НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ

імені ІГОРЯ СІКОРСЬКОГО»

Кафедра системного програмування і

спеціалізованих комп’ютерних систем

**КУРСОВИЙ ПРОЄКТ**

з дисципліни «Комп’ютерні мережі»

на тему: Маршрутизація в мережі передачі даних

Студентки \_IV\_ курсу, групи\_КВ-11\_\_

за спеціальностю

123 «Комп’ютерна інженерія»

\_\_\_\_\_\_\_Нестерук А.О.\_\_\_\_\_\_

Керівник: доцент, к.т.н., доц. Орлова М.М.

Національна оцінка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кількість балів: \_\_\_\_\_ Оцінка: ECTS \_\_\_\_\_

Члени комісії \_\_\_\_\_\_\_\_\_\_ доцент, к.т.н. Орлова М.М.

\_\_\_\_\_\_\_\_\_\_\_ доцент, к.т.н. Щербина О.А.

Київ- 2024 рік

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ**

**«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ**

**імені ІГОРЯ СІКОРСЬКОГО»**

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра системного програмування і спеціалізованих комп‘ютерних систем

Рівень вищої освіти – перший (бакалаврський)

Спеціальність 123 «Комп’ютерна інженерія»

**ЗАВДАННЯ**

**на курсовий проєкт студента**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Нестерук А.О.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(прізвище, ім’я, по батькові)

1. Тема проєкту Маршрутизація в мережі передачі даних

керівник проєкту к.т.н., доцент Мартинова Оксана Петрівна ,

2. Термін подання студентом проєкту \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

3. Вихідні дані до проєкту: Мінімум 25 комунікаційних вузлів, 1 канал – супутниковий, Середній ступінь мережі -3.5, ваги каналів - 1, 2, 3, 5, 7, 11, 12, 15, 18 , 32, алгоритм - метод каталогів, орієнтованих на сеанс (пакетна мережа загального призначення Tymnet)

4. Зміст пояснювальної записки:

Календарний план

|  |  |  |  |
| --- | --- | --- | --- |
| №  з/п | Назва етапів виконання  дипломного проєкту | Термін виконання  етапів проєкту | Примітка |
| 1. | Отримання теми та завдання на курсовий проект |  | Виконано |
| 2. | Вивчення літератури за тематикою проекту |  | Виконано |
| 3. | Виконання завдання 1 |  | Виконано |
| 4. | Виконання завдання 2 |  | Виконано |
| 5. | Виконання завдання 3 |  | Виконано |
| 6. | Виконання завдання 4 |  | Виконано |
| 7. | Подання курсового проекту на перевірку |  | Виконано |
| 8. | Захист курсового проекту |  | Виконано |

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Анастасія НЕСТЕРУК

(підпис)

Керівник проекту \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Оксана МАРТИНОВА (підпис)

**АНОТАЦІЯ**

Курсовий проєкт включає пояснювальну записку (27 с., 21 рис., список використаних джерел з 4 найменувань).

У цьому курсовому проєкті була створена програма для моделювання мережі передачі даних. Програма може аналізувати час передачі повідомлень, кількість інформаційних та службових пакетів. Також є можливість моделювання процесу визначення маршруту передачі повідомлень в мережі передачі даних з вказаною топологією.

У роботі наведено основні теоретичні відомості про концепцію маршрутизації у мережі передачі даних, а також алгоритми, що визначені для даного варіанту. Проведені тести з різними наборами вхідних даних, і представлено аналіз результатів цих тестів.

Для створення графічного інтерфейсу програми була використана мова програмування Python.

*Ключові слова:*

**ANNOTATION**

ЗМІСТ

1. НАЙМЕНУВАННЯ ТА ГАЛУЗЬ РОЗРОБКИ 2
2. ПІДСТАВА ДЛЯ РОЗРОБКИ 2
3. ЦІЛЬ І ПРИЗНАЧЕННЯ РОБОТИ 2
4. ДЖЕРЕЛА РОЗРОБКИ 2
5. ТЕХНІЧНІ ВИМОГИ 3
   1. Вимоги до програмного продукту, що розробляється 3
   2. Вимоги до апаратного забезпечення 3
   3. Вимоги до програмного та апаратного забезпечення користувача 3
6. ЕТАПИ РОЗРОБКИ 4

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

1

ІАЛЦ.467200.002 ТЗ

Розроб.

Нестерук А.О.

Перевір.

Мартинова О.П. ГЕРАСИМЧУК

*№*

Н. Контр.

Затверд.

*Маршрутизація в мережі передачі даних.*

***Пояснювальна записка***

Літ.

Акрушів

4

КПІ ім. Ігоря Сікорського,

ФПМ КВ-11

1. НАЙМЕНУВАННЯ ТА ГАЛУЗЬ РОЗРОБКИ

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 2 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |

Назва розробки: «Маршрутизація в мережі передачі даних».

Галузь застосування: системне адміністрування мережами, планування побудови мережі передачі даних.

1. ПІДСТАВА ДЛЯ РОЗРОБКИ

Підставою для розробки є завдання на виконання курсового проєкту, затверджене кафедрою системного програмування і спеціалізованих комп’ютерних систем Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського».

1. Мета І ПРИЗНАЧЕННЯ РОБОТИ

Метою даного проекту є моделювання процесу визначення маршруту передачі повідомлень в мережі передачі даних.

1. ДЖЕРЕЛА РОЗРОБКИ

Джерелом інформації є технічна та науково-технічна література, технічна документація, публікації в періодичних виданнях та електронні статті у мережі Інтернет.

1. ТЕХНІЧНІ ВИМОГИ
   1. Вимоги до програмного продукту, що розробляється

* швидке введення вручну компонентів топології мережі мишкою – вузлів і каналів (повний дуплекс та напівдуплекс);
* випадкова генерація структури мережі з заданими політиками створення;
* політики для каналів: випадкового вибору ваги в заданих межах та константне значення із заданої множини значень;
* політики для буферів каналів: випадкового вибору ваги в заданих межах та константне значення з заданої множини значень;
* реалізацію основних звичних для користувача елементів управління: додавання, видалення, виділення, перетягування вузлів і каналів;
* при захопленні об'єкта мишкою відображається основна інформація про об'єкт (наприклад, таблиця маршрутизації, вага каналів, завантаження буферів тощо);
* можливість відключати, включати обрані вузли і канали;
* перегляд покрокового виконання алгоритмів;
* генерування випадкового трафіку повідомлень;
* меню відправлення конкретних повідомлень із однієї в іншу робочу станцію мережі з зазначенням їх розміру.
  1. Вимоги до апаратного забезпечення
* Процесор AMD ryzen 3/Intel Core i3 або вище;
* оперативна пам’ять: 8 Гб;
* наявність доступу до мережі.
  1. Вимоги до програмного та апаратного забезпечення користувача
* операційна система Windows/Linux/macOS;
* Встановлено Python версія 3.12 або вище.

# 

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 3 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |

ЕТАПИ РОЗРОБКИ

|  |  |  |
| --- | --- | --- |
| №  з/п | Назва етапів виконання  дипломного проєкту | Термін виконання  етапів проєкту |
| 1. | Вивчення літератури за тематикою проєкту |  |
| 2. | Розроблення та узгодження технічного завдання |  |
| 3. | Аналіз існуючих рішень |  |
| 4. | Підготовка матеріалів першого розділу курсового  проєкту |  |
| 5. | Підготовка матеріалів другого розділу курсового  проєкту |  |
| 6. | Підготовка дослідницької частини курсового  проєкту |  |
| 7. | Оформлення документації курсового проєкту |  |
| 8. | Попередній огляд матеріалів курсового проєкту |  |
| 9. | Захист курсового проєкту |  |

# 

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 4 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |

ЗМІСТ

ВСТУП 2

1. ТЕОРЕТИЧНІ ВІДОМОСТІ 3
2. ОПИС ЗАДАНОГО АЛГОРИТМУ МАРШРУТИЗАЦІЇ 8
3. ОСНОВНА ЧАСТИНА 10
   1. ОПИС ТА ПРИЗНАЧЕННЯ РОЗРОБЛЕНОЇ ПРОГРАМИ 10
   2. СТРУКТУРА РОЗРОБЛЕНОЇ ПРОГРАМИ 15
   3. ОПИС ІНТЕРФЕЙСУ РОЗРОБЛЕНОЇ ПРОГРАМИ 17
4. ОПИС ПРОЦЕСУ ТЕСТУВАННЯ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ 19
5. АНАЛІЗ ТА ПОРІВНЯННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ 28

ВИСНОВКИ 29

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ 30

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

1

ІАЛЦ.467200.004 ПЗ

Розроб.

Нестерук А.О.

Перевір.

Мартинова О.П. ГЕРАСИМЧУК

*№*

Н. Контр.

Затверд.

*Маршрутизація в мережі передачі даних.*

***Пояснювальна записка***

Літ.

Акрушів

КПІ ім. Ігоря Сікорського,

ФПМ КВ-11

ВСТУП

Комп'ютерна мережа — це комплекс апаратних і програмних компонентів, що забезпечують обмін даними між кінцевими пристроями, такими як комп'ютери, сервери та інші пристрої. Апаратні компоненти включають кінцеві пристрої, що безпосередньо обробляють або зберігають дані, і проміжні пристрої, які сприяють передачі даних між вузлами. Програмні компоненти забезпечують роботу сервісів і процесів, що підтримують функціонування мережі, зокрема маршрутизацію даних, управління трафіком і забезпечення безпеки.

Маршрутизація є однією з ключових функцій мережного рівня, що визначає, як пакети даних передаються між вузлами в мережі. Вона відповідає за побудову та оновлення таблиць маршрутизації на основі обраного алгоритму. Ефективність маршрутизації впливає на продуктивність і надійність мережі, а тому дослідження алгоритмів маршрутизації є важливим етапом проектування мереж.

Існує два основні типи алгоритмів маршрутизації: адаптивні та неадаптивні. Адаптивні алгоритми враховують зміни в топології мережі та навантаження на канали зв’язку, що дозволяє їм гнучко реагувати на зміну умов. Неадаптивні алгоритми використовують статичні маршрути і не змінюють вибір шляхів у відповідь на зміну мережевого трафіку.

У сучасному світі ефективна робота комп'ютерних мереж є критично важливою для багатьох сфер, включаючи науку, бізнес і повсякденне життя. Моделювання мережевих процесів дозволяє оцінити ефективність роботи мережі, визначити можливі слабкі місця і підвищити якість передачі даних ще до впровадження мережі. Це дає змогу оптимізувати використання ресурсів і мінімізувати ризики помилок у реальному середовищі.

Проведення моделювання є обов'язковим етапом при проектуванні сучасних комп'ютерних мереж, адже це дозволяє уникнути неефективних рішень і забезпечити надійність та продуктивність мережі передачі даних.

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 2 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |

1. ТЕОРЕТИЧНІ ВІДОМОСТІ

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 3 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |

Мережевий рівень (Network layer) — це третій рівень мережевої моделі OSI, що відповідає за вибір маршруту для передачі даних між різними пристроями в мережі. Основною задачею мережевого рівня є забезпечення доставки пакетів від джерела до отримувача через проміжні вузли. Цей рівень здійснює трансляцію логічних адрес в фізичні та забезпечує визначення найбільш оптимальних маршрутів для передачі даних.

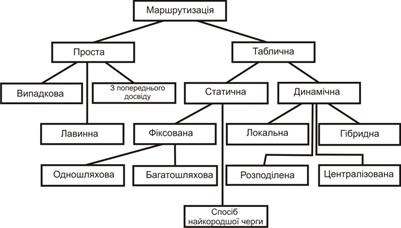
Основні функції мережевого рівня включають маршрутизацію, комутацію, фрагментацію та відновлення пакетів. На цьому рівні працюють маршрутизатори — мережеві пристрої, що збирають інформацію про топологію мережі і на її основі обирають шляхи для передачі пакетів до призначення. Протокол передачі даних на мережевому рівні — це протокол IP (Internet Protocol), який визначає формат даних (дейтаграм) та правила їхньої передачі.

Маршрутизація є ключовою функцією на мережевому рівні, оскільки вона визначає шлях, яким пакет буде передаватися від початкового вузла до кінцевого, через один або кілька проміжних вузлів. Щоб забезпечити ефективну маршрутизацію, мережеві пристрої (наприклад, маршрутизатори) повинні мати таблиці маршрутизації, які містять інформацію про всі можливі шляхи в мережі та використовують алгоритми маршрутизації для обрання оптимального маршруту.

Маршрутизація складається з двох основних етапів: в изначення оптимального маршруту, який включає вибір маршруту на основі різних критеріїв, таких як кількість хопів (проміжних пристроїв), швидкість каналу, надійність мережі, пропускна здатність або затримка та транспортування інформації (комутація), яке полягає у фактичній передачі даних через визначений маршрут з мінімальними втратами та затримками.

Мережа передачі даних зазвичай складається з численних підмереж і хостів, які з'єднані між собою за допомогою високошвидкісних магістралей і маршрутизаторів. В Інтернеті хостом є будь-який пристрій, підключений до мережі, який використовує протокол TCP/IP. Кожен вузол або хост у мережі має унікальну IP-адресу, яка дозволяє ідентифікувати його в рамках глобальної мережі.

Маршрутизатор, який є основним пристроєм на мережевому рівні, отримує пакети з інформацією про призначення і вирішує, куди направити ці пакети далі. Пакет проходить через кілька маршрутизаторів, що визначають шлях до кінцевого вузла. Кожен проміжний вузол або маршрутизатор здійснює вибір наступного кроку, і так продовжується до того моменту, поки пакет не досягне свого кінцевого призначення.



*Рис.1.1 – Види маршрутизації*

Маршрутизація є важливим процесом в комп'ютерних мережах, що визначає шлях, яким пакети даних повинні пройти від джерела до призначення. Це завдання вимагає визначення оптимального маршруту в умовах змінної мережевої інфраструктури, трафіку та вимог до якості обслуговування. Існує кілька підходів до маршрутизації, які можна поділити на дві основні категорії: просту та табличну. Кожна з цих категорій має свої підвиди, що забезпечують різні стратегії вибору маршруту в залежності від вимог мережі та характеристик трафіку.

Проста маршрутизація є однією з найбільш базових форм маршрутизації, при якій маршрути вибираються без урахування складних обчислень чи таблиць маршрутизації. Одним із прикладів є випадкова маршрутизація. У цьому методі пакети даних передаються випадковим чином до одного з доступних сусідів, без визначення конкретного маршруту або оцінки його ефективності.

Це може бути підходом для простих мереж або для ситуацій, коли потрібно мінімізувати складність алгоритмів маршрутизації, але цей метод має значні обмеження щодо ефективості і підходить для великих чи складних мереж.

Іншим методом простої маршрутизації є лавинна маршрутизація. Вона працює за принципом передачі кожного пакета до всіх можливих сусідів. Це створює високе навантаження на мережу, оскільки кожен пакет передається через всі доступні вузли, але з іншого боку підвищується ймовірність доставки пакета до його призначення.

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 4 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |

Лавинна маршрутизація може бути ефективною в ситуаціях, коли потрібно забезпечити максимальну ймовірність доставки, наприклад, у специфічних протоколах або в умовах дуже нестабільної мережі, однак вона малоефективна для великих і навантажених мереж.

До простих методів також відноситься маршрутизація за попереднім досвідом. Цей підхід передбачає, що маршрути вибираються на основі історії передачі пакетів. Якщо певний маршрут був ефективним у минулому, він може бути вибраний для майбутніх передач, що дозволяє зменшити час обробки пакета в реальному часі. Проте, цей метод вимагає ведення певних статистичних даних і може бути неефективним у мережах, де топологія часто змінюється або де є велика кількість непередбачених факторів, що можуть впливати на маршрути.

Таблична маршрутизація є більш складним і ефективним методом, в якому для кожного маршруту зберігається таблиця маршрутизації, що дозволяє більш точно і оперативно визначати оптимальний шлях для передачі даних. Таблична маршрутизація поділяється на два основних підвиди: статичну та динамічну.

Статична маршрутизація характеризується тим, що маршрути в таблиці задаються вручну і не змінюються автоматично. Це підходить для мереж з фіксованою топологією, де маршрути не змінюються, і є певна стабільність у мережевому середовищі. Одним із прикладів статичної маршрутизації є одношляхова маршрутизація, де для кожної пари джерело-призначення обирається один фіксований маршрут. Це дуже простий і передбачуваний метод, який добре працює в невеликих мережах або в умовах, коли мережа стабільна і не зазнає змін. Однак цей метод не є гнучким і не підходить для великих або динамічних мереж, оскільки не враховує змін, що можуть виникнути у топології чи навантаженні мережі.

Багатошляхова маршрутизація в якій, для кожної пари джерело-призначення визначається кілька можливих маршрутів, і пакети можуть передаватися через будь-який з них. Це дозволяє підвищити надійність і забезпечити відмовостійкість мережі, оскільки навіть у випадку відмови одного маршруту можуть використовуватись інші шляхи. Такий підхід може бути ефективним у ситуаціях, коли мережа має кілька незалежних каналів зв'язку або коли важлива безперервна доставка пакетів.

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 5 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |

Динамічна маршрутизація є більш складною і гнучкою, оскільки вона дозволяє мережі автоматично адаптувати маршрути в залежності від змін у топології або стані мережі. Одним із підвидів динамічної маршрутизації є гібридна маршрутизація, яка поєднує елементи статичної і динамічної маршрутизації. У такій системі основний маршрут визначається статично, але може бути змінений в разі потреби, що дозволяє поєднувати надійність і гнучкість.

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 6 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |

Локальна динамічна маршрутизація ґрунтується на тому, що кожен вузол приймає рішення про маршрутизацію на основі інформації лише від своїх безпосередніх сусідів. Цей метод дозволяє швидко адаптуватися до змін у місцевій топології, але не враховує глобальні зміни в мережі. Локальна маршрутизація є дуже швидкою і економною з точки зору ресурсів, однак її ефективність обмежена, оскільки вона не має повної картини стану мережі.

Розподілена маршрутизація забезпечує більш комплексний підхід, оскільки в ній всі вузли обмінюються інформацією про стан мережі і приймають спільне рішення щодо маршрутизації. Це дозволяє створити більш точні та надійні маршрути, оскільки кожен вузол має більш повну інформацію про стан мережі. Однак цей метод вимагає більше ресурсів для збору і обробки даних, що може збільшити затримку і навантаження на мережу.

Централізована маршрутизація передбачає, що рішення про вибір маршруту приймається на одному центральному вузлі, який має повну інформацію про мережу. Цей підхід дозволяє зберегти єдність управління і покращити ефективність маршрутизації, але має велику уразливість у разі відмови центрального вузла.

Вибір методу маршрутизації значною мірою залежить від вимог конкретної мережі, її топології, навантаження та характеристик трафіку. Кожен метод має свої переваги та недоліки, і його ефективність можна значно підвищити завдяки правильному вибору протоколів маршрутизації та використанню метрики для оцінки якості маршрутів.

Протоколи маршрутизації забезпечують основу для обміну інформацією між маршрутизаторами, що дозволяє їм визначати, через які шляхи повинні передаватися пакети для досягнення своєї мети. Існують два основних типи протоколів маршрутизації: протоколи дистанційного вектору і протоколи стану зв'язку.

Протоколи дистанційного вектору, такі як RIP (Routing Information Protocol), визначають маршрути через обмін таблицями маршрутизації між сусідніми маршрутизаторами. Кожен маршрутизатор в такій мережі повідомляє своїх сусідів про відстань до всіх можливих призначень, і на основі цих даних маршрутизатор вибирає найкращий шлях. Один з важливих аспектів цих протоколів — це використання метрики для оцінки "вартості" шляху, наприклад, кількість хопів (перепризначених маршрутизаторів) або час затримки.

Протоколи стану зв'язку, наприклад OSPF (Open Shortest Path First), використовують іншу методику. У таких протоколах кожен маршрутизатор має повну картину топології мережі і може самостійно обчислювати найкоротші шляхи за допомогою алгоритмів, таких як алгоритм Дейкстри. Це дозволяє обчислювати найбільш ефективні маршрути без необхідності обміну таблицями маршрутів, що може знижувати навантаження на мережу.

Метрика маршрутизації – це характеристика, роль якої полягає у визначенні оптимального маршруту передачі даних. У різних протоколах маршрутизації метрика використовується по-різному, від простого підрахунку кількості проміжних вузлів у RIP до складніших алгоритмів у OSPF та EIGRP, які враховують затримку, пропускну здатність і надійність мережі. Основна мета метрики — забезпечити баланс між ефективністю маршруту і стабільністю мережевого трафіку, що є ключовим фактором для стабільної роботи будь-якої мережі.

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 7 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |

2. ОПИС ЗАДАНОГО АЛГОРИТМУ МАРШРУТИЗАЦІЇ

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 8 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |

Метод каталогів, орієнтованих на сеанс, є специфічним підходом до маршрутизації в пакетних мережах, де маршрути вибираються і визначаються на основі попередньо встановлених шляхів і зв'язків, які існують між вузлами мережі протягом певного сеансу комунікації. Цей алгоритм орієнтований на розподілену маршрутизацію, де інформація про маршрути зберігається не в кожному окремому маршрутизаторі, а в центральних каталогах, які використовуються для визначення шляху для передачі пакетів.

У цьому випадку маршрутизація здійснюється через каталог, який є своєрідним набором даних про мережеву топологію. У ньому зберігаються всі активні маршрути та зв'язки між вузлами мережі. Кожен каталог містить записи, які вказують на те, через які маршрутизатори проходять пакети для конкретних сеансів зв'язку. Це дозволяє здійснювати більш ефективну маршрутизацію, зменшуючи навантаження на кожен окремий маршрутизатор, оскільки вони не повинні постійно приймати рішення щодо маршруту.

Особливість методу, полягає в тому, що маршрути не змінюються під час сеансу. Як тільки шлях був вибраний і встановлений в каталозі на початку сеансу, він залишається сталим для всього цього сеансу передачі даних, що дозволяє мінімізувати затримки при передачі пакетів. Цей підхід також забезпечує більш ефективне використання мережевих ресурсів, оскільки маршрути вже відомі і не потребують частого перерахунку.

Tymnet — це одна з перших пакетних мереж загального призначення, яка була розроблена для забезпечення обміну даними між різними комп'ютерними системами через широкомасштабні мережі. Її архітектура передбачала використання методів маршрутизації, які дозволяли передавати пакети інформації між віддаленими користувачами і серверами без необхідності встановлення прямого з'єднання між ними. Tymnet є прикладом мережі, в якій передача даних здійснюється за допомогою пакетів, і кожен пакет може проходити різними маршрутами через мережу в залежності від топології і стану мережі в момент відправки.

Один з важливих аспектів Tymnet полягає в тому, що вона підтримує концепцію орієнтованих на сеанс з'єднань. Коли користувач ініціює з'єднання з іншим користувачем або сервером, створюється сеанс, який має певний маршрут через мережу для передачі даних.

Цей сеанс є основою маршрутизації в системі, і інформація про маршрут зберігається в центральному каталозі. Каталог містить всю інформацію про шляхи передачі для поточних сеансів і визначає, як пакети повинні рухатись через мережу для досягнення кінцевої точки призначення.

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 9 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |

Важливою особливістю є здатність адаптуватися до змін у мережевій топології. Якщо виникають зміни, наприклад, через відмову каналу або зниження пропускної здатності, ці зміни можуть бути відображені в нових записах каталогів, що дозволяє відновити або змінити маршрути для нових сеансів. Проте для вже існуючих сеансів маршрути залишаються незмінними протягом усього сеансу, що гарантує стабільність і знижує ймовірність помилок в передачі даних.

Метод каталогів, орієнтованих на сеанс, є важливим компонентом маршрутизації в мережах типу Tymnet, дозволяючи ефективно передавати пакети через складні мережеві топології. Він забезпечує стабільність і низьку затримку під час передачі даних завдяки фіксації маршруту на початку сеансу. Пакетна мережа Tymnet, в свою чергу, дозволяє здійснювати комунікацію між віддаленими користувачами і серверами, використовуючи централізовані каталоги для зберігання і оновлення інформації про маршрути. Цей підхід забезпечує високу ефективність і надійність мережевих з'єднань при мінімальних витратах на ресурси.

3. ОСНОВНА ЧАСТИНА

3.1. ОПИС ТА ПРИЗНАЧЕННЯ РОЗРОБЛЕНОЇ ПРОГРАМИ

В данному курсовому проєкті було розроблено програму для моделювання комп'ютерної мережі та симуляції передачі повідомлень в мережі незалежно від її топології.

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 10 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |

В програмі в якості алгоритму маршрутизації використовується метод каталогів, орієнтвоаних на сеанс.

Програма написана на мові програмування Python, з використанням графічної бібліотеки tkinter. Створений графічний інтерфейс дозволяє користувачу самостійно задати структуру мережі, кількість вузлів та робочих станцій, визначити тип та вагу каналів, задавати розмір інформаційної та службової частин повідомлення, що буде передаватися.

Результати роботи програми надають користувачеві інформацію про:

* Обсяг службової інформації, яка передалася;
* Кількість пакетів, на які було розбите початкове повідомлення;
* Час, витрачений на передачу від одного вузла до іншого;
* Маршрут, яким передавалося повідомлення.

Функціонал програми дозволяє:

* Створення компонентів топології завдяки графічному інтерфейсу, та розташування їх на робочому вікні;
* Зручно налаштовувати мережу, завдяки функціям додавання, видалення, виділення та перетягування елементів топології;
* Активувати або деактивувати елементи мережі для аналізу їх впливу на мережу;
* Визначити характеристики мережі, задаючи тип каналів зв’язку та їх вагу;
* Задавати розмір повідомлення, розмір пакетів, а також вибирати метод передачі даних (датаграма або віртуальний канал);
* **Зберігати** отримані результати у вигляді .xlsx файлів, включаючи обсяг трафіку, час передачі, кількість пакетів і маршрути, що дозволяє детально аналізувати роботу мережі.

3.2. СТРУКТУРА РОЗРОБЛЕНОЇ ПРОГРАМИ

Програмний код розділений на декілька модулів в залежності від функціоналу який він виконує.

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

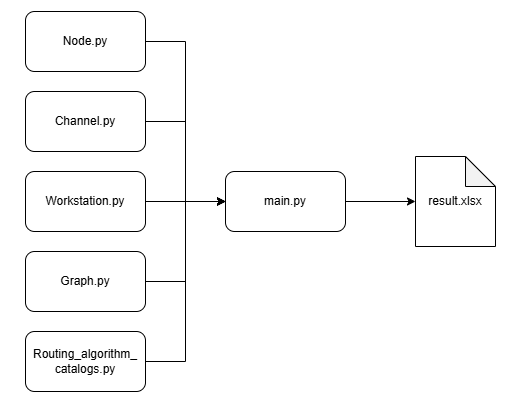
|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 11 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |



*Рис.3.1 – Схема розробленої програми*

Файл main.py є головним модулем програми, який виконує координацію роботи всіх інших модулів і забезпечує основну логіку для досягнення мети програми. Результати роботи програми, зберігаються у файл result.xlsx.

Node.py - відповідає за створення та представлення вузла в мережі. В цьому модулі виконуються такі функції як: створення, видалення, активація/дезактивація вузла.

Channel.py - відповідає за створення та представлення каналу в мережі. В цьому модулі виконуються такі функції як: створення, видалення, активація/дезактивація та задання ваги каналу.

Workstation.py - відповідає за створення та представлення робочої станції в мережі. В цьому модулі виконуються такі функції як: створення, видалення, активація/дезактивація робочої станції.

Graph.py - відповідає за відображення мережі у вигляді графа для зручної роботи з нею.

Routing\_algorith\_catalogs.py – відповідає за реалізацію алгоритму маршрутизації відповідно до варіанту.

3.3. ОПИС ІНТЕРФЕЙСУ РОЗРОБЛЕНОЇ ПРОГРАМИ

Програмний інтерфейс складається з двох частин, а саме з області для побудови мережі, її керуванням, та область з інструментами.

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

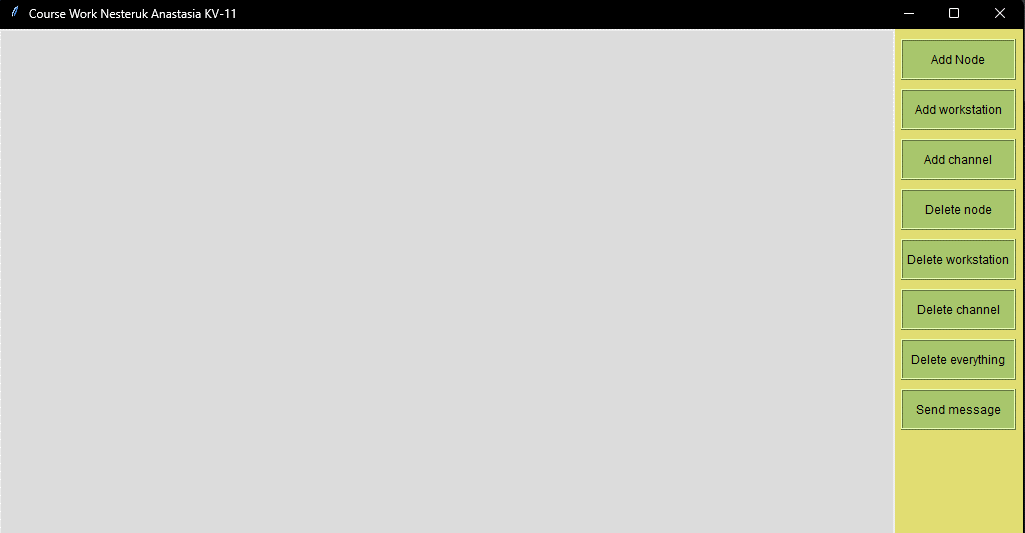
|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 12 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |



*Рис.3.2 – Головне вікно програми*

Як можна побачити на Рис.3.2 область з інструментами має 8 кнопок:

Add Node – додає новий комунікаційний вузол на екран.

Add Workstation – додає нову робочу станцію на екран.

Add Channel – додає з’єднання (канал) між вузлами або робочими станціями.

Delete Node – видаляє обраний вузол з екрану.

Delete Workstation – видаляє обрану робочу станцію з екрану.

Delete Channel – видаляє з’єднання (канал) між вузлами або робочими станціями.

Delete Channel – повністю очищує робочу область від усіх елементів.

Send Message – надсилання повідомлення з обраного вузла.

Область для побудови мережі відображає створені користувачем елементи мережі а також дозволяє зручно ними керувати, розташовувати у потрібному місці та інше.

Розглянемо детальніше як створюється мережа. Для того щоб створити мережі користувач має натиснути кнопку «Add Node» або «Add Workstation», після чого на екрані з’явиться вузол/робоча станція яку можна перетягнути в бажане місце. В даній програмі вузол представлений у вигляді кола, а робоча станція у вигляді квадрата. За замовчування вузол та роб. Станція мають жовто-зелений колір, якщо користувач хоче виділити об’єкт то потрібно натиснути лівою кнопкою миші по ньому, після чого об’єкт змінить колір на зелений. Відключити елемент можна натиснувши правою кнопкою миші по ньому, після чого він стане червоним.

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

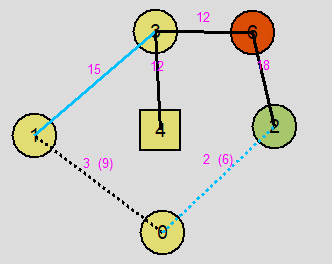
|  |
| --- |
| Арк. |

|  |
| --- |
| 13 |

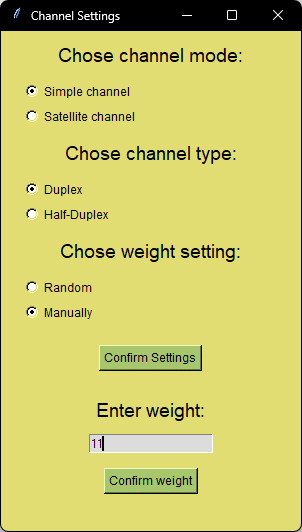
|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |

Для того щоб з'єднати два вузли, потрібно їх виділити та натиснути кнопку

«Add Channel» та вибрати тип каналу - звичайний або супутниковий, дуплекс або напівдуплекс та задати вагу каналу. Дуплексний канал має чорний колір, напівдуплексний – синій, звичайний канал – суцільною лінією, а супутниковий – пукнтиром.



*Рис.3.3 – Приклад створення мережі*



|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

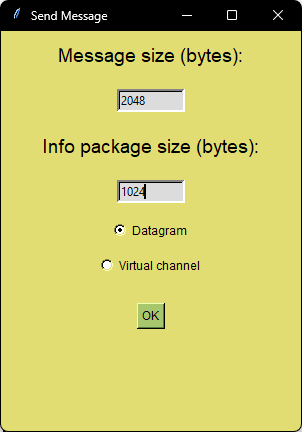
|  |
| --- |
| Арк. |

|  |
| --- |
| 14 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |

*Рис.3.4 – Меню для налаштування каналу*

Для відправення повідомлення потрібно обрати вузол з якого буде надіслано повідомлення та натиснути кнопку «Send message». У вікні яке відкрилося потрібно вказати розмір повідомлення та тип з’єднання.



*Рис.3.5 – Меню для налаштування повідомлення*

Після відправлення повідомлення можна буде побачити таблицю маршруту яке пройшло повідомлення, а всі дані отримані під час передачі будуть збережні в .xlsx файл.

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

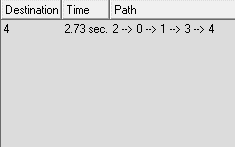
|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 15 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |



*Рис.3.6 – Маршрут, що пройшов пакет*

4. ОПИС ПРОЦЕСУ ТЕСТУВАННЯ ПЕРЕДАЧІ ПОВІДОМЛЕНЬ

Для тестування побудуємо мережу згідно варіанту: Мінімум 25 комунікаційних вузлів, 1 канал – супутниковий; Середній ступінь мережі – 3,5; ваги каналів - 1, 2, 3, 5, 7, 11, 12, 15, 18 , 32.

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

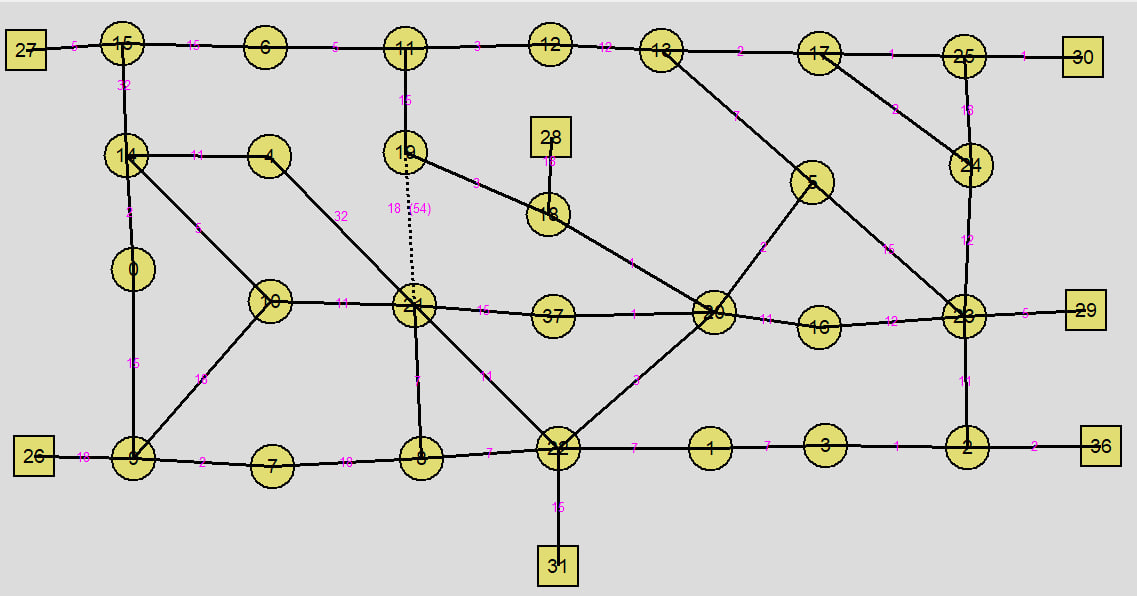
|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 16 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |



*Рис.4.1 – Мережа заданої топології (дуплексний канал зв'язку)*

Проведемо тестування використовуючи спочатку дуплексний канал зв’язку, а потім напівдуплексний.

Протестуємо мережу зі сталим розміром інформаційного пакету - 1024 байти, і різними розмірами самого повідомлення.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Тип | Розмір повідомлення | Розмір інформаційного пакету | Кількість інформаційних пакетів | Кількість службових пакетів | Розмір інформаційного пакету (байт) | Розмір службового трафіку (байт) | Загальний трафік(байт) | Час |
| Datagram | 2048 | 1024 | 105 | 105 | 71680 | 5880 | 77560 | 8,33 |
| Datagram | 4096 | 1024 | 175 | 175 | 143360 | 9800 | 153160 | 16,18 |
| Datagram | 8192 | 1024 | 315 | 315 | 286720 | 17640 | 304360 | 32,45 |
| Datagram | 16384 | 1024 | 595 | 595 | 573440 | 33320 | 606760 | 64,59 |
| Virtual channel | 2048 | 1024 | 105 | 385 | 71680 | 12250 | 83930 | 11,63 |
| Virtual channel | 4096 | 1024 | 175 | 455 | 143360 | 12950 | 156310 | 22,41 |
| Virtual channel | 8192 | 1024 | 315 | 595 | 286720 | 14350 | 301070 | 43,93 |

*Таблиця 1 – Дані отримані при тестуванні мережі з дуплексним типом каналу, різним розміром повідомлення та сталим розміром інформаційного пакету*

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 13 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |

|  |
| --- |
| Зм. |

|  |
| --- |
| Арк. |

|  |
| --- |
| № докум. |

|  |
| --- |
| Підпис |

|  |
| --- |
| Дата |

|  |
| --- |
| Арк. |

|  |
| --- |
| 13 |

|  |
| --- |
| ІАЛЦ.467200.002 ТЗ |