Cat Simulation

Fundamentals of Programming – Assessment 1

### By Viola Landucci

### Curtin ID: 20769446

Date: 10/10/2021

# Overview

## **Purpose:**

The program is a simulation of cats and how they behave based on terrain, other cats, the cats age, threats, and much more. The simulation can perhaps be used to see how cats in the real world, or in a certain situation may act.

The purpose of the simulation was to expand my knowledge, the creator of the program, on programming simulations and implementing multiple different programming skill to create the product.

## Features:

* **Age** – One feature in my program is age. In the program, the cats have three different ages: young, adult, and elderly. These three ages impact how the cats act, interact, and move.
* **Predators** – The program has the option for predators which can interact with the cats
* **Food** – The program contains food which attracts cats that are hungry.
* **Beds** – If a cat comes across a bed it will sleep for a short amount of time.
* **Time (Day/Night)** – Depending on the time of the day, that cats will act, interact, and move.
* **Terrain** – A hill is located in the program that makes the cats move slower to simulate moving up uneven terrain.
* **Visuals** – The program features an opening page which then leads into a clean and visually appealing display for the simulation using pygame. With simplistic colours, custom characters/icons, and visual text that displays information in a visually pleasing way.
* **Interactive** – The program contains two interactive buttons located at a lower corner of the screen. The buttons allow you to add cat and add predators

# User Guide

## Installations:

The program requires python3 and the module pygame.

To install the module, run the following command:

python3 -m pip install -U pygame --user

And to check if it has been installed, run:

python3 -m pygame.examples.aliens

If installation failed, refer to pygames documentation: <https://www.pygame.org/wiki/GettingStarted>

## Running:

To run the simulation, just run the main.py program file as you would any other python3 program.

## Reasons for Issues:

* Make sure you are running the program from the correct location
* Program may fail if pygame module is not ready/loaded completely
* Make sure pygame is installed and you have the correct python

## Notes:

Buttons in program needed to be clicked to work and please refrain from clicking button fast as it can freeze pygame.

# Traceability Matrix

|  |  |  |
| --- | --- | --- |
| Feature | Code – where code is located | Test – how I tested and what I was looking for |
| 1. Object Behaviour |  |  |
| 1.1 Cat object has position | Class Cats, self.x and self.y in \_\_init\_\_ function | [PASSED] Create Cat object with an assigned position  Tested: displayed positions |
| 1.2 Cat has speed attribute | Class Cats, self.speed in \_\_init\_\_ function | [PASSED] Create multiple cats with different speeds  Tested: observe for expected behaviour differences |
| 1.3 Cat has sleep attribute | Class Cats, self.sleep in \_\_init\_\_ function | [PASSED] Create multiple cats with different sleep values depending  Tested: outputted sleeping when sleeping |
| 1.4 Cat has hungry attribute | Class Cats, self.hungry in \_\_init\_\_ function | [PASSED] Create multiple cats with hungry values  Tested: output is cat is hungry |
| 1.5 Cat has age attribute | Class Cats, self.age in \_\_init\_\_ function | [PASSED] Create multiple cats with age values  Tested: outputted age |
| 1.6 Cat has age counter attribute | Class Cats, self.counter in \_\_init\_\_ function | [PASSED] Create multiple cats with age counter values  Tested: outputted age counter |
| 1.7 Cat has hunger counter attribute | Class Cats, self.hungryCounter in \_\_init\_\_ function | [PASSED] Create multiple cats with hunger counter values  Tested: outputted hunger counter |
| 1.8 Cat has sleep counter attribute | Class Cats, self.sleepCounter in \_\_init\_\_ function | [PASSED] Create multiple cats with sleep counter values  Tested: outputted sleep counter |
| 1.9 Cat has birth counter attribute | Class Cats, self.birthCounter in \_\_init\_\_ function | [PASSED] Create multiple cats with birth counter values  Tested: outputted birth counter |
| 1.10 Predator has position attribute | Class Predator, self.x and self.y in \_\_init\_\_ function | [PASSED] Create Predator object with an assigned position  Tested: displayed positions |
| 1.11 Predator has speed attribute | Class Predator, self.speed in \_\_init\_\_ function | [PASSED] Create multiple predators with different speeds  Tested: observe for expected behaviour differences |
| 1.12 Predator has counter attribute | Class Predator, self.counter in \_\_init\_\_ function | [PASSED] Create multiple predators with counter values  Tested: outputted counter |
| 1.13 Food has position attribute | Class Food, self.x and self.y in \_\_init\_\_ function | [PASSED] Create Food object with an assigned position  Tested: displayed positions |
| 1.14 Beds has position attribute | Class Bed, self.x and self.y in \_\_init\_\_ function | [PASSED] Create Bed object with an assigned position  Tested: displayed positions |
| 2. Terrain/Time (Day/Night) |  |  |
| 2.1 Food generates randomly on map | Under comment “# Setting up food random coordinates” | [PASSED] x and y cords are random and different  Tested: check x and y are random |
| 2.2 Beds generate randomly on map | Under comment “# Setting up bed random coordinates” | [PASSED] x and y cords are random and different  Tested: check x and y are random |
| 2.3 Hill generates on map from file | Under comment “# Gets hill x and y from file” – hill location in for loop | [PASSED] x and y cords are taken from file and used to specify x and y cords of hill  Tested: check if hill is in correct location and showing up |
| 2.4 Day/Night cycle | Under “if day <= 6” | [PASSED] Day and night are set depending on day counter  Tested: check and output day counter and see if day and night are changing correctly |
| 2.5 Day counter | Under “if day > 11” | [PASSED] Days counter increases after counter has reached a certain amount and stores days  Tested: outputting and check if days counter is increasing correctly |
| 3. Object and Terrain Interaction |  |  |
| 3.1 Adult Cat has breeding attribute | Under if I + 1 < len(adult) | [PASSED] Create adult Cat object when in contact is other adult cat and when birth counter allows it  Tested: printed “new cat” and viewed if young cats list went up |
| 3.2 Cat moves slower on hill | Under eatFood() function, under for i in range(len(young)) under comment “# moves slower on hill” (this is same for adult and elderly) | [PASSED] Cat moves slower when in hill boundary  Tested: output cat speed and see if there is a change when in hill boundary |
| 3.3 Cat sleeps in bed | In checkBed() function | [PASSED] Cat sleeps temporarily when comes across a bed  Tested: output when cat hits bed and output cats speed to if is 0 |
| 3.4 Cat moves to food when hungry | In findClosest() function and eatFood() function | [PASSED] When cat is hungry, find food and move towards it  Tested: output when cat is hungry and view if it moves towards food source |
| 3.5 Cat dies when interact with predator object | Under comment “# Predator Kill” and in function predKill() | [PASSED] When predator interacts with cat, cat dies  Tested: output when interaction occurs and output cats objects list to see if there was a decrease |
| 4. Display |  |  |
| 4.1 Cat button display | Under comment “# Button” and “# Button Display” | [PASSED] When button pressed, new cat appears on screen  Tested: test button and view if young cat counter increases |
| 4.2 Predator button display | Under comment “# Button” and “# Button Display” | [PASSED] When button pressed, new predator appears on screen  Tested: test button and view if predator counter increases |
| 4.3 key display | Under comment “# Key” | [PASSED] Key instructions are viewable on display  Tested: view if text is viewable |
| 4.4 Day/Night display | Under comment “# Setting Up Screen” | [PASSED] Time of day is viewable on display  Tested: view if text is viewable and if changes depending on time of day |
| 4.5 Day display | Under comment “# Setting Up Screen” | [PASSED] Days passed is viewable on display  Tested: view if text is viewable and if changes depending on days passed |
| 4.6 Colour of text and background depending on time of day | Under comment “# Setting up screen, buttons and text” | [PASSED] Colours of display change depending on time of day  Tested: view if colours change depending on time of day |
| 4.7 Grid display | In drawGrid() function | [PASSED] Grid is displayed on screen  Tested: view if grid is viewable |
| 4.8 Hill display | Under comment “# Draw circles for food and hill” | [PASSED] Hill is displayed on screen  Tested: view if hill is viewable |
| 4.9 Food display | Under comment “# Draw circles for food and hill” | [PASSED] Food is displayed on screen  Tested: view if food is viewable |
| 4.10 Bed display | Under comment “# Draw circles for food and hill” | [PASSED] Bed is displayed on screen  Tested: view if bed is viewable |
| 4.11 Young cat display | First thing under for i in range(len(young)) loop | [PASSED] Young cat is displayed on screen  Tested: view if young cat is viewable |
| 4.12 Adult cat display | First thing under for i in range(len(adult)) loop | [PASSED] Adult cat is displayed on screen  Tested: view if adult cat is viewable |
| 4.13 Elderly cat display | First thing under for i in range(len(elderly)) loop | [PASSED] Elderly cat is displayed on screen  Tested: view if elderly cat is viewable |
| 4.14 Predator display | First thing under for i in range(len(predator)) loop | [PASSED] Predator is displayed on screen  Tested: view if Predator is viewable |
| 4.15 Stats display | Under comment “# Display Stats” | [PASSED] Stats are viewable on display  Tested: view if text is viewable |
| 5. Object Movement |  |  |
| 5.1 Cat has random speed attribute depending on age | Class Cats, self.speed in \_\_init\_\_ function which is then randomly assigned a number for max speed in function newcat() and other places | [PASSED] Create multiple cats with different speeds  Tested: observe for expected behaviour differences |
| 5.2 Cat has random sleep Counter attribute | In newCat() function | [PASSED] Cat sometimes slept at night depending on age  Tested: outputted sleeping when sleeping |
| 5.3 Cat has random hungry attribute | In newCat() function and wherever a new cat object is being created | [PASSED] Cats randomly get hungry depending on hungry counter and random  Tested: outputted hungry when hungry |
| 5.4 Cat has random movement | In Cat() class, in randomCordClass() | [PASSED] Cat moved in random direction depending on random speed based on cat age and cat random number  Tested: outputted cat x and y coordinates |
| 5.5 Cat and Predator doesn’t move out of boundary | In Cat() class, in randomCordClass() | [PASSED] Predator moved in random direction by set random steps  Tested: outputted predator x and y coordinates |
| 6. Object Age/Life |  |  |
| 6.1 Young cat ages into adult cat | In “for i in young” loop | [PASSED] Young cat object has chance to turns into adult cat object after certain time  Tested: check young total and adult total to see if one decreases and one increases |
| 6.2 Adult cat ages into elderly cat | In “for i in adult” loop | [PASSED] Adult cat object has chance to turns into elderly cat object after certain time  Tested: check adult total and elderly total to see if one decreases and one increases |
| 6.3 Elderly cat dies after certain amount of time | In “for i in elderly” loop | [PASSED] Elderly cat object has chance to die after certain time  Tested: check if elderly total decreases randomly |
| 6.4 Predator cat dies after certain amount of time | In “for i in predator” loop | [PASSED] Predator object has chance to die/disappear after certain time  Tested: check if predator total decreases randomly |

# Showcase

## Introduction:

Cat simulation is a program based to display a simulation or cats and how they move and interact depending on other individuals, terrain and more. The code uses multiple different classes and function that allow the simulation to display different circumstances. The program has a boundary to make sure that all objects remain in there designated location.

### Features –

#### **1. Object Behaviour:**

##### 1.1 Cat object has position:

* Each separate cat object has a unique and different x and y coordinate that is within the designated boundaries that can be viewed on the program display.
* Later these values can be changed separately specific to the cat object

##### 1.2 Cat has speed attribute:

* Each separate cat object has a unique and different max speed value that is determined by the cats age (starts of as young cat limiters) that determines how fast/far a cat can move each interval.

##### 1.3 Cat has sleep attribute:

* Each separate cat object has a unique and different sleep probability value that is determined by the cats age (starts of as young cat limiters) that determines how high the chance of a cat sleeping is.
* Sleep is also impacted by time of day. During night, cats are most likely to sleep.

##### 1.4 Cat has hungry attribute:

* Each separate cat object has a hungry value that determines if the cat is hungry or not.
* Depends on hunger counter.

##### 1.5 Cat has age attribute:

* Each separate cat object has a unique and different age attribute which determines is the cat is young, an adult, or elderly.
* Age impacts multiple different things such as how long the cat has to live, likeliness of sleeping at night, if breeding is possible, and much more.

##### 1.6 Cat has age counter attribute:

* Each separate cat object has a unique age counter that begins at 0 and reverts back to 0 whenever a cat ages up.
* The age counter is used to determine when a cat is eligible to age up.

##### 1.7 Cat has hunger counter attribute:

* Each separate cat object has a unique and different hungry counter value which determines how fast a cat gets hungry depending on the cat object.

##### 1.8 Cat has sleep counter attribute:

* Each separate cat object has a unique and different sleep counter value which determines how fast a cat gets hungry depending on the cat object.

##### 1.9 Cat has birth counter attribute:

* Each separate cat object has a birth counter value which determines how fast a cat gets hungry depending on the cat object.

##### 1.10 Predator has position attribute:

* Each separate predator object has a unique and different x and y coordinate that is within the designated boundaries that can be viewed on the program display.
* Later these values can be changed separately specific to the predator object.

##### 1.11 Predator has speed attribute:

* Each predator object has a speed attribute that is set to 20.

##### 1.12 Predator has counter attribute:

* Each predator has a unique and different counter attribute.
* The counter allows the predator to be displayed for a set amount of time.

##### 1.13 Food has position attribute:

* Each food object has a unique and different x and y value that remains the same through the whole simulation. However, the cords change each simulation.

##### 1.14 Beds has position attribute:

* Each bed object has a unique and different x and y value that remains the same through the whole simulation. However, the cords change each simulation.

#### **2. Terrain/Time (Day/Night):**

##### 2.1 Food generates randomly on map

* Food object is assigned a random x and y cord that is within the set boundary.
* The food x and y cord remain the same through the simulation.

##### 2.2 Beds generate randomly on map

* Bed object is assigned a random x and y cord that is within the set boundary. However, they change each time the simulation is run.
* The bed x and y cord remain the same through the simulation. However, they change each time the simulation is run.

##### 2.3 Hill generates on map from file

* The hills x and y cords are read from a file and assigned to the hill.
* The cords are within the boundaries.
* The hills x and y cords remain the same through the whole simulation.

##### 2.4 Day/Night cycle

* Time of day is set depending on what number the day counter is on. It determines if it is day or night.
* This effects the cats object likeliness of sleep.

##### 2.5 Day counter

* Day counter is a counter that starts from 0 and resets one the counter has reached a value that is considered a day, which is 10 in this case.
* The counter is used to determine the time of day and how many days it has been since the simulation has started.

#### **3. Object and Terrain Interaction:**

##### 3.1 Adult Cat has breeding attribute

* When cats are of age (adults), they get given the chance to breed with other adult cats.
* The adults cats birth counter needs to be at a certain value to prevent excessive breeding between the cats.

##### 3.2 Cat moves slower on hill

* If cat object is located or becomes located in the boundary of the hill, the cats speed with decrease to simulate the change in terrain.
* However, if hungry, this change in speed will not affect the cat

##### 3.3 Cat sleeps in bed

* If cat object becomes in the presence of a bed, the cat will sleep for a short amount of time before moving on.

##### 3.4 Cat moves to food when hungry

* When cat objects hungry counter reaches 0, indicating that cat is hungry, the cat will choose a random food source and move towards it at max speed.
* Once cat reaches food source, the cat will not become hungry anymore and its hunger counter will reset.

##### 3.5 Cat dies when interact with predator object

* When cat and predator object come in contact, the predator will kill the cat to simulate cat predators in real life.
* The predator and cat interaction dose not change depending on age.

#### **4. Display:**

##### 4.1 Cat button display

* On the simulation display there is a button labelled “Add Cat”, if pressed, a cat will be added to the simulation and join in with the rest.
* The new cat will always be a young cat.

##### 4.2 Predator button display

* On the simulation there is another button labelled “Add Predator”, if pressed, a predator will be added to the simulation.

##### 4.3 key display

* A key will be on display on the simulation to indicate what everything is and means to avoid confusion.

##### 4.4 Day/Night display

* On the display, there is text that indicated the time of day.
* The text changes from “Day” to “Night” and so forth depending on the time of day.

##### 4.5 Day display

* There is a display of how many days it has been since the simulation commenced.
* Display increases as time increases and a new day has begun

##### 4.6 Colour of text and background depending on time of day

* Depending on the time of day, the colours of the simulation will differ.
* If day, the lightest colour will be a light pink and the darkest colour will be a dark pink
* If night, the lightest colour will be a light blue and the darkest colour will be a dark blue
* This affects the display text, grid, background, and buttons

##### 4.7 Grid display

* Using a function, a grid is drawn on the screen to help specify the simulation space.

##### 4.8 Hill, Food and Bed display

* Depending on the designated x and y cords on the hill, food, and beds, visuals are plotted on display.
* Gray circle is for hill
* Green circle is for food
* Cream circle is for beds

##### 4.9 Young, Adult, Elderly and Predator cat display

* Depending on the designated x and y cords on the young cat, adult cat, elderly cat, and the predator’s visuals are plotted on display.
* The images used for these object custom made to look visually appealing and to also make sure that each object can be identified easily.
* Young cat is an orange cat
* Adult cat is a brown cat
* Elderly cat is a grey cat
* The predators are red snake icons

Logo

Description automatically generatedShape, icon, circle

Description automatically generatedShape, icon, circle

Description automatically generatedShape

Description automatically generated

##### 4.10 Stats display

* Stats such as how many cats there are depending on age group there are alive and how many predators there are on the display are displayed on the simulation.

##### 4.11 Starter Screen

* Before the simulation commences, a starter screen is displayed to make the simulation seem more welcoming and professional.
* The screen has the title of the simulation and a button listed “Start” which when clicked, starts the simulation and displays it.

#### **5. Object Movement**

##### 5.1 Cat has random speed attribute depending on age

* Depending on the age of the cat, the cat object is assigned a random max speed that is within the age max speed boundaries.
* Younger cats have a medium chance of getting a higher speed, they are either slow or fast speed. They have a chance of getting a speed between 5 and 10.
* Adult cats have a medium chance of getting a higher speed, they are either medium speed or fast. They have a chance of getting a speed between 10 and 15.
* Elderly cats have a small chance of getting a higher speed, they can either move slow or medium speed. They have a chance of getting a speed between 3 and 6.

##### 5.2 Cat has random sleep Counter attribute

* Depending on the age of the cat, the cat object is assigned a random number that determines its likeliness of sleeping at night that is within the age max sleep boundaries.
* Younger cats have a small sleep range, they have a ½ chance of sleeping at night.
* Adult cats have a large sleep range, they have a ¼ chance of sleeping at night.
* Elderly cats have a medium sleep range, they have a 1/3 chance of sleeping at night.

##### 5.3 Cat has random hungry attribute

* When a cat object is created, a random sleep counter value is assigned to the cat. This determines how fast a cat gets hungry.

##### 5.4 Cat has random movement

* Depending on the cat’s object speed, the cat will move a random distance in a random direction each time step.
* This dose not apply when cat is hungry.

##### 5.5 Cat and Predator doesn’t move out of boundary

* Before cat moves to new location, program checks to make sure that cat is within boundaries of the designated area.
* Prevents objects from moving out of screen and keeps simulation clean.

#### **6. Object Age/Life:**

##### 6.1 Young cat ages into adult cat

* Young cat starts will a counter of 0, and if counter reaches specified (44/4 simulation days), young cat is given the possibility to age up to an adult cat.
* If young cat ages up, young cat object is removed and a new adult cat is created.

##### 6.2 Adult cat ages into elderly cat

* Adult cat starts will a counter of 0, and if counter reaches specified (44/4 simulation days), adult cat is given the possibility to age up to an elderly cat.
* If adult cat ages up, adult cat object is removed and a new elderly cat is created.

##### 6.3 Elderly cat dies after certain amount of time

* Elderly cat starts will a counter of 0, and if counter reaches specified (44/4 simulation days), elderly cat is given the possibility to die.
* If elderly cat dies, elderly cat object is removed.

##### 6.4 Predator cat dies after certain amount of time

* Predator starts will a counter of 0, and if counter reaches specified (33/3 simulation days), predator is given the possibility to leave.
* If predator leaves, predator object is removed.

## Methodology and Result:

### Simulation 1 –

#### Variables –

* PlayerNum = 10
* timeWait = 700

#### Terrain File –

200,250

#### Output –

* simulation commences with 10 young cats on display and 10 cat objects
* hill is located at cords x = 200 and y = 250
* young cat total display on simulation is displayed as 10 and 0 for the rest

### Simulation 2 –

#### Variables –

* PlayerNum = 5
* timeWait = 600

#### Terrain File –

300,270

#### Output –

* simulation commences with 5 young cats on display and 5 cat objects
* simulation is moving faster then simulation 1 because timeWait is smaller
* hill is located at cords x = 300 and y = 270
* young cat total display on simulation is displayed as 5 and 0 for the rest

### Simulation 3 –

#### Variables –

* PlayerNum = 20
* timeWait = 800

#### Terrain File –

700,370

#### Output –

* simulation commences with 20 young cats on display and 20 cat objects
* simulation is moving slower then simulation 1 because timeWait is larger
* hill is located at cords x = 700 and y = 370
* young cat total display on simulation is displayed as 20 and 0 for the rest

## Conclusion and Future Work:

Further investigations related to animals and simulations could include how different animals besides cat’s act, how multiple different species of an area interact (eg. Australian animals)

Further investigations related to just simulations could include a countless amount of thing such as covid/disease simulation, simulation on had people feel depending on a situation, and so much more.

There are multiple different features that could be added to the cat simulation. Some examples of features include territory setting and expanding, aggressive interaction between cats, other obstacle such as a lake, different weather and how it affects the cats, and much more.