

Звіт

**Комп'ютерний проект з дискретної математики
Тема №6: Бібліотека для роботи з відношеннями**

**Роботу виконали студенти 1 курсу
факультету Прикладних Наук:**

Аліна Вороніна
Богдан Михайлів
Гліб Солоджук
Марта Нагорнюк
Оксана Садова

Зміст:

- 1) Загальна інформація
- 2) Основні модулі та функції
- 3) Алгоритми
- 4) Принципи дискретної математики
- 5) Процес виконання проекту
- 6) Використання проекту
- 7) Результати роботи
- 8) Розподіл роботи

Загальна інформація

Бібліотека для роботи з відношеннями створена для аналізу відношень, заданих nхn матрицею. Усі обчислення виконуються на основі принципів дискретної математики.

Бібліотека складається з декількох модулів, які реалізують наступні можливості:

- 1) Читання відношення з файлу та запис відношення у файл формату csv.
- 2) Пошук симетричного замикання відношення.
- 3) Пошук рефлексивного замикання відношення.
- 4) Пошук транзитивного замикання відношення, на основі алгоритму Уоршелла.
- 5) Перевірка відношення на транзитивність.
- 6) Розбиття відношення еквівалентності на класи еквівалентності.
- 7) Підрахунок кількості усіх різних транзитивних відношень на множині з n елементів.

Основні модулі та функції

Імпортовано модуль Numpy для роботи з матрицями; модуль Itertools для альтернативної роботи по типу for-loop циклів; модуль Doctest для проходження функціями докстести.

Модуль transitive.py містить функції для генерації, доповнення, перевірки, обрахунку транзитивних відношень: функція build_transition_func будує транзитивне замикання введеного відношення й повертає його у вигляді масивів у масиві;

- функція transition_check перевіряє чи введене відношення транзитивне й повертає True або False;
- функція generating_relations генерує транзитивне відношення введеного розміру;
- функція count_transitive_relations повертає кількість можливих транзитивних замикань для матриці введеного розміру;

Модуль symmetry.py містить функції для перевірки та побудови симетричних відношень:

- функція symmetry_check_func перевіряє чи введене відношення симетричне та повертає True або False;

- функція `build_symmetry_func` перетворює введені відношення у симетричне й повертає його у вигляді масивів у масиві

Модуль `reflection.py` містить функції для перевірки та побудови рефлексивних відношень:

- функція `reflection_check_func` перевіряє чи введені відношення рефлексивні та повертає `True` або `False`
- функція `build_reflection_func` перетворює введені відношення у рефлексивні й повертає його у вигляді масивів у масиві

Модуль `read_relation.py` містить функції для зчитування відношень:

- функція `read_relat` зчитує відношення з введенного файлу й повертає його у вигляді масиву масивів

Модуль `equality_relation.py` містить функції для знаходження класів еквівалентності відношень:

- функція `equality_classes` повертає масив масивів, кожен з яких - це клас еквівалентності до введенного відношення.

Алгоритми

Використано алгоритми перевірки відношення на рефлексивність, симетричність, транзитивність і побудова рефлексивного, симетричного та транзитивного замикань. Також використано алгоритм розподілу відношення на класи еквівалентності.

Принципи дискретної математики

Обчислення виконуються на основі принципів дискретної математики. При створенні бібліотеки було використано означення рефлексивного, симетричного й транзитивного відношень та класів еквівалентності, а також їх властивості при відображенні на матриці.

При роботі з транзитивними відношеннями було використано алгоритм Уоршелла, який дозволяє перевірити відношення на транзитивність та знайти його транзитивне замикання. Даний алгоритм можна зустріти у модулі `transitive.py`.

Здійснення перевірки на рефлексивність відбувається перевіркою головної діагоналі матриці: якщо всі значення $= 1$, то відношення рефлексивне, якщо є хоча б одне значення, яке $= 0$, таке відношення не є рефлексивним.

Здійснення перевірки на симетричність відбувається перевіркою симетричності матриці відносно головної діагоналі, при цьому значення на власне головній діагоналі матриці не беруться до уваги.

Розбиття відношення еквівалентності на класи еквівалентності відбувається безпосередньо за допомогою комп'ютерного алгоритму, адже у модуль передається відношення, яке вже є еквівалентним і не потребує перевірки на рефлексивність, симетричність та транзитивність.

Процес виконання проекту

Упродовж останнього місяця наша команда провела три зустрічі для роботи над комп'ютерним проектом з дискретної математики:

- Під час першої ми обрали тему проекту, розподілили між собою завдання та визначили дедлайни для їх виконання.
- Друге зібрання було проміжним. На ньому ми ділились інформацією про стан своїх завдань, обговорювали проблеми, що виникли, та планували подальшу роботу над проектом
- Третя (фінальна) зустріч складалась із підбиття підсумків та планування захисту проекту. Також кожен мав змогу пояснити іншим принципи виконання своєї частини роботи та поділитись враженнями про проект.

У зв'язку з пандемічною ситуацією та великою відстанню між місцями знаходження учасників команди, усі зустрічі проводилися дистанційно. Комунікація творців проекту здійснювалась за допомогою онлайн сервісів Telegram та Google Meet. Також для збереження усіх необхідних матеріалів використовувалась онлайн-платформа GitHub

Використання проекту

Бібліотека для роботи з відношеннями є надзвичайно корисною та потрібною. Можливість її використання розширює комп'ютерну сферу праці над відношеннями і дозволяє шукати їх транзитивні, рефлексивні та симетричні замикання, розбивати на класи еквівалентності та здійснювати перевірку на транзитивність майже миттєво.

Бібліотека може використовуватись для перевірки виконання завдань з дискретної математики на правильність, швидкого аналізу відношень, а також буде корисною при створенні інших проектів із дискретної математики.

Результати роботи

Робота над проектом дала змогу вдосконалити знання з дискретної математики, а саме знання відношень та їхніх властивостей, таких як рефлексивність, симетричність та транзитивність та розбиття на класи еквівалентності.

Працюючи в команді, ми змогли заглибитися в тему відношень та створити бібліотеку для роботи з ними, що стане корисною для майбутніх проектів. Ми також дізналися багато нової інформації, яка допомогла краще засвоїти матеріал та поглибити свої знання з дискретної математики. Ми усі отримали позитивні враження від роботи над проектом, адже це був новий досвід для кожного.

Викладачі надали повний список тем, кожна з яких була неймовірно цікавою. Викладачі були відкриті для запитань та обговорень щодо проекту у позакласний час. Асистенти допомогли з тестуванням бібліотеки, а саме надали доктести для перевірки працездатності модулів.

Результатом роботи команди є бібліотека для роботи з відношеннями, яка може стати допоміжним ресурсом для інших програм у майбутньому.

Репозиторій проекту на GitHub:

https://github.com/Team-8-Discrete-computer-project/project_topic_6

Над проектом працювали:



Аліна Вороніна

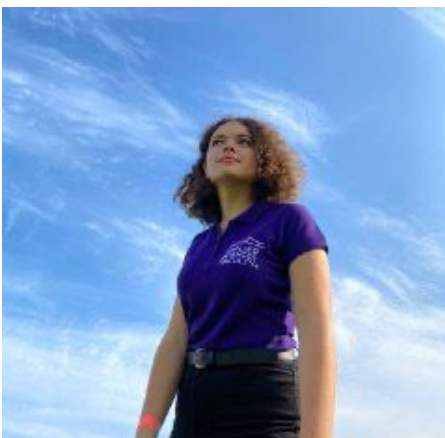
- написання модуля для розбиття відношення еквівалентності на класи еквівалентності
- написання звіту
- тестування модулів

Гліб Солоджук

- написання модуля для роботи з транзитивними відношеннями
 - написання звіту
 - тестування модулів



Марта Нагорнюк



- написання модуля для запису відношення у файл
- написання звіту
- тестування модулів

Богдан Михайлів

- написання модуля для читання відношення з файлу
- написання модуля для роботи з симетричними відношеннями
 - написання звіту
 - тестування модулів



Оксана Садова

- написання модуля для роботи з рефлексивними відношеннями
- написання звіту
- тестування модулів