НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. ІГОРЯ СІКОРСЬКОГО”

КАФЕДРА АВТОМАТИЗОВАНИХ СИСТЕМ ОБРОБКИ ІНФОРМАЦІЇ І УПРАВЛІННЯ

Комп’ютерний практикум № 3

з дисципліни

“Теорія прийняття рішень”

Варіант 5

Виконала:

студентка групи ІС-71

Вознюк О.В.

Перевірила:

доцент

Жураковська О.С.

Київ-2020

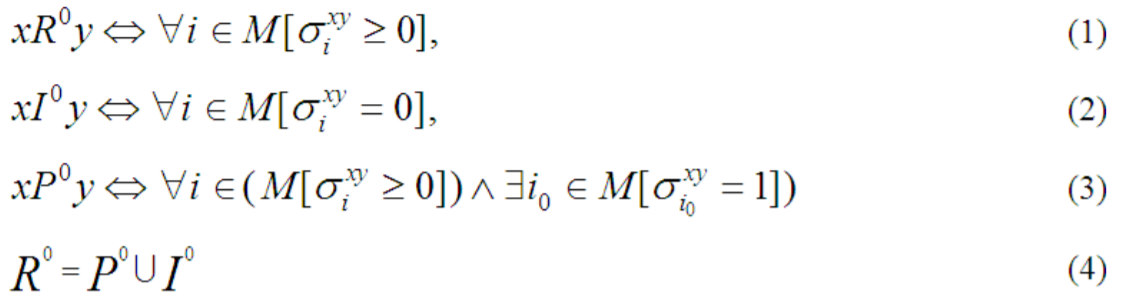
**Постановка задачі**

**Опис етапів вирішення задачі, короткі теоретичні відомості**

Порівнюваність критеріїв – можливість деякого співставлення за перевагою альтернатив, що відрізняються оцінками за даними критеріями.

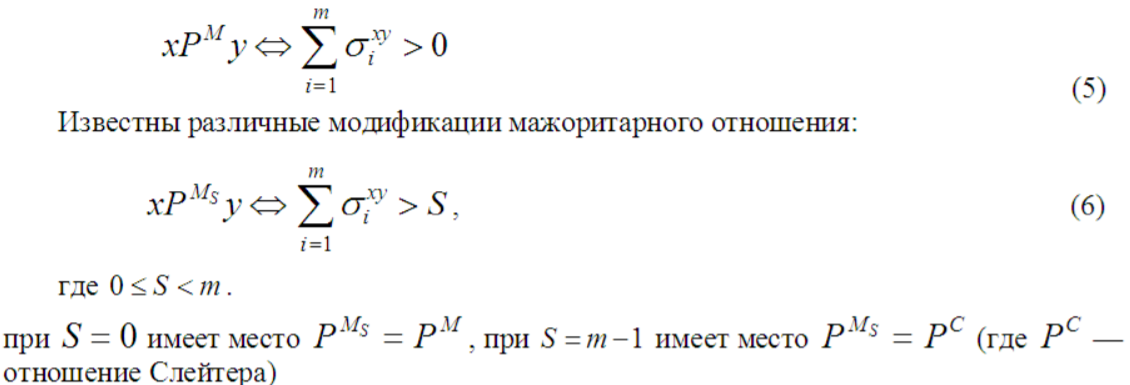
Координатне відношення – для порівняння за яким двох альтернатив не потрібна інформація про ступінь близькості оцінок цих альтернатив за відповідними координатами. Потрібна тільки інформація про порядок оцінок на відповідних координатних вісях (тобто про знаки різниць одноіменних координат)

Відношення Парето

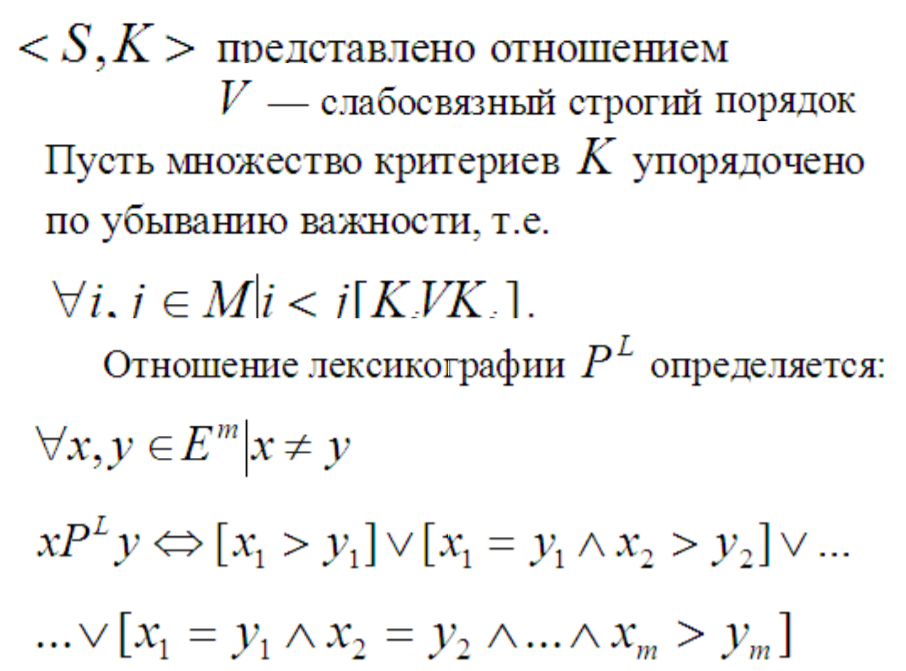


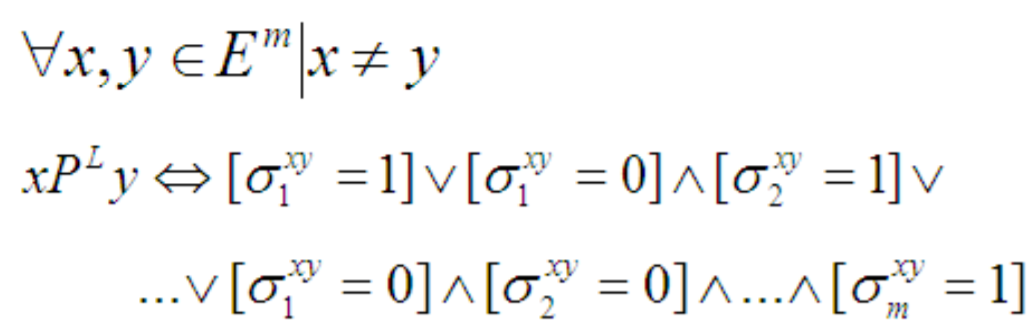
Мажоритарне відношення

Всі критерії рівноважливі, тобто на безлічі критеріїв K задане зв’язне відношення еквівалентності E.

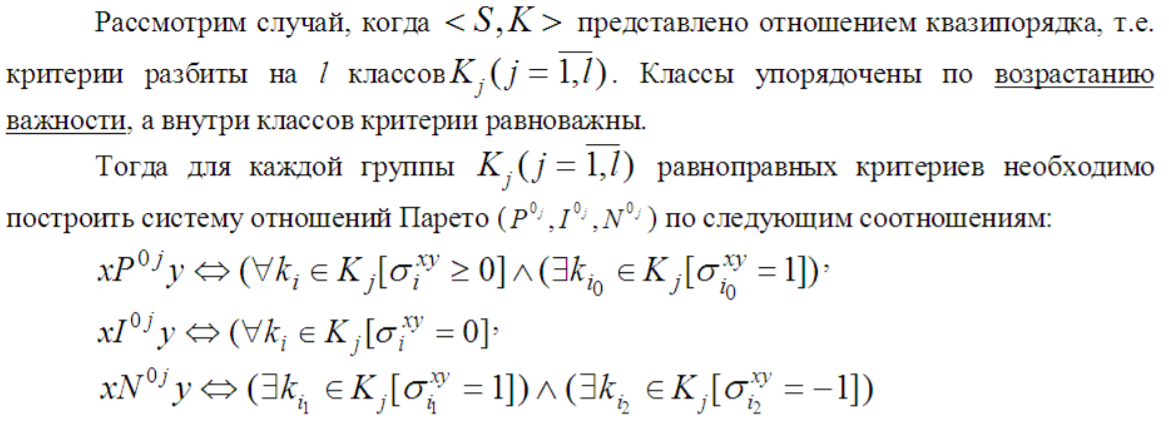


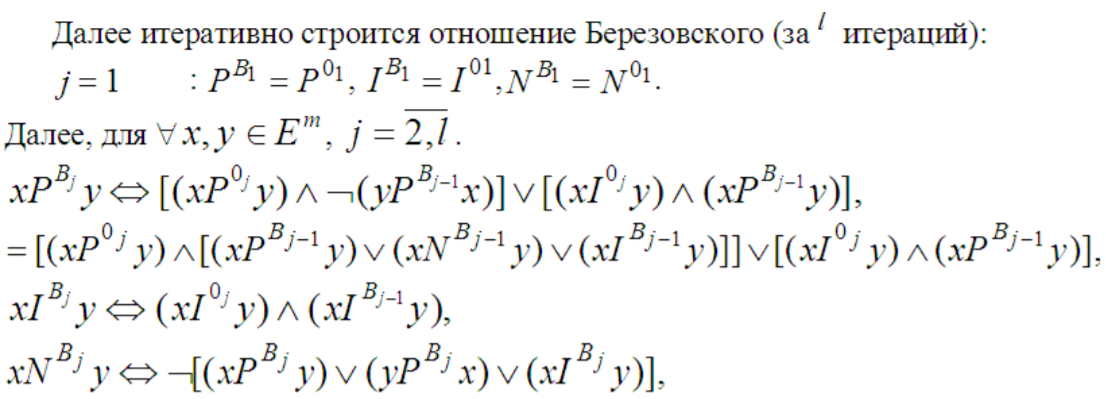
Відношення лексикографії

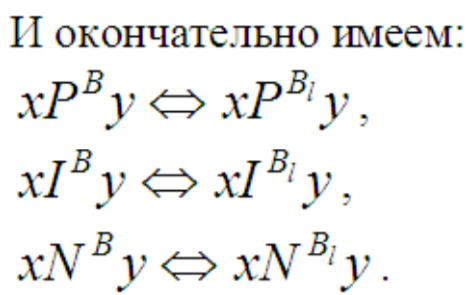




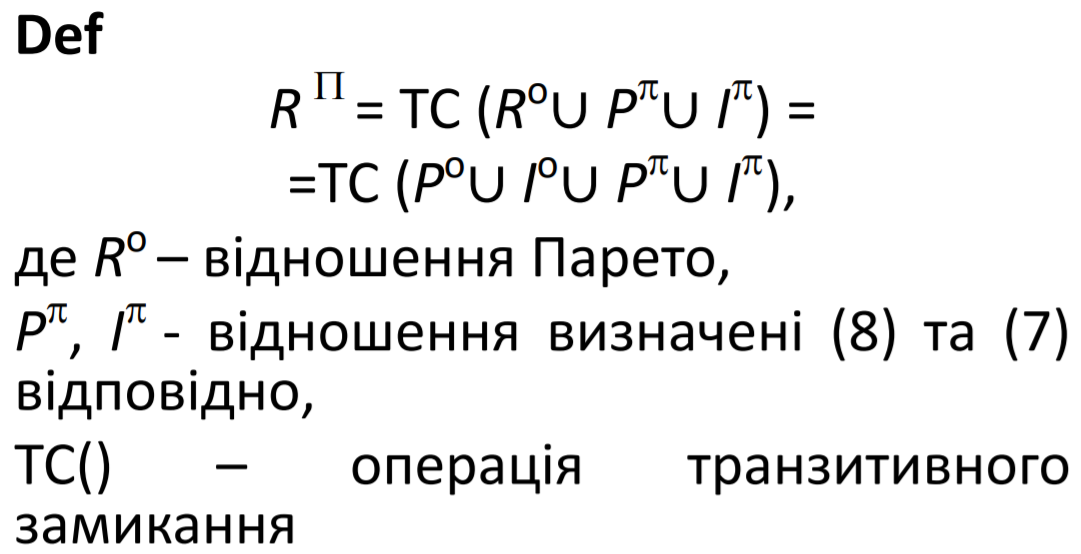
Відношення Березовського

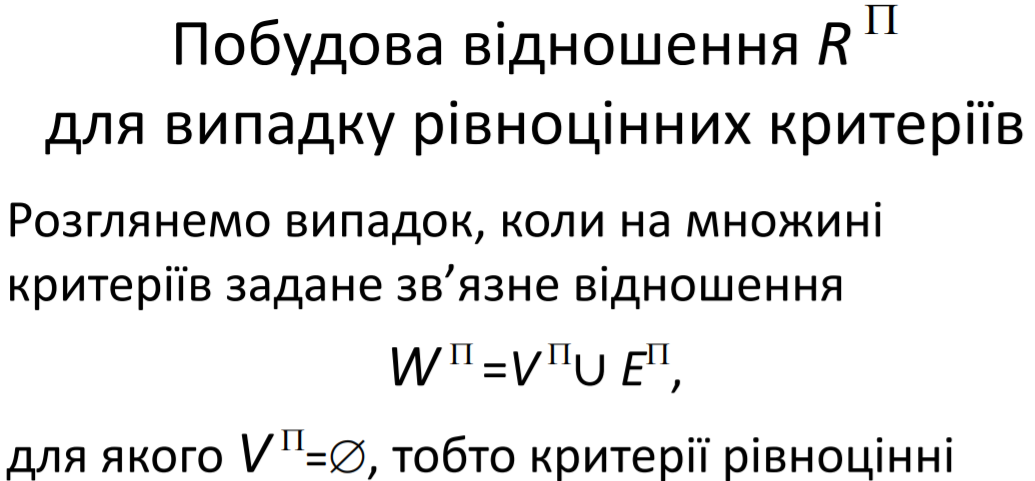






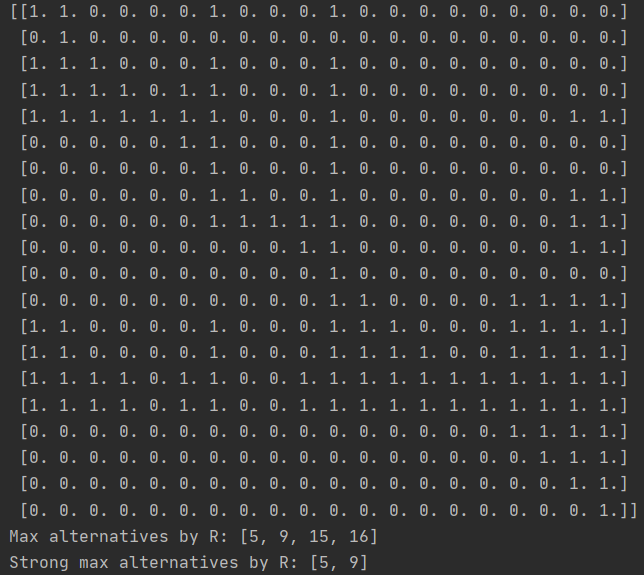
Відношення Подиновського





**Опис вирішення задачі з поясненнями та обгрунтуванням усіх проміжних (якщо метод містить більше одного етапу) та кінцевих результатів**Відношення Парето

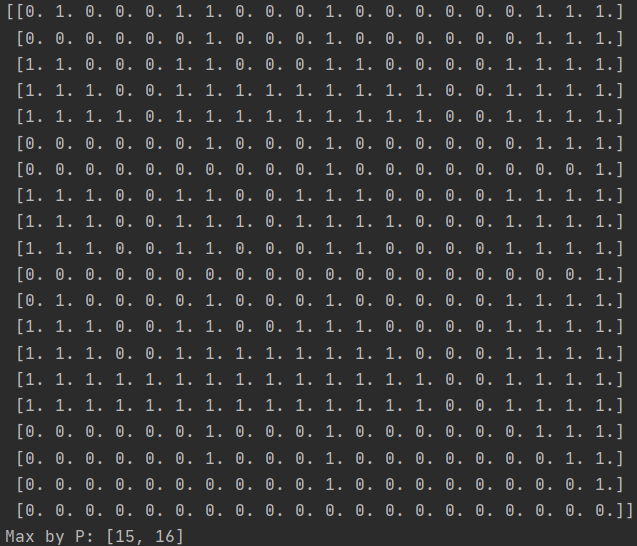
* Для кожної пари альтернатив визначити чи має відповідний вектор сігма -1
* На основі векторів сігма зробити висновок для кожної пари альтернатив чи належать вони відношенню Парето (так, якщо відсутні -1 у векторі сігма)
* Заповнити матрицю відношення Парето



Матриця відношення Парето а також максимальні та строго максимальні по R елементи

Мажоритарне відношення

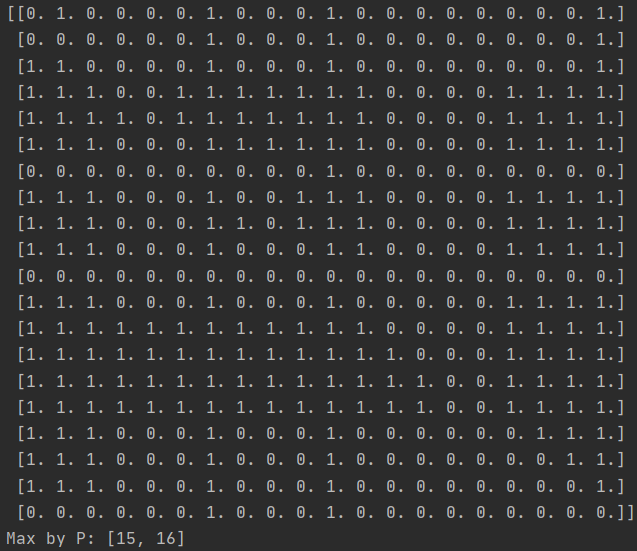
* Для кожної пари альтернатив визначити чи сума елементів відповідного вектора сігма >0
* На основі векторів сігма зробити висновок для кожної пари альтернатив чи належать вони мажоритарному відношенню (так, якщо сума елементів вектора сігма >0)
* Заповнити матрицю мажоритарного відношення



Матриця мажоритарного відношення та максимальні по P елементи

Відношення лексикографії

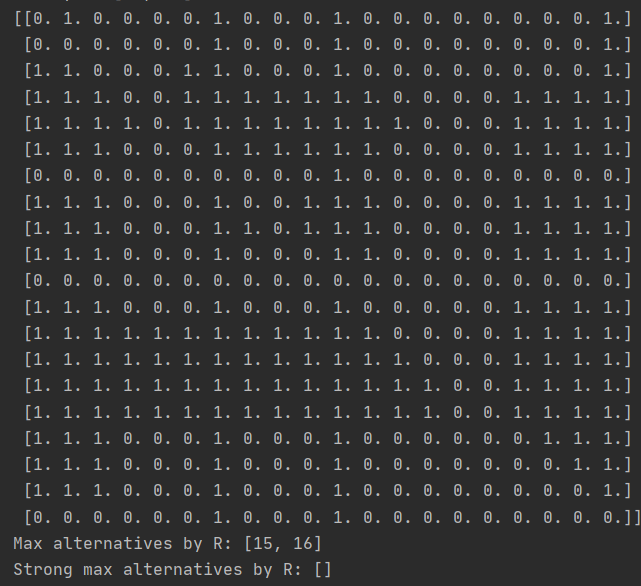
* Знайти вектори сігма для кожної пари альтернатив.
* Відсортувати елементи у кожному векторі сігма за даним порядком важливості критеріїв
* Визначити чи належить пара алтернатив відношенню лексикографії (належить, якщо елемент у векторі =1 і перед цим не було -1)
* Заповнити матрицю відношення лексикографії



Матриця відношення лексикографії та максимальні по P елементи

Відношення Березовського

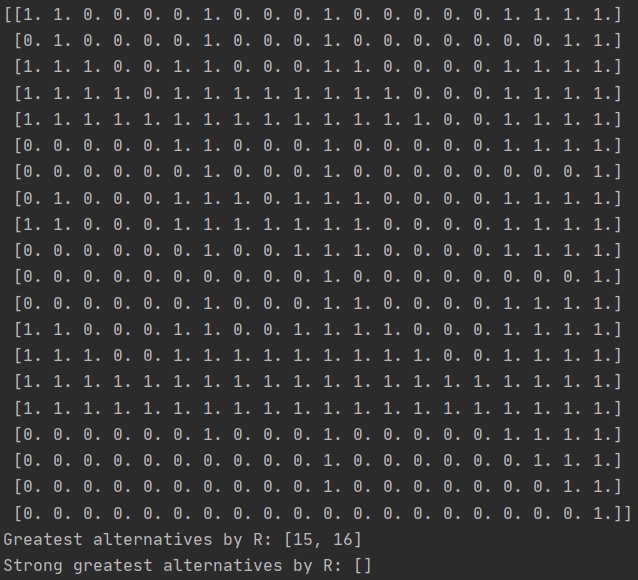
* Шукаємо відношення Парето для кожного класу критеріїв (окремо симетричну частину, асиметричну та непорівнюваності (для першого класу))
* Попарно для симетричних частин, асиметричних та непорівнюваності классів знайти перетин та таким чином визначити чи належать пари альтернатив відношенню Березовського



Матриця відношення Березовського та найбільші та строго найбільші по R елементи

Відношення Подиновського

* На основі початкової матриці знайти матрицю псі
* Знайти за матрицею псі відношення Парето



Матриця відношення Подиновського та найбільші та строго найбільші елементи по R

**Лістинг програми з необхідними коментарями**

https://github.com/oleksandravozniuk/DecisionMakingTheory/tree/main/tpr2

**Короткий опис використаних в програмі бібліотек та методів**

Numpy — розширення мови Python, що додає підтримку великих багатовимірних масивів і матриць, разом з великою бібліотекою високорівневих математичних функцій для операцій з цими масивами.

Використані методи та об’єкти бібліотеки Numpy:

1. Масив NumPy - це не те ж саме, що і клас array.array з Стандартної бібліотеки Python, який працює тільки з одновимірними масивами.
2. numpy.zeroes ()

numpy.zeros ((rows, columns), dtype)

Ця функція створить масив numpy із заданою кількістю вимірів, де кожен елемент буде дорівнювати 0. Якщо dtype не вказано, за замовчуванням буде використовуватися dtype.

1. numpy.loadtxt(fname, dtype=<class 'float'>, comments='#', delimiter=None, converters=None, skiprows=0, usecols=None, unpack=False, ndmin=0, encoding='bytes', max\_rows=None)

**Опис класів. Перелік розроблених функцій та методів**

Файл task1.py

|  |  |  |  |
| --- | --- | --- | --- |
| Ф-ція/Метод | Параметри | Опис | Значення,  що повертає |
| pareto | sigmas | Знаходить відношення Парето | Матрицю відношення Парето |

Файл task2.py

|  |  |  |  |
| --- | --- | --- | --- |
| Ф-ція/Метод | Параметри | Опис | Значення,  що повертає |
| majoritar | sigmas | Знаходить мажоритарне відношення | Матрицю мажоритарного відношення |

Файл task3.py

|  |  |  |  |
| --- | --- | --- | --- |
| Ф-ція/Метод | Параметри | Опис | Значення,  що повертає |
| lexicographic | \_sigmas | Знаходить лексикографічне відношення | Матрицю лексикографічного відношення |
| get\_sorted\_sigma\_by\_criteria | \_sigmas, criteria\_order | Сортує елементи векторів сігма у порядку важливості критерії | Нові вектори сігма |
| set\_right\_criteria\_order | arr, criteria\_order | Сортує елементи одного вектора сігма у порядку важливості критерії | Новий вектор сігма |

Файл task4.py

|  |  |  |  |
| --- | --- | --- | --- |
| Ф-ція/Метод | Параметри | Опис | Значення,  що повертає |
| berezovskiy | \_sigmas | Знаходить відношення Березовського | Матрицю відношення Березовського |
| div\_vectors\_into\_classes | sigma\_vectors | Ділить критерії на класи | Три матриці для кожного класу |
| set\_classes | а (сігма вектор) | Визначає у сігма векторі який елемент належить якому класу | Три вектора для кожного класу |

Файл task5.py

|  |  |  |  |
| --- | --- | --- | --- |
| Ф-ція/Метод | Параметри | Опис | Значення,  що повертає |
| podinovskiy | input\_matrix | Знаходить відношення Подиновського | Матрицю відношення Подиновського |
| make\_psi\_matrix | input\_matrix | Створює матрицю псі | Матрицю псі |

Файл optimization.py

|  |  |  |  |
| --- | --- | --- | --- |
| Ф-ція/Метод | Параметри | Опис | Значення,  що повертає |
| domination\_P\_max | \_matrix | Знаходить найбільші альтернативи по Р (домінування) | Вектор найбільших альтернатив по P |
| domination\_R\_max | \_matrix | Знаходить найбільші альтернативи по R (домінування) | Вектор найбільших альтернатив та строго найбільших по R |
| blocking\_P\_max | \_matrix | Знаходить найбільші альтернативи по Р (блокування) | Вектор найбільших альтернатив по P |
| blocking\_R\_max | \_matrix | Знаходить найбільші альтернативи по R (блокування) | Вектор найбільших альтернатив по R та строго найбільших |
| check\_symmetrical\_part | \_matrix | Перевіряє наявність симетричної частини | True or False |
| part\_I | \_matrix | Виділяє симетричну частину | Матрицю симетричної частини |
| part\_P | \_matrix | Виділяє асиметричну частину | Матрицю асиметричної частини |
| part\_N | \_matrix | Виділяє частину непорівнюваності | Матрицю частини непорівнюваності |

Файл calculations.py

|  |  |  |  |
| --- | --- | --- | --- |
| Ф-ція/Метод | Параметри | Опис | Значення,  що повертає |
| get\_sigmas | matrix | Шукає вектор сігма для кожної пари альтернатив | Матрицю векторів сігма |

Файл file\_manager.py

|  |  |  |  |
| --- | --- | --- | --- |
| Ф-ція/Метод | Параметри | Опис | Значення,  що повертає |
| get\_input | - | Читає з текстового файлу початкові файлі | Початкову матрицю |
| write\_number | number | Записує у текстовий файл число | - |
| write\_matrix | matrix | Записує у текстовий файл матрицю | - |

**Висновки**

Були досліджені відношення Парето, мажоритарне відношення, лексикографічне відношення, відношення Березовського та відношення Подиновського та проведена оптимізація цих відношень. Відношення Парето використовується також у знаходженні відношень Подиновського та Березовського.

Відношення Березовського є найбільш складним за кількістю операцій, так як потрібно декілька разів проходитися по матриці, щоб виділити по класам альтернативи, а саме окремо симетричну, асиметичну частину та частину непорівнюванності, а потім проводити порівняння отриманих матриць з різних класів.