

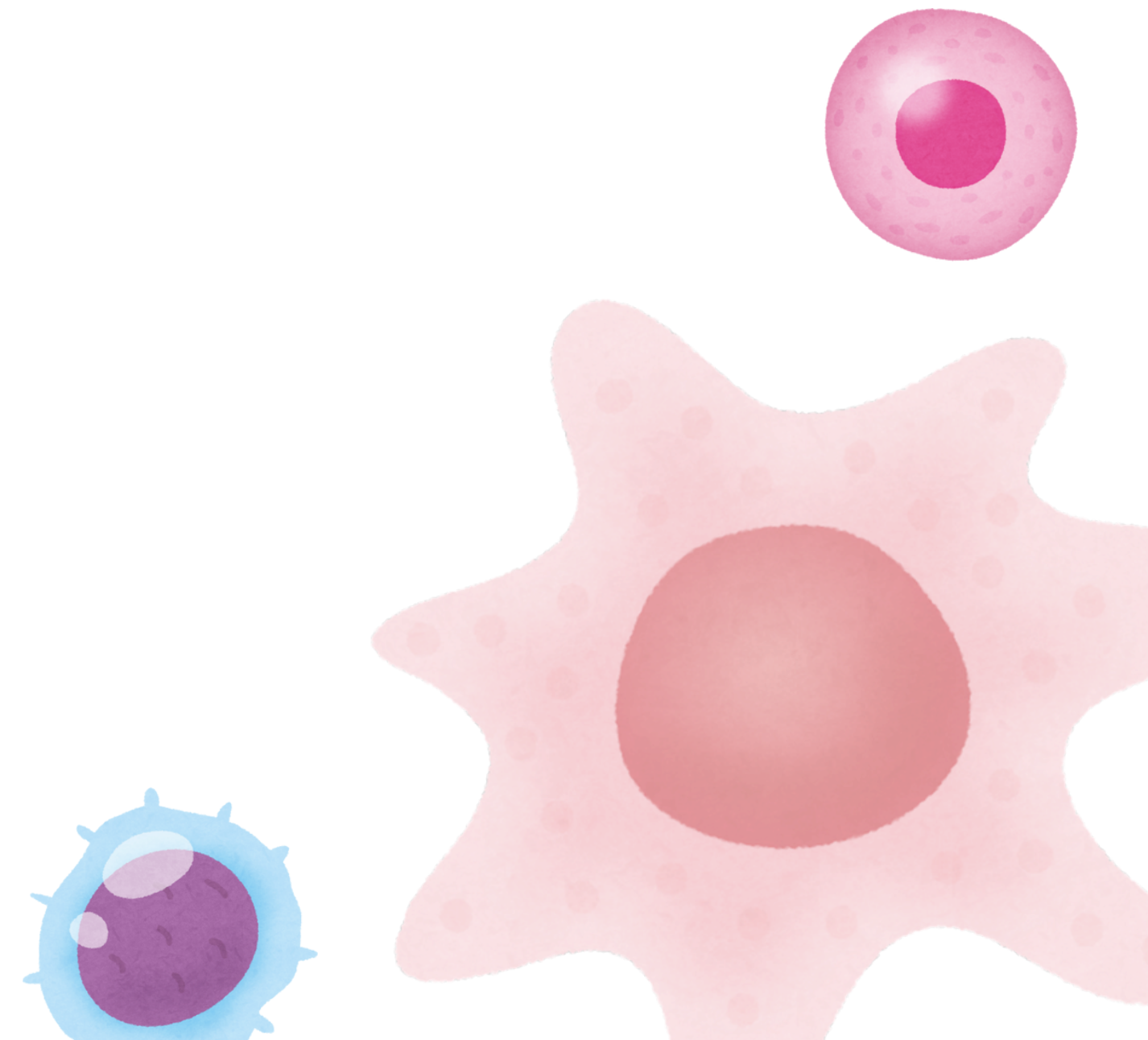
Epidemic Simulation

AI4423 Computer Simulation

Theeranop Laohaadet 650020

Nirawit Meeyindee 650023

Chaibodin Norasing 651975



Problem



Since the COVID-19 pandemic began, the virus has been spreading very quickly across the world.

Sometimes it's because we don't often encounter these types of events, or our response **plans just aren't good enough**

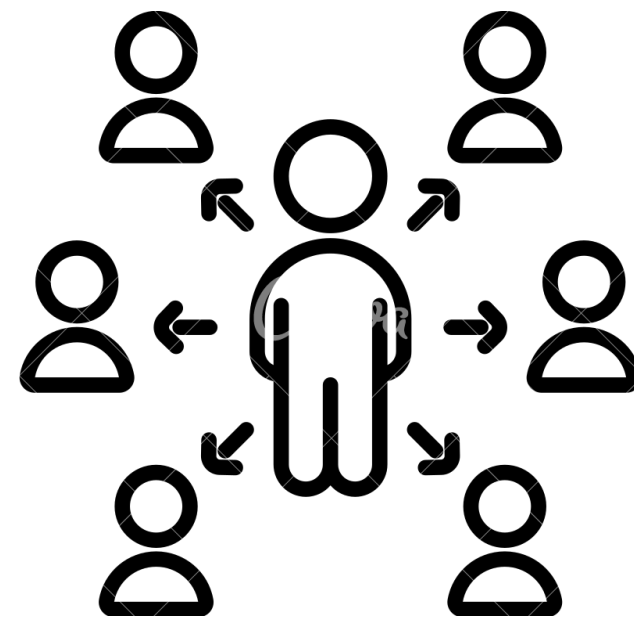
Problem

HOW MAKE A BETTER PLAN TO
PREVENT THE PANDEMIC? EX.



Develop a Vaccine

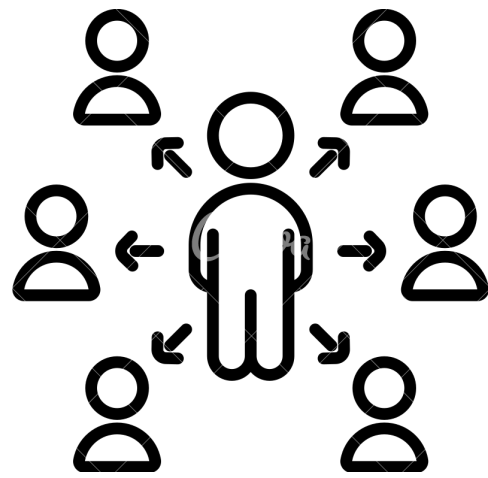
For prevent from
the virus and stop
the spreading



Trends in the spread and quarantine measures

Monitor the spread of the
disease to understand how to
separate the infected from the
healthy

Objective



**Trends in the spread and
quarantine measures**

Problem



To create a simulation that
shows the characteristics of
virus spread and the trends
in deaths, healthy, and
infections

Objective

Precondition and Postcondition

def _init_ ():

สำหรับกำหนดค่าเริ่มต้นของ Person

Precondition :

- Velocity > 0 #ความเร็วของแต่ละตัว
- infection radius > 0 #ระยะในการแพร่กระจาย
- Life_Time > 0 #เวลาที่คนติดเชื่อสามารถมีชีวิตอยู่ได้
- Avoidance_radius > 0 #ระยะห่างของผู้รอดชีวิตกับผู้ติดเชื่อ
- startingposition = $600 < x < 1920$ && $0 < y < 1080$
- status >= 0 = susceptible, 1 = infected, 2 = death

Postcondition :

- a Person object is create.
- set default status to susceptible
- random Person.position

Precondition and postcondition

def move():

สำหรับให้ Person เคลื่อนไหวตัว

Precondition :

- self ต้องมีค่า x, y
- ใช้ค่าคงที่ VELOCITY, GRAPH_WIDTH, SCREEN_WIDTH, SCREEN_HEIGHT, TAU

postcondition :

- update Person.position by random angle
- ป้องกันไม่ให้เคลื่อนที่ออกนอกจอ

Precondition and postcondition

def move_away_from():

สำหรับเพิ่มระยะห่างระหว่าง Person

Precondition :

- อ็อบเจกต์ self ต้องมีพิกัด x, y
- มีลิสต์ others ที่ประกอบด้วยบุคคล (Person) ที่ต้องการหลบเลี่ยง
- ใช้ค่าคงที่ VELOCITY, AVOIDANCE_RADIUS_MULTIPLIER

Postcondition :

- person.position is update
by moving away from
nearest infected person

Precondition and postcondition

def draw_person():

แสดงผลตำแหน่งของ person ทุกตัว

Precondition :

- ต้องมีหน้าจอ screen ที่สร้างด้วย pygame
- ต้องมีอ็อบเจกต์ person ที่มีแอตทริบิวต์ x, y, และ status

Postcondition :

- draw a circle on screen at
Person.position

Precondition and postcondition

def draw_graph():

สำหรับสร้างกราฟ

Precondition :

while running == true

ต้องมีข้อมูลอยู่ใน history_S, history_I, history_D

Postcondition :

- a Graph is create. using data from history_S, history_I, history_D

Project Overview

Simulation area

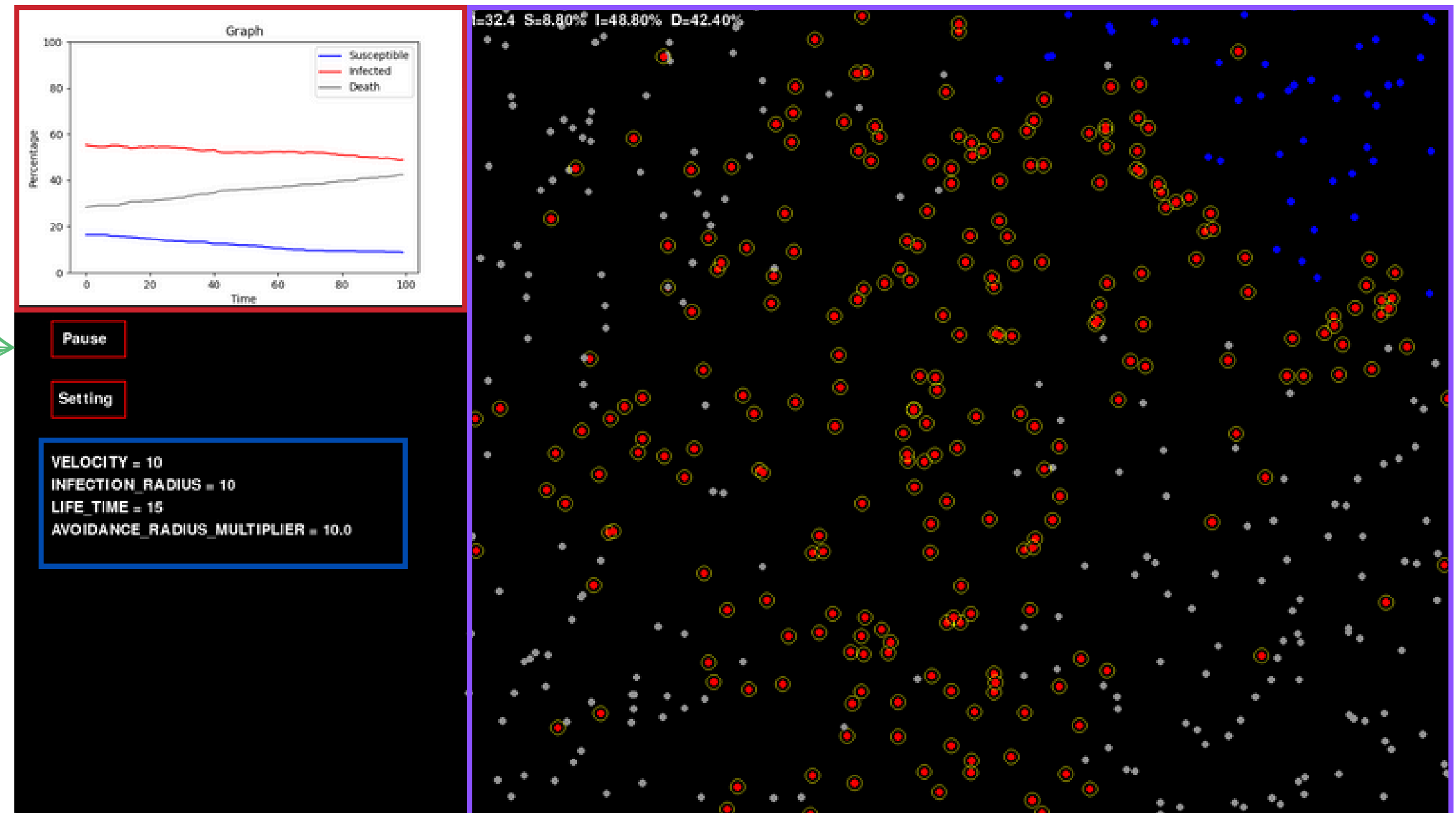
Monitor show a trends

- Susceptible (people with normal condition)
- Infected (person who infected from virus)
- Death (person who died from the virus)

Pause simulation

Setting (environment and agent control)

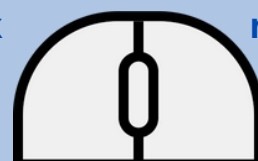
- **VELOCITY** (how fast the person goes)
- **INFECTION_RADIUS** (the size of the infected area around infected person)
- **LIFE_TIME** (how long the infection will turn death)
- **AVOIDANCE_RADIUS_MULTIPLIER** (how far the susceptible far from the infection)



Control

hover on the variable
you want to set

left click
Decrease



right click
Increase



+

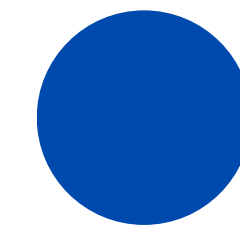
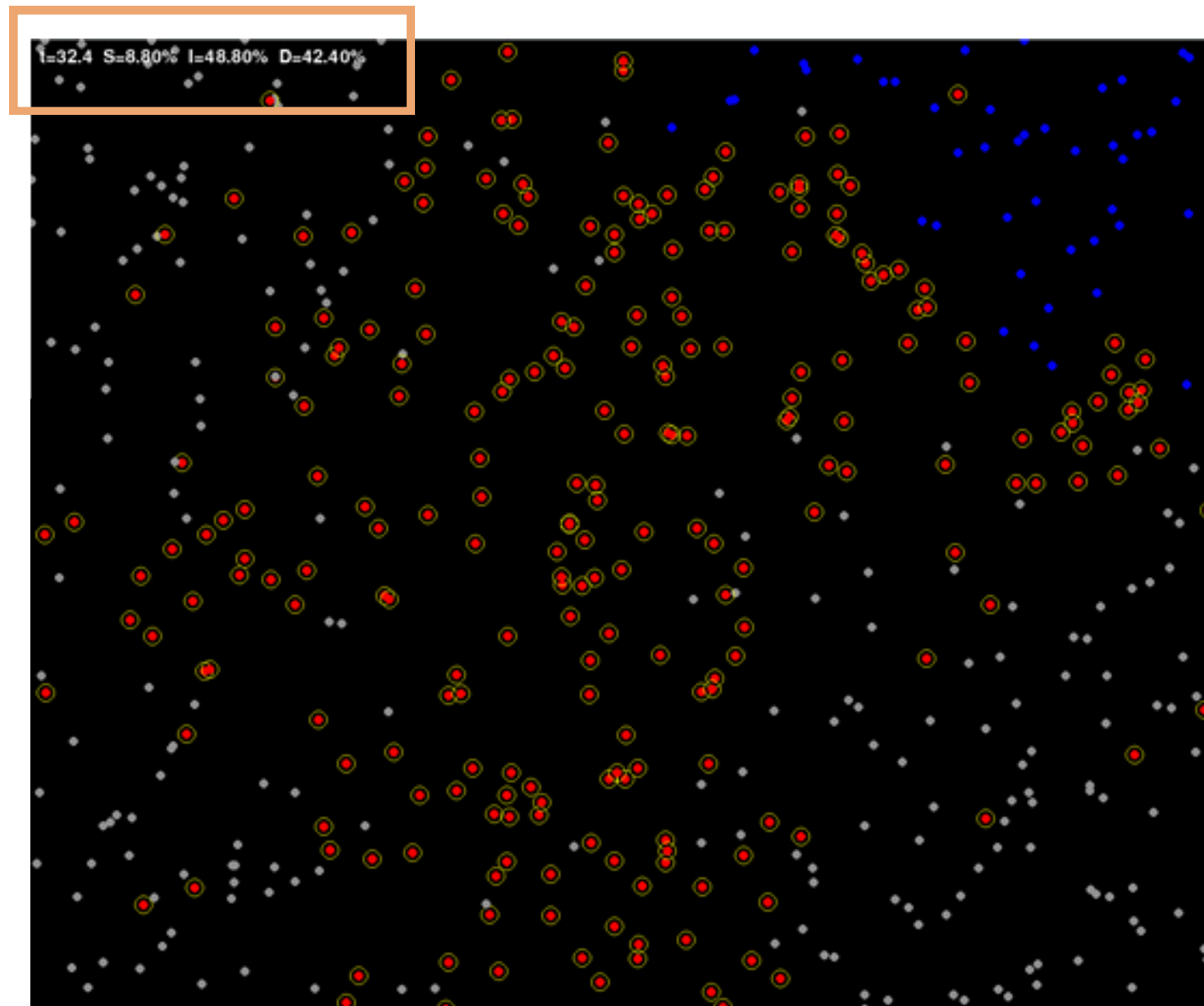


left click

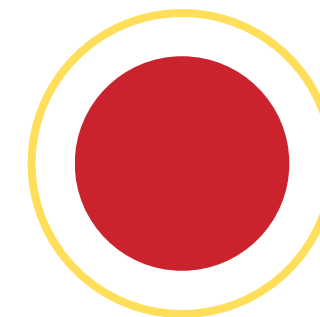
Get a new infected
person in map

Project Overview

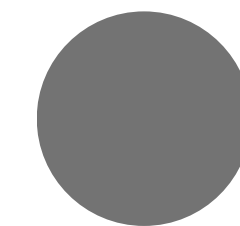
Ratio of population in simulation



Susceptible



Infection



Death

Project Preview

Future Development

Disease Variable



Study the disease further to have more variables for controlling or simulating a specific disease

Incubation phase ,how to spread a disease, symptom , recovery day

Future Development

Expand Simulation to Pandemic



Geography

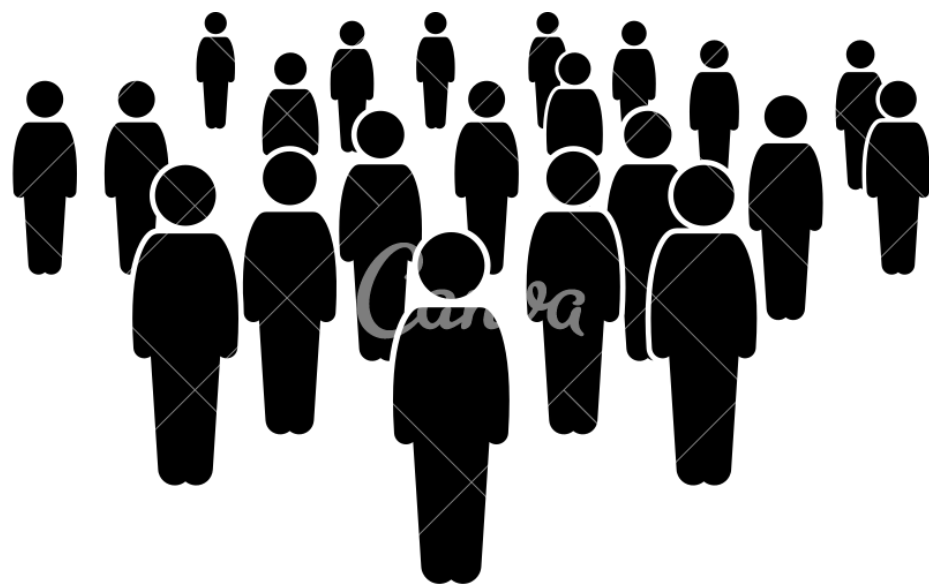
The geography of a country also affects the spread of disease for example, some diseases might spread more slowly in colder countries



Transport

The transportation routes in a country also reflect the trend of infection and how quickly it spreads and also about across the countries

Problems and Obstacles



ไม่สามารถ ใส่จำนวน person มากเกิน
ไปได้เพราะจะทำให้ทำงานได้ไม่เต็ม
ประสิทธิภาพ



matplotlib

การประสานระหว่าง Matplotlib กับ Pygame เนื่องจาก
Pygame ไม่รองรับการวาดกราฟในตัว จึงต้องใช้
Matplotlib ร่วมด้วย ซึ่งทำให้ต้องแปลงภาพกราฟจาก
Matplotlib ให้กลายเป็นภาพที่ Pygame แสดงผลได้ ซึ่งม
ความยุ่งยากและอาจเกิดข้อผิดพลาดได้ง่าย

Conclusion

The development of this epidemic simulation program provides a clear visualization of how a contagious disease spreads within a population, along with the trends in infection, recovery, and death over time. This program can serve as a basic tool for studying the behavior of infectious diseases and analyzing the effects of various parameters such as VELOCITY, INFECTED_RADIUS, LIFE_TIME and AVOIDANCE_RADIUS_MULTIPLIER.

Although there are limitations in terms of performance—especially when simulating a large number of people—and some complexity in integrating Matplotlib graphs with Pygame, the program successfully achieves its primary objectives. It also shows potential for future development, such as simulating an incubation period, adding symptom variables, or implementing pandemic control measures.

Refference

3Blue1Brown (2020) Simulating an epidemic. <https://www.youtube.com/watch?v=gxAaO2rsdIs>.