ANGRY BIRDS – SCRATCH CODE

Build a game similar to Angry Birds.

Create 3 sprites:
1. A Bird
2. A Launcher (or catapult)
3. A Target (something to aim the bird at)

Add the script on this page to the bird. The code will make the bird fly in an arc.

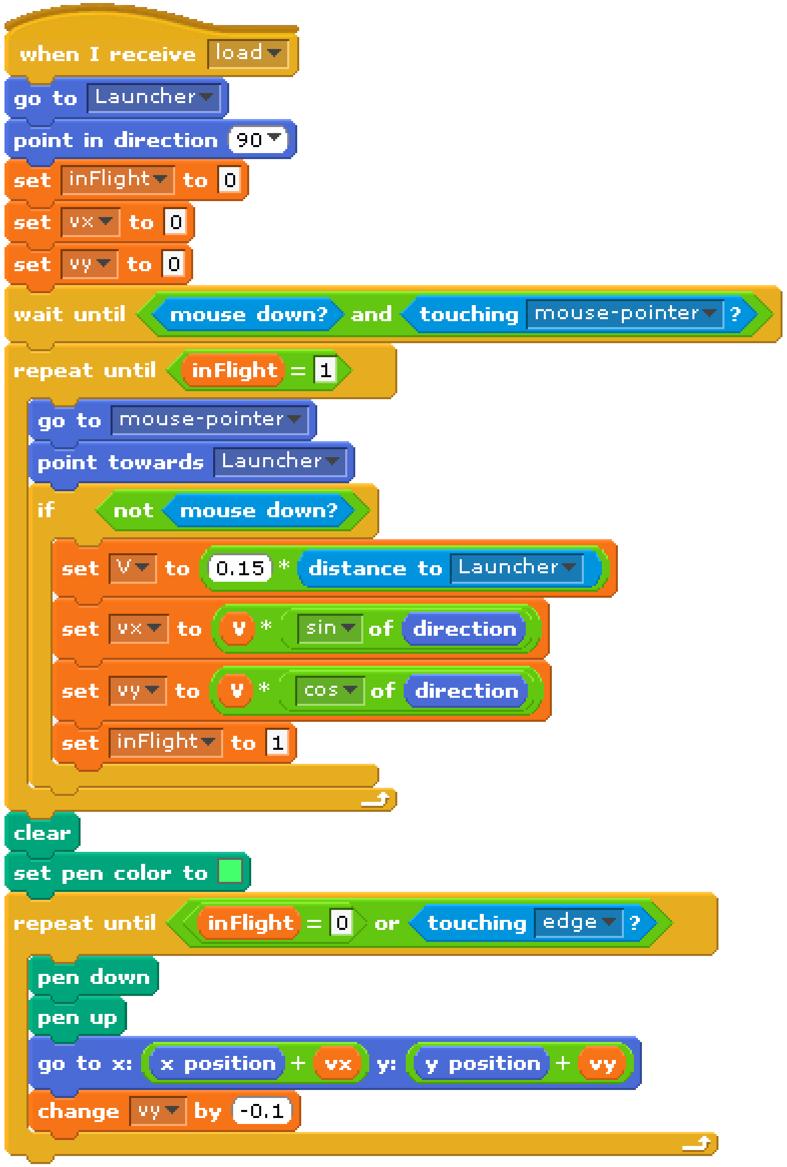
Test the game.

Does the bird always launch?

If not, is it because the bird has touched the edge of the screen before launching?

Can you improve the bird's flying motion?

What else do you want to add to the game?



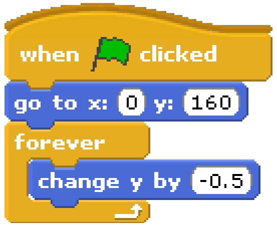
```
when I receive [load]
go to [Launcher]
point in direction (90)
set [inFlight] to 0
set [vx] to 0
set [vy] to 0
wait until <<mouse down?> and <touching [mouse-pointer] ?>>
repeat until <(inFlight) = 1>
    go to [mouse-pointer]
    point towards [Launcher]
    if <not <mouse down?>>
        set [V] to ((0.15) * (distance to [Launcher]))
        set [vx] to ((v) * ([sin] of (direction)))
        set [vy] to ((v) * ([cos] of (direction)))
        set [inFlight] to 1
clear
set pen color to 🟩
repeat until <<(inFlight) = 0> or <touching [edge] ?>>
    pen down
    pen up
    go to x: ((x position) + (vx)) y: ((y position) + (vy))
    change [vy] by -0.1
```

CoderDojo Malahide 2014 – Scratch Hackers group

ANGRY BIRDS – HOW IT WORKS

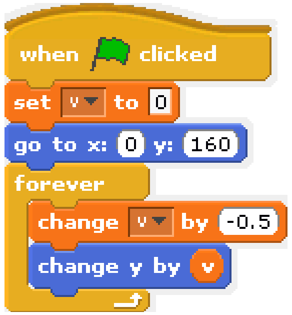
SPRITE FALLING DOWN

What happens when something falls? In Scratch, we would say that a sprite's y-position decreases when it falls. Try this code. Does the falling sprite look realistic?



```
when ⚑ clicked
go to x: 0 y: 160
forever
    change y by -0.5
```

Does the sprite seem to be falling too slowly? This is because it is falling at a constant rate. In real life, falling objects move faster and faster downwards. Try this code instead:



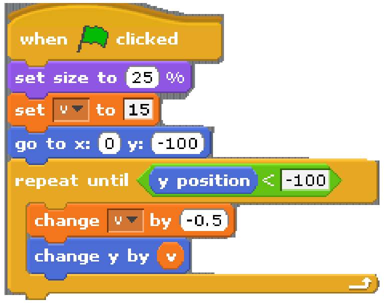
```
when ⚑ clicked
set v to 0
go to x: 0 y: 160
forever
    change v by -0.5
    change y by v
```

Why do we change v by minus 0.5 when we had said we wanted to increase the speed? The minus sign shows that the direction of the change is downwards.

This is all the code we need for a falling object.

SPRITE THROWN STRAIGHT UP IN THE AIR

What happens when we throw something straight up in the air? The harder we throw, the higher it will go, but will eventually fall back down. Try different starting values of v in this code to throw the sprite higher or lower.



```
when ⚑ clicked
set size to 25 %
set v to 15
go to x: 0 y: -100
repeat until y position < -100
    change v by -0.5
    change y by v
```

SPRITE THROWN UP AND SIDEWAYS

When we throw a basketball, or water the garden with a hose, we aim up and towards a particular direction. (If you aim the ball or the garden hose straight up you will get a bump on the head or get wet!)

So let's think about throwing something (let's call it a projectile) UP and SIDEWAYS. The projectile will want to keep moving in the direction you have thrown it, but gravity will make it fall down. So the projectile will move in a curved path.

FORMULA FOR PROJECTILE PATH

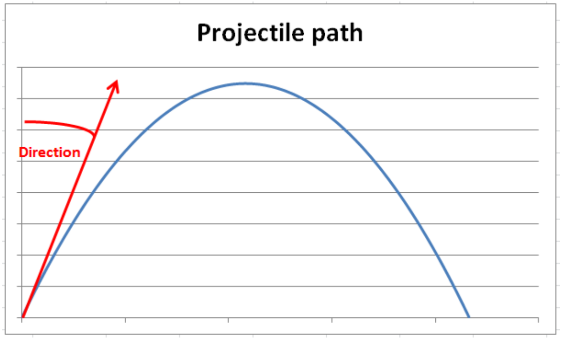
A formula is a rule for how things work, that uses maths symbols.

We will use two maths symbols that you might not have seen before – these are **sin** and **cos**. Sin and cos are very handy for finding the x and y positions when you know the direction. So here is our formula for how the projectile moves. V is the starting speed.

$$change\ in\ x = V \times \sin(Direction)$$

$$change\ in\ y = V \times \cos(Direction) - gravity \times time$$

In our Scratch code, the $change\ in\ x$ is easy to program, but the $change\ in\ y$ is tricky. We have to program the $gravity \times time$ by reducing the $change\ in\ y$ by 0.1 at the end of the block of code.



Now try the code on the other side of this page and make your own Angry Birds game.

CoderDojo Malahide 2014 – Scratch Hackers group