

The background of the slide is black, covered with a pattern of white-outlined triangles of various sizes and orientations, scattered across the entire area.

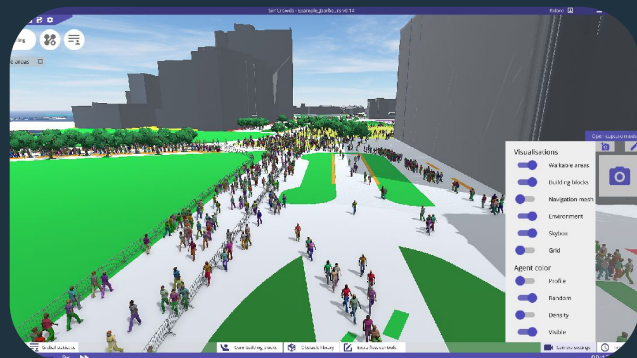
DGP

Boids Project

Harshpreet Singh
Tahseen Rayhan
Yawen Zhang

What is our Project?

- Crowd Simulation
- Simulates Birds
- All moving, accelerating,



What we learned!

- Boid's Principles
- Physics!
- Programming

THE BIG FIVE EQUATIONS

Equation 1: $\Delta x = \bar{v} \times t$

Equation 2: $v = v_0 + at$

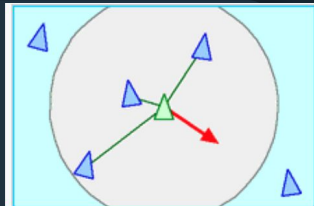
Equation 3: $x = x_0 + v_0 \times t + \frac{1}{2} at^2$

Equation 4: $x = x_0 + v \times t - \frac{1}{2} at^2$

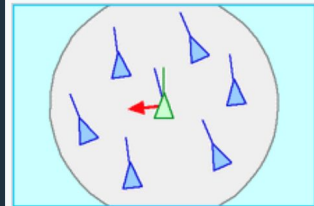
Equation 5: $v^2 = v_0^2 + 2a(x - x_0)$

Study.com

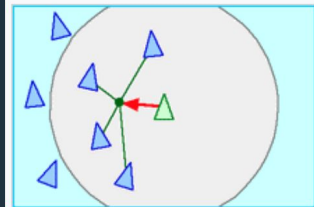
```
import pygame
import tools as tl
import numpy as np
import scipy as sp
import random
from copy import deepcopy
import math
```



Separation



Alignment

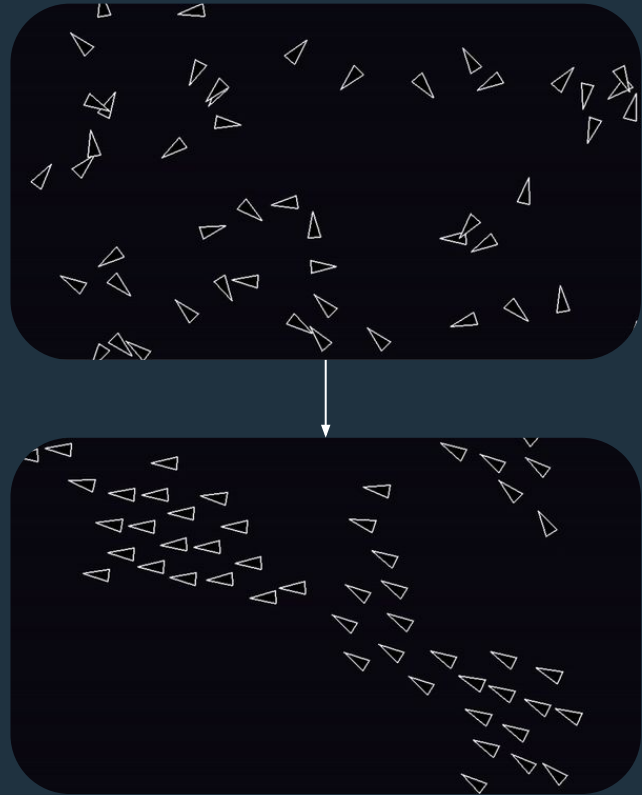


Cohesion

Cr: Craig Reynolds

The 5 Rules We Added

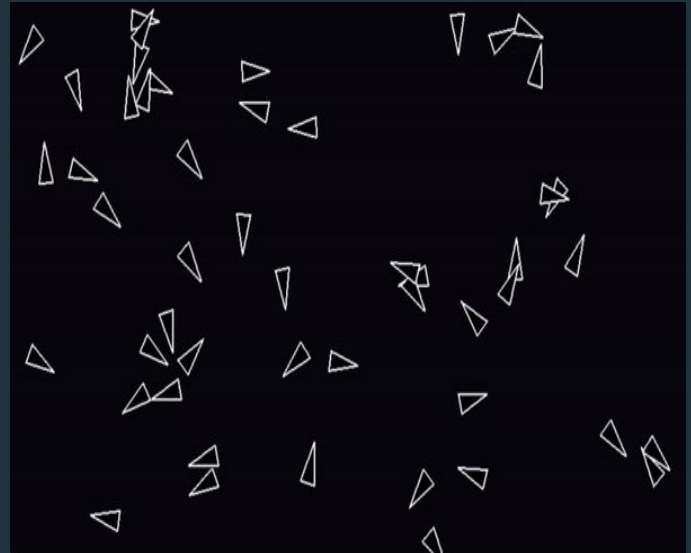
Cohesion, Separation,
Alignment, Random, Mouse Rules



Cohesion

Travel towards the average position of other nearby objects, “flocking” together

1. Look at every bird nearby
2. Take the average position of those birds
3. Update our bird's acceleration



Alignment – fly in same direction!

Purpose:

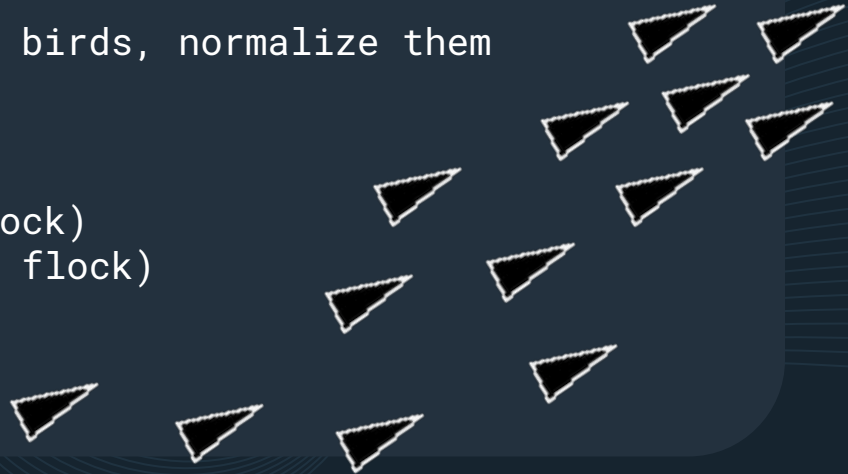
- ↑ efficiency and ↑ protection.
- Migration

Process:

1. Loop through all “nearby” birds
2. Find average velocity of all the birds, normalize them

Playing around:

- Default “nearby” radius = 70
- ↑ radius = widely spread (big flock)
- ↓ radius = tightly packed (small flock)



Separation – Create space between birds!

- Creates space between birds
- Ensured no overlapping birds
- Created a function that was being called constantly
- Would move the birds in the opposite direction



Mouse – interactive element!

Purpose:

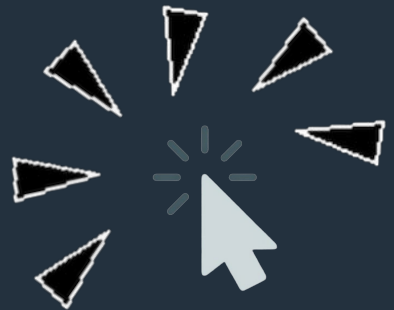
- ↑ application.
 - Mouse = predators → run away
 - Mouse = preys → approach

Process:

1. Built-in functions: `mouseDown` & `mousePos`
2. mouse position - current position

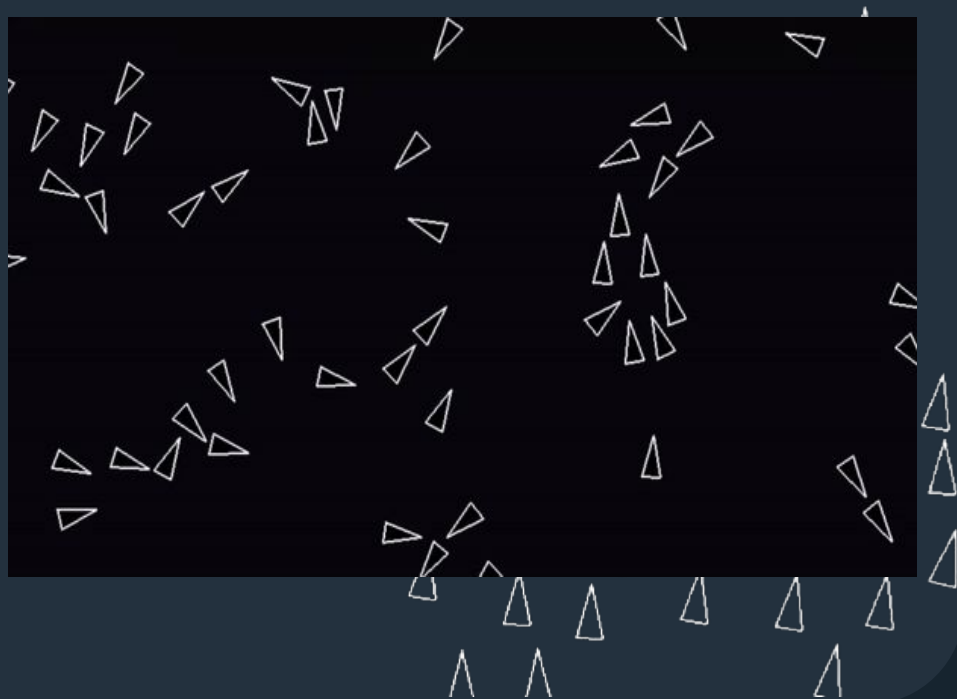
Playing around:

- Default weight: 0.2
- ↑ weight = ↑ effect
- ↓ weight = ↓ effect



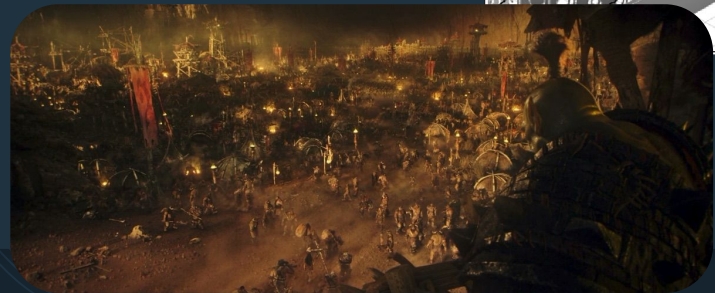
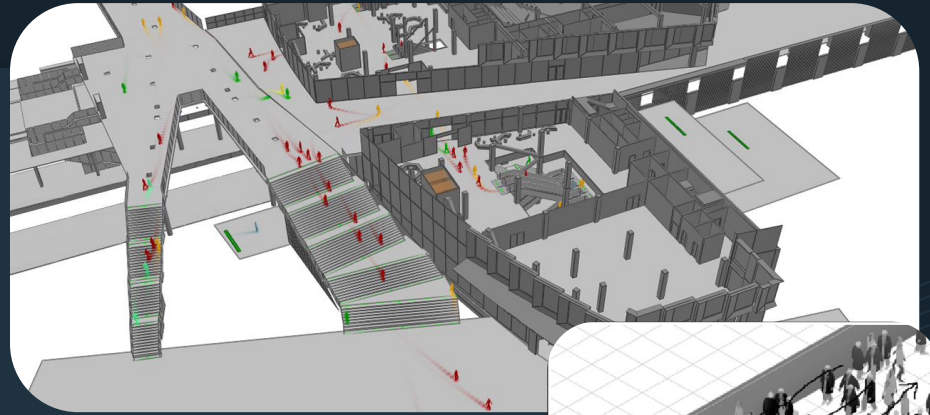
Random

Objects deviate from the flock, simulating having their own “mind”



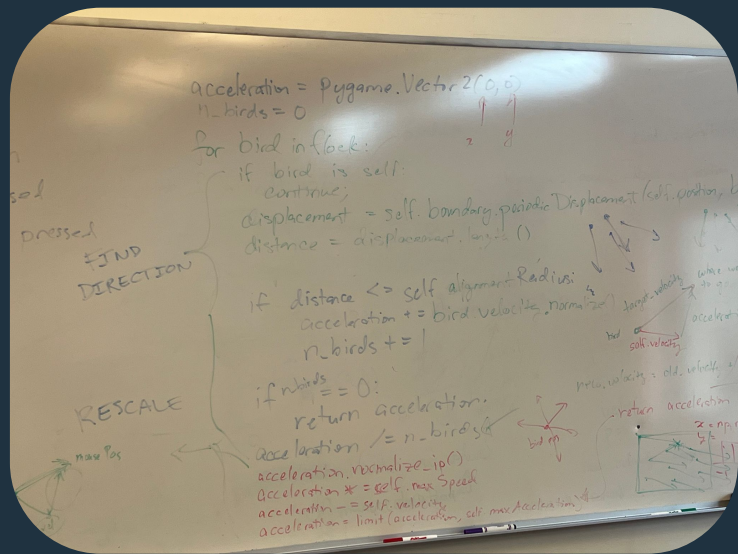
Applications

- Entertainment
- Urban Planning
- Crisis Training



Shortcomings

- Idea Planning
- Familiarity with Pygame, NumPy
- Code Readability



Future Work



Predator/Prey

Obstacles

Boids in realistic environments



THANK YOU!!!

Boids Project