

FIND THE GOLD: PART 2

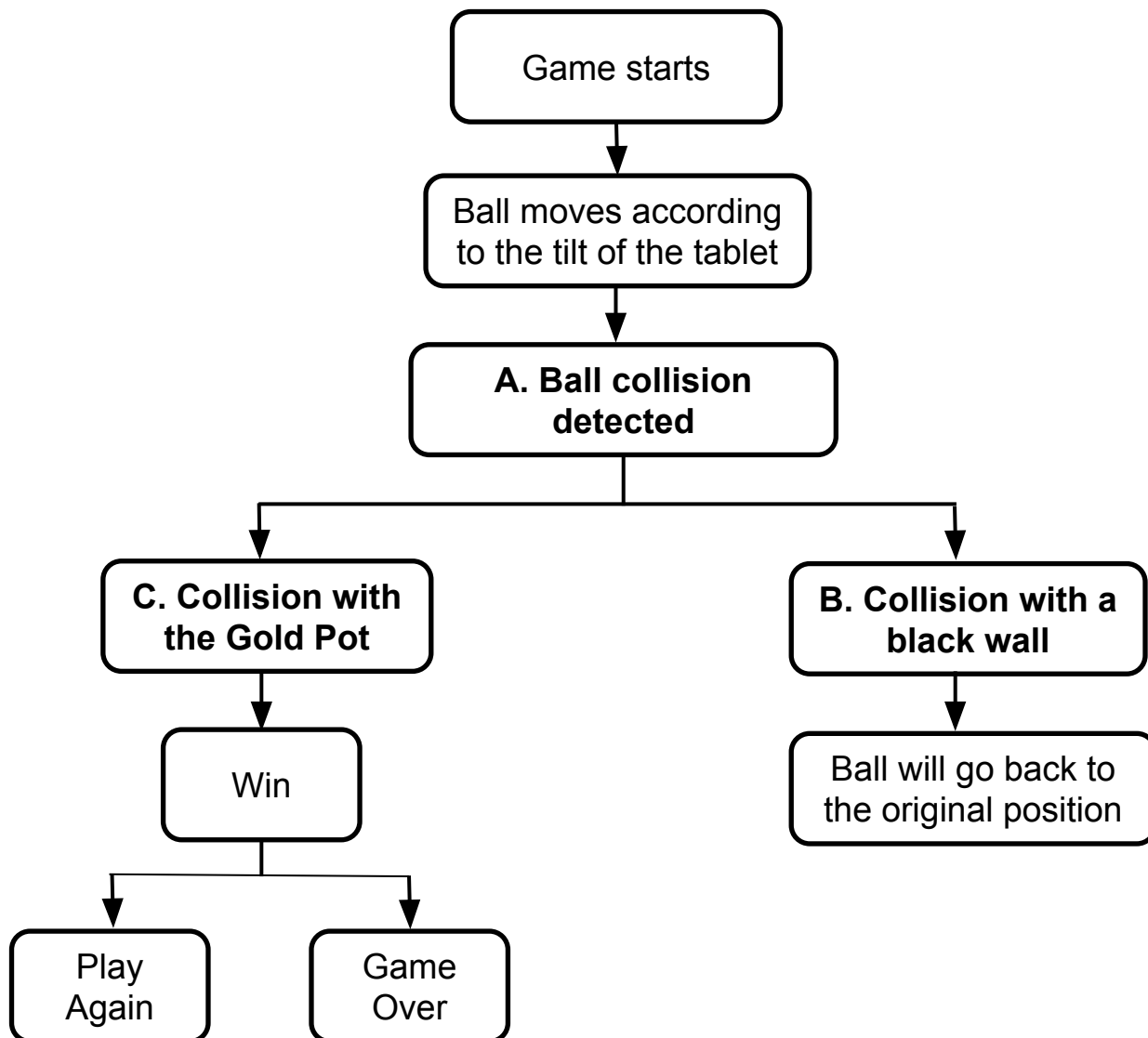


You will now
add Ball movement to your
maze game so the Ball
moves as you tilt the
mobile device

REVIEW

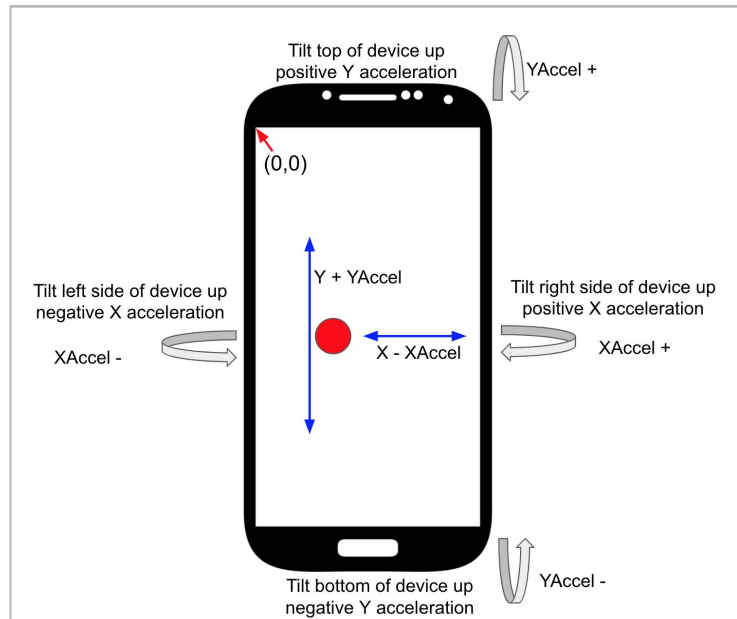
1

Review the diagrams below with your partner. Check that you understand the sequence of steps for the Find the Gold app below.



ACCELEROMETER

The **Ball** will move as the user tilts the mobile device. Use the **AccelerometerSensor** to measure the tilt in the X and Y direction to update the **Ball**'s position.



2

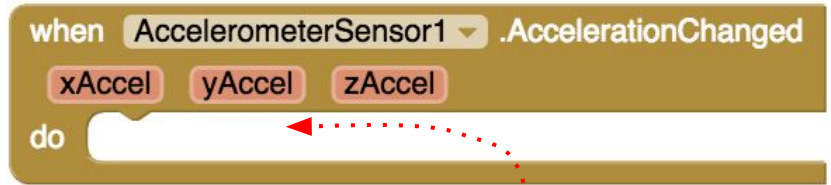
Drag out an **Accelerometer1.AccelerationChanged** block.

The screenshot shows the MIT App Inventor interface. In the 'Blocks' panel on the left, the 'AccelerometerSensor1' block is circled in red. In the 'Viewer' panel on the right, the 'when AccelerometerSensor1 .AccelerationChanged' block is circled in red. A red arrow points from the circled block in the Viewer to a larger, detailed view of the block below.

```

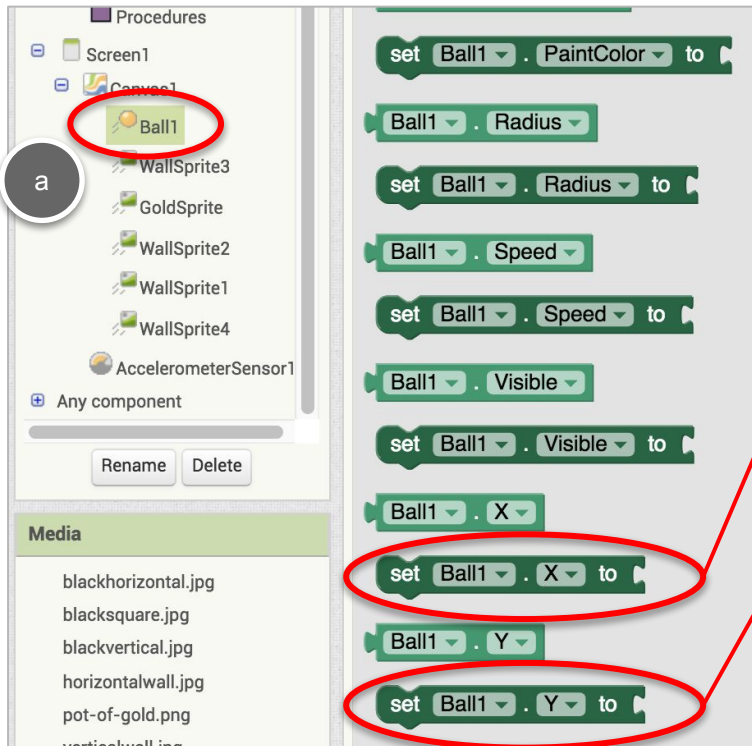
when AccelerometerSensor1 .AccelerationChanged
  xAccel yAccel zAccel
do
  
```

MOVING THE BALL



3

Drag out a **set Ball1.X** block and a **set Ball1.Y** block and snap both into the **Accelerometer1.AccelerationChanged** block.

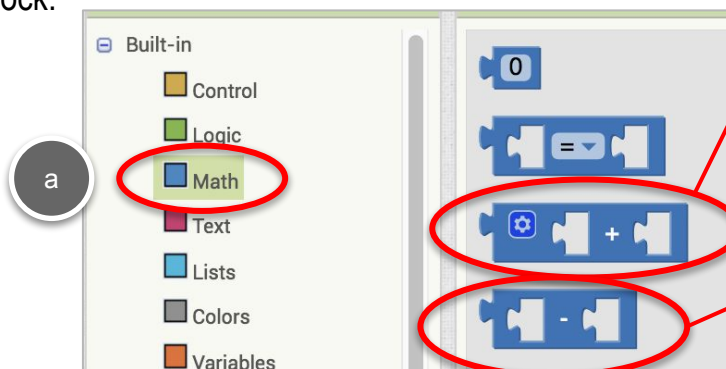


b

c

4

From the Math drawer, drag out a minus (-) block and snap to the **set Ball1.X**. Drag a plus (+) block and snap to the **set Ball1.Y** block.



a

c

b

UPDATE BALL X,Y COORDINATES

- 5 From the **Ball1** drawer, drag out **Ball1.X** and snap to the left side of the minus (-) block. Snap **Ball1.Y** to the left side of the plus (+) block.

Update the X and Y positions by adding to or subtracting from the current values

The screenshot shows the App Inventor interface. On the left, the 'Procedures' pane shows 'Screen1' with 'Canvas1' containing 'Ball1'. A red circle 'a' highlights 'Ball1' in the drawer. On the right, the 'when AccelerometerSensor1 .AccelerationChanged' event handler is shown. It contains two 'do' blocks: 'set Ball1 . X to' followed by a minus (-) block, and 'set Ball1 . Y to' followed by a plus (+) block. Red circles 'b' and 'c' highlight the 'Ball1 . X' and 'Ball1 . Y' blocks being added to the minus and plus blocks respectively.

- 6 Hover over the **xAccel** and **yAccel** input parameters. Snap **get xAccel** to the minus (-) block, and **get yAccel** to the plus (+) block.

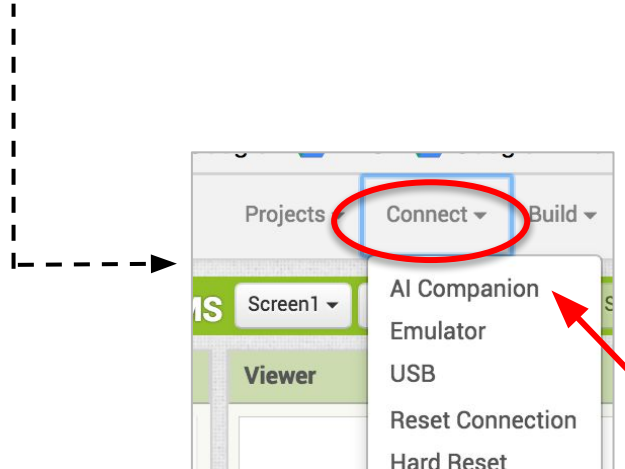
Because XAccel is negative when tilting right, subtract XAccel to make it move according to how you tilt the device.

The screenshot shows the App Inventor interface. The 'when AccelerometerSensor1 .AccelerationChanged' event handler is shown. It contains two 'do' blocks: 'set Ball1 . X to' followed by a minus (-) block, and 'set Ball1 . Y to' followed by a plus (+) block. A red circle highlights the 'xAccel' input parameter in the event handler. Below the event handler, a 'get xAccel' block is shown. A red arrow points from the 'get xAccel' block to the minus (-) block. Another red circle highlights the 'yAccel' input parameter in the event handler. Below the event handler, a 'get yAccel' block is shown. A red arrow points from the 'get yAccel' block to the plus (+) block.

TESTING!

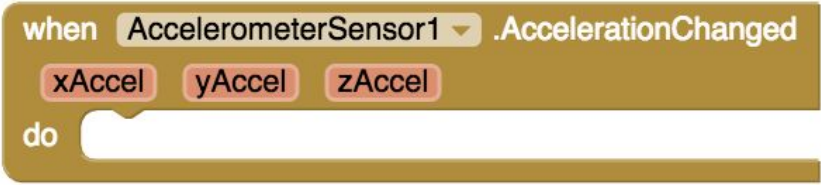


Test the app with the MIT AI2 Companion. Your ball should move according to how you tilt the device!



COMPUTATIONAL THINKING CONCEPTS

The following are the Computational Thinking Concepts learned in Part 2.

Find The Gold	
1. Events:	
2. Operators:	