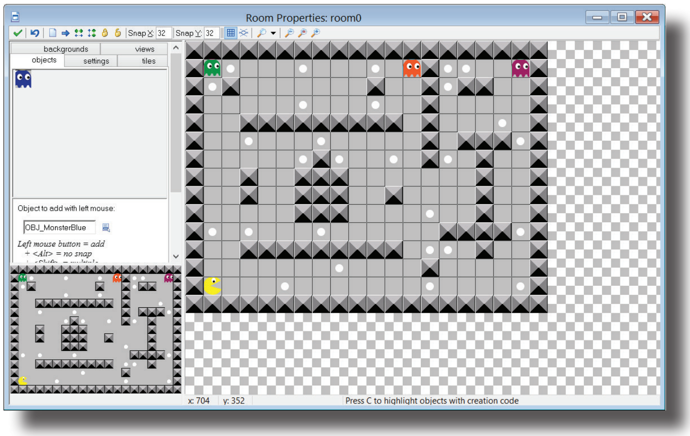
19. Save the game and clcik the Run in Debug mode icon to make sure the purple monster functions properly. just like OBJ_MonsterGreen, it should start the game by moving down and reverse its direction when it collides with the wall or pacman. Two monsters down, three more to go.



Name: OBJ_MonsterOrange
Sprite: SPR_MonsterOra
☑ Visible ☑ Solid
Depth: 0
☐ Persistent
Parent: <no parent>
Mask: <same as sprite>
Events: Create, OBJ_Pacman, OBJ_Wall
Actions: Reverse horizontal direction

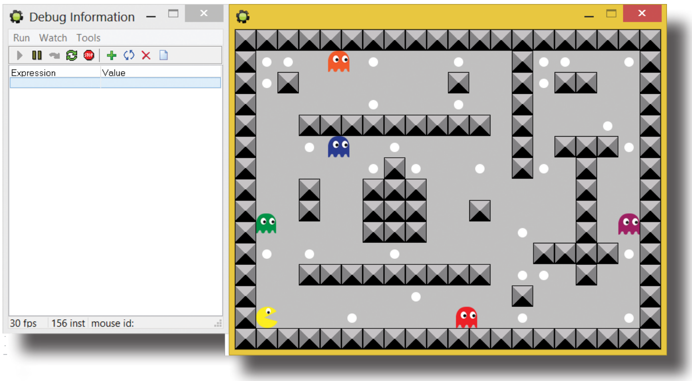
20. Double click on OBJ_MonsterOrange in the resource tree. Click on the Add Event button and choose Create. Next, drag in the Move Fixed icon. On the new dialogue box, clcik the left arrow an set the speed at 2. Choose ok to close the dialogue

box. Next click Add Event>Collision>Obj_Wall. Drag in the Reverse Horizontal icon and click ok. Again click Add Event>Collision>OBJ_Pacman. Drag in the Reverse Horizontal icon and click ok. To close the OBJ_MonsterOrange properties box, click ok once more.



21. With one of the horizontal moving monsters programmed, it can be duplicated to save time for the other two. Start by right clicking on OBJ_MonsterRed and choosing delete. Click yes to permanently delete the object. Then do the same thing for OBJ_MonsterBlue. Duplicate OBJ_MonsterOrange by right clicking on it in the resource tree and choosing duplicate. The property box will appear, here change the name to OBJ_MonsterBlue and change the sprite to SPR_MonsterBlue. Click ok to close the properties window. Duplicate OBJ_Monster Orange again and this time change the object name to OBJ_MonsterRed and the sprite to SPR_MonsterRed. Click ok to close its properties window.

22. Double click on the room in the resource tree. Just like the purple monster earlier, the blue and red monsters disappeared when the objects were deleted. Change the 'Object to add' to OBJ_MonsterBlue and click in the location the same location that he was in earlier. Then change the 'Object to add' to OBJ_MonsterRed and add him to the room too. Click the green check in the top right corner to close the room properties.

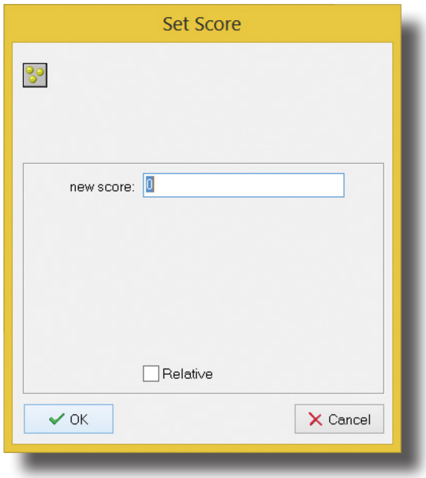


23. Save the game and Run in Debug Mode once again to test the movement and collisions of the orange, blue and red monsters. They should start the game by moving to the left, and change direction when they collide with the wall. Now all of the monsters should be programmed and work correctly.

24. Now OBJ_Pacman needs to be programmed. Unfortunately Pacman is the most involved to program. These are the events and actions that need to be programmed:

Event	Action
Create Event	- Set score to zero - Set Lives to zero - Display Score & Lives
Keyboard Event	- Move in direction of cursor key and change sprite to SPR_PacmanDown, SPR_PacmanUp, SPR_PacmanLeft, SPR_PacmanRight
Collision with Wall	- OBJ_Pacman must stop - Play SND_WallCollision
Collision with OBJ_Food	- Play SND_FoodCollision - Increase Score by 1 - Test variable score - If Score = 25 then display message & move to the next room

Collision with OBJ_Monster	- Play SND_MonsterCollision - Decrease lives by 1 - Test variable lives - If Lives = 0 then display message & restart game

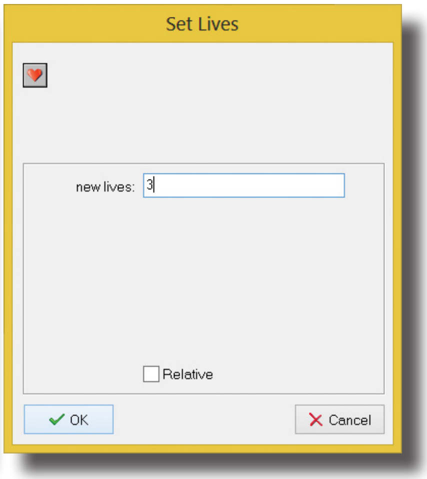


Set Score

new score: 0

□ Relative

✓ OK ✗ Cancel

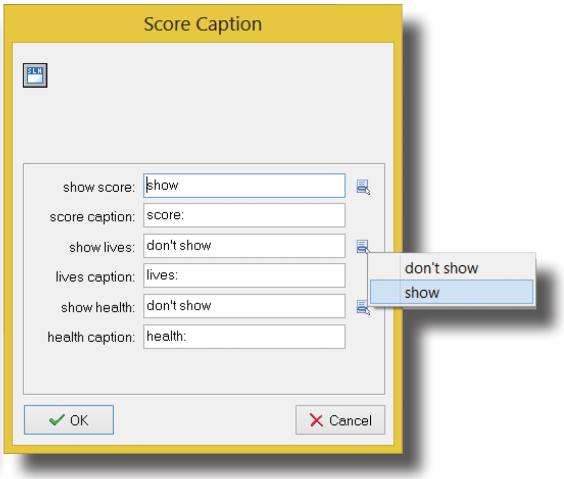


Set Lives

new lives: 3

□ Relative

✓ OK ✗ Cancel



Score Caption

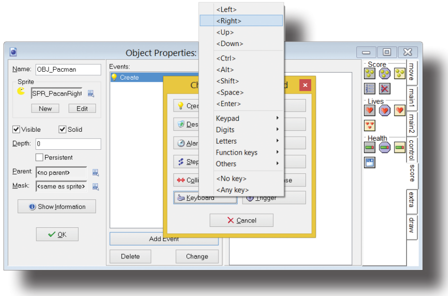
show score: show
score caption: score:
show lives: don't show
lives caption: lives:
show health: don't show
health caption: health:

don't show
show

✓ OK ✗ Cancel

25. Double click on OBJ_Pacman in the resource tree on the left to open its properties. When the game starts, OBJ_Pacman must be created, the score set to zero and the lives set at 3. The score and lives must be displayed too. Click on the Add Event button and choose create. Click on the score tab along the right and drag in the Set Score icon. On the dialogue box, set the score to zero and click ok. Next drag in the Set Lives icon and set the new lives to 3. Click ok. Drag in the Score Caption icon to the actions area. Changes Show Lives and Show Score to Show and click ok. These actions set up the game, but now the movements must be programmed.

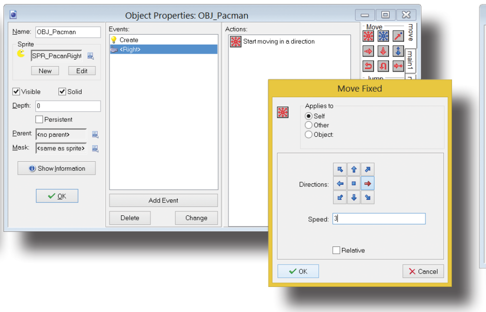
26. With the OBJ_Pacman properties still open (if not, double click on OBJ_Pacman) click AddEvent>Keyboard><Right>. When the right keyboard button is press, the sprite should change to SPR_PacmanRight so that pacman is facing the direction he is moving in and Pacman need to move at a speed of 3. Click on the move tab along the right side and drag the Move Fixed icon into the actions



Object Properties:

Name: OBJ_Pacman
Sprite: SPR_PacanRight
☑ Visible ☑ Solid
Depth: 0
□ Persistent
Parent: <no parent>
Mask: <same as sprite>

Events: Create



Object Properties: OBJ_Pacman

Name: OBJ_Pacman
Sprite: SPR_PacanRight
☑ Visible ☑ Solid
Depth: 0
□ Persistent
Parent: <no parent>
Mask: <same as sprite>

Events: Create, <Right>

Actions: Start moving in a direction

Move Fixed

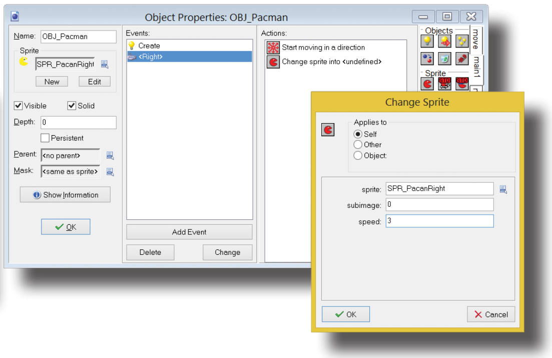
Applies to ● Self ○ Other ○ Object

Directions: →

Speed: 3

□ Relative

✓ OK ✗ Cancel



Object Properties: OBJ_Pacman

Name: OBJ_Pacman
Sprite: SPR_PacanRight
☑ Visible ☑ Solid
Depth: 0
□ Persistent
Parent: <no parent>
Mask: <same as sprite>

Events: Create, <Right>

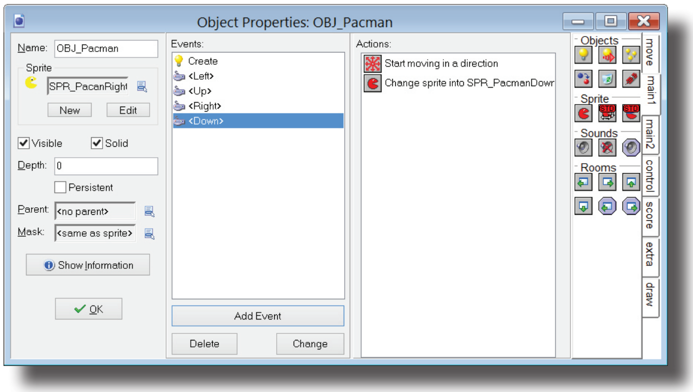
Actions:
Start moving in a direction
Change sprite into <undefined>

Change Sprite

Applies to ● Self ○ Other ○ Object

sprite: SPR_PacanRight
subimage: 0
speed: 3

✓ OK ✗ Cancel



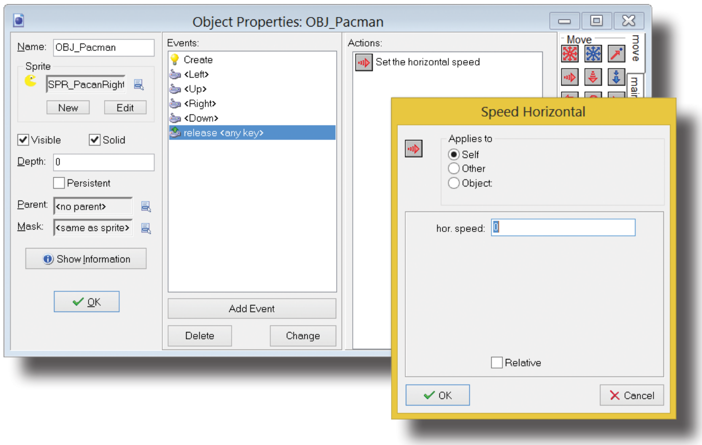
Object Properties: OBJ_Pacman

Name: OBJ_Pacman
Sprite: SPR_PacanRight
☑ Visible ☑ Solid
Depth: 0
□ Persistent
Parent: <no parent>
Mask: <same as sprite>

Events:
Create
<Left>
<Up>
<Right>
<Down>

Actions:
Start moving in a direction
Change sprite into SPR_PacmanDowr

area. On the ew dialogue box, click on the right arrow and set the speed to 3. Then click ok. The movement is set, but the sprite still needs to change so choose the main1 tab and drag the Change Sprite icon into the actions area. On the new dialogue box, change the speed to 3and the sprite to SPR_PacmanRight. Click ok to close the dialogue box.

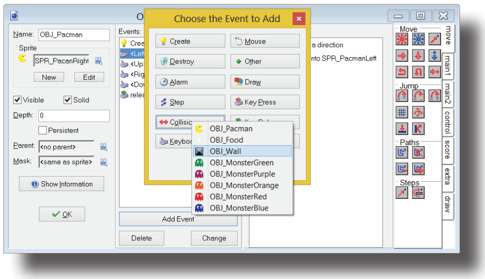

Object Properties: OBJ_Pacman
Name: OBJ_Pacman
Sprite: SPR_PacanRight
☑ Visible ☑ Solid
Depth: 0
□ Persistent
Parent: <no parent>
Mask: <same as sprite>
Events: Create, <Left>, <Up>, <Right>, <Down>, release <any key>
Actions: Set the horizontal speed
Speed Horizontal
Applies to: ● Self ○ Other ○ Object:
hor. speed: 0
□ Relative

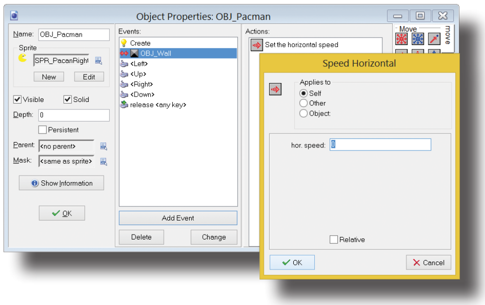
27. Repeat step 26 for a the left, up, and down arrows. Each time, change the sprite to the corresponding pacman (for the left arrow, SPR_PacmanLeft) and set the speed at 3. There should be a create event, <Left>, <Right>, <Up>, and <Down> event.

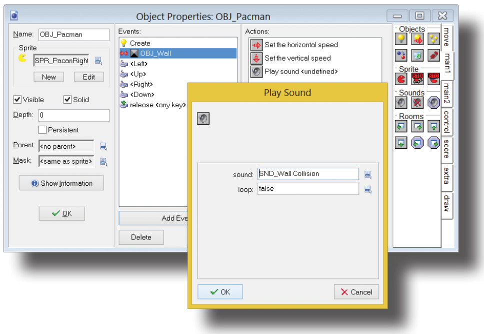
28. When any arrow key is released, pacman needs to stop moving. With the OBJ_Pacman properties window still open (if not, double clcik on OBJ_Pacman in the resource tree). Click on the Add Event button>Key Release><Any Key>. Click on the move tab at the right side and drag the Speed Horizontal icon into the actions area. In the dialogue box, change the speed to 0 and click ok. Do the same thing with the Speed Vertical icon. That should do it for OBJ_Pacman's movements. Click ok to close the OBJ_Pacman properties window.

29. Save the game and test it by clicking on the Run in Debug Mode icon. Does pacman move in each of the directions? Does the sprite change each time so that pacman is facing the direction he is moving? Does he stop moving when a key is released?

30. So pacman and the ghost should both move correctly, but the collisions aren't programmed yet. Pacman can collide with the monsters, food, and the wall. Each collision will trigger a sound to play and possibly the score(if it's a collision with food) or lives (if it's a collision with a monster) to change.


Choose the Event to Add
Collision: OBJ_Pacman, OBJ_Food, OBJ_Wall, OBJ_MonsterGreen, OBJ_MonsterPurple, OBJ_MonsterOrange, OBJ_MonsterRed, OBJ_MonsterBlue

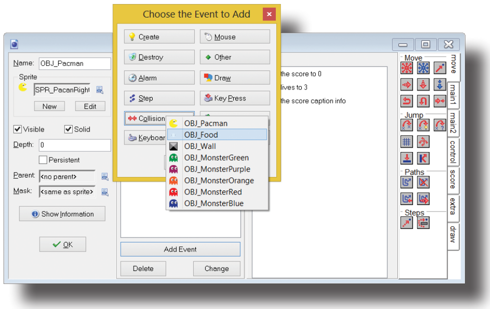

Events: Create, OBJ_Wall, <Left>, <Up>, <Right>, <Down>, release <any key>
Actions: Set the horizontal speed
Speed Horizontal
Applies to: ● Self ○ Other ○ Object:
hor. speed: 0
□ Relative

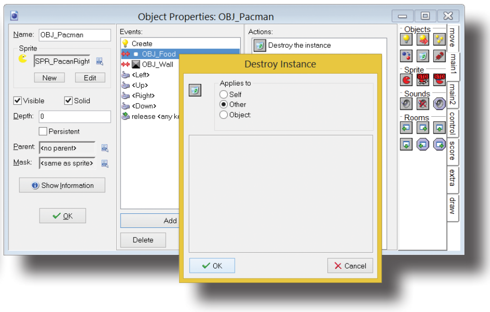

Actions: Set the horizontal speed, Set the vertical speed, Play sound <undefined>
Play Sound
sound: SND_Wall Collision
loop: false

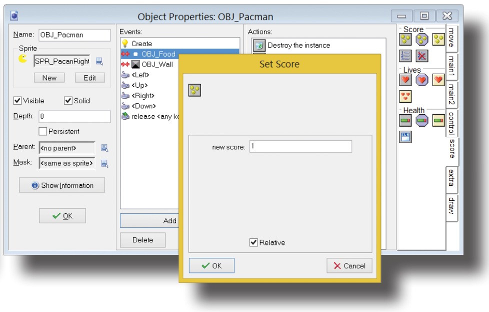
31. Double click on OBJ_Pacman to open his properties. Click Add Event>Collision>OBJ_Wall. The movement (both horizontal and vertical) must be set to zero when pacman collides with the wall. Drag the Speed Horizontal icon (in the move tab) to the actions area and set the speed to zero. Do the same for the Speed Vertical icon. This will make pacman stop moving upon a collision with the wall. The collision with the wall still needs a sound. Click on the main 1 tab and drag in the Play Sound icon . On the new dialogue box, change the sound to SND_WallCollision and click ok to close the window. The wall collision is now complete.

32. Next, create the food collision. When pacman collides with OBJ_Food, that particular instance of food needs to disappear (or be destroyed), the score should increase by 1, be tested to see if the player has reached 25 and if they have, a message should be shown and the next room should load.

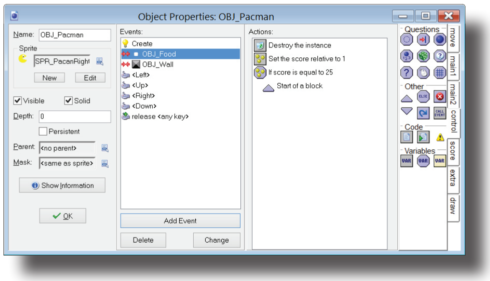
33. While still in OBJ_Pacman's properties, click the Add Event button>Colli-


Choose the Event to Add
Collision: OBJ_Pacman, OBJ_Food, OBJ_Wall, OBJ_MonsterGreen, OBJ_MonsterPurple, OBJ_MonsterOrange, OBJ_MonsterRed, OBJ_MonsterBlue


Events: Create, OBJ_Food, OBJ_Wall, <Left>, <Up>, <Right>, <Down>, release <any key>
Actions: Destroy the instance
Destroy Instance
Applies to: ○ Self ● Other ○ Object:


Events: Create, OBJ_Food, OBJ_Wall, <Left>, <Up>, <Right>, <Down>, release <any key>
Actions: Destroy the instance
Set Score
new score: 1
☑ Relative

sion>OBJ_Food. Along the right side of the window, choose the main1 tab and click and drag in the Destroy Instance icon (looks like a recycle bin). When the new dialogue window open, click other so that the destruction applies to the food and to pacman. Then click ok to close the dialogue box. The score still needs to change, so click on the score tab along the right and drag in the Set Score icon (first square with three yellow circles in it). On the new dialogue box, enter 1 as the new score and be sure to check the Relative box so that it adds 1 each time and doesn't just reset the score to 1 and click ok. The program also needs to check the score to see if it is 25, so with the score tab still open, drag the Test Score icon (looks like a stop sign with three yellow circles in it) into the actions area. In the new dialogue box, set the value to 25 and operation as equal to and then click ok.

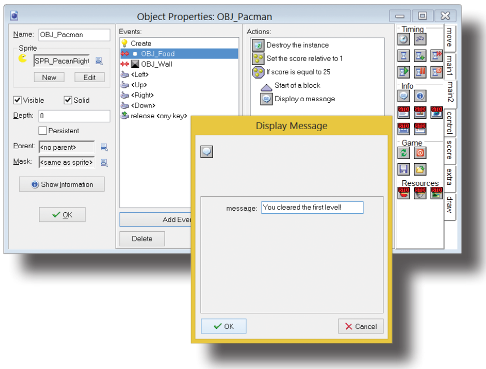

Object Properties: OBJ_Pacman

Name: OBJ_Pacman
Sprite: SPR_PacanRight
☑ Visible ☑ Solid
Depth: 0
☐ Persistent
Parent: <no parent>
Mask: <same as sprite>

Events: Create; OBJ_Food; OBJ_Wall; <Left>; <Up>; <Right>; <Down>; release <any key>

Actions:
Destroy the instance
Set the score relative to 1
If score is equal to 25
Start of a block

34. After the If Statement that was just added, there needs to be an action for if the score is 25. This set of actions will need to be grouped in a block so they are performed together if the score is 25. Start this set of actions by clicking on the control tab along the right side of the properties window. Drag the Start Block icon (looks like a triangle pointing up) into the actions area directly below Test Score. Now click the main2 tab and drag Display Message into the actions area. Type some sort of message telling the player that they have won into the new dialogue box and click ok to close it. Next click on the main1 tab along the right side of the properties window and drag in the Next Room icon (has a green arrow point to the right in front of a blue and white box). Choose an effect for how the new room will appear and select ok. To end this sequence, click on the control tab and drag in the End Block (triangle pointing down). The final thing that has to be added to the collision with food is sound. The sound has already been loaded, so it just has to be programmed to play when the two collide. Click on the main 1 tab and drag the Play Sound (speaker inside a square) icon into the actions area. When the new dialogue box appears, change the sound to SND_FoodColli-sion. Leave loop false and click ok. That should complete the food collision. Click ok to close the OBJ_Pacman properties box.

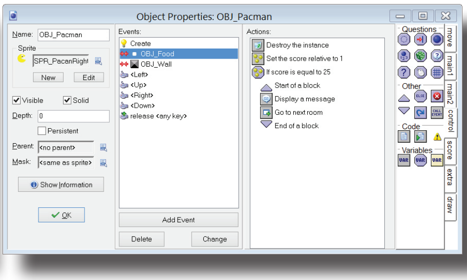

Object Properties: OBJ_Pacman

Name: OBJ_Pacman
Sprite: SPR_PacanRight
☑ Visible ☑ Solid
Depth: 0
☐ Persistent
Parent: <no parent>
Mask: <same as sprite>

Events: Create; OBJ_Food; OBJ_Wall; <Left>; <Up>; <Right>; <Down>; release <any key>

Actions:
Destroy the instance
Set the score relative to 1
If score is equal to 25
Start of a block
Display a message

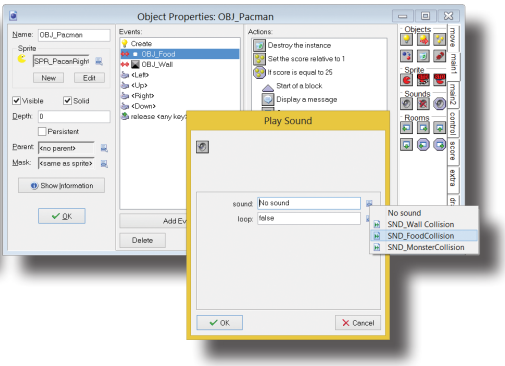
Display Message
message: You cleared the first level!


Object Properties: OBJ_Pacman

Name: OBJ_Pacman
Sprite: SPR_PacanRight
☑ Visible ☑ Solid
Depth: 0
☐ Persistent
Parent: <no parent>
Mask: <same as sprite>

Events: Create; OBJ_Food; OBJ_Wall; <Left>; <Up>; <Right>; <Down>; release <any key>

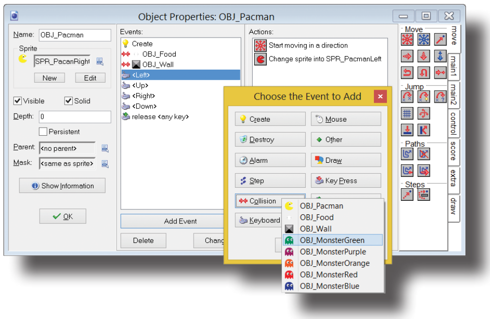
Actions:
Destroy the instance
Set the score relative to 1
If score is equal to 25
Start of a block
Display a message
Go to next room
End of a block


Object Properties: OBJ_Pacman

Actions:
Destroy the instance
Set the score relative to 1
If score is equal to 25
Start of a block
Display a message

Play Sound
sound: No sound
loop: false

No sound
SND_Wall Collision
SND_FoodCollision
SND_MonsterCollision


Object Properties: OBJ_Pacman

Name: OBJ_Pacman
Sprite: SPR_PacanRight
☑ Visible ☑ Solid
Depth: 0
☐ Persistent
Parent: <no parent>
Mask: <same as sprite>

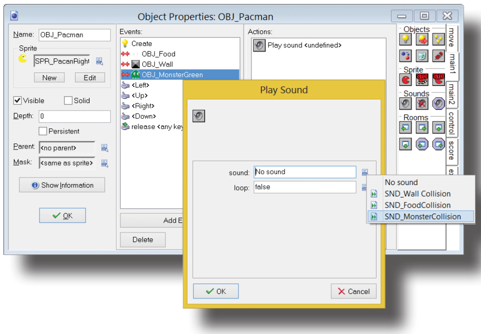
Events: Create; OBJ_Food; OBJ_Wall; <Left>; <Up>; <Right>; <Down>; release <any key>

Actions:
Start moving in a direction
Change sprite into SPR_PacmanLeft

Choose the Event to Add

35. Save the game and click the Run Game in Debug Mode to test the program-ming. When pacman collides with food Is the instance of food destroyed? Does the score increase by 1? Is there a sound that plays when the collision occurs? Hopefully all of these actions are working properly!

36. As seen when testing the game, pacman can earn points but he can't lose


Object Properties: OBJ_Pacman

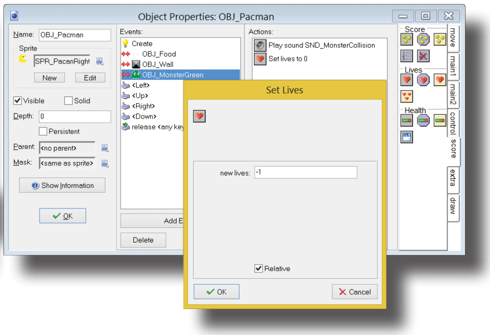
Name: OBJ_Pacman
Sprite: SPR_PacanRight
☑ Visible ☐ Solid
Depth: 0
☐ Persistent
Parent: <no parent>
Mask: <same as sprite>

Events: Create; OBJ_Food; OBJ_Wall; OBJ_MonsterGreen; <Left>; <Up>; <Right>; <Down>; release <any key>

Actions:
Play sound <undefined>

Play Sound
sound: No sound
loop: false

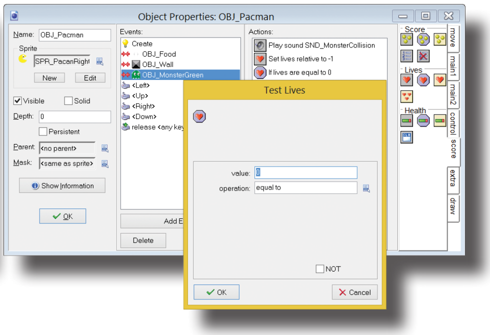
No sound
SND_Wall Collision
SND_FoodCollision
SND_MonsterCollision


Object Properties: OBJ_Pacman

Events: Create; OBJ_Food; OBJ_Wall; OBJ_MonsterGreen; <Left>; <Up>; <Right>; <Down>; release <any key>

Actions:
Play sound SND_MonsterCollision
Set lives to 0

Set Lives
new lives: -1
☑ Relative

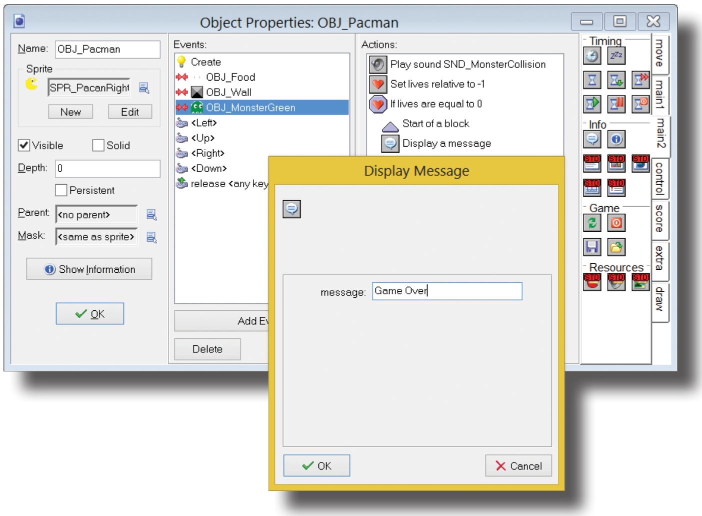

Object Properties: OBJ_Pacman

Events: Create; OBJ_Food; OBJ_Wall; OBJ_MonsterGreen; <Left>; <Up>; <Right>; <Down>; release <any key>

Actions:
Play sound SND_MonsterCollision
Set lives relative to -1
If lives are equal to 0

Test Lives
value: 0
operation: equal to
☐ NOT

lives from colliding with any of the monsters. This will be the next step. Double click on OBJ_Pacman to open the objects properties. Choose Add Event>Collison>OBJ_MonsterGreen. Click the main1 tab along the right side of the properties window and drag in the Play Sound icon. On the new dialogue box, change the sound to SND_MonsterCollision, leave the loop on false and click ok. Now choose the score tab and drag the Set lives (red heart inside a square) icon into the actions area. the new lives should be -1 and relative should be checked so that each time one life is lost. Click ok to close the dialogue box. The number of lives needs to be checked to make sure the game hasn't run out of lives yet, so drag in the Test Lives (red heart inside a so sign) icon into the actions area. On the new dialogue box, the value should be set to zero and the operation as equal to. Once these are set, click ok to close the dialogue box.



Object Properties: OBJ_Pacman

Name: OBJ_Pacman
Sprite: SPR_PacanRight
☑ Visible □ Solid
Depth: 0
□ Persistent
Parent: <no parent>
Mask: <same as sprite>

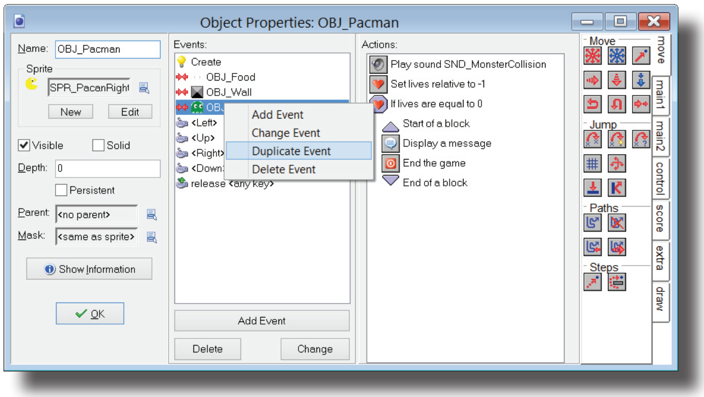
Events: Create, OBJ_Food, OBJ_Wall, OBJ_MonsterGreen, <Left>, <Up>, <Right>, <Down>, release <any key>

Actions:
Play sound SND_MonsterCollision
Set lives relative to -1
If lives are equal to 0
Start of a block
Display a message

Display Message

message: Game Over

37. Just like the score earlier, a group of actions will need to be performed after running the test. If the lives equal zero, a message will have to appear to tell the player they have lost and the game will have to restart. These action will have to be placed in a block. Click on the control tab along the right and drag the Start Block icon into the actions area. Now click

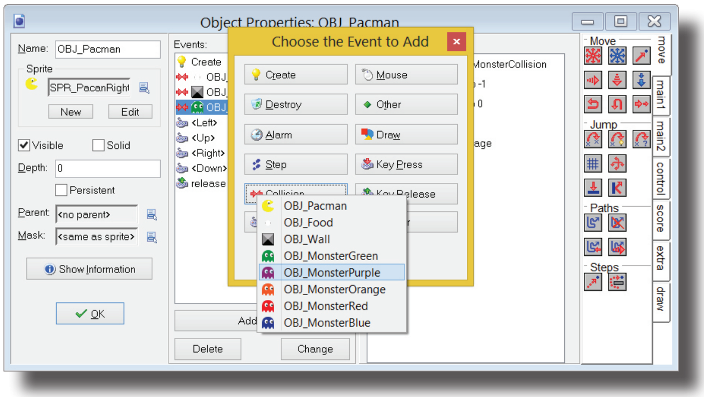


Object Properties: OBJ_Pacman

Name: OBJ_Pacman
Sprite: SPR_PacanRight
☑ Visible □ Solid
Depth: 0
□ Persistent
Parent: <no parent>
Mask: <same as sprite>

Events: Create, OBJ_Food, OBJ_Wall, OBJ_MonsterGreen, <Left>, <Up>, <Right>, <Down>, release <any key>

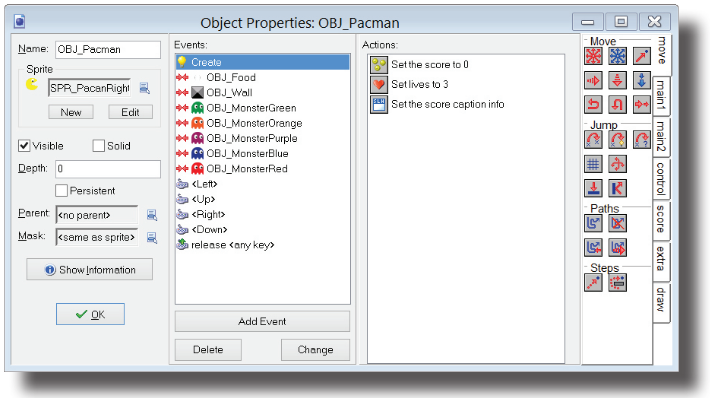
Actions:
Play sound SND_MonsterCollision
Set lives relative to -1
If lives are equal to 0
Start of a block
Display a message
End the game
End of a block



Object Properties: OBJ_Pacman

Choose the Event to Add

on the main2 tab and drag the Display Message icon into the actions area. Type some sort of message to the player to let them know they have lost and then click ok. Drag in the Restart Game (red power button) icon into the actions area. That is it for this block, so click on the control tab and add the End Block icon to the actions area to end the block. That ends the collision with the green monster.

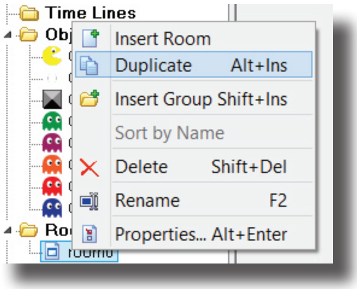


Object Properties: OBJ_Pacman

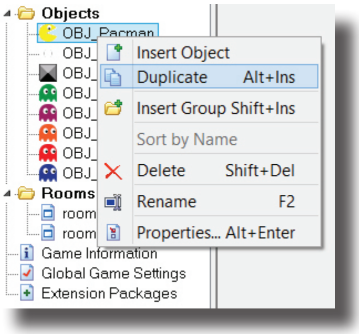
Name: OBJ_Pacman
Sprite: SPR_PacanRight
☑ Visible □ Solid
Depth: 0
□ Persistent
Parent: <no parent>
Mask: <same as sprite>

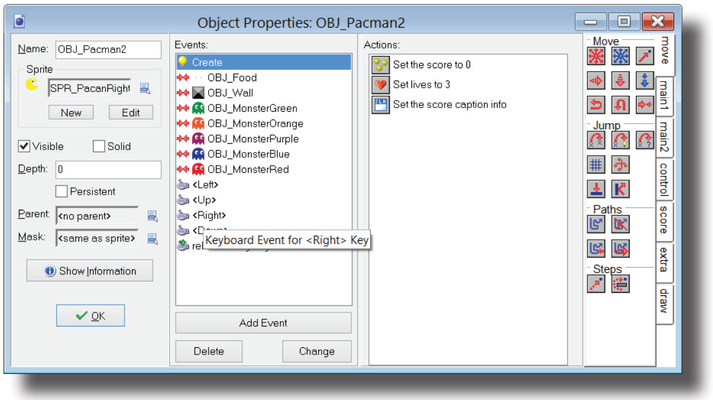
Events: Create, OBJ_Food, OBJ_Wall, OBJ_MonsterGreen, OBJ_MonsterOrange, OBJ_MonsterPurple, OBJ_MonsterBlue, OBJ_MonsterRed, <Left>, <Up>, <Right>, <Down>, release <any key>

Actions:
Set the score to 0
Set lives to 3
Set the score caption info



38. Rather than programming this collision event for each of the other four monster it would be easier just to duplicate this event and add it to each monster. How? With the OBJ_Pacman properties window still open, right click on the Collision event with object OBJ_MonsterGreen event and choose duplicate. Click Collision on the Choose Event to Add dialogue box and then as the new pull down menu appears, select OBJ_Monster-





Object Properties: OBJ_Pacman2

Name: OBJ_Pacman2
Sprite: SPR_PacanRight
☑ Visible □ Solid
Depth: 0
□ Persistent
Parent: <no parent>
Mask: <same as sprite>

Events: Create, OBJ_Food, OBJ_Wall, OBJ_MonsterGreen, OBJ_MonsterOrange, OBJ_MonsterPurple, OBJ_MonsterBlue, OBJ_MonsterRed, <Left>, <Up>, <Right>, <Down>, release

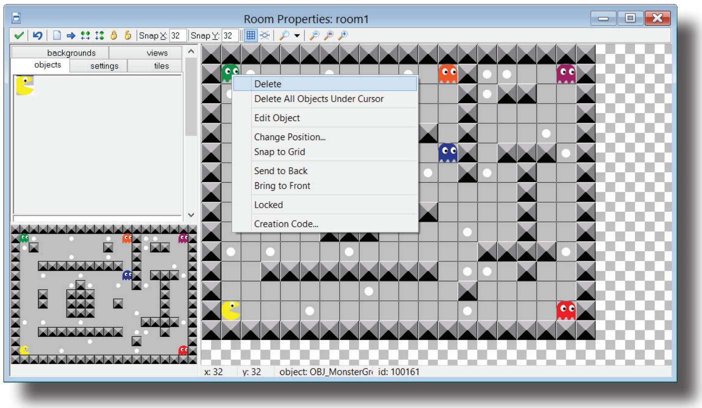
Keyboard Event for <Right> Key

Actions:
Set the score to 0
Set lives to 3
Set the score caption info

Purple. This added all of the actions from the collision with the green monster to a new event on the purple monster... it saved quite a bit of work!

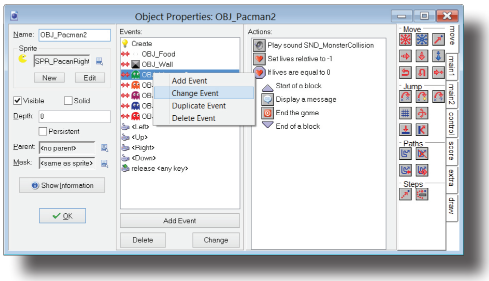
39. Repeat step 38 for OBJ_MonsterOrange, OBJ_MonsterRed, and OBJ_MonsterBlue. When finished, click ok to close the OBJ_Pacman properties.

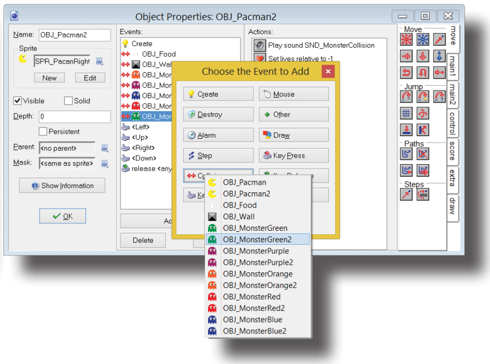
40. Save the game and run it in debug mode to see if the collisions between pacman and the monsters work.

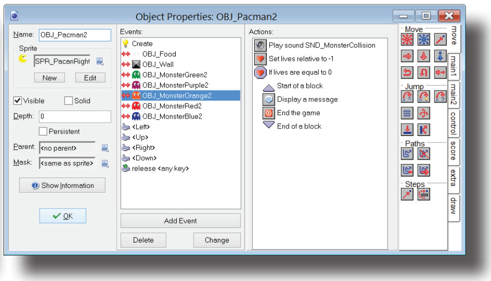


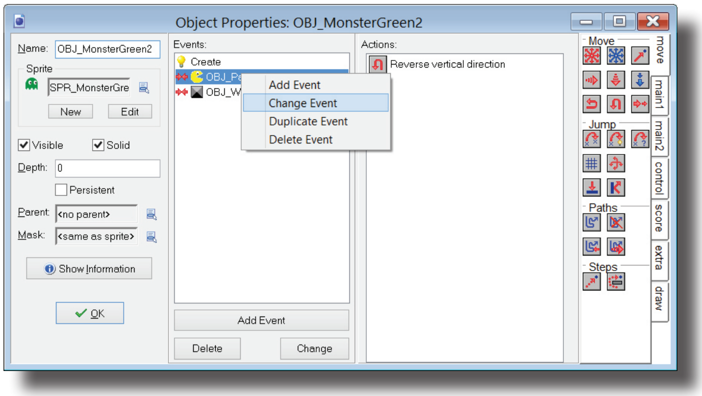
41. Almost there... the game plays! Lives can be lost, points can be earned when food is eaten, but there is not a second level to advance to. This is the final addition to the game. In the rooms folder in the resource tree at the left, right click on room0. Choose duplicate to copy the room.

42. Next all of the objects from room 1 have to be duplicated. Right click on OBJ_Pacman in the resource tree and select duplicate. Rename the object OBJ_Pacman2and then click ok. Do this for each of the monsters as well. When renaming them, just add 2 (since they are in room 2) to the name.

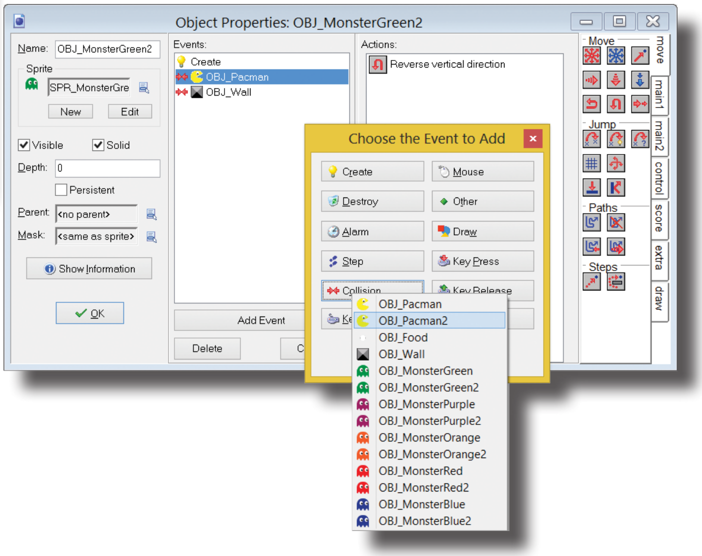


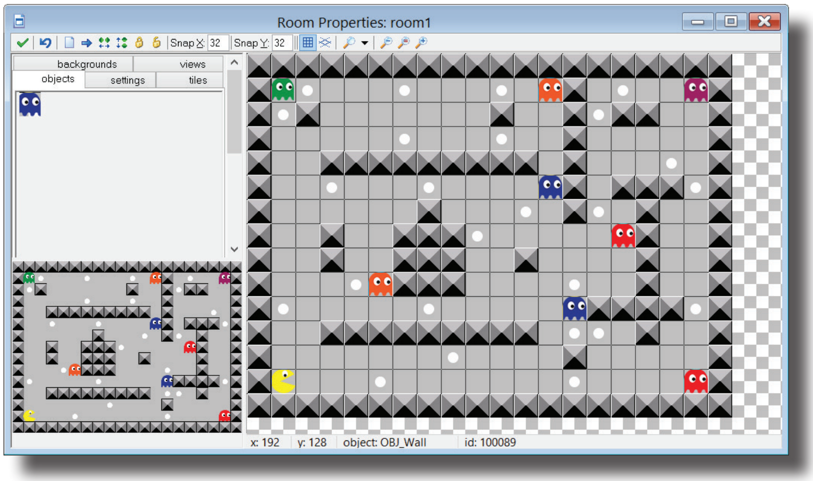






43. Remove pacman and each monster from room 1 by right clicking on each of the objects and choosing delete. Replace the objects with their duplicate (OBJ_MonsterGreen replaced by OBJ_MonsterGreen2). When finished, click on the green check on in the top left corner to close

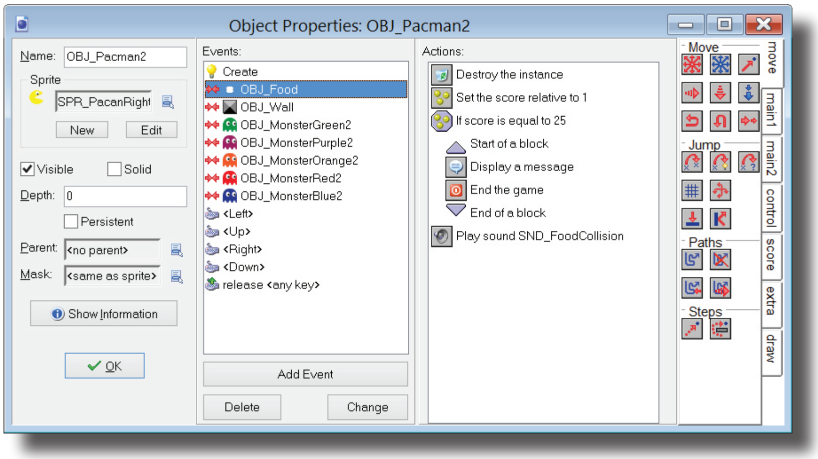




he room properties.

44. The collision events will have to be edited too because they reference objects in room0. Double click on OBJ_Pacman2 in the resource tree. Right click on Collision Event with OBJ_MonsterGreen and select Change Event. Choose Collision Event and OBJ_MonsterGreen2. Do the same thing for the other four monsters collision events with pacman. When finished, click ok to close the OBJ_Pacman properties window.

45. Now the collision events for the monster have to be changed to match the new pacman. Double click on OBJ_MonsterGreen2. Right click on the Collision Event with OBJ_Pacman and select Change Event. Choose Collision as the type of event and then from the popup menu, choose OBJ_Pacman2. Click ok to close the properties window. Repeat this project for the other 4 monsters.


Object Properties: OBJ_Pacman2

Name: OBJ_Pacman2

Sprite: SPR_PacanRight

☑ Visible ☐ Solid

Depth: 0

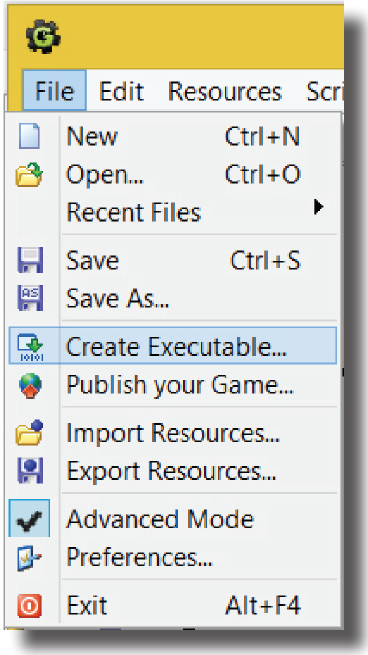
☐ Persistent

Parent: <no parent>

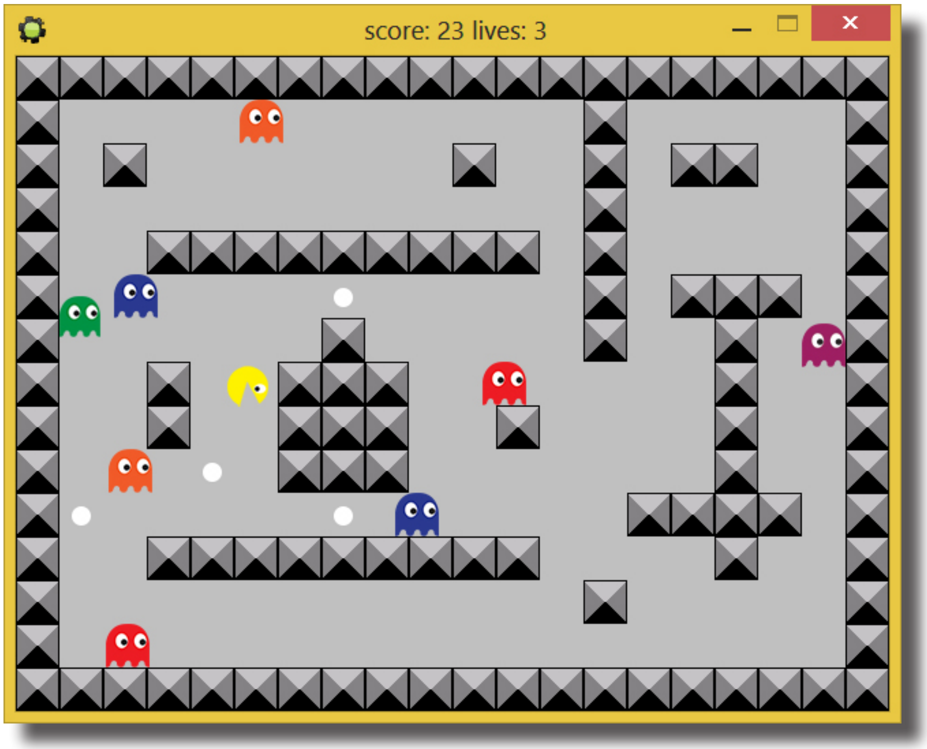
Mask: <same as sprite>

Events:
- Create
- OBJ_Food
- OBJ_Wall
- OBJ_MonsterGreen2
- OBJ_MonsterPurple2
- OBJ_MonsterOrange2
- OBJ_MonsterRed2
- OBJ_MonsterBlue2
- <Left>
- <Up>
- <Right>
- <Down>
- release <any key>

Actions:
- Destroy the instance
- Set the score relative to 1
- If score is equal to 25
- Start of a block
- Display a message
- End the game
- End of a block
- Play sound SND_FoodCollision



46. Now change the room a little bit to make it more difficult than the first room. Take away some food so there are less instances of food to collect. Place food in more difficult to reach locations. Add more monsters to the room. Change the layout of the walls to make it harder to navigate. Make the monsters move faster. Just make level 2 more challenging. In the example here, the blue and red monster are faster in room1, there is less food available to eat, and there are a few additional monster in the room.



47. Save the game. Click the Run Game in Debug Mode icon and play all the way through the game. Does it work? Is the second room challenging enough?

48. When playing the game through, upon

beating the second room, GameMaker encounters a fatal error. There isn't a message telling the player the game is over and it may even try to go to another level (but the level doesn't exist).

49. Double click on OBJ_Pacman2. Click on Collision Event with OBJ_Food. Right click on the Next Room icon and choose delete. From the main2 tab, drag in the End Game icon to replace the Next Room icon. Double click on the Display Message icon and change the player to a message that tells the player they have won. Click ok to close the properties window.

50. Save the game and run it again in debug mode to test how it works. Are there any issues? If not, choose File>Create Executable. Name the file and click save. It is now playable on any computer as an .exe file. That's it! The game is made and exported to be played, so play a bit and enjoy it!