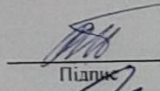
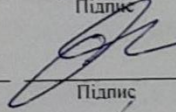
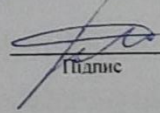


Хмельницький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерних наук

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему Імітаційне моделювання розповсюдження респіраторних захворювань

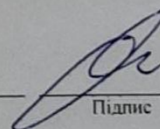
Галузь знань 12 – Інформаційні технології
Шифр і назва галузі знань
Спеціальність 122 – Комп'ютерні науки
Шифр і назва спеціальності
Освітня програма Комп'ютерні науки
Назва освітньої програми

Виконав: студент 4 курсу, група КН-18-1  В.О. Алексейко
Курс, група виконавця Підпис Ініціали, прізвище
Керівник: д.т.н., професор  О.В. Бармак
Науковий ступінь, посада Підпис Ініціали, прізвище
Нормоконтроль: к.т.н., доцент  Р.О. Багрій
Науковий ступінь, посада Підпис Ініціали, прізвище

До захисту допускаю:

Зав. кафедри КН, д.т.н., професор

01 червня 2022 р.


Підпис

О.В. Бармак
Ініціали, прізвище

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій

Кафедра комп'ютерних наук

Освітній ступінь бакалавр

Галузь знань 12 – Інформаційні технології

Спеціальність 122 – Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри комп'ютерних наук

(підпис)

д.т.н., професор О.В. Бармак

« 25 » березня 2022 року

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА**

1. Тема кваліфікаційної роботи бакалавра: «Імітаційне моделювання розповсюдження респіраторних захворювань»

2. Завдання видано студенту Алексеїку Віталію Олександровичу
(прізвище, ім'я, по батькові)

3. Керівник роботи професор кафедри КН Бармак Олександр Володимирович
(посада, прізвище, ім'я, по батькові)

4. Затверджено наказом університету від « 01 » березня 2022 р. № 18

5. Зміст пояснювальної записки (перелік задач) та вихідні дані:

Мета роботи – розробка імітаційної моделі розповсюдження респіраторних захворювань. Для валідації та верифікації імітаційної моделі необхідно розробити інформаційну систему на мові програмування Python. Система призначена для медичних працівників та організаторів масових заходів. Слід забезпечити можливість прогнозування зростання випадків респіраторних захворювань після контактів здорових людей з інфікованими в закритих приміщеннях. Провести аналіз отриманих результатів та порівняти зі статистичними даними, отриманими під час поширення захворювань.

Виконавець: студент 4 курсу, група КН-18-1
Курс, група виконавця

Підпис

В.О. Алексеїко
Ініціали, прізвище

Керівник: д.т.н., професор кафедри КН
Науковий ступінь, посада

Підпис

О.В. Бармак
Ініціали, прізвище

Анотація

Тема кваліфікаційної роботи бакалавра: «Імітаційне моделювання розповсюдження респіраторних захворювань»

Виконавець кваліфікаційної роботи бакалавра: студент групи КН-18-1
Алексейко Віталій Олександрович

Керівник кваліфікаційної роботи бакалавра: д.т.н., професор Бармак
Олександр Володимирович

Кваліфікаційна робота бакалавра містить:

Пояснювальна записка				Кількість додатків
Сторінок	Рисунків	Таблиць	Джерел інформації	
60	36	1	24	4

Метою кваліфікаційної роботи бакалавра є розробка імітаційної моделі розповсюдження респіраторних захворювань. Для валідації та верифікації розробленої імітаційної моделі розроблено інформаційну систему на мові програмування Python.

Розроблена система призначена для медичних працівників та організаторів масових заходів. Реалізована імітаційна модель дозволяє спрогнозувати зростання випадків респіраторних захворювань після контактів здорових людей з інфікованими.

Напрямами практичного використання розробленої інформаційної системи визначено вибір оптимальної кількості присутніх людей, відповідно до конфігурації приміщення, і як наслідок запобігання сплеску респіраторних захворювань.

Ключові слова: імітаційна модель, SI модель, гострі респіраторні захворювання, інформаційна система, агенти, інфікування, симуляція.

Виконавець: студент 4 курсу, група КН-18-1
Курс, група виконавця


Підпис

В.О. Алексейко
Ініціали, прізвище

Зміст

Перелік скорочень	3
Вступ.....	4
1 Характеристика предметної області та постановка задачі	6
1.1 Аналіз предметної області	6
1.2 Аналіз інформаційного забезпечення предметної області	9
1.2.1 Аналіз існуючого програмного забезпечення предметної області.....	9
1.2.2 Аналіз математичних методів.....	11
1.3 Аналіз сучасних засобів створення програмного забезпечення	16
1.4 Постановка задачі та вимоги до розробки інформаційної системи.....	17
2 Проектування інформаційної системи.....	18
2.1. Імітаційна модель розповсюдження респіраторних захворювань.....	18
2.2. Функціональна модель системи	30
2.3 Інформаційна структура системи	32
2.3 Вибір засобів розробки інформаційної системи	33
2.3.1 Вибір мови програмування	34
2.3.2 Вибір додаткових засобів розробки	35
2.3.3 Вибір середовища збереження та керування даними	36
2.3.4 Візуалізація процесу розповсюдження гострих респіраторних захворювань	38
3 Програмна реалізація інформаційної системи	400
3.1 Структура та функціональне призначення складових системи	400
3.2 Особливості реалізації складових системи	422
3.3 Інструкція користувача.....	42
3.4 Оцінка результатів	477
3.5 Вимоги до розгортання інформаційної системи.....	558
Висновки	600
Перелік посилань.....	611
Додатки	

Перелік скорочень

Скорочення, термін, позначення	Пояснення
ВООЗ	Всесвітня організація охорони здоров'я
ГРВЗ	Гострі респіраторні вірусні захворювання
ГРЗ	Гострі респіраторні захворювання
ІТ	Інформаційні технології
КН	Комп'ютерні науки
КРБ	Кваліфікаційна робота бакалавра
МОЗ	Міністерство охорони здоров'я
НСЗУ	Національна служба здоров'я України
ПЗ	Програмне забезпечення
ПП	Програмний продукт

Вступ

Збереження та зміцнення здоров'я населення є важливим соціально-економічним завданням, невід'ємним аспектом якого є зниження рівня захворювань. Серед великої кількості захворювань, одними з найбільш критичних для населення є хвороби, які передаються від людини до людини повітряно-крапельними шляхами (наприклад, краплі, що виділяються з дихальних шляхів людини під час кашлю, чхання або навіть спілкування) через швидкість їх поширення, масштаб, велику кількість хворих і померлих тощо. З часом з'являються все більш прогресуючі в просторі і часі інфекційні захворювання, симптоми яких менш помітні, вірулентність вища, з широким спектром способів поширення. Такі захворювання важко контролювати, вони навіть переходять в надзвичайні стани, їх називають епідеміями. Епідемія (від грецького *επί* – «серед» і *δῆμος* – народ; застаріле – пошесть, мор, моровиця) – тип хвороби, яка є новою для даної популяції впродовж періоду збереження імунної «пам'яті» та поширюється зі швидкістю, що значно перевищує очікувану, ґрунтуючись на попередньому досвіді (тобто, числі нових випадків за одиницю часу). Це висока ступінь інтенсивності епідемічного процесу. Епідемічний процес – безперервний процес передачі інфекції від хворих людей або хворих тварин (джерела інфекції) здоровим людям [5].

Українське законодавство визначає, що «епідемія – це масове поширення інфекційного захворювання серед населення відповідної території за короткий проміжок часу» [3]. Термін «епідемія» використовують, коли захворювання охоплює значні регіони. У разі надзвичайно широкого поширення епідемії (кілька країн, чи навіть на кількох континентах) часто вживають термін «пандемія» (від грецького *παν* – увесь і *δῆμος* – народ). Це найвища інтенсивність розвитку епідемічного процесу, що характеризується прогресуючим поширенням інфекційного захворювання з надзвичайно високим ураженням населення на значних територіях материків або всієї Землі [1, 22].

Об'єкт дослідження – інфекційні захворювання.

Предмет дослідження – процес поширення інфекційних захворювань.

Мета кваліфікаційної роботи – провести аналіз існуючих моделей поширення інфекційних захворювань та розробити власну імітаційну модель та провести її валідацію та верифікацію за допомогою розробки відповідної інформаційної системи. Виходячи з поставленої мети, в роботі необхідно вирішити наступні завдання:

- розглянути теоретичні аспекти поширення інфекційних захворювань;
- розглянути математичні методи, за допомогою яких можна описати характер поширення захворювань, що передаються повітряно-крапельним шляхом;
- розглянути технології, які можуть бути використані при розробці імітаційної моделі респіраторних захворювань і перспективи їх використання;
- розробити імітаційну модель респіраторних захворювань та реалізувати її у вигляді інформаційної системи.

Методи дослідження: описовий, аналіз, порівняльний, моделювання, експерименту.

1. Характеристика предметної області та постановка задачі

1.1 Аналіз предметної області

Гострі респіраторні захворювання (ГРЗ) – найбільш поширені інфекційні захворювання, що вражають усі вікові групи населення. Серед причин тимчасової втрати працездатності вони посідають перше місце – навіть у міжепідемічний період на них хворіє 1/6 частина населення планети. В Україні щорічно на ГРЗ хворіють 10–14 млн осіб, що становить 25–30% усієї та близько 75–90% інфекційної захворюваності у країні. Експерти Всесвітньої організації охорони здоров'я (ВООЗ) відзначають, що ця найпоширеніша в людській популяції група захворювань в останні роки має постійну тенденцію до збільшення. Соціальні причини, пов'язані з глобальним процесом постійного зростання урбанізації, більш тісні контакти людей практично в будь-якій точці земної кулі та посилення міжконтинентальних міграційних процесів сприятимуть подальшому поширенню ГРЗ.

Для ГРЗ властиві дві форми епідемічного процесу – спорадичні захворювання та епідемічні спалахи. У період епідемічного спалаху відзначається переважання певної нозології, однак ніколи етіологічна структура не буває однорідною. Навіть під час епідемії грипу реєструються й інші ГРЗ. Тим більше, у міжепідемічний період грипу сезонне підвищення захворюваності на ГРЗ завжди має «строкату» структуру, в якій зазвичай відстежуються збудники, що мають найбільше клінічне та епідеміологічне значення. Обидві групи захворювань постійно поповнюються новими представниками, які можуть стати причиною важкої патології. При ГРВЗ особа з клінічними проявами захворювання завжди є джерелом інфекції. Виділення збудника відбувається уже в період інкубації, однак найбільша небезпека виникає саме на початку гострого періоду. Тривалість виділення вірусу іноді може становити 1 – 2 місяці і довше. Проте, якщо ще донедавна джерелом інфекції при ГРВЗ вважалася лише людина, то сьогодні зараження, для прикладу, деякими варіантами вірусу грипу можливе від птахів, свиней із модифікацією шляхів передавання.

Повітряний шлях передавання збудника, висока сприйнятливість населення практично до всіх вірусів-збудників ГРВЗ – зумовлює основну епідеміологічну особливість – швидкість та широту їхнього розповсюдження. Вони практично необмежені, суттєво зростають в умовах у періоди сезонного імунодефіциту, що виникає в зимовий та зимово-весняний періоди [7].

Сьогодні актуальною проблемою людства стала коронавірусна інфекція Covid-19. Спочатку цій хворобі не надавалося великого значення, але з часом, масштаб проблеми збільшився до рівня усієї планети: кількість смертей досягла сотень тисяч, згодом – мільйонів; були нанесені великі збитки в економіці багатьох держав через карантинні обмеження тощо. Але коронавірус є лише частковим випадком більшості інфекційних захворювань, які можуть найближчим часом з'явитися, тому потрібно напрацювати методи моделювання, а згодом і аналізу розповсюдження, а також боротьби з цими проблемами.

Попереджувальні заходи є ключовими у вирішенні цього завдання. Прогнозування динаміки розповсюдження захворювань дозволяє розробити і застосувати адекватні заходи протидії, забезпечити раціональне використання матеріальних і людських ресурсів.

Імітаційне моделювання захворювань є актуальною проблемою в сучасному світі. Усе більше дослідників звертаються до математичних моделей для створення прогнозів тих чи інших захворювань, так як вони допомагають найбільш вірно і точно вивчити зміни певних процесів, які відбуваються в соціумі. Математичне моделювання незамінне в певних галузях медицини, де неможливі або утруднені реальні експерименти, наприклад в епідеміології.

Наріжним принципом математичного моделювання є так званий «тезис надійності»: модель, значення якої приблизно відповідає реальності буде давати прогноз, який буде приблизно достовірним. Важливо вивчити результати різних моделей, щоб виявити, яким моделям можна довіряти.

Математичне моделювання в прогнозуванні питань розповсюдження захворювань може надати глобальну допомогу в прийнятті рішень їх профілактики, лікування, прогнозування.

Математична модель – абстрактне втілення нашого уявлення про систему або про процес, представлена у вигляді математичних символів, формул, рівнянь. Комп'ютерна модель – це математична модель, записана на мові програмування і реалізована у вигляді програми для електронно-обчислювальної машини. [6].

Коли математична модель побудована, математичний аналіз в поєднанні з комп'ютерним моделюванням допомагає нам дослідити глобальну поведінку моделі, виявляючи наслідки зроблених припущень [14].

Метод математичного моделювання – це важливий інструмент вивчення закономірностей, що лежать в основі функціонування складних систем довільної природи, і біомедичних зокрема. Основний принцип математичного моделювання складних систем – принцип оптимальності. Модель має бути максимально простою, тобто має містити мінімальну кількість змінних (відповідно і рівнянь), а також мати порівняно прості зв'язки між змінними. Важливою процедурою, яку використовують розробники моделей для перевірки надійності передбачень, зроблених математичною моделлю є порівняння різних моделей [19, 21, 24]. Таким чином, якщо проста модель робить прогноз, і якщо такий самий, або дуже схожий прогноз робиться за допомогою більш складної моделі, яка містить деякі механізми чи деталі, яких не було в першій моделі, то можна припустити, що прогноз надійний.

Математична модель розповсюдження респіраторного захворювання в популяції описує передачу патогена між господарями залежно від характеру контактів серед хворих і сприйнятливих людей, латентного періоду, від зараження до зараження, тривалості хвороби, ступінь набутого імунітету після хвороби тощо. Після того, як усі ці фактори будуть змодельовані, можна зробити прогнози про кількість інфікованих людей, пікової захворюваності, тобто передбачити усю епідемію і модель представить очікувану кількість випадків на кожен момент часу.

Отже, при дослідженні поширення респіраторних захворювань важливим етапом є створення моделі, яка була б наближеною до реальної. Таким чином, доцільно використати імітаційну модель, яка побудована на основі теорії

ймовірності та SI моделі. Вхідними параметрами для імітаційної моделі будуть: розмір вибірки, початкова кількість інфікованих, мінімальна та максимальна ймовірність інфікування. Оскільки імунітет кожної людини індивідуальний, ймовірність захворіти для конкретної людини є неперервною випадковою величиною. Межі цієї величини обмежені параметрами: мінімальна та максимальна ймовірність інфікування. Ці параметри залежать безпосередньо від захворювання, що досліджується.

1.2 Аналіз інформаційного забезпечення предметної області

1.2.1 Аналіз існуючого програмного забезпечення предметної області

Програмне забезпечення, що використовується для моделювання респіраторних захворювань дуже різноманітне та постійно вдосконалюється. Проте не кожен програмний продукт є зрозумілим у користуванні пересічній людині, оскільки націлений на вузьку аудиторію. Одна частина програмного забезпечення має недосконалу схему інфікування, інша – не дає можливості візуалізувати процес поширення захворювань. Також далеко не всі програми забезпечують можливість виведення отриманої в ході експерименту інформації на графіки чи діаграми, з метою подальшої обробки.

Розглянемо деяке існуюче програмне забезпечення, що використовується для моделювання респіраторних захворювань.

EpiModel – це пакет, який надає інструменти для моделювання та аналізу математичних моделей динаміки інфекційних захворювань (рис. 1.1). Підтримувані класи моделей епідемій включають детерміновані компартментальні моделі, стохастичні моделі індивідуальних контактів і стохастичні мережеві моделі. Типи захворювань включають епідемії SI, SIR та SIS з демографією та без неї, з утилітами, доступними для розширення для створення та моделювання епідемічних моделей довільної складності. Клас мережевої моделі заснований на статистичній структурі тимчасових

експоненційних моделей випадкових графів (ERGM), реалізованих у пакеті програмного забезпечення Statnet.

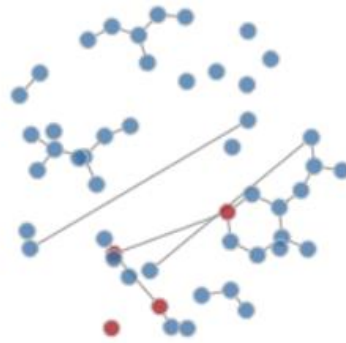


Рисунок 1.1 – Модель, розроблена в середовищі EpiModel

Імітаційна модель Covid-19 в AnyLogic Cloud допомагає змоделювати поширення хвороби в сім'ї (рис. 1.2). Програмне забезпечення передбачає ріст захворюваності на основі кількості людей та числа близьких контактів. Також прогнозується необхідність госпіталізації.



Рисунок 1.2 – Імітаційна модель Covid-19 в AnyLogic Cloud

Дослідження подібних моделей призводить до висновку, що самоізоляція та скорочення контактів допоможе впоратися із поширенням вірусу.

Безумовними перевагами подібних програмних продуктів є можливість спрогнозувати ріст захворювання, передбачити можливі сплески. Проте, багато моделей розробляються під одну конкретну хворобу, що унеможлиблює їх широке застосування. Велика кількість вхідних даних буде доцільною для експертів у галузі епідеміології, проте пересічним користувачам це ускладнить, або навіть унеможливить роботу. Вищевикладені аспекти необхідно врахувати при розробці імітаційної моделі, адже доцільним буде максимально зменшити кількість вхідних параметрів, і в той же час адаптувати модель до реальної.

Отже, програмне забезпечення має бути зручним та зрозумілим у використанні, забезпечувати можливість візуалізації протікання захворювання, передбачати виведення інформації на графіки або діаграми, задля подальшого аналізу, бути ергономічним. Математична модель, що використовується програмним продуктом має бути максимально наближеною до реальної.

1.2.2 Аналіз математичних методів

Респіраторні вірусні інфекції – це найбільш масові захворювання, які займають провідне місце у структурі інфекційних хвороб і складають 80-90 % від загальної кількості захворювань. Враховуючи здатність вірусу грипу викликати щорічні епідемії у масштабах всієї земної кулі, можна стверджувати, що грип є проблемою світового значення. Під час епідемії хворіє від 5% до 20% населення, а у період пандемічного розповсюдження грипу – інфікується абсолютна більшість населення [17].

Під час чхання або кашля, з рота хворої людини вилітають дрібні частки слини і мокротиння, у яких віруси містяться у величезних кількостях. Тому основний механізм передачі грипу та ГРВІ називається – повітряно-крапельний. Інший механізм передачі респіраторної інфекції – контактний. Він довгий час

залишався недоведеним і менш очевидним, ніж повітряно-крапельний. Тим не менш, він відіграє не меншу, а можливо, і більшу роль у поширенні захворювань. Як правило, людина, яка чхає або кашляє, прикриває рот рукою, сподіваючись запобігти поширенню інфекції повітряно-крапельним шляхом. При цьому вона і не підозрює, наскільки полегшує передачу своєї інфекції контактним шляхом. Це пояснюється тим, що вся колосальна маса мікробів, яка повинна була вийти у відкритий простір, осідає на руці людини, що чхає або кашляє. Після цього вони розносяться на предмети побуту, зокрема й ті, яких торкаються інші люди. Також передача збудників відбувається при рукостисканнях. Після дотику руки до рота, носа або очей, які також вкриті сприйнятливою до ГРВІ слизовою оболонкою, і складний повітряно-крапельний шлях передачі скорочується для вірусу за часом і складністю в десятки разів.

Одним із найбільш ефективних шляхів боротьби з поширенням респіраторних захворювань може стати моделювання їх розповсюдження.

Математична модель дозволяє описати можливості досліджуваного об'єкта за допомогою системи формул математичних функцій та рівнянь. Наприклад, до математичних моделей природних явищ відносяться загальновідомі закони тяжіння, закон Ома. Для моделювання динамічних процесів використовується система диференціальних рівнянь, що дають змогу відобразити зміну величин у дослідній системі за допомогою похідних. Математичне моделювання будь-якого процесу можливе лише при умові наявності достатньої інформації щодо його фізичних та біологічних закономірностей. Але розвиток технологій сучасності дає змогу значно розширити можливості застосування математичної моделі в медицині не тільки для моделювання простих систем, але й для цілого комплексу складних.

Математичне моделювання дозволяє дослідити поведінку біологічних систем в умовах, які дуже складно й затратно відтворити в клінічному експерименті. Ще однією перевагою застосування математичної моделі є скорочення часу дослідження, оскільки комп'ютер розбирає достатньо велику кількість можливих варіантів досліду за короткий проміжок часу. Також

математична модель спрощує розв'язання задач із лікування хвороб, оскільки вона прискорює аналіз вхідних даних та вирішення завдань, які можуть виникнути в процесі лікування.

Математичні моделі дають змогу продемонструвати розвиток інфекційних захворювань, щоб показати можливі наслідки епідемій та вчасно проінформувати заклади охорони здоров'я, щодо можливих спалахів хвороби. Ці моделі використовують деякі базові припущення в галузі медицини та математичні перетворення, для того щоб знайти параметри для різних інфекційних захворювань, та використати ці параметри для обчислення наслідків від можливих заходів, таких як програма масової вакцинації.

На сьогоднішній день не вироблено загальновизнаного методу опису інфекційних захворювань, оскільки необхідні умови для їх створення з'явилися тільки два-три десятиліття тому, а саме:

- рівень людського розвитку в потрібних галузях математики, для того, щоб правильно описати модель, яка буде апроксимувати реальну ситуацію;
- рівень людського розвитку в програмуванні (алгоритмах і структурах даних), для оптимізованої і швидкої роботи програм;
- рівень технологічного розвитку, для швидкого виконання програм за допомогою розпаралелювання та векторизації певних моментів в коді.

Математичне моделювання розповсюдження інфекційних захворювань має давню історію. Розглянемо історичні аспекти вивчення можливостей застосування математичного моделювання в медицині.

В 1760 році D. Bernoulli (1700 – 1782) вперше застосував математичний аналіз для вивчення інфекційних захворювань і оцінював з його допомогою ефективність різних способів щеплення проти віспи. Цей аналіз був вперше представлений в Королівській академії наук в Парижі 1760 року і опублікований в 1766 році [8].

У 1840 році W. Farr за допомогою кривої нормального розподілу описав дані смертей від віспи в Англії та Уельсі за 1837-1839 р.р. В 1906 році John Brownlee продовжив роботу над цим методом, співставивши ряди

епідеміологічних даних на основі розподілу Пірсона, про що описав в статті «Статистичний підхід до імунного захисту: теорія епідемій» [9].

Найбільш відома модель в математичній епідеміології – це проста модель SIR, описана W.O. Kermack та A.G. McKendrick в 1927 році. У ній за допомогою систем диференціальних рівнянь (неперервний час) або різницевих рівнянь (дискретний час) описується динаміка груп сприйнятливих, інфікованих та осіб, які одужали [16].

У цій моделі популяція поділяється на три групи людей: сприйнятливих до захворювання (Susceptible), інфікованих (Infected) і осіб, які одужали після захворювання і до нього не сприйнятливі (Recovered). Припускають, що розповсюдження інфекційного захворювання проходить через контакт сприйнятливих з інфікованими, а ті що одужали перестають розповсюджувати хворобу.

В цих працях для завдань епідеміології був вперше застосований так званий «закон діючих мас», згідно якого кількість осіб, знову інфікованих в популяції прямо пропорційна добутку поточної чисельності сприйнятливих та інфікованих індивідів. При такому математичному моделюванні можна зробити висновки про епідеміологічний поріг, розмах епідемії, коли вона відбудеться. Епідемічний поріг передбачає, якщо провакцинувати частину населення до прибуття патогена, щоб знизити початкову долю сприйнятливих осіб, то можна запобігти епідемії. Цей результат лежить в основі концепції загального імунітету, згідно якої профілактика епідемії може бути досягнена, якщо вакцинується більша частка населення. Якщо рівень вакцинації невеликий, то можна зменшити масштаби епідемії, але не запобігти їй.

В Російській імперії в 1889 році лікар-епідеміолог П. Д. Єнько розробив і опублікував модель розповсюдження інфекційного захворювання в дискретному часі, рівняння якої описують середнє значення чисельності груп, отриманих в моделі Ріда-Фроста. Про це стало відомо завдяки огляду Клауса Дітца і Дітера Шенцле, присвяченому історії застосування математичних моделей в епідеміології. Вчені в своїх працях обговорювали способи узагальнення моделі

Єнько на основі використання різних законів розподілу для числа контактів. Робота Єнько П. Д. В 1989 році була перевидана на англійську мову, а вчений був визнаний першим в історії, хто досліджував моделювання епідемій.

Роботи M.S. Bartlett по дослідженню стохастичної SIR-моделі в безперервному часі, опубліковані ним в 1949 році, дали поштовх розвитку стохастичних моделей епідеміологічних процесів. Крім того, великий внесок в розвиток застосування теорії випадкових процесів до моделювання епідемій внесли роботи N. Vaily (1970). Одна з перших просторових моделей розповсюдження епідемій була описана D. G. Kendall в 1957 році на основі рівнянь в частинних похідних. В той самий час M.S. Bartlett моделює розповсюдження епідемій на вузлах просторової структури на основі імітаційного моделювання. Це був новий напрям в науці, заснований на комп'ютерному моделюванні[15]. J. Fox и L. Elveback в 1971 році описали імітаційну, індивідуум-орієнтовану модель розповсюдження інфекції, яка не одразу була визнана світовим науковим співтовариством через недостатню кількість даних для налаштування індивідуум-орієнтованих моделей та недостатньою продуктивністю ЕОМ того часу [12].

З 80-х років минулого століття почали з'являтися математичні моделі оцінювання ефективності різних методів діагностування і лікування різних захворювань. Ці моделі розроблялися як на основі детермінованого, так і на основі ймовірнісного підходів.

Математичне представлення біологічних процесів забезпечує прозорість і точність по відношенню епідеміологічних припущень, дозволяючи перевірити розуміння епідеміології захворювання шляхом порівняння результатів моделі і закономірностей, що спостерігаються [18].

Поступово, на основі SIR-моделі була побудована ціла ієрархія математичних моделей розповсюдження інфекційних захворювань (SI, SIS, SEI, SIRS, SEIS, SEIR, SEIRS, MSIR, MSEIR, MSEIRS, LISEIR) [11, 23], які враховують велику кількість груп епідемічного процесу і параметрів, які впливають на перебіг епідемії.

Модель також може допомогти в прийнятті рішень, прогнозуючи важливі питання, такі як зміни в розповсюдженні захворювання.

Отже, важлива роль математичних моделей полягає в тому, що вони можуть попередити нас про недоліки епідеміології різних інфекційних захворювань та сформулювати мету та завдання для подальших досліджень.

1.3 Аналіз сучасних засобів створення програмного забезпечення

Невід’ємним елементом імітаційного моделювання є візуалізація досліджуваних процесів або явищ. Математична модель дає змогу не лише спрогнозувати зміну епідеміологічної ситуації, а й забезпечити проведення віртуального експерименту. Сьогодні існує безліч засобів для створення візуалізації. Для цього добре підходять платформи, що використовуються при розробці комп’ютерних ігор, адже симуляція подібна до гри.

Однією з найбільш популярних платформ є Pygame.

Pygame – бібліотека модулів для мови Python, створена для розробки 2D ігор.

Переваги:

- легко розпочати роботу та створити якусь візуалізацію;
- кросплатформенність;
- багато ігор з відкритим кодом доступні для роботи.

Мінуси:

- з точки зору продуктивності не можна масштабувати до дуже великих ігор;
- в основному підходить для 2D розробок, хоча 3D проекти можливі;
- важко поширювати як закрите джерело.

Pygame – бібліотека мови програмування python з відкритим кодом для створення мультимедійних додатків. Це кросплатформний набір модулів Python, призначений для написання відеоігор, який включає бібліотеки комп’ютерної графіки та звуку, призначені для використання з мовою програмування Python.

1.4 Постановка задачі та вимоги до розробки інформаційної системи

Метою кваліфікаційної роботи є аналіз існуючих моделей поширення інфекційних захворювань та розробка власної імітаційної моделі, проведення її валідації та верифікації за допомогою розробленої відповідної інформаційної системи. Виходячи з поставленої мети, в роботі необхідно вирішити наступні завдання:

- розглянути теоретичні аспекти поширення інфекційних захворювань;
- розглянути математичні методи, за допомогою яких можна описати характер поширення захворювань, що передаються повітряно-крапельним шляхом;
- розглянути технології, які можуть бути використані при розробці імітаційної моделі респіраторних захворювань і перспективи їх використання;
- застосувати SI модель для побудови математичної моделі, що описує перебіг гострих респіраторних захворювань;
- застосувати теорію ймовірностей для оптимізації і кращої апроксимації моделі;
- створити інформаційну систему для візуалізації процесу розповсюдження гострих респіраторних захворювань за запропонованою імітаційною моделлю;
- провести експерименти і їх аналіз для виявлення закономірностей.

2. Проектування інформаційної системи

2.1 Імітаційна модель розповсюдження респіраторних захворювань

Серед великої кількості захворювань, одними з найбільш критичних для населення є хвороби, що передаються від людини до людини повітряно-крапельними шляхами через швидкість їх поширення, масштаб, велику кількість хворих і померлих тощо.

До певного часу інфекції були головною причиною високої смертності та малої тривалості життя людини, вражаючи величезну кількість людей та великі території земної кулі.

В історії людства перша пандемія чуми датується серединою VI століття – «Юстиніанова чума» – призвела до загибелі близько 100 мільйонів, вбивши від 50 до 60% населення Європи. Друга пандемія чуми, названа Чорна смерть, виникла в середині XIV століття і стала причиною загибелі 30% населення Азії і до 50% жителів Європи. Третя пандемія чуми почалася у Китаї у XIX столітті і протягом кількох десятиліть охопила всю Землю.

Під час останньої епідемії краснухи в США (60-і роки XX століття), захворіло 12,5 мільйонів людей, більш ніж у двох тисяч людей розвинувся енцефаліт і більше 11 тисяч жінок були змушені перервати вагітність через ризик розвитку у дітей синдрому вродженої краснухи (СВК). Було народжено понад 20 тисяч дітей із СВК. При цьому мали глухоту понад 11 тисяч дітей, були сліпі від народження понад 3,5 тисячі дітей, розвинулася розумова відсталість майже у 2 тисяч малюків. Тільки широкомасштабна вакцинація останнього десятиліття змогла призвести до практично повної ліквідації краснухи та СВК у багатьох розвинених та окремих країнах, що розвиваються.

Два великі спалахи епідемічного паротиту були зареєстровані в США: у 2006 р. – понад 6,5 тисяч випадків серед студентів університетів Середнього заходу та у 2010 р. – понад 3,5 тисяч випадків серед старшокласників кількох шкіл, членів релігійної громади ортодоксальних євреїв-хасидів. Ризик формування шкоди, передусім «чоловічого» безпліддя, яким ускладнюються

середні і важкі форми епідемічного паротиту в 20 – 50% випадків, завданий цій громаді, досить великий.

В сучасності причиною епідемій та пандемій залишається вірус грипу. Епідемії грипу багатьом відомі, наприклад, «Іспанський грип» у 1918–1919 роках забрав життя 50-100 мільйонів людей; Азіатський грип у 1957 р. – близько 2 мільйонів осіб, Гонконзький грип у 1968 р. – близько 34 тисяч людей.

Найближче майбутнє людства, незважаючи на розвиток систем охорони здоров'я, появу нових засобів та способів діагностики, лікування та профілактики буде пов'язане з інфекційними хворобами. Їхньому поширенню сприятимуть збройні конфлікти, економічні кризи, глобальні міграційні процеси, зміни клімату тощо.

Сьогодні актуальною проблемою людства стала коронавірусна інфекція Covid-19, вона є лише частковим випадком більшості інфекційних захворювань, які можуть з'явитися найближчим часом.

Генеральний секретар Організації об'єднаних націй Антоніу Гутерреш вважає, що спалах нового коронавірусу продемонстрував, що міжнародне співтовариство, як і раніше, не здатне зупинити поширення захворювань, щоб запобігти глобальній пандемії. Крім того він попередив, що: «COVID-19 – аж ніяк не остання пандемія, з якою зіткнеться людство. Інфекційні захворювання, як і раніше, становлять явну і реальну загрозу для будь-якої країни».

У грудні 2021 року очільник ВООЗ Тедрос Адан Гебрейесус підтримав рішення спеціальної сесії Всесвітньої асамблеї охорони здоров'я розробити нову глобальну угоду про готовність до майбутніх пандемій.

Людство має бути готовим до появи нових захворювань, тому критично важливо знизити негативні явища, які можуть стати наслідком поширення інфекцій.

Попереджувальні заходи є ключовими у вирішенні цього завдання. Прогнозування динаміки розповсюдження захворювань дозволяє розробити і застосувати адекватні заходи протидії, забезпечити раціональне використання матеріальних і людських ресурсів. Усе більше дослідників звертаються до

математичних моделей для створення прогнозів інфекційних захворювань, так як вони допомагають найбільш вірно і точно вивчити зміни певних процесів, які відбуваються в соціумі.

Аналіз і математичне моделювання спалахів захворювань відіграють важливу роль у плануванні заходів охорони здоров'я у відповідь на спалахи інфекційних захворювань, епідемій та пандемій. Основна мета будь-якого такого моделювання – це визначення характеристик, динаміки та впливу пандемій, а також оцінки ефективності заходів у різних умовах. Тим не менш, слід враховувати, що основна інформація для органів, що приймають рішення надходить на щоденній основі не за результатами складного імітаційного моделювання, а з простого і статистичного аналізу, що проводиться в реальному часі, який базується на механістичних моделях передачі, що використовують доступні епідеміологічні та вірусологічні дані. Математичне моделювання може лише служити інструментом для обробки доступних даних та вказувати, які додаткові відомості можуть бути корисними при прийнятті управлінських рішень.

На початку дослідження варто розглянути шляхи вирішення досліджуваної проблеми та визначити, чи можливо розв'язати поставлену задачу на сучасному розвитку науки та технологій. Оскільки спроби побудови моделей захворювань застосовуються досить довгий час, сьогодні існує досить надійний математичний апарат, який може бути використано для опису узагальненого захворювання. Проте через постійні мутації та появу нових, незнаних досі хвороб або спалахи маловивчених захворювань, кожену модель необхідно адаптовувати під конкретний випадок, що є затратним як з фінансової точки зору, так і з погляду втрати найбільш дорогоцінного ресурсу – часу, що в умовах пандемій або епідемій може мати катастрофічні наслідки.

Загалом процес моделювання будь-якого процесу можна поділити на три етапи [10, с. 232]:

– розробка математичної моделі як системи формул та рівнянь на основі результатів емпіричного аналізу процесів, що відбуваються в системі;

– перевірка та коригування моделі, що включає визначення коефіцієнтів та початкові умови, рішення систем рівнянь та порівняння одержаних результатів з експериментальними даними;

– аналіз математичної моделі та її практичне значення щодо отримання нової інформації про об'єкт, що вивчається.

Моделювання є циклічним процесом. Це означає, що кожен наступний етап опирається на результат отриманий на попередньому етапі.

Модель також може допомогти в прийнятті рішень, прогнозуючи важливі питання, такі як зміни в розповсюдженні захворювання. Моделі передавання базуються на сучасному розумінні інфекційного процесу імунологічних реакцій, коли такі знання відсутні, можна зробити припущення відносно цих процесів. Однак, в таких випадках може бути декілька можливих механізмів, і відповідно декілька різних моделей, які можуть призвести до схожих закономірностей, так що не завжди можливо дізнатися про базові механізми шляхом порівняння результатів моделі. Але різні моделі, що призводять до подібних результатів в одному контексті, можуть не досягати цього в іншому. В таких випадках краще проводити подальші епідеміологічні та експериментальні дослідження, щоб знайти різні можливі механізми.

Сьогодні все більшої популярності набувають так звані методи імітаційного моделювання та агентного підходу. Сутність імітаційного моделювання у тому, що з досліджуваної системи (розвитку епідемії) будуються графічні діаграми зв'язків і глобальних впливів одних параметрів на інші в часі. Створена на основі цих діаграм модель імітується на комп'ютері. Такий вид моделювання дозволяє проникнути в суть того, що відбувається в системі та виявити причинно-наслідкові зв'язки між об'єктами та явищами. Метою агентного моделювання є отримання уявлення про ці глобальні правила, загальну поведінку системи, виходячи з припущень про індивідуальну, приватну поведінку її окремих активних об'єктів та взаємодію цих об'єктів у системі. Саме такий метод був застосований китайськими вченими для моделювання епідемії коронавірусу в Китаї [4].

Загалом, в імітаційному моделюванні існує кілька парадигм, які використовуються як основа, свого роду це каркас при побудові моделей. Виділяють, як правило, три основні системи поглядів: системна динаміка, дискретно-подійне моделювання, агентні моделі, що відрізняються не стільки областями застосування, як концепціями та підходами до вирішення проблеми. Інколи виділяють четвертий підхід – динамічні системи, що ґрунтуються на аналізі моделей, що моделюються, формалізованих у вигляді систем алгебро-диференціальних рівнянь. Вибір підходу до побудови імітаційної моделі визначається метою моделювання та вибирається відповідно до цієї мети рівнем абстракції при описі досліджуваної системи або процесу.

Умовно виділяють три рівні абстракції: оперативний (низький), тактичний (середній) та стратегічний (високий). Низький рівень абстракції відповідає деталізованому обліку у моделі індивідуальних характеристик та логіки поведінки окремих об'єктів. Такі моделі можуть використовуватися, наприклад, при вирішенні завдань, пов'язаних з організацією пішохідного та транспортного руху, або з розробкою заходів протидії розповсюдженню епідемії [2].

На середньому рівні абстракції індивідуальні об'єкти знеособлюються, розглядаються потоки, затримки, пропускні можливості, ємності, час обслуговування. Характерні моделі, що відповідають цьому рівню абстракції – моделі бізнес-процесів, ланцюжків поставок, роботи сервісних центрів.

Моделі, що відповідають найвищому рівню абстракції, оперують поняттями тенденцій, впливів, зворотних зв'язків. Приклади таких моделей – великомасштабні соціально-економічні моделі, моделі ринкової рівноваги, фондового ринку, екологічні моделі.

Проаналізувавши рівні абстракцій, визначаємо, що для створення імітаційної моделі поширення респіраторних захворювань доцільно застосувати оперативний (низький) рівень абстракції.

Підхід, який називається системною динамікою, відповідає високому рівню абстракції. Відповідно до цього підходу, запропонованого в 1950-х рр. американським вченим Джеєм Форрестером, структура і поведінка системи, що

моделюється, представляються як безліч взаємодіючих позитивних та негативних зворотних зв'язків та затримок.

Дискретно-подійне моделювання – підхід, що відповідає низькому та середньому рівням абстракції. Родоначальником цього підходу, заснованого на концепції заявок (вимог), ресурсів та потокових діаграм, вважається Джефрі Гордон, який розробив у 1960-х роках моделювання GPSS. Термін «дискретно-подійне моделювання» історично закріпився за моделюванням систем обслуговування потоків об'єктів певної природи: клієнтів банку, автомобілів на заправній станції, телефонних викликів, пацієнтів у поліклініках тощо.

Третя парадигма в імітаційному моделюванні – агентне моделювання. Цей підхід вважається відносно новим, на відміну від системної динаміки і дискретно-подійного моделювання, що вже склалися. Поява ідеї агентного моделювання пов'язують із роботами Томаса Шеллінга щодо дослідження проблеми ненавмисних соціальних наслідків нескоординованих дій окремих людей, зокрема расової сегрегації в американських містах. У запропонованій Шеллінгом моделі «сусідства, що самоформується» розглядалися два класи агентів: чорні та білі жителі, які проживають в одному районі по сусідству один з одним. Для проведення експериментів використовувався аналог шахівниці (64 поля).

Відповідно до заданих правил у моделі Шеллінга цілком можлива рівновага – комфортне співіснування незгрупованих чорних та білих агентів (рис. 2.1). Якщо випадково кілька агентів (наприклад, три) видаляються з поля (звільняються клітини C4, D3 і E2), то рівновага порушується (рис. 2.2). Сусіди того ж кольору, що знаходяться поблизу від звільнених клітин, що й віддалені, усвідомлюють, що баланс порушений, і вирішують перебраться на клітини, де вони будуть в оточенні агентів того ж кольору (наприклад, на клітини A і 7/8). Це запускає механізм зворотного зв'язку, що спонукає інших агентів також пристосовуватися до ситуації, що змінилася.

	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	<i>F</i>	<i>G</i>	<i>H</i>
1		■	○	■	○	■	○	
2	■	○	■	○		○	■	○
3	○	■	○		○	■	○	■
4	■	○		○	■	○	■	○
5	○	■	○	■	○	■	○	■
6	■	○	■	○	■	○	■	○
7	○	■	○	■	○	■	○	■
8		○	■	○	■	○	■	

Рисунок 2.1 – Рівновага у шаховій моделі Шеллінга

	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	<i>F</i>	<i>G</i>	<i>H</i>
1	■	■	○		○		○	
2	■	○		○		○		○
3	○	■	○		○		○	
4	■	○	■	○		○		○
5	○	■	○	■	○		○	
6	■	○	■	○	■	○		○
7	○	■	○	■	○	■	○	■
8		○	■	○	■	○	■	■

Рисунок 2.2 – Порушення рівноваги у шаховій моделі Шеллінга

Одне з найбільш повних визначень агента наводять Ч. Макал і М. Норт, відповідно до нього агент повинен мати наступні характеристики:

– агент ідентифікується, тобто являє собою кінцевого індивідуума з набором певних характеристик і правил, що визначають його поведінку та

правила прийняття рішень, він автономний і може незалежно діяти та приймати рішення щодо взаємодії з іншими агентами;

– агент знаходиться у певному середовищі, що дозволяє йому взаємодіяти з іншими агентами, агент може контактувати з іншими агентами за певних умов та відповідати на контакт;

– агент має певну мету (необов'язково метою є максимізація блага), що впливає на його поведінку;

– агент гнучкий і має здатність самонавчання з часом на основі власного досвіду. У ряді випадків агент може навіть змінювати правила поведінки на основі отриманого досвіду.

Отже, даний підхід доцільно застосовувати при імітаційному моделюванні поширення респіраторних захворювань. Таким чином люди можуть бути представлені у вигляді агентів, яким притаманні певні властивості (рис. 2.3). Агенти можуть взаємодіяти між собою та з навколишнім середовищем.

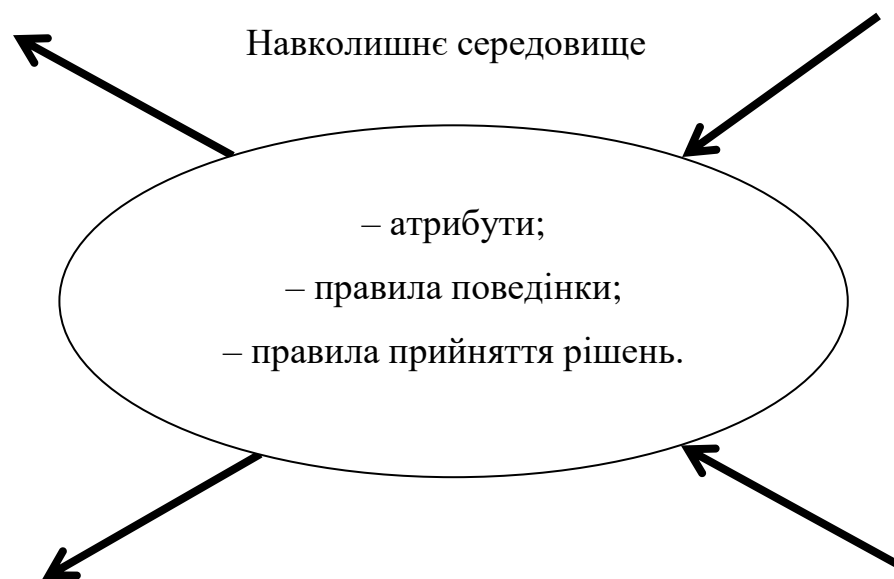


Рисунок 2.3 – Характеристики агента

Таким чином атрибутами агента є розмір, стан (здоровий або хворий) – позначається кольором (синій та червоний відповідно). Правила поведінки агента визначаються швидкістю та напрямком їх руху. Дані параметри доцільно генерувати псевдовипадковим чином. Правила прийняття рішень базуються на

зміні стану агента зі «здоровий» на «хворий». Така зміна можлива лише під час взаємодії агентів з різними станами. Взаємодія агентів з навколишнім середовищем базується на їх переміщенні по заданій площі, неможливості виходу за її межі або проходу крізь стіни та зачинені двері, внаслідок зміни траєкторії руху.

Наступним етапом є визначення моделі взаємодії агентів. Доцільно застосовувати модель, побудовану на основі SIR моделі. Сьогодні створено значну кількість таких моделей, в тому числі і в Україні.

Робоча група з математичного моделювання проблем, пов'язаних з епідемією коронавірусу SARS-CoV-2 в Україні (створена розпорядженням Президії НАН України від 3 квітня 2020 р. № 118) розробила модель SEIR-U. З початку квітня 2020 р. міжвідомчою Робочою групою представників НАН України, КНУ імені Тараса Шевченка та НАМН України, з урахуванням світового досвіду математичного моделювання розвитку епідемії COVID-19, на основі статистичних даних про динаміку епідемії в Україні та країнах Європи створювалась і тестувалася математична модель SEIR-U. Модель SEIR-U, як інші моделі класу SEIR, розраховує балансові відношення для чотирьох компартментів основних категорій населення під час епідемії: S – сприйнятливі до захворювання, E – інфіковані, але без проявів хвороби, I – інфіковані з підтвердженою хворобою, R – ті, що вже не можуть захворіти, бо мають імунітет до хвороби чи померли. Така схема може нарощуватись введенням додаткових компартментів: наприклад, частина категорії S, яка на суворому карантині не хворіє, підкатегорії R, яка отримала вакцинації, підкатегорії I, яка не має симптомів і не потребує госпіталізації, і підкатегорія з госпіталізованих, яка потребує вентиляції легень, та інші підкатегорії. В силу гнучкості такої структури моделі SEIR адаптують для специфічних особливостей різних інфекційних хвороб. Поточна реалізація моделі SEIR-U дозволяє враховувати наявність безсимптомних хворих, має 3 рівні складності протікання хвороби для хворих із симптомами, дозволяє обчислювати кількість пацієнтів, що знаходяться на госпіталізації.

Математичний апарат SEIR моделей – чисельний розв’язок систем детермінованих чи стохастичних звичайних диференціальних рівнянь. Моделі, що базуються на диференціальних рівняннях, посідають особливе місце серед існуючих моделей поширення інфекційних захворювань. Диференціальні динамічні моделі дають змогу не тільки проводити прогнозування розвитку ситуації, але й оцінити різні епідеміологічні параметри, а також ефективність тих чи інших протиінфекційних заходів.

Проте, виходячи з поставлених завдань недоцільно застосовувати більш складні моделі, оскільки досліджується лише поширення захворювання, а не його перебіг впродовж тривалого часу. Крім того, розглядається лише невелика частина людей (популяція). Дане дослідження націлене на створення безпечних умов в окремо взятому приміщенні, наприклад школа, корпус університету, або його частина, зал для проведення масових заходів, конкурсів, виставок тощо. Для дослідження більш великих популяцій на рівні загальноміському, загальнодержавному, чи навіть світовому безумовно слід надати перевагу більш складним моделям.

Таким чином, взаємодію агентів доцільно моделювати на основі SI моделі.

Відповідно до поставленої задачі, в подальшому детально розглядатимемо SI модель.

Методи математичної епідеміології стосуються здебільшого, так званої, «кількісної теорії», тобто знаходження чисельних розв’язків розроблених математичних моделей з метою визначення епідемічних порогів та прогнозування перебігу захворювань [20]. У той же час залишаються недостатньо вивченими задачі якісного аналізу епідеміологічних процесів. Таким чином, практично всі методи побудови математичних моделей для епідеміології мають багато спільного, що в свою чергу відображається і на самих моделях, які мають більше спільних властивостей, ніж відмінностей, але залишаються недостатньо вивченими задачі якісного аналізу епідеміологічних процесів, які дають відповіді на запитання про стійкість та стабілізацію епідемії [13].

Розглянемо можливість зараження при наближенні двох людей на певну відстань L . Ця відстань буде залежати від виду вірусу та його життєстійкості. Більшість вірусів швидко гинуть поза організмом, проте деякі можуть виживати протягом декількох днів і навіть місяців. Повітряно-крапельний шлях передачі передбачає вдихання розпилених у повітрі частинок слизу внаслідок чхання чи кашлю інфікованим у радіусі 0,9 метра. При дрібнодисперсному розпиленні з виникненням частинок діаметром менше 10 мкм радіус може збільшитись до 1,8 метра і більше. Таким чином ймовірність інфікування при дотриманні дистанції більше двох метрів дуже незначна і нею можна знехтувати. Тому доцільно приймати L рівним 2 м. В той же час навіть при контакті з хворим ймовірність захворіти в здорової людини, у якої відсутній імунітет не рівний 100%. Для визначення, захворіє особа чи ні використовується теорія ймовірності.

Розглянемо подію A – «людина захворіє». Ймовірність того, що подія A справдиться $P(A)$. Числове значення такої ймовірності визначається експериментальним шляхом або на основі статистичних даних. Проте ці дані не можна вважати абсолютно точними. Тому при моделюванні поширення ГРВІ варто розглядати діапазон ймовірностей від мінімальної ймовірності зараження $P_{min}(A)$ до максимальної $P_{max}(A)$. Отже значення $P(A)$ обирається випадковим чином з діапазону $[P_{min}(A); P_{max}(A)]$.

Припустимо, що на певному обмеженому просторі перебуває деяка кількість людей протягом досить тривалого часу. Якщо серед цієї групи перебуватиме хоча б одна інфікована особа, яка безпосередньо контактує з іншими, існує ризик сплеску захворювання. Отже, кожна особа отримує статус: «вразливий» або «хворий».

Для прогнозування поширення захворювання необхідно визначити кількість контактів між людьми за одиницю часу. Слід зауважити, що на початковому етапі більшість контактів будуть відбуватися між здоровими людьми. Проте, з часом кількість хворих буде зростати. В свою чергу зростання кількості хворих у вибірці, що досліджується, призведе до збільшення кількості контактів хворих та здорових, внаслідок чого кількість хворих різко зростатиме.

Позначимо залежність кількості здорових людей, які є вразливими до хвороби від часу. Аналогічно залежність кількості інфікованих від часу. Оскільки популяція, що розглядається є невеликою, то швидкість зміни частки вразливих пропорційна з коефіцієнтом пропорційності, що чисельно рівний значенню $P(A)$. Звідси отримуємо рівняння (1) залежності кількості вразливих від часу:

$$\frac{d\tilde{S}(t)}{dt} = -\beta \cdot \tilde{S}(t) \cdot \tilde{I}(t). \quad (2.1)$$

Знак мінус вказує на зменшення кількості вразливих, оскільки вони змінюють свій статус на «хворий» і не можуть інфікуватися повторно, поки через певний проміжок часу знову не стануть вразливими. В той же час кількість хворих зростає. Таким чином рівняння (2) залежності кількості хворих від часу:

$$\frac{d\tilde{I}(t)}{dt} = \beta \cdot \tilde{S}(t) \cdot \tilde{I}(t). \quad (2.2)$$

Об'єднавши рівняння (1) і (2) отримаємо модель для невеликої популяції, яка задана системою рівнянь (3):

$$\begin{cases} \frac{d\tilde{S}(t)}{dt} = -\beta \cdot \tilde{S}(t) \cdot \tilde{I}(t), \\ \frac{d\tilde{I}(t)}{dt} = \beta \cdot \tilde{S}(t) \cdot \tilde{I}(t). \end{cases} \quad (2.3)$$

Де:

– залежність кількості здорових людей, які є вразливими до хвороби від часу;

– залежність кількості хворих людей від часу;

– ймовірність інфікування вразливої людини при контакті з хворим.

Таким чином, математичне представлення біологічних процесів забезпечує прозорість і точність по відношенню до епідеміологічних припущень, дозволяючи перевірити розуміння епідеміології захворювання шляхом порівняння результатів моделі і закономірностей, що спостерігаються [18]. Модель також може допомогти в прийнятті рішень, прогнозуючи важливі питання, такі як зміни в розповсюдженні захворювання.

2.2. Функціональна модель системи

Програмний продукт повинен виконувати наступні функції: проведення дослідження згідно заданих параметрів методами математичного моделювання, перегляд симуляції, на основі імітаційної моделі, запис отриманих результатів у файл, перегляд результатів проведених раніше досліджень, а також передбачати можливість додавання нових параметрів для проведення досліджень: нових конфігурацій приміщень та вхідних параметрів. Оскільки користувачеві необхідним є виконання лише частини функцій, доцільно розділити функції, виконувані програмним продуктом між різними групами користувачів (рис.2.3).



Рисунок 2.4 – Функції користувача та адміністратора

Таким чином, адміністратору надається змога коригувати відповідні вхідні дані, оскільки передбачається, що адміністратором є людина, у якої є мінімальні знання у сфері епідеміології, в той час, як пересічні користувачі можуть використовувати програмний засіб без будь-якої підготовки. В той же час, розширені права адміністратора дозволять експертам виконувати дослідження в більш широкому діапазоні.

Програмний продукт надає можливість користувачам провести дослідження, виходячи із заданих параметрів, за необхідності виконати оптимізацію вхідних параметрів та переглянути відповідні симуляції (рис. 2.5).

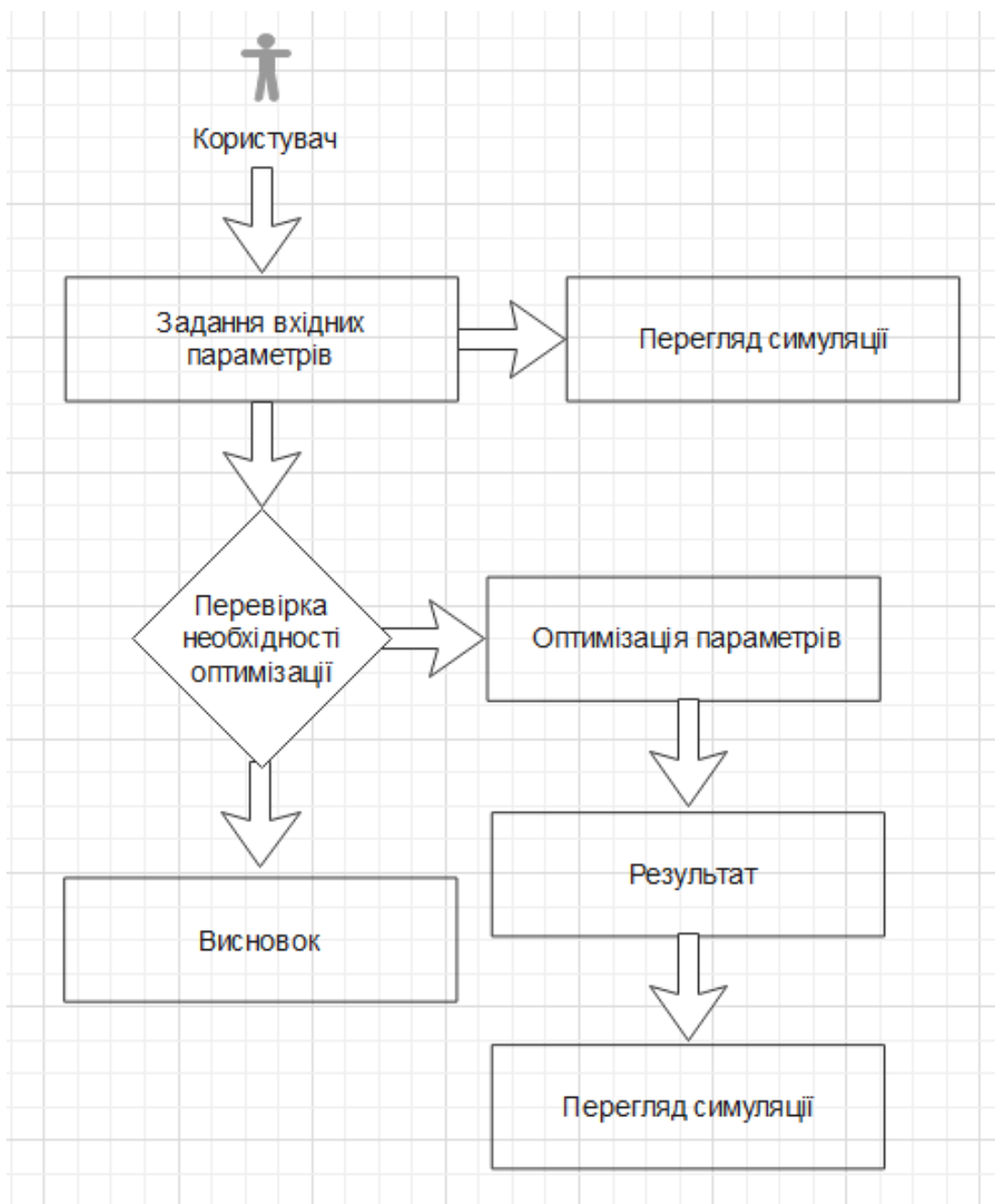


Рисунок 2.5 – Послідовність дій користувача

2.3 Інформаційна структура системи

Інформаційна система призначена для обробки вхідних даних та формування на їх основі висновку, щодо безпечності перебування певної кількості людей на певній території. Крім того, варто передбачити можливість оптимізації вхідних даних.

Система створює прогноз на основі SI моделі. Симуляція дає змогу наочно переглянути поширення захворювання.

Відповідно до обраних математичних методів вхідними даними для системи будуть: конфігурація приміщення, кількість осіб, що одночасно перебувають в приміщенні, початкова кількість інфікованих та діапазон ймовірності інфікування.

Висновок про безпеку перебування людей у приміщенні варто робити виходячи з кількості інфікованих через деякий, досить тривалий проміжок часу. Якщо отриманий результат буде меншим за епідеміологічний поріг, система робить висновок про безпечність змодельованої ситуації, в іншому випадку буде запропоновано оптимізувати вхідні дані. Епідеміологічним порогом прийнято вважати інфікування 5% членів популяції. Оскільки всі дані, крім початкової кількості людей в приміщенні є статичним, оптимізацію проводимо зменшуючи кількість контактів та збільшуючи дистанцію між людьми. Для цього поступово зменшуємо розмір популяції та проводимо розрахунки. Узагальнена схема роботи інформаційної системи зображена на рис. 2.6.

Результат моделювання можна переглянути наочно за допомогою проведення симуляції. Параметри, які використовує система зведені в таблицю 2.1.

Розроблена інформаційна система в повній мірі описує поставлену задачу і може використовуватись при розробці програмного забезпечення.

Таблиця 2.1. Параметри системи

id	Параметр	Тип даних
1	Номер дослідження	int
2	Конфігурація приміщення	int
3	Чисельність популяції	int
4	Кількість інфікованих на початку дослідження	int
5	Мінімальна ймовірність інфікування	double
6	Максимальна ймовірність інфікування	double
7	Тривалість дослідження	double
8	Висновок	string

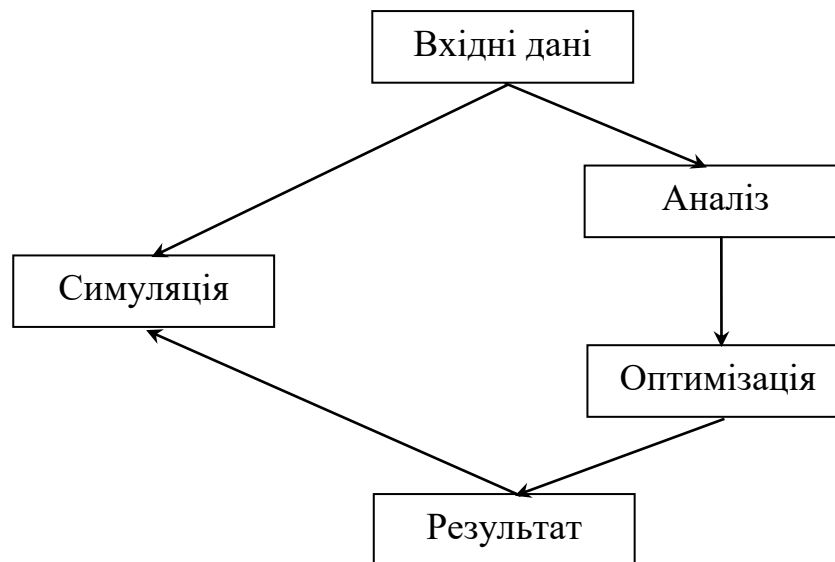


Рисунок 2.6 – Схема роботи програмного продукту

2.3 Вибір засобів розробки інформаційної системи

Відповідно до поставленого завдання необхідно розробити десктопний додаток для прогнозування поширення респіраторних захворювань з реалізацією можливості збереження даних та візуалізацією процесу поширення респіраторних захворювань.

2.3.1 Вибір мови програмування

Сьогодні існує безліч мов програмування, що використовуються для розв'язання різноманітних задач. Проте кожне завдання є унікальним, і задля створення актуального програмного продукту необхідно проаналізувати переваги та недоліки мов програмування та обрати найбільш доцільну.

Відповідно до обраної платформи PyGame, обираємо мову програмування Python.

Сьогодні, в час, коли обчислювальні потужності персональних комп'ютерів і серверів стали досить високими, з'явився величезний попит на інтерпретовані мови програмування. Адже, крім запуску самої програми, необхідний запуск інтерпретатора, що потребує додаткових ресурсів. Саме такою мовою є Python.

Переваги мови програмування Python наступні:

- Гнучкість – це, напевно, основна перевага мови, оскільки завдяки своїй гнучкості мова набула популярності серед багатьох розробників.

- Розширюваність – один зі слоганів мови звучить як – Just Import! – що повністю пояснює, наскільки мову можна розширити та було розширено за останні роки. Існують бібліотеки та фреймворки під будь-який тип завдань та потреб. Також величезним плюсом є можливість використання C-коду з Python.

- Простота синтаксиса.

- Інтерпретованість. Інтерпретатор Python існує для всіх популярних платформ і за замовчуванням входить до більшості дистрибутивів Linux, а отже є на більшості серверів.

- PEP – єдиний стандарт для написання коду, що робить код підтримуваним та читабельним навіть при переході від одного програміста до іншого. Це підтримує популярність Python.

- Open Source – код інтерпретатора Python є відкритим, що дозволяє будь-кому, хто зацікавлений у розвитку мови взяти участь у її розробці та покращити

її. Якщо дивитися деталі релізу однієї з версій мови, можна помітити, що великі частини нового функціоналу реалізовані сторонніми розробниками.

– Ком'юніті – навколо Python утворилося досить дружнє ком'юніті, яке готове прийти на допомогу будь-якому початківцю або вмілому розробнику і допомогти розібратися в його проблемі.

Проте є й недоліки:

– Продуктивність. Більшість розробників, та й сам автор мови, сходяться на думці, що Python не настільки швидкий, наскільки хотілося б. Це пов'язано з тим, що Python інтерпретується мова. Але навіть у порівнянні з іншими мовами, що інтерпретуються, помітно, що Python програє в продуктивності. Але це легко можна нівелювати за допомогою C реалізацій тієї чи іншої проблемної ділянки коду. В умовах сьгоднішніх потужностей це не дуже помітно.

– Синтаксис буде незвичним і трохи дивним при переході з інших мов програмування.

– Динамічна типізація. Через динамічну типізацію Python споживає більше ресурсів, ніж міг би, але це часто компенсується внутрішнім кешуванням.

– Global Interpreter Lock. На даний момент це є основною проблемою продуктивності Python, а також цим обумовлена погана реалізація багатопоточності. Код GIL не змінювався з першої версії мови. Це явно свідчить про те, що він застарів.

Виходячи з вищепереліченого, можна стверджувати, що всі недоліки мови повністю нівелюються її перевагами, які набагато вагоміші в сьгоднішніх реаліях. Крім того, мова розвивається. Всі недоліки можуть бути або скорочені, або усунуті зовсім. Python допомагає вирішити величезний спектр завдань.

2.3.2 Вибір додаткових засобів розробки

Найбільш поширеними є бібліотеки tkinter та wxPython.

Переваги tkinter:

– є частиною python, нічого додатково завантажувати не потрібно;

- простий синтаксис. Текстовий віджет потужний і з ним легко працювати;
- використовує рідні віджети на mac та windows. Tk - це надійний інструмент з невеликою кількістю кросплатформних ідіосинкразій;
- механізм прив'язки Tkinter набагато кращий, ніж wxPython, він більш гнучкий і послідовний;
- три менеджери геометрії Tkinter - pack, place і grid – набагато потужніші й прості у використанні, ніж сайзер wxPython.

Недоліки tkinter:

- використовує мотивоподібні віджети на боксах *nix, які вважаються непривабливими (хоча деякі ретельні налаштування можуть зробити графічні інтерфейси цілком прийнятними);
- віджети Tkinter у своїй основі не є об'єктами python, tkinter надає обгортку навколо реальних віджетів tk, що іноді може призводити до появи повідомлень про помилки;
- існує дуже слабка підтримка друку (в текстовому віджеті немає вбудованої підтримки друку).

Виходячи з дослідження предметної області можна зробити висновок, що для досліджуваної задачі оптимальним буде використання tkinter у поєднанні з PyGame, як з точки зору зручності програмування, так і доцільного використання ресурсів.

2.3.3 Вибір середовища збереження та керування даними

Відповідно до поставленої задачі немає необхідності у створенні великої кількості таблиць, тому зберігати інформацію доцільно у файлі.

JSON (JavaScript Object Notation) – це формат тексту для обміну даними, який базується на мові програмування (JavaScript) і в основному використовується тільки з цією мовою. Як і інші текстові формати, легкий та простий у розумінні.

Навіть не зважаючи на виникнення від JavaScript, цей формат тексту є незалежним і в повніше може застосовуватися практично з різноманітними мовами програмування. Вже не нове, що для багатьох мов є готовий код, який служить для обробки та створення даних у форматі JSON.

JSON має розширення (.json) та тип формату Data interchange. Також стандартизований RFC 4627.

У порівнянні з XML (мова розмітки), JSON більш короткий і зручний, що може бути найкращим варіантом для серіалізації складних структур.

– Розширюваність. Збільшити дані легше, ніж XML. Така функція дозволяє зменшити кількість зв'язків між постачальником та адресатом даних. У AJAX, скрипти з боку клієнта повинні бути досить інваріантними в порівнянні з сумісностями зміни в даних.

– Налаштування та налагодження помилок. Даний аспект безпосередньо відноситься однаково до серверної частини програми, і до клієнтської. Потрібно переконатися, що на сервері відомості грамотно створені та коректні. З боку клієнта ж, має бути налагодження помилок у вирішенні.

За допомогою JSON ця проблема є ручною, тим самим може вимагати перевірку того, що в результаті відповіді є правильні визначення.

– Безпека. JSON може бути причиною виклику проблем безпеки. Це говорить про наступне: як тільки відбувається вставка викликів функцій паралельно з матеріалами в JSON і користуєтеся eval() для формування рішення, в подальшому створюється довільний код, який практично вже може бути загрозою безпеці.

Отже, аналізуючи з вищевикладений матеріал, можна зробити висновок про доцільність використання JSON.

2.3.4 Візуалізація процесу розповсюдження гострих респіраторних захворювань

Візуалізація процесу розповсюдження гострих респіраторних захворювань відбувається за допомогою платформи PyGame. Засоби бібліотеки допомагають створити план приміщення, шляхом відтворення на екрані відрізків заданої довжини, що імітують стіни чорного кольору та закритих дверей у приміщення – червоного кольору. Людей доцільно позначати кругами з діаметром 15 пікселів (рис. 2.7). За необхідності можна створити умови дотримання соціальної дистанції. Для цього необхідно зменшити діаметр кругів, що позначають людей (рис. 2.8). На початку симуляції об'єкти, що позначають людей розміщуються випадковим чином на площині, що імітує приміщення. «Вразливі» об'єкти позначаємо синім кольором.

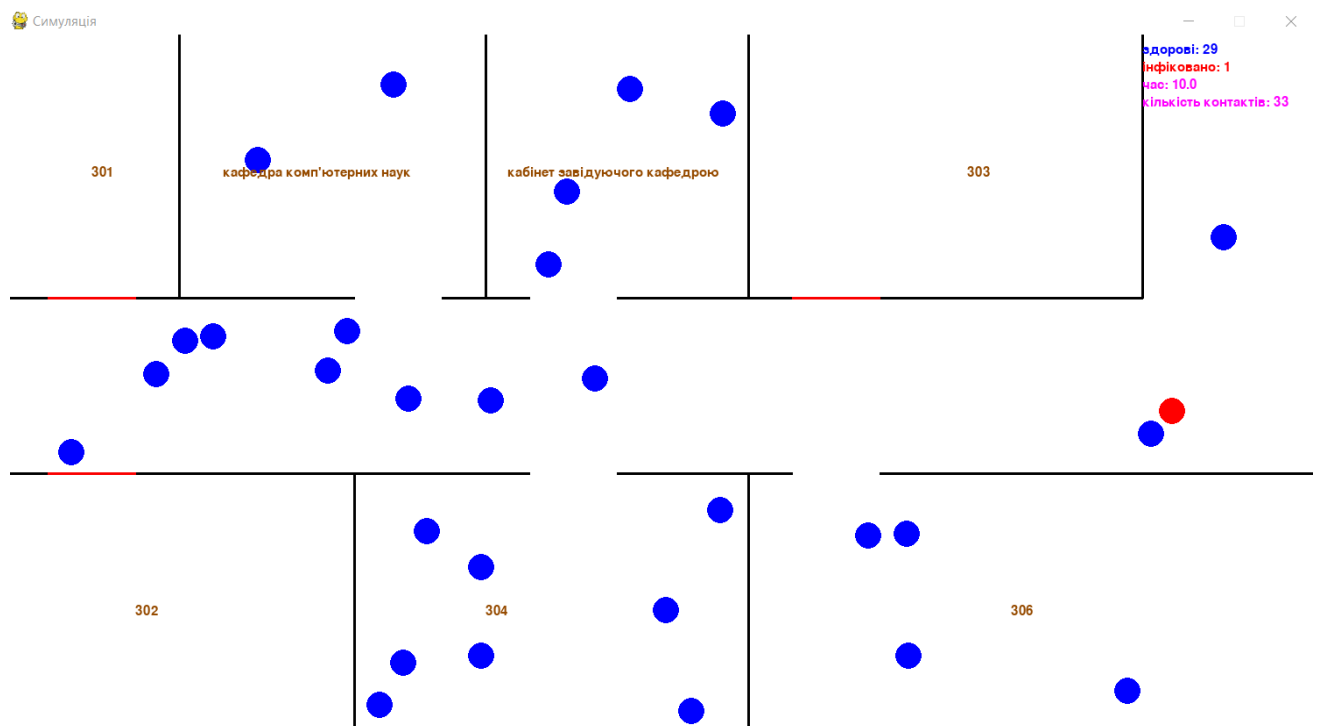


Рисунок 2.7 – Симуляція за звичайних умов

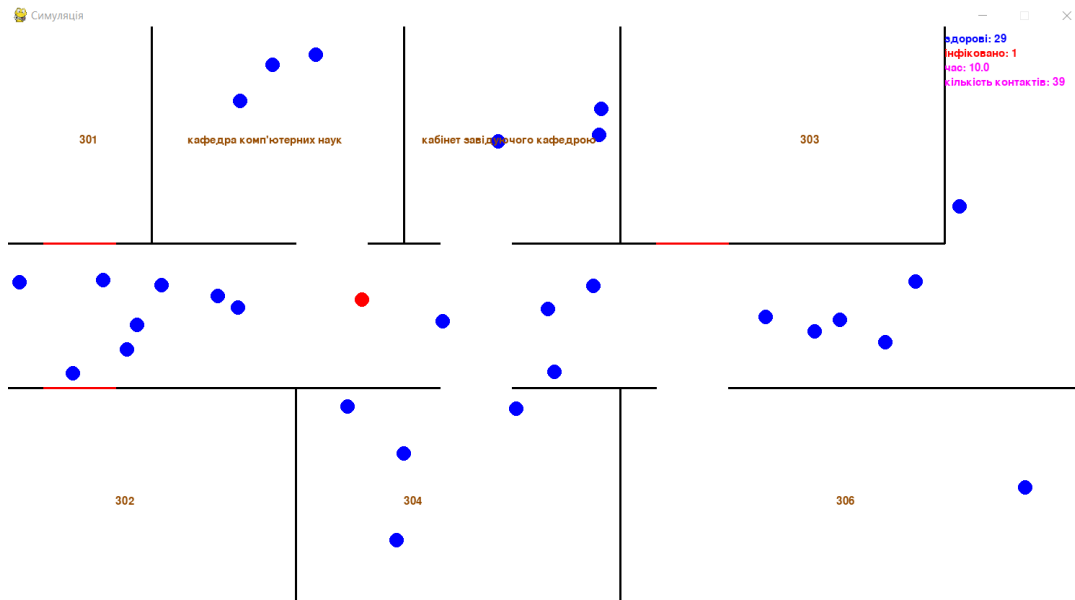


Рисунок 2.8 – Симуляція за умови дотримання соціальної дистанції

Кожен об'єкт починає рух у випадковому напрямку з початком симуляції. Серед об'єктів обирається задане користувачем число «хворих». Такі об'єкти позначаємо червоним кольором. Під час руху об'єктів відбувається перевірка зіткнень об'єктів зі стінами та зачиненими дверима (за наявності), а також між собою. Після таких зіткнень об'єкти змінюють свою швидкість та напрям руху. У випадку наближення двох об'єктів на відстань, що становить менше двох радіусів об'єкта, відбувається перевірка статусу об'єктів. Якщо вони мають різний статус: один «хворий», інший – «здоровий» відбувається процес «інфікування». Відповідно до встановлених користувачем значень, за допомогою теорії ймовірності, визначається, чи «захворіє» «здоровий» об'єкт. Симуляція припиняється, якщо всі об'єкти «захворіли» або час симуляції перевищив заданий.

3 Програмна реалізація інформаційної системи

3.1 Структура та функціональне призначення складових системи

Програмний продукт складається з п'яти модулів – «Графічний інтерфейс», «Модуль роботи з середовищем керування даними», «Модуль оптимізації параметрів», «Модуль опрацювання результатів» і «Модуль симуляції». Кожен з модулів має своє функціональне призначення. Поєднання усіх модулів забезпечує роботу системи. В той же час некоректна робота одного з модулів може порушити роботу додатка в цілому. Тому необхідно забезпечити їх надійну та коректну роботу, а також взаємодію між модулями.

В архітектурі системи виділено 5 модулів. Кожен з модулів займається конкретною задачею. Отже, Система – це взаємодія цих п'яти модулів: «Графічний інтерфейс», «Модуль роботи з середовищем керування даними», «Модуль оптимізації параметрів», «Модуль опрацювання результатів» і «Модуль симуляції».

1. «Графічний інтерфейс»

Модуль включає дизайн додатка.

2. Модуль роботи з середовищем керування даними

Здійснює підключення до файлу, де зберігається інформація та здійснює необхідні операції з даними.

3. Модуль оптимізації параметрів – здійснює підбір оптимальних параметрів, згідно поставленої задачі.

4. Модуль опрацювання результатів – здійснює опрацювання отриманих результатів та формує висновок, щодо проведеного дослідження.

5. Модуль симуляції – здійснює наочне представлення результатів дослідження, використовуючи імітаційну модель.

Взаємодія модулів показана на рисунку 3.1.

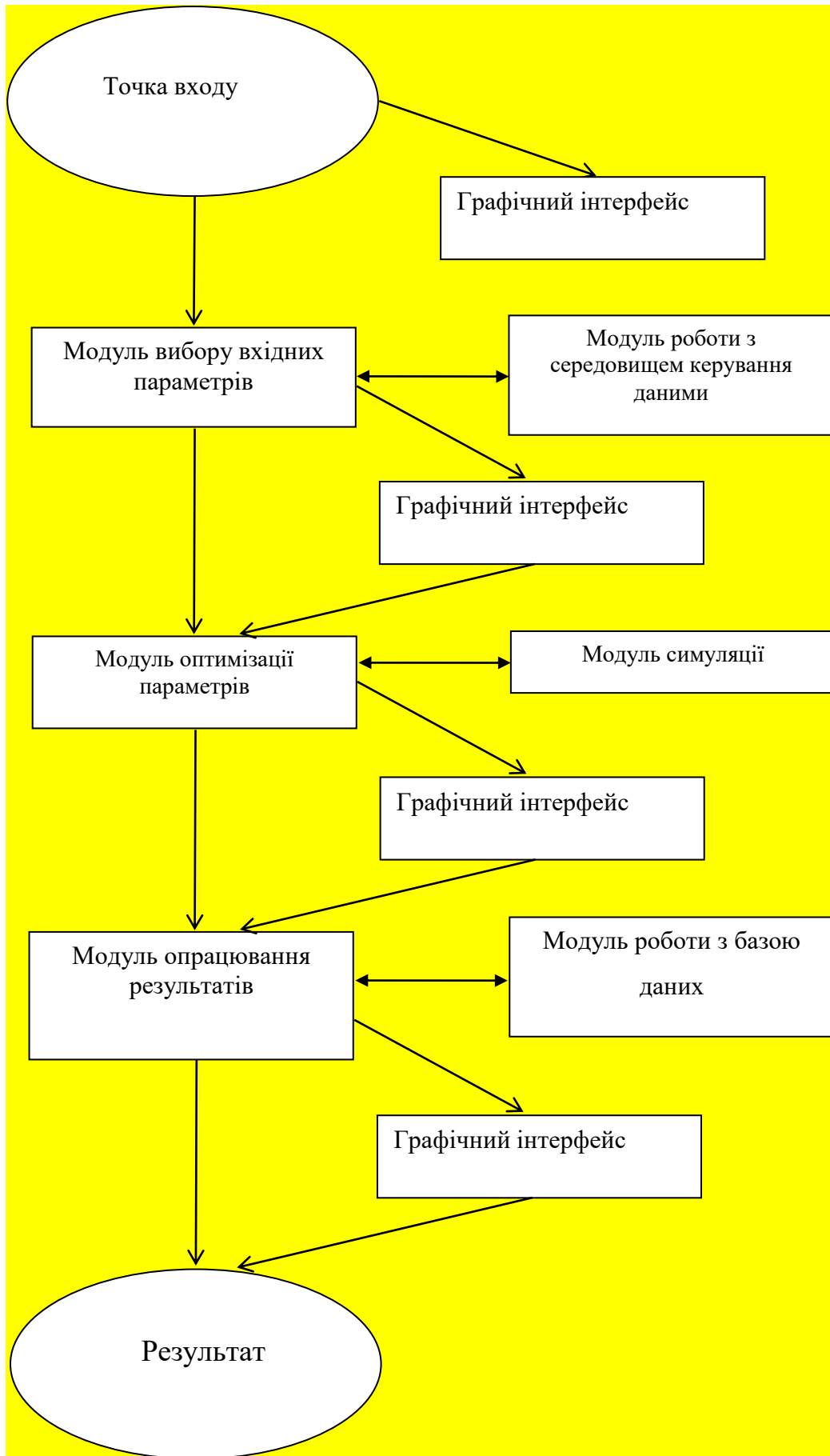


Рисунок 3.1 – Взаємодія модулів інформаційної системи

3.2 Особливості реалізації складових системи

Програмний продукт виконує прогнозування динаміки поширення респіраторних інфекцій. Виходячи з вхідних параметрів система оцінює тип інфекції, конфігурацію приміщення та розмір популяції. Застосування математичної моделі дає змогу визначити кількість хворих у популяції через певний проміжок часу, використовуючи ітераційний процес. Таким чином для різних типів приміщень спостерігається різна кількість контактів. Аналогічно, на це впливає величина популяції. Перевищенням епідеміологічного порогу вважається така кількість хворих, що перевищує 5% розміру популяції. На основі проведених розрахунків програмний продукт формує висновок про можливу небезпеку сплеску захворювання.

Імітаційна модель дає змогу наочно пересвідчитись у правильності прогнозу математичної моделі. Слід зауважити, що імітаційна та математична модель функціонують незалежно одна від одної, тому результати досліджень можуть мати відхилення у межах допустимих похибок.

3.3 Інструкція користувача

При завантаженні програмного продукту користувачеві пропонується задати вхідні параметри для проведення дослідження. Конфігурація приміщення обирається зі списку заданих, аналогічно обирається діапазон ймовірності інфікування (рис. 3.2). Користувач вводить величину вибірки та кількість інфікованих на початковому етапі. Слід зауважити, що ці числа мають бути цілими та додатними.

Налаштування

Місце для дослідження:

- Відкритий простір
- ХНУ, 3 корпус, 3 поверх, праве крило

Розмір вибірки:

Початкова кількість інфікованих:

Ймовірність інфікування: 0.1-0.3 0.3-0.5 0.5-0.7

Рисунок 3.2 – Вибір вхідних параметрів

В іншому випадку програма видасть повідомлення про помилку (рис. 3.3). Також кількість інфікованих не може перевищувати розмір популяції. Крім того, кількість інфікованих осіб на початковому етапі слід задавати із врахуванням тенденцій, але їх велика кількість може призвести до відсутності вирішення поставленої задачі, оскільки епідеміологічний поріг в будь-якому випадку буде перевищено.

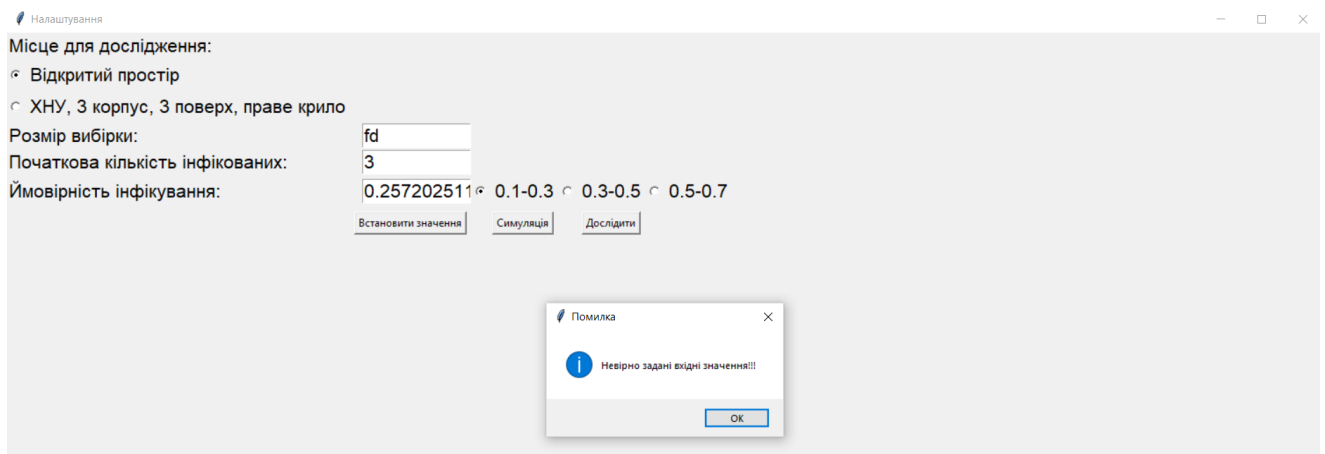


Рисунок 3.3 – Повідомлення про невірно задані параметри

Після проведення розрахунку та створення прогнозу користувачеві буде виведена відповідна інформація. Всього існує три типи висновків:

1. Задані параметри задовольняють усім вимогам безпеки. Найоптимальніший варіант, за якого подальші дослідження не мають сенсу (рис. 3.4).

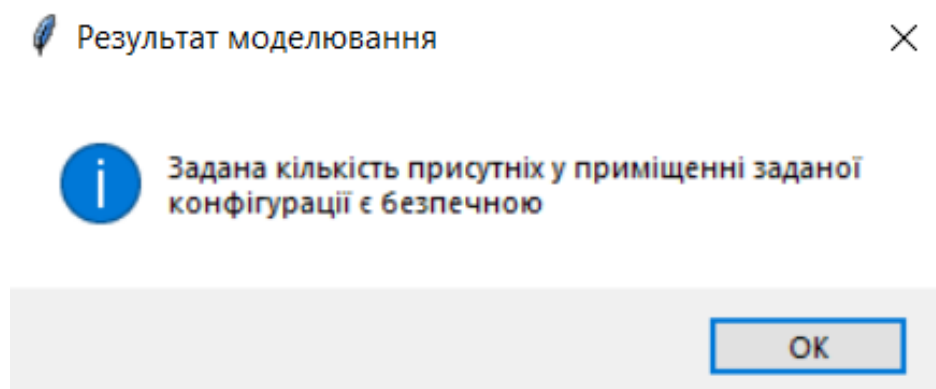


Рисунок 3.4 – Повідомлення про результат моделювання

Користувачам виводиться інформація про стан об'єктів вибірки на кінцевій ітерації (рис. 3.5). Також виводиться графік (рис. 3.6) при натисканні на відповідну кнопку.

Налаштування

Місце для дослідження:

- Відкритий простір
- ХНУ, 3 корпус, 3 поверх, праве крило

Розмір вибірки:

Початкова кількість інфікованих:

Ймовірність інфікування: 0.1-0.3 0.3-0.5 0.5-0.7

Результати дослідження

Вибірка:

Кількість інфікованих:

Кількість здорових:

Рисунок 3.5 – Результати моделювання

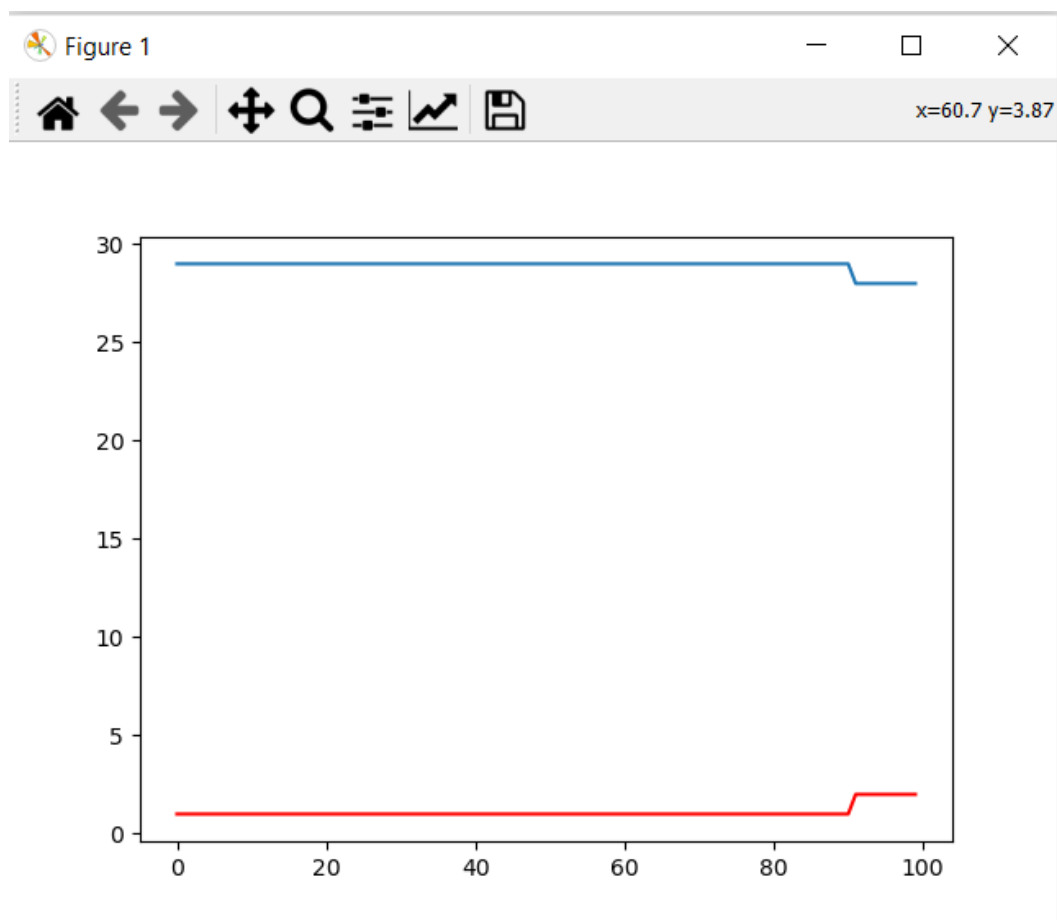


Рисунок 3.6 – Графік, побудований на основі проведеного дослідження

2. Задані параметри частково задовольняють вимоги безпеки (рис. 3.7). В такому випадку знаходження людей в приміщенні в заданій кількості може бути дозволено лише за умови наявності засобів індивідуального захисту та з

дотриманням соціальної дистанції (1,5 – 2 м). Дана ситуація може бути застосована лише у випадку знаходження людей, які здатні до самоконтролю та дисципліни. Не рекомендується застосовувати до закладів дошкільної освіти та загальноосвітніх навчальних закладів I ступеня, а також до позашкільних закладів, де можуть бути присутні діти дошкільного та молодшого шкільного віку.

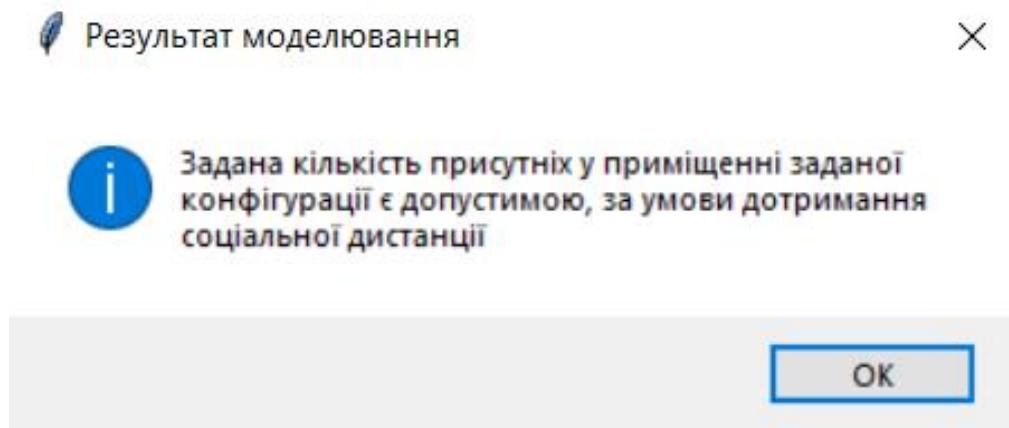


Рисунок 3.7 – Повідомлення про результат моделювання

3. Задані параметри не задовольняють вимогам безпеки (рис. 3.8). В даному випадку існує небезпека сплеску захворюваності. Для запобігання епідеміологічно небезпечної ситуації рекомендується оптимізувати вхідні параметри, а саме перевірити правильність зазначеної кількості хворих або провести дослідження з метою визначення максимально допустимої кількості осіб в приміщенні.

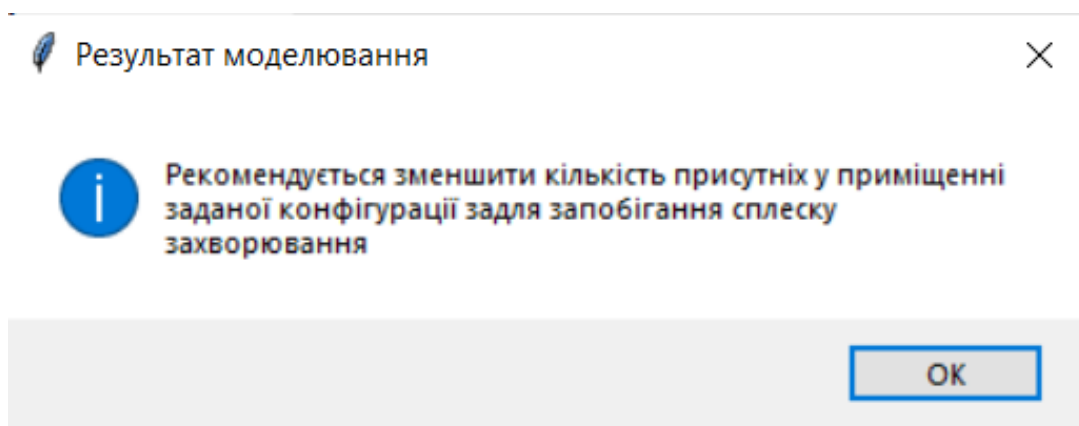


Рисунок 3.8 – Повідомлення про результат моделювання

Після проведеного дослідження користувачеві буде запропоновано переглянути інформацію в узагальненому виді, у вигляді графіків (рис. 3.9). Крім того, переконатися у правильності моделі можна за допомогою симуляції. У випадку несприятливого прогнозу, система запропонує провести оптимізацію моделі, в інших випадках проведення такого дослідження не є доцільним.

Для проведення дослідження, щодо визначення максимальної кількості осіб, що можуть перебувати у приміщенні необхідно натиснути кнопку «Оптимізувати».

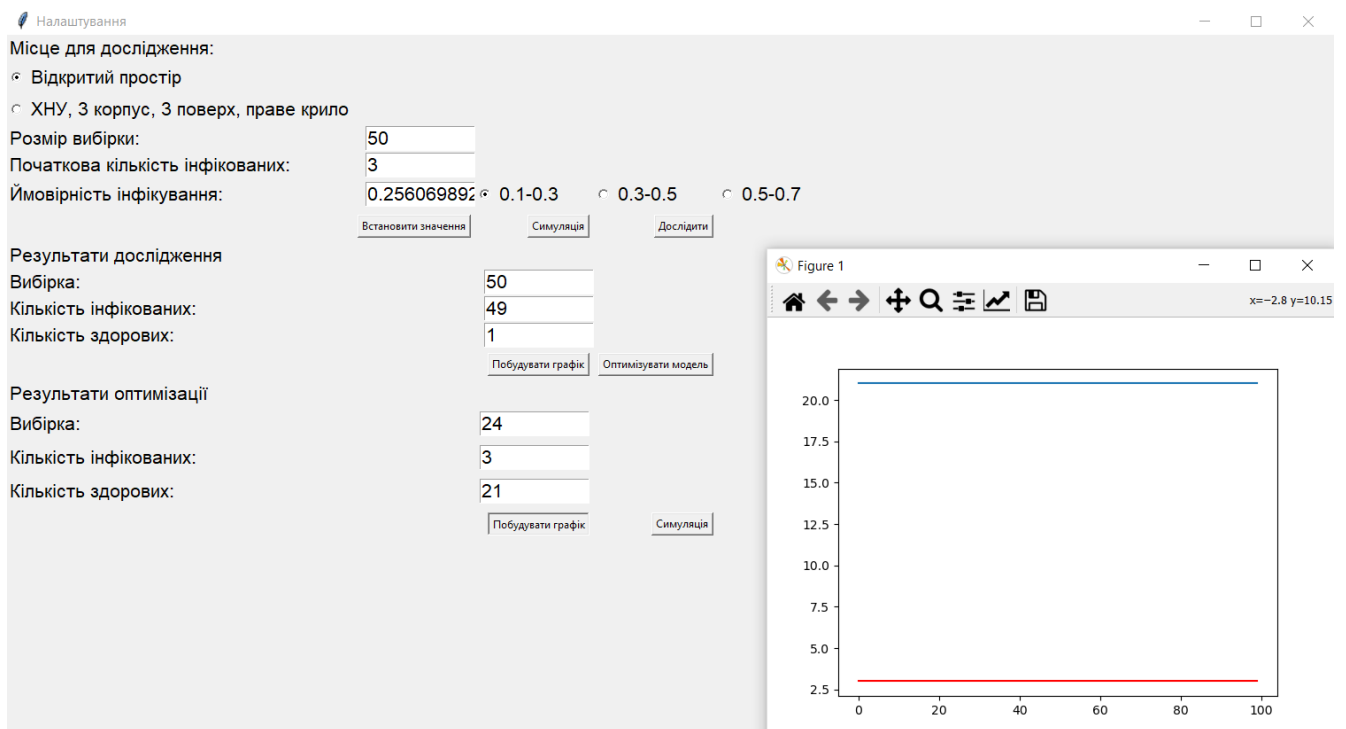


Рисунок 3.9 – Графік, отриманий в результаті оптимізації

Система, опрацювавши інформацію сформує відповідний висновок із представлених варіантів:

1. Оптимізація проведена успішно. Система запропонує оптимальну максимально можливу кількість присутніх у приміщенні заданої конфігурації. Знаходження такої кількості осіб є безпечним. За прогнозом перевищення епідеміологічного порогу не станеться.

2. Не вдалось провести оптимізацію. Система не змогла знайти оптимальної кількості присутніх у приміщенні. Це може бути пов'язано з особливістю конфігурації приміщення, або з «агресивністю» захворювання, що досліджується, що виражається у високій ймовірності інфікування. Також подібний результат може бути пов'язаний з неправильно введеною інформацією, наприклад значною кількістю інфікованих. В такому випадку проведення будь-яких заходів за даних умов є не рекомендованим. Система запропонує змінити вхідні дані, а користувачеві слід оцінити доцільність проведення запланованих заходів в інших приміщеннях або у дистанційному форматі.

Після завершення дослідження з будь-яким результатом користувач матиме змогу подивитись симуляцію, а також переглянути та завантажити графіки за результатами дослідження математичної та імітаційної моделей.

3.4 Оцінка результатів

Проведення дослідження швидкості поширення захворювання із застосуванням розробленої моделі дозволяє пересвідчитись в певних закономірностях. Так, очевидно, що кількість інфікованих буде зростати при збільшенні початкової кількості інфікованих, збільшенні розміру вибірки або збільшенні ймовірності інфікування. Таким чином, для того, щоб зменшити кількість інфікованих необхідно зменшити кількість контактів між об'єктами. Такий результат може бути досягнуто трьома способами: зменшити розмір вибірки, зменшити час перебування осіб в приміщенні, або змінити конфігурацію приміщення, якщо це можливо.

Визначити небезпеку поширення захворювання можна з графіка (рис. 3.10).

Якщо існують такі ділянки, де графік інфікованих різко зростає, тобто похідна функції у певних точках значно відрізняється від нуля, то це свідчить про сплеск захворювання. Тобто, про безпечний підбір параметрів буде свідчити

графік функції, наближений до прямої, тобто похідна у всіх точках графіка наближена до нуля (рис. 3.11).

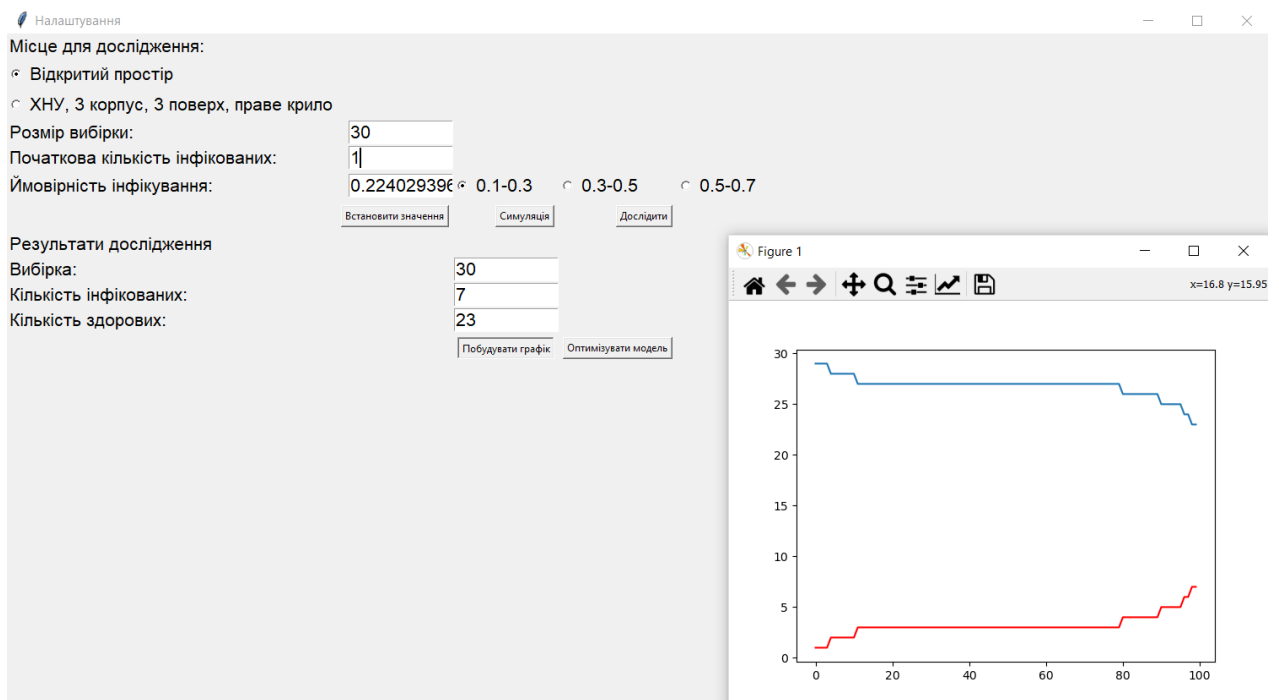


Рисунок 3.10 – Графік дослідження

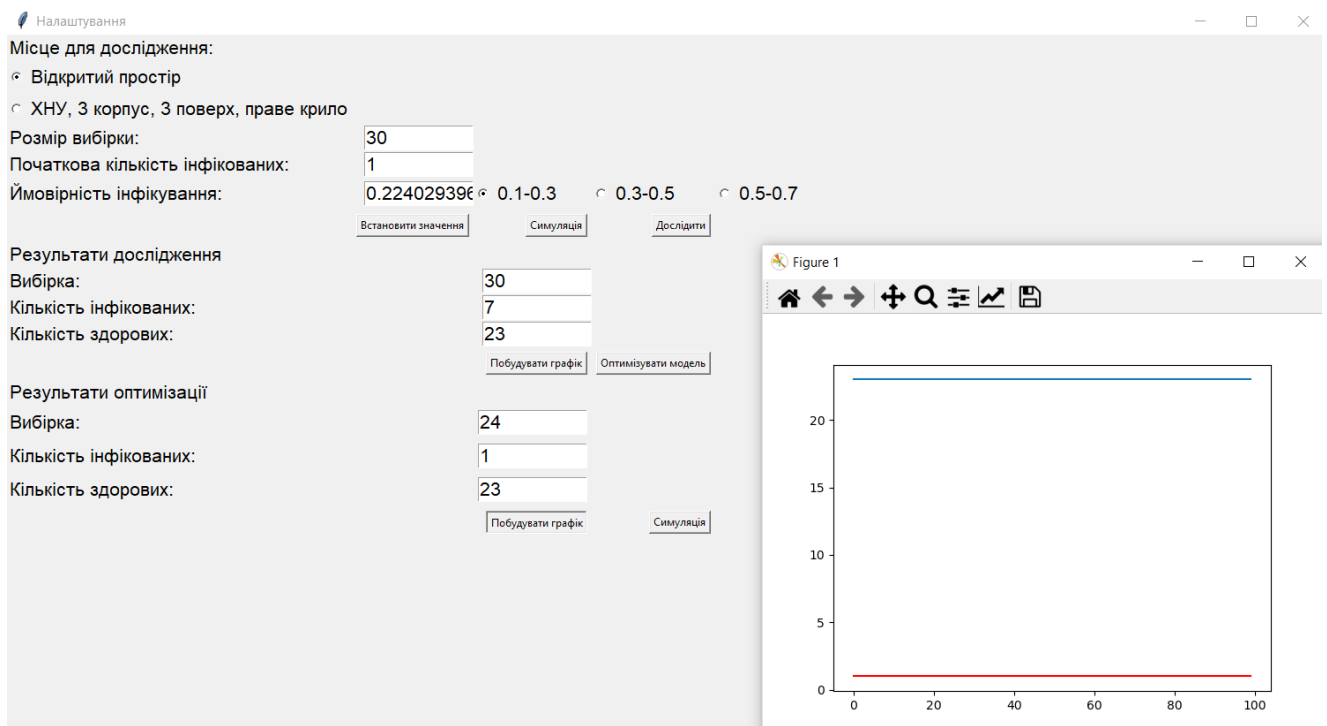


Рисунок 3.11 – Графік дослідження після оптимізації

Місце для дослідження може мати кілька приміщень, які можуть змінювати статус: «відчинено», «зачинено» (рис. 3.12). Залежно від кількості доступних приміщень, змінюється загальна площа приміщення, таким чином, змінюється кількість контактів між присутніми.

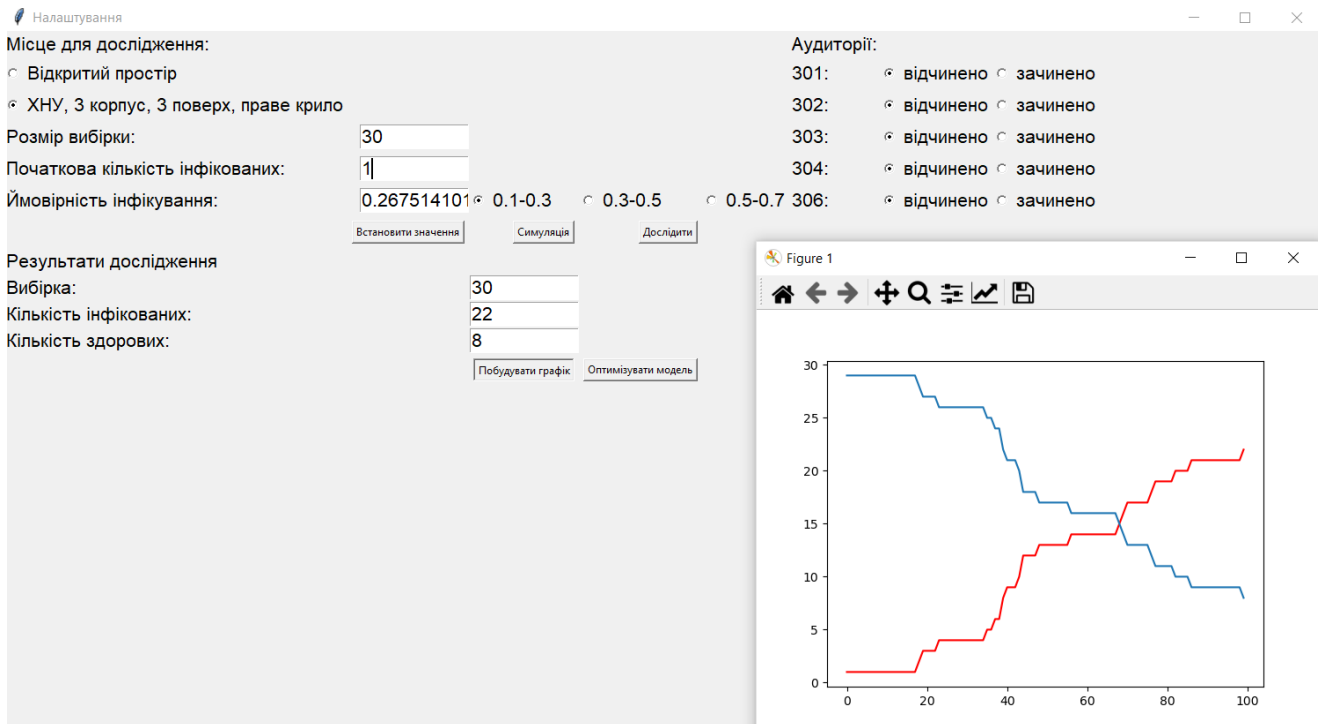


Рисунок 3.12 – Дослідження для кафедри університету

Під час проведення дослідження було виявлено, що розмір початкової вибірки не впливає на результат оптимізації. Суттєвий вплив мають лише конфігурація приміщення та параметр, що відповідає за ймовірність інфікування. Зменшення початкової вибірки до певної межі запобігає сплеску захворювання (рис. 3.13).

Також значний вплив на поширення захворювання має дотримання, або не дотримання соціальної дистанції. Для моделювання ситуації з дотриманням усіма учасниками соціальної дистанції доцільно зменшити діаметр кожного об'єкта, наприклад вдвічі. Це дасть змогу зменшити кількість контактів, порівняно з дослідженням за звичайних умов (рис. 3.14).

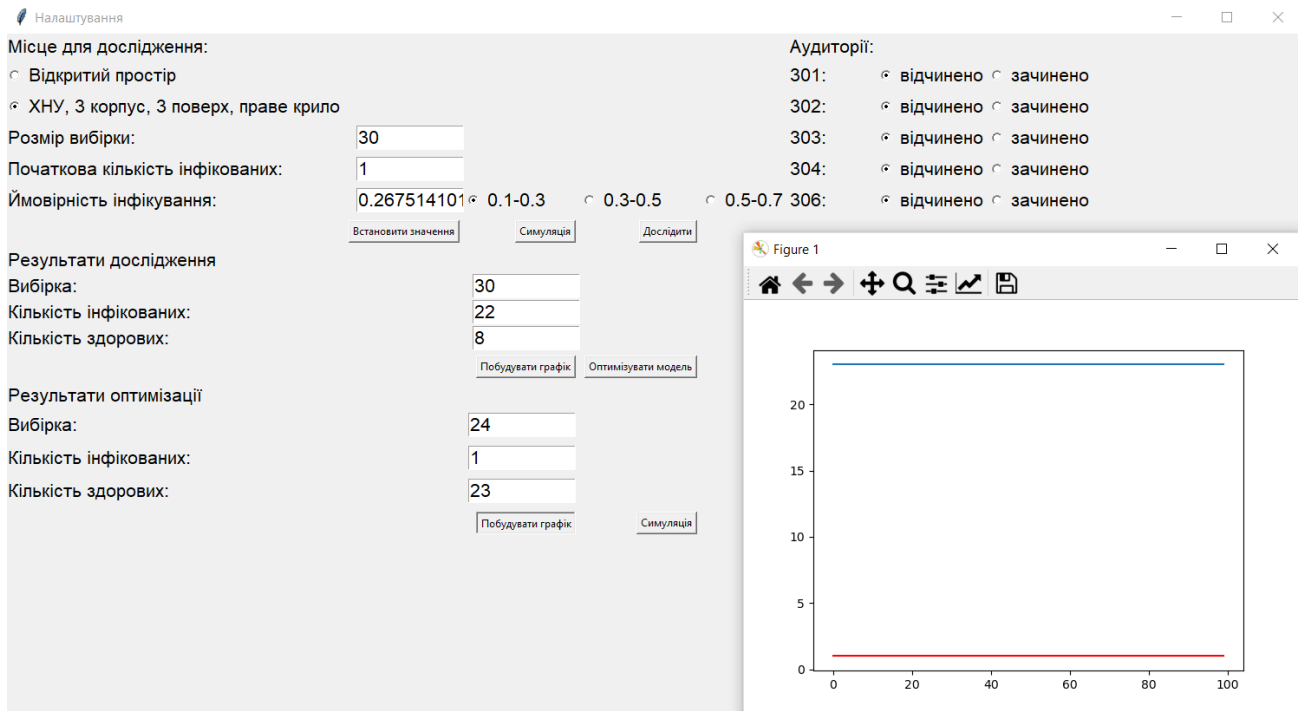


Рисунок 3.13 – Оптимізація параметрів для кафедри університету

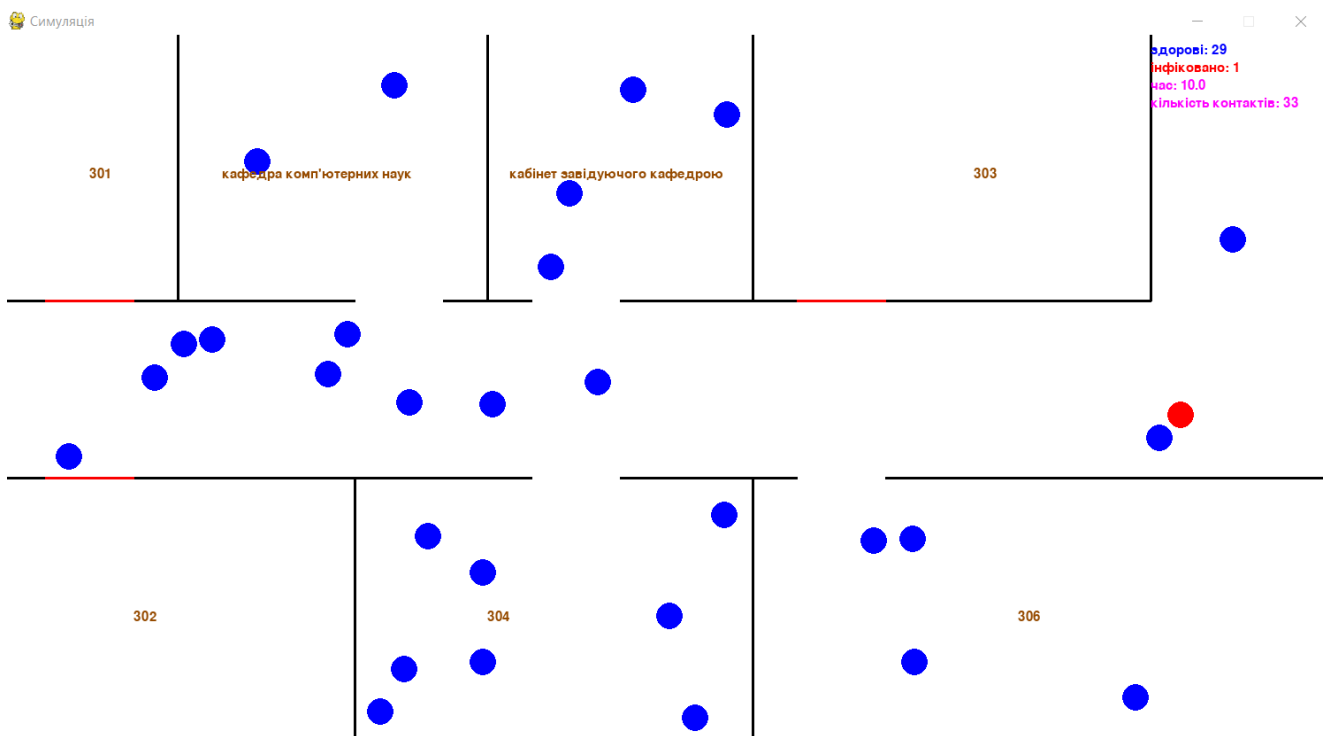


Рисунок 3.14 – Симуляція за звичайних умов

Тобто, навіть значно збільшуючи кількість присутніх, кількість контактів зростає не так суттєво (рис. 3.15).

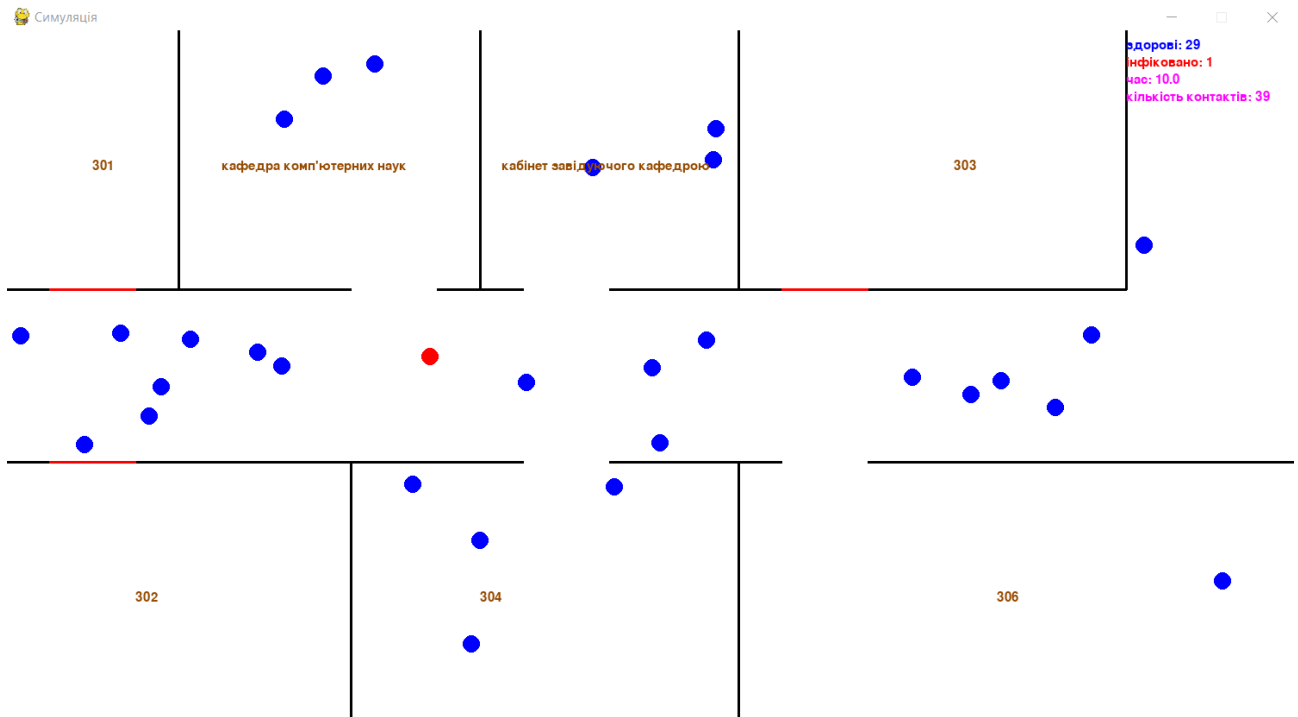


Рисунок 3.15 – Симуляція за умови дотримання соціальної дистанції

Якщо виникає необхідність зменшити кількість доступних приміщень, варто пересвідчитись, чи це не призведе до суттєвого збільшення кількості контактів, адже збільшиться навантаження на інші приміщення, і як наслідок зросте кількість контактів (рис. 3.16).

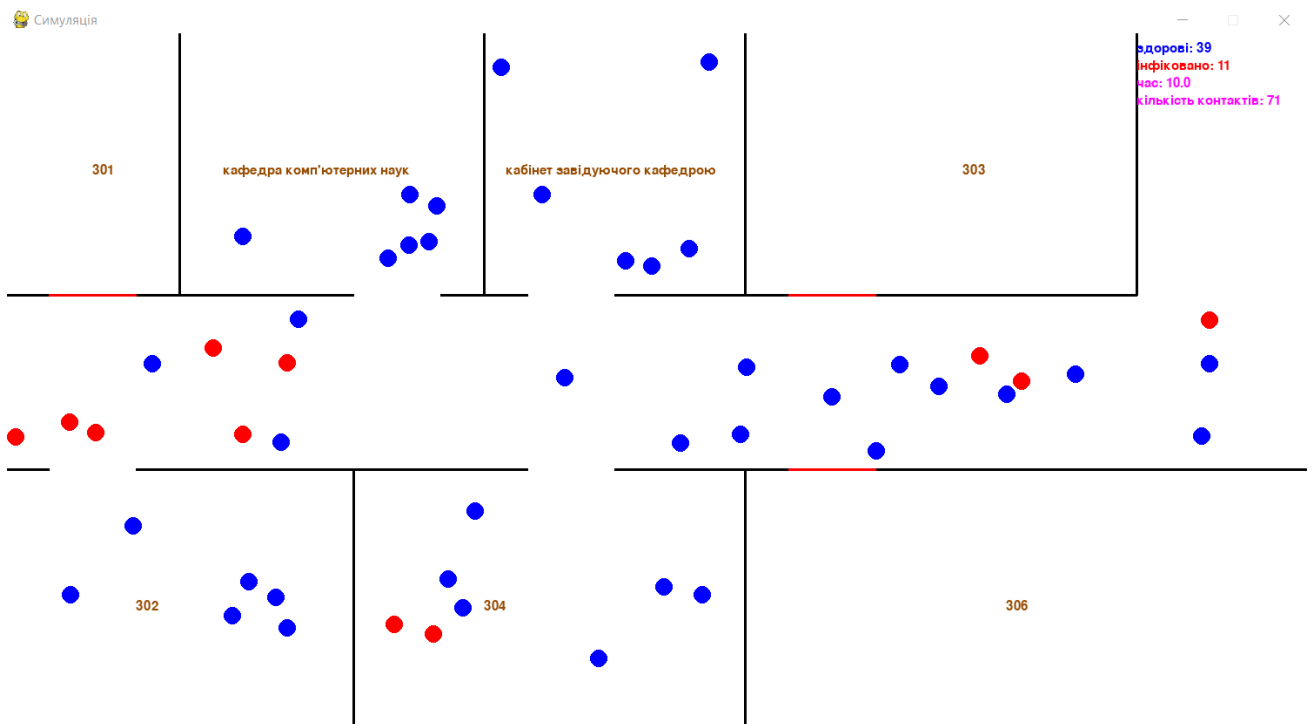


Рисунок 3.16 – Симуляція за умови відсутності доступу до частини приміщень

В такому випадку існує небезпека швидкого зростання кількості інфікованих (рис. 3.17).



Рисунок 3.17 – Графік дослідження симуляції

Вирішенням цієї проблеми може стати збільшення території, тобто відкриття додаткових кімнат, або, за неможливості реалізації цього, зменшення кількості осіб, які одночасно перебувають в приміщенні. Це дає змогу запобігти негативних наслідків та запобігти поширенню захворювання (рис. 3.18).

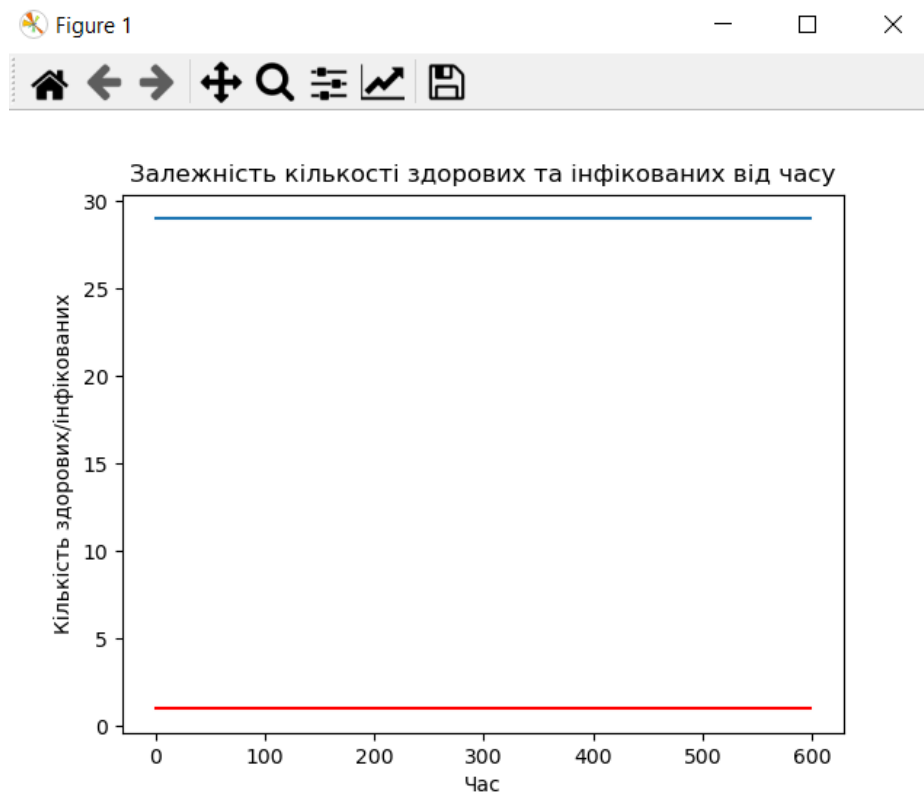


Рисунок 3.19 – Графік дослідження за ідеальних умов

Досить важливим аспектом є можливість зменшення ризику зараження великої кількості людей. На рисунку 3.20 червоним кольором обведено осередок розповсюдження захворювання.

В зоні безпосередньої небезпеки перебувають 5 людей. Один хворий може інфікувати усіх, хто перебуває в аудиторії 304, тому за наявності симптомів важливо ізолювати тих, хто перебував у безпосередньому контакті з хворим, оскільки вони з деякою ймовірністю можуть стати латентними хворими, тобто такими, у яких ще не розвинулись симптоми захворювання, проте вони вже являються носіями та несуть загрозу оточуючим. Ізолювавши одного хворого та п'ять контактних осіб, можна запобігти поширенню захворювання серед інших сорока чотирьох осіб, що перебувають в приміщенні.

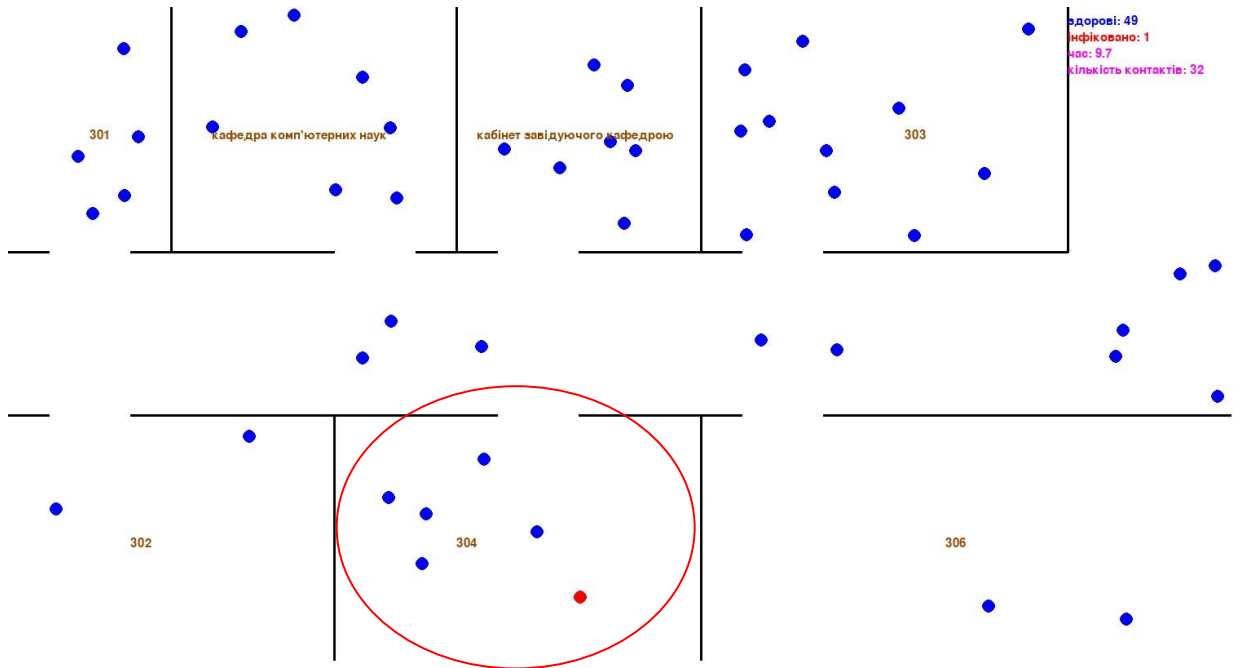


Рисунок 3.20 – Осередок поширення захворювання

Варіант даної ситуації представлено на рисунку 3.21.

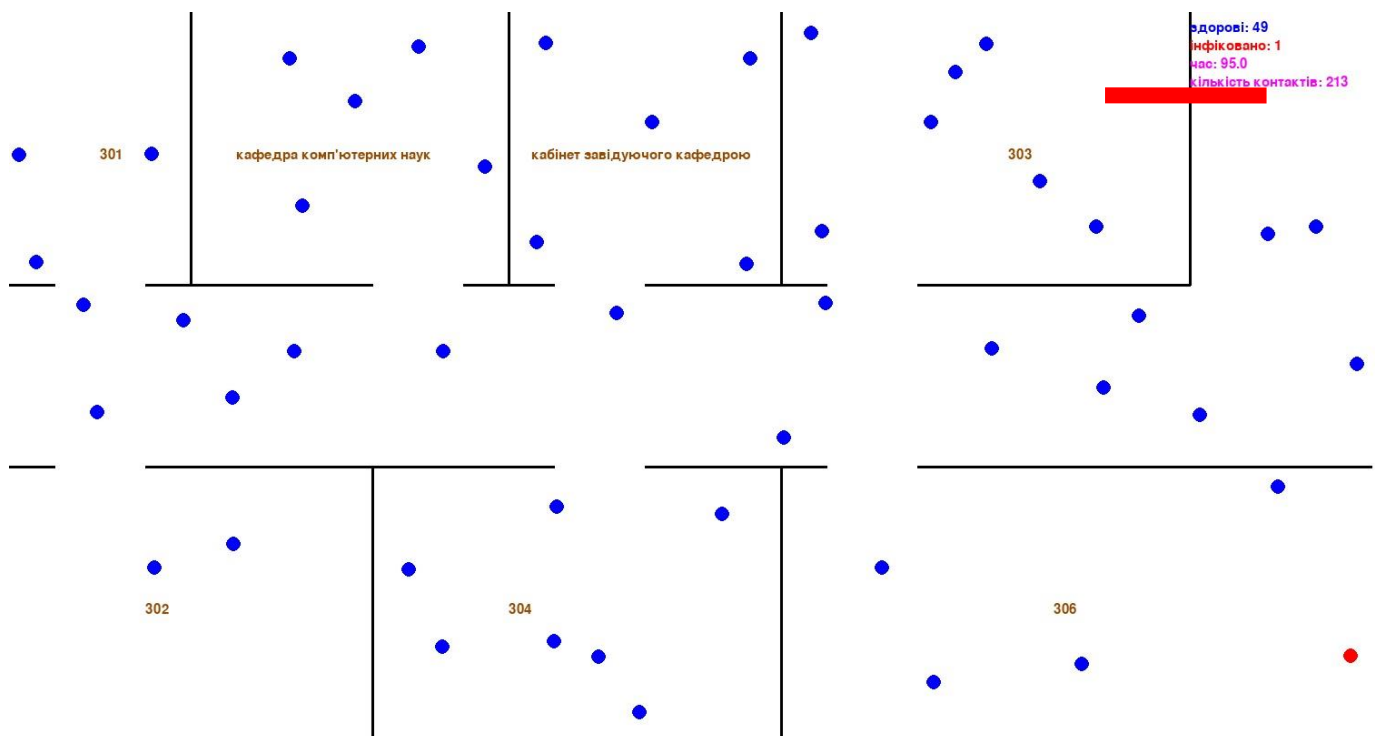


Рисунок 3.21 – Приклад ідеальної ситуації

Слід звернути увагу, що незважаючи на велику кількість контактів та значний час перебування у кімнаті, не відбулось жодного інфікування.

Дотримання соціальної дистанції та ізоляція невеликих груп в окремих аудиторіях може сприятливо вплинути на епідеміологічну ситуацію.

Гіршою ситуацією є наявність одразу кількох осередків зараження. На рисунку 3.22 зображено три таких осередки. Дана ситуація виникає за наявності трьох хворих на початку дослідження. Таким чином група 1 знаходиться в одній аудиторії, за умови, що люди залишатимуться там, вони не будуть контактувати з іншими. В той же час група 2 знаходиться в коридорі, що ставить під загрозу досить значну кількість людей. При умові їх вільного пересування в подальшому під загрозу підпадає значна кількість людей.

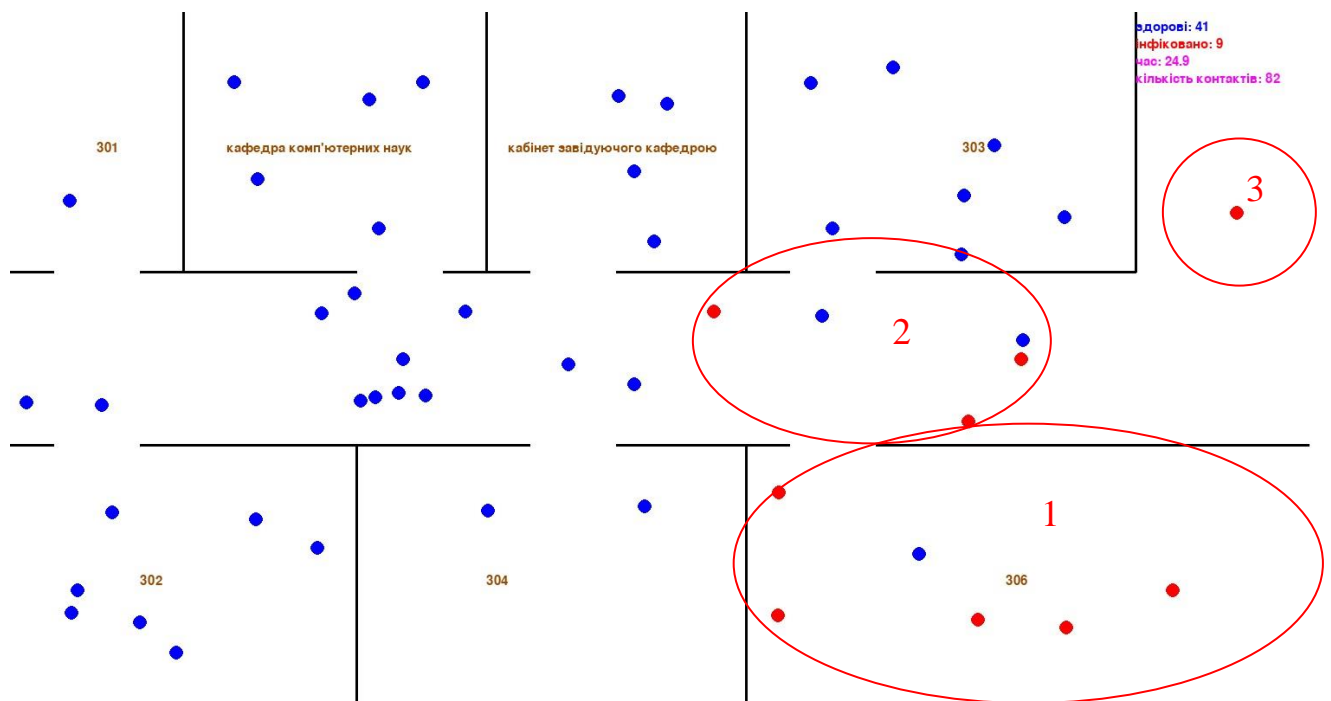


Рисунок 3.22 – Осередки поширення захворювання

Результат дослідження даної ситуації можна побачити на графіку (рис. 3.23). Помаранчева пунктирна лінія вказує межу епідеміологічного порогу для заданої популяції. Як бачимо, цю позначку подолано за досить короткий проміжок часу, що свідчить про небезпеку для людей, що знаходяться в приміщенні.

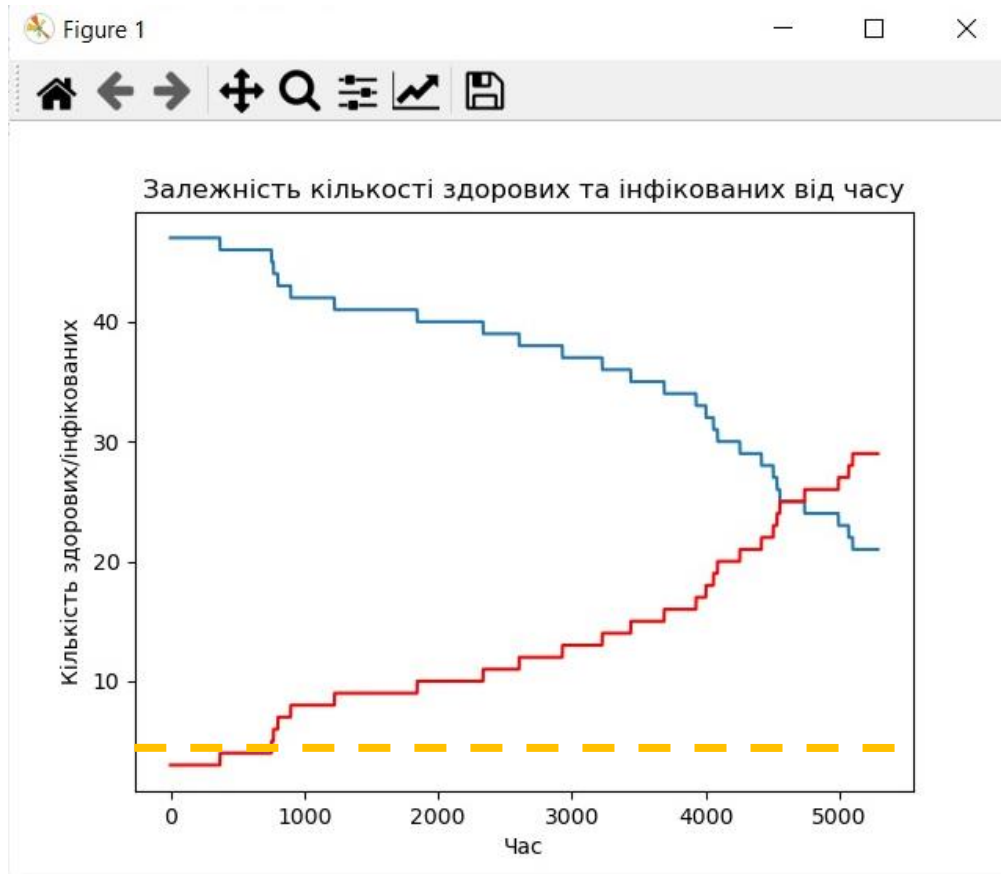


Рисунок 3.23 – Результат дослідження сплеску захворювання

Головною небезпекою респіраторних захворювань є їх швидке поширення, оскільки воно відбувається згідно з експоненціальним законом (рис. 3.24).

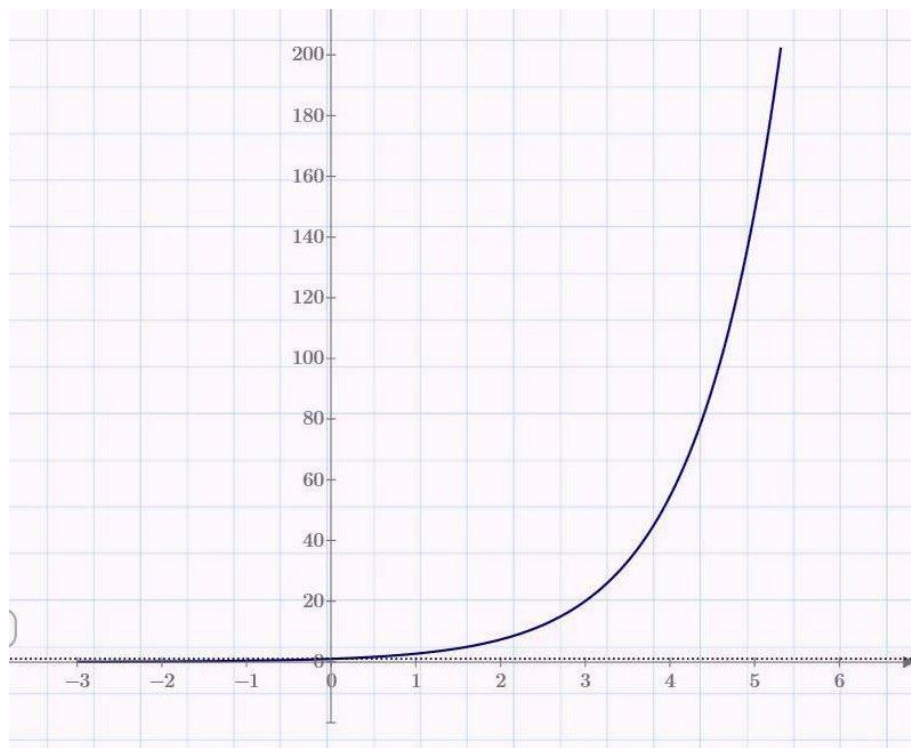


Рисунок 3.24 – Графік експоненти

Зменшення кількості нових випадків з математичної точки зору означає випрямлення цього графіка. Експоненційну залежність можна спостерігати при проведенні досліджень (рис. 3.12, 3.23). У випадках, коли захворювання вдається взяти під контроль спостерігається часткове або значне згладження кривої експоненти (рис. 3.10, 3.17). Ідеальним варіантом є локалізація захворювання та нівелювання нових випадків. Це визначається перетворенням кривої в графік прямої виду $y = \text{const}$, де константа відповідає кількості інфікованих на початку дослідження (рис. 3.11, 3.13, 3.18).

Згідно зі звітами МОЗ України прослідковується, що поширення коронавірусної інфекції, спричиненої вірусом SARS-CoV-2 відбувалося за експоненційним законом (рис. 3.25).

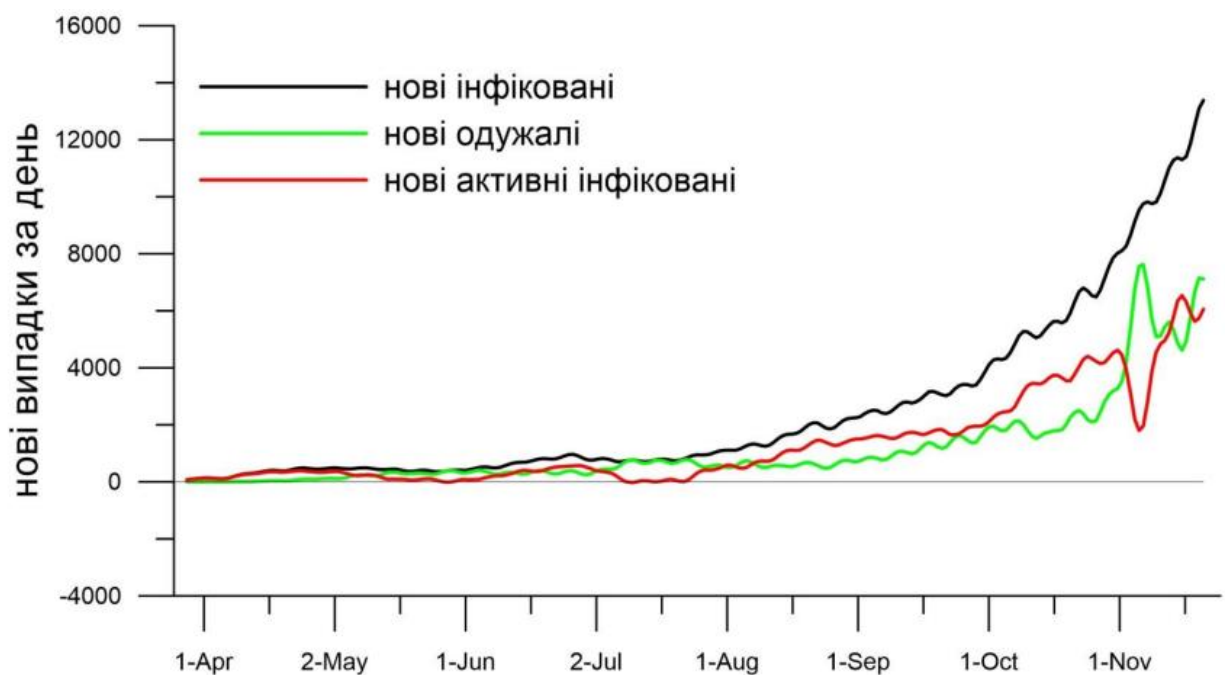


Рисунок 3.25 – Інфікування згідно з даними МОЗ України

Дані прогнозу робочої групи з математичного моделювання проблем, пов'язаних з епідемією коронавірусу SARS-CoV-2 в Україні щодо кількості нових виявлень, зведеної до дати оприлюднення випадку (чорна крива) та дати реєстрації випадку (синя крива) відображено згладженими 7-денним рухомим середнім на рисунку 3.26.

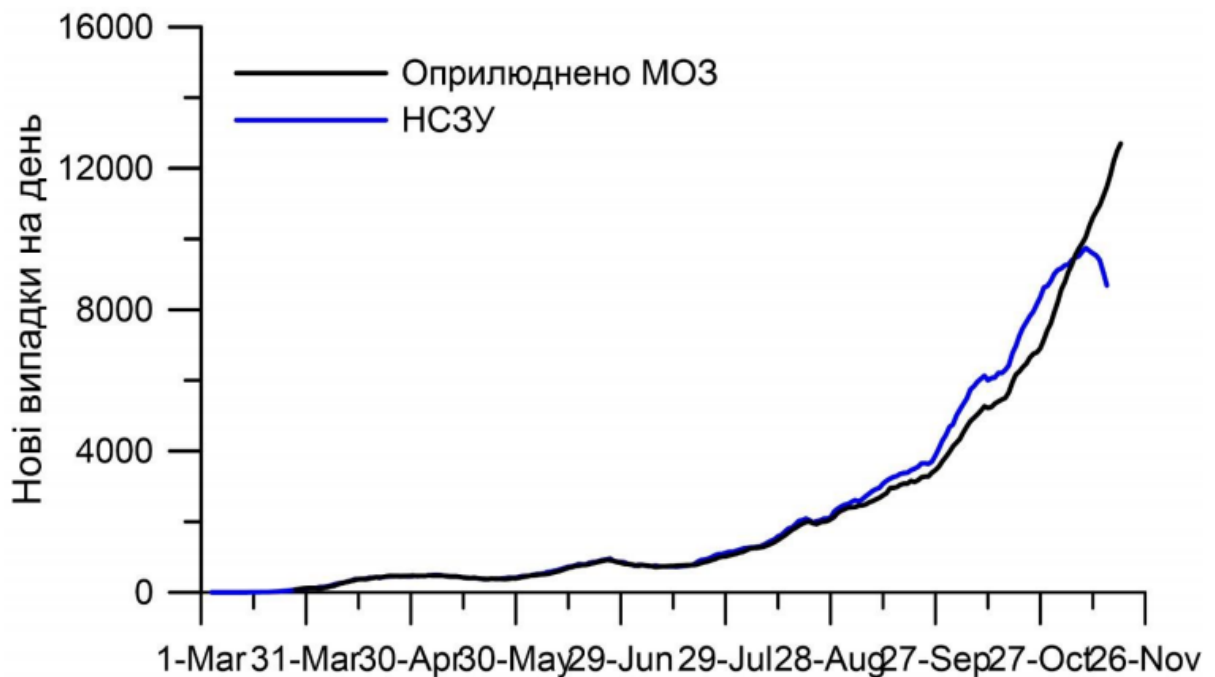


Рисунок 3.26 – Кількість нових випадків захворювання

Наведені дані свідчать про коректність роботи розробленого застосунку, оскільки прогнозування підтверджується статистичними даними реальної пандемії. Таким чином, використання прогнозу може попередити сплески захворювань шляхом екстреного реагування на нові випадки та застосування протиепідеміологічних заходів. Це допоможе знизити ризик захворювань та негативних наслідків, у тому числі летальних.

3.5 Вимоги до розгортання інформаційної системи

В зв'язку з проведенням великої кількості розрахунків та необхідності візуалізації даних програмний продукт може працювати не коректно на слабких пристроях. Рекомендовано використовувати програмний продукт лише на пристроях, які задовольняють мінімальні вимоги.

Вимоги до обладнання та програмного забезпечення клієнта є наступними:

Мінімальні:

- ОС: Windows 7.
- Процесор: x64, Intel Core i3.

- Пам'ять: 1 GB.
- Дисплей: 1400x1050.
- Частота оновлення екрана: 60 Гц.
- Тип відеокарти: інтегрована.
- Модель відеокарти: Intel UHD Graphics.

Рекомендовані:

- ОС: Windows 10.
- Процесор: x64, Intel Core i7- 8300 HQ.
- Пам'ять: 4 GB.
- Дисплей: 1920x1080.
- Частота оновлення екрана: 144 Гц.
- Тип відеокарти: дискретна.
- Модель відеокарти: NVIDIA GeForce GTX.
- Обсяг відеопам'яті: 4 GB.

Висновки

В процесі проведення дослідження було проаналізовано предметну область моделювання епідеміологічних процесів та існуюче програмне забезпечення у цій сфері, розглянуто основні аспекти імітаційного моделювання та його розвиток відповідно до історичних викликів. Поставлене завдання про розробку застосунку для прогнозування поширення респіраторних захворювань на основі SI моделі.

У результаті роботи спроектовано структуру інформаційної системи, створено комп'ютерну та імітаційну моделі на основі математичної моделі та отримано десктопний застосунок для прогнозування поширення респіраторних захворювань. Система здійснює обробку вхідних даних, створює прогноз та надає рекомендації щодо зменшення або нівелювання нових випадків захворювання. У випадку серйозної небезпеки захворювання програмний продукт попереджає про небезпеку та її характер. Для візуалізації процесу розповсюдження респіраторних захворювань вирішено застосовувати бібліотеку модулів Pygame.

Розроблена система може бути впроваджена для здійснення прогнозування поширення респіраторних захворювань. Застосунок буде корисним для організаторів масових заходів, а також медичних працівників в галузі епідеміології.

Проведено порівняння отриманих результатів зі статистичними даними МОЗ України щодо пандемії Covid-19. Визначено загальні тенденції поширення респіраторних захворювань.

Відповідно до отриманого результату, можна зробити висновок, що розроблений програмний продукт є ергономічним, має інтуїтивно-зрозумілий інтерфейс, працює вірно, а вимоги технічного завдання виконані в повному обсязі.

Перелік посилань

1. Андрейчин А. М., Василюшин З. П., Виноград, Н. О. Колесникова, І. П. Епідеміологія: Підручник для ВМНЗ IV р.а.. Вінниця: Нова Книга, 2012. 576 с.
2. Вьюненко Л. Ф. Имитационное моделирование. Юрайт, 2016, 284 с.
3. Кодекс цивільного захисту України від 02.10.2012 № 5403-VI (Редакція від 12.05.2017)
4. Иванов М.В. Математическое моделирование процесса пандемии: теория и практика. URL: <https://indsi.ru/2020/04/30/>
5. Медична енциклопедія. Режим доступу: <http://medical-enc.com.ua>
6. Мезенцева Л.В., Перцов С.С. Математическое моделирование в биомедицине. Вестник новых медицинских технологий. 2013. Т. XX. № 1. С. 11.
7. Печінка А. М., Дзедман М. І. Гострі респіраторні захворювання: питання клінічної діагностики та лікування. Український медичний часопис. Актуальні питання клінічної практики. Режим доступу: <https://www.umj.com.ua/article/6986/>
8. Bernoulli D. Essai d'une nouvelle analyse de la mortalite causee par la petite verole // Mem Math Phy AcadRoy Sci Paris, 1766.
9. Brownlee J. Statistical studies in immunity: the theory of an epidemic // Proceedings of the Royal Society of Edinburgh 26.1, 1906. P. 484 – 521.
10. Bulakh I. Y., Liakh Y. Y., Martseniuk V. P., Khaimzon I. Y. Medical Informatics. AUS Medicine Publishing. Kyiv, 2017, 368p.
11. Edelstein-Keshet L., Mathematical Models in Biology, Society for Industrial and Applied Mathematics, 2005, 586 p.
12. Elveback L., Ackerman E., Gatewood L., Fox J. Stochastic two-agent epidemic simulation models for a community of families. American Journal of Epidemiology. 1971. N. 93. P. 267 – 280.
13. Goh, B. Global stability in many-species systems, Amer. Naturalist, 1977. p. 135 – 143. URL: <https://www.jstor.org/stable/2459985>.

14. Huppert G. Katriel. Mathematical modelling and prediction in infectious disease epidemiology // *Clinical Microbiology and Infection*. 2013. Vol. 19. N 11. P. 999 – 1005
15. Kendall D.G. Discussion of 'Measles periodicity and community size' by M.S. Bartlett // *J Roy Stat Soc. Ser. A*. 1957. N. 120. P. 64 – 76.
16. Kermack W.O., McKendrick A.G. A contribution to the mathematical theory of epidemics // *Proc R Soc Lond*. 1927. N. 115. P. 700 – 721.
17. Longini I., Nizam A., Xu S., Ungchusak K., Hanshaoworakul W., Cummings D. & Halloran M. E. Containing pandemic influenza at the source, *Science* 309. 2004 P. 623 – 633. Retrieved from <https://pubmed.ncbi.nlm.nih.gov/16079251/>.
18. May R. M. Uses and abuses of mathematics in biology. *Science*. 2004. N. 303. P. 790 – 793.
19. Merler S., Ajelli M., Pugliese A., Ferguson N.M. Determinants of the spatiotemporal dynamics of the 2009 H1N1 pandemic in Europe: implications for real-time modelling. *PLoS Comput Biol*. 2011. Vol. 7. N 9: e1002205.
20. Myers, M., Rogers, D., Cox, J., Flahault, A. & Hay, S. (2000). Forecasting Disease Risk for Increased Epidemic Preparedness in Public Health, *Advances in Parasitology*, Vol. 47, p. 309 – 330. Retrieved from <https://www.sciencedirect.com/science/article/abs/pii/S0065308X00470132>.
21. Pitzer V.E., Atkins K.E., de Blasio B.F. et al. Direct and indirect effects of rotavirus vaccination: comparing predictions from transmission dynamic models. *PLoS ONE*. 2012. Vol. 7. N 8: 42320.
22. Porta, Miquel, ed. *Dictionary of Epidemiology*. Oxford University Press, 2008. 179 p.
23. *The Mathematics of Infectious Diseases*, *SIAM Review*, 2000, 599 – 653 p.
24. Yaari R., Katriel G., Huppert A., Axelsen J.B., Stone L. Modelling seasonal influenza: the role of weather and punctuated antigenic drift. *J R Soc Interface* 2013. N. 10: 20130298.

ДОДАТКИ

Додаток А

Перевірка на плагіат в системі Anti-Plagiarism

27.05.2022, 16:34

result_1265928292198119463.html

Fri May 27 15:46:22 EEST 2022, Петровський Сергій Степанович, Хмельницький національний університет, ХНУ

Anti-Plagiarism v-15.257

Максимальное совпадение с одним документом 2.0%

Словари проверки: en_US, ru_RU, ua_UA. Ошибок в документах: 8%

ID: 104117 Название: КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА Імітаційне моделювання розповсюдження респіраторних захворювань Добавлено в БД: 2022-05-27 Авторы: В.О. Алексейко Руководители: О.В. Бармак Консультанты: Оponentы:	Документ		Суммарное совпадение по Базе Данных	
	Символы	Лексемы	Символы	Лексемы
	63961	989	2313 (4%)	41 (4%)

Источник плагиата

ID	Описание	Наличие плагиата в документе	
		Символы	Лексемы

Додаток Б

Перевірка на плагіат в системі Unicheck



Ім'я користувача:
Кафедра КН

ID перевірки:
1011350305

Дата перевірки:
27.05.2022 16:37:39 EEST

Тип перевірки:
Docvs Internet + Library

Дата звіту:
27.05.2022 16:38:39 EEST

ID користувача:
100005671

Назва документа: **Алексейко_ЗАПИСКА_short**

Кількість сторінок: 60 Кількість слів: 10077 Кількість символів: 77938 Розмір файлу: 1.90 MB ID файлу: 1011236218

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

13.9%

Схожість

Найбільша схожість: 2.89% з Інтернет-джерелом (<https://www.umj.com.ua/article/6986/gost-ri-respiratori-zaxvoryuvan..>)

11.8% Джерела з Інтернету 165 Сторінка 62

2.9% Джерела з Бібліотеки 77 Сторінка 63

0% Цитат

Вилучення цитат вимкнено

Вилучення списку бібліографічних посилань вимкнено

0%

Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи 1

Підозріле форматування 15 сторінок

Додаток В

Лістинг програмного коду

```

# import the modules we'll need:
import pygame #to draw window, particles
import random #to determine random movement
import math #to use math functions
import time #to keep track of infection time
import matplotlib.pyplot as plt #to create a graph
from tkinter import messagebox
import json
from tkinter import *
from array import *
root = Tk()
EntryA = Entry(root, width=10, font='Arial 16')
EntryB = Entry(root, width=10, font='Arial 16')
EntryC = Entry(root, width=10, font='Arial 16')
EntryD = Entry(root, width=10, font='Arial 16')
EntryE = Entry(root, width=10, font='Arial 16')
EntryF = Entry(root, width=10, font='Arial 16')
EntryG = Entry(root, width=10, font='Arial 16')
EntryH = Entry(root, width=10, font='Arial 16')
EntryI = Entry(root, width=10, font='Arial 16')
EntryJ = Entry(root, width=10, font='Arial 16')
EntryK = Entry(root, width=10, font='Arial 16')
EntryL = Entry(root, width=10, font='Arial 16')
EntryM = Entry(root, width=10, font='Arial 16')
EntryN = Entry(root, width=10, font='Arial 16')
EntryO = Entry(root, width=10, font='Arial 16')
variant = IntVar()
variant.set(0)
soc_dist = IntVar()
soc_dist.set(0)
time_to_recover = IntVar()
time_to_recover.set(0)
min_inf = IntVar()
min_inf.set(0)
min_recovery = IntVar()
min_recovery.set(0)
min_symp = IntVar()
min_symp.set(0)
#аудиторії
room_301 = IntVar()
room_301.set(0)
room_302 = IntVar()
room_302.set(0)
room_303 = IntVar()
room_303.set(0)
room_304 = IntVar()
room_304.set(0)
room_306 = IntVar()
room_306.set(0)
global INF_PROBABILITY
def main():
    root.title("Налаштування")
    root.geometry("1500x800")
    Label(root, text='Місце для дослідження:', font='Arial 16').grid(row=0, sticky=W)
    variant1 = Radiobutton(text="Відкритий простір", font='Arial 16',
        variable=variant, value=0)
    variant2 = Radiobutton(text="ХНУ, 3 корпус, 3 поверх, праве крило", font='Arial 16',
        variable=variant, value=1)
    variant1.grid(column=0, row=1, sticky=W)
    variant2.grid(column=0, row=2, sticky=W)
    Label(root, text='Дотримання соціальної дистанції:', font='Arial 16').grid(column=1,
row=0, sticky=W)
    variant_soc_dist1 = Radiobutton(text="Так", font='Arial 16',
        variable=soc_dist, value=0)
    variant_soc_dist2 = Radiobutton(text="Ні", font='Arial 16',
        variable=soc_dist, value=1)
    variant_soc_dist1.grid(column=1, row=1, sticky=W)
    variant_soc_dist2.grid(column=1, row=2, sticky=W)
    Label(root, text='Розмір вибірки:', font='Arial 16').grid(row=3, sticky=W)
    Label(root, text='Початкова кількість інфікованих:', font='Arial 16').grid(row=4,
sticky=W)
    Label(root, text='Ймовірність інфікування:', font='Arial 16').grid(row=5, sticky=W)
    var_min_inf1 = Radiobutton(text="0.1-0.3", font='Arial 16',

```

```

        variable=min_inf, value=0)
var_min_inf2 = Radiobutton(text="0.3-0.5", font='Arial 16',
        variable=min_inf, value=1)
var_min_inf3 = Radiobutton(text="0.5-0.7", font='Arial 16',
        variable=min_inf, value=2)
var_min_inf1.grid(column=2, row=5, sticky=W)
var_min_inf2.grid(column=3, row=5, sticky=W)
var_min_inf3.grid(column=4, row=5, sticky=W)
EntryA.grid(row=3, column=1, sticky=W)
EntryA.delete(0, 'end')
EntryA.insert(0, "50")
EntryB.grid(row=4, column=1, sticky=W)
EntryB.delete(0, 'end')
EntryB.insert(0, "3")
EntryD.grid(row=5, column=1, sticky=W)
EntryD.delete(0, 'end')
EntryD.insert(0, INF_PROBABILITY)
btn1 = Button(text="Встановити значення", command=set_parameters)
btn2 = Button(text="Симуляція", command=sim1)
btn3 = Button(text="Дослідити", command=model)
btn1.grid(row=12, column=1, padx=5, pady=5, sticky="W")
btn2.grid(row=12, column=2, padx=5, pady=5, sticky="W")
btn3.grid(row=12, column=3, padx=5, pady=5, sticky="W")
root.mainloop()
def sim1():
    global NUM_OBJECTS
    NUM_OBJECTS=EntryA.get()
    NUM_OBJECTS=int(NUM_OBJECTS)
    global INFECTED
    INFECTED=EntryB.get()
    INFECTED=int(INFECTED)
    simulation()
def set_parameters():
    global RECOVERY
    global INFECTION_MIN
    global INFECTION_MAX
    global RECOVERY_MIN
    global SYMPTOMATIC_MIN
    global SYMPTOMATIC_MAX
    global INF_PROBABILITY
    if (time_to_recover.get()==0):
        RECOVERY=random.uniform(3,\
            7)
    elif (time_to_recover.get()==1):
        RECOVERY=random.uniform(7,\
            11)
    else:
        RECOVERY=random.uniform(11,\
            14)
    RECOVERY=1000
    if (min_inf.get()==0):
        INF_PROBABILITY = random.uniform(0.1,\
            0.3)
    elif (min_inf.get()==1):
        INF_PROBABILITY=random.uniform(0.3,\
            0.5)
    else:
        INF_PROBABILITY=random.uniform(0.5,\
            0.7)
    if (min_recovery.get()==0):
        RECOVERY_MIN=random.uniform(0.7,\
            0.8)
    elif (min_recovery.get()==1):
        RECOVERY_MIN=random.uniform(0.8,\
            0.9)
    else:
        RECOVERY_MIN=random.uniform(0.9,\
            0.99)
    if (min_symp.get()==0):
        SYMPTOMATIC_MIN=random.uniform(0.1,\
            0.3)
    elif (min_symp.get()==1):
        SYMPTOMATIC_MIN=random.uniform(0.3,\
            0.6)
    else:
        SYMPTOMATIC_MIN=random.uniform(0.6,\

```

```

                                0.8)
SYMPTOMATIC_MAX = random.uniform(SYMPTOMATIC_MIN,\
                                0.9)
if variant.get() == 1:
    Label(root, text='Аудиторiї:', font='Arial 16').grid(column=5, row=0, sticky=W)
    Label(root, text='301:', font='Arial 16').grid(column=5, row=1, sticky=W)
    room_301_open = Radiobutton(text="Вiдчинено", font='Arial 16',
                                variable=room_301, value=0)
    room_301_closed = Radiobutton(text="зачинено", font='Arial 16',
                                variable=room_301, value=1)
    room_301_open.grid(column=6, row=1, sticky=W)
    room_301_closed.grid(column=7, row=1, sticky=W)
    Label(root, text='302:', font='Arial 16').grid(column=5, row=2, sticky=W)
    room_302_open = Radiobutton(text="Вiдчинено", font='Arial 16',
                                variable=room_302, value=0)
    room_302_closed = Radiobutton(text="зачинено", font='Arial 16',
                                variable=room_302, value=1)
    room_302_open.grid(column=6, row=2, sticky=W)
    room_302_closed.grid(column=7, row=2, sticky=W)
    Label(root, text='303:', font='Arial 16').grid(column=5, row=3, sticky=W)
    room_303_open = Radiobutton(text="Вiдчинено", font='Arial 16',
                                variable=room_303, value=0)
    room_303_closed = Radiobutton(text="зачинено", font='Arial 16',
                                variable=room_303, value=1)
    room_303_open.grid(column=6, row=3, sticky=W)
    room_303_closed.grid(column=7, row=3, sticky=W)
    Label(root, text='304:', font='Arial 16').grid(column=5, row=4, sticky=W)
    room_304_open = Radiobutton(text="Вiдчинено", font='Arial 16',
                                variable=room_304, value=0)
    room_304_closed = Radiobutton(text="зачинено", font='Arial 16',
                                variable=room_304, value=1)
    room_304_open.grid(column=6, row=4, sticky=W)
    room_304_closed.grid(column=7, row=4, sticky=W)
    Label(root, text='306:', font='Arial 16').grid(column=5, row=5, sticky=W)
    room_306_open = Radiobutton(text="Вiдчинено", font='Arial 16',
                                variable=room_306, value=0)
    room_306_closed = Radiobutton(text="зачинено", font='Arial 16',
                                variable=room_306, value=1)
    room_306_open.grid(column=6, row=5, sticky=W)
    room_306_closed.grid(column=7, row=5, sticky=W)
def model():
    global NUM_OBJECTS
    global INFECTED
    global objects
    NUM_OBJECTS=EntryA.get()
    try:
        NUM_OBJECTS=int(NUM_OBJECTS)
    except ValueError:
        messagebox.showinfo("Помилка", "Невiрно заданi вхiднi значення!!!")
    INFECTED=EntryB.get()
    INFECTED=int(INFECTED)
    objects = array('i', [0])
    for i in range (0, NUM_OBJECTS - 1):
        objects.append(0)
    for i in range (0, INFECTED):
        objects.pop()
        objects.insert(0, 1)
    Label(root, text='Результати дослiдження', font='Arial 16').grid(column=0, row=13,
    sticky=W)
    Label(root, text='Вибiрка:', font='Arial 16').grid(column=0, row=14, sticky=W)
    Label(root, text='Кiлькiсть iнфiкованих:', font='Arial 16').grid(column=0, row=15,
    sticky=W)
    Label(root, text='Кiлькiсть здорових:', font='Arial 16').grid(column=0, row=16,
    sticky=W)
    EntryM.grid(row=14, column=1, sticky=W)
    EntryN.grid(row=15, column=1, sticky=W)
    EntryO.grid(row=16, column=1, sticky=W)
    inf()
def inf():
    global NUM_OBJECTS
    global INFECTED
    global objects
    global INF_PROBABILITY
    global NUM_CONTACTS
    if NUM_OBJECTS < 25:
        NUM_CONTACTS = 1
    elif NUM_OBJECTS >= 25 and NUM_OBJECTS < 35:
        NUM_CONTACTS = 3

```

```

elif NUM_OBJECTS >= 35 and NUM_OBJECTS < 50:
    NUM_CONTACTS = 5
elif NUM_OBJECTS >= 50 and NUM_OBJECTS < 75:
    NUM_CONTACTS = 7
elif NUM_OBJECTS >= 75 and NUM_OBJECTS < 90:
    NUM_CONTACTS = 15
elif NUM_OBJECTS >= 90 and NUM_OBJECTS < 100:
    NUM_CONTACTS = 28
else:
    NUM_CONTACTS = 35
global x
global y1
global y2
x = array('i', [0])
y1 = array('i', [INFECTED])
y2 = array('i', [NUM_OBJECTS - INFECTED])
for i in range(0, 99):
    for j in range(0, NUM_CONTACTS - 1):
        obj_1 = random.randint(0, NUM_OBJECTS - 1)
        obj_2 = random.randint(0, NUM_OBJECTS - 1)
        if obj_1 != obj_2:
            if objects[obj_1] == 1 and objects[obj_2] == 0:
                num = random.random()
                if num <= INF_PROBABILITY:
                    objects[obj_2] = 1
            if objects[obj_1] == 0 and objects[obj_2] == 1:
                num = random.random()
                if num <= INF_PROBABILITY:
                    objects[obj_1] = 1
        x.append(i + 1)
        y1.append(objects.count(1))
        y2.append(objects.count(0))
EntryM.delete(0, 'end')
EntryN.delete(0, 'end')
EntryO.delete(0, 'end')
EntryM.insert(0, objects.count(1) + objects.count(0))
EntryN.insert(0, objects.count(1))
EntryO.insert(0, objects.count(0))
with open('data.txt') as json_file:
    data = json.load(json_file)
data['research'].append({
    "building_config": variant.get(),
    'num_objects': NUM_OBJECTS,
    'infected': INFECTED,
    'inf_probability': INF_PROBABILITY,
    'infected_particles': objects.count(1)
})
with open('data.txt', 'w') as outfile:
    json.dump(data, outfile)
btn4 = Button(text="Побудувати графік", command=graph)
btn4.grid(row=17, column=1, padx=5, pady=5, sticky="W")
if objects.count(1) - INFECTED > NUM_OBJECTS*0.1:
    messagebox.showinfo("Результат моделювання", "Рекомендується зменшити кількість
присутніх у приміщенні заданої конфігурації задля запобігання сплеску захворювання")
    #Label(root, text='Рекомендується зменшити кількість присутніх у приміщенні
заданої конфігурації задля запобігання сплеску захворювання', font='Arial
16').grid(column=0, row=14, sticky=W)
    btn5 = Button(text="Оптимізувати модель", command=optim)
    btn5.grid(row=17, column=2, padx=5, pady=5, sticky="e")
    elif objects.count(1) - INFECTED >= NUM_OBJECTS*0.05 and objects.count(1) - INFECTED
<= NUM_OBJECTS*0.1:
    messagebox.showinfo("Результат моделювання", "Задана кількість присутніх у
приміщенні заданої конфігурації є допустимою, за умови дотримання соціальної дистанції")
    btn11 = Button(text="Нове дослідження", command = new)
    btn11.grid(row=24, column=2, padx=5, pady=5, sticky="e")
    btn12 = Button(text="Вийти", command = exit)
    btn12.grid(row=24, column=3, padx=5, pady=5, sticky="e")
else:
    messagebox.showinfo("Результат моделювання", "Задана кількість присутніх у
приміщенні заданої конфігурації є безпечною")
    #Label(root, text='Задана кількість присутніх у приміщенні заданої конфігурації
є безпечною', font='Arial 16').grid(column=0, row=14, sticky=W)
def graph():
    global x
    global y1
    global y2
    plt.plot(x, y1, 'r')
    plt.plot(x, y2)

```

```

plt.show()
def optim():
    global NUM_OBJECTS
    global INFECTED
    global objects
    global INF_PROBABILITY
    global NUM_CONTACTS
    global objects
    global xx
    global yy1
    global yy2
    while objects.count(1) - INFECTED > NUM_OBJECTS*0.05:
        NUM_OBJECTS = NUM_OBJECTS - 1
        objects.pop()
        for i in range(0, NUM_OBJECTS - 1):
            objects[i] = 0
        for i in range(0, INFECTED):
            objects.pop()
            objects.insert(0, 1)
        if NUM_OBJECTS < 25:
            NUM_CONTACTS = 1
        elif NUM_OBJECTS >= 25 and NUM_OBJECTS < 35:
            NUM_CONTACTS = 3
        elif NUM_OBJECTS >= 35 and NUM_OBJECTS < 50:
            NUM_CONTACTS = 5
        elif NUM_OBJECTS >= 50 and NUM_OBJECTS < 75:
            NUM_CONTACTS = 7
        elif NUM_OBJECTS >= 75 and NUM_OBJECTS < 90:
            NUM_CONTACTS = 15
        elif NUM_OBJECTS >= 90 and NUM_OBJECTS < 100:
            NUM_CONTACTS = 28
        else:
            NUM_CONTACTS = 35
        if soc_dist == 1:
            NUM_CONTACTS = NUM_CONTACTS + 5
        if room_301 == 1:
            NUM_CONTACTS = NUM_CONTACTS + 1
        if room_302 == 1:
            NUM_CONTACTS = NUM_CONTACTS + 1
        if room_303 == 1:
            NUM_CONTACTS = NUM_CONTACTS + 1
        xx = []
        yy1 = []
        yy2 = []
        xx.append(0)
        yy1.append(INFECTED)
        yy2.append(NUM_OBJECTS - INFECTED)
        for i in range(0, 99):
            for j in range(0, NUM_CONTACTS - 1):
                obj_1 = random.randint(0, NUM_OBJECTS - 1)
                obj_2 = random.randint(0, NUM_OBJECTS - 1)
                if obj_1 != obj_2:
                    if objects[obj_1] == 1 and objects[obj_2] == 0:
                        num = random.random()
                        if num <= INF_PROBABILITY:
                            objects[obj_2] = 1
                    if objects[obj_1] == 0 and objects[obj_2] == 1:
                        num = random.random()
                        if num <= INF_PROBABILITY:
                            objects[obj_1] = 1
            xx.append(i + 1)
            yy1.append(objects.count(1))
            yy2.append(objects.count(0))
        Label(root, text='Результати оптимізації', font='Arial 16').grid(column=0, row=19,
        sticky=W)
        Label(root, text='Вибірка:', font='Arial 16').grid(column=0, row=20, sticky=W)
        Label(root, text='Кількість інфікованих:', font='Arial 16').grid(column=0, row=21,
        sticky=W)
        Label(root, text='Кількість здорових:', font='Arial 16').grid(column=0, row=22,
        sticky=W)
        EntryP = Entry(root, width=10, font='Arial 16')
        EntryP.grid(row=20, column=1, padx=5, pady=5, sticky="W")
        EntryQ = Entry(root, width=10, font='Arial 16')
        EntryQ.grid(row=21, column=1, padx=5, pady=5, sticky="W")
        EntryR = Entry(root, width=10, font='Arial 16')
        EntryR.grid(row=22, column=1, padx=5, pady=5, sticky="W")
        EntryP.insert(0, objects.count(1) + objects.count(0))
        EntryQ.insert(0, objects.count(1))

```

```

EntryR.insert(0, objects.count(0))
btn5 = Button(text="Побудувати графік", command = graph2)
btn5.grid(row=23 ,column=1, padx=5, pady=5, sticky="W")
btn6 = Button(text="Симуляція", command = simulation)
btn6.grid(row=23 ,column=2, padx=5, pady=5, sticky="W")
btn7 = Button(text="Нове дослідження", command = new)
btn7.grid(row=24 ,column=1, padx=5, pady=5, sticky="W")
btn8 = Button(text="Вийти", command = exit)
btn8.grid(row=24 ,column=2, padx=5, pady=5, sticky="W")
def graph2():
    global xx
    global yy1
    global yy2
    plt.plot(xx, yy1, 'r')
    plt.plot(xx, yy2)
    plt.show()
def new():
    EntryA.delete(0, END)
    EntryB.delete(0, END)
    EntryC.delete(0, END)
    EntryM.delete(0, END)
    EntryN.delete(0, END)
    EntryO.delete(0, END)
    EntryM.grid_forget()
    EntryN.grid_forget()
    EntryO.grid_forget()
    Label(root, text='Результати оптимізації', font='Arial 16').grid_forget()
if soc_dist.get() == 0:
    RADIUS = 8
elif soc_dist.get() == 1:
    RADIUS = 15
else:
    RADIUS = 15
(WIDTH, HEIGHT) = (1500, 800)
SPEED = 2
RECOVERY=100
WINDOW_COLOR = (255, 255, 255)
WINDOW_TITLE = "Симуляція"
NUM_OBJECTS=IntVar()
INFECTED=IntVar()
INF_PROBABILITY = random.uniform(0.1,\
                                0.7)
RECOVERY_MAX = 1
RECOVERY_MIN = random.uniform(0.5,\
                               0.9)
SYMPTOMATIC_MIN = random.uniform(0.1,\
                                  0.8)
SYMPTOMATIC_MAX = random.uniform(SYMPTOMATIC_MIN,\
                                  0.9)
# activate pygame library
pygame.init()
# create pygame window object and display it
screen = pygame.display.set_mode((WIDTH, HEIGHT))
pygame.display.set_caption(WINDOW_TITLE)
screen.fill(WINDOW_COLOR)
pygame.display.iconify()
#represents a particle
class Particle:
    def __init__(self, x, y, angle):
        self.angle = angle
        self.speed = random.random() # a characteristic of the Particle
        self.current_speed = self.speed # actual speed, adjusted after collisions
        self.collission_time = 0 # time since latest collision
        self.collided = False
        self.x = x
        self.y = y
        self.radius = RADIUS
        self.color = (0, 0, 255)
        self.status = "healthy"
        self.onset = 0 # this doesn't become relevant until infection
        # rates
        self.recovery_probability = random.uniform(RECOVERY_MIN, \
                                                  RECOVERY_MAX)
        self.symptomatic_probability = random.uniform(SYMPTOMATIC_MIN, \
                                                     SYMPTOMATIC_MAX)
        self.infection_probability = INF_PROBABILITY
# draw particle on screen
def display(self):

```

```

#поле для інформації
#pygame.draw.line(screen, (0,0,0),
# [WIDTH - 155, 95],
# [WIDTH, 95], 3)
#pygame.draw.line(screen, (0,0,0),
# [WIDTH - 155, 0],
# [WIDTH - 155, 95], 3)
if variant.get()==1:
    pygame.draw.line(screen, (0,0,0),
    [0, 300],
    [50, 300], 3)
    pygame.draw.line(screen, (0,0,0),
    [150, 300],
    [400, 300], 3)
    pygame.draw.line(screen, (0,0,0),
    [500, 300],
    [600, 300], 3)
    pygame.draw.line(screen, (0,0,0),
    [700, 300],
    [900, 300], 3)
    pygame.draw.line(screen, (0,0,0),
    [1000, 300],
    [1300, 300], 3)

    pygame.draw.line(screen, (0,0,0),
    [0, 500],
    [50, 500], 3)
    pygame.draw.line(screen, (0,0,0),
    [150, 500],
    [600, 500], 3)
    pygame.draw.line(screen, (0,0,0),
    [700, 500],
    [900, 500], 3)
    pygame.draw.line(screen, (0,0,0),
    [1000, 500],
    [1500, 500], 3)

    pygame.draw.line(screen, (0,0,0),
    [200, 0],
    [200, 300], 3)
    pygame.draw.line(screen, (0,0,0),
    [550, 0],
    [550, 300], 3)
    pygame.draw.line(screen, (0,0,0),
    [850, 0],
    [850, 300], 3)
    pygame.draw.line(screen, (0,0,0),
    [1300, 0],
    [1300, 300], 3)

    pygame.draw.line(screen, (0,0,0),
    [400, 500],
    [400, 800], 3)
    pygame.draw.line(screen, (0,0,0),
    [850, 500],
    [850, 800], 3)
#301 closed
if room_301.get()==1:
    pygame.draw.line(screen, (250,0,0),
    [50, 300],
    [150, 300], 3)
if room_302.get()==1:
    pygame.draw.line(screen, (250,0,0),
    [50, 500],
    [150, 500], 3)
if room_303.get()==1:
    pygame.draw.line(screen, (250,0,0),
    [900, 300],
    [1000, 300], 3)
if room_304.get()==1:
    pygame.draw.line(screen, (250,0,0),
    [600, 500],
    [700, 500], 3)
if room_306.get()==1:
    pygame.draw.line(screen, (250,0,0),
    [900, 500],
    [1000, 500], 3)

```

```

pygame.draw.circle(screen, self.color, (int(self.x),
int(self.y)), self.radius)
# update x and y position of particle
def move(self):
    self.x += math.cos(self.angle) * self.current_speed * SPEED
    self.y += -math.sin(self.angle) * self.current_speed * SPEED
# check if particle needs to bounce off the wall
def bounce(self):
    if self.x > WIDTH - self.radius:
        self.x = 2 * (WIDTH - self.radius) - self.x
        self.angle = math.pi - self.angle
    elif self.x < self.radius:
        self.x = 2 * self.radius - self.x
        self.angle = math.pi - self.angle
    if self.y > HEIGHT - 10 - self.radius:
        self.y = 2 * (HEIGHT - 10 - self.radius) - self.y
        self.angle = - self.angle
    elif self.y < self.radius:
        self.y = 2 * self.radius - self.y
        self.angle = - self.angle
    if self.x > WIDTH - 155 - self.radius and self.y > 95 + self.radius and self.y <
105 + self.radius:
        self.y = self.y + self.radius/4
        self.angle = - self.angle
    if self.x > WIDTH - 165 - self.radius and self.x < WIDTH - 155 - self.radius and
self.y < 95 + self.radius:
        self.x = self.x - self.radius/4
        self.angle = math.pi - self.angle
    #в приміщенні
    if variant.get() == 1:
        if self.x < 50 + self.radius and self.y > 290 - self.radius and self.y < 300
- self.radius:
            self.y = self.y - self.radius/4
            self.angle = - self.angle
        if self.x < 50 + self.radius and self.y < 310 + self.radius and self.y > 300
+ self.radius:
            self.y = self.y + self.radius/4
            self.angle = - self.angle
        if self.x > 150 - self.radius and self.x < 400 + self.radius and self.y >
290 - self.radius and self.y < 300 - self.radius:
            self.y = self.y - self.radius/4
            self.angle = - self.angle
        if self.x > 150 - self.radius and self.x < 400 + self.radius and self.y <
310 + self.radius and self.y > 300 + self.radius:
            self.y = self.y + self.radius/4
            self.angle = - self.angle
        if self.x > 500 - self.radius and self.x < 600 + self.radius and self.y >
290 - self.radius and self.y < 300 - self.radius:
            self.y = self.y - self.radius/4
            self.angle = - self.angle
        if self.x > 500 - self.radius and self.x < 600 + self.radius and self.y <
310 + self.radius and self.y > 300 + self.radius:
            self.y = self.y + self.radius/4
            self.angle = - self.angle
        if self.x > 700 - self.radius and self.x < 900 + self.radius and self.y >
290 - self.radius and self.y < 300 - self.radius:
            self.y = self.y - self.radius/4
            self.angle = - self.angle
        if self.x > 700 - self.radius and self.x < 900 + self.radius and self.y <
310 + self.radius and self.y > 300 + self.radius:
            self.y = self.y + self.radius/4
            self.angle = - self.angle
        if self.x > 1000 - self.radius and self.x < 1300 + self.radius and self.y >
290 - self.radius and self.y < 300 - self.radius:
            self.y = self.y - self.radius/4
            self.angle = - self.angle
        if self.x > 1000 - self.radius and self.x < 1300 + self.radius and self.y <
310 + self.radius and self.y > 300 + self.radius:
            self.y = self.y + self.radius/4
            self.angle = - self.angle
        if self.x < 50 + self.radius and self.y > 490 - self.radius and self.y < 500
- self.radius:
            self.y = self.y - self.radius/4
            self.angle = - self.angle

```



```

        if self.x < 200 + self.radius and self.y < 310 - self.radius:
            self.y = 320
            self.angle = - self.angle
        if self.x < 200 + self.radius and self.y > 330 - self.radius and self.y
< 340 - self.radius:
            self.y = self.y + self.radius/4
            self.angle = - self.angle
        if room_302.get()==1:
            if self.x < 400 + self.radius and self.y > 490 - self.radius:
                self.y = 480
                self.angle = self.angle
            if self.x < 300 + self.radius and self.y > 490 - self.radius and self.y
< 500 - self.radius:
                self.y = self.y - self.radius/4
                self.angle = - self.angle
        if room_303.get()==1:
            if self.x > 850 - self.radius and self.x < 1300 + self.radius and self.y
< 310 - self.radius:
                self.y = 320
                self.angle = - self.angle
            if self.x > 850 - self.radius and self.x < 1300 + self.radius and self.y
> 330 - self.radius and self.y < 340 - self.radius:
                self.y = self.y + self.radius/4
                self.angle = - self.angle
        if room_304.get()==1:
            if self.x > 400 - self.radius and self.x < 850 + self.radius and self.y
> 500 - self.radius:
                self.y = 480
                self.angle = - self.angle
            if self.x > 400 - self.radius and self.x < 850 + self.radius and self.y
> 490 - self.radius and self.y < 500 - self.radius:
                self.y = self.y - self.radius/4
                self.angle = - self.angle
        if room_306.get()==1:
            if self.x > 850 - self.radius and self.x < 1300 + self.radius and self.y
> 500 - self.radius:
                self.y = 480
                self.angle = - self.angle
            if self.x > 850 - self.radius and self.x < 1300 + self.radius and self.y
> 490 - self.radius and self.y < 500 - self.radius:
                self.y = self.y - self.radius/4
                self.angle = - self.angle
        # infect this particle, if it is not already
        def infect(self):
            if self.status != "infected":
                num = random.random()
                if num <= self.infection_probability:
                    num = random.random()
                    self.color = (255, 0, 0)
                    #if num <= self.symptomatic_probability:
                    #    self.color = (255, 0, 0)
                    self.status = "infected"
                    self.onset = time.time()
        def infect_particle(self):
            if self.status != "infected":
                num = random.random()
                self.color = (255, 0, 0)
                self.status = "infected"
                self.onset = time.time()
        # check if the particle should recover, if so, recover
        def recover(self):
            if self.status == "infected" and\
time.time() - self.onset > RECOVERY:
                num = random.random()
                if num < self.recovery_probability:
                    self.color = (0, 255, 0)
                    self.status = "recovered"
                    self.infection_probability = random.uniform(0, 0.01)
            else:
                self.status = "dead"
#cellular automata
def cellular_automata(particles):
    for k in range(0, len(particles) - 1):
        if particles[k].status != "infected":
            quantity = 0
            for m in range(0, len(particles) - 1):
                if k!=m:
                    difference_x=particles[k].x-particles[m].x

```

```

        difference_y=particles[k].y-particles[m].y
        dist=math.sqrt(difference_x**2 + difference_y**2)
        if dist < RADIUS * 5:
            quantity = quantity + 1
    if quantity > 2:
        particles[k].infect_particle()
global count_contacts
count_contacts = 0
# adjust direction if centers of any two particles are too close
# param particles is a list of all particles in the window
def collide(particles):
    global count_contacts
    for i in range(0, len(particles) - 1):
        target = particles[i]
        for j in range(i + 1, len(particles)):
            other = particles[j]
            diff_x = target.x - other.x
            diff_y = target.y - other.y
            distance = math.sqrt(diff_x**2 + diff_y**2)
            if distance < RADIUS * 2:
                count_contacts += 1
                # adjust directions (switch velocities)
                temp_angle = target.angle
                temp_speed = target.current_speed
                target.angle = other.angle
                target.current_speed = other.current_speed
                other.angle = temp_angle
                other.current_speed = temp_speed
                target.collison_time = time.time()
                other.collison_time = time.time()
                target.collided = True
                other.collided = True
                if (target.status == "infected" \
                    or other.status == "infected"):
                    target.infect()
                    other.infect()
                # adjust target x and y to be next to other
                if distance != 0:
                    target.x = other.x + diff_x / distance * 2 * RADIUS
                    target.y = other.y + diff_y / distance * 2 * RADIUS
def simulation():
    global NUM_OBJECTS
    global INFECTED
    pygame.display.get_active()
    pygame.display.toggle_fullscreen()
    particles = []
    for i in range(NUM_OBJECTS):
        x = random.randint(RADIUS, WIDTH - RADIUS - 155)
        y = random.randint(RADIUS, HEIGHT - RADIUS - 95)
        angle = random.uniform(0, 2 * math.pi)
        particles.append(Particle(x, y, angle))
    for i in range(INFECTED):
        particles[i].infect_particle()
    # initialize status counts
    healthy = 0
    infected = 0
    #infected = 1
    recovered = 0
    dead = 0
    #create array to store healthy counts
    global healthy_counts
    global infected_counts
    global recovered_counts
    global death_counts
    healthy_counts = []
    infected_counts = []
    recovered_counts = []
    death_counts = []
    # make screen persist until user closes it
    running = True
    start_ticks = pygame.time.get_ticks()
    while running:
        pygame.time.delay(10)
        timer =(pygame.time.get_ticks() - start_ticks)/1000
        for event in pygame.event.get():
            if event.type == pygame.QUIT:
                pygame.quit() # we could also set running = False
    # check if particles run into each other

```

```

collide(particles)
#reset particle status counts
healthy = 0
infected = 0
recovered = 0
# update each particle in our list and the display
for p in particles:
    p.move()
    p.bounce()
    p.recover()
    p.display()
    if p.status == "healthy":
        healthy += 1
    elif p.status == "infected":
        infected += 1
    elif p.status == "recovered":
        recovered += 1
    elif p.status == "dead":
        dead += 1
    particles.remove(p)
# add counts at this point in time, for graphing later
healthy_counts.append(healthy)
infected_counts.append(infected)
recovered_counts.append(recovered)
death_counts.append(dead)
# create font object
font = pygame.font.Font(pygame.font.get_default_font(), 15)
# create Surfaces with text and combine it with the screen
# surface to display status counts
text = font.render("здорові: " + str(healthy), True, (0, 0, 255))
screen.blit(text, (WIDTH - 200, 10))
text = font.render("інфіковано: " + str(infected), True, (255, 0, 0))
screen.blit(text, (WIDTH - 200, 30))
text = font.render("час: " + str(round(timer, 1)), True, (255, 0, 255))
screen.blit(text, (WIDTH - 200, 50))
text = font.render("кількість контактів: " + str(count_contacts), True, (255, 0,
255))
screen.blit(text, (WIDTH - 200, 70))
if variant.get()==1:
    # create font object
    #font = pygame.font.Font(pygame.font.get_default_font(), 15)
    text = font.render("301", True, (150, 75, 0))
    screen.blit(text, (100, 150))
    text = font.render("кафедра комп'ютерних наук", True, (150, 75, 0))
    screen.blit(text, (250, 150))
    text = font.render("кабінет завідуючого кафедром", True, (150, 75, 0))
    screen.blit(text, (575, 150))
    text = font.render("303", True, (150, 75, 0))
    screen.blit(text, (1100, 150))
    text = font.render("302", True, (150, 75, 0))
    screen.blit(text, (150, 650))
    text = font.render("304", True, (150, 75, 0))
    screen.blit(text, (550, 650))
    text = font.render("306", True, (150, 75, 0))
    screen.blit(text, (1150, 650))
# update the display
pygame.display.update()
screen.fill(WINDOW_COLOR)
# once no particles are infected, stop running and display graph
if (infected == NUM_OBJECTS or timer > 10):
    running = False
    #pygame.display.update()
    #screen.fill(WINDOW_COLOR)
    pygame.display.iconify()
    sim_graph()
def sim_graph():
    global healthy_counts
    global infected_counts
    # plot infections over time
    #healthy_counts.plot()
    #plt.plot(x, x**2, Label="здорові")
    #plt.plot(x, x**3, Label="інфіковані")
    plt.plot(healthy_counts)
    plt.plot(infected_counts, 'r')
    plt.xlabel("Час")
    plt.ylabel("Кількість здорових/інфікованих")
    plt.title("Залежність кількості здорових та інфікованих від часу")
    #plt.legend()

```

```
plt.savefig('saved_fig.jpg')
plt.show()
#save graph
plt.savefig('saved_fig.jpg')
#json file
#data = '{"experiment_id": "0", "building_config":' + (str)variant + ', "particles":
NUM_OBJECTS, "infected_particles": INFECTED, "min_inf": min_inf}'
#with open('data111.json', 'w') as f:
#json.dump([{"experiment_id": '0', "building_config": variant, "particles":
NUM_OBJECTS, "infected_particles": INFECTED, "min_inf": min_inf}], f)
#json.dump(data, f)
#simulation()
main()
```

Додаток Г

Наукові публікації

COLLECTION OF SCIENTIFIC PAPERS
SCIENTIA

CERTIFICATE OF PARTICIPATION

Certificate provides at least a 0.1 ECTS credits to awarded participants for being involved

Vitalii Alekseiko

participated in the I International Scientific and Theoretical Conference
**CURRENT ISSUES OF SCIENCE,
PROSPECTS AND CHALLENGES**

Scan the code to get access to the conference proceedings



 December 17, 2021
Sydney, Australia

The conference is included in the Academic Resource Index ResearchBib catalog and UKRISTEI catalog (Certificate № 226 dated February 25th, 2021);

Head of the European Scientific Platform
Chairman of the Organizing committee
MARIIA HOLDENBLAT



 EUROPEAN SCIENTIFIC PLATFORM

Conference proceedings are publicly available under terms of the Creative Commons Attribution 4.0 International License (CC BY 4.0).

COLLECTION OF SCIENTIFIC PAPERS
SCIENTIA

17 DECEMBER, 2021
SYDNEY, AUSTRALIA

**CURRENT ISSUES OF SCIENCE,
PROSPECTS AND CHALLENGES**

I INTERNATIONAL SCIENTIFIC AND THEORETICAL CONFERENCE

VOLUME 2

 EUROPEAN SCIENTIFIC PLATFORM 

DOI 10.36074/scientia-17.12.2021 ISBN 978-1-68564-135-1 00002

COLLECTION OF SCIENTIFIC PAPERS
SCIENTIA

17 December, 2021
Sydney, Australia

**CURRENT ISSUES OF SCIENCE,
PROSPECTS AND CHALLENGES**

I International Scientific and Theoretical Conference

VOLUME 2

Sydney, 2021

SECTION 18.
SYSTEM ANALYSIS, MODELING AND OPTIMIZATION

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ РЕСПІРАТОРНИХ ЗАХВОРИВАНЬ В НЕВЕЛИКИХ ПОПУЛЯЦІЯХ
Алексейко В.О.38

SECTION 19.
INFORMATION TECHNOLOGIES AND SYSTEMS

FEATURES OF MANAGEMENT OF INFRASTRUCTURE PROJECTS AT THE CURRENT STAGE
Guseva-Bozhalkina V., Kozyrko A., Nazarko A.41

PROOF OF ZERO-KNOWLEDGE IN THE TASKS OF ANONYMIZATION OF FINANCIAL TRANSACTIONS
Prokhorov E.43

ВДОСКОНАЛЕННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ КОНТРОЛЮ ТЕМПЕРАТУРНОГО РЕЖИМУ ПРИ ВИГОТОВЛЕННІ ЦЕЛІ
Лукашук Г.О., Козогова М.А.45

ОТРАБОТКА КОНЦЕПТА МНОГОУРОВНЕВОГО МУЛЬТИПЛЕКСА ДАНИХ ГІБРИДНОГО СТЕГА АЛОГОРИТМА
Лесная Ю.Е., Гончаров Н.А., Малахов С.В.48

SECTION 20.
PHYSICS AND MATHEMATICS

CATALAN NUMBERS AND THEIR APPLICATION IN PROBABILITY THEORY
Spektriv D.56

ВІЗУАЛЬНИЙ КОНСПЕКТ НА ТЕМУ «КОМПЛЕКСНІ ЧИСЛА»
Костенко Я.С.59

РЕАЛІЗАЦІЯ АСИНХРОННОГО ЕТАПУ ДИСТАНЦІЙНОГО ЗАНЯТТЯ З ТЕМИ «ПОХІДНА ТА ПІЗАСТОСУВАННЯ»
Зікратий Д.О.61

SECTION 21.
PHILOLOGY AND JOURNALISM

NEGATIVE PREFIXES IN ENGLISH AND UZBEK LANGUAGES
Iskandarova Sh., Karimova Sh., Abdulkarimova M.63

SOCIO-CULTURAL AND DISCOURSE ASPECTS OF LANGUAGE COMPETENCIES IN TEACHING AND LEARNING FOREIGN LANGUAGES
Eshanova M.66

TEACHING GRAMMAR COMMUNICATIVELY: INTERRELATIONS BETWEEN PRAGMATICS, SPEECH ACTS AND GRAMMAR
Sadikov E.T.69

THE CONCEPTS OF "GENRE", "FORM" AND "FORMAT" IN ONLINE JOURNALISM
Hrozna O.73

THE ROLE OF THE MODERN PEDAGOGICAL TECHNOLOGIES IN FOREIGN LANGUAGE TEACHING
Isomov O.A. o'g'li.75

4

БЕЛГІШ КАЗАК СЫНШЫСЫ ҚАЛЖАН НҮРМАХАНОВТЫҢ ШЫҒАРМАШЫЛЫҒЫНДАҒЫ ҚАЗАК ЖӘНЕ УКРАИН ӘДЕБИЕТТЕРІНІҢ ӨЗАРА БАЙЛАНЫСЫ ТУРАЛЫ
Нурмаханова Маржан Қалжанқызы, Советова Зуре Советқызы 77

БЕЛГІШ СЫНШЫ ҚАЛЖАН НҮРМАХАНОВТЫҢ ЕҢБЕКТЕРІНДЕГІ КӨРКЕМ АУДАРМАНЫҢ БАРАБАРШЫҒЫН САҚТАУ МӨСЕЛЕСІ (XX ҒАСЫРДЫҢ 50-60-ЖЫЛДАРЫ)
Нурмаханова Маржан Қалжанқызы, Советова Зуре Советқызы 80

ЛЕКСИЧНІ ІННОВАЦІЇ У ПОЛІТРИЧНОМУ ЛЕКСИКОНІ ІВАНА ФРАНКА (НА МАТЕРІАЛІ ПРАЦІ ІВАНА ФРАНКА «ЩО ТАКЕ ПОСТУП?» У ЗІСТАВЛЕННІ ЗІ «СЛОВНИКОМ УКРАЇНСЬКОЇ МОВИ» ЗА РЕДАКЦІЄЮ Б. ГРИЧЕНКА)
Разумна А.А. 82

ОСОБЛИВОСТІ НАПРЯМКІВ КИТАЙСЬКОЇ СЕРЕДНЬОВІСНОЇ ПОЕЗІЇ ЖАНРУ ЦИ
Русаконя К.О., Дашенко Г.В. 85

СИСТЕМНІ ВІДНОШЕННЯ У ФРАЗЕОЛОГІЗМАХ УКРАЇНСЬКОЇ МОВИ З КОМПОНЕНТОМ «ЯЗИК»
Ладава С.П. 88

SECTION 22.
PHILOSOPHY AND POLITICAL SCIENCE

PROBLEMS OF THE INFLUENCE OF POLITICAL ISLAM ON SOCIAL STABILITY
Behzod Soipov 94

THE OVERTON WINDOW AND THE SOCIOLOGY OF FRAMES - A POST-PHILOSOPHICAL THINKING OF THE EXPANDING BOUNDARIES OF THE PERMISSIBLE POSSIBLE
Kuruchenko M.S. 99

АБСЕНТЕІЗМ У ПОЛІТИЦІ: СВІТОВИЙ ДОСВІД ПОДОЛАННЯ
Гусева Н.Ю. 102

СВРАЗІЙСЬКІ СПОДІВАННЯ З. БЖЕЗІНСЬКОГО ТА ГЕОПОЛІТИЧНІ РЕАЛІЇ
Понков В.В., Хамід Фуад 104

5

SECTION 18.
SYSTEM ANALYSIS, MODELING AND OPTIMIZATION

Алексейко Віталій Олександрович
здобувач вищої освіти факультету інформаційних технологій
Хмельницький національний університет, Україна
Науковий керівник: Скрипник Тетяна Казимирівна
старший викладач кафедри комп'ютерних наук
Хмельницький національний університет, Україна

**МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ
РЕСПІРАТОРНИХ ЗАХВОРИВАНЬ
В НЕВЕЛИКИХ ПОПУЛЯЦІЯХ**

Респіраторні віруси інфекції – це найбільш масові захворювання, які займають провідне місце у структурі інфекційних хвороб і складають 80-90 % від загальної кількості захворювань. Враховуючи здатність вірусу грипу викликати щорічні епідемії у масштабах всієї земної кулі, можна стверджувати, що грип є проблемою світового значення. Під час епідемії хворіє від 5% до 20% населення, а у період пандемічного розповсюдження грипу – інфікується абсолютна більшість населення [2].

Під час чхання або кашля, з рота хворої людини випають дрібні частки слини і мокротиння, у яких віруси містяться у величезних кількостях. Тому основний механізм передачі грипу та ГРВІ називається – повітряно-крапельний. Інший механізм передачі респіраторної інфекції – контактний. Він довгий час залишався невідомим і менш очевидним, ніж повітряно-крапельний. Тим не менш, він відіграє не меншу, а можливо, і більшу роль у поширенні захворювань. Як правило, людина, яка чхає або кашляє, прикриває рот рукою, сподіваючись запобігти поширенню інфекції повітряно-крапельним шляхом. При цьому вона і не підозрює, наскільки полегшує передачу своєї інфекції контактним шляхом. Це пояснюється тим, що вся колональна маса мікробів, яка повинна була виліти у відкритий простір, оскільки на руці людини, що чхає або кашляє. Після цього вони розносяться на предмети побуту, зокрема в ті, яких торкаються інші люди. Також передача збудників відбувається при рукошлясаннях. Після дотику руки до рота, носа або очей, які також вкриті сприйнятливою до ГРВІ слизовою оболонкою, і складний повітряно-крапельний шлях передачі скорочується для вірусу за часом і складністю в десятки разів.

Одним із найбільш ефективних шляхів боротьби з поширенням респіраторних захворювань може стати моделювання їх розповсюдження. Методи математичної епідеміології стосуються здебільшого, так званої, «кількісної теорії», тобто знаходження чисельних розв'язків розроблених математичних моделей з метою визначення епідемічних порогів та прогнозування перебігу захворювань [4]. У той же час залишаються недостатньо вивченими задачі якісного аналізу епідеміологічних процесів. Таким чином, практично всі методи побудови математичних моделей для епідеміології мають багаті спільного, що в свою чергу відображається і на самих моделях, які мають більше спільних властивостей, ніж відмінностей, але залишаються недостатньо вивченими задачі якісного аналізу епідеміологічних процесів, які дають відповіді на запитання про стійкість та стабілізацію епідемії [1].

Розглянемо можливість зараження при наближенні двох людей на відстані l . Ця відстань буде залежати від виду вірусу та його життєстійкості. Більшість вірусів швидко гинуть поза організмом, проте деякі можуть виживати протягом декількох днів і навіть місяців. Повітряно-крапельний шлях передачі передбачає вдихання розпиленних у повітрі частинок слизу внаслідок чхання чи кашлю інфікованим у радіусі 0,9 метра. При дрібнодисперсному розпиленні з виникненням частинок діаметром менше 10 мкм радіус може збільшитися до 1,8 метра і більше. Таким чином ймовірність інфікування при дотриманні дистанції більше двох метрів дуже незначна і нею можна знехтувати. Тому доцільно приймати l рівним 2 м. В той же час навіть при контакті з хворим ймовірність захворювання в здорової людини, у якій відсутній імунітет не рівний 100%. Для визначення, хворіє особа чи ні використовується теорія ймовірностей.

Розглянемо подію A – «людина захворіє». Ймовірність того, що подія A справдиться $P(A)$. Числове значення такої ймовірності визначається експериментальним шляхом або на основі статистичних даних. Проте ці дані не можна вважати абсолютно точними. Тому при моделюванні поширення ГРВІ варто розглядати діапазон ймовірностей від мінімальної ймовірності зараження $P_{min}(A)$ до максимальної $P_{max}(A)$. Отже значення $P(A)$ обирається випадковим чином з діапазону $[P_{min}(A), P_{max}(A)]$.

Припустимо, що на певному обмеженому просторі перебуває деяка кількість людей протягом досить тривалого часу. Якщо серед цієї групи перебуватиме хоча б одна інфікована особа, яка безпосередньо контактує з іншими, існує ризик сплеску захворювання. Отже, кожна особа отримує статус: «вразливі» або «хворий».

Для прогнозування поширення захворювання необхідно визначити кількість контактів між людьми за одиницю часу. Слід зауважити, що на початковому етапі більшість контактів будуть відбуватися між здоровими людьми. Проте, з часом кількість хворих буде зростати. В свою чергу зростання кількості хворих у вибірці, що досліджується, призведе до збільшення кількості контактів хворих та здорових, внаслідок чого кількість хворих різко зростатиме.

Позначимо залежність кількості здорових людей, які є вразливими до хвороби від часу $S(t)$. Аналогічно залежність кількості інфікованих від часу – $I(t)$. Оскільки популяція, що розглядається є невеликою, то швидкість зміни частки вразливих $\frac{dS(t)}{dt}$ пропорційна $S(t) \cdot I(t)$ з коефіцієнтом пропорційності β , що чисельно рівний значенню $P(A)$. Звідси отримуємо рівняння (1) залежності кількості вразливих від часу:

$$\frac{dS(t)}{dt} = -\beta \cdot S(t) \cdot I(t). \quad (1)$$

Знак мінус вказує на зменшення кількості вразливих, оскільки вони змінюють свій статус на «хворий» і не можуть інфікуватися повторно, поки через певний проміжок часу знову не стануть вразливими. В той же час кількість хворих зростає. Таким чином рівняння (2) залежності кількості хворих від часу:

$$\frac{dI(t)}{dt} = \beta \cdot S(t) \cdot I(t). \quad (2)$$

Об'єднавши рівняння (1) і (2) отримуємо модель для невеликої популяції, яка задана системою рівнянь (3):

$$\begin{cases} \frac{dS(t)}{dt} = -\beta \cdot S(t) \cdot I(t), \\ \frac{dI(t)}{dt} = \beta \cdot S(t) \cdot I(t). \end{cases} \quad (3)$$

Де:
 $S(t)$ – залежність кількості здорових людей, які є вразливими до хвороби від часу;
 $I(t)$ – залежність кількості хворих людей від часу;
 β – ймовірність інфікування вразливої людини при контакті з хворим.

Таким чином, математичне представлення біологічних процесів забезпечує прозорість і точність по відношенню до епідеміологічних припущень, дозволяючи перевірити розуміння епідеміології захворювання шляхом порівняння результатів моделі і закономірностей, що спостерігаються [3]. Модель також може допомогти в прийнятті рішень, прогнозуючи важливі питання, такі як зміни в розповсюдженні захворювання.

Сьогодні існує безліч респіраторних хвороб, що описуються різноманітними математичними моделями, проте не існує ідеальної моделі, і при використанні математичного моделювання поширення захворювання, модель має бути адаптована до відповідних умов.

Отже, важлива роль математичних моделей полягає в тому, що вони можуть попередити нас про недоліки епідеміології різних захворювань та сформулювати мету та завдання для подальших досліджень.

Список використаних джерел:

1. Goh, B. (1977). Global stability in many-species systems, *Amer. Naturalist*, p.135 – 143. Retrieved from <https://www.jstor.org/stable/2459985>.
2. Longini, I., Nizam, A., Xu, S., Ungchaisak, K., Hanshaworakul, W., Cummings, D. & Halloran, M. E. (2004). Containing pandemic influenza at the source, *Science* 309. P. 623 – 633. Retrieved from <https://pubmed.ncbi.nlm.nih.gov/16079251/>.
3. May, R. M. (2004). Uses and abuses of mathematics in biology. *Science*. N. 303. P. 790 – 793. Retrieved from https://www.researchgate.net/publication/8884371_Use_and_Abuses_of_Mathematics_in_Biology.
4. Myers, M., Rogers, D., Cox, J., Flahault, A. & Hay, S. (2000). Forecasting Disease Risk for Increased Epidemic Preparedness in Public Health, *Advances in Parasitology*, Vol. 47, p. 309 – 330. Retrieved from <https://www.sciencedirect.com/science/article/abs/pii/S0065308X00470132>.


SECTION 19. INFORMATION TECHNOLOGIES AND SYSTEMS

Guseva-Bozhatkina Victoria 

Senior Lecturer, Department of Automated Systems Software,
Admiral Makarov National University of Shipbuilding, Ukraine

Kozyrko Anastasiya

higher education seeker
Admiral Makarov National University of Shipbuilding, Ukraine

Nazarko Artem 

higher education seeker
Admiral Makarov National University of Shipbuilding, Ukraine

FEATURES OF MANAGEMENT OF INFRASTRUCTURE PROJECTS AT THE CURRENT STAGE

Current trends in the global development of project-oriented organizations implementing infrastructure projects are accompanied by the development of their competence (technological maturity) [7].

Difficulties in developing an adequate strategy for the implementation of organizational development projects form new strategies and ways of qualitative change. Currently, the issue of assessing the readiness of management teams for the successful implementation of infrastructure projects comes to the fore. This is especially true during the "infodemic" that accompanies the coronavirus pandemic and the global recession. Such global external challenges significantly increase the uncertainty of the environment and affect the processes of strategy formation and analysis [1].

Lack of informal communication, language barriers, differences in national cultural traditions, including differences in ethical norms and laws of countries, can often serve as barriers to trust [5].


In conditions of economic downturn, anti-crisis management of infrastructure projects, based on the analysis of problems, determining their priorities and finding approaches to their solution [3].

The technology of strategic management of infrastructure projects is determined by the peculiarities of the combined influence of external and internal factors, their relationship in time and space, stages of turbulence and crisis, the speed of their development. In the field of processes and technologies, extremely important management elements are: clearly defined strategies, mobility and dynamism in use.

resources and change, realization of innovative potential of integrated value chains and application of modern information technologies [7].


In an infrastructure project created to develop a new type of product (service), the use of new technologies is crucial [7].

The success of infrastructure projects depends on how widely and effectively modern knowledge and breakthrough innovative technologies are used. In infrastructure projects, traditional factors (land, labor and capital) operate only if the effective use of modern knowledge and technology [3].



СЕРТИФІКАТ

учасника конференції



IN 8020
від 10.12.2021

АЛЕКСЕЙКО ВІТАЛІЙ ОЛЕКСАНДРОВИЧ

ВЗЯВ(-ЛА) УЧАСТЬ У ІІІ МІЖНАРОДНІЙ СТУДЕНТСЬКІЙ НАУКОВІЙ КОНФЕРЕНЦІЇ
АКТУАЛЬНІ ПИТАННЯ ТА ПЕРСПЕКТИВИ
ПРОВЕДЕННЯ НАУКОВИХ ДОСЛІДЖЕНЬ


10 ГРУДНЯ 2021 РІК • М. ТРУСКАВЕЦЬ, УКРАЇНА

В рамках участі було опубліковано тези доповіді учасника на тему:

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ІНФЕКЦІЙНИХ ЗАХВОРИВАНЬ

ДИРЕКТОР МОЛОДІЖНОЇ НАУКОВОЇ ЛІГИ
ГОЛОВА ОРГКОМІТЕТУ КОНФЕРЕНЦІЇ

ІГОР КОРЕНЮК



МАТЕРІАЛИ ІІІ МІЖНАРОДНОЇ СТУДЕНТСЬКОЇ НАУКОВОЇ КОНФЕРЕНЦІЇ

АКТУАЛЬНІ ПИТАННЯ ТА
ПЕРСПЕКТИВИ ПРОВЕДЕННЯ
НАУКОВИХ ДОСЛІДЖЕНЬ

ТОМ 3



М. ТРУСКАВЕЦЬ, УКРАЇНА

10 ГРУДНЯ
2021 РІК

МОЛОДІЖНА
НАУКОВА
ЛІГА

МАТЕРІАЛИ ІІІ МІЖНАРОДНОЇ СТУДЕНТСЬКОЇ НАУКОВОЇ КОНФЕРЕНЦІЇ

АКТУАЛЬНІ ПИТАННЯ ТА
ПЕРСПЕКТИВИ ПРОВЕДЕННЯ
НАУКОВИХ ДОСЛІДЖЕНЬ

3 ТОМ

м. Трускавець, Україна
10 грудня 2021 рік

Вінниця, Україна
«Європейська наукова платформа»
2021

Актуальні питання та перспективи проведення наукових досліджень | Том 3

ПОРІВНЯЛЬНИЙ АНАЛІЗ СПОСОБІВ ІНТЕГРАЦІЇ СИСТЕМ Малачов К.Д., Науковий керівник: Хацько Н.С.	34
СПОСОБИ КЕРУВАННЯ НАВАНТАЖЕННЯМ ПРИ ТЕСТУВАННІ ПРОДУКТИВНОСТІ Гайсва Н.Л.	37
СЕКЦІЯ 21. СИСТЕМНИЙ АНАЛІЗ, МОДЕЛЮВАННЯ ТА ОПТИМІЗАЦІЯ	
ДОСЛІДЖЕННЯ ВІДКРИТИХ СИСТЕМ ДЛЯ РОЗВИТКУ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ Могилюк М.Л., Науковий керівник: Кузьбовський І.І.	40
МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ІНФЕКЦІЙНИХ ЗАХВОРЮВАНЬ Алексейко В.О., Науковий керівник: Скрипник Т.К.	42
РЕТРОСПЕКТИВА РОЗВИТКУ МЕТОДІВ ПРОЄКТУВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ Король І.В., Науковий керівник: Кузьбовський І.І.	44
СИСТЕМНІ ТЕХНОЛОГІЇ ПРОЄКТУВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ Явчук М.М., Науковий керівник: Кузьбовський І.І.	46
СЕКЦІЯ 22. ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА СИСТЕМИ	
FEATURES OF SATELLITE NAVIGATION SYSTEM FOR AUTOMOTIVE TRANSPORT Трофімова У.У., Scientific adviser: Fedik L. Y.	49
MODEL OF COMPETENCE LEVELS OF THE LOCAL COMMUNITY IN PROJECTS TO IMPROVE MUNICIPAL ENERGY EFFICIENCY Sorochan V., Scientific adviser: Sachenko O.	51
OVERVIEW AND PROBLEMS OF BIOMETRIC AUTHENTICATION METHODS Yezhova Y.A.	53
АВТОМАТИЗАЦІЯ ВЕНТИЛЯЦІЙНОЇ СИСТЕМИ Чайка Д.А., Науковий керівник: Шибко О.М.	56
ВИКОРИСТАННЯ 3D ТЕХНОЛОГІЙ У WEB РОЗРОБЦІ Гуменюк Д.В.	59
ВИКОРИСТАННЯ СИСТЕМИ АВТОМАТИЗОВАНОГО КЕРУВАННЯ В ПРОЄКТУВАННІ СИСТЕМ АВТОМАТИЗАЦІЇ Тривичук Б.В., Науковий керівник: Фефік Л.Ю.	61
ВРАХУВАННЯ ГНУЧКОСТІ ТА АДАПТИВНОСТІ WEB-ДОДАТКУ ПРИ ВИКОРИСТАННІ У БУДЬ-ЯКІЙ СФЕРІ ДІЯЛЬНОСТІ Риваль І.Т.	63
МЕТОДИ ВИЯВЛЕННЯ АНОМАЛІЙ В БІОСИГНАЛАХ НА ОСНОВІ МАШИНОГО НАВЧАННЯ Дека В.	65
ПЕРЕВАГИ ТА НЕДОЛІКИ АВТОМАТИЗАЦІЇ ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ Сухий Д.Р.	67

4

10 грудня 2021 рік • Трускавець, Україна • Молодіжна наукова ліга

ТЕОРЕТИЧНІ АСПЕКТИ СТВОРЕННЯ ТА ВИКОРИСТАННЯ ДИСТАНЦІЙНИХ ОСВІТНИХ ТЕХНОЛОГІЙ Шибко Д.О., Науковий керівник: Шибко О.М.	69
ІШТУЧНИЙ ІНТЕЛЕКТ ДЛЯ БЕЗПЕКИ КІБЕРПРОСТОРУ Юрченко О.О., Науковий керівник: Кузьовий О.Я.	71
СЕКЦІЯ 23. ФІЗИКО-МАТЕМАТИЧНІ НАУКИ	
ВІБЕКВЕСТ З АНАЛІТИЧНОЇ ГЕОМЕТРІЇ Костенко Ю.В., Науковий керівник: Жиленко Т.І.	74
РЕАЛІЗАЦІЯ МІЖПРЕДМЕТНИХ ЗВ'ЯЗКІВ НА УРОКАХ МАТЕМАТИКИ ТА ІНФОРМАТИКИ Прудка А.О.	76
СКЕЧНОУНІТІНГ ЯК ІНФОРМАЦІЙНИЙ ІНСТРУМЕНТ СТУДЕНТА НА ЗАНЯТТЯХ З ВИЩОЇ МАТЕМАТИКИ Сосолова А.П., Науковий керівник: Жиленко Т.І.	78
СЕКЦІЯ 24. СОЦІОЛОГІЯ ТА СТАТИСТИКА	
РОЗВИТОК СОЦІОЛОГІЧНОЇ ДУМКИ В УКРАЇНІ НА СУЧАСНОМУ ЕТАПІ Токаренко І.О., Науковий керівник: Наушкіна О.А.	80
СЕКЦІЯ 25. ФІЛОЛОГІЯ ТА ЖУРНАЛІСТИКА	
АКТУАЛЬНІСТЬ ГАСТРОНОМІЧНОЇ ТРЕВЕЛ-ЖУРНАЛІСТИКИ В СУЧАСНОМУ ТЕЛЕВІЗІЙНОМУ ПРОСТОРІ Гужва О.С., Науковий керівник: Мітюк О.А.	83
АТЕІСТИЧНИЙ ДЕКАДАНС ВАЛЕРІЯ МАРЧЕНКА (НА МАТЕРІАЛАХ ЗБІРКИ В. МАРЧЕНКА «ЛІСТИ ДО МАТЕРІ З НЕВОЛІ») Довгішова В.С., Науковий керівник: Раріцький О.А.	85
ВИКОРИСТАННЯ ІКТ НА УРОКАХ ІНОЗЕМНОЇ МОВИ ЯК СПОСІБ ПІДВИЩЕННЯ МОТИВАЦІЇ УЧНІВ Даченко Т.О., Науковий керівник: Трухан М.М.	88
ВИКОРИСТАННЯ СЛОВОТВОРЧИХ СУФІКСІВ У СУЧАСНІЙ АНГЛІЙСЬКІЙ МОВІ Зиченко К., Науковий керівник: Сітко А.В.	91
ВПЛИВ КАТЕГОРІЇ РОДУ НА ВИЗНАЧЕННЯ ГЕНДЕРНОЇ ПРИНАЛЕЖНОСТІ ІМЕННИКІВ Надточій Д.А., Науковий керівник: Сітко А.В.	93
ДО ПИТАННЯ НАУКОВОГО СТИЛЮ СУЧАСНОЇ АНГЛІЙСЬКОЇ МОВИ Шарапаюк Ю.А., Науковий керівник: Білецька І.О.	95
ДО ПИТАННЯ ПЕРЕКЛАДУ ФРАЗЕОЛОГІЗМІВ Юрик А.Л., Науковий керівник: Сидоренко Ю.І.	98
ЛЕКСИКО-СЕМАНТИЧНІ ОСОБЛИВОСТІ ПЕРЕКЛАДУ АНГЛОМОВНИХ КАЗОК Василенко К.А., Науковий керівник: Корольова Т.М.	101

5

Алексейко Віталій Олександрович, здобувач вищої освіти
факультету інформаційних технологій
Хмельницький національний університет, Україна

Науковий керівник: Скрипник Тетяна Казимирівна, старший викладач
кафедри комп'ютерних наук
Хмельницький національний університет, Україна

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ІНФЕКЦІЙНИХ ЗАХВОРЮВАНЬ

Серед великої кількості захворювань, одними з найбільш критичних для населення є хвороби, що передаються від людини до людини повітряно-крапельними шляхами через швидкість їх поширення, масштаб, велику кількість хворих і померлих тощо. Сьогодні актуальною проблемою людства стала коронавірусна інфекція Covid-19, вона є лише частковим випадком більшості інфекційних захворювань, які можуть з'явитися найближчим часом.

Попереджувальні заходи є ключовими у вирішенні цього завдання. Прогнозування динаміки розповсюдження захворювань дозволяє розробити і застосувати адекватні заходи протидії, забезпечити раціональне використання матеріальних і людських ресурсів. Усе більше дослідників звертаються до математичних моделей для створення прогнозів інфекційних захворювань, так вони допомагають найбільш вірно і точно вивчити зміни певних процесів, які відбуваються в соціумі.

Математичне моделювання розповсюдження інфекційних захворювань має давню історію. У 1840 році W. Farr за допомогою кривої нормального розподілу описав дані смертей від віспи в Англії та Уельсі за 1837-1839 р.р. В 1906 році John Brownlee продовжив роботу над цим методом, співставивши ряди епідеміологічних даних на основі розподілу Пірсона, про що описав в статті "Статистичний підхід до імунного захисту: теорія епідемії" [1]. Найбільш відома модель в математичній епідеміології – це проста модель SIR, описана W.O. Kermack та A.G. McKendrick в 1927 році. У ній за допомогою систем диференціальних рівнянь (неперервний час) або різницевих рівнянь (дискретний час) описується динаміка груп сприйнятливих, інфікованих та осіб, які одужали [4]. У цій моделі популяція поділяється на три групи людей: сприйнятливих до захворювання (Susceptible), інфікованих (Infected) і осіб, які одужали після захворювання і до нього не сприйнятливих (Recovered). Припускають, що розповсюдження інфекційного захворювання проходить через контакт сприйнятливих з інфікованими, а ті що одужали перестають розповсюджувати хворобу. Поступово, на основі SIR-моделі була побудована ціла ієрархія математичних моделей розповсюдження інфекційних захворювань (SI, SIS, SEI, SIRS, SEIS, SEIR, SEIRS, MSIR, MSEIR, MSEIRS, LISEIR) [2, 3], які враховують велику кількість груп епідемічного процесу і параметрів, які впливають на перебіг епідемії.

Робоча група з математичного моделювання проблем, пов'язаних з епідемією коронавірусу SARS-CoV-2 в Україні (створена розпорядженням Президії НАН України від 3 квітня 2020 р. № 118) розробила модель SEIR-U. З початку квітня

2020 р. міжвідомчою Робочою групою представників НАН України, КНУ імені Тараса Шевченка та НАМН України, з урахуванням світового досвіду математичного моделювання розвитку епідемії COVID-19, на основі статистичних даних про динаміку епідемії в Україні та країнах Європи створювалася і тестувалася математична модель SEIR-U. Модель SEIR-U, як інші моделі класу SEIR, розраховує балансові відношення для чотирьох компартментів основних категорій населення під час епідемії: S – сприйнятливий до захворювання, E – інфікований, але без проявів хвороби, I – інфікований з підтвердженою хворобою, R – ті, що вже не можуть захворіти, бо мають імунітет до хвороби чи померли. Така схема може нарощуватись введенням додаткових компартментів: наприклад, частина категорії S, яка на суворому карантині не хворіє, підкатегорія R, яка отримала вакцинації, підкатегорія I, яка не має симптомів і не потребує госпіталізації, і підкатегорія з госпіталізованих, яка потребує вентиляції легень, та інші підкатегорії. В силу гнучкості такої структури моделі SEIR адаптують для специфічних особливостей різних інфекційних хвороб. Поточна реалізація моделі SEIR-U дозволяє враховувати наявність безсимптомних хворих, має 3 рівні складності протікання хвороби для хворих із симптомами, дозволяє обчислювати кількість пацієнтів, що знаходяться на госпіталізації

Математичний апарат SEIR моделей – чисельний розв'язок систем детермінованих чи стохастичних звичайних диференціальних рівнянь. Моделі, що базуються на диференціальних рівняннях, посідають особисте місце серед існуючих моделей поширення інфекційних захворювань. Диференціальні динамічні моделі дають змогу не тільки проводити прогнозування розвитку ситуації, але й оцінити різні епідеміологічні параметри, а також ефективність тих чи інших протипісляхворобних заходів. Математичне представлення біологічних процесів забезпечує прозорість і точність по відношенню епідеміологічних припущень, дозволяючи перевірити розуміння епідеміології захворювання шляхом порівняння результатів моделі і закономірностей, що спостерігаються [5]. Модель також може допомогти в прийнятті рішень, прогнозуючи важливі питання, такі як зміни в розповсюдженні захворювання.

Отже, важлива роль математичних моделей полягає в тому, що вони можуть попередити нас про недоліки епідеміологічних рішень інфекційних захворювань та сформулювати мету та завдання для подальших досліджень.

Список використаних джерел:


1. Brownlee J. Statistical studies in immunity: the theory of an epidemic // Proceedings of the Royal Society of Edinburgh 26.1, 1906. P. 484-521.
2. Edelstein-Keshet L. Mathematical Models in Biology. Society for Industrial and Applied Mathematics, 2005, 586 pp.
3. Herbert W., Hethcote H.W., "The Mathematics of Infectious Diseases", SIAM Review, 42:4 (2000), 599-653.
4. Kermack W.O., McKendrick A.G. A contribution to the mathematical theory of epidemics // Proc R Soc Lond. 1927. N. 115. P. 700-721.
5. May R.M. Uses and abuses of mathematics in biology // Science. 2004. N. 303. P. 790-793.



СЕРТИФІКАТ

учасника конференції

IN 5029
від 19.11.2021



АЛЕКСЕЙКО ВІТАЛІЙ ОЛЕКСАНДРОВИЧ


ВЗЯВ(-ЛА) УЧАСТЬ У ІІ МІЖНАРОДНІЙ СТУДЕНТСЬКІЙ НАУКОВІЙ КОНФЕРЕНЦІЇ
ПРІОРИТЕТНІ НАПРЯМКИ ТА ВЕКТОРИ
РОЗВИТКУ СВІТОВОЇ НАУКИ

19 ЛИСТОПАДА 2021 РІК • М. ДРОГОБИЧ, УКРАЇНА

В рамках участі було опубліковано тези доповіді учасника на тему:
**ДЕЯКІ АСПЕКТИ МАТЕМАТИЧНОГО ТА ІМІТАЦІЙНОГО
МОДЕЛЮВАННЯ В МЕДИЦИНІ**

ДИРЕКТОР МОЛОДІЖНОЇ НАУКОВОЇ ЛІГИ
ГОЛОВА ОРГКОМІТЕТУ КОНФЕРЕНЦІЇ

ІГОР КОРЕНЮК







Пріоритетні напрямки та вектори розвитку світової науки | Том 2

СЕКЦІЯ 14. ЕКОЛОГІЯ ТА ТЕХНОЛОГІЇ ЗАХИСТУ НАВКОЛИШНЬОГО СЕРЕДОВИЩА	
ДОТРИМАННЯ ЕКОЛОГІЧНИХ ВІМОГ ПРИ ОЧИЩЕННІ МІСЬКИХ СПІЧНИХ ВОД Мовчан А.П., Горбань Д.Г., Науковий керівник: Горносталь С.А.	30
ПОВИДЖЕННЯ З ПОБУТОВИМИ ВІДХОДАМИ В МАЛИХ МІСТАХ УКРАЇНИ НА ПРИКЛАДІ МІСТА ДОЛІНСЬКА, КІРОВОГРАДСЬКОЇ ОБЛАСТІ Поворозок І.І., Науковий керівник: Доценко Л.В.	34
СЕКЦІЯ 15. КОМП'ЮТЕРНА ТА ПРОГРАМНА ІНЖЕНЕРІЯ	
TARGETED ADVERTISING AS AN EFFECTIVE METHOD OF INCREASING DEMAND Maidebura D., Supervisor of diploma work: Melnyk K.	37
АНАЛІЗ ПРИКЛАДНОГО ДОМЕНУ В АВТОМАТИЗОВАНОМУ ТЕСТУВАННІ Гайсіва Н.І.	40
ВИЯВЛЕННЯ СПУФІНГ У ТА ФЕЙКІВ ЗА ДОПОМОГОЮ ДОДАТКОВИХ ПОЯСНЕНЬ ШЕПЛІ НА БАЗІ ТЕОРІЇ ІГОР Шевченко О.Т.	43
ОСОБЛИВОСТІ ВИКОРИСТАННЯ КРОСПЛАТФОРМЕННОСТІ У РОЗРОБЦІ ПРОГРАМНИХ ДОДАТКІВ Куциш В.Ю., Науковий керівник: Сердюк Н.М.	45
ПРОЄКТУВАННЯ ТА РОЗРОБЛЕННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ ОБЛІКУ ДАНИХ ПРО СТУДЕНТІВ Ганжа А.С., Науковий керівник: Антоненко С.В.	47
СЕКЦІЯ 16. ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА СИСТЕМИ	
THE PRINCIPLE OF OPERATION OF THE ELECTROCARDIOGRAPH Serdechenko I., Chernykh M., Hryh O., Supervisor: Kompanets N.	49
АНАЛІЗ МЕДИЧНИХ ЗОБРАЖЕНЬ ДЛЯ ДІАГНОСТУВАННЯ ХВОРОБИ Коваль Б.В., Науковий керівник: Кузьомін О.Я.	51
ВИКОРИСТАННЯ НАЙВНОГО АЛГОРИТМУ БАЙЄСА Болтов Д.Р., Науковий керівник: Кузьомін О.Я.	54
ВИКОРИСТАННЯ НЕЙРОННИХ МЕРЕЖ ДЛЯ РОЗПІЗНАВАННЯ ОБ'ЄКТІВ НА ПРИКЛАДІ ІМАГЕНЕТ Ситало А.В., Науковий керівник: Вельмагіна Н.О.	57
ДЕЯКІ АСПЕКТИ МАТЕМАТИЧНОГО ТА ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ В МЕДИЦИНІ Александров В.О.	59
ІНФОРМАЦІЙНІ РЕСУРСИ В СУЧАСНОМУ СУСПІЛЬСТВІ Шибко Д.О., Науковий керівник: Шибко О.М.	61
ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ WEB- ОРІЄНТОВАНИХ ПРОГРАМНИХ ПРОДУКТІВ Пустовіт О.В., Науковий керівник: Трус Ю.В.	64

4

19 листопада 2021 рік • Дрогобич, Україна • Молодіжна наукова ліга

МЕТОДИ ІПТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ДОСЛІДЖЕННЯ МУТАЦІЇ COVID-19 Музыка Р.В., Науковий керівник: Кузьомін О.Я.	66
ПРОГРАМНІ ТА АПАРАТНІ МЕТОДИ ГЕНЕРУВАННЯ КРИПТОВАЛЮТ Винту Д., Науковий керівник: Політаський Р.Л.	69
ПРОЄКТУВАННЯ І РОЗРОБКА ІГРОВОГО ДОДАТКУ З ВИКОРИСТАННЯМ ДОПОВНЕНОЇ РЕАЛЬНОСТІ (AUGMENTED REALITY) Випирський І.Ю., Науковий керівник: Даченко П.В.	72
ТЕСТУВАННЯ ЯК МЕТОД КОНТРОЛЮ І ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАЛЬНОЇ ДІЯЛЬНОСТІ УЧНІЯ НА УРОКАХ ІНФОРМАТИКИ Пруда А.О.	73
УПРАВЛІННЯ ІТ-ПРОЕКТОМ З РОЗРОБКИ ІГРОВОГО ДОДАТКУ З ВИКОРИСТАННЯМ ДОПОВНЕНОЇ РЕАЛЬНОСТІ (AUGMENTED REALITY) Слущкий О.О., Науковий керівник: Даченко П.В.	75
СЕКЦІЯ 17. ТРАНСПОРТ ТА ТРАНСПОРТНІ ТЕХНОЛОГІЇ	
ПЕРСПЕКТИВИ РОЗВИТКУ ВНУТРІШНЬОГО ВОДНОГО ТРАНСПОРТУ УКРАЇНИ Мора С.М., Жилин К.В., Лютов З.В.	76
СЕКЦІЯ 18. ФІЗИКО-МАТЕМАТИЧНІ НАУКИ	
FTIR-ATR SPECTROSCOPY FOR STUDYING LIPID RAFTS TARGETING BY ANTIVIRAL DRUG REMDESIVIR Abolina U., Supervisor: Dovbeshko G.	78
STUDIES OF MITOCHONDRIA - FTIR AND NANOIR Hnatiuk S.I., Supervisor: Dovbeshko G.I.	82
ВИКОРИСТАННЯ ВІЗУАЛЬНОГО КОНСПЕКТУ ПРИ ВИВЧЕННІ ТЕМИ «ПОВЕРХНІ ДРУГОГО ПОРЯДКУ» Сокрута С.О., Науковий керівник: Жиленко Т.І.	86
ВИКОРИСТАННЯ ІГРОВИХ ФОРМ НАВЧАННЯ НА ЗАНЯТТЯХ З ВИЩОЇ МАТЕМАТИКИ Вишченко О.С., Науковий керівник: Жиленко Т.І.	88
ВИКОРИСТАННЯ ІНТЕРАКТИВНИХ ПЛАКАТІВ НА ЗАНЯТТЯХ З ВИЩОЇ МАТЕМАТИКИ Копил Д.А., Науковий керівник: Жиленко Т.І.	89
ЗАСТОСУВАННЯ СИСТЕМИ ДЕСЛІ В ЕКОЛОГІЇ Єфремова А.А., Науковий керівник: Жовтківська І.М.	91
СКЕТЧНОУТІНГ ЯК ІНФОРМАЦІЙНИЙ ХЕНДМЕЙД СТУДЕНТА Кучач П., Науковий керівник: Жиленко Т.І.	94

5

об'єкт, що шукається на картинці або правильно класифікувати картинку.

На процес навчання нейромережі впливає кількість, різноманітність матеріалів та якість самих зображень. Тому для навчання нейронної мережі а образ базу даних ImageNet. Ця велика база складається з тисяч різноманітних категорій зображень, а самі категорії містять сотні зображень, які можна використовувати для навчання мережі [3].

Список використаних джерел:

1. <https://habr.com/ru/post/348000/>
2. <https://neurohive.io/ru/osnovny-data-science/glubokaya-svetochnaja-nejronnaja-set/>
3. <https://www.image-net.org/>

Алексейко Віталій Олександрович, здобувач вищої освіти факультету інформаційних технологій Хмельницький національний університет, Україна

ДЕЯКІ АСПЕКТИ МАТЕМАТИЧНОГО ТА ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ В МЕДИЦИНІ

Збереження та зміцнення здоров'я населення є важливим соціально-економічним завданням, невід'ємним аспектом якого є зниження рівня різних захворювань. Попереджувальні заходи є ключовими у вирішенні цього завдання. Прогнозування динаміки розвитку захворювань дозволяє розробити і застосувати адекватні заходи протидії, забезпечити раціональне використання матеріальних і людських ресурсів.

Сучасні інформаційні технології сприяють розвитку галузей наук і техніки, пов'язаних з медициною. Актуальною проблемою в сучасній медицині є імітаційне моделювання. Це метод дослідження, при якому система, що вивчається, замінюється моделлю, яка з достатньою точністю описує процеси так, як вони проходили б насправді. Імітація – це розуміння суті явища, не вдаючись до експериментів на реальному об'єкті. Імітаційне моделювання відкриває нові можливості для моделювання стану хворого під час важких оперативних втручань, відновлення порушених функцій організму, удосконалення методик проведення операцій, зменшення економічних витрат та витрат часу.

Імітаційне моделювання – це окремий випадок математичного моделювання. Математична модель – абстрактне відображення нашого уявлення про систему або про процес, представлена у вигляді математичних символів, формул, рівнянь, комп'ютерна модель – це математична модель, записана на мові програмування і реалізована у вигляді програми для електронно-обчислювальної машини [2].

Коли математична модель побудована, математичний аналіз в подальшому і комп'ютерним моделюванням, допомагає нам дослідити глобальну поведінку моделі, виявляючи наслідки зроблених припущень [3].

Метод математичного моделювання – це важливий інструмент вивчення закономірностей, що лежать в основі функціонування складних систем довільної природи, і біомедичних зокрема. Основний принцип математичного моделювання складних систем – принцип оптимальності. Модель має бути максимально простою, тобто має містити мінімальну кількість змінних (відповідно і рівнянь), а також мати порівняно прості зв'язки між змінними. Важливою процедурою, яку використовують розробники моделей для перевірки надійності передбачень, зроблених математичною моделлю є порівняння різних моделей [4].

Математичне моделювання незамінне у певних галузях медицини, де неможливо, або складно проводити реальні експерименти, наприклад, в епідеміології. Математичне моделювання в прогнозуванні питань розповсюдження захворювань може надати суттєву допомогу в прийнятті рішень їх профілактики, лікування, прогнозування. При математичному моделюванні скорочується час дослідження, оскільки за короткий проміжок часу комп'ютер розбирає велику кількість можливих варіантів дослідження. Математична модель спрощує розв'язання завдань з лікування хвороб, оскільки прискорюється аналіз вхідних

даних та вирішуються завдання, які можуть виникнути в процесі лікування.

Математичне моделювання дозволяє без значних матеріальних витрат досліджувати поведінку біологічних систем у таких умовах, що складно відтворити в умовах експерименту або клініки, прогнозувати деякі нові явища, скоротити час дослідження і забезпечити оптимальну методику для лікування захворювань [1].

Завдання ендпротезування вирішуються за допомогою імітаційного моделювання. Відомі основні особливості застосування імітаційного моделювання в медико-технічних системах, а системи автоматизованого проектування можуть використовуватися для оцінювання працездатності технічних виробів. Моделювання ендпротезу дозволяє підвищити рівень надійності виробу. Наприклад, математичне моделювання тестових електрокардіосигналів неможливо здійснити без опрацювання з застосуванням сучасних інформаційних технологій.

Моделювання знаходить широке застосування в галузі медицини не тільки через те, що може замінити експеримент. Воно має велике самостійне значення, що виражається в цілому ряді переваг: за допомогою методу моделювання на одному комплексі даних можна розробити цілий ряд різних моделей, по-решному інтерпретувати досліджуване явище, і вибрати найбільш вдалу з них для теоретичного тлумачення; у процесі побудови моделі можна зробити різні доповнення до досліджуваної гіпотези й дістати її спрощення; у випадку складних математичних моделей можна застосовувати електронно-обчислювальні машини [1].

Підсумовуючи вищевикладене можна зробити висновок, що математичне та імітаційне моделювання в медицині дає новий поштовх для розвитку медичної науки.

Напрями подальших досліджень пов'язані з розробленням імітаційної моделі респіраторних захворювань.

Список використаних джерел:

1. Загальні поняття в моделюванні біомедичних процесів та систем: метод, вказівки для самостійного вивчення дисципліни «Моделювання біологічних процесів і систем» для студентів денної форми навчання першого рівня вищої освіти (бакалавр), спеціальностей 163 Біомедична інженерія, 208 Агроінженерія; Харків. нац. техн. У-т сіл. Госп-на ім. П. Василенка; уклад. В. О. Шигімага – Харків [6. в.]. 2020. – 28 с.
2. Мезенцева Л.В., Перцов С.С. Математическое моделирование в биомедицине // Вестник новых медицинских технологий. 2013. Т. XX. № 1. С. 11.
3. Hippert G, Katriel. Mathematical modelling and prediction in infectious disease epidemiology // Clinical Microbiology and Infection. 2013. Vol. 19. N 11. P. 999–1005
4. Merler S., Ajelli M., Pugliese A., Ferguson N.M. Determinants of the spatiotemporal dynamics of the 2009 H1N1 pandemic in Europe: implications for real-time modelling // PLoS Comput Biol. 2011. Vol. 7. N 9: e1002205.

Шибко Дмитро Олександрович, здобувач вищої освіти факультету інформаційних технологій та механіки інженерії Придніпровська державна академія будівництва та архітектури, Україна

Науковий керівник: Шибко Оксана Миколаївна, канд. техн. наук, доцент, доцент кафедри комп'ютерних наук, інформаційних технологій та прикладної математики

Придніпровська державна академія будівництва та архітектури, Україна

ІНФОРМАЦІЙНІ РЕСУРСИ В СУЧАСНОМУ СУСПІЛЬСТВІ

На початку третього тисячоліття інформація стає одним із найбільш важливих ресурсів. Зростаюча залежність від наявності інформації, рівня розвитку та ефективності використання засобів її обробки та передачі призвела до виникнення такого принципово нового поняття як інформаційні ресурси.

За своєю значимістю інформаційні ресурси не тільки не поступаються іншим видам стратегічної сировини, але і відносяться до найбільш важливих ресурсів.

Інформаційний ресурс – це інтелектуальний ресурс, фактор колективної творчості і головні труднощі в розумінні його природи і функцій поділяють у розкритті механізму перетворення знань на силу, способів його впливу на матеріальні фактори прогресу.

Виділяють сім основних особливостей інформаційних ресурсів:

1. на відміну від інших видів ресурсів (зокрема матеріальних) інформаційний ресурс практично невичерпний;
2. з використанням інформаційний ресурс не зникає, а зберігається і навіть збільшується;
3. інформаційний ресурс не є самостійним і сам по собі має лише потенційне значення. Тільки поєднуючись з іншими ресурсами – досвідом, працею, кваліфікацією, технікою, енергією, сировиною - він є рушійною силою;
4. ефективність застосування інформаційного ресурсу пов'язана з ефектом повторного виробництва знань. Інформаційна взаємодія дозволяє одержати нові знання ціною менших витрат, порівняно з витратами праці, енергії, часу на його пряме генерування;
5. інформаційний ресурс є формою безпосереднього включення науки до складу виробничих сил. В індустріальному суспільстві наука виступає опосередкованою і безпосередньою продуктивною силою. Посередником при цьому виступають машини, нові матеріали і препарати. Радикально змінюється характер дії інформаційного ресурсу: продуктивність праці підвищується;
6. інформаційні ресурси виникають в результаті не просто розумової праці, а її творчої частини;
7. перетворення знань в інформаційному ресурсі залежить від можливості їхнього кодування.

Інформаційні ресурси можна класифікувати за такими ознаками:

1. приналежність ресурсу до певної організаційно-технологічної системи

Додаток Д

Презентаційний матеріал



Факультет інформаційних технологій
Кафедра комп'ютерних наук

Кваліфікаційна робота бакалавра

Імітаційне моделювання розповсюдження респіраторних захворювань

Керівник: Завідувач кафедри комп'ютерних наук,
д.т.н., професор Бармак Олександр Володимирович
Студент групи КН-18-1 Алексейко Віталій Олександрович

Хмельницький 2022



Актуальність

- ❑ Гострі респіраторні захворювання – найбільш поширені інфекційні захворювання, що вражають усі вікові групи населення.
- ❑ Імітаційне моделювання захворювань є актуальною проблемою в сучасному світі.
- ❑ Математичне моделювання незамінне в певних галузях медицини, де неможливі або утруднені реальні експерименти, наприклад в епідеміології.



Що таке епідемія?

Епідемія (від грецького $\epsilon\pi\acute{\iota}$ – «серед» і $\beta\epsilon\mu\omicron\varsigma$ – народ; застаріле – пошесть, мор, моровиця) – тип хвороби, яка є новою для даної популяції впродовж періоду збереження імунної «пам'яті» та поширюється зі швидкістю, що значно перевищує очікувану, ґрунтуючись на попередньому досвіді (тобто, числі нових випадків за одиницю часу).




Завдання

Мета кваліфікаційної роботи:

проведення аналізу існуючих моделей поширення інфекційних захворювань, розробка власної імітаційної моделі, її валідація та верифікація за допомогою створення відповідної інформаційної системи.

Результатом виконання кваліфікаційної роботи є застосунок, побудований на основі імітаційної моделі. Виходячи з поставленої мети, в роботі необхідно вирішити наступні завдання :

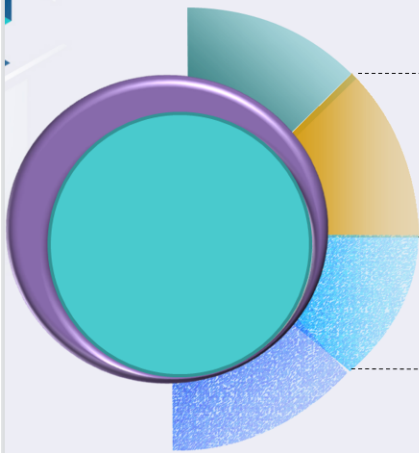
- 
- розглянути теоретичні аспекти поширення інфекційних захворювань
 - розглянути математичні методи, за допомогою яких можна описати характер поширення захворювань, що передаються повітряно-крапельним шляхом
 - розглянути технології, які можуть бути використані при розробці імітаційної моделі респіраторних захворювань і перспективи їх використання
 - застосувати SI модель для побудови математичної моделі, що описує перебіг гострих респіраторних захворювань
 - застосувати теорію ймовірностей для оптимізації і кращої апроксимації моделі
 - створити інформаційну систему
 - провести експерименти і їх аналіз для виявлення закономірностей

Парадигми імітаційного моделювання

системна динаміка

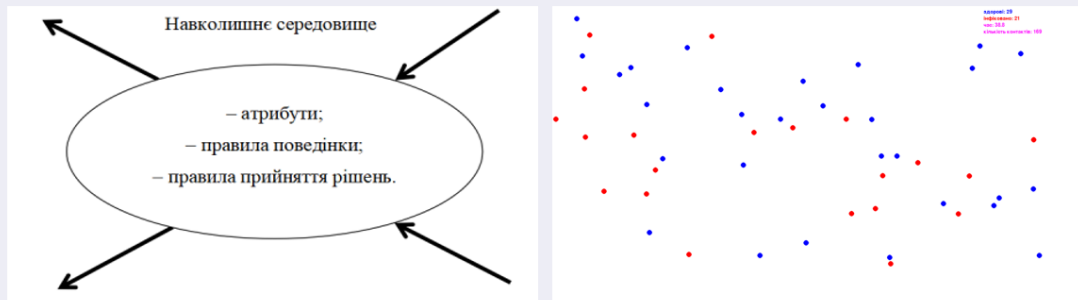
дискретно-подійне моделювання

агентні моделі



Характеристики агента

- 1 Розмір, стан (відображається кольором)
- 2 Правила взаємодії між агентами та зовнішнім середовищем
- 3 Зміна стану (відображається зміною кольору)



SI модель

Позначимо залежність кількості здорових людей, які є вразливими до хвороби від часу. Аналогічно залежність кількості інфікованих від часу. Оскільки популяція, що розглядається є невеликою, то швидкість зміни частки вразливих пропорційна з коефіцієнтом пропорційності, що чисельно рівний значенню $P(A)$.

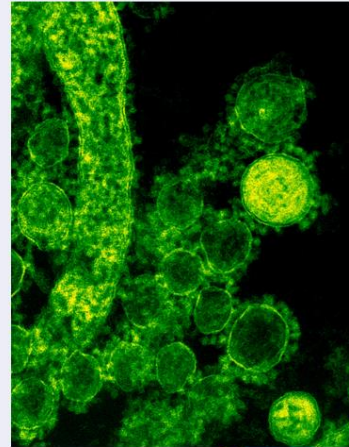
$$\frac{d\tilde{S}(t)}{dt} = -\beta \cdot \tilde{S}(t) \cdot \tilde{I}(t).$$

$$\frac{d\tilde{I}(t)}{dt} = \beta \cdot \tilde{S}(t) \cdot \tilde{I}(t).$$

$$\begin{cases} \frac{d\tilde{S}(t)}{dt} = -\beta \cdot \tilde{S}(t) \cdot \tilde{I}(t), \\ \frac{d\tilde{I}(t)}{dt} = \beta \cdot \tilde{S}(t) \cdot \tilde{I}(t). \end{cases}$$

Теорія ймовірності

Розглянемо подію A – «людина захворіє». Ймовірність того, що подія A справдиться $P(A)$. Числове значення такої ймовірності визначається експериментальним шляхом або на основі статистичних даних. Проте ці дані не можна вважати абсолютно точними. Тому при моделюванні поширення ГРВІ варто розглядати діапазон ймовірностей від мінімальної ймовірності зараження $P_{\min}(A)$ до максимальної $P_{\max}(A)$. Отже значення $P(A)$ обирається випадковим чином з діапазону $[P_{\min}(A); P_{\max}(A)]$.



Послідовність дій користувача



Місце для дослідження:
 - Відкритий простір
 - ХНУ, 3 корпус, 3 поверх, праве крило
 Розмір вибірки: 50
 Початкова кількість інфікованих: 3
 Ймовірність інфікування: 0.256058892 > 0.1-0.3 < 0.3-0.5 < 0.5-0.7

Результати дослідження
 Вибірка: 50
 Кількість інфікованих: 49
 Кількість здорових: 1

Результати оптимізації
 Вибірка: 24
 Кількість інфікованих: 3
 Кількість здорових: 21

Результат моделювання

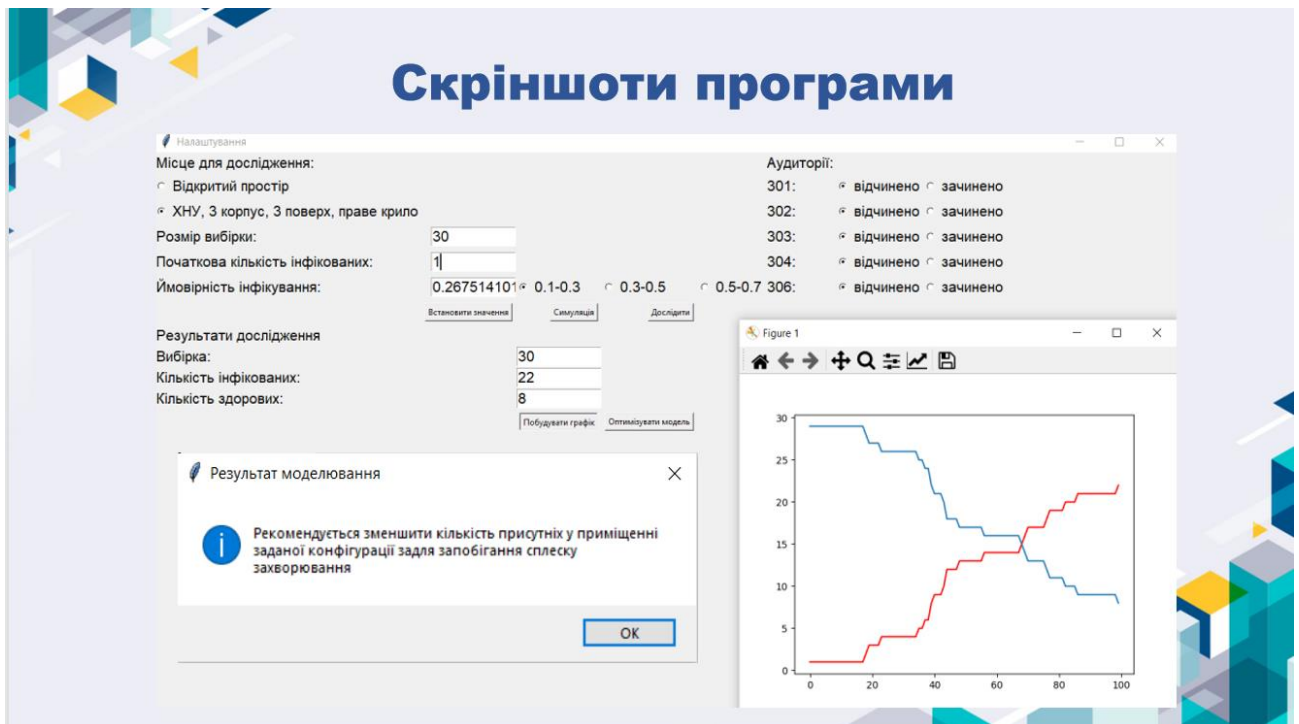
Задана кількість присутніх у приміщенні заданої конфігурації є допустимою, за умови дотримання соціальної дистанції

OK

Скріншоти програми



Скріншоти програми





Дякую за увагу!

Anti-Plagiarism v-15.257

Максимальное совпадение с одним документом 2.0%

Словари проверки: en_US, ru_RU, ua_UA. **Ошибок в документах: 8%**

ID: 104117 Название: КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА Імітаційне моделювання розповсюдження респіраторних захворювань Добавлено в БД: 2022-05-27 Авторы: В.О. Алексейко Руководители: О.В. Бармак Консультанты: Оponentы:	Документ		Суммарное совпадение по Базе Данных	
	Символы	Лексемы	Символы	Лексемы
	63961	989	2313 (4%)	41 (4%)

Источник плагиата

ID	Описание	Наличие плагиата в документе	
		Символы	Лексемы

Ім'я користувача:
Кафедра КН

ID перевірки:
1011350305

Дата перевірки:
27.05.2022 16:37:39 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
27.05.2022 16:38:39 EEST

ID користувача:
100005671

Назва документа: Алексейко_ЗАПИСКА_short

Кількість сторінок: 60 Кількість слів: 10077 Кількість символів: 77938 Розмір файлу: 1.90 MB ID файлу: 1011236218

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

13.9% Схожість

Найбільша схожість: 2.89% з Інтернет-джерелом (<https://www.umj.com.ua/article/6986/gostri-respiratorni-zaxvoryuvan..>

11.8% Джерела з Інтернету

165

Сторінка 62

2.9% Джерела з Бібліотеки

77

Сторінка 63

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0% Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи

1

Підозріле форматування

15
сторінок

**РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ КАФЕДРИ КОМП'ЮТЕРНИХ НАУК
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ**

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Імітаційне моделювання розповсюдження респіраторних захворювань

Автор: студент групи КН-18-1 Алексейко Віталій Олександрович

Спеціальність: 122 – Комп'ютерні науки

Освітня програма: освітньо-професійна

Науковий керівник: д.т.н., проф. Бармак О.В.

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

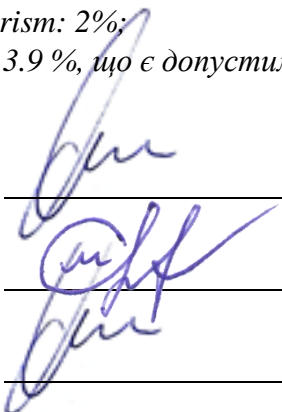
Запозичення, виявлені в роботі Алексеяка В.О., не є плагіатом, оскільки: запозичення розміщені в розділі огляду існуючих підходів, не описують безпосередньо авторську роботу і не стосуються її результатів; усі запозичення фрагментарні; до запозичень входять фрагменти програмного коду, що не мають авторства і містять поширені конструкції; серед запозичень знаходяться загальновідомі терміни та скорочення.

Обсяг запозичень, визначений системами виявлення збігів/ідентичності/схожості, складає:

- за системою Anti-Plagiarism: 2%;

- за системою Unichек: 13.9 %, що є допустимими запозиченнями які відносяться до описаних вище.

Керівник роботи



Олександр БАРМАК

Гарант ОП

Олександр МАЗУРЕЦЬ

Завідувач кафедри КН

Олександр БАРМАК



ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
МОН УКРАЇНИ

Кафедра комп'ютерних наук



РЕЦЕНЗІЯ

на кваліфікаційну роботу бакалавра

студента *гр. КН-18-1* Алексейка Віталія Олександровича

за темою: Імітаційне моделювання розповсюдження респіраторних захворювань

1. Актуальність обраної теми

Гострі респіраторні захворювання (ГРЗ) – найбільш поширені інфекційні захворювання, що вражають усі вікові групи населення. Попереджувальні заходи є ключовими у вирішенні цього завдання. Прогнозування динаміки розповсюдження захворювань дозволяє розробити і застосувати адекватні заходи протидії, забезпечити раціональне використання матеріальних і людських ресурсів. Імітаційне моделювання захворювань є актуальною проблемою в сучасному світі. Усе більше дослідників звертаються до математичних моделей для створення прогнозів тих чи інших захворювань, так як вони допомагають найбільш вірно і точно вивчити зміни певних процесів, які відбуваються в соціумі.

2. Повнота розкриття мети та завдань роботи

Мета та завдання розкриті у повному обсязі. Обґрунтування викладені чітко та зрозуміло. Усі поставлені завдання виконані у повній мірі.

3. Зміст кожного розділу роботи

Робота складається із вступу, трьох розділів, висновків та додатків.

У першому розділі проаналізовано предметну область, її інформаційне забезпечення. Розглянуто існуюче програмне забезпечення, а також аспекти розвитку математичного моделювання для опису поширення респіраторних захворювань відповідно до історичних викликів.

Другий розділ присвячений створенню імітаційної моделі на основі SI моделі. Детально розкрито етапи створення імітаційної моделі. Спроектовано функціональну модель системи та її інформаційну структуру. Обрано найоптимальніші засоби розробки інформаційної системи.

У третьому розділі описано структуру та функціональне призначення складових інформаційної системи, наведено детальну інструкцію користувача та вимоги до розгортання інформаційної системи. Опрацьовано отримані в ході дослідження результати та проведено порівняння із статистичними даними.

Загальні висновки відповідають поставленим завданням дослідження.

4. Оцінка розробленої інформаційної системи, її практична цінність

Отримана в результаті проведених досліджень імітаційна модель та програмна реалізація її візуалізації може бути використана для підвищення ефективності виявлення розповсюдження гострих респіраторних захворювань у закритих приміщеннях.

Практичне значення роботи полягає у тому, що її результати можуть бути використані для прогнозування поширення респіраторних захворювань при проведенні масових заходів та підборі оптимальної кількості присутніх у приміщеннях відповідних

конфігурації. Результати проведеного дослідження оприлюднені на міжнародних студентських наукових конференціях: «Пріоритетні напрямки та вектори розвитку світової науки» (Дрогобич, 2021 р.), «Актуальні питання та перспективи проведення наукових досліджень» (Трускавець, 2021 р.), «Current issues of science, prospects and challenges» (Sydney, Australia, 2021 р.)

5. Якість оформлення кваліфікаційної роботи бакалавра

Робота оформлена згідно вимог

6. Недоліки кваліфікаційної роботи бакалавра

Зважаючи на актуальність теми дослідження, його наукову новизну, практичне значення, обґрунтованість висновків, вважаємо що кваліфікаційна робота бакалавра Алексея Віталія Олександровича «Імітаційне моделювання розповсюдження респіраторних захворювань» є завершеною, самостійно виконаною науковою працею, що має вагомим теоретичне і прикладне значення.

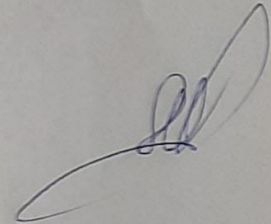
7. Загальний висновок (допускається чи не допускається до захисту), та оцінка на яку заслуговує кваліфікаційна робота.

Враховуючи рівень виконання та забезпечення усіх необхідних вимог, робота може бути допущена до захисту. Рекомендована оцінка «відмінно».

Рецензент професор д.т.н.

професор кафедри комп'ютерної інженерії ІІІУ

Лисенко Сергій Миколайович





ВІДГУК НАУКОВОГО КЕРІВНИКА на кваліфікаційну роботу бакалавра

студента гр. КН-18-1 Алексейка Віталія Олександровича

за темою: Імітаційне моделювання розповсюдження респіраторних захворювань

1. Актуальність теми

Гострі респіраторні захворювання – найбільш поширені інфекційні захворювання, що вражають усі вікові групи населення. Імітаційне моделювання захворювань є актуальною проблемою в сучасному світі. Усе більше дослідників звертаються до математичних моделей для створення прогнозів тих чи інших захворювань, так як вони допомагають найбільш вірно і точно вивчити зміни певних процесів, які відбуваються в соціумі. Математичне моделювання незамінне в певних галузях медицини, де неможливі або утруднені реальні експерименти, наприклад в епідеміології.

2. Відповідність роботи предметній області Стандарту спеціальності 122 Комп'ютерні науки

За стандартом, а саме описом предметної області, об'єктами вивчення та діяльності є математичні, інформаційні, імітаційні моделі реальних явищ, об'єктів, систем і процесів та методи і технології отримання, зберігання, обробки, передачі та використання інформації. Метою роботи саме є розробка імітаційної моделі розповсюдження респіраторних захворювань. При вирішенні поставленої задачі використано математичні моделі, методи та алгоритми розв'язання теоретичних і прикладних задач, що виникають при розробці інформаційних технологій. Тому результати виконання кваліфікаційної роботи бакалавра відповідають стандарту бакалавра спеціальності 122 – Комп'ютерні науки.

3. Професійні та особистісні якості бакалавра

При роботі над кваліфікаційною роботою бакалавра Алексейко Віталій Олександрович проявив себе кваліфікованим фахівцем та дисциплінованим студентом, вчасно виконуючи поставлені етапи дослідження. Як в процесі написання пояснювальної записки, так і при розробці прикладного програмного забезпечення проявив достатні для одержання успішного результату компетентності та результати навчання. Опанував професійні скіли за напрямком «Комп'ютерні науки» та достатньо значний софт скіл.

4. Ступінь самостійності під час виконання кваліфікаційної роботи

Одержані в роботі результати є наслідком особистої діяльності студента, який самостійно виконував всі поставлені задачі.

5. Ступінь оволодіння методами дослідження

При реалізації кваліфікаційної роботи показав достатній рівень компетентностей та володіння необхідними інструментами та обладнанням, методами, методиками та технологіями предметної області комп'ютерних наук.

6. Повнота та якість розкриття теми роботи

Тема роботи в повній мірі обґрунтована й розкрита, проведено аналіз актуальності та відомих досліджень в межах обраної теми, поставлені завдання, які у роботі виконані, та розроблено програмне забезпечення для валідації та верифікації запропонованого імітаційного моделювання.

7. Логічність, послідовність, аргументованість, літературна грамотність викладення матеріалу

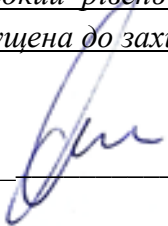
Структура роботи та послідовність викладення логічні та відповідають поставленій меті. Викладення матеріалу послідовне, аргументоване, літературно грамотне.

8. Можливість практичного застосування кваліфікаційної роботи бакалавра, окремих її частин

Розроблена у роботі імітаційна модель та програмна реалізація її візуалізації може бути використана для підвищення ефективності виявлення розповсюдження гострих респіраторних захворювань у закритих приміщеннях.

9. Висновок про можливість допуску кваліфікаційної роботи бакалавра до захисту, на яку оцінку заслуговує робота

Враховуючи високий рівень виконання та забезпечення усіх необхідних вимог, робота може бути допущена до захисту. Рекомендована оцінка «**відмінно**».

Керівник _____  _____ д.т.н., проф. зав. каф. КН Олександр БАРМАК