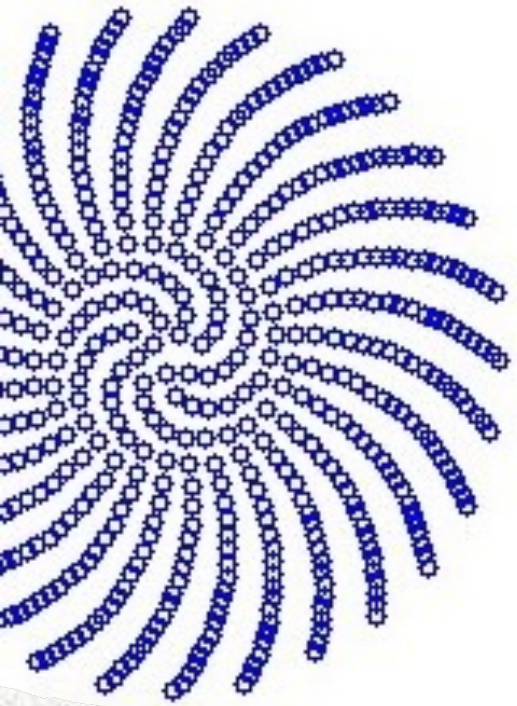
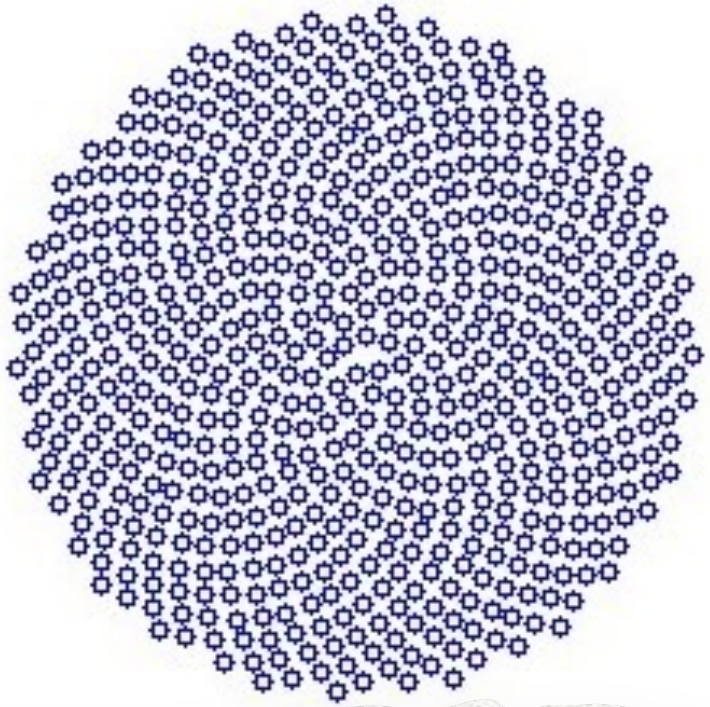
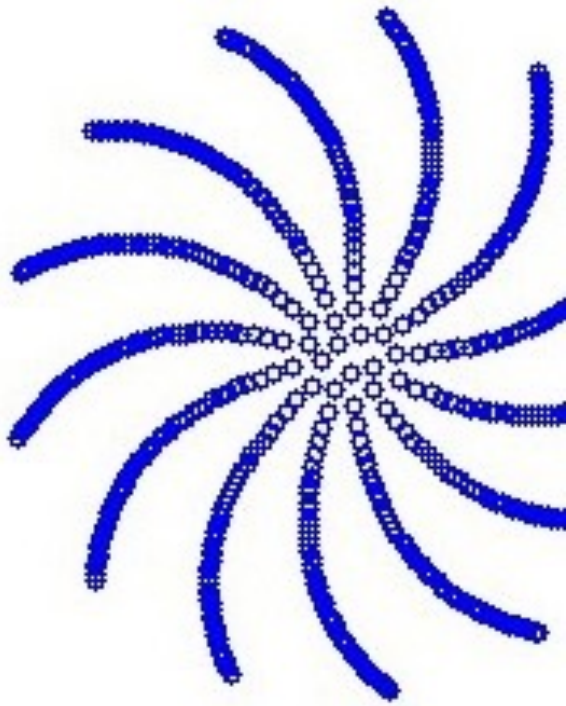
Fibonacci Flower

Adding and Manipulating Graphic Elements during Run-time

Topics

- Quick Review of Adding Graphics during runtime
- Fibonnaci Flower App
- Build as (MVC) Model View Controller

Fibonacci Flower Animation

Fibonacci Sequence

1, 1 , 3, 5, 8, 13, 21, 34, 55, 89, 144, etc.

Golden Ratio

0.61818182.

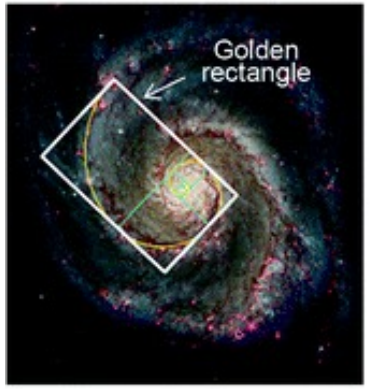
1, 1 , 3, 5, 8, 13, 21, 34, 55, 89, 144, etc.

Golden Ratio is the ratio of successive terms, when the terms become very large.

Golden Ratio in Nature

Golden Ratio is considered to be extraordinary because of its common occurrence in nature and art.

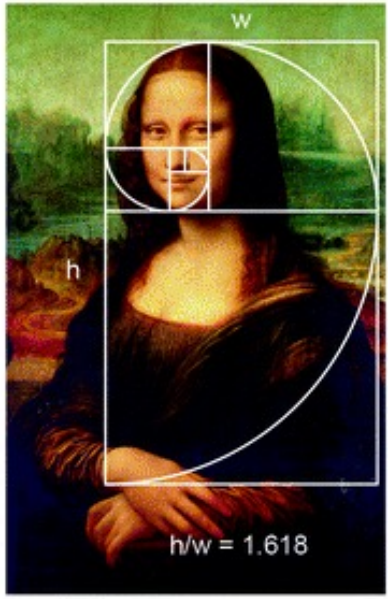
Galaxy



Sunflower

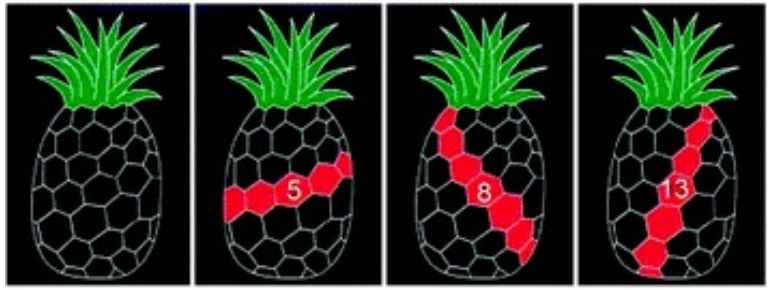
Dahlia flower

Mona Lisa



Sea shell

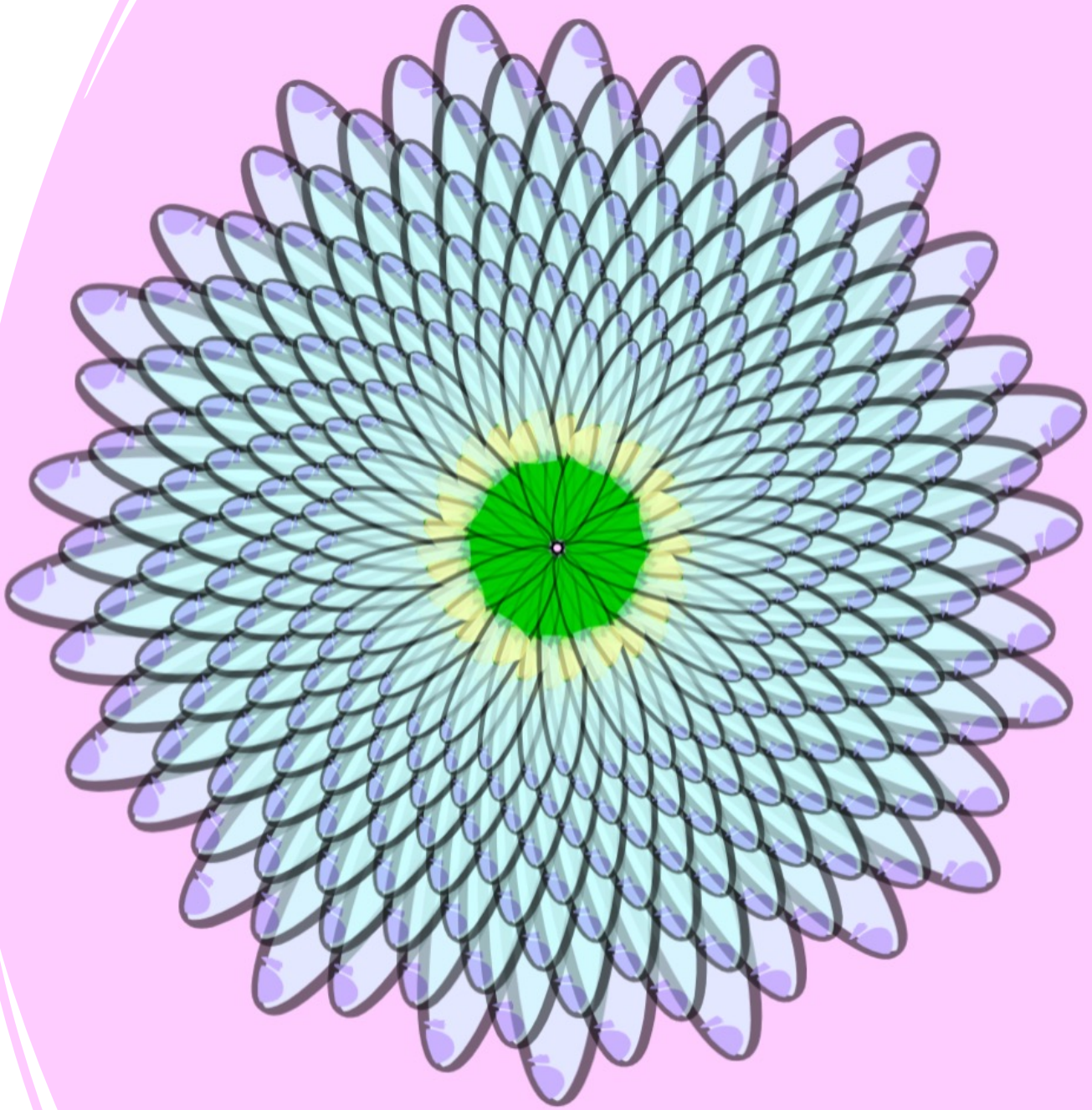
Pineapple



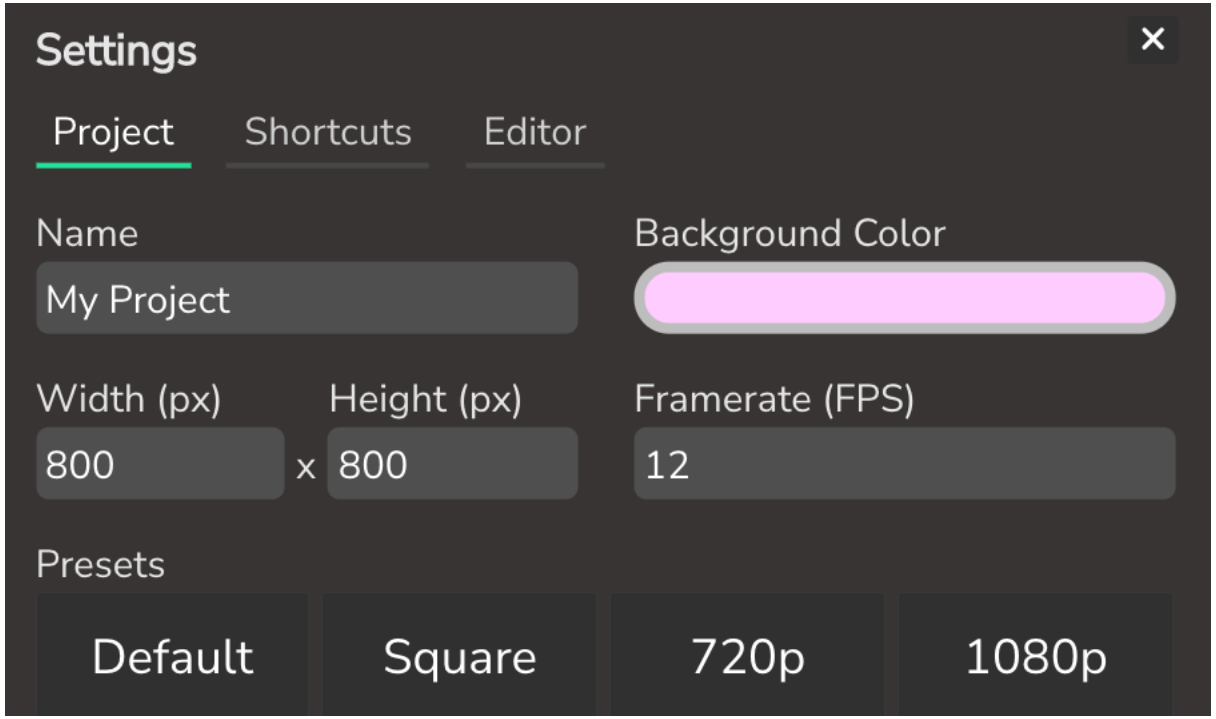
Fibonacci Flower App

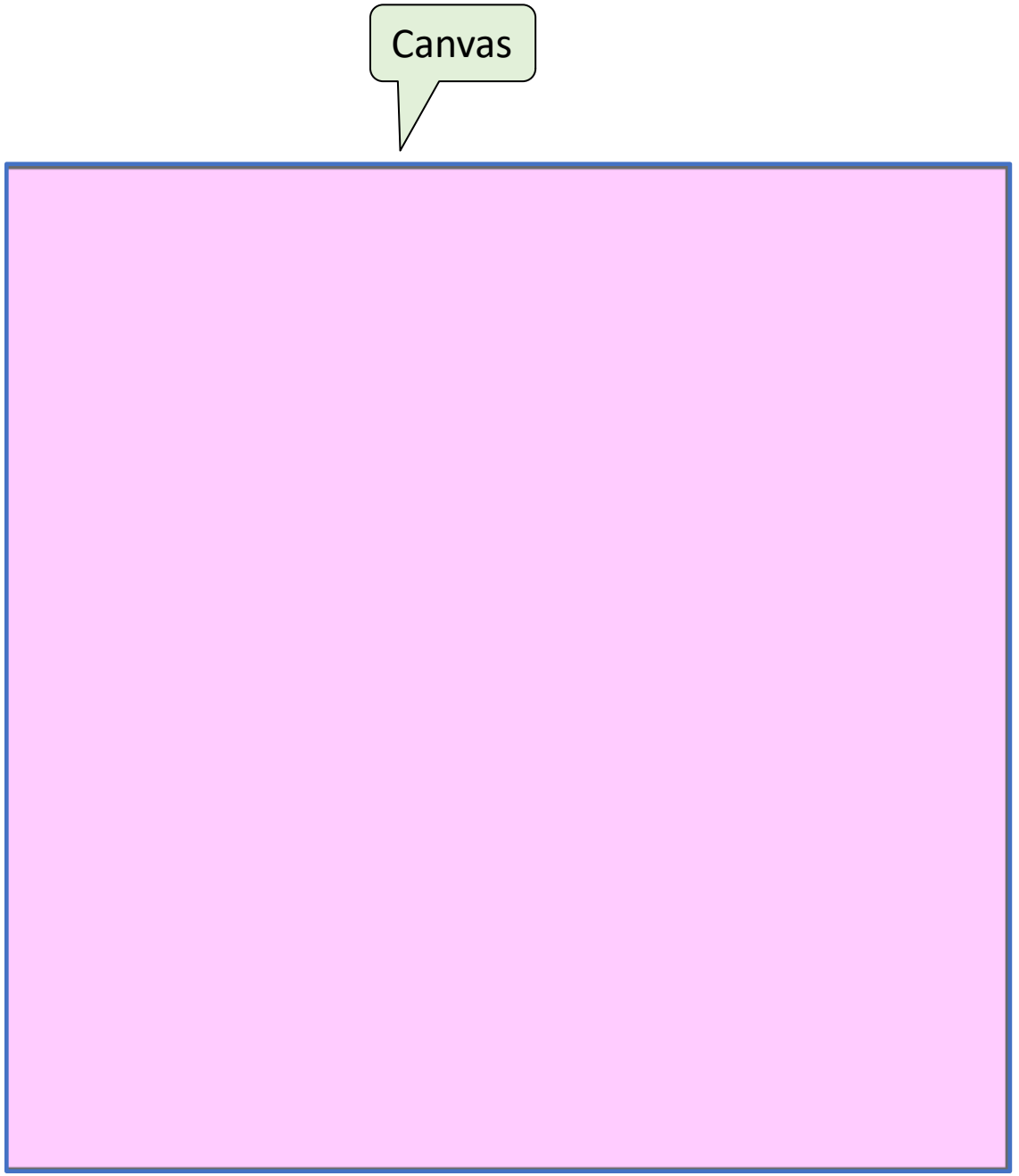
Goals

1. Build a Fibonacci flower that uses a golden ratio when computing the angle of a flower petal.
2. Use a timeline loop to build 40 petals.



Fibonacci App Settings



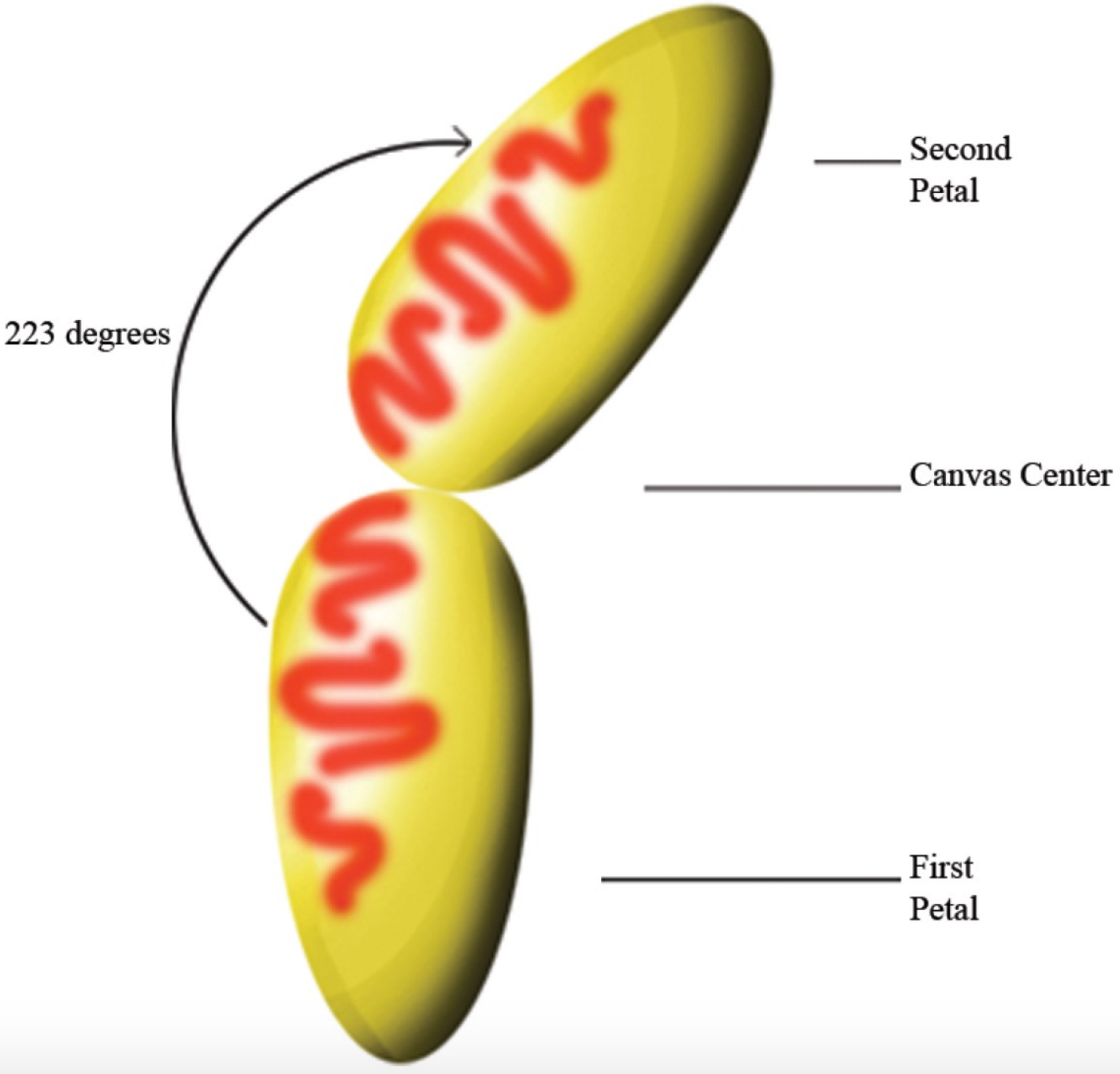


Petal Leaf calculations

1. The Golden Ratio will be used to offset the angle of each petal leaf before it is added to the canvas.

2. The angle of rotation for each new petal leaf is increased by 360 degrees multiplied by the Golden Ratio.

3. Each new petal leaf is shrunk in width and height than the previous petal.

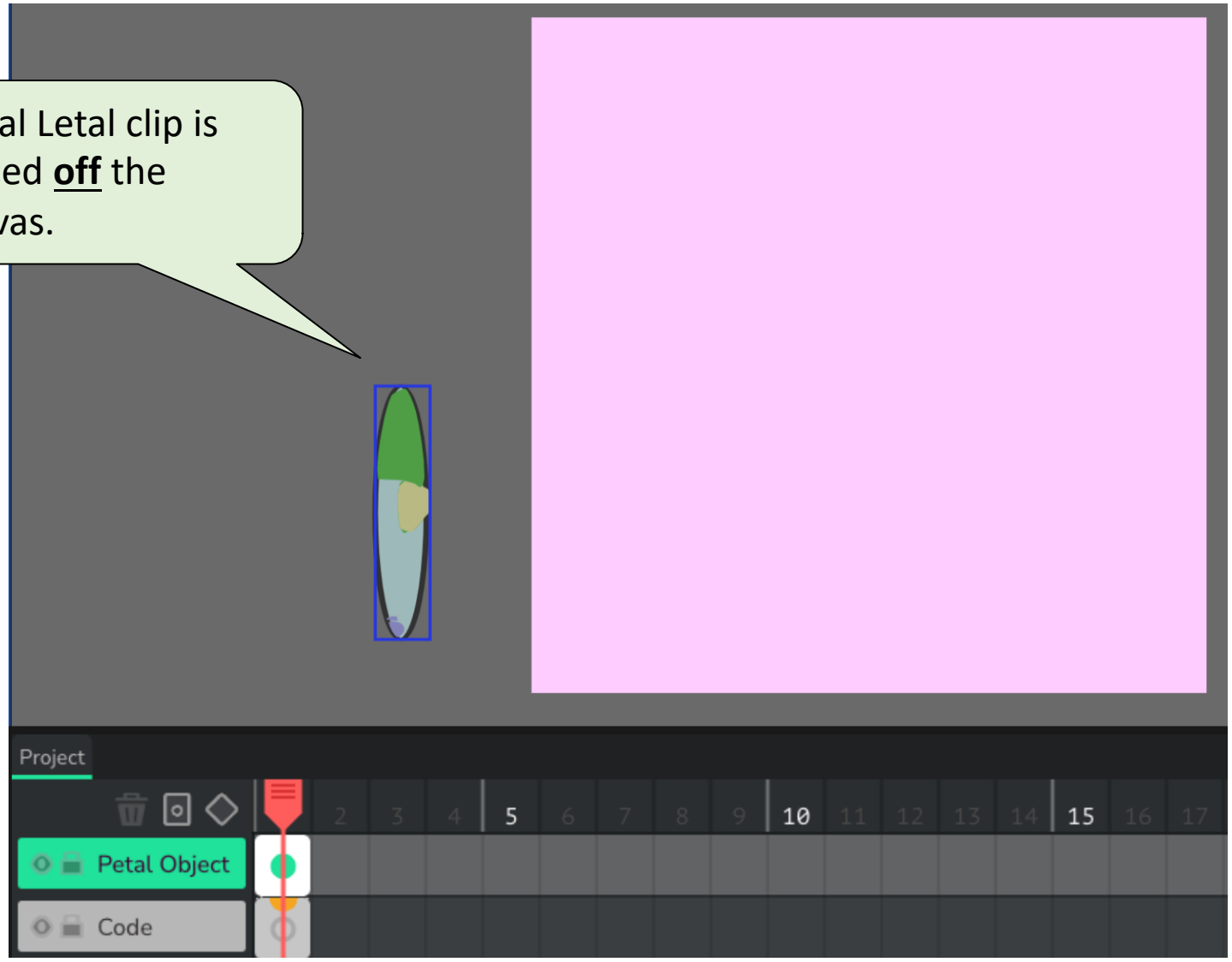


Petal Structure: **Leaf**

- Build a Clip representing a petal leaf.
- Name it **Leaf**.
- The **Leaf** Clip will be *cloned* many times during execution.

App Structure

Initial Letal clip is placed **off** the canvas.



All code is placed in the timeline.

TimeLine Code

- The code for the app will be placed in a TimeLine frame.
- Each petal will be cloned and added to the ***canvas.***
- ***Option:*** Use the timeline as a gameloop.

Default timeline code

Game Constants for Golden Angle and Petal Growth

Constants for Petal Shrinking and Total Number of Petals

Variables for Updating Petal Clones after each Iteration

Default timeline code

Game Constants for Golden Angle and Petal Growth

```
GOLDEN_RATIO = .618033989;
GOLDEN_ANGLE = 360 * GOLDEN_RATIO;
```

Constants for Petal Shrinking and Total Number of Petals

```
SHRINK_SIZE = GOLDEN_RATIO * .005;
TOTAL_PETALS = 300;
```

Variables for Updating Petal Clones after each Iteration

```
myWidth = 1.2;
myHeight = 1.2;
myAngle = 0;
```

GameLoop Tasks

```
TOTAL_PETALS--;

if (TOTAL_PETALS >= 0){

    //TASK 1: CREATE A PETAL CLONE
    //TASK 2: POSITION THE PETAL IN THE CENTER OF THE STAGE
    //TASK 3: ROTATE THE PETAL AND UPDATE THE ANGLE FOR THE NEXT PETAL
    //TASK 4: MODIFY THE WIDTH OF THE PETAL

}
```

let leafClone = Leaf.clone();

GameLoop

```
TOTAL_PETALS--;

if (TOTAL_PETALS >= 0){

   //TASK 1: CREATE A LEAF CLONE
   let leafClone = Leaf.clone();

   //TASK 2: POSITION THE LEAF IN THE CENTER
   leafClone.x = 400;
   leafClone.y = 400;

   //TASK 3: ROTATE THE LEAF AND UPDATE THE ANGLE
   leafClone.rotation = myAngle;
   myAngle += GOLDEN_ANGLE;

   //TASK 4: MODIFY THE WIDTH
   myWidth -= myWidth * SHRINK_SIZE;
   myHeight -= myHeight * SHRINK_SIZE;
   leafClone.scaleX *= myWidth;
   leafClone.scaleY *= myHeight;
}
```