Requirements: Initial Meeting Fall 2023

Name: Predator-Prey

Objects:

Predator: any sprite

Prey: any sprite

Obstacles: randomized “circles”

Game: Turn-by-Turn

Keep track of location and direction facing (direction)

Each turn, pick a location and based on the location we have a possibility of the direction of the face, and then this is selected.

Centering based on turn.

MVP

* Infinite plane
* Two objects (any) with turned-based movement
* Camera moves to object with a turn
* Keyboard for camera controls

Updated: 3/20/2024

# Legal Move Area

The mesh of each object's valid moves after initializing the rules of the game. The two parameters needed to create the mesh will be the maximum distance traveled in a single move and the minimum turning radius which I will call d and R.

The parameterization of the mesh is given by   
 with going from 0 to 1 and a going from to .

If you go to <https://sagecell.sagemath.org/> The code below shows you the shape of the region:  
R=1.4  
d=2  
var('t,a')  
parametric\_plot3d((1/a\*(1-cos(d\*a\*t)),1/a\*sin(d\*a\*t),0),(a,-1/R,1/R),(t,0,1))

If you are at a point given by the a and t value clicked (as given by the parameterization I sent you), the tangent vector to the curve (the direction of travel) should be given by <d sin(d a t), d cos(d a t)>. The unit vector in the direction of travel would be <sin(d a t), cos(d a t)>.

# Equation Inversion

Given a point the user clicks on, we often need to find the values for that produce this so we know what path the object will follow. The inverse equation is given here:  
[solve for u and v {x = 1/u\*(1-cos(d\*u\*v)), y = 1/u\*sin(d\*u\*v) } - Wolfram|Alpha (wolframalpha.com)](https://www.wolframalpha.com/input?i=solve+for+u+and+v+%7Bx+%3D+1%2Fu*%281-cos%28d*u*v%29%29%2C+y+%3D+1%2Fu*sin%28d*u*v%29+%7D)

A black background with white text

Description automatically generated

Note that in the above equation is substituted for and for in the previous equation for the legal move area. Also, n = 0, is used for term.

Once we know the values for (a,t) that produce the (x,y) we have the object follow the appropriate path:



Note this is the same equation as the Legal Move area, but a (or u) is fixed and t varies from 0 to the t value computed from the inversion. The object is also rotated to the unit tangent direction, so it’s facing the correct way

A screen shot of a computer screen

Description automatically generated

Game Design

## Options

These are the options that should be present in the loading/start screen

* Number of Predators (1-10)
* Number of Prey (1-10)
* Turning Radius for all Predators (0.5 – 5.0)
* Turning Radius for all Prey
* Max Travel Distance for all Predators (0.5 – 5.0)
* Max Travel Distance for all
* Number of Turns
* Location – 3 options
  + Close (4 times max travel distance)
  + Mid (8 times max travel distance)
  + Far (12 times max travel distance)
  + Note: Unless your predators take a herding strategy, group pursuits will split into several smaller case pursuits pretty quickly. So the initial configuration doesn’t have a huge effect on the long term behaviors.

## Flow

**Start Screen** with Start, Options, Exit button

* Exit click: Program closes
* Start click: Game starts
* Options click: Options panel opens

**Game**

1. Game loads with Prey and Predators shown
2. Legal movement area for prey is displayed
3. All Prey movement locations on this turn are selected with the mouse
4. The Predator legal movement area is displayed
5. All Predator legal movement locations are selected with the mouse
6. All Characters move to the desired location
   1. Scoreboard with turn number is updated
   2. Collisions between predator and prey result in the removal of prey
   3. The game stop condition is checked
      1. If the stop condition is met, go to End Game
      2. If the stop condition NOT met, go to 2.

**End Game Screen**

* Show the number of turns, the number of prey that escaped, and the winner
* Show button to return to start

## Game Stop Condition

* All prey captured.
* Number of Turns reached.

## Games States

* PreyTurn – the next location for all prey to move is selected.
* Predator Turn – the next location for all predators to move is selected
* MoveAvatars – predator and prey move “simultaneously”
* GameOver – reached when Game Stop Condition is met

# Notes

* LegalMoveArea on UI Layer and Predator/Prey on Default Layer, so they do not cause collisions, via setting:  
   